

Leard de Oliveira Fernandes
Luan Diego de Lima Pereira
Marcelo Mendonça dos Santos
(Orgs.)



embarca tech

Uma rede de formação tecnológica
no interior da Bahia

EmbarcaTech:

**uma rede de formação tecnológica no
interior da Bahia**



Pedro & João
editores

**Leard de Oliveira Fernandes
Luan Diego de Lima Pereira
Marcelo Mendonça dos Santos
(Organizadores)**

**EmbarcaTech:
uma rede de formação tecnológica no
interior da Bahia**

Copyright © Autoras e autores

Todos os direitos garantidos. Qualquer parte desta obra pode ser reproduzida, transmitida ou arquivada desde que levados em conta os direitos das autoras e dos autores.

Leard de Oliveira Fernandes; Luan Diego de Lima Pereira; Marcelo Mendonça dos Santos [Orgs.]

EmbarcaTech: uma rede de formação tecnológica no interior da Bahia. São Carlos: Pedro & João Editores, 2026. 185p. 16 x 23 cm.

**ISBN: 978-65-265-2910-2 [Impresso]
978-65-265-2911-9 [Digital]**

1. Sistemas embarcados. 2. Internet das Coisas. 3. Field-Programmable Gate Arrays (FPGA). 4. Aprendizagem baseada em projetos. 5. Residência Tecnológica. I. Título.

CDD – 370

Capa: Marcos Della Porta

Ficha Catalográfica: Hélio Márcio Pajeú – CRB – 8-8828

Revisão: Lourdes Kaminski

Diagramação: Diany Akiko Lee

Editor: Valdemir Miotello

Diretores executivos: Pedro Amaro de Moura Brito & João Rodrigo de Moura Brito

Conselho Editorial da Pedro & João Editores:

Augusto Ponzio (Bari/Itália); João Wanderley Geraldi (Unicamp/Brasil); Hélio Márcio Pajeú (UFPE/Brasil); Maria Isabel de Moura (UFSCar/Brasil); Maria da Piedade Resende da Costa (UFSCar/Brasil); Valdemir Miotello (UFSCar/Brasil); Ana Cláudia Bortolozzi (UNESP/Bauru/Brasil); Mariangela Lima de Almeida (UFES/Brasil); José Kuiava (UNIOESTE/Brasil); Marisol Barenco de Mello (UFF/Brasil); Camila Caracelli Scherma (UFFS/Brasil); Luís Fernando Soares Zuin (USP/Brasil); Ana Patrícia da Silva (UERJ/Brasil).



Pedro & João Editores

www.pedroejoaoeditores.com.br

13568-878 – São Carlos – SP

2026

Sumário

Introdução	11
Leard de Oliveira Fernandes	
Coordenação acadêmica e arquitetura formativa do programa .	17
Leard de Oliveira Fernandes	
Das aulas à prática: o papel dos instrutores	23
Reinaldo Götz de Oliveira Junior	
Ricardo Menezes Prates	
Da formulação dos desafios à validação dos protótipos	37
Luan Diego de Lima Pereira	
Marcelo Mendonça dos Santos	
Polo Bom Jesus da Lapa.....	41
Manoel Messias Silva Júnior	
Elias Guimarães Miranda Barbosa da Silva	
Projeto 1: AGV com Comando de Voz para Tarefas de Logística Interna	46
Bruna Alves de Barros	
Eder Renato da Silva Cardoso Casar	
Mariana Farias da Silva	
Matheus Pereira Alves	
Projeto 2: IR REMOTE - Controle Universal Smart para Ar-condicionado	51
Jadson de Jesus Santos	
Mateus Moreira da Silva	
Rafael Souza Barbosa	
Sauan de Souza Santos	

Projeto 3: Desenvolvimento de Estação de Monitoramento Ambiental com Controle Automatizado	55
João Pedro Soares Racolto	
Nivaldo Rodrigues da Silva Júnior	
Rian de Sena Mendes	
Samuel Guedes Canário	
Projeto 4: Implantação de Sistema RFID para Gerenciamento de Produção	61
Atenilton Santos de Souza Júnior	
Daniel Silva de Souza	
José Vinicius Rodrigues Soares	
Ronaldo Cesar Santos Rocha	
Luiz Eduardo Pereira Gomes	
Polo Feira de Santana.....	67
Ramón Luz Lemos Santos	
João Paulo Just Peixoto	
Projeto 1: Sistema cartesiano de aplicação de cola.....	72
Camila Queiroz Boa Morte Pereira	
Davi Oliveira Santana Carvalho	
Hilquias Rodrigues de Oliveira	
Mateus Fernandes Santos	
Ronaldo Ruiz Filho	
Projeto 2: Verificador de Funcionamento de Queimadores em Cooktop	77
Camila de Araújo Bastos	
Jackson Silva dos Santos	
Lucas Carneiro de Araújo Lima	
Luis Felipe Pereira de Carvalho	
Mariana da Silva Lima Santos	
Projeto 3: Parafusadeira CNC.....	83
Caique de Brito Freitas	
Cláudio Evaristo Júnior	
Davi Leão	
Hugo Martins Santana	
Naylane do Nascimento Ribeiro	

Projeto 4: <i>SafeWay</i> - Sistema Embarcado para Controle de Acesso com <i>RFID</i> e <i>Cache Offline</i>.....	88
Anna Beatriz S. Lima	
Carlos Henrique O. Valadão	
Daniel P. Braz	
Lorenzo Giuseppe O. Baroni	
Elmer C. de Oliveira Filho	
Polo Ilhéus/Itabuna.....	93
Aline Silva Ramos	
Fábio Oliveira Silva	
Projeto 1: <i>Conectsim</i> - Comunicação <i>NB-IOT</i> com módulo <i>SIM7000G</i>.....	98
Ariel Pina Ribeiro	
Davi Dantas Mendez Sanchez	
Emyle Santana da Silva	
João Vitor de Siqueira Santana Costa	
Projeto 2: Desenvolvimento de Manipulador Robótico para Automação de Paletização Logística.....	103
Gabriel Marques de Andrade	
Jabson Gama Santana Júnior	
Leonam Sousa Rabelo	
Lucas Gabriel Ferreira	
Projeto 3: Dispositivo para Coleta de Dados Industriais (<i>IIoT Edge Device</i>).....	108
Henrique Oliveira dos Santos	
Levi Silva Freitas	
Rodrigo Almeida Pirôpo	
Projeto 4: Transformando Eletrodomésticos <i>Smart</i> com o <i>Tuya</i>	114
Matheus Santos Silva	
Juan Pablo Azevedo Sousa	
Arthur de Oliveira Moreira	
Yuri Coutinho Costa	
Leonardo Bonifácio Vieira Santos	

Polo Juazeiro.....	119
Giovanni Antherrely Lima da Silva	
Wedson Pereira da Silva	
Projeto 1: Mecanismo XYZ para Testes Automatizados em Teclados.....	124
Roberto Vítor Lima Gomes Rodrigues	
Gabriel Shiva	
Andressa Sousa	
Pedro Lucas	
Projeto 2: <i>ApiSSense</i>	130
Gabriel Cosmo	
Taylan Mayckon	
Victor Gabriel	
Projeto 3: Desenvolvendo a Inteligência do Futuro: Sistemas Embarcados para Eficiência Energética.....	135
Ariagildo Souza Lins	
Guilherme Miller Gama Cardoso	
Maic de Oliveira Santos	
Richard Lima Ribeiro	
Projeto 4: Sistema de Coleta e Envio de Dados de Temperatura de Sistemas de Refrigeração Usando <i>Raspberry Pi Pico</i> e Protocolo RS485	140
João Felipe Teles Monte	
Marco Antonio de Jesus Saturnino	
Pedro Henrique Carvalho Felix	
Thalles Inácio Araújo	
Polo Vitória da Conquista.....	145
Auerê Vasconcelos Veras	
Cléia Santos Libarino	
Projeto 1: Controle XYZ: Automação de <i>Warehouses</i>	149
Carlos Henrique	
Guilherme Aguiar Silva	
Maria Luiza Brasil Medrado	
Muriel da Costa Santos	

Projeto 2: Fonte Inteligente de Backup para Sistema de Monitoramento	153
Luiz Filipe Ribeiro	
Heitor Rodrigues Lemos Dias	
Paulo César de Jesus Di Lauro	
Roberto Souza Cardoso	
Projeto 3: Inovação em Sistemas Embarcados: Módulo Inteligente para Diagnóstico e Carga de Baterias	157
Aulo Cezar Ferreira dos Anjos Filho	
Leonardo Romão Queiroz Araújo	
Marcus Vinicius Magalhães França Filho	
Matheus Nepomuceno Souza	
Projeto 4: Sistema de Controle AGV com Dashboard em Tempo Real.....	161
Jeová Cosme de Jesus Pinheiro	
Jonas Souza Pinto	
Leonardo Rodrigues Luz	
Wilton Lacerda Silva Jr.	
Trilha FPGA.....	165
Amanda Costa Martinez	
Pedro Henrique Andrade Trindade	
Projetos Desenvolvidos ao Longo da Trilha	169
Camila Ramos Gomes	
Daniel Alencar Penha Carvalho	
Gabriel Cavalcanti Coelho	
Gustavo Oliveira Alves	
Jacson Souza dos Santos	
Kaian de Souza Gonçalves	
Lucas Almeida Pereira	
Lucas Fialho Xavier	
Luiz Felipe do Rosário Alves Silva	
Maryana Souza Silveira	
Matheus Santos Souza	
Theógenes Gabriel Araújo de Andrade	
Thiago Roberto de Lima Ribeiro	
Vitor Coutinho Lima	

Yago Rodrigo Morato Marques Guirra
Yohanan Santos de Santana

Introdução

Leard de Oliveira Fernandes
(Coordenador Acadêmico)

Este livro foi concebido como registro da experiência do *EmbarcaTech* na Bahia. Seu ponto de partida não é apenas a apresentação de projetos concluídos, mas a documentação de um processo de formação tecnológica organizado em rede, distribuído territorialmente e executado a partir de uma lógica de capacitação, mentoria, acompanhamento e desenvolvimento aplicado.

Trata-se de um livro voltado a organizar, em uma única narrativa, diferentes camadas de uma mesma experiência: a estrutura do programa, a coordenação acadêmica, o trabalho dos instrutores, a formulação dos desafios, a atuação dos polos e a singularidade das trilhas desenvolvidas. A obra, portanto, não deve ser lida como simples reunião de relatos isolados, mas como documento de uma experiência coletiva de formação em sistemas embarcados no interior da Bahia.

O *EmbarcaTech* na Bahia foi, ao mesmo tempo, programa de capacitação, residência tecnológica, experiência de organização em rede e exercício de formação distribuída em sistemas embarcados e *FPGA*. O livro foi estruturado para registrar esse processo em suas diferentes escalas: a escala institucional do projeto, a escala acadêmica do percurso formativo, a escala territorial dos polos e a escala técnica dos projetos desenvolvidos. Assim, a leitura que se segue deve ser entendida menos como catálogo de resultados e mais como documentação de uma experiência de formação tecnológica construída coletivamente no interior da Bahia.

Contextualização do *EmbarcaTech* na Bahia

No âmbito institucional, o programa executado na Bahia integra o projeto Residência em TIC 37 – Sistemas Embarcados, desenvolvido com apoio do Ministério da Ciência, Tecnologia e Inovações, no contexto dos Programas e Projetos Prioritários em TIC, com coordenação da Softex (Associação para Promoção da Excelência do Software Brasileiro) e execução estadual pelo CEPEDI (Centro de Pesquisa, Desenvolvimento Tecnológico e Inovação).

O projeto é um programa de capacitação profissional técnica em tecnologias de sistemas embarcados, articulado a uma residência tecnológica voltada à prática do conhecimento adquirido e ao desenvolvimento de estudos de caso, funcionalidades e soluções para problemas em aplicações de Internet das Coisas, incluindo áreas como educação, segurança, indústria e saúde. Entre os objetivos específicos do projeto estão o aprofundamento da capacitação, a aplicação prática do conhecimento, a disseminação de resultados no meio acadêmico e a criação de uma rede de pesquisa em sistemas embarcados.

Na Bahia, essa proposta foi territorializada em cinco polos do interior: Ilhéus/Itabuna, Vitória da Conquista, Bom Jesus da Lapa, Feira de Santana e Juazeiro. A distribuição dos polos deu ao programa uma configuração descentralizada e exigiu uma forma de organização acadêmica e técnica capaz de manter unidade de objetivos, critérios de acompanhamento e coerência formativa, mesmo diante de realidades locais distintas.

Visão geral das fases do programa

No recorte baiano, o edital estruturou o programa em duas fases complementares. A Fase 1, denominada Capacitação Tecnológica, foi organizada como formação de três meses, em formato online, com atividades síncronas e assíncronas. A Fase 2, denominada Residência Tecnológica, foi prevista como etapa

imersiva de doze meses, voltada ao aprofundamento e à aplicação dos conhecimentos na construção de soluções em sistemas embarcados ou *FPGA* para *IoT*. Para a execução na Bahia, foram previstas 1.000 vagas na primeira fase e a seleção de 100 residentes para a fase seguinte.

Ao final da Fase 1, 508 participantes cumpriram os critérios de aprovação, que incluíam aproveitamento mínimo nas atividades avaliativas e frequência mínima. Na Fase 2, após convocação dos 150 melhores alunos para entrevista, foram selecionados 100 residentes, distribuídos em 20 vagas por polo.

Estrutura em trilhas e polos

No *EmbarcaTech* Bahia, foram realizadas duas formas de organização: uma territorial, por polos, e uma temática, por trilhas. Os polos permitiram distribuir a execução pelo interior do estado, aproximando o programa de diferentes contextos locais. As trilhas, por sua vez, permitiram diferenciar os percursos técnicos dentro da residência, especialmente entre *Software* Embarcado e *FPGA*, respeitando conteúdos, ferramentas e dinâmicas de projeto distintas. Essa combinação entre territorialização e especialização técnica é uma das principais marcas do *EmbarcaTech* na Bahia.

Na residência, a trilha de *Software* Embarcado contou com instrutores e mentores distribuídos pelos polos, enquanto a trilha de *FPGA* foi organizada com equipe própria e com uma dinâmica distinta, marcada por um projeto integrador coletivo. Foram alocados 84 residentes em *Software* Embarcado e 16 na trilha de *FPGA*.

Propósito formativo e extensionista da obra

O propósito deste livro é formativo em dois sentidos. Primeiro, porque ele organiza a memória técnica e pedagógica do programa, registrando como foram concebidos e executados seus percursos, suas mediações e seus resultados. Segundo, porque ele transforma a experiência vivida em material para leitura, análise e

eventual reaplicação em outras iniciativas de formação tecnológica. Ao registrar processos, dificuldades, ajustes e soluções, o livro preserva informação útil não apenas para quem participou do programa, mas também para quem deseja compreender como uma formação distribuída em sistemas embarcados pode ser estruturada num contexto territorial ampliado.

O sentido extensionista da obra aparece na própria forma de implantação do programa. A execução em cinco polos do interior, o atendimento a públicos com diferentes níveis de formação, a articulação entre formação técnica e problemas aplicados e a construção de uma rede entre coordenação, instrutores, mentores, estudantes e instituições parceiras mostram que a experiência ultrapassou a lógica de oferta concentrada em um único centro. O *EmbarcaTech*, na Bahia, foi operado como ação de formação em rede, com presença territorial, circulação de conteúdo, interação entre equipes e desenvolvimento de projetos vinculados a demandas técnicas de empresas ou pesquisa.

Por essa razão, a obra não se limita à descrição dos protótipos finais. Os projetos são parte importante do resultado, mas não esgotam o significado da experiência. O que este livro procura tornar visível é o percurso que levou da abertura do programa à organização dos conteúdos, da formação inicial à residência, da orientação remota aos encontros presenciais, da definição dos desafios ao estágio de prototipagem, e da atuação dos mentores ao amadurecimento técnico dos grupos.

Organização dos capítulos

A estrutura do livro foi desenhada para acompanhar a lógica do Programa *EmbarcaTech* na Bahia. Após esta introdução, o **Capítulo 2** trata da coordenação acadêmica, explicando a concepção didática do programa, a organização das trilhas, o perfil formativo buscado e a articulação entre conteúdo, prática, mentoria e projeto. O **Capítulo 3** é dedicado aos instrutores e ao desenho das aulas. O **Capítulo 4** aborda o caminho dos desafios técnicos aos

protótipos, mostrando a relação entre problema, escopo, mentoria e validação.

Na sequência, os **Capítulos 5 a 9** são reservados aos polos de Bom Jesus da Lapa, Feira de Santana, Ilhéus/Itabuna, Juazeiro e Vitória da Conquista. Em cada um deles, a abertura dos mentores contextualiza o polo, o acompanhamento realizado e a evolução observada nos estudantes, enquanto os grupos registram a trajetória de seus projetos, as soluções desenvolvidas, os resultados alcançados, os desafios enfrentados e as perspectivas de continuidade. O **Capítulo 10**, por fim, trata da trilha *FPGA*, cuja configuração integrada exige uma narrativa própria.

Agradecimentos

Este projeto foi apoiado pelo Ministério da Ciência, Tecnologia e Inovações, com recursos da Lei nº 8.248, de 23 de outubro de 1991, no âmbito do PPI-SOFTEX, coordenado pela Softex e publicado Residência em TIC 37, DOU 01245.001849/2024-76.

Coordenação acadêmica e arquitetura formativa do programa

Leard de Oliveira Fernandes
(Coordenador Acadêmico)

A coordenação acadêmica do *EmbarcaTech*, executado pelo CEPEDI na Bahia, operou sobre um programa concebido em duas fases com funções complementares. A Fase 1 foi estruturada como capacitação tecnológica de três meses, em formato online, com atividades síncronas e assíncronas. A Fase 2 foi organizada como residência tecnológica de doze meses, voltada ao aprofundamento e à aplicação dos conhecimentos na construção de soluções em sistemas embarcados ou *FPGA* (*Field-Programmable Gate Array*) para *IoT* (*Internet of Things*). Para execução na Bahia, foram previstas 1.000 vagas distribuídas em cinco polos no estado e, ao término da primeira etapa, a seleção de 100 residentes para a fase avançada.

Esse desenho em duas fases não representou apenas uma divisão administrativa, mas também a lógica de formação do programa. A primeira etapa teve função de base, nivelamento e identificação de aderência técnica e operacional dos participantes. A segunda etapa teve função de aprofundamento, aplicação orientada e consolidação de competências em projetos aplicados. Em outras palavras, foi necessário trabalhar simultaneamente com escala e com progressão: primeiro, oferecer formação suficientemente ampla para um público numeroso e heterogêneo; depois, reorganizar esse conjunto em trilhas de maior densidade técnica, com acompanhamento mais próximo, metas de entrega e produção aplicada.

No caso da Bahia, a descentralização territorial foi elemento estruturante do modelo acadêmico. A execução ocorreu a partir dos polos de Ilhéus/Itabuna, Vitória da Conquista, Juazeiro, Feira de Santana e Bom Jesus da Lapa. Essa distribuição exigiu uma coordenação capaz de padronizar diretrizes, materiais, instrumentos de avaliação e critérios de acompanhamento, sem perder de vista as diferenças de infraestrutura, disponibilidade local e perfil dos estudantes em cada cidade. Assim, além da adequação das ementas, foi necessário transformar um programa estadual distribuído em um percurso formativo coerente, comparável entre polos e operacionalmente executável.

A estrutura de gestão apresentada no material de lançamento já indicava essa separação de responsabilidades. O programa foi organizado com comitê acadêmico, coordenação pedagógica, coordenações de curso, coordenações de residência, instrutores, mentores e designer instrucional. Para a coordenação acadêmica, isso significou trabalhar em uma camada intermediária entre formulação pedagógica e execução técnica: definir a lógica de execução do programa, organizar a relação entre conteúdo e prática, alinhar o trabalho dos instrutores, estruturar a atuação dos mentores e garantir que a avaliação medisse a aprendizagem e não apenas presença ou cumprimento de tarefas.

A primeira consequência prática desse arranjo foi a adoção de uma estrutura que combinou ensino remoto, acompanhamento contínuo e organização modular do conteúdo. Na capacitação inicial, o público atendido foi amplo, incluindo estudantes do Ensino Médio, Ensino Técnico, Ensino Superior, Pós-graduação e egressos com diploma em áreas relacionadas. Esse perfil diverso ampliou o alcance do programa, mas também impôs um problema pedagógico central: a heterogeneidade de formação na entrada. A coordenação acadêmica precisou, portanto, partir de uma arquitetura didática capaz de acomodar diferentes níveis de conhecimento prévio sem descaracterizar o conteúdo técnico do programa.

Para responder à heterogeneidade da Fase 1, o conjunto de 1.000 alunos foi inicialmente organizado em 10 turmas de 100

participantes, com um mentor responsável por cada turma. Em seguida, cada turma foi subdividida em 10 subgrupos, cada um com um líder, buscando-se uma composição tão heterogênea quanto possível, por meio de distribuição aleatória quanto a sexo, polo de origem e nível de formação, de modo a reunir, em cada turma, participantes dos cinco polos e de diferentes perfis educacionais, incluindo ensino médio, técnico e graduação. Além dessa organização interna, cada um dos 10 subgrupos podia realizar até dois encontros remotos por semana com seu respectivo mentor. Essa organização permitiu aproximar os participantes e a equipe de acompanhamento, melhorar o fluxo de atendimento das dúvidas, diminuir o volume de correções centralizadas e criar um arranjo mais favorável ao engajamento. Além disso, foram incorporados encontros síncronos, palestras e o projeto final.

A proposta educacional da residência aprofundou essa mesma lógica. O documento de referência previa doze meses de residência, com dedicação contínua de vinte horas semanais, e uma capacitação avançada de 240 horas ao longo dos seis primeiros meses, articulando aulas online síncronas, estudo assíncrono, mentorias presenciais ou remotas e *workshops* práticos. Na execução efetiva, a residência foi mantida como percurso formativo paralelo ao desenvolvimento dos projetos, e não como mera alocação de bolsistas em tarefas isoladas. Assim, a Fase 2 foi ajustada como uma formação avançada em conjunto com a execução dos projetos, e não como simples estágio de execução.

Nesse contexto, a divisão em trilhas foi uma consequência natural da maturação do programa. O edital já previa que os residentes poderiam ser alocados em uma de duas trilhas de aprendizado, associadas aos domínios de *software* embarcado e *FPGA*. Na execução da Fase 2, a trilha de *Software Embarcado* foi iniciada com 84 residentes, enquanto a trilha de *FPGA* foi iniciada com 16. Essa distribuição, por si só, já indicava a necessidade de estratégias acadêmicas distintas. A trilha de *Software Embarcado* exigiu maior capilaridade temática e maior acomodação de perfis; a trilha de *FPGA*, por sua vez, demandou progressão conceitual

mais controlada, materiais mais estruturados e aprofundamento em uma cadeia ferramental mais específica.

Na trilha de *Software Embarcado*, a lógica de formação adotada foi explicitamente bifurcada em dois eixos complementares. Um eixo concentrou programação embarcada, conectividade, *IoT*, *SBC* e inteligência computacional. Outro eixo concentrou eletrônica aplicada, sistemas de tempo real, sensores, atuadores, robustez e integração *hardware-software*. Essa divisão evitou que a trilha fosse reduzida a uma formação exclusivamente em *firmware* ou, no extremo oposto, a uma abordagem apenas instrumental de prototipagem. A articulação entre os dois eixos permitiu que o residente transitasse da programação e da comunicação para a integração *hardware-software*.

O desenho metodológico dessa trilha foi orientado por Aprendizagem Baseada em Projetos (*PBL*), com adaptação às demandas reais dos grupos. A *BitDogLab* foi utilizada como plataforma comum de nivelamento, enquanto a *SBC Labrador* passou a sustentar atividades de maior complexidade. A metodologia incluiu fichas de tarefa, vídeos, repositórios *GitHub* e marcos contínuos de entrega. Do ponto de vista da coordenação acadêmica, essa escolha foi relevante por três motivos: padronizou o ambiente mínimo de desenvolvimento, aumentou a rastreabilidade da aprendizagem e aproximou o processo formativo das práticas usuais de documentação e versionamento em engenharia.

Nesse contexto, a *BitDogLab* foi adotada como ferramenta de desenvolvimento didática de referência para a etapa inicial da trilha de *Software Embarcado*. Sendo este um projeto de *hardware open source* voltado ao ensino de sistemas embarcados, programação e eletrônica, baseado nas placas *Raspberry Pi Pico*, com integração de componentes embarcados na própria placa e possibilidade de conexão organizada de periféricos externos. Do ponto de vista acadêmico, essa característica permitiu utilizar uma mesma base material para atividades de programação de microcontroladores, estudo de entradas e saídas digitais, *PWM*,

integração de sensores e atuadores e experimentação inicial de comunicação com dispositivos externos.

No início da residência, a trilha concentrou acolhimento, revisão de conteúdos anteriores e competências comportamentais, como inteligência emocional, gestão do estresse, trabalho em equipe, resolução de conflitos e comunicação interpessoal. Na sequência, o conteúdo avançou para multitarefa em sistemas embarcados, comunicação sem fio para *IoT*, protocolos de comunicação, aplicações com microcontroladores nas plataformas *BitDogLab* e *Labrador*, desenvolvimento de sensores e atuadores, segurança em *IoT*, integração com *hardware* e, posteriormente, redes neurais profundas, redes neurais convolucionais e *TinyML*.

A trilha de *FPGA* foi iniciada com os fundamentos de sistemas digitais, incluindo sistemas de numeração, portas lógicas, lógica combinacional, *latch*, *flip-flop*, lógica sequencial e máquinas de estados finitas. Em seguida, foram introduzidas ferramentas como *DigitalJS*, *Visual Studio Code*, *Icarus Verilog* e *AMD Vivado*, além de temas como *Verilog*, *SystemVerilog*, protocolos *UART (Universal Asynchronous Receiver/Transmitter)*, *SPI (Serial Peripheral Interface)* e *I2C (Inter-Integrated Circuit)*, serialização, *FIFO (First In, First Out)*, temporização, uso da plataforma *Colorlight i9+*, escrita de *constraints*, *MicroBlaze*, *AMD Vitis* e integração com periféricos como *ADS1115*, *BMP280* e *MPU6050*. Assim, a trilha de *FPGA* avançou de forma incremental, com ênfase em modelagem de *hardware*, verificação, síntese e integração.

A própria avaliação da mentoria de *FPGA* ajudou a justificar essa escolha. Entre os respondentes, 92,3% declararam não ter contato prévio com *FPGA* e apenas 7,7% relataram experiência profissional anterior na área. A coordenação acadêmica, portanto, não poderia presumir familiaridade dos alunos com *HDL (Hardware Description Language)*, *toolchain* de síntese ou depuração temporal. O modelo gradual, com materiais estruturados, nivelamento inicial e aumento progressivo da complexidade, foi uma decisão pedagógica decorrente do perfil dos alunos da trilha de *FPGA*.

Também na trilha de *FPGA*, os exemplos de mentoria exigiram uma arquitetura completa, com implementação em *HDL*, uso de *testbench*, documentação técnica, repositório *Git* e apresentação das limitações e dos resultados. Em um dos projetos, os alunos desenvolveram um sistema de processamento de imagens com comunicação *UART* em *FPGA* sem uso de *IP Cores* (*Semiconductor Intellectual Property Cores*), o que obrigou o grupo a tratar interface, *buffers*, sincronização, modularização e verificação. Esse tipo de desenho reforça o perfil de egresso pretendido: um residente capaz de compreender a estrutura interna da solução e não apenas reproduzir blocos prontos.

A avaliação final da residência forneceu indícios de que o perfil de egresso foi, em boa medida, atingido. Os questionários apontam aumento de prontidão profissional, ganhos em programação embarcada, *IoT* e protocolos, trabalho em equipe, gestão de projeto, documentação técnica, *RTOS* (*Real-Time Operating System*), testes e validação. Entre os resultados mencionados pelos residentes, aparecem protótipos desenvolvidos e validados em laboratório, provas de conceito apresentadas a parceiros, documentação técnica e demonstrações finais.

Dessa forma, a lógica de formação adotada apresenta uma progressão por complexidade, com plataformas padronizadas, uma mentoria próxima e avaliação contínua. Enquanto o desenho da capacitação buscou manter vínculo entre o fundamento, a prática e a solução desenvolvida. Para um programa nacional executado em recorte estadual, distribuído em polos e voltado a alunos com diferentes perfis de entrada, esse arranjo mostrou consistência pedagógica e aderência ao objetivo de formar pessoal para as áreas de sistemas embarcados e *FPGA*.

Como resultado, a coordenação acadêmica consolidou uma base metodológica que pode orientar novas ofertas do programa, com ajustes pontuais em cronograma, comunicação e previsibilidade operacional, mas preservando a lógica central de formação integrada entre conteúdo, mentoria e projeto aplicado.

Das aulas à prática: o papel dos instrutores

Reinaldo Götz de Oliveira Junior
(Instrutor)

Ricardo Menezes Prates
(Instrutor)

1. Introdução

A estruturação de cursos e capacitações tecnológicas direcionadas a sistemas embarcados e Internet das Coisas (*IoT*) exige uma abordagem pedagógica que vai além da organização linear de conteúdo. Trata-se de um domínio tecnológico emergente caracterizado pela integração entre *hardware*, *software*, comunicação e processamento digital de informações, no qual o aprendizado efetivo depende da articulação entre teoria, prática, experimentação e atualização contínua.

Nesse contexto, o curso desenvolvido no âmbito do Projeto *EmbarcaTech* foi concebido como uma trajetória formativa progressiva, estruturada em encontros síncronos semanais e orientada ao desenvolvimento de soluções aplicadas, alinhadas às demandas tecnológicas e práticas do mercado. O planejamento das atividades contemplou uma sequência de encontros, iniciando com fundamentos de programação, eletrônica básica e sistemas microcontrolados e evoluindo, de forma estruturada, até a concepção e implementação de tecnologias *IoT* e modelos avançados de inteligência artificial embarcada. Ao longo da formação, também foram abordados tópicos relacionados à integração de sensores e atuadores, armazenamento de dados,

processamento multicore e aplicações avançadas envolvendo RFID, NFC, depuração de sistemas embarcados e geração de sinais digitais de vídeo no RP2040. Mais do que uma sequência de aulas, o curso de capacitação e a residência em *software* embarcado constituíram um ambiente formativo dinâmico, no qual conteúdos, estratégias pedagógicas e práticas de ensino foram continuamente ajustados ao longo da experiência, em função das demandas técnicas e da evolução dos estudantes.

2. Papel dos Instrutores: Mediação, Adaptação e auxílio na Mentoria

O papel dos instrutores foi determinante na condução do processo formativo, assumindo diferentes funções ao longo do curso. Na fase inicial, sua atuação concentrou-se na mediação do conhecimento, envolvendo a organização estruturada dos conteúdos, a condução das aulas e o suporte técnico contínuo aos estudantes e mentores. Essa atuação mostrou-se especialmente relevante nas etapas iniciais, que demandaram acompanhamento intensivo na configuração do ambiente de desenvolvimento, na programação em linguagem C para microcontroladores, bem como na simulação e manipulação de periféricos do microcontrolador RP2040 e da plataforma educacional *BitDogLab*.

À medida que os conteúdos avançaram e as atividades e projetos ganharam maior complexidade, os instrutores passaram a atuar de forma mais estratégica, orientando a adaptação dos conteúdos às demandas técnicas emergentes dos residentes. Nesse estágio, a atuação dos instrutores se concentrou na orientação de decisões técnicas, na contextualização dos conteúdos às necessidades específicas de cada projeto e no acompanhamento sistemático do desenvolvimento dos protótipos.

Destaca-se, nesse contexto, a crescente relevância das demandas relacionadas à Inteligência Artificial (IA), incorporadas progressivamente aos projetos desenvolvidos. Em resposta a esse movimento, os conteúdos associados à IA foram intensificados nos

estágios finais do curso de residência, permitindo aos estudantes explorar desde conceitos fundamentais até a implementação de modelos inteligentes aplicados em sistemas embarcados.

Na etapa final do curso de residência, destacou-se também a interação direta dos residentes com os *POs* (*Product Owners*) das empresas parceiras, o que contribuiu para uma aproximação mais estreita entre o ambiente formativo e as demandas reais do setor produtivo. Estas atividades, caracterizadas por requisitos técnicos específicos e cenários de aplicação concretos, foram compartilhadas com instrutores e mentores, permitindo uma melhor distribuição das responsabilidades de acompanhamento e suporte ao aprendizado. Essa dinâmica colaborativa favoreceu ajustes contínuos na condução das aulas, bem como no direcionamento dos projetos desenvolvidos, assegurando maior aderência às expectativas das empresas parceiras e promoção de uma formação mais alinhada às práticas contemporâneas de projetos de sistemas embarcados.

3. Concepção e Desenho das Aulas

O desenho das aulas foi estruturado de forma progressiva e modular, permitindo uma evolução consistente do conhecimento teórico e prático de sistemas embarcados. A sequência didática iniciou com fundamentos essenciais — como conceitos básicos de eletrônica, programação em linguagem *C*, *GPIO*, temporização e sistemas operacionais de tempo real — e avançou gradualmente para tópicos mais complexos, incluindo multitarefa com *FreeRTOS*, sincronização e controle de concorrência, integração de sensores e atuadores, comunicação *IoT*, protocolos de rede, segurança na informação e inteligência computacional.

A partir da metade do curso, observa-se uma transição consistente no desenho pedagógico: os conteúdos passam a ser organizados não apenas pela temática e complexidade técnica, mas por sua capacidade de integração em sistemas completos. Tecnologias como *MQTT*, *LoRa*, *Bluetooth* de Baixa Energia (*BLE*) e

redes *LoRaWAN* são introduzidas em conjunto com aplicações práticas, permitindo ao estudante compreender o funcionamento de sistemas reais complexos e distribuídos.

Nos estágios finais, o curso de residência incorpora conteúdos avançados associados à integração de hardware e software de baixo nível, depuração de sistemas embarcados, paralelismo computacional e de aprendizagem de máquina para sistemas embarcados, incluindo *TinyML*, Redes Neurais Convolucionais (CNN), Redes Neurais Recorrentes (RNN) e modelos generativos, a exemplo das redes Generativas Adversárias (GANs) e Redes de Difusão. Essa evolução culmina na construção de uma visão sistêmica, na qual dispositivos embarcados, comunicação e inteligência computacional são tratados como partes de um mesmo ecossistema tecnológico.

4. Estratégias Pedagógicas Adotadas

Parte da metodologia do curso foi fundamentada na **Aprendizagem Baseada em Projetos (PBL)**, estruturada de forma adaptativa para atender à complexidade de grupos de trabalho atuando em projetos distintos. Essa abordagem permitiu transformar o ambiente de ensino em um espaço de desenvolvimento formativo e tecnológico, no qual os estudantes atuam como agentes ativos na construção de soluções.

Entre as principais estratégias pedagógicas adotadas, destacam-se:

- **Centralidade da prática**, com integração contínua entre teoria e experimentação;
- **Progressão incremental de complexidade**, garantindo aprofundamento técnico;
- **PBL multidisciplinar**, orientado a problemas reais e aplicações de mercado;
- **Instrução e mentoria especializada**, promovendo acompanhamento individualizado;

- **Flexibilização dos conteúdos**, permitindo adaptação às demandas específicas dos projetos.

Adicionalmente, a introdução de plataformas de desenvolvimento, a exemplo da *BitDogLab* e da SBC (*Single Board Computer*) Labrador, permitiu trabalhar diferentes níveis de complexidade computacional, desde programação *bare metal* (desenvolvimento diretamente no *firmware* do microcontrolador) até *hardware* embarcado com suporte a sistemas operacionais e execução de modelos de aprendizagem de máquina e tarefas de visão computacional e geração de dados.

5. Experiência com Aulas Remotas Semanais

A execução do curso em formato remoto síncrono, com encontros semanais, constituiu um elemento estruturante da dinâmica pedagógica. Esse modelo possibilitou a continuidade do acompanhamento, a organização modular dos conteúdos e a manutenção de um ritmo adequado e constante de aprendizagem.

Por outro lado, a modalidade remota também impôs desafios relevantes, como a heterogeneidade da infraestrutura dos estudantes e dificuldades na montagem e configuração de ambientes de desenvolvimento. Esses desafios exigiram adaptações contínuas na condução das aulas e reforçaram a importância do suporte técnico e das mentorias.

Nesse contexto, destacou-se o papel fundamental dos mentores presenciais em cada um dos polos, que atuaram como elementos-chave na mediação prática do processo formativo. Sua atuação possibilitou o acompanhamento direto das atividades experimentais, o suporte na configuração de *hardware e software*, bem como a orientação na execução de testes e validações dos protótipos. Essa presença local contribuiu significativamente para mitigar as limitações inerentes ao ensino remoto, promovendo maior efetividade na aprendizagem prática e o fortalecimento da integração entre teoria e aplicação no desenvolvimento dos projetos.

6. Integração entre Conteúdo Teórico e Desenvolvimento de Projetos

Um dos principais diferenciais do curso de residência em *software* embarcado foi a forte integração entre teoria e prática, materializada por meio do desenvolvimento de projetos ao longo de parte da trajetória formativa. Cada grupo ficou responsável por um projeto específico, com demandas próprias e requisitos técnicos distintos.

Essa abordagem permitiu que os estudantes aplicassem, de forma integrada, conhecimentos relacionados a:

- Programação embarcada;
- Sistemas operacionais de tempo real (RTOS);
- Multitarefa e sincronização de processos;
- Comunicação entre tarefas;
- Programação multicore;
- Comunicação IoT;
- Protocolos RFID/NFC;
- Integração de sensores e atuadores;
- Sensoriamento analógico e digital;
- Controle de motores e acionamentos;
- Aquisição, processamento e armazenamento de dados;
- Paralelismo computacional e DMA;
- Depuração e análise de sistemas embarcados;
- Sistemas tolerantes a falhas com Watchdog Timer;
- Segurança na informação;
- Sistemas embarcados conectados e resilientes;
- Aprendizagem de máquina embarcada.

A introdução de técnicas de aprendizagem de máquina e do uso da SBC Labrador possibilitaram ampliar o escopo dos projetos, incorporando funcionalidades inteligentes e aproximando o desenvolvimento de cenários reais de aplicação.

7. Principais Dificuldades Observadas

Durante a execução do curso, foram identificadas algumas dificuldades, que variaram ao longo das etapas. Entre as principais, destacam-se:

- Configuração inicial do ambiente de desenvolvimento – nivelamento dos discentes para configuração do *VS Code* e Ferramenta de desenvolvimento *Pico SDK*;
- Assimilação de conceitos de baixo nível;
- Integração de múltiplas tecnologias (*hardware*, comunicação e *software*);
- Configuração de protocolos de comunicação, a exemplo de *MQTT* e *LoRa*;
- Execução e otimização de modelos de aprendizagem de máquina em *hardware* restrito.

A identificação dessas dificuldades foi fundamental para o ajuste contínuo das estratégias pedagógicas e para o aprimoramento do suporte oferecido aos estudantes.

8. Evolução dos Estudantes

A evolução dos estudantes ao longo do curso foi progressiva e bastante consistente. Inicialmente, observou-se uma dependência maior de orientação direta, com foco na execução de tarefas específicas. Com o avanço das atividades e a introdução de projetos mais complexos, os estudantes passaram a demonstrar maior autonomia, capacidade de integração de conhecimentos e habilidade para resolver problemas de forma independente.

Essa evolução reflete não apenas a aquisição de conhecimentos técnicos, mas o desenvolvimento de competências mais amplas, relacionadas à prática da engenharia e desenvolvimento de sistemas complexos, a exemplo do pensamento crítico, tomada de decisão e da capacidade de adaptação.

9. Síntese dos Aprendizados Pedagógicos

A experiência formativa viabilizou a consolidação de importantes aprendizados técnicos e pedagógicos. Destaca-se a importância da prática (aprendizado baseado em projetos) como elemento central do processo de aprendizagem, especialmente em áreas que envolvem interação com sistemas tecnológicos. A progressão estruturada dos conteúdos mostrou-se essencial para garantir continuidade e reduzir a sobrecarga cognitiva.

A aprendizagem baseada em projetos revelou-se particularmente eficaz para promover engajamento e contextualização do conhecimento, enquanto a estrutura de mentoria demonstrou ser fundamental para lidar com a diversidade de demandas técnicas presentes no ambiente de desenvolvimento.

Por fim, a experiência evidenciou que o papel do instrutor deve evoluir ao longo do processo formativo, acompanhando a maturidade e desenvolvimento das competências técnicas dos estudantes.

10. Trilha FPGA

De forma concomitante ao percurso de sistemas embarcados, porém direcionada a uma turma distinta de residentes, foi estruturada uma trilha formativa voltada ao desenvolvimento de sistemas digitais em plataformas *FPGA* (*Field Programmable Gate Array*). Essa trilha foi concebida com o objetivo de introduzir os participantes ao projeto em *hardware*, enfatizando o uso de arquiteturas capazes de explorar paralelismo intrínseco e atender a requisitos de desempenho com baixa latência e execução determinística. Nesse contexto, a proposta pedagógica buscou promover a transição do pensamento algorítmico sequencial para uma abordagem orientada à descrição e organização estrutural de sistemas digitais.

O percurso dessa trilha iniciou-se com a consolidação dos conceitos essenciais de sistemas digitais e evoluiu para a aplicação desses conhecimentos em atividades de projeto, simulação e implementação de sistemas funcionais em dispositivos *FPGA*. A atuação do instrutor acompanhou a evolução dos residentes, partindo de uma condução mais próxima e orientada nas etapas iniciais — especialmente diante das dificuldades associadas às linguagens de descrição de *hardware* — e avançando gradualmente para uma abordagem centrada na leitura e interpretação de documentação técnica, bem como na tradução das demandas e requisitos de projeto em descrições formais em código. Esse movimento favoreceu o desenvolvimento da autonomia dos estudantes, especialmente na capacidade de compreender especificações, interpretar requisitos e implementá-los em sistemas digitais de maior complexidade.

11. Desenvolvimento de competências em HDL

O desenvolvimento das competências em *HDL* (*Hardware Description Language*) pautou-se inicialmente na fundamentação dos conceitos de sistemas digitais, abrangendo sistemas de numeração, funções de lógica combinacional e o comportamento de elementos de lógica sequencial, como *latches*, *flip-flops* e máquinas de estado finitas. A partir dessa base conceitual, avançou-se para a aplicação prática dos conhecimentos por meio do projeto e simulação de circuitos digitais, utilizando ferramentas computacionais que possibilitam a análise do comportamento lógico e temporal dos sistemas. Nesse estágio, os residentes foram introduzidos às linguagens de descrição de *hardware*, com ênfase em *Verilog* e *SystemVerilog*, explorando aspectos como tipos de dados, operadores lógicos e blocos procedurais.

A evolução da trilha foi marcada pela transição de módulos isolados para a concepção de sistemas digitais mais complexos e integrados. Inicialmente, os residentes dedicaram-se à implementação de blocos elementares, progredindo

posteriormente para o desenvolvimento de módulos mais elaborados, como máquinas de estado finitas, serializadores e desserializadores. Esses elementos serviram como base para a construção de arquiteturas compostas por múltiplos módulos interconectados, que interagem de forma coordenada.

Os residentes foram orientados à utilização do ambiente *Visual Studio Code*, com extensões voltadas à interpretação de *HDL*. Nesse contexto, desenvolveu-se uma prática sistemática de simulação, com ênfase na construção de *testbenches* e na análise de formas de onda, incorporando a validação funcional como etapa integrante do processo de projeto.

12. Implementação de Protocolos e Integração de Sistemas

A ampliação da complexidade dos sistemas desenvolvidos ocorreu com a introdução de protocolos de comunicação serial, como *UART*, *SPI* e *I2C*. Essa etapa exigiu dos residentes uma compreensão mais aprofundada dos aspectos temporais dos sistemas digitais, especialmente no que se refere à sincronização de sinais, organização de dados e controle de fluxo.

Os estudantes desenvolveram módulos serializadores, desserializadores e *buffers FIFO*, além de mecanismos de *handshaking* para controle da comunicação entre diferentes blocos. A metodologia priorizou a análise de diagramas de tempo e a interpretação de especificações técnicas, promovendo a tradução desses requisitos em implementações funcionais em *HDL*.

A validação prática foi realizada por meio da implementação desses módulos em plataformas reais, como a *GOWIN Tang Nano 9K*, permitindo a observação do comportamento dos sistemas em operação e a interação com dispositivos periféricos.

A evolução para sistemas mais complexos foi impulsionada pela utilização da plataforma *Colorlight i9+*, baseada em *FPGA* da família *Xilinx Artix™ 7*. As atividades envolveram desde a exploração do *hardware* e análise de sua documentação até o

mapeamento de pinos e configuração do dispositivo por meio das chamadas *constraints*.

Como etapa de integração, foi introduzido o uso do processador *soft AMD MicroBlaze™*. Os residentes realizaram sua configuração e síntese no ambiente *AMD Vivado™*, integrando posteriormente o desenvolvimento de *software* em linguagem C no ambiente *AMD Vitis™*. Essa abordagem possibilitou a implementação de sistemas híbridos, nos quais *hardware* e *software* operam de forma complementar, sendo sua validação realizada por meio da integração com sensores externos, utilizando interfaces de comunicação serial previamente implementadas e configuradas no sistema.

13. Processamento de Dados em Fluxo e Sistemas de Vídeo

Na etapa final da trilha, os residentes foram desafiados a desenvolver aplicações envolvendo processamento de dados em fluxo (*streaming*) e requisitos elevados de desempenho. O estudo do protocolo *HDMI (High-Definition Multimedia Interface)* introduziu conceitos relacionados à transmissão digital de vídeo, incluindo sincronização, temporização e estruturação de quadros. Nesse contexto, destacou-se a utilização da codificação *TMDS (Transition Minimized Differential Signaling)*, responsável por reduzir a interferência eletromagnética e garantir a integridade dos dados durante a transmissão em alta velocidade, por meio de um esquema de codificação que minimiza transições de sinal e balanceia o número de bits transmitidos.

A partir desses fundamentos, os estudantes desenvolveram módulos para geração de sinais de sincronismo horizontal e vertical, bem como para a criação de padrões de vídeo, considerando rigorosamente os requisitos de temporização associados à resolução *VGA*. Adicionalmente, foram abordados aspectos relacionados à serialização dos dados de vídeo e ao mapeamento dos sinais para as linhas diferenciais do padrão *HDMI*.

A ampliação do escopo dos projetos ocorreu com a integração do sensor de imagem *OV2640*. Os residentes implementaram a

captura e a organização dos dados no formato YUV422, estruturando o fluxo de *pixels* para etapas subsequentes de processamento. Inicialmente, foi realizada a reprodução da imagem em escala de cinza, o que possibilitou a validação do *pipeline* de captura e exibição por meio da interface *HDMI*.

Na sequência, foram incorporadas técnicas de processamento digital em tempo real, com destaque para a aplicação do filtro de Sobel para detecção de bordas. Essa etapa exigiu o uso de estratégias como janelas deslizantes (*sliding window*) e *buffers* de linha, viabilizando a aplicação de máscaras de convolução diretamente sobre o fluxo de dados. A construção de um *pipeline* completo — desde a aquisição dos dados da câmera, passando pelo processamento, até a exibição da imagem filtrada — explorou o paralelismo do *hardware* e consolidou a compreensão dos residentes sobre o projeto de sistemas digitais complexos, nos quais múltiplos módulos operam de forma integrada e sincronizada.

14. Síntese dos Aprendizados em FPGA

A trilha de *FPGA* evidenciou a importância da prática — por meio da modelagem, simulação e implementação em *hardware* — como elemento central para a consolidação dos conceitos. A progressão estruturada dos conteúdos permitiu a evolução consistente dos estudantes, da lógica digital básica ao desenvolvimento de sistemas mais complexos.

A abordagem baseada em projetos contribuiu para o desenvolvimento da capacidade de interpretar requisitos e traduzi-los em implementações funcionais em *HDL*. Por fim, observou-se que a atuação do instrutor deve evoluir ao longo do processo, passando de uma condução inicial mais próxima para uma supervisão técnica, acompanhando o ganho de autonomia dos estudantes.

15. Considerações finais

O curso desenvolvido no âmbito do Projeto *EmbarcaTech* demonstra que a formação em sistemas embarcados e *IoT* requer uma abordagem pedagógica integrada, capaz de articular conteúdos, práticas e estratégias de ensino em um modelo coerente.

A combinação entre aulas estruturadas, projetos aplicados, mentoria especializada e uso de tecnologias reais contribuiu para a formação de profissionais capazes de atuar em ambientes de alta complexidade tecnológica, alinhando-se às demandas contemporâneas dos setores produtivos e de ambientes de inovação.

Da formulação dos desafios à validação dos protótipos

Luan Diego de Lima Pereira

(Coordenador Técnico)

Marcelo Mendonça dos Santos

(Coordenador Técnico)

Na estruturação do Programa *EmbarcaTech*, a coordenação técnica foi responsável por atuar como elo integrador entre os diferentes atores envolvidos no processo formativo e de desenvolvimento tecnológico do programa. Foi responsabilidade da coordenação articular com as equipes de residentes e seus respectivos mentores, com a coordenação acadêmica, com a equipe de gestão do programa e os parceiros externos responsáveis pela proposição dos projetos. Mais do que um papel operacional, tratou-se de uma função estratégica, voltada à garantia da coerência técnica, viabilidade e alinhamento das iniciativas ao longo de toda a residência.

A coordenação buscou estruturar uma dinâmica de trabalho que combinasse divisão de responsabilidades e atuação conjunta nos momentos críticos. Considerando a organização do programa em cinco polos, cada um com quatro equipes de desenvolvimento, estabelecemos uma divisão inicial de acompanhamento, na qual cada coordenador ficou responsável por dez equipes. Essa estratégia permitiu um acompanhamento mais próximo e contínuo, sem perder a visão sistêmica do programa, que era mantida por meio de reuniões frequentes e decisões compartilhadas.

Ao mesmo tempo, diversas atividades exigiram atuação conjunta da coordenação técnica, especialmente aquelas relacionadas à definição de diretrizes, avaliação de marcos importantes e interlocução com a equipe de gestão e parceiros externos. Esse equilíbrio entre autonomia e colaboração foi fundamental para dar conta da complexidade e da escala do programa.

Do ponto de vista cronológico, o trabalho da coordenação técnica foi iniciado na fase de seleção dos projetos. Foram analisadas aproximadamente quarenta propostas submetidas por empresas e instituições parceiras, das quais vinte foram selecionadas para compor a trilha de *Software Embarcado* do programa. Esse processo envolveu não apenas a avaliação do mérito das propostas, mas também uma análise criteriosa de viabilidade técnica, considerando fatores como escopo, complexidade, recursos necessários e aderência ao perfil formativo da residência.

Uma vez definidos os projetos, a etapa seguinte consistiu na formação das equipes de desenvolvimento. Buscou-se, nesse momento, a construção de grupos equilibrados, levando em conta os diferentes perfis, níveis de conhecimento e áreas de interesse dos residentes. Essa composição foi fundamental para promover tanto a aprendizagem colaborativa quanto a capacidade de execução dos projetos.

Na sequência, foi realizada a distribuição dos projetos entre as equipes, considerando não apenas o perfil dos estudantes, mas também a expertise dos mentores e a infraestrutura disponível em cada polo. Esse alinhamento foi essencial para aumentar as chances de sucesso dos projetos, respeitando as especificidades locais e as competências envolvidas.

Outro marco importante foi a realização de reuniões iniciais envolvendo as empresas parceiras, as equipes de desenvolvimento e a coordenação técnica, com o objetivo de adequar e refinar os escopos dos projetos. Em muitos casos, foi necessário ajustar expectativas, redefinir entregáveis e delimitar melhor os objetivos,

de modo a torná-los compatíveis com o tempo e os recursos disponíveis no programa.

Ao longo do desenvolvimento, a coordenação técnica acompanhou de perto o levantamento e a análise dos requisitos realizados pelas equipes, buscando garantir consistência, clareza e viabilidade nas especificações. Paralelamente, também foi conduzido o processo de composição e análise dos orçamentos de componentes necessários à execução dos projetos, etapa especialmente relevante no contexto de sistemas embarcados, em que a escolha de *hardware* impacta diretamente o desenvolvimento.

A rotina de acompanhamento incluiu reuniões periódicas com a equipe de gestão e a coordenação acadêmica, nas quais eram apresentados relatórios de andamento, discutidos desafios e definidas estratégias de intervenção quando necessário. Esse fluxo contínuo de comunicação foi essencial para a tomada de decisões e para o alinhamento geral do programa.

A coordenação técnica também teve papel ativo no apoio à realização dos seminários de conceituação dos projetos, momentos importantes para validação das propostas e compartilhamento de conhecimento entre as equipes. Em alguns casos, foi necessária a intervenção direta em projetos que apresentaram dificuldades significativas, seja para readequação de escopo ou, em situações mais críticas, para substituição por alternativas mais viáveis do ponto de vista técnico.

Na fase intermediária do programa, destacaram-se as ações de apoio à organização do evento *Business Innovate Summit (BIS)*, estruturado em um formato semelhante ao de *hackathons*. Nesse contexto, as equipes foram desafiadas a revisitar seus projetos sob uma nova perspectiva, incorporando uma abordagem orientada à inovação e ao empreendedorismo. Mais do que soluções técnicas, os trabalhos passaram a ser apresentados com uma roupagem alinhada à concepção de produtos ou serviços com potencial de inserção no mercado, incluindo a definição de propostas de valor, público-alvo e modelos iniciais de negócio. Esse processo contribuiu para ampliar a visão dos residentes, estimulando não

apenas o desenvolvimento tecnológico, mas também competências relacionadas à comunicação, validação de ideias e pensamento estratégico. O evento, além de proporcionar visibilidade aos projetos, favoreceu a troca de experiências entre os participantes e a aproximação com o ecossistema de inovação.

Posteriormente, foi organizada uma segunda rodada de seminários, já com os projetos em estágio avançado de desenvolvimento, permitindo uma avaliação mais aprofundada dos resultados alcançados e dos desafios enfrentados ao longo do processo.

Na etapa final, a coordenação técnica concentrou esforços na análise dos entregáveis dos projetos, verificando o atendimento aos requisitos definidos e a qualidade das soluções desenvolvidas. Entre os principais artefatos avaliados estavam os códigos-fonte das soluções implementadas, a documentação técnica completa, os relatórios finais, vídeos demonstrativos, *dashboards* de acompanhamento das etapas de desenvolvimento e os registros formais das reuniões realizadas ao longo do projeto. Dentre essas, destacou-se a reunião de validação junto ao *Product Owner (PO)*, considerada um marco fundamental para a confirmação de que as entregas correspondiam às expectativas e necessidades dos parceiros proponentes. Esse momento marcou também o encerramento do ciclo da residência, consolidando os aprendizados e resultados obtidos.

Polo Bom Jesus da Lapa

Manoel Messias Silva Júnior

(Mentor)

Elias Guimarães Miranda Barbosa da Silva

(Mentor)

O Centro Multidisciplinar de Bom Jesus da Lapa, vinculado à Universidade Federal do Oeste da Bahia (UFOB), está localizado na região do Médio São Francisco, na porção Oeste do estado da Bahia. Nessa unidade são ofertados os cursos de Engenharia Elétrica e Engenharia Mecânica, cada um com 45 vagas em regime de período integral, atendendo estudantes provenientes de diversos municípios baianos e de outros estados da federação.

O *campus* conta com uma infraestrutura laboratorial voltada ao ensino e à pesquisa, incluindo o Complexo Multi Laboratorial de Engenharia Mecânica e o pavilhão Laboratorial de Engenharia Elétrica, que abrange os Laboratórios de Máquinas Elétricas, Energias Renováveis, Automação e Controle, Eletrotécnica, Sistemas Digitais e Eletrônica Analógica.

No contexto do programa *EmbarcaTech*, o polo contou com a participação de 20 estudantes, dos quais 17 integraram a trilha de Sistemas Embarcados. O perfil do grupo era diversificado: a maioria dos participantes cursava Engenharia Elétrica, em estágios que variavam entre o 5º e o 10º semestre, mas a turma também incluía um aluno de Ensino Médio, dois estudantes do curso de Gestão da Tecnologia e Inovação e uma aluna de Mestrado em Engenharia Elétrica e de Computação. Essa heterogeneidade de formações e níveis de maturidade acadêmica representou, ao mesmo tempo, um desafio e uma riqueza para o processo de

aprendizagem, pois favorece trocas de conhecimento entre perfis complementares.

Os estudantes foram organizados em quatro equipes, compostas de modo a equilibrar competências distintas dentro de cada grupo: alunos com maior afinidade em *hardware* e eletrônica foram mesclados com aqueles de perfil mais voltado a *software* e programação. Essa estratégia de composição buscou reproduzir a dinâmica de equipes multidisciplinares encontrada em ambientes profissionais e contribuiu para que os integrantes desenvolvessem habilidades além de suas zonas de conforto.

As atividades de mentoria foram realizadas nos Laboratórios de Máquinas Elétricas e de Eletrônica Analógica, com encontros presenciais aos sábados pela manhã durante a Residência Tecnológica. Por se tratar de um *campus* de pequeno porte e considerando que a maioria dos participantes residia no próprio município, os estudantes também dispunham de acesso aos laboratórios ao longo da semana, o que ampliou significativamente o tempo disponível para o desenvolvimento dos projetos.

A atuação dos mentores concentrou-se em três frentes articuladas. A primeira consistiu no acompanhamento das tarefas solicitadas pelos instrutores, oferecendo reforço dos conteúdos ministrados nas aulas remotas às terças-feiras e suporte direto às implementações práticas. A segunda frente envolveu a realização de aulas complementares, abrangendo desde fundamentos de eletrônica até temas mais avançados, passando pelo uso adequado de equipamentos de bancada, como multímetro, osciloscópio, gerador de sinais e fontes de alimentação e pela prototipação de aplicações com sensores e atuadores. A terceira frente, de natureza mais transversal, dedicou-se à orientação metodológica das equipes: organização das atividades, gestão de escopo e tempo, e articulação entre os conhecimentos teóricos das aulas remotas e a prática laboratorial.

Essa articulação entre o conteúdo remoto e a vivência presencial revelou-se um dos aspectos mais relevantes da mentoria. Os encontros presenciais funcionavam como espaço de

consolidação: os conceitos apresentados nas aulas síncronas ganhavam concretude quando os estudantes os aplicavam nos circuitos, nos microcontroladores e nos protótipos em construção. O papel do mentor, nesse contexto, era menos o de transmissor de conteúdo e mais o de mediador entre a teoria e a prática, ajudando as equipes a identificar quais conhecimentos eram necessários para avançar em cada etapa dos projetos.

No início da residência, observou-se que muitos estudantes apresentavam dificuldades na consolidação de conhecimentos básicos, especialmente em programação, eletrônica e integração de *hardware* com *software*. Aspectos como gestão do tempo, organização das atividades e trabalho em equipe também se mostraram desafiadores, sobretudo em um contexto híbrido de aprendizagem que exigia maior autonomia. A diversidade do público contribuiu para diferentes ritmos de aprendizagem dentro das equipes, demandando dos mentores atenção individualizada e estratégias diferenciadas de acompanhamento.

Outro desafio significativo esteve associado à adaptação à metodologia de Aprendizagem Baseada em Problemas (ABP). Muitos estudantes não estavam familiarizados com abordagens centradas no protagonismo do aluno e, inicialmente, demonstraram dificuldades na identificação de problemas, na formulação de hipóteses e na proposição de soluções de forma estruturada. A utilização simultânea de múltiplas estratégias pedagógicas, como ensino a distância, encontros síncronos e atividades presenciais, também demandou dos alunos maior disciplina e engajamento contínuo.

A localização do polo no interior do estado impôs ainda um desafio de ordem logística: a ausência de lojas especializadas em componentes eletrônicos na região. Diferentemente dos polos situados em grandes centros urbanos, onde sensores, módulos e componentes discretos podem ser adquiridos com facilidade e agilidade, em Bom Jesus da Lapa toda a obtenção de materiais dependia de compras online, sujeitas a prazos de entrega prolongados e, por vezes, a extravios ou atrasos. Essa limitação

exigiu das equipes um planejamento antecipado mais rigoroso e, em alguns momentos, criatividade para adaptar soluções com os recursos disponíveis, o que também contribuiu para o desenvolvimento da capacidade de resolver problemas sob restrições reais.

Apesar desses desafios, ao longo do programa foi possível perceber uma evolução significativa dos participantes. Gradualmente, os estudantes desenvolveram maior autonomia na busca por soluções, aprimoraram suas habilidades técnicas e passaram a demonstrar maior capacidade de integração entre teoria e prática. Observou-se também uma melhora expressiva no trabalho colaborativo, na comunicação interpessoal e na organização das atividades em grupo. Alunos que no início da residência dependiam fortemente da orientação dos mentores passaram a tomar decisões técnicas com mais segurança e a conduzir etapas do projeto de forma independente.

A utilização de práticas como gamificação contribuiu positivamente para o engajamento, estimulando a participação ativa e a continuidade no programa. Os encontros presenciais e as atividades práticas nos laboratórios foram fundamentais para consolidar o aprendizado, proporcionando experiências concretas e alinhadas às demandas do mercado de trabalho. O amadurecimento mais visível, contudo, não foi apenas técnico: os estudantes passaram a lidar melhor com frustrações, a negociar soluções dentro das equipes e a encarar problemas inesperados como parte natural do processo de desenvolvimento.

As quatro equipes do polo desenvolveram projetos que refletem a diversidade de aplicações possíveis na área de sistemas embarcados e Internet das Coisas. O primeiro projeto consistiu em um sistema de rastreamento industrial por *RFID*, voltado ao controle de produtos em tempo real no ambiente fabril. O segundo projeto, denominado *Environmental Monitor*, propôs uma estação de monitoramento ambiental de baixo custo baseada em *edge computing*, destinada à proteção de equipamentos sensíveis em ambientes industriais. O terceiro projeto desenvolveu um Veículo

Autoguiado (AGV) controlado por comandos de voz via *Amazon Alexa*, integrando navegação autônoma por seguimento de linha com comunicação em tempo real via *AWS*. O quarto projeto, *IR Remote*, criou um controle universal *smart* para ar-condicionado por infravermelho, com interface *web* e funcionalidades de temporização programável.

A experiência do polo Bom Jesus da Lapa evidencia que o *EmbarcaTech* cumpre com eficácia seu propósito de formação técnica e socioemocional, e revela algo ainda mais significativo: o potencial transformador da interiorização do desenvolvimento tecnológico. Para muitos dos estudantes que participaram do programa, esta foi uma oportunidade única, a primeira vivência de um ciclo completo de desenvolvimento de produto em parceria com empresas reais, algo que, até então, parecia acessível apenas a quem estivesse nos grandes centros. O programa demonstrou que é possível formar profissionais de alto nível em regiões distantes dos polos tecnológicos tradicionais, desde que haja infraestrutura adequada, orientação qualificada e, sobretudo, a decisão institucional de levar oportunidades a quem delas mais necessitam.

Os desafios enfrentados, como a escassez de componentes eletrônicos na região e a heterogeneidade das equipes, foram parte essencial do processo formativo. Foi justamente ao lidar com restrições reais que os estudantes mais se desenvolveram, não apenas tecnicamente, mas como profissionais capazes de tomar decisões, negociar soluções e trabalhar colaborativamente. Ao final da residência, o polo entregou não apenas quatro projetos funcionais, mas também a evidência concreta de que investir na formação tecnológica no interior é investir no futuro de uma geração inteira de profissionais que, de outra forma, poderiam não ter tido essa chance.

Projeto 1: AGV com Comando de Voz para Tarefas de Logística Interna

Bruna Alves de Barros
(Residente)

Eder Renato da Silva Cardoso Casar
(Residente)

Mariana Farias da Silva
(Residente)

Matheus Pereira Alves
(Residente)

A movimentação interna de materiais em ambientes industriais é uma das etapas que mais consome tempo e recursos em plantas fabris. Embora existam soluções comerciais de Veículos Autoguiados (AGVs) para automatizar esse transporte, o custo elevado e a complexidade das interfaces de operação tornam essas tecnologias inacessíveis para pequenas e médias empresas — realidade predominante no contexto industrial brasileiro. Operar esses sistemas geralmente exige treinamento especializado, afastando o operador comum da interação com o robô.

O projeto surgiu da proposta de enfrentar essas barreiras: desenvolver um AGV de baixo custo, controlado por comandos de voz via *Amazon Alexa*, operável de forma intuitiva por qualquer pessoa.

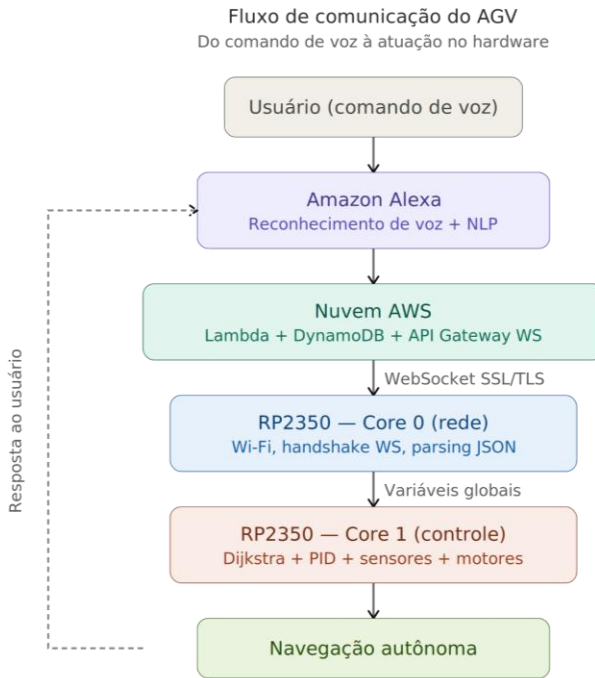
O desenvolvimento do projeto foi marcado por uma mudança significativa logo no início. O grupo havia sido designado para um projeto de agropecuária de precisão — um colar inteligente para monitoramento bovino —, mas os custos dos componentes tornaram a proposta inviável, levando ao cancelamento. Com a atribuição do AGV, a equipe recomeçou do zero: nova área, novo escopo, novo *Product Owner (PO)* e novas referências.

No primeiro seminário, a equipe ainda trabalhava com sugestões de tecnologias como *ROS* e *LiDAR*, que a banca reconheceu como sofisticadas demais para o tempo disponível. A partir daí, o escopo foi delimitado com mais rigor. No segundo seminário, o projeto já tinha identidade clara: substituição do *Raspberry Pi 4* pelo microcontrolador *RP2350*, mais acessível e adequado; e troca das *tags RFID* por marcos transversais detectados por sensores infravermelhos, solução mais simples e econômica.

A organização apoiou-se no *Airtable* para gestão de tarefas e no *GitHub* para versionamento. As reuniões com o *PO* foram fundamentais para alinhar expectativas e manter o foco no essencial. O mentor orientou decisões técnicas e ajudou a avaliar o que precisava ser refinado. Apesar das funções formalmente divididas, montagem, testes e desenvolvimento foram realizados de forma colaborativa no laboratório do polo em Bom Jesus da Lapa.

O *AGV* é um robô seguidor de linha que navega autonomamente entre pontos predefinidos, recebendo comandos de voz pela *Alexa*. A arquitetura opera em três camadas, conforme o diagrama a seguir.

Figura 1: Fluxograma do processo de comunicação: do comando de voz à atuação no hardware



Fonte: Os autores.

Na camada cognitiva, a *Alexa* interpreta o comando de voz e aciona uma função na nuvem *AWS*. Na camada de conectividade, um servidor *WebSocket* mantém comunicação bidirecional persistente com o robô, encaminhando comandos em *JSON*. Na camada embarcada, o *RP2350* opera em *Dual-Core*: o *Core 0* gerencia a conexão *Wi-Fi* e o recebimento de comandos via *WebSocket (SSL/TLS)*; o *Core 1* executa o controle de navegação — leitura de sensores, algoritmo *PID* de rastreamento de linha e algoritmo de *Dijkstra* para cálculo da rota de menor custo.

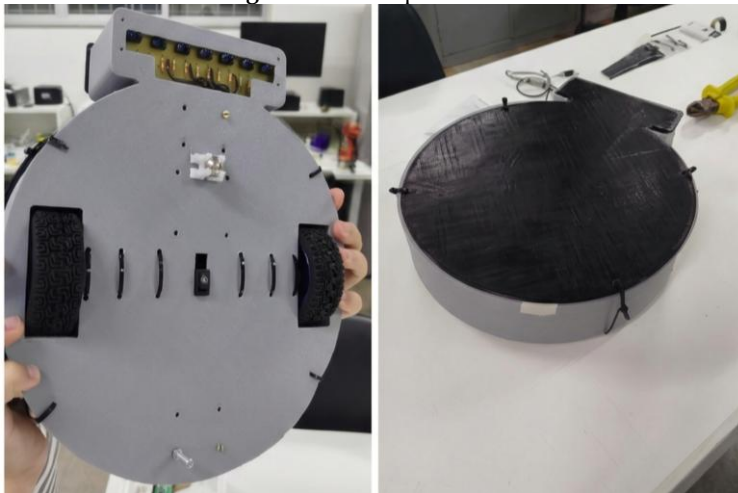
O *hardware* inclui chassi impresso em 3D e uma *PCB* customizada projetada no *KiCad* e fabricada por fresagem *CNC*, integrando sete sensores infravermelhos multiplexados, sensor de distância laser *VL53L0X* e *IMU MPU6050* para odometria. O

ambiente é modelado como um grafo, com cruzamentos de fita no solo como nós/conexões e linhas-guia como arestas.

A equipe entregou um protótipo funcional, validado em simulação (*Webots*) e em testes físicos. O sistema processou comandos de voz e os transmitiu ao robô em menos de 200 milissegundos. O AGV navegou autonomamente entre pontos do mapa, com paradas precisas nos marcos transversais. A arquitetura na nuvem suporta múltiplos dispositivos simultâneos.

Na reunião de apresentação ao *PO* em fevereiro de 2026, o Mínimo Produto Viável (*MVP*) foi formalmente aprovado, com sugestões de evolução como interface *web* para configuração de rotas e possibilidade de embarcar a *Alexa* no próprio dispositivo.

Figura 2: Protótipo do AGV



Fonte: Os autores.

O maior desafio foi a mudança de projeto no início da residência, exigindo adaptação rápida a uma área completamente nova. A localização no Oeste da Bahia trouxe atrasos no envio de componentes, impactando o cronograma.

Tecnicamente, problemas como incompatibilidade mecânica entre rodas e eixos dos motores, deslocamento do centro de gravidade do chassi, interferência eletromagnética nos sensores e a

sincronização entre controle *PID*, comunicação *WebSocket* e máquina de estados do sensor de obstáculos no ambiente *Dual-Core* exigiram soluções criativas e iterativas.

A equipe aprendeu prototipação completa (da modelagem 3D à fabricação de *PCB*), programação embarcada em *C/C++*, serviços *AWS* e integração de assistentes virtuais a dispositivos embarcados. Conceitos de controle *PID*, teoria dos grafos, odometria e fusão sensorial foram aplicados na prática.

A experiência de trabalhar com metodologias ágeis, gerenciar mudanças de escopo e colaborar em equipe multidisciplinar foi transformadora. A participação no *Bahia Innovate Summit* ampliou a visão sobre sistemas embarcados e fortaleceu o senso de comunidade do programa.

Para integrantes vindos do interior da Bahia, desenvolver um projeto com esse nível de integração tecnológica representou um marco na formação profissional, demonstrando que inovação pode acontecer fora dos grandes centros.

O projeto oferece caminhos claros de evolução: migração para *FreeRTOS*, desenvolvimento de interface *web* para configuração de rotas, integração de visão computacional para reconhecimento dinâmico de obstáculos e exploração de carregamento por indução. A base construída durante a residência constitui uma plataforma sólida para essas melhorias.

Projeto 2: *IR REMOTE* - Controle Universal *Smart* para Ar-condicionado

Jadson de Jesus Santos
(Residente)

Mateus Moreira da Silva
(Residente)

Rafael Souza Barbosa
(Residente)

Sauan de Souza Santos
(Residente)

O projeto foi motivado pelas limitações dos aparelhos de ar-condicionado convencionais, que ainda dependem de controles infravermelhos simples, com baixa integração digital e ausência de recursos de automação. Além disso, observa-se a falta de equipamentos inteligentes acessíveis que permitam controle remoto e gerenciamento mais eficiente desses dispositivos. O impacto do projeto está na possibilidade de modernizar equipamentos já existentes de forma acessível, ampliando o uso de automação residencial sem a necessidade de substituição por dispositivos mais caros.

O desenvolvimento do projeto iniciou com a definição dos requisitos e das funcionalidades principais, como controle universal de múltiplas marcas, temporização programável e interface *web* local. Inicialmente, foi escolhido o microcontrolador *Raspberry Pi Pico W*, porém surgiram dificuldades relacionadas à disponibilidade de bibliotecas infravermelhas compatíveis, o que aumentaria o tempo de desenvolvimento.

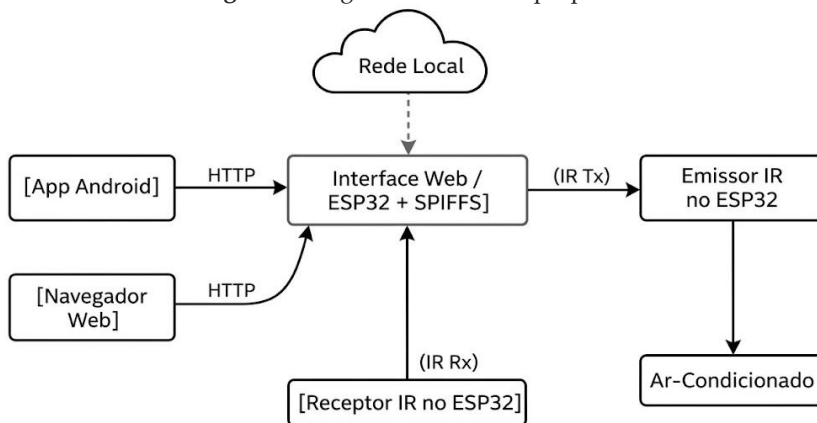
Diante disso, a equipe optou pela migração para o *ESP32*, que oferece melhor suporte para comunicação infravermelha e

conectividade *Wi-Fi* integrada. Após essa decisão, o desenvolvimento foi organizado em etapas incrementais, incluindo prototipação de *hardware*, desenvolvimento do *firmware* e criação da interface *web*, com reuniões periódicas para alinhamento dos requisitos.

A solução desenvolvida consiste em um dispositivo baseado no *ESP32* que atua como intermediário entre o usuário e o ar-condicionado. O sistema hospeda uma interface *web* acessível pela rede local, permitindo controle por *smartphone*, tablet ou computador.

Os comandos enviados pela interface são processados pelo *ESP32* e transmitidos ao ar-condicionado por meio de um emissor infravermelho. O receptor infravermelho permite capturar e identificar protocolos de diferentes marcas, garantindo compatibilidade com diversos modelos. A Figura 1 demonstra a arquitetura modular, composta por interface, controle embarcado e comunicação infravermelha, e permite fácil expansão e maior flexibilidade no desenvolvimento.

Figura 1: Diagrama do sistema proposto

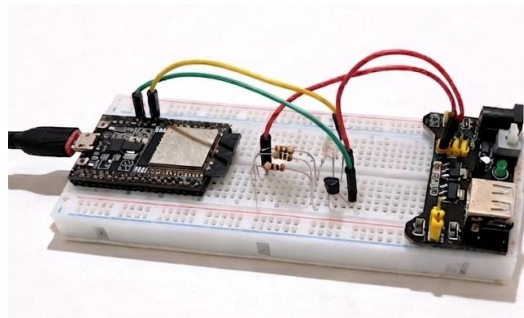


Fonte: Os autores.

O sistema desenvolvido consegue controlar aparelhos de ar-condicionado de algumas marcas por meio do emissor *IR*, que envia os sinais conforme os protocolos de cada fabricante, tudo isso acessível pela interface *web* através do painel de controle. As funcionalidades de descoberta de protocolo, detecção de controle e criação de temporizações também foram testadas e funcionaram como esperado.

Os testes foram feitos com um protótipo montado em *protoboard*, reunindo todos os componentes necessários para o funcionamento do sistema, conforme mostrado na Figura 02. O protótipo foi validado com sucesso no acionamento de aparelhos de ar-condicionado reais.

Figura 2: Protótipo no Protoboard



Fonte: Os autores.

O aplicativo desenvolvido para facilitar a configuração inicial e melhorar a experiência do usuário, também foi validado e testado, contemplando todas as funcionalidades disponíveis. Além disso, foi confirmada a persistência dos dados, com salvamento em memória não volátil, garantindo que as temporizações sejam mantidas mesmo após reinicializações, e com sincronização de horário via *NTP*.

Durante o desenvolvimento do projeto, a equipe se deparou com alguns desafios técnicos que acabaram servindo como

aprendizado. Um dos principais foi a busca por bibliotecas *IR* para ar-condicionado que oferecessem suporte ao *Raspberry Pi Pico W*, microcontrolador inicialmente escolhido para o projeto.

Entretanto, para avançar seria necessário reestruturar essas bibliotecas ou até mesmo criar uma nova, o que demandaria um trabalho grande e desnecessário dentro das condições do projeto. Além disso, havia também o tempo limitado para a conclusão. Por conta disso, chegou-se à decisão de migrar para o *ESP32*, o que permitiu um avanço mais rápido e preciso no desenvolvimento do projeto.

O desenvolvimento do projeto proporcionou evolução técnica e pessoal ao grupo, com aprofundamento em sistemas embarcados, uso dos microcontroladores *Raspberry Pi Pico W* e *ESP32*, integração *hardware-software*, comunicação *Wi-Fi*, sinais infravermelhos e organização modular de *firmware*. Também foram consolidados conhecimentos em interfaces *web* embarcadas e persistência de dados. No aspecto humano, a experiência fortaleceu o trabalho em equipe, a comunicação e a tomada de decisões, além de desenvolver habilidades como liderança, responsabilidade e resolução de problemas, contribuindo diretamente para a formação profissional dos integrantes.

Como continuidade, o projeto pode ser aprimorado com a ampliação do suporte a novos modelos e marcas de ar-condicionado, bem como a implementação de funcionalidades adicionais, como controle completo de recursos avançados (ex.: *swing* e modos específicos). Também é possível evoluir o sistema com acesso remoto via internet, utilizando mecanismos seguros de comunicação. No *hardware*, recomenda-se a finalização da *PCB* e o desenvolvimento de um encapsulamento adequado, além da realização de testes mais extensivos de desempenho e usabilidade, que contribuirão para aumentar a robustez e a confiabilidade do sistema, indicando potencial de evolução tanto acadêmica quanto comercial.

Projeto 3: Desenvolvimento de Estação de Monitoramento Ambiental com Controle Automatizado

João Pedro Soares Racolto
(Residente)

Nivaldo Rodrigues da Silva Júnior
(Residente)

Rian de Sena Mendes
(Residente)

Samuel Guedes Canário
(Residente)

O monitoramento e controle de variáveis ambientais são essenciais para o desenvolvimento de diversas atividades atuais, desde o setor de serviços até o industrial. Dentre eles, um exemplo de importância é a produção de componentes eletrônicos, em razão da elevada quantidade de partes sensíveis que podem ser danificadas se expostas a condições inadequadas de manuseio e armazenamento, as quais podem propiciar descargas eletrostáticas e oscilações de temperaturas, causas de falhas severas em tais componentes.

Observa-se que muitas empresas não dispõem de sistemas de monitoramento e/ou controle ambiental, em razão, sobretudo, de seu elevado custo ou difícil integração. O desafio, então, é propor uma solução de baixo custo e fácil integração que consiga realizar o monitoramento e controle de variáveis ambientais em uma instalação da empresa parceira, atuando para prevenir que falhas severas ou catastróficas ocorram e danifiquem componentes importantes.

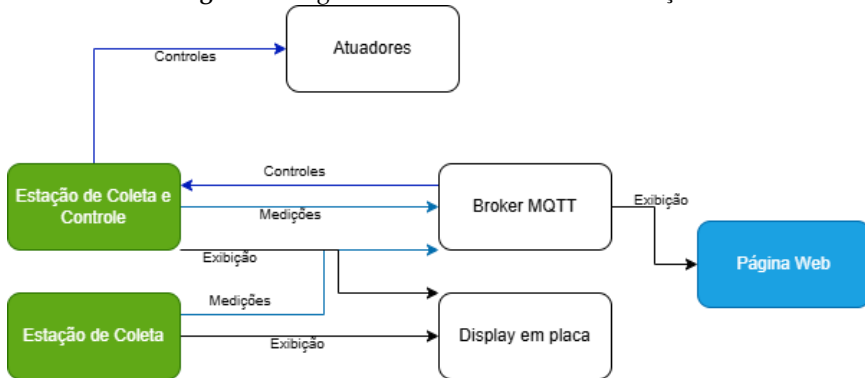
A ideia de solução, desde sua concepção inicial, esteve voltada a um sistema de monitoramento distribuído com o intuito de manter bom desempenho em ambientes fabris de maiores dimensões, sendo ajustada conforme era desenvolvida.

A equipe se organizou de forma a segmentar o desenvolvimento da solução em diferentes partes, como o desenvolvimento da comunicação via protocolo *MQTT*, montagem de *hardware* e criação de interfaces *web*, cada uma sob responsabilidade de um integrante, o que permitiu maior autonomia e controle para cada um em seu respectivo setor. A evolução desse projeto não ocorreu de forma direta ou linear, de forma que algumas etapas necessitaram de maior atenção ou tempo, devido a uma maior complexidade ou dependência da conclusão de atividades anteriores.

A solução proposta consiste em um sistema de monitoramento e controle ambiental que realiza medições de variáveis de ambiente e controla diretamente atuadores com o fim de manter a temperatura, a umidade e a luminosidade dentro dos limites adequados para a conservação de componentes eletrônicos.

Ela é construída utilizando o microcontrolador *Raspberry Pi Pico 2 W*, e com base no protocolo de comunicação sem fio *MQTT*, o qual é um protocolo leve de comunicação baseado no modelo *publish/subscribe* (publicação/assinatura), projetado para troca eficiente de mensagens entre dispositivos em redes com baixa largura de banda e alta latência, permitindo que dispositivos (clientes) publiquem dados em tópicos e outros dispositivos se inscrevam nesses tópicos para receber as informações, o qual permite que estações de coleta de dados sejam distribuídas pelo ambiente a ser monitorado, desde que conectadas a um *broker MQTT* (ponto de comunicação central), criado em um computador de placa única *Raspberry Pi 3*. As estações são equipadas com conjuntos de sensores (*SHT3x* para temperatura e umidade e *BH1750* para luminosidade), que aferem as variáveis desejadas em diversos pontos do ambiente e um conjunto de atuadores (*LED* emissor para controle de ar-condicionado e relé para controle de lâmpadas) que permitem controle direto sobre os dispositivos do espaço alvo. O fluxo de dados planejado para a solução é mostrado na Figura 1.

Figura 1: Diagrama de fluxo de dados da solução



Fonte: Os autores.

A ideia trabalhada também busca fornecer ao usuário dados sobre o estado do ambiente onde o sensoriamento é realizado, de forma que uma interface *web* foi desenvolvida para mostrar os dados coletados em tempo real, além de exibir sua variação ao longo do tempo. Algumas estações contam ainda com *displays* que exibem as médias das medições realizadas pelos conjuntos de sensores.

De modo a garantir a conectividade e funcionamento da rede de comunicação em condições adversas, o sistema também é capaz de criar um *Access Point*, ou seja, caso não seja possível se conectar a uma rede *Wi-Fi*, um ponto de acesso é criado para que as estações se conectem e a comunicação *MQTT* seja realizada.

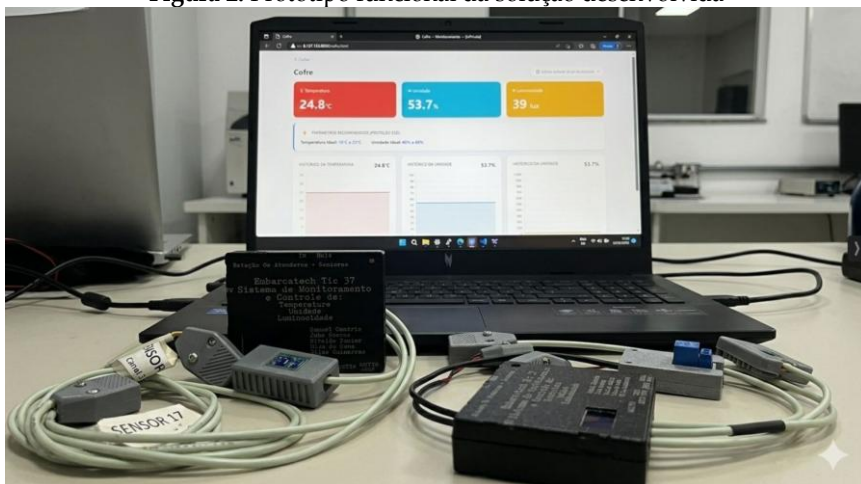
O desenvolvimento da solução levou a um protótipo inicial funcional, conforme Figura 2, que apresenta todas as estruturas previstas para o sistema. Dessa forma, ele realiza medições de temperatura, umidade e luminosidade que servem de base para o controle de lâmpadas e aparelhos de ar-condicionado por meio de relé e emissor infravermelho, respectivamente, sendo ambos os meios de controle funcionais.

Além disso, os valores coletados do ambiente são utilizados para alimentar as interfaces designadas: o *display* incluído na placa e a página *web*, também em pleno funcionamento. O *Access Point* também já se encontra desenvolvido e funcional no protótipo.

Em relação aos testes realizados, o sistema foi aplicado apenas em ambientes laboratoriais de menor escala e com um número reduzido de estações de coleta e controle, sendo validado com sucesso para esse tipo de ambiente.

O sistema ainda não foi validado em ambientes de maior escala, como a indústria que é alvo da solução, nem com uma quantidade mais elevada de estações conectadas ao *broker*, para que seja atestada a sua capacidade de funcionamento em diferentes distâncias de comunicação e/ou com número elevado de dispositivos conectados à rede *MQTT*.

Figura 2: Protótipo funcional da solução desenvolvida



Fonte: Os autores.

O projeto, naturalmente, encontrou obstáculos que desafiaram a sua implementação com a estrutura e duração desejadas. Algumas partes do *hardware* planejadas acabaram não sendo viáveis para a implementação final, como componentes que apresentaram defeitos de fabricação e não puderam ser integrados ao sistema final, precisando ser substituídos por outros, ainda que de modelos diferentes, no intuito de manter o prazo de conclusão estipulado para a atividade.

Houveram também dificuldades relacionadas ao *software*, como uma busca por modularidade em uma solução que apresenta grande quantidade de rotinas de coleta e tratamento de dados, além de controle de atuadores e funções de comunicação *MQTT*. Foi necessária uma maior quantidade de tempo e atenção para a realização de testes e reorganização de código, com o fim de garantir o funcionamento do sistema na estrutura de baixo custo utilizada.

O prazo de entrega de componentes também tornou algumas seções do trabalho mais densas, considerando que alguns sensores não foram obtidos em prazo viável para sua utilização no protótipo. A rota utilizada, então, foi implementar o sistema utilizando sensores da mesma família daqueles inicialmente selecionados e de acesso facilitado no ambiente laboratorial, o que permitiria uma correção futura extremamente simples para os modelos iniciais.

A residência, durante todo o período de estudos teóricos e desenvolvimento de projetos práticos, representou uma fonte notável de conhecimento técnico e amadurecimento profissional. Por meio dela, os integrantes da equipe puderam ter contato com teorias e materiais até então parcialmente compreendidos ou totalmente desconhecidos, expandindo suas habilidades em sistemas digitais, protocolos de comunicação e *software* embarcado.

Ademais, a residência permitiu que os estudantes compreendessem de forma profunda como ocorre o gerenciamento de projetos e organização profissional durante o trabalho com a empresa parceira que solicitou a solução implementada, trazendo para a prática as qualidades necessárias para que o projeto fosse finalizado conforme determinado.

A experiência adquirida durante o projeto final, considerando os conhecimentos teóricos adquiridos em aula e a contribuição para o planejamento da solução por parte da empresa parceira, tornaram a residência tecnológica um conjunto intenso de aprendizados, seja na parte ética, profissional ou técnica da formação dos residentes.

O projeto, embora já seja funcional em relação aos testes realizados, pode ser finalizado, com a execução de mais testes com maior número de estações em um ambiente fabril real, ou melhorado, explorando caminhos como a implementação de análise preditiva, a automação e verificação de funcionamento dos controles de atuadores ou mesmo a criação de subdivisões do ambiente monitorado em seções menores.

Essas mudanças podem melhorar a autonomia e a cobertura do sistema de monitoramento, além de evitar a sobrecarga da rede *MQTT* e falhas no controle do ambiente devido a erros de montagem do *hardware*, tornando a solução ainda mais funcional e útil para a proteção de componentes eletrônicos sensíveis.

Projeto 4: Implantação de Sistema RFID para Gerenciamento de Produção

Atenilton Santos de Souza Júnior
(Residente)

Daniel Silva de Souza
(Residente)

José Vinicius Rodrigues Soares
(Residente)

Ronaldo Cesar Santos Rocha
(Residente)

Luiz Eduardo Pereira Gomes
(Residente)

Em ambientes industriais, o controle e a rastreabilidade de equipamentos ao longo das etapas de produção são fundamentais para garantir organização e eficiência. No cenário observado, esse processo ainda dependia de registros manuais, o que dificultava a localização dos itens e aumentava a chance de erros e retrabalho. Diante disso, a equipe buscou desenvolver uma solução automatizada utilizando tecnologia *RFID*, com o objetivo de tornar o acompanhamento dos equipamentos mais ágil e confiável. Além de atender a uma necessidade real da empresa parceira, o projeto também contribuiu para a formação dos integrantes, ao proporcionar uma experiência prática com um problema do mundo real.

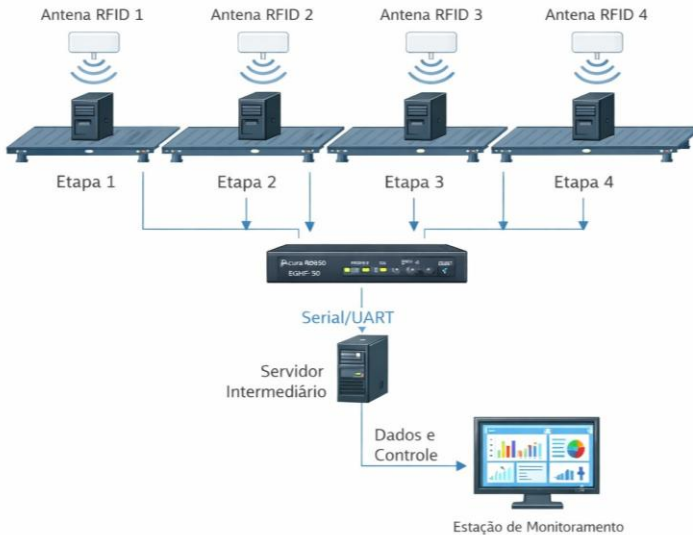
A trajetória da equipe começou com diferentes ideias sobre como resolver o problema proposto. Inicialmente, foi considerada a construção de um sistema *RFID* totalmente do zero, utilizando sensores e antenas mais simples. No entanto, ao longo das discussões e testes iniciais, percebeu-se que essa abordagem poderia limitar o desempenho e a confiabilidade do sistema,

levando à decisão de utilizar um leitor *RFID* de padrão industrial como base da solução. A organização do grupo ocorreu de forma prática, com divisão das atividades entre integração com o leitor, desenvolvimento da aplicação em *Python* e implementação do sistema embarcado. Essa divisão permitiu que as frentes evoluíssem em paralelo, com integração gradual entre os módulos. O desenvolvimento aconteceu de forma iterativa, começando pela comunicação com o leitor e leitura das *tags*. Em seguida, foram incorporadas etapas de armazenamento dos dados, criação da interface e ajustes no controle de antenas e potência. Ao final, o sistema foi integrado e testado em conjunto, permitindo verificar seu funcionamento e identificar ajustes necessários.

A solução consiste em um sistema de rastreamento industrial baseado em tecnologia *RFID UHF* para monitoramento de ativos em tempo real, conforme Figura 1. O núcleo do *hardware* utiliza o leitor industrial *Acura Edge-50 TCP/IP*, operando entre 860 MHz e 960 MHz, conectado a antenas que identificam etiquetas nos produtos. Essa abordagem automatiza o controle logístico, garantindo visibilidade precisa da movimentação de produtos entre as etapas de montagem e inspeção, superando as limitações de métodos manuais ou baseados em códigos de barras.

A arquitetura modular integra um *driver* de baixo nível em Linguagem C via *Mercury API* para controle direto das funções de *hardware*, como ajuste de potência e seleção de antenas. Este comunica-se com uma interface em *Python*, responsável pela lógica de rastreamento e persistência dos registros. O sistema opera de forma híbrida, utilizando o microcontrolador *Raspberry Pi Pico W* para acionamentos manuais via botões, que envia comandos por um servidor *web*, enquanto a gestão e o monitoramento dos dados ocorrem através da interface gráfica desenvolvida.

Figura 1: Arquitetura conceitual do sistema RFID



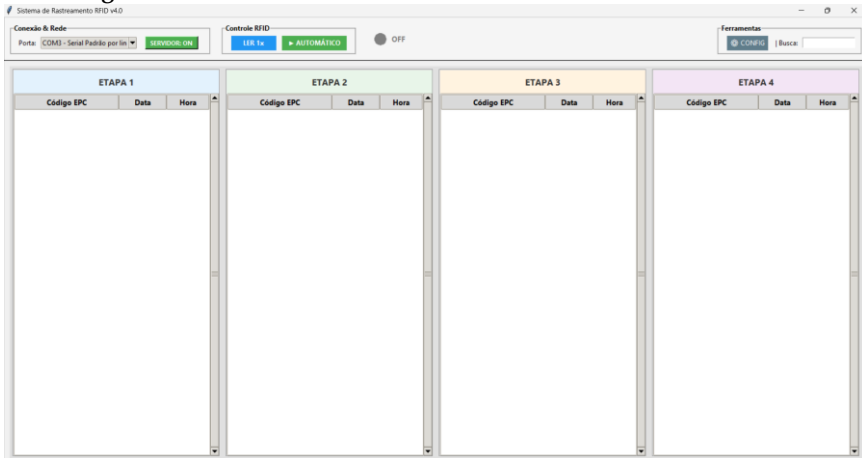
Fonte: Os autores.

O projeto resultou na construção de um protótipo funcional de uma arquitetura distribuída para rastreamento industrial, validado como uma prova de conceito tecnicamente bem-sucedida. A solução integra um leitor industrial multiprotocolo e antenas de alto desempenho a um ecossistema de *software* modular, composto por *drivers* de baixo nível em linguagem C, um camada de processamento lógico em *Python* para a gestão do fluxo produtivo e uma interface gráfica intuitiva. O estágio atual do protótipo permite uma operação híbrida, sendo capaz de processar tanto leituras automáticas contínuas quanto acionamentos manuais via um nó de borda microcontrolado, que oferece *feedback* imediato ao operador através de um *display* e botões físicos.

A eficácia do sistema foi validada por meio de ensaios experimentais em ambiente laboratorial, projetados para reproduzir condições típicas do chão de fábrica. Durante os testes, o sistema demonstrou uma latência inferior a 300 ms entre a detecção física da etiqueta e a atualização visual na tela, confirmando sua aptidão para o monitoramento em tempo real.

Além disso, os testes de estresse evidenciaram a capacidade de identificar múltiplas etiquetas simultaneamente, enquanto a otimização do posicionamento das etiquetas e o ajuste de potência garantiram leituras estáveis a aproximadamente 30 cm de distância, superando com êxito as interferências eletromagnéticas típicas de superfícies metálicas.

Figura 2: Versão final da Interface de monitoramento desenvolvida



Fonte: Os autores.

O desenvolvimento exigiu adaptações rápidas, iniciando com a transição de um protótipo simplificado para uma solução baseada em *hardware* de padrão industrial, a fim de atender às exigências do ambiente fabril. Essa decisão elevou a complexidade técnica do projeto, demandando a integração de um driver em linguagem C (*Mercury API*) com um servidor desenvolvido em *Python*, o que exigiu um esforço significativo de arquitetura de *software* para assegurar a comunicação confiável entre as camadas do sistema.

Essa mudança de abordagem representou um avanço no nível de maturidade do projeto, ao mesmo tempo em que impôs maior rigor no desenvolvimento, especialmente no tratamento de baixo nível e na integração entre componentes.

Além do código, enfrentamos o desafio físico das ondas de rádio refletindo nas carcaças metálicas dos produtos, o que causava interferências e dificultava a precisão das leituras. A solução veio com testes exaustivos de bancada que definiram o posicionamento frontal das etiquetas e a distância ideal de 30 cm para estabilizar o sinal. Essa calibração física expôs nossa maior limitação: o tempo, já que os ajustes de *hardware* exigiam presença física e repetições constantes, ao contrário da flexibilidade do desenvolvimento de *software* remoto. Conciliar esses prazos apertados com a realidade do chão de fábrica foi o verdadeiro teste de fogo para a organização da nossa equipe.

De forma geral, a experiência da residência como um todo foi única e transformadora. Do ponto de vista teórico, as aulas ministradas pelos professores permitiram aos residentes acessar os conhecimentos necessários para a realização do projeto. Além disso, as aulas de mentorias possibilitaram um aprofundamento nos mesmos conhecimentos. Do ponto de vista prático, os testes realizados pelo grupo ao longo dos meses potencializaram ainda mais o aprendizado. Nesses testes, feitos em laboratório do polo, os residentes tiveram contato direto com a tecnologia *RFID*, seus problemas (como a interferência de objetos metálicos nas leituras) e formas de solução. Do ponto de vista humano, cada residente aprendeu, um pouco mais, a como trabalhar em grupo, a como se organizar para cumprir metas e prazos, a ajudar e ser ajudado e principalmente a como se comunicar em um cenário novo, como foi o projeto com a empresa parceira. Por fim, conclui-se que a experiência *Embarcatech* - residência tecnológica em sistemas embarcados foi única, realizadora e transformadora para cada um dos residentes.

As perspectivas futuras para o projeto focam na transição da validação laboratorial para testes em larga escala em ambiente produtivo real. A continuidade prevê a substituição do armazenamento em arquivos locais por uma integração direta com sistemas de gestão empresarial (*ERP*). Entre as melhorias técnicas planejadas estão a gravação de dados diretamente nas etiquetas

RFID e a utilização de multiplexadores para expandir a cobertura de leitura com múltiplas antenas. Essas evoluções visam consolidar o protótipo como uma solução industrial robusta e escalável, deixando para trás as limitações do ambiente controlado.

Polo Feira de Santana

Ramón Luz Lemos Santos

(Mentor)

João Paulo Just Peixoto

(Mentor)

Feira de Santana, localizada no interior da Bahia, é a segunda maior cidade do estado e um dos mais importantes polos econômicos, educacionais e logísticos do Nordeste brasileiro. Conhecida como o “Portal do Sertão”, a cidade ocupa uma posição estratégica ao conectar a capital Salvador a diversas regiões do interior e a outros estados do país. Essa posição privilegiada impulsionou seu crescimento e consolidou um ambiente dinâmico, marcado pela força do comércio, pela diversidade cultural e por uma crescente vocação tecnológica.

Historicamente, Feira de Santana nasceu como ponto de encontro de tropeiros e comerciantes, tornando-se um centro relevante de circulação de pessoas e mercadorias. Com o passar do tempo, evoluiu para além de sua vocação comercial, destacando-se também na educação, nos serviços, na indústria e, mais recentemente, na inovação tecnológica. Hoje, abriga instituições de Ensino Superior, centros de pesquisa e iniciativas que impulsionam o desenvolvimento regional.

É nesse cenário que o *EmbarcaTech* se insere como um agente transformador. Mais do que um programa de formação, ele representa uma ponte entre o conhecimento acadêmico e a aplicação prática na indústria. No polo de Feira de Santana, o programa não apenas ensinou tecnologia, ele formou profissionais capazes de pensar, construir e entregar soluções reais. O polo contou com quatro equipes, mentoradas pelos professores João

Paulo Just Peixoto e Ramon Luz Lemos Santos, ambos do IFBA. Os estudantes, em sua maioria oriundos do IFBA e da UEFS, além de participantes de cidades vizinhas, foram inseridos em uma jornada intensa de aprendizado e desenvolvimento. O laboratório de eletrônica do IFBA tornou-se um verdadeiro ambiente de inovação, onde ideias saíram do papel e ganharam forma.

Os mentorandos foram inicialmente divididos em dois grupos, cada um acompanhado por um mentor. Durante a fase de formação, participaram de aulas estruturadas e receberam desafios semanais que exigiam aplicação prática do conteúdo. As mentorias, realizadas tanto presencialmente quanto de forma remota, foram fundamentais para consolidar o aprendizado. No laboratório, o ambiente era colaborativo, dinâmico e orientado à resolução de problemas. Os mentores atuavam como facilitadores, conduzindo os estudantes na construção das soluções, mas sempre incentivando autonomia e pensamento crítico. Já nas aulas remotas, o conteúdo era apresentado de forma clara e aprofundada, permitindo que os estudantes chegassem às mentorias preparados para executar. Essa integração entre teoria e prática foi um dos grandes diferenciais do programa, potencializando o aprendizado e tornando-o altamente aplicado. Com o início dos projetos finais, essa conexão ficou ainda mais evidente. Tudo o que foi aprendido passou a ser utilizado em soluções reais, com objetivos definidos e aplicabilidade direta no contexto industrial.

Como em todo projeto de grande relevância, os desafios fizeram parte da jornada. Um dos principais foi a falta de experiência prévia com sistemas *CNC*, tecnologia essencial para dois dos projetos desenvolvidos. Esse obstáculo exigiu pesquisa, resiliência e colaboração. Mentores e estudantes precisaram aprender juntos, testando, errando, ajustando e evoluindo. O que inicialmente parecia uma limitação transformou-se em uma das maiores conquistas do polo. Outro ponto fundamental foi a organização das equipes. A formação dos grupos considerou as habilidades e afinidades dos participantes, o que contribuiu significativamente para o desempenho coletivo. Ainda assim, os estudantes precisaram

desenvolver competências além do técnico: gestão de tempo, divisão de tarefas, comunicação e tomada de decisão.

As reuniões com os responsáveis pelos projetos (POs) trouxeram uma visão mais próxima do ambiente corporativo, permitindo que as equipes trabalhassem com escopo definido, prazos e entregas claras. Ao final, todos os projetos foram concluídos, validados e entregues dentro do esperado, um resultado que demonstra a maturidade alcançada pelos participantes.

A evolução dos estudantes ao longo do programa foi evidente e, em muitos casos, surpreendente. Participantes que inicialmente apresentavam dificuldades com conceitos básicos de sistemas embarcados passaram a atuar diretamente na construção de soluções complexas. Mais do que aprender tecnologias, os estudantes desenvolveram confiança, autonomia e capacidade de resolver problemas reais.

Outro aspecto marcante foi a transformação na forma de trabalhar. Os grupos deixaram de atuar como estudantes e passaram a se comportar como times de desenvolvimento, com responsabilidade compartilhada, foco em resultados e comprometimento com entregas. Essa mudança de postura é, talvez, um dos maiores legados do *EmbarcaTech*: formar não apenas profissionais técnicos, mas indivíduos preparados para atuar em ambientes reais de inovação.

No polo Feira de Santana foram desenvolvidos os seguintes projetos:

1. SafeWay: um sistema de fechadura eletrônica com microcontrolador, capaz de gerenciar o acesso a partir de credenciais. O sistema se integra a um serviço *back-end* que valida as tentativas de acesso, permitindo o cadastro de usuários e definição de horários autorizados. A solução demonstra aplicação direta em controle de acesso inteligente.

2. Parafusadeira CNC: equipamento automatizado para parafusamento, capaz de interpretar arquivos *G-Code* e executar operações com precisão. A solução oferece flexibilidade e

padronização no processo industrial, permitindo adaptação a diferentes produtos.

3. Sistema de Verificação de Funcionamento de Queimadores: este projeto representou um dos maiores desafios técnicos do polo. Seu objetivo foi desenvolver um sistema automatizado para controle de qualidade de *cooktops*, utilizando inteligência artificial para verificar, em tempo real, se a chama dos queimadores estava adequada. A equipe precisou se aprofundar em visão computacional e redes neurais, desenvolvendo e treinando modelos capazes de identificar a presença e a qualidade da chama em diferentes condições. Além disso, o sistema integrou sensores de presença e sensores de gás, ampliando a segurança e a confiabilidade da solução. Executar esse tipo de processamento em *hardware* embarcado, com limitações de recursos, foi um desafio significativo. A construção do *dataset*, o treinamento dos modelos e a integração com o sistema físico exigiram organização, disciplina e forte colaboração entre os membros da equipe.

4. Sistema Cartesiano de Aplicação de Cola: neste projeto, os estudantes desenvolveram um sistema automatizado para aplicação de adesivo com alta precisão, substituindo um processo manual sujeito a falhas e desperdícios.

O grande desafio esteve na construção da estrutura física. Os alunos precisaram aprender, na prática, conceitos de mecânica e mecatrônica, desenvolvendo um sistema com movimentação nos eixos cartesianos utilizando motores de passo, trilhos e atuadores lineares. Além disso, foi necessário calcular parâmetros críticos, como velocidade de deslocamento e pressão de aplicação da cola, garantindo qualidade e padronização no processo. O projeto integrou *hardware* e *software* de forma robusta, incluindo controle embarcado, sensores e interface com o operador.

O polo Feira de Santana encerrou sua participação no *EmbarcaTech* com um resultado extremamente positivo. Mais do que projetos concluídos, o programa deixou um legado de formação, transformação e impacto. Os estudantes foram desafiados a sair da zona de conforto, enfrentar problemas reais e

construir soluções com aplicação prática na indústria. Ao longo dessa jornada, aprenderam a trabalhar em equipe, a lidar com incertezas e a transformar conhecimento em entrega. Com o *EmbarcaTech* foi possível mostrar na prática um desenvolvimento tecnológico de alto nível fora dos grandes centros, formando talentos capazes de contribuir diretamente com a indústria baiana. Mais do que ensinar tecnologia, o programa formou profissionais preparados para o futuro, profissionais que entendem que inovação não é apenas teoria, mas execução, colaboração e coragem para enfrentar desafios.

Projeto 1: Sistema cartesiano de aplicação de cola

Camila Queiroz Boa Morte Pereira
(Residente)

Davi Oliveira Santana Carvalho
(Residente)

Hilquias Rodrigues de Oliveira
(Residente)

Mateus Fernandes Santos
(Residente)

Ronaldo Ruiz Filho
(Residente)

No Programa *Embarcatech*, a equipe dedicou-se à automatização da vedação de *cooktops*, processo que utiliza a cola especial PU-55 para garantir a segurança e a durabilidade do produto. O problema central reside na execução estritamente manual dessa tarefa, tornando o resultado dependente da habilidade do operador ao contornar o vidro. Devido à repetitividade e à alta viscosidade do adesivo, a fadiga compromete a padronização do processo, gerando inconsistências na dosagem. Quando há falta de material, aumentam-se os riscos de infiltração e curto-circuito; por outro lado, o excesso exige limpeza manual e retrabalho.

Diante desse cenário, a solução do problema torna-se essencial para assegurar a qualidade técnica do produto e reduzir custos operacionais, uma vez que o desperdício de insumos em uma linha de produção contínua gera prejuízos significativos. Nesse contexto, a equipe propôs a conversão do processo manual em um sistema semiautomático, com maior precisão e repetibilidade, visando garantir autonomia produtiva e uma aplicação uniforme, eficiente e adaptável aos diferentes modelos fabricados pela empresa.

O projeto iniciou a partir de uma proposta externa, cuja diretriz já determinava o uso de um sistema cartesiano para a aplicação de cola. A partir disso, realizaram-se reuniões para o planejamento da arquitetura do sistema. Após a análise de diversas alternativas, optou-se por utilizar uma máquina comercial de Controle Numérico Computadorizado (CNC) e adaptá-la às exigências do projeto. A etapa seguinte consistiu no orçamento e aquisição dos equipamentos, cujo atraso na entrega afetou o cronograma inicial. Contudo, esta eventualidade foi aproveitada para aprofundar o planejamento, o estudo das funcionalidades da CNC e o refinamento das estratégias de execução.

A organização do grupo ocorreu de forma colaborativa, com divisão em dois núcleos principais: um responsável pelo desenvolvimento do *software* e outro pela montagem e testes da parte física e mecânica. Apesar dessa divisão, a equipe manteve uma integração constante, com alinhamento contínuo entre os subgrupos. O trabalho foi conduzido em etapas, priorizando o entendimento do sistema e o planejamento da solução, permitindo avançar de forma estruturada conforme os recursos se tornaram disponíveis.

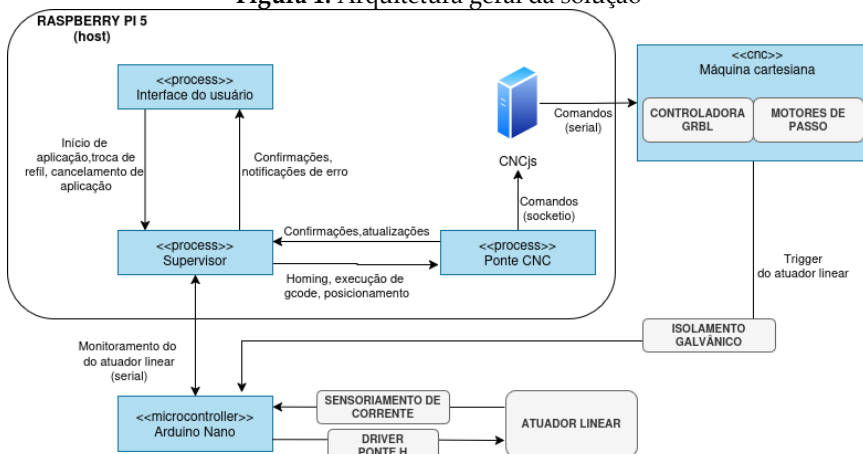
A falta de padronização e o desperdício gerados pela aplicação manual incentivaram o projeto de uma alternativa semiautomatizada, ou seja, capaz de executar as tarefas repetitivas de forma autônoma sob supervisão humana. A pesquisa e prática da equipe guiaram a evolução até a configuração atual da solução: um sistema cartesiano de aplicação de cola. Esse equipamento mecânico se movimenta de forma programada em três eixos, correspondentes aos lados (X), frente/trás (Y) e acima/abaixo (Z), guiando o dosador com precisão milimétrica para garantir a aplicação exata do adesivo.

A arquitetura da solução foi projetada considerando os requisitos operacionais mapeados de tempo, força, velocidade, peso e vazão. A integração entre os módulos de *software* e o *hardware* do sistema é ilustrada pela Figura 1. Com base nesta arquitetura, foi estruturado um fluxo prático bem direto: após o

operador posicionar o *cooktop* na área de trabalho e escolher o modelo, o computador central assume o controle, processando a trajetória e distribuindo sinais de comando. Quaisquer erros e interrupções são devidamente tratados e notificados ao operador. Compreender e construir essa troca de informações entre a interface do usuário, o processamento de dados e a execução física representou uma das etapas de maior aprendizado para a equipe.

A implementação exigiu uma seleção rigorosa de componentes, utilizando a *CNC FoxAlien Router Masuter Pro* como base. No seu eixo Z, acoplou-se um suporte customizado para cola *PU-55* e um atuador linear de 750 N, controlado por um *Arduino Nano* e sensor *ACS712 5A*. A unidade de processamento central, uma *Raspberry Pi 5*, gerencia a interface gráfica e a comunicação com a CNC via servidor *CNCjs* e *Socket.IO*, utilizando *Python* como linguagem principal.

Figura 1: Arquitetura geral da solução



Fonte: Os autores.

Ao final do projeto, a equipe desenvolveu um protótipo funcional para o sistema de aplicação de cola, integrando a CNC, o sistema de controle embarcado e a interface gráfica de operação. O sistema é capaz de executar trajetórias programadas e realizar a aplicação do adesivo de forma sincronizada com o movimento dos

eixos. Apresentado na Figura 2, o protótipo é um mínimo produto viável voltado para validação funcional. O sistema atende aos requisitos de operação, controle e monitoramento, embora demande refinamentos de robustez para a aplicação em ambiente industrial.

Figura 2: Protótipo desenvolvido



Fonte: Os autores.

Os testes foram realizados para validar o funcionamento do protótipo, com foco no controle da vazão e na sincronização do sistema. Os resultados demonstraram aplicação consistente da cola após ajustes iterativos feitos a cada execução, além de estabilidade na comunicação entre os módulos e resposta adequada aos comandos. Essas evidências confirmam a viabilidade da solução proposta.

O principal desafio do projeto consistiu na integração entre os diferentes módulos do sistema, especialmente a sincronização entre o envio de comandos *G-code* e o acionamento do atuador. Pequenas latências impactaram diretamente a qualidade da aplicação da cola, exigindo ajustes finos e empíricos no controle do processo. O controle da vazão de adesivo representou outro ponto crítico devido à viscosidade do material. A solução final exigiu testes práticos e sucessivos ajustes até atingir um comportamento consistente.

O projeto consolidou conhecimentos em sistemas embarcados, automação e integração *hardware-software*, destacando a importância da modularidade e do controle em tempo real. A equipe também desenvolveu habilidades em comunicação entre sistemas, tratamento de falhas e validação prática, entendendo que testes são essenciais para ajustar o comportamento real do sistema. No aspecto organizacional, houve evolução em trabalho em equipe, coordenação e tomada de decisão, especialmente diante de limitações e imprevistos. Ademais, a experiência aproximou a equipe de um problema real da indústria, contribuindo para uma formação mais prática, crítica e alinhada às demandas profissionais.

Embora satisfatório, o protótipo elaborado permite melhorias, como a substituição do atuador linear atual por um modelo de 350 mm. Como o tubo de cola *PU-55* possui 34 cm, um atuador menor e de mesma capacidade atenderia à demanda, reduzindo o peso sobre a CNC e permitindo um suporte de fixação mais compacto. Outro ponto de melhoria consiste na remoção do *Arduino Nano*, transferindo para o *Raspberry Pi 5* a responsabilidade pelo controle e monitoramento do atuador linear. Ambas as modificações possuem potencial de reduzir o custo final do projeto, otimizar o uso do espaço e melhorar o desempenho do sistema.

Apesar da conclusão do projeto, sua aplicação restrita à colagem de *cooktops* limita o aproveitamento em outros segmentos industriais que ainda dependem de processos manuais de colagem. Em função de sua natureza bidimensional e da flexibilidade na definição de trajetórias, a plataforma pode ser adaptada para novas categorias de produtos com ajustes mínimos no *software*, ampliando significativamente seu campo de aplicação. Ainda visando expansão para novos mercados, uma evolução natural do projeto é a incorporação de técnicas de visão computacional e computação gráfica, permitindo ao usuário definir, de forma intuitiva, os pontos de aplicação de cola diretamente sobre a imagem da superfície de trabalho, tornando o sistema mais versátil, preciso e acessível.

Projeto 2: Verificador de Funcionamento de Queimadores em *Cooktop*

Camila de Araújo Bastos
(Residente)

Jackson Silva dos Santos
(Residente)

Lucas Carneiro de Araújo Lima
(Residente)

Luis Felipe Pereira de Carvalho
(Residente)

Mariana da Silva Lima Santos
(Residente)

Na linha de produção de uma fabricante de eletrodomésticos, o controle de qualidade de *cooktops* exige que operadores verifiquem manualmente a ignição de cada queimador antes da aprovação do produto. Por se tratar de uma atividade repetitiva, a fadiga acumulada ao longo da jornada eleva o risco de erros de inspeção. Falhas nesse processo podem resultar em dois cenários problemáticos: falsos positivos, em que produtos defeituosos são aprovados e chegam ao mercado; ou falsos negativos, que provocam retrabalho desnecessário sobre produtos funcionais.

A proposta deste projeto foi desenvolver um sistema capaz de fazer a verificação de queimadores de forma automática, sem alterar o fluxo de trabalho convencional do operador. Esse critério de não intrusividade orientou as escolhas da equipe do começo ao fim.

O ponto de partida foi uma reunião de *kickoff* com o *Product Owner (PO)* da empresa parceira, na qual foram definidos os requisitos do sistema em termos técnicos e operacionais. Com esses critérios em mãos, a equipe organizou o desenvolvimento em *sprints* semanais usando o *Scrum*, a partir da plataforma *Airtable*.

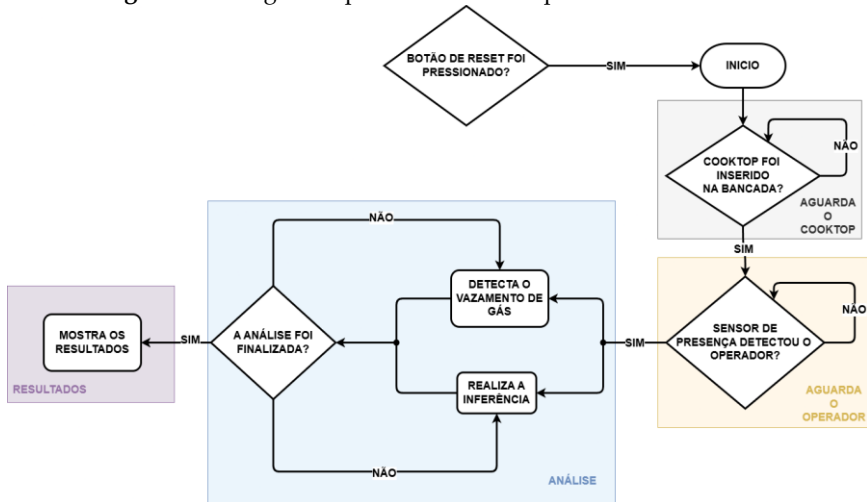
Cada *sprint* tinha um foco claro: desenvolvimento da bancada de testes, configuração do *hardware*, construção do *dataset*, treinamento do modelo e integração dos componentes.

A equipe se dividiu em dois grupos: um focado na obtenção do modelo de detecção de objetos a partir de imagens reais do *cooktop* e outro na integração dos sensores, *display* e lógica do sistema. Na reunião final com o *PO*, o sistema foi aprovado como “atende integralmente aos requisitos estabelecidos”.

O sistema automatiza a inspeção dos queimadores por meio de visão computacional embarcada. Uma *Raspberry Pi 5* atua como controlador central, integrando a *Raspberry Pi AI Camera* com o sensor *Sony IMX500*, capaz de executar inferências de IA (Inteligência Artificial) diretamente no *hardware* da câmera. O modelo de detecção utilizado foi o *YOLOv8n*, escolhido pelo baixo custo computacional e compatibilidade com dispositivos de borda. A interface com o operador é feita por um *display* de 3,5" e um *buzzer* para alertas sonoros. A presença do operador é detectada pelo sensor *E18-D80NK*, e possíveis vazamentos de gás são monitorados por dois sensores *MQ-5*.

Conforme a Figura 1, o sistema funciona de maneira cíclica, iniciando quando o operador é detectado e o *cooktop* está posicionado na bancada, e encerrando quando o operador se afasta. Após a inserção do equipamento, o sistema monitora a presença do operador e, em seguida, executa a inferência e a detecção de gás, compondo a fase de análise, que permanece ativa até a saída do operador ou acionamento do botão de resultado. Ao final, o sistema atualiza o status dos queimadores no *display*. Um botão de *reset* permite reiniciar o sistema a qualquer momento em caso de falhas.

Figura 1: Fluxograma que ilustra o ciclo operacional do sistema



Fonte: Os autores.

A fim de simular o cenário de aplicação real, foi necessário montar uma bancada de teste para integrar os periféricos e um *cooktop*, conforme ilustrado na Figura 2:

Figura 2: Bancada de teste



Fonte: Os autores.

A estrutura é composta por uma bancada que acomoda o *cooktop*, os sensores e os periféricos de forma organizada. Um suporte de perfil de alumínio foi integrado à bancada para fixar a câmera, a *Raspberry Pi 5*, o *buzzer*, o *display* e o sensor de presença. Neste projeto, o suporte garante o posicionamento da câmera acima do centro do *cooktop* e afastada do calor emitido pelos queimadores. Além disso, canaletas laterais foram incorporadas para organizar a fiação.

O sistema foi validado na bancada, demonstrando ciclos completos de operação: inserção do *cooktop* e detecção do operador, monitoramento em tempo real dos queimadores, exibição dos resultados no *display* e acionamento do alerta sonoro em caso de vazamento de gás. A tela de resultados indica, para cada queimador, se foi aceso (aprovado) ou não (reprovado), além do status do gás.

Um dos obstáculos centrais foi definir a lógica de classificação do estado dos queimadores: um queimador apagado no momento de uma leitura não é necessariamente um defeito, pois pode ainda não

ter sido acionado pelo operador. A solução adotada foi registrar se cada queimador foi aceso em algum momento durante todo o ciclo de monitoramento, e não se permanece continuamente aceso.

Outro desafio relevante foi o mapeamento espacial das detecções. Embora o modelo identifique os queimadores na imagem, ele não distingue automaticamente suas posições no *cooktop*, como superior esquerdo ou inferior direito. Para contornar isso, as coordenadas centrais de cada *bounding box* detectada foram comparadas com pontos de referência previamente definidos, adotando uma tolerância espacial de 50 *pixels*. Com essa estratégia, foi possível associar corretamente cada detecção ao queimador físico correspondente e exibir o estado adequado no *display*.

No desenvolvimento do *software*, a principal dificuldade foi garantir a execução simultânea da inferência da câmera, da leitura dos sensores e da atualização do *display* sem causar travamentos ou perda de desempenho. Para resolver isso, a equipe aprofundou seus estudos em paralelismo e programação concorrente em *Python*, chegando a uma arquitetura baseada em *threads* com sincronização por eventos compartilhados.

O projeto marcou o primeiro contato prático da equipe com visão computacional e inteligência artificial aplicadas a sistemas embarcados. Mais do que os aprendizados técnicos, como redes neurais convolucionais, detecção de objetos, programação concorrente em *Python*, desenvolvimento de Placa de Circuito Impresso e integração de *hardware*, a experiência também contribuiu para o desenvolvimento de habilidades profissionais importantes, como trabalho em equipe, gestão colaborativa, comunicação com um cliente real, adaptação a imprevistos e apresentação de soluções em formato de *pitch*.

Três caminhos se destacam como continuidade natural do projeto. O primeiro é substituir o botão de confirmação manual por um sensor que detecta o posicionamento do *cooktop* automaticamente, eliminando a única interação extra que ainda existe no fluxo. O segundo é implementar o registro dos resultados por ciclo, criando um histórico que permita acompanhar a

qualidade ao longo da produção. O terceiro é ampliar a análise para além do estado aceso/apagado, avaliando também estabilidade e uniformidade da chama.

Projeto 3: Parafusadeira CNC

Caique de Brito Freitas
(Residente)
Cláudio Evaristo Júnior
(Residente)
Davi Leão
(Residente)
Hugo Martins Santana
(Residente)
Naylane do Nascimento Ribeiro
(Residente)

O projeto surgiu de um problema recorrente em processos de montagem: o aperto manual ou semiautomático de parafusos pode gerar variações de execução, comprometendo a padronização, aumentando o retrabalho e dificultando a rastreabilidade. Diante disso, a equipe propôs adaptar uma estrutura de Controle Numérico Computadorizado (CNC) para automatizar essa etapa, buscando maior repetibilidade, segurança operacional e possibilidade de registro de dados.

A proposta central foi validar o uso de uma CNC comercial em ciclos automatizados de aparafusamento, com foco em segurança, repetibilidade e potencial de evolução. Esse objetivo orientou as decisões técnicas ao longo do desenvolvimento.

A equipe iniciou a residência com conhecimentos prévios em programação e eletrônica, mas com pouca experiência prática na integração entre *hardware* e *software* em sistemas embarcados. Ao longo do programa, a formação técnica foi ampliada por palestras e aulas, o que se refletiu diretamente nas experimentações realizadas pela equipe e no amadurecimento de seus conhecimentos. A execução do projeto, por sua vez, consolidou

esse processo ao exigir adaptação, autonomia e aplicação prática do que foi aprendido.

O desenvolvimento da solução ocorreu de forma incremental, com organização por etapas: levantamento de requisitos, adaptação mecânica, integração eletrônica, desenvolvimento da interface, implementação dos *logs* e testes funcionais. Essa abordagem ajudou a reduzir riscos e ajustar o escopo conforme surgiam limitações técnicas.

Durante o processo, também foi necessário redefinir prioridades. Algumas ideias previstas inicialmente foram reduzidas para garantir a entrega de um núcleo funcional do protótipo.

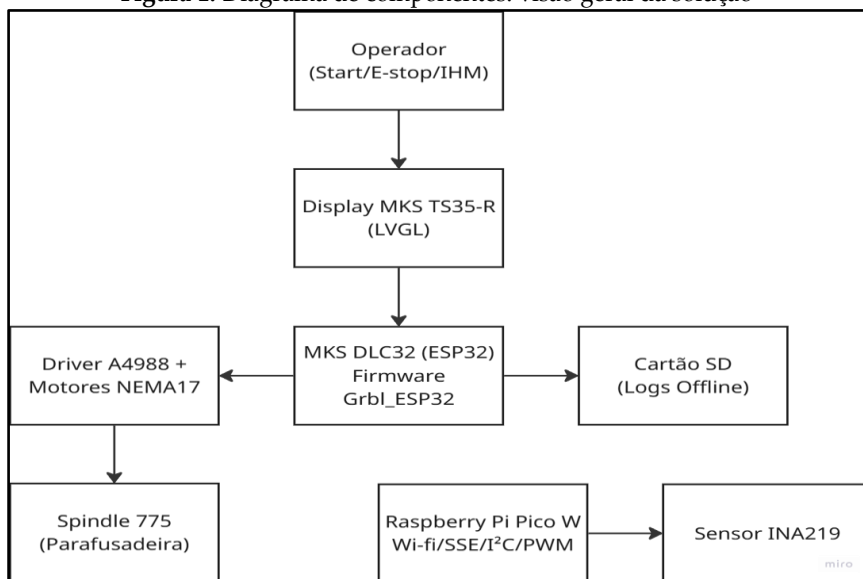
A solução consistiu na adaptação de uma máquina CNC comercial para servir de base a uma parafusadeira automatizada. A estrutura mecânica foi aproveitada para a movimentação nos eixos X, Y e Z, permitindo concentrar esforços na integração eletrônica e no desenvolvimento embarcado.

O controle principal foi realizado por uma placa baseada em *ESP32*, com uso de comandos em *G-code* para definir movimentos e sequências operacionais. Como parte da arquitetura proposta, também foi previsto o uso de um *Raspberry Pi Pico W* para apoiar funções de monitoramento e comunicação. No entanto, essa integração não pôde ser validada integralmente no protótipo final, pois não foi possível estabelecer comunicação estável entre os módulos de *hardware* dentro do tempo disponível. A estrutura do protótipo foi projetada para integrar movimentação, interface com o operador, registro de eventos e possibilidades de expansão.

Foi desenvolvida uma interface homem-máquina em monitor tátil para exibir informações operacionais, coordenadas e mensagens do sistema. Também foi implementado um sistema de registro *offline* em cartão de memória, além de um botão de parada de emergência como elemento central de segurança. Houve ainda testes com leitura de corrente elétrica para acompanhar o esforço do motor durante o aparafusamento, visando futuras melhorias no controle de torque.

A Figura 1 mostra um diagrama de componentes apresentando a visão geral da solução desenvolvida.

Figura 1: Diagrama de componentes: visão geral da solução



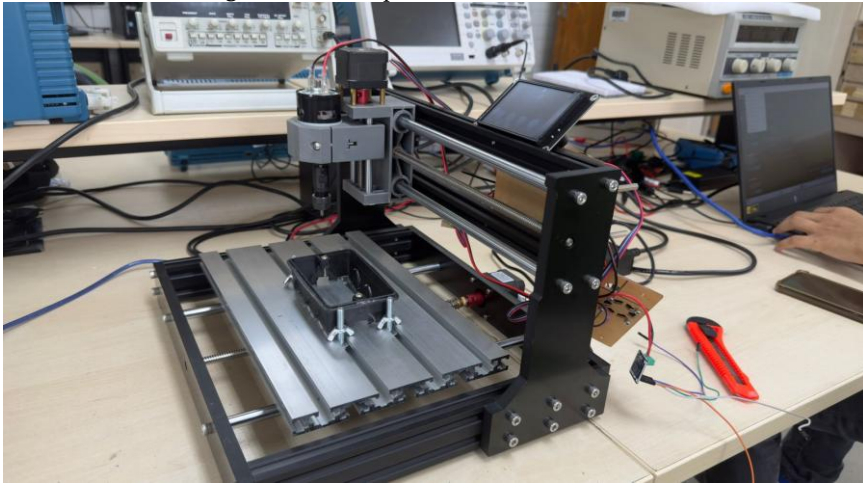
Fonte: Os autores.

Ao final do projeto, a equipe consolidou um protótipo funcional de automação parcial do processo de aparafusamento, apresentado na Figura 2. O sistema passou a executar movimentação coordenada nos eixos, posicionar a ferramenta, registrar eventos em cartão SD, fornecer retorno visual ao operador e responder corretamente à parada de emergência.

Os testes contemplaram movimentação dos eixos, execução de comandos, funcionamento da interface, registro de *logs*, leitura de corrente e validação da parada de emergência. Mesmo sem instrumentação industrial avançada, os resultados indicaram a viabilidade técnica da proposta como prova de conceito.

Em termos de escopo, as funções centrais foram consolidadas, embora algumas frentes tenham permanecido parciais, sobretudo, aquelas que exigiam maior refinamento mecânico ou estabilidade de integração entre módulos.

Figura 2: Protótipo funcional desenvolvido



Fonte: Os autores.

O principal desafio técnico esteve no controle de torque. A equipe tentou utilizar a leitura de corrente elétrica para acompanhar o esforço do motor durante o aperto, mas encontrou limitações como ruído e instabilidade no comportamento do protótipo. Ainda assim, essa etapa foi importante para indicar limites da abordagem adotada e orientar melhorias futuras.

Outro obstáculo relevante foi a comunicação entre os módulos eletrônicos, especialmente nas tentativas de integrar monitoramento e telemetria com maior estabilidade. Além disso, a alimentação automática de parafusos foi identificada como uma funcionalidade desejável, mas incompatível com o tempo disponível, por exigir esforço mecânico adicional. Assim, o grupo priorizou a entrega de um protótipo funcional, seguro e demonstrável.

O projeto contribuiu para a formação da equipe ao integrar conhecimentos de eletrônica, programação, sistemas embarcados e testes práticos. Ao longo do desenvolvimento, os integrantes precisaram diagnosticar falhas, validar hipóteses, revisar decisões e lidar com restrições reais de tempo e escopo, fortalecendo

competências técnicas e habilidades de organização, colaboração e tomada de decisão.

A experiência também evidenciou que o desenvolvimento de uma solução embarcada depende fortemente da integração entre módulos. Isso ampliou a visão da equipe sobre a construção de sistemas aplicados e reforçou a importância de planejamento, documentação e priorização.

Como continuidade, o projeto pode avançar em três frentes principais: aperfeiçoamento do controle de torque, consolidação da comunicação entre módulos e implementação da alimentação automática de parafusos. Esses avanços tendem a ampliar a autonomia e a robustez do protótipo.

Mesmo com essas pendências, o projeto validou a adaptação de uma CNC comercial para o aparafusamento automatizado e consolidou uma base técnica capaz de orientar versões futuras mais completas.

Projeto 4: *SafeWay* - Sistema Embarcado para Controle de Acesso com *RFID* e *Cache Offline*

Anna Beatriz S. Lima
(Residente)

Carlos Henrique O. Valadão
(Residente)

Daniel P. Braz
(Residente)

Lorenzo Giuseppe O. Baroni
(Residente)

Elmer C. de Oliveira Filho
(Residente)

A proposta teve origem na identificação de vulnerabilidades em sistemas de controle de acesso físico dependentes de conexões externas. Nesse contexto, foi desenvolvido o *SafeWay*, um sistema embarcado capaz de processar autorizações localmente, eliminando a necessidade de comunicação com a nuvem para liberação de acesso. O sistema utiliza identificação por cartões e regras de restrição por horários, garantindo operação segura mesmo em cenários de falha de rede ou interrupções de energia.

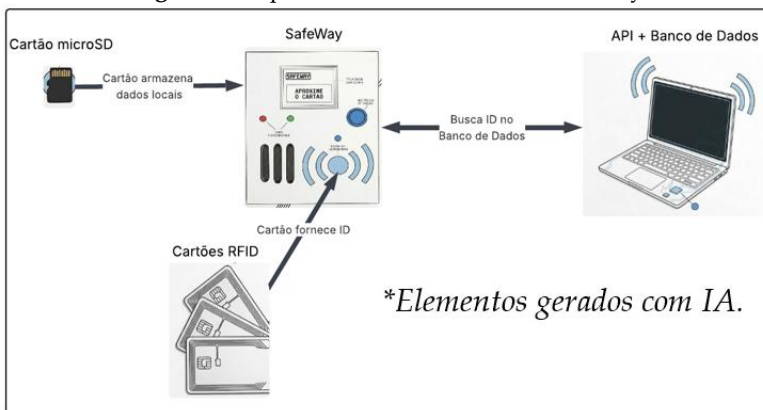
A solução caracteriza-se como uma arquitetura de computação em borda resiliente, adequada a ambientes que demandam alta disponibilidade e operação contínua. Dessa forma, a abordagem adotada reduz a dependência de infraestrutura externa, promovendo um controle de acesso mais autônomo, robusto e confiável.

Superando a base teórica e as experiências iniciais com placas didáticas, evoluímos para a montagem real e a organização de tarefas simultâneas. Adotamos métodos ágeis com ciclos de duas semanas para ajustar rotas e dividir responsabilidades conforme

surgiam falhas físicas de comunicação. Aprendemos a ser persistentes na investigação de erros e na gestão eficiente de recursos do sistema. O maior legado foi o amadurecimento profissional e a conquista de uma visão sistêmica da engenharia. O processo transformou uma demanda complexa em um protótipo físico funcional e resiliente.

A arquitetura do *SafeWay* nasceu de um processo coletivo de criação, no qual cada detalhe foi planejado pelo grupo para transformar conceitos teóricos em um protótipo físico e funcional. Tirar o projeto do papel exigiu que fosse integrado, dentro de um único terminal, componentes como o leitor de cartões, uma tela interativa e sinalizações por luzes e sons que guiam o usuário em cada tentativa de acesso. A inclusão de um cartão de memória interno permite que o dispositivo armazene dados locais e opere com total autonomia, garantindo a segurança mesmo em momentos de isolamento da rede. Com isso, a arquitetura foi moldada até chegar no nível que se apresenta no diagrama da Figura 1, a seguir.

Figura 1: Arquitetura Geral do sistema SafeWay



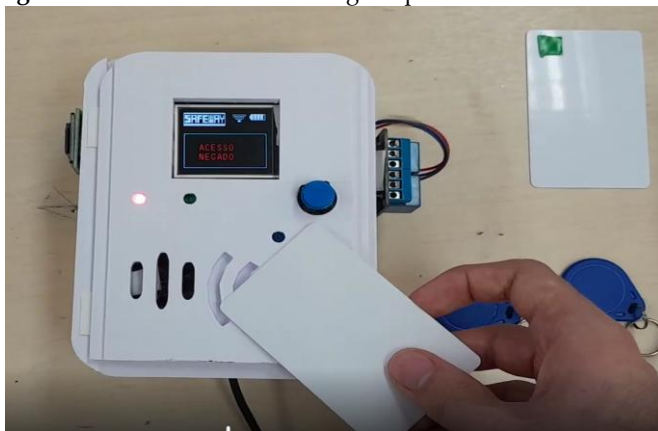
Fonte: Os autores.

Conforme apresentado, a estrutura interna do sistema é complementada pela integração com a arquitetura externa, composta por um banco de dados e um sistema de gestão

centralizado em um *laptop* (a exemplo), em que o terminal busca as identificações nessa base externa para manter as permissões sempre atualizadas, permitindo um controle inteligente e auditável de todo o fluxo de pessoas. Ver a harmonia entre o *hardware* que montamos e a inteligência que reside fora dele foi o ponto alto da nossa jornada, simbolizando o sucesso de um sistema que pensamos e construímos do zero.

O *SafeWay* constitui um protótipo funcional que integra *hardware* embarcado a um sistema de gestão autônoma, evidenciando a viabilidade técnica da solução proposta. O dispositivo, protegido por um invólucro em impressão 3D, opera de forma independente com bateria de emergência e tela interativa. Como ilustrado na Figura 2, o sistema oferece respostas imediatas, validando permissões localmente sem depender da rede. O resultado é um conjunto robusto e modular, preparado para garantir segurança e agilidade em diversos cenários. Essa entrega materializa a solidez técnica e o compromisso da equipe com uma solução de engenharia completa.

Figura 2: Tentativa de Acesso Negada pelo Sistema ao ler cartão *RFID*



Fonte: Os autores.

A conclusão do sistema representou o encerramento do ciclo de desenvolvimento, marcado pela integração entre aprendizado

técnico e trabalho colaborativo. A etapa de testes uniu teoria e colaboração para resolver desafios reais, como garantir a continuidade das autorizações mesmo com o sistema desconectado da rede. Verificamos que a resposta ao usuário é extremamente ágil, ocorrendo em menos de três segundos, o que validou o esforço coletivo em otimizar o funcionamento do dispositivo. A estabilidade sob estresse e a autonomia de bateria superior a duas horas e meia demonstraram nossa capacidade de trabalhar de forma resiliente e síncrona. Este protótipo representa o amadurecimento de uma equipe que superou dificuldades iniciais para entregar uma solução de engenharia sólida.

Dentre os desafios, enfrentamos o problema de garantir que o sistema nunca travasse ao buscar conexão, o que exigiu um cuidado minucioso com a organização do *software*. Tivemos de integrar diversos componentes para que operassem em harmonia, superando conflitos de comunicação interna entre as peças. Limitações de tempo e materiais nos forçaram a priorizar a robustez da segurança, adaptando a montagem física para um formato manual e criativo. Foram esses, alguns dos ajustes fundamentais para tornar o sistema leve e estável, transformando obstáculos em aprendizado sobre resiliência e eficiência.

O projeto demonstrou a importância da autonomia em sistemas de segurança, priorizando o funcionamento local de forma independente. A prática ensinou a equilibrar limitações de *hardware* com a necessidade de um código organizado e estável. Através do trabalho em equipe e de métodos de gestão, aprendemos a planejar atividades sob prazos reais e restrições de materiais. O suporte de mentorias auxiliou na transição de uma visão acadêmica para uma postura profissional orientada ao desenvolvimento de produtos.

Após esta longa jornada, chegou-se ao fim o desenvolvimento do *SafeWay*, que demonstrou um projeto completo e funcional. Mas o fato é que ele ainda tem potencial para muitas melhorias que podem ser feitas, dentre elas a integração de novas formas de identificação para tornar o acesso ainda mais seguro. A inclusão de

tecnologias como o reconhecimento facial permitiria uma camada extra de proteção, garantindo maior confiabilidade na verificação de quem circula pelo ambiente. No aspecto físico, o próximo passo envolve a evolução do protótipo para uma versão definitiva, utilizando placas de circuito industrializadas e testando a resistência do conjunto em condições reais de uso contínuo.

Desta forma, o *SafeWay* permanece como uma base sólida para futuras adaptações e integrações com outros sistemas de gestão, refletindo a maturidade técnica alcançada ao longo de todo o desenvolvimento.

Polo Ilhéus/Itabuna

Aline Silva Ramos

(Mentora)

Fábio Oliveira Silva

(Mentor)

O Centro Multidisciplinar de Bom Jesus da Lapa, vinculado à Universidade Federal do Oeste da Bahia (UFOB), está localizado na região do Médio São Francisco, na porção Oeste do estado da Bahia. Nessa unidade são ofertados os cursos de Engenharia Elétrica e Engenharia Mecânica, cada um com 45 vagas em regime de período integral, atendendo estudantes provenientes de diversos municípios baianos e de outros estados da federação.

O *campus* conta com uma infraestrutura laboratorial voltada ao ensino e à pesquisa, incluindo o Complexo Multi Laboratorial de Engenharia Mecânica e o pavilhão Laboratorial de Engenharia Elétrica, que abrange os Laboratórios de Máquinas Elétricas, Energias Renováveis, Automação e Controle, Eletrotécnica, Sistemas Digitais e Eletrônica Analógica.

No contexto do programa *EmbarcaTech*, o polo contou com a participação de 20 estudantes, dos quais 17 integraram a trilha de Sistemas Embarcados. O perfil do grupo era diversificado: a maioria dos participantes cursava Engenharia Elétrica, em estágios que variavam entre o 5º e o 10º semestre, mas a turma também incluía um aluno de Ensino Médio, dois estudantes do curso de Gestão da Tecnologia e Inovação e uma aluna de Mestrado em Engenharia Elétrica e de Computação. Essa heterogeneidade de formações e níveis de maturidade acadêmica representou, ao mesmo tempo, um desafio e uma riqueza para o processo de

aprendizagem, pois favoreceu trocas de conhecimento entre perfis complementares.

Os estudantes foram organizados em quatro equipes, compostas de modo a equilibrar competências distintas dentro de cada grupo: alunos com maior afinidade em *hardware* e eletrônica foram mesclados com aqueles de perfil mais voltado a *software* e programação. Essa estratégia de composição buscou reproduzir a dinâmica de equipes multidisciplinares encontrada em ambientes profissionais e contribuiu para que os integrantes desenvolvessem habilidades além de suas zonas de conforto.

As atividades de mentoria foram realizadas nos Laboratórios de Máquinas Elétricas e de Eletrônica Analógica, com encontros presenciais aos sábados pela manhã durante a Residência Tecnológica. Por se tratar de um *campus* de pequeno porte e considerando que a maioria dos participantes residia no próprio município, os estudantes também dispunham de acesso aos laboratórios ao longo da semana, o que ampliou significativamente o tempo disponível para o desenvolvimento dos projetos.

A atuação dos mentores concentrou-se em três frentes articuladas. A primeira consistiu no acompanhamento das tarefas solicitadas pelos instrutores, oferecendo reforço dos conteúdos ministrados nas aulas remotas às terças-feiras e suporte direto às implementações práticas. A segunda frente envolveu a realização de aulas complementares, abrangendo desde fundamentos de eletrônica até temas mais avançados, passando pelo uso adequado de equipamentos de bancada, como multímetro, osciloscópio, gerador de sinais e fontes de alimentação e pela prototipação de aplicações com sensores e atuadores. A terceira frente, de natureza mais transversal, dedicou-se à orientação metodológica das equipes: organização das atividades, gestão de escopo e tempo, e articulação entre os conhecimentos teóricos das aulas remotas e a prática laboratorial.

Essa articulação entre o conteúdo remoto e a vivência presencial evidenciou-se um dos aspectos mais relevantes da mentoria. Os encontros presenciais funcionavam como espaço de

consolidação: os conceitos apresentados nas aulas síncronas ganhavam concretude quando os estudantes os aplicavam nos circuitos, nos microcontroladores e nos protótipos em construção. O papel do mentor, nesse contexto, era menos o de transmissor de conteúdo e mais o de mediador entre a teoria e a prática, ajudando as equipes a identificar quais conhecimentos eram necessários para avançar em cada etapa dos projetos.

No início da residência, observou-se que muitos estudantes apresentavam dificuldades na consolidação de conhecimentos básicos, especialmente em programação, eletrônica e integração de *hardware* com *software*. Aspectos como gestão do tempo, organização das atividades e trabalho em equipe também se mostraram desafiadores, sobretudo em um contexto híbrido de aprendizagem que exigia maior autonomia. A diversidade do público contribuiu para diferentes ritmos de aprendizagem dentro das equipes, demandando dos mentores atenção individualizada e estratégias diferenciadas de acompanhamento.

Outro desafio significativo esteve associado à adaptação à metodologia de Aprendizagem Baseada em Projetos (ABP). Muitos estudantes não estavam familiarizados com abordagens centradas no protagonismo do aluno e, inicialmente, demonstraram dificuldades na identificação de problemas, na formulação de hipóteses e na proposição de soluções de forma estruturada. A utilização simultânea de múltiplas estratégias pedagógicas, como ensino a distância, encontros síncronos e atividades presenciais, também demandou dos alunos maior disciplina e engajamento contínuo.

A localização do polo no interior do estado impôs ainda um desafio de ordem logística: a ausência de lojas especializadas em componentes eletrônicos na região. Diferentemente dos polos situados em grandes centros urbanos, onde sensores, módulos e componentes discretos podem ser adquiridos com facilidade e agilidade, em Bom Jesus da Lapa toda a obtenção de materiais dependia de compras online, sujeitas a prazos de entrega prolongados e, por vezes, a extravios ou atrasos. Essa limitação

exigiu das equipes um planejamento antecipado mais rigoroso e, em alguns momentos, criatividade para adaptar soluções com os recursos disponíveis, o que também contribuiu para o desenvolvimento da capacidade de resolver problemas sob restrições reais.

Apesar desses desafios, ao longo do programa foi possível perceber uma evolução significativa dos participantes. Gradualmente, os estudantes desenvolveram maior autonomia na busca por soluções, aprimoraram suas habilidades técnicas e passaram a demonstrar maior capacidade de integração entre teoria e prática. Observou-se também uma melhora expressiva no trabalho colaborativo, na comunicação interpessoal e na organização das atividades em grupo. Alunos que no início da residência dependiam fortemente da orientação dos mentores passaram a tomar decisões técnicas com mais segurança e a conduzir etapas do projeto de forma independente.

A utilização de práticas como gamificação contribuiu positivamente para o engajamento, estimulando a participação ativa e a continuidade no programa. Os encontros presenciais e as atividades práticas nos laboratórios foram fundamentais para consolidar o aprendizado, proporcionando experiências concretas e alinhadas às demandas do mercado de trabalho. O amadurecimento mais visível, contudo, não foi apenas técnico: os estudantes passaram a lidar melhor com frustrações, a negociar soluções dentro das equipes e a encarar problemas inesperados como parte natural do processo de desenvolvimento.

As quatro equipes do polo desenvolveram projetos que refletem a diversidade de aplicações possíveis na área de sistemas embarcados e Internet das Coisas. O primeiro projeto consistiu em um sistema de rastreamento industrial por *RFID*, voltado ao controle de produtos em tempo real no ambiente fabril. O segundo projeto, denominado *Environmental Monitor*, propôs uma estação de monitoramento ambiental de baixo custo baseada em *edge computing*, destinada à proteção de equipamentos sensíveis em ambientes industriais. O terceiro projeto desenvolveu um Veículo

Autoguiado (AGV) controlado por comandos de voz via *Amazon Alexa*, integrando navegação autônoma por seguimento de linha com comunicação em tempo real via *AWS*. O quarto projeto, *IR Remote*, criou um controle universal *smart* para ar-condicionado por infravermelho, com interface *web* e funcionalidades de temporização programável.

A experiência do polo Bom Jesus da Lapa evidencia que o *EmbarcaTech* cumpre com eficácia seu propósito de formação técnica e socioemocional, e revela algo ainda mais significativo: o potencial transformador da interiorização do desenvolvimento tecnológico. Para muitos dos estudantes que participaram do programa, esta foi uma oportunidade única, a primeira vivência de um ciclo completo de desenvolvimento de produto em parceria com empresas reais, algo que, até então, parecia acessível apenas a quem estivesse nos grandes centros. O programa demonstrou que é possível formar profissionais de alto nível em regiões distantes dos polos tecnológicos tradicionais, desde que haja infraestrutura adequada, orientação qualificada e, sobretudo, a decisão institucional de levar oportunidades a quem delas mais necessitam.

Os desafios enfrentados, como a escassez de componentes eletrônicos na região e a heterogeneidade das equipes, foram parte essencial do processo formativo. Foi justamente ao lidar com restrições reais que os estudantes mais se desenvolveram, não apenas tecnicamente, mas como profissionais capazes de tomar decisões, negociar soluções e trabalhar colaborativamente. Ao final da residência, o polo entregou não apenas quatro projetos funcionais, mas também a evidência concreta de que investir na formação tecnológica no interior é investir no futuro de uma geração inteira de profissionais que, de outra forma, poderiam não ter tido essa chance.

Projeto 1: *Conectsim* - Comunicação NB-IOT com módulo SIM7000G

Ariel Pina Ribeiro
(Residente)

Davi Dantas Mendez Sanchez
(Residente)

Emyle Santana da Silva
(Residente)

João Vitor de Siqueira Santana Costa
(Residente)

Contexto e Motivação

A expansão da Internet das Coisas (*IoT*) tem impulsionado a demanda por dispositivos conectados capazes de operar em longas distâncias com baixo consumo energético. Tecnologias como o *NB-IoT* vêm se consolidando como soluções relevantes nesse cenário, sendo amplamente utilizadas em aplicações como medição inteligente e rastreamento de ativos. Esse crescimento evidencia a necessidade de soluções cada vez mais eficientes e escaláveis no desenvolvimento de sistemas embarcados conectados.

Nesse contexto, a empresa parceira, visando o aceleração da criação de novas soluções, apresenta ao grupo um problema recorrente no desenvolvimento de aplicações *IoT*: a complexidade da comunicação entre microcontroladores e módulos celulares, como o *SIM7000G*. Essa comunicação é baseada em comandos *AT*, um protocolo textual que exige controle rigoroso de tempo, tratamento de erros e gerenciamento de respostas assíncronas. A implementação de funcionalidades mais avançadas, como conexões seguras via *MQTT* ou modos de economia de energia, torna esse processo ainda mais complexo, aumentando a

dificuldade de desenvolvimento. Além disso, a necessidade de um código específico para atender às particularidades de comunicação serial de cada plataforma é um desafio notório no processo de desenvolvimento. Esse cenário gera impactos diretos no tempo de produção e dificuldade na manutenção e portabilidade do sistema.

Dessa forma, o problema abordado pelo grupo não se limita apenas à conectividade entre dispositivos, mas à ineficiência no processo de desenvolvimento de soluções *IoT*. Esse desafio se mostra relevante por afetar diretamente a produtividade das equipes, a qualidade dos sistemas desenvolvidos e a capacidade de adaptação a novas tecnologias.

A formação do grupo foi marcada por uma forte multidisciplinaridade, reunindo estudantes com bases voltadas tanto para a engenharia de *software* quanto para a engenharia elétrica. Para atender ao desafio proposto pela empresa parceira, que demandava uma biblioteca de conectividade escalável e modular para seus produtos *IoT*, a equipe organizou-se de forma colaborativa.

As etapas de desenvolvimento ocorreram de forma progressiva. A fase inicial foi dedicada à superação da curva de aprendizado: os integrantes focados em *software* precisaram mergulhar nas limitações físicas de um microcontrolador e na interpretação de *datasheets* complexos do módulo *SIM7000G*; simultaneamente, quem possuía conhecimento prévio em *hardware* precisou se adaptar aos paradigmas de programação, compreendendo a importância de estruturar uma Camada de Abstração de *Hardware* (*HAL*).

Vencida essa barreira inicial, a ideia do projeto evoluiu rapidamente. A equipe passou da construção da comunicação serial básica (*UART*) para o desenvolvimento dos módulos funcionais independentes, culminando na implementação dos protocolos de rede (*NB-IoT*) e de segurança (*TLS*). A troca contínua de experiências permitiu que o conhecimento previamente dividido se fundisse, convertendo uma equipe com formações

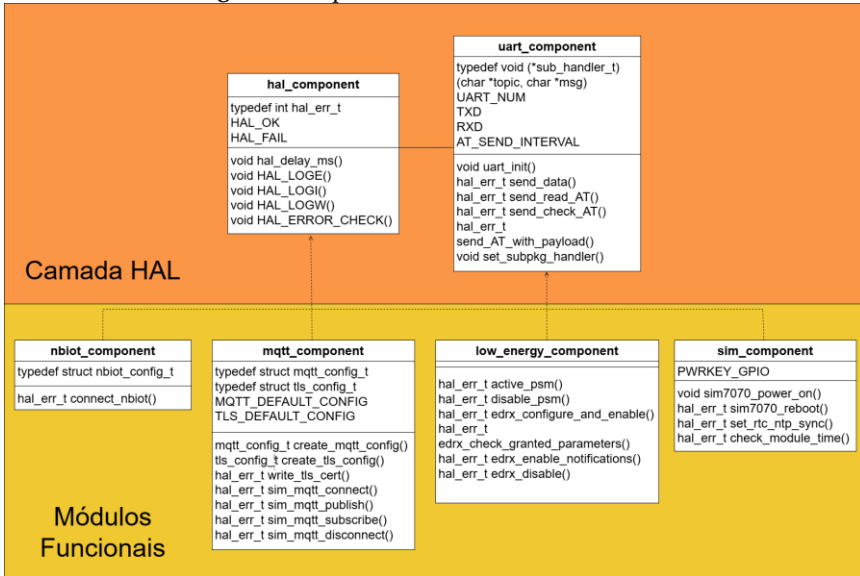
distintas em um time congruente, com uma visão geral sobre o desenvolvimento de projetos.

A solução desenvolvida consiste na criação de uma biblioteca de *software* modular e reutilizável, voltada para facilitar a comunicação entre microcontroladores e módulos celulares *NB-IoT*, com foco no modelo *SIM7000G*. O projeto busca abstrair a complexidade inerente ao uso de comandos *AT*, oferecendo ao desenvolvedor uma interface mais simples e intuitiva para a implementação de funcionalidades de conectividade em aplicações embarcadas.

De forma geral, a biblioteca permite que dispositivos realizem conexão com redes *NB-IoT*, estabeleçam comunicação com *brokers MQTT* (com suporte à segurança via *TLS*) e operem em modos de baixo consumo de energia. Além disso, a solução foi projetada com base na criação de uma Camada de Abstração de *Hardware (HAL)*, responsável por desacoplar a lógica de comunicação das particularidades de cada microcontrolador. Isso garante reutilização para diferentes plataformas de *hardware* sem a necessidade de reescrita de código em novos projetos. De modo a garantir maior portabilidade, o projeto foi desenvolvido utilizando a linguagem C, amplamente empregada em sistemas embarcados.

A arquitetura do sistema foi organizada de forma modular, separando claramente as responsabilidades entre diferentes componentes, como demonstrado na Figura 1. Um módulo central é responsável pela comunicação básica com o módulo *SIM7000G*, incluindo o envio de comandos *AT*, leitura de respostas e tratamento de mensagens assíncronas. A partir dessa base, foram desenvolvidos módulos funcionais independentes, cada um dedicado a uma funcionalidade específica, como conexão *NB-IoT*, comunicação *MQTT* e gerenciamento de modos de economia de energia.

Figura 1: Arquitetura da biblioteca *ConectSim*



Fonte: Os autores.

Ao final do desenvolvimento, foi construída uma biblioteca modular capaz de gerenciar a comunicação entre microcontroladores e o módulo *SIM7000G*, abstraindo a complexidade dos comandos *AT* e oferecendo funções de alto nível para o desenvolvedor. Foram implementadas funcionalidades essenciais, como a conexão à rede *NB-IoT*, o estabelecimento de comunicação com brokers *MQTT* com suporte a *TLS* e o controle de modos de economia de energia.

O protótipo encontra-se em estágio funcional (*MVP*), tendo sido validado em uma plataforma baseada em *ESP32*. A biblioteca demonstrou ser capaz de executar, de forma integrada, as principais operações previstas, incluindo o envio de dados em formato *JSON* para um broker *MQTT*.

Foram realizados testes práticos para verificar a estabilidade da conexão, o correto envio de telemetrias e o comportamento dos modos de baixo consumo. Esses testes serviram como evidência da viabilidade da solução, ainda que ajustes e otimizações adicionais sejam possíveis em versões futuras.

Durante o desenvolvimento, o grupo enfrentou desafios técnicos significativos, especialmente relacionados ao controle rigoroso de tempo e tratamento de respostas assíncronas da comunicação via comandos *AT*. Além disso, a necessidade de lidar com mensagens não solicitadas (*URCs*) trouxe uma camada adicional de complexidade, exigindo uma abordagem mais robusta para garantir a confiabilidade da comunicação. Os integrantes do grupo também apresentavam pouca familiaridade com a leitura e interpretação de *datasheets* e manuais especializados, que são, em geral, extensos e pouco didáticos. Essa dificuldade impactou diretamente o ritmo inicial do desenvolvimento, demandando um esforço adicional de estudo e adaptação para que fosse possível extrair informações relevantes e aplicá-las corretamente no projeto.

O desenvolvimento do projeto proporcionou aprendizados significativos em comunicação serial, uso de comandos *AT*, integração com redes *NB-IoT* e implementação de protocolos como *MQTT* e *TLS* em sistemas embarcados. Além dos aspectos técnicos, o grupo aprimorou o trabalho em equipe, organização e comunicação no contexto de desenvolvimento de *software*. A resolução de problemas complexos, aliada ao contato com desafios reais de mercado, fortaleceu a capacidade de análise e impactou diretamente a formação dos integrantes.

Como continuidade do projeto, pretende-se aprimorar a biblioteca com foco em desempenho, robustez e facilidade de uso. Entre as melhorias possíveis, destacam-se a otimização do consumo de recursos, a ampliação da cobertura de funcionalidades e a melhoria da documentação para facilitar sua adoção por outros desenvolvedores.

Adicionalmente, a expansão da compatibilidade da biblioteca para incorporação de suporte a outros protocolos ou serviço de nuvem representa uma evolução natural do projeto. Desse modo, o projeto se apresenta não apenas como um protótipo funcional, mas como uma base sólida para futuras aplicações em sistemas *IoT*.

Projeto 2: Desenvolvimento de Manipulador Robótico para Automação de Paletização Logística

Gabriel Marques de Andrade
(Residente)

Jabson Gama Santana Júnior
(Residente)

Leonam Sousa Rabelo
(Residente)

Lucas Gabriel Ferreira
(Residente)

Com o avanço das tecnologias industriais, a eficiência nos processos logísticos tornou-se um fator essencial para a competitividade. Nesse contexto, observou-se que a organização manual de caixas sobre paletes ainda representa um ponto crítico, pois além de reduzir a produtividade, expõe os trabalhadores a esforços físicos que podem causar desconforto e lesões. Diante desse cenário, propôs-se o desenvolvimento de uma maquete de uma solução automatizada de baixo custo, capaz de organizar as cargas de forma segura e estável, ao mesmo tempo em que contribui para a preservação da saúde dos trabalhadores.

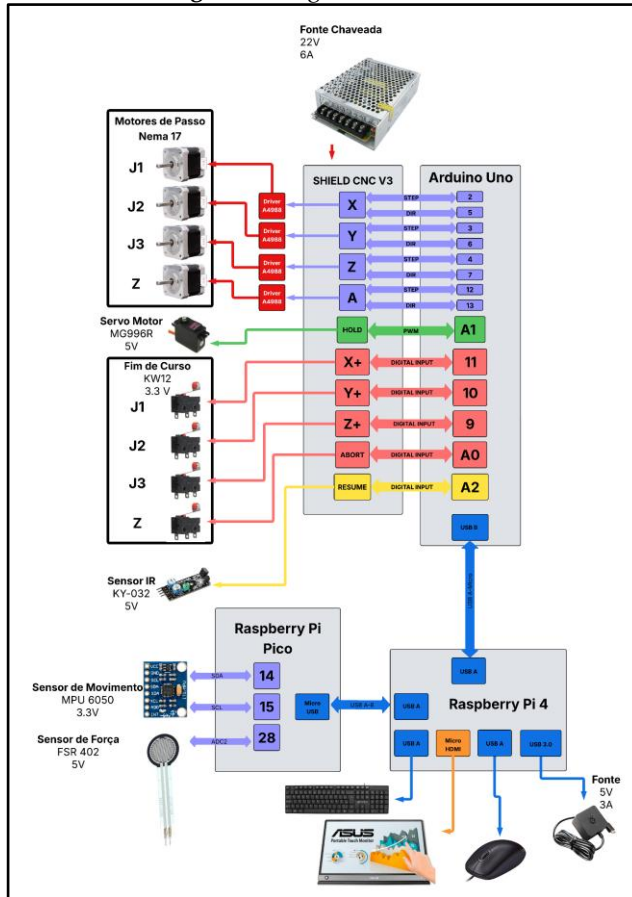
A equipe foi formada por estudantes de Ciência da Computação, Engenharia Mecânica e Engenharia Elétrica, unindo saberes em manufatura aditiva, eletrônica de potência e lógica de programação. O percurso foi marcado por uma mudança de planejamento: o plano inicial previa uma estrutura cartesiana similar às impressoras 3D e máquinas CNC.

Contudo, simulações em *software* revelaram que a estrutura não seria sólida nem confiável, inviabilizando a precisão e gerando um produto final inferior em termos de aplicabilidade real. Através de reuniões e estudos, foi possível converter os materiais já

adquiridos em outro projeto, migrando para a configuração de um braço robótico do tipo SCARA (*Selective Compliance Assembly Robot Arm*). A organização modular e o uso da metodologia PBL (Aprendizagem Baseada em Problemas) permitiram que o grupo nivelasse os conhecimentos.

A solução proposta consiste no desenvolvimento de um braço robótico projetado para realizar movimentos de elevação vertical e rotação, permitindo a manipulação e organização de caixas no processo de paletização. O movimento vertical foi posicionado na base do sistema, com o objetivo de aumentar a estabilidade estrutural e melhorar o desempenho durante a operação. O sistema foi desenvolvido com uma arquitetura distribuída, na qual diferentes unidades desempenham funções específicas. Uma *Raspberry Pi 4* é responsável pela interface com o usuário e pela supervisão geral do funcionamento. Um *Arduino Uno* executa os comandos de movimentação do braço, controlando diretamente os motores. Já uma *Raspberry Pi Pico* é dedicada ao monitoramento dos sensores e às funções de segurança do sistema. Essa organização permite que cada parte do sistema opere de forma independente, aumentando a confiabilidade e garantindo respostas rápidas, especialmente em situações críticas. A Figura 1 apresenta o diagrama de blocos que ilustra a estrutura geral da solução.

Figura 1: Diagrama de blocos



Fonte: Os autores.

A estrutura foi fabricada via manufatura aditiva (impressão 3D) em polímero *PLA*, utilizando preenchimento reforçado para suportar as tensões de torção. A transmissão de potência ocorre por meio de polias e correias sincronizadoras *GT2*, garantindo movimentos suaves e precisos.

Resultados alcançados

O resultado final, apresentado na Figura 2, é um protótipo funcional capaz de executar o ciclo completo de coleta e posicionamento de caixas, tanto de forma automática quanto controlada pelo operador. O sistema demonstrou consistência na execução dos movimentos, validada por meio de testes de elevação de carga e ciclos contínuos de empilhamento. O *software* de supervisão permite o acompanhamento, em tempo real, de informações relevantes do processo, como a força aplicada na preensão e o comportamento do braço durante a operação. Isso possibilita ao operador monitorar o funcionamento do sistema e garantir que ele opere dentro de condições seguras.

Figura 2: Protótipo funcional do paletizador



Fonte: Os autores.

Dentre os desafios, além da mudança da configuração inicial, que se mostrou o maior obstáculo que o grupo superou, identificou que as peças impressas em 3D não possuíam a tolerância necessária para o encaixe perfeito dos rolamentos. A solução foi a usinagem

manual de retífica, um processo de ajuste fino que garante a precisão milimétrica. No campo eletrônico, o grupo enfrentou ruídos nos sensores de fim de curso, resolvidos através da implementação de lógica de *debounce* no *firmware*, e a necessidade de lubrificação constante das hastes com graxa branca para reduzir o atrito nos motores.

O Programa *EmbarcaTech* contribuiu de forma significativa para o desenvolvimento de competências técnicas relacionadas à programação de baixo nível e à integração entre diferentes plataformas computacionais. No entanto, os principais resultados observados foram de natureza formativa, especialmente no que se refere à capacidade de lidar com falhas ao longo do processo de desenvolvimento e à atuação em equipe multidisciplinar.

O projeto possui um caminho claro de evolução, a migração para *drivers* de melhor qualidade para reduzir o ruído térmico e a inclusão de visão computacional para que o robô possa identificar e organizar caixas de diferentes tamanhos de forma dinâmica. Além disso, planeja-se otimizar a estrutura para um melhor equilíbrio do centro de massa, tornando o sistema ainda mais estável para operações de longa duração.

Projeto 3: Dispositivo para Coleta de Dados Industriais (*IIoT Edge Device*)

Henrique Oliveira dos Santos
(Residente)

Levi Silva Freitas
(Residente)

Rodrigo Almeida Pirôpo
(Residente)

A crescente adoção de tecnologias ligadas à Indústria 4.0 tem ampliado a necessidade de monitoramento e coleta de dados em ambientes industriais. Entretanto, muitas empresas ainda operam com máquinas antigas, sem conectividade digital e sem mecanismos adequados de acompanhamento em tempo real. Em diversos casos, informações importantes sobre funcionamento, falhas ou desempenho dos equipamentos dependem de registros manuais ou verificações presenciais.

Esse cenário é especialmente desafiador para pequenas e médias empresas, que frequentemente enfrentam dificuldades para modernizar seus processos devido ao alto custo de soluções industriais proprietárias e fechadas, cenário amplamente discutido no contexto da Indústria 4.0. Além disso, muitos equipamentos antigos continuam operando adequadamente do ponto de vista mecânico, mas não possuem integração com sistemas digitais.

Diante desse contexto, o projeto “Dispositivo para Coleta de Dados Industriais (*IIoT Edge Device*)” foi desenvolvido no âmbito do Programa *EmbarcaTech* em parceria com o CEPEDI. O objetivo foi construir um dispositivo capaz de coletar e transmitir dados de sensores industriais de maneira flexível, acessível e adaptável a diferentes cenários, contribuindo para aplicações de *retrofit* industrial e monitoramento remoto.

A equipe iniciou o programa com conhecimentos voltados principalmente ao desenvolvimento de *software* e sistemas embarcados em ambientes acadêmicos. Embora os integrantes já possuíssem experiência com programação, lógica computacional e microcontroladores, o contato com o desenvolvimento de um produto físico voltado a aplicações industriais representou um novo desafio.

Durante o projeto, o grupo precisou aprender e aplicar conceitos relacionados ao desenvolvimento de *hardware*, criação de placas de circuito impresso (*PCB*), integração de componentes eletrônicos, testes elétricos e organização de *firmware* multitarefa.

A divisão das responsabilidades ocorreu de forma colaborativa. Parte da equipe concentrou-se no desenvolvimento da *PCB*, interfaces e conectividade de rede, enquanto a outra atuou na estruturação do *firmware*, gerenciamento das tarefas e armazenamento dos dados. Ao longo das *sprints*, algumas decisões técnicas precisaram ser revistas após testes práticos, principalmente em relação ao *hardware*, alimentação elétrica e estabilidade do armazenamento em cartão *SD*.

Além da construção do protótipo, a experiência contribuiu para o amadurecimento técnico e organizacional da equipe, exigindo adaptação à dinâmica de desenvolvimento orientada por entregas, revisões constantes e resolução colaborativa de problemas.

O projeto resultou em um dispositivo embarcado capaz de coletar dados de diferentes sensores industriais, processar essas informações localmente e transmiti-las para sistemas externos por meio de conexão *Ethernet* ou *Wi-Fi*.

A solução foi construída utilizando o microcontrolador *RP2040*, empregado através da *Raspberry Pi Pico W*, juntamente com o sistema operacional *FreeRTOS* para gerenciamento das tarefas concorrentes do dispositivo.

O equipamento foi projetado para operar com diferentes tipos de sensores e interfaces de comunicação, incluindo *I2C*, *SPI*, *UART* e entradas analógicas. Para transmissão dos dados, foi utilizado o

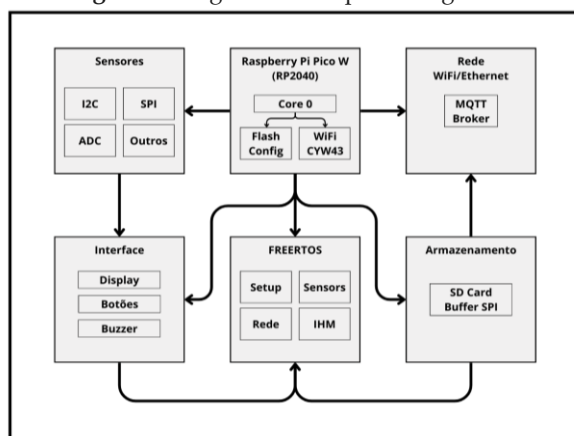
protocolo *MQTT*, amplamente empregado em aplicações *IoT* devido à sua leveza e eficiência.

Além da coleta de dados, o dispositivo também foi pensado para lidar com situações de falha de conectividade. Nesses casos, os dados permanecem armazenados localmente em cartão *SD* e são enviados posteriormente após o restabelecimento da conexão. O sistema também conta com bateria recarregável para manter o funcionamento mesmo em situações de interrupção da alimentação principal.

O dispositivo foi estruturado em módulos responsáveis por diferentes funções do sistema. De forma geral, o funcionamento ocorre da seguinte maneira:

1. Sensores industriais enviam dados para o microcontrolador;
2. O *firmware* realiza o processamento e organização das leituras;
3. As informações são transmitidas via *Ethernet* ou *Wi-Fi* utilizando *MQTT*;
4. Em caso de falha de conexão, os dados são armazenados localmente no cartão *SD*;
5. Após o retorno da rede, os registros acumulados são enviados automaticamente.

Figura 1: Diagrama da arquitetura geral do sistema



Fonte: Os autores.

Ao final do programa, a equipe conseguiu desenvolver um protótipo funcional capaz de realizar coleta, armazenamento e transmissão de dados de sensores industriais.

O dispositivo passou por diferentes versões de *hardware* até alcançar uma configuração mais estável. Durante os testes, foi possível validar o funcionamento simultâneo das interfaces de comunicação, a transmissão de dados via *Ethernet* e *Wi-Fi*, a alternância automática entre conexões de rede e o armazenamento local em situações de perda de conectividade.

Também foram realizados testes de interrupção de rede e alimentação elétrica para verificar o comportamento do sistema em cenários de falha. Nessas situações, o dispositivo conseguiu manter os dados armazenados localmente e retomar o envio após o retorno da conexão.

O desenvolvimento do protótipo envolveu ainda a criação de uma *PCB* personalizada e de um invólucro produzido por impressão 3D.

Figura 2: Protótipo físico desenvolvido



Fonte: Os autores.

Os maiores desafios do projeto estiveram relacionados principalmente à integração entre *hardware* e *software*. Durante o

desenvolvimento da *PCB*, a equipe enfrentou dificuldades ligadas ao roteamento de trilhas, estabilidade elétrica e organização física dos componentes. Algumas falhas só puderam ser identificadas após os primeiros testes práticos.

Um dos problemas mais significativos ocorreu na interface de armazenamento em cartão *SD*. Inicialmente, acreditava-se que os erros estavam relacionados ao *software* ou ao sistema de arquivos. Após diversas tentativas de depuração, verificou-se que o problema estava ligado ao aquecimento de um componente responsável pela alimentação do módulo *SD*, exigindo revisão do *hardware* e substituição de componentes.

Outro desafio importante envolveu o gerenciamento das múltiplas tarefas executadas simultaneamente no *FreeRTOS*, especialmente na sincronização entre leitura de sensores, armazenamento e conectividade.

Além das dificuldades técnicas, o projeto também exigiu adaptação ao tempo limitado das entregas e à necessidade constante de alinhar expectativas, escopo e viabilidade de implementação.

A experiência proporcionada pelo *EmbarcaTech* ampliou significativamente a formação técnica e profissional da equipe. Os integrantes tiveram contato direto com etapas que normalmente não fazem parte de projetos acadêmicos convencionais, como desenvolvimento de *PCB*, integração física de *hardware*, validação elétrica, testes de estresse e depuração de falhas em sistemas reais.

O projeto também contribuiu para o aprofundamento em sistemas embarcados, comunicação de rede, sistemas operacionais de tempo real e integração entre *hardware* e *software*. Além dos conhecimentos técnicos, houve desenvolvimento de habilidades relacionadas ao trabalho em equipe, organização de tarefas, resolução colaborativa de problemas e adaptação diante de mudanças de rota.

Outro impacto importante foi a compreensão mais ampla do processo de desenvolvimento de um produto tecnológico, desde a concepção inicial até os testes e refinamentos finais.

Como continuidade do projeto, a equipe considera importante avançar em testes do dispositivo em ambientes industriais reais, permitindo avaliar o comportamento da solução sob condições mais próximas da aplicação prática.

Também são consideradas possibilidades futuras como expansão do suporte a novos sensores e protocolos, melhorias na interface de configuração, otimização do consumo energético e refinamentos mecânicos da estrutura física do dispositivo.

De forma geral, o projeto representou não apenas a construção de um protótipo funcional, mas também um importante processo de aprendizado e consolidação de conhecimentos na área de sistemas embarcados e Internet das Coisas Industrial.

Projeto 4: Transformando Eletrodomésticos *Smart* com o *Tuya*

Matheus Santos Silva
(Residente)

Juan Pablo Azevedo Sousa
(Residente)

Arthur de Oliveira Moreira
(Residente)

Yuri Coutinho Costa
(Residente)

Leonardo Bonifácio Vieira Santos
(Residente)

A expansão da Internet das Coisas (*IoT*) exige que eletrodomésticos tradicionais se tornem inteligentes, mas o alto custo e a complexidade de desenvolver infraestruturas de nuvem e cibersegurança do zero atrasam o lançamento de produtos. Além disso, o descarte de aparelhos funcionais gera impactos econômicos e ambientais negativos. Para solucionar esse gargalo, o Projeto *Tuya* propõe um *retrofit* digital que utiliza uma plataforma de nuvem consolidada para terceirizar a conectividade. Essa estratégia reduz custos e tempo de desenvolvimento, permitindo que a engenharia foque exclusivamente na criação da placa inteligente interna.

A jornada no polo Ilhéus-Itabuna foi consolidada por uma equipe multidisciplinar de cinco residentes, composta por graduandos em Ciência da Computação, Análise e Desenvolvimento de Sistemas e um mestrando em Computação. O projeto evoluiu em imersões semanais de oito horas no laboratório, onde a sinergia do grupo permitiu dividir as frentes de trabalho entre engenharia reversa, design de *PCBs*, lógica de programação e protocolos de rede. O desenvolvimento foi estruturado em cinco etapas iterativas: estudo da plataforma *Tuya*, implementação das

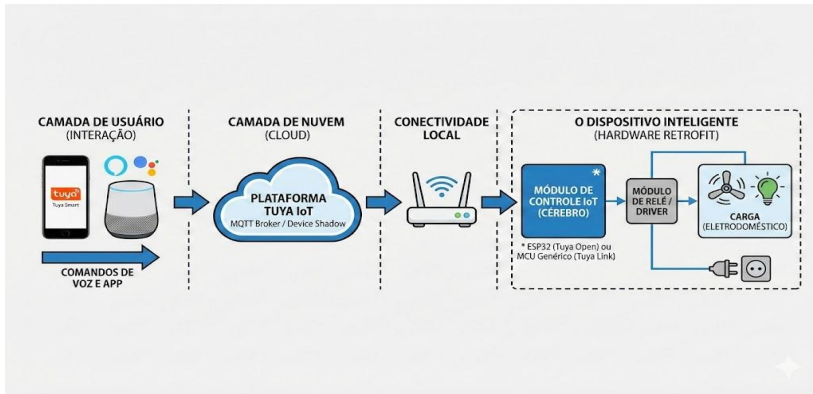
arquiteturas *Link* e *Open*, desenvolvimento do *hardware* e elaboração da documentação técnica.

Para provar o conceito, foi decidido converter um ventilador convencional em um dispositivo inteligente. A nossa abordagem adotou o microcontrolador *ESP32* como o elemento central da operação, atuando como um intermediário (*gateway*) que conecta o ventilador diretamente ao roteador *Wi-Fi* da casa e, conseqüentemente, à nuvem da *Tuya*.

A arquitetura do projeto foi concebida com foco em modularidade e segurança, sendo estruturada em três blocos principais. O primeiro corresponde ao processamento e à conectividade, realizados por um módulo *ESP32*, responsável pelo acesso à rede e pela gestão dos mecanismos de criptografia dos dados. O segundo bloco refere-se à interface de potência, composta por um conjunto de relés destinado a isolar a rede elétrica de alta tensão (127 V/220 V) da eletrônica de baixa tensão, permitindo o chaveamento seguro das velocidades do motor.

O terceiro bloco envolve a eletrônica do sistema, organizada em duas placas de circuito distintas: uma placa principal de controle e uma placa de interface, responsável pelos botões de interação com o usuário. Ambas são acomodadas em um invólucro plástico desenvolvido por meio de impressão 3D, contribuindo para a organização, proteção e integridade do conjunto.

Figura 1: Diagrama evidenciando o fluxo de comunicação entre o usuário (via aplicativo/voz), a nuvem, o microcontrolador e o acionamento físico do ventilador.

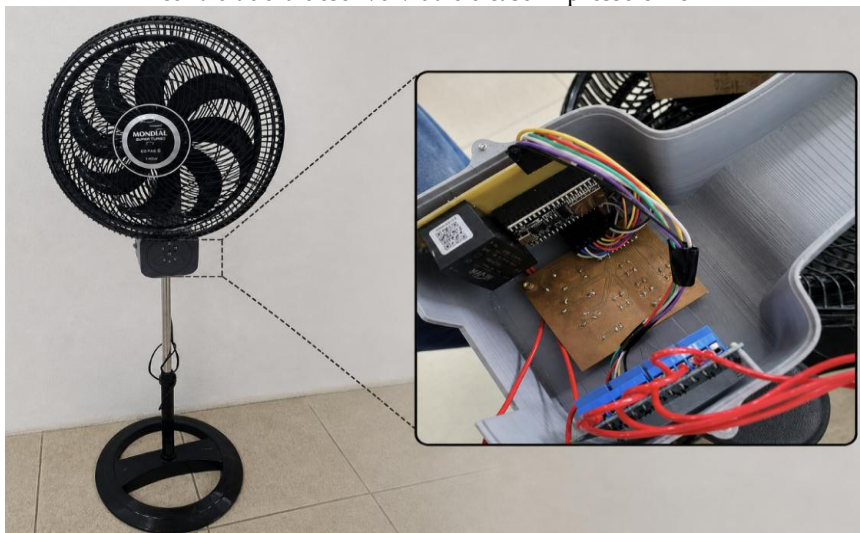


Fonte: Os autores.

O projeto culminou na entrega de dois resultados de alto impacto. O primeiro foi a construção de um protótipo físico plenamente funcional. O ventilador modernizado habilitou o controle remoto de velocidade e acionamento por meio do aplicativo de celular e provou ser totalmente interoperável com assistentes de voz do mercado, como a *Amazon Alexa* e o *Google Assistant*, sem a necessidade de criarmos servidores próprios. Além do *ESP32*, também validamos o controle usando uma placa *Raspberry Pi Pico W*, provando que o nosso método funciona em diferentes plataformas de *hardware*.

O segundo resultado foi a elaboração do "Guia de Desenvolvimento - Projeto *Tuya*". Este documento sistematiza todo o ciclo de engenharia que vivenciamos, entregando um manual passo a passo para orientar fabricantes a modernizarem seus próprios portfólios de eletrodomésticos com agilidade.

Figura 2: Protótipo físico do ventilador inteligente instrumentado com a placa controladora desenvolvida e o case impresso em 3D



Fonte: Os autores.

O principal desafio inicial concentrou-se na etapa de desenvolvimento de *software*, durante a implementação da arquitetura *Tuya Link*. Observou-se uma limitação significativa na disponibilidade de documentação técnica oficial, o que exigiu um esforço analítico aprofundado para compreender o funcionamento do protocolo. Como consequência, foi necessário estruturar manualmente os pacotes de dados em formato *JSON* e realizar a configuração dos parâmetros de comunicação diretamente no microcontrolador, garantindo a correta integração com a plataforma.

A realização da engenharia reversa da placa lógica original do ventilador apresentou elevado grau de complexidade, sendo necessária para a compreensão de seu funcionamento e integração com o novo sistema. Contudo, o principal desafio enfrentado esteve relacionado à limitação de espaço físico no interior da carcaça do equipamento.

Essa restrição impôs a necessidade de revisão do projeto elétrico inicialmente concebido, levando à adoção de uma arquitetura distribuída. Em vez de uma única placa de maiores

dimensões, a solução consistiu na divisão da eletrônica em duas placas menores e interdependentes, possibilitando sua adequada acomodação na base do ventilador.

O Programa *EmbarcaTech* desempenhou um papel fundamental na transição entre a formação acadêmica e as demandas do mercado de trabalho. Do ponto de vista técnico, a experiência possibilitou a aplicação prática dos conhecimentos teóricos, exigindo maior integração entre diferentes áreas da engenharia e maior rigor no desenvolvimento de soluções.

Nesse contexto, foram desenvolvidas competências relacionadas ao roteamento de placas utilizando o *Altium Designer*, à modelagem mecânica com o *Autodesk Fusion 360* e à implementação de *firmware* em sistemas embarcados, com o uso de *frameworks* como o *ESP-IDF* e o *TuyaOS*. Essa imersão contribuiu para o desenvolvimento de habilidades práticas essenciais à concepção de soluções tecnológicas integradas.

A convivência intensiva no laboratório exigiu comunicação constante, flexibilidade para contornar gargalos e uma visão verdadeiramente sistêmica de que um produto inteligente não é apenas código, mas a orquestração perfeita entre eletrônica, mecânica e redes. A oportunidade de absorver o conhecimento dos profissionais seniores da empresa parceira, mitigou a lacuna de experiência comum em recém-graduados, moldando-nos como profissionais mais autônomos e preparados.

Vislumbrando a continuidade desta iniciativa, mapeamos caminhos claros para a evolução do projeto. Em um curto prazo, o "Guia de Desenvolvimento" criado pode ser aplicado na conversão de eletrodomésticos de maior complexidade e potência, como refrigeradores e aparelhos de ar-condicionado. Tecnicamente, os próximos passos envolvem a adição de sensores de consumo energético à placa principal, permitindo o monitoramento de gastos em tempo real, e a migração da comunicação para o novo padrão universal de casas inteligentes, o protocolo *Matter*, mantendo a solução alinhada com as tendências definitivas do mercado global de automação residencial.

Polo Juazeiro

Giovanni Antherrely Lima da Silva
(Mentor)

Wedson Pereira da Silva
(Mentor)

Juazeiro, localizada no Norte do estado da Bahia, destaca-se como um dos principais polos educacionais, agrícolas e tecnológicos do semiárido brasileiro. Situada às margens do Rio São Francisco e integrada à região do Vale do São Francisco, a cidade possui uma forte vocação para inovação, impulsionada pela presença de instituições de Ensino Superior, centros de pesquisa e iniciativas voltadas ao desenvolvimento tecnológico.

Historicamente, Juazeiro consolidou-se como um importante centro econômico regional, especialmente nas áreas de fruticultura irrigada, comércio e serviços. Com o avanço das políticas de incentivo à ciência, tecnologia e inovação, a cidade passou a fortalecer também seu papel na formação de profissionais qualificados, ampliando sua atuação no cenário tecnológico do estado da Bahia. No polo de Juazeiro, sediado na Universidade Federal do Vale do São Francisco (UNIVASF), o programa proporcionou uma experiência imersiva de aprendizado, permitindo que os participantes desenvolvessem competências técnicas e comportamentais fundamentais para atuação no mercado de tecnologia. Inicialmente, os participantes foram organizados em equipes, sendo que cada mentor acompanhou duas equipes, e cada equipe foi subdividida em dois grupos independentes, totalizando quatro projetos por mentor.

A organização dos estudantes em equipes e grupos favoreceu uma dinâmica de aprendizado colaborativa e orientada à prática.

Cada mentor atuou diretamente no acompanhamento de duas equipes, garantindo suporte técnico, orientação metodológica e incentivo ao desenvolvimento da autonomia dos participantes.

Durante a fase de formação, os estudantes participaram de atividades estruturadas, incluindo aulas teóricas, desafios práticos e exercícios aplicados, com foco em sistemas embarcados, programação, integração de *hardware* e *software*, e desenvolvimento de soluções tecnológicas.

As mentorias ocorreram tanto de forma presencial, nas dependências da UNIVASF, quanto de forma remota, proporcionando flexibilidade e continuidade no processo de aprendizagem. O ambiente de desenvolvimento foi marcado pela colaboração entre os participantes, troca de conhecimentos e resolução conjunta de problemas.

A integração entre teoria e prática foi um dos principais diferenciais do programa. Os conteúdos apresentados nas aulas eram rapidamente aplicados nas atividades práticas e, posteriormente, nos projetos finais, o que potencializou significativamente o aprendizado dos estudantes.

Ao longo do programa, diversos desafios foram enfrentados pelos participantes, contribuindo diretamente para o desenvolvimento técnico e pessoal dos estudantes. Um dos principais desafios esteve relacionado à adaptação às tecnologias de sistemas embarcados, especialmente para aqueles que não possuíam experiência prévia na área. A necessidade de compreender conceitos de eletrônica, programação e integração de sistemas exigiu dedicação, estudo contínuo e colaboração entre os membros das equipes. Outro ponto relevante foi a organização do trabalho em equipe. A divisão em grupos menores, com projetos específicos, demandou habilidades de gestão de tempo, comunicação eficaz, divisão de tarefas e tomada de decisão. Esses aspectos foram fundamentais para garantir o andamento e a conclusão dos projetos.

Além disso, os participantes precisaram lidar com limitações técnicas, ajustes de escopo e desafios práticos durante o

desenvolvimento das soluções, o que exigiu resiliência, criatividade e capacidade de adaptação.

A evolução dos estudantes ao longo do programa foi significativa e perceptível em diferentes aspectos. Do ponto de vista técnico, os participantes passaram de um nível inicial de conhecimento para a capacidade de desenvolver soluções completas em sistemas embarcados, integrando *hardware* e *software* de forma funcional e eficiente. Além disso, houve um avanço importante no desenvolvimento de competências comportamentais. Os estudantes passaram a atuar de forma mais autônoma, assumindo responsabilidades dentro dos grupos e demonstrando maior confiança na resolução de problemas.

A experiência prática proporcionada pelo programa também contribuiu para a mudança de postura dos participantes, que passaram a se comportar como equipes de desenvolvimento, com foco em entregas, cumprimento de prazos e qualidade dos projetos.

No polo de Juazeiro, os projetos foram desenvolvidos pelos grupos formados a partir das equipes orientadas pelos mentores. Cada grupo ficou responsável por uma solução específica, permitindo maior foco e aprofundamento técnico. Os projetos envolveram o desenvolvimento de sistemas embarcados aplicados a diferentes contextos, incluindo automação, monitoramento, controle e integração com plataformas digitais. As soluções desenvolvidas buscaram atender demandas reais, aproximando os estudantes do contexto industrial e tecnológico. Cada projeto exigiu a aplicação de conhecimentos multidisciplinares, incluindo eletrônica, programação, comunicação de dados e integração com sensores e atuadores, consolidando a proposta do programa de formação prática e aplicada.

Ao longo do desenvolvimento dos projetos, a interação direta com empresas parceiras desempenhou um papel fundamental na consolidação de competências interpessoais e no aprimoramento de *soft skills* tanto dos estudantes quanto dos mentores. Diferentemente de ambientes puramente acadêmicos, a necessidade de alinhar expectativas com demandas reais do setor

produtivo exigiu dos participantes uma postura mais profissional, orientada a resultados, prazos e qualidade das entregas.

Nesse contexto, habilidades como comunicação eficaz, escuta ativa e negociação tornaram-se essenciais. Os estudantes precisaram aprender a traduzir conceitos técnicos em uma linguagem acessível para os parceiros, ao mesmo tempo em que interpretavam corretamente as necessidades e restrições apresentadas pelas empresas. Esse processo contribuiu diretamente para o desenvolvimento da capacidade de mediação entre diferentes perfis profissionais, uma competência altamente valorizada no mercado de trabalho.

Além disso, o trabalho em equipe foi intensificado pela complexidade dos projetos e pela divisão de responsabilidades entre os membros dos grupos. A convivência com diferentes níveis de conhecimento, estilos de trabalho e formas de resolução de problemas favoreceu o desenvolvimento de empatia, colaboração e gestão de conflitos. Os mentores, por sua vez, atuaram não apenas como orientadores técnicos, mas também como facilitadores dessas relações, promovendo um ambiente de respeito, organização e responsabilidade coletiva.

Outro aspecto relevante foi o fortalecimento da autonomia e do senso de responsabilidade dos participantes. A interação com empresas externas trouxe maior rigor ao acompanhamento das atividades, exigindo comprometimento com prazos, documentação adequada e validação contínua das soluções propostas. Esse cenário aproximou os estudantes das dinâmicas reais do ambiente profissional, proporcionando uma vivência prática que vai além do domínio técnico.

O polo de Juazeiro encerra sua participação no Programa de Residência Tecnológica em Sistemas Embarcados com resultados extremamente positivos. Mais do que o desenvolvimento de projetos, o programa proporcionou a formação de profissionais capacitados, preparados para atuar em ambientes reais de inovação e desenvolvimento tecnológico. Os estudantes foram desafiados a sair da zona de conforto, enfrentar problemas reais e construir

soluções com aplicabilidade prática. Ao longo dessa trajetória, desenvolveram competências técnicas e habilidades essenciais para o mercado de trabalho.

A realização do programa na UNIVASF reforça o potencial do interior da Bahia como espaço de produção de conhecimento e inovação, demonstrando que é possível formar talentos altamente qualificados fora dos grandes centros urbanos. O programa deixa como legado não apenas os projetos desenvolvidos, mas também a transformação dos participantes, que passam a compreender a tecnologia como ferramenta de impacto, inovação e desenvolvimento regional.

Projeto 1: Mecanismo XYZ para Testes Automatizados em Teclados

Roberto Vítor Lima Gomes Rodrigues
(Residente)

Gabriel Shiva
(Residente)

Andressa Sousa
(Residente)

Pedro Lucas
(Residente)

Durante os meses da etapa final da Residência em *Software Embarcado – EmbarcaTech* –, cada residente teve a oportunidade de se engajar no desenvolvimento de um projeto prático, juntamente com um grupo dentro de seu polo e guiado por um mentor. Dentro dessa perspectiva, este texto propõe a descrever o processo de desenvolvimento do projeto: o Mecanismo XYZ para Testes Automatizados em Teclados; levantando, sobretudo, os tópicos de motivação, dificuldades, trajetória e desenvolvimento, que permearam a execução desta atividade.

O problema que nos foi apresentado tangeu em especial a questão da falta de automatização nos processos de testes de teclados em esteiras fabris, que são por, muitas vezes, feitos manualmente. De acordo com levantamentos, esse tipo de trabalho, além de lento e custoso, pode acarretar problemas diversos para os operadores, tais como o desenvolvimento de Lesão por Esforço Repetitivo (LER). Dessa forma, o projeto almejou, desde seu início, a confecção de um mecanismo que pudesse reduzir a dependência do processo de testes inteiramente manuais, reduzindo assim os mencionados problemas.

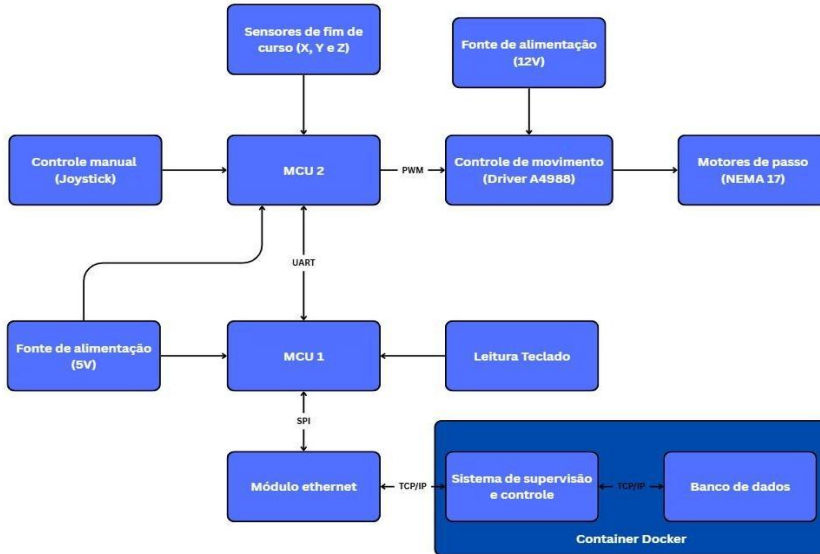
Na etapa de apresentação da problemática do tema, e da definição do escopo do projeto, o grupo trabalhou em delinear muito bem o que seria futuramente desenvolvido, tomando em conta principalmente o tempo que havia sido estipulado para tal. Esteve aí a primeira grande dificuldade da equipe, haja vista que houve um grande volume de sugestões que, em decorrência da duração estipulada do desenvolvimento, não haveria de ser colocada em pauta.

A superação desse problema deu-se através da realização de inúmeras reuniões iniciais: entre os integrantes do grupo, e com o *Product Owner*¹ do projeto. Essas foram ocasiões que permitiram uma coleta aprofundada de requisitos gerais e específicos, através da qual foram feitas concisas e definidas deliberações para o escopo do projeto. É importante observar que as reuniões eram feitas e documentadas, para que pudessem ser revistas nos momentos subsequentes.

Nesta etapa, foram definidas as tecnologias que seriam empregadas para o desenvolvimento dos *firmwares* do projeto, com vista de realizar controle de uma máquina capaz de realizar testes em teclados. Além disso, tomou-se a decisão de dividir o processamento em duas placas de microcontrolador: uma para o controle da estrutura física, e outra para o recebimento dos caracteres e validação do funcionamento do teclado. A estrutura computacional definida, incluindo a comunicação entre os *MCUs*, foi organizada para funcionar segundo o diagrama de blocos da Figura 1.

¹ No método ágil *Scrum* – que será definido mais à frente no texto –, cada produto a ser desenvolvido tem um *Product Owner (P.O.)*, responsável por descrevê-lo para que a equipe de desenvolvimento possa coletar os requisitos de desenvolvimento. Neste projeto, houve a mudança de *P.O.* na reta final do projeto.

Figura 1: Diagrama de blocos referente à organização definida para o projeto



Fonte: Os autores.

É importante observar que o desenvolvimento, segundo a residência, do projeto havia sido pensado apenas para o *software* embarcado de controle do dispositivo XYZ. No entanto, nessa etapa também decidiu-se pela confecção de um mecanismo físico de controle de eixos tridimensionais, para que houvesse o teste dos *firmwares*: é importante notar que definiu-se que a confecção da estrutura era condição indispensável no desenvolvimento, pois não seria possível observar a adequação do controle dos motores e a calibração do dispositivo sem testá-lo diretamente numa estrutura física. Dessa forma, foram levantados também requisitos

mecânicos² e elétricos³ para a construção de tal estrutura, além dos já citados requisitos computacionais⁴.

A construção do protótipo da estrutura mecânica XYZ foi realizada com materiais de sucata de impressoras e mesas. Já para resolver a implementação das questões computacionais e elétricas, houve a subdivisão da equipe do projeto em dois setores principais e independentes: um setor preponderantemente focado na parte de *hardware*, e outro setor preponderantemente focado na parte de *software*. Dessa forma obteve-se um ritmo de desenvolvimento praticamente independente e paralelo, cujo problema futuro viria a ser a junção de funcionalidades; surgiu neste ponto o segundo problema do grupo.

A esta altura, é necessário observar que todos os residentes do *EmbarcaTech* haviam recebido a orientação de utilizar o método ágil *Scrum*, sob o qual são realizadas reuniões de atualização periodicamente (*sprints*), nas quais se dividem novas tarefas e prazos para o grupo. Como apoio deste método, foi utilizado uma aplicação *web* chamada *AirTable*, para centralizar todos os *sprints* e documentação obtida. Foi devido a esta metodologia que o segundo problema – o da junção de funcionalidades – foi resolvido. Através da utilização do *AirTable*, foi possível a ampla comunicação entre os setores do grupo (de *hardware* e *software*), de forma que o desenvolvimento, ainda que independente, realizou-se de forma compatível entre os módulos do projeto.

Ao final do desenvolvimento, a equipe obteve êxito em integrar os módulos do projeto (Figura 1) e conseguiu não somente realizar o controle integrado sobre o protótipo do mecanismo XYZ,

² Nomeadamente: eixos, correias e placas de alumínio (tábuas de compensado, para o caso do protótipo).

³ Constituídos por motores de passo *NEMA 17* (mitsumi para o protótipo – oriundos de sucata), sensores de fim de curso, drivers *A4988*, um *joystick* e duas placas *placas de circuito impresso (PCIs)* para os dois microcontroladores.

⁴ Compostos por um ecossistema *NET MVC* dockerizado, com *API* e interface de controle em *JavaScript* e banco de dados *MySQL* (*front-end* e *back-end* em *.NET* no caso do protótipo), *firmwares* em C e duas placas de microcontrolador *RP2040*.

mas também evoluir o projeto para uma estrutura final, com refinamento acima do que havia sido proposto inicialmente (Figura 2). A validação de funcionamento da estrutura foi realizada em múltiplas etapas, com uma validação interna tendo sido realizada pelos integrantes e mentor e, sendo esta atestada, uma validação externa realizada pelo P.O.

Figura 2: Estrutura mecânica e placas de microcontrolador finais do projeto.



Fonte: Os autores.

Para verificar a validação interna do resultado produzido, foram realizados testes de leitura dos pacotes *USB* (*Universal Serial Bus*) recebidos pelo teclado, comparando-os visualmente às teclas pressionadas pelo mecanismo. Além disso, foram testados o controle dos motores e o armazenamento dos passos (distância) de cada eixo para a correta calibração do mecanismo. Todos os resultados foram gravados em fotos e vídeos e avaliados externamente junto à instituição solicitante, para a validação externa. É possível pensar ainda em caminhos de evolução para este projeto, como na utilização de visão computacional para realizar o mapeamento dos modelos dos teclados e na utilização de *PIO* (*Programmable I/O*) para implementação em *GPIO* (*General*

Purpose Input/Output) da pilha USB, de forma que a entrada USB/comum da *Raspberry Pi Pico* fique livre.

Ao longo do processo de desenvolvimento, o grupo obteve valiosas habilidades de trabalho em equipe, organização e delegação de tarefas, haja vista que todos estiveram envolvidos em todas as etapas da produção. Além disso, na obtenção de requisitos, a equipe desenvolveu a percepção do quanto é importante uma boa definição e projeção daquilo que se deseja desenvolver, pois está aí a base do projeto que guiará todas as etapas futuras. Também foi muito valiosa a oportunidade de lidar com diferentes pessoas – no caso, os *Product Owners* da empresa solicitante, e os integrantes das outras equipes do *Embarcatech* – e ver suas perspectivas, que muitas vezes diferem daquelas já arraigadas pela equipe, mas que devem invariavelmente ser absorvidas e levadas em conta durante o desenvolvimento.

Este projeto, em especial, oportunizou à equipe a comunicação interespecífica quanto às áreas envolvidas, levando em conta a união dos membros, do mentor e do técnico do laboratório que construiu a estrutura física. Esta união de conhecimentos de Engenharia da Computação, Elétrica e Mecânica foi uma oportunidade ímpar que deu a este grupo também novas perspectivas, habilidades e formas de trabalhar.

Projeto 2: *ApiSSense*

Gabriel Cosmo

(Residente)

Taylan Mayckon

(Residente)

Victor Gabriel

(Residente)

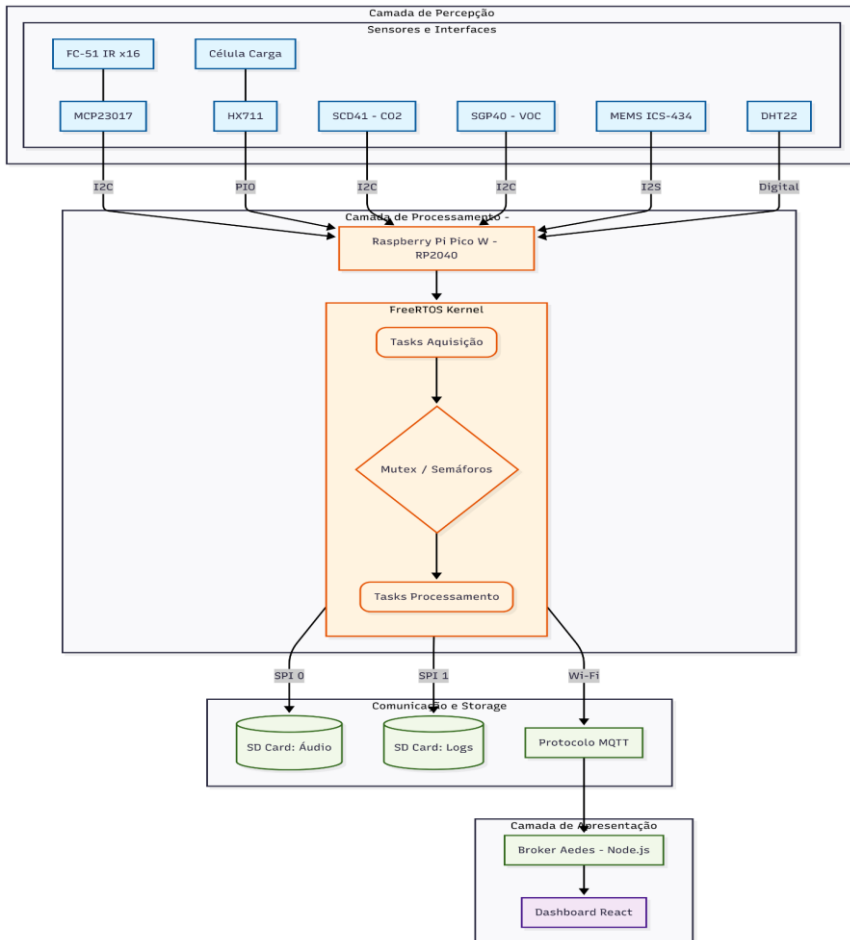
A apicultura desempenha papel fundamental na manutenção da biodiversidade e na segurança alimentar global, mas o monitoramento das colmeias ainda depende, em grande parte, de visitas manuais: uma prática trabalhosa, invasiva e sujeita a erros. As alternativas tecnológicas disponíveis comercialmente são caras e exigem infraestrutura especializada — barreiras que colocam essas soluções fora do alcance da maioria dos produtores brasileiros, num setor que conta com mais de 150 mil apicultores, a maioria vinculada à agricultura familiar. Foi a partir dessa lacuna que o *ApiSSense* tomou forma: um dispositivo de monitoramento apícola de baixo custo, simples de adotar e capaz de funcionar de forma autônoma em áreas rurais, mesmo sem acesso à internet.

A equipe chegou ao *Embarcatech* com perfis complementares e a expectativa de colocar em prática o que havia aprendido na graduação. No início, o projeto tinha um escopo mais simples, voltado apenas para a coleta de dados básicos do ambiente da colmeia. Com o passar das semanas, foi ficando claro que um monitoramento realmente útil precisava ir além: era necessário acompanhar o movimento das abelhas na entrada da colmeia, os sons produzidos pelo enxame, a qualidade do ar interno e o peso dos favos. Cada uma dessas informações, combinada com as demais, permite ao apicultor entender o que está acontecendo dentro da colmeia sem precisar abri-la. O trabalho foi organizado

de forma colaborativa, com cada membro assumindo responsabilidades específicas, e o projeto evoluiu da definição dos requisitos à montagem de um primeiro protótipo em bancada, e deste a uma versão mais robusta com componentes soldados definitivamente.

O *ApiSSense* é um dispositivo eletrônico que coleta, de forma contínua e não invasiva, informações vitais sobre o estado do enxame: Sensores infravermelhos posicionados na entrada da colmeia contam quantas abelhas entram e saem; um microfone eletrônico de alta sensibilidade captura os sons do enxame, que podem indicar eventos como enxameação ou estresse da colônia; sensores de qualidade do ar medem o nível de gás carbônico, temperatura, umidade e compostos químicos emitidos pelas abelhas; e uma célula de carga acompanha a variação de peso dos favos ao longo do tempo. Juntos, esses dados formam um retrato detalhado da saúde e da atividade da colônia. Quando há conexão disponível, as informações são enviadas para um painel de visualização acessível pelo navegador; caso contrário, ficam armazenadas localmente até que a transmissão seja possível. A Figura 1 apresenta o diagrama da arquitetura geral do sistema.

Figura 1: Diagrama de blocos da arquitetura geral do *ApiSSense*



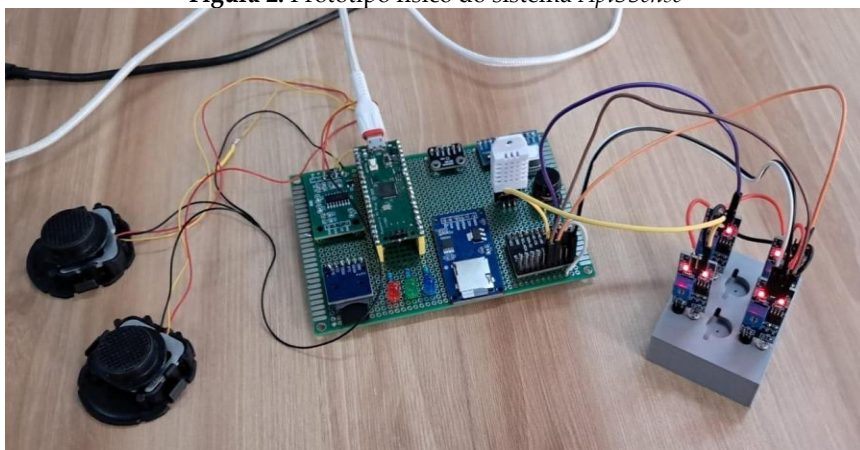
Fonte: Os autores.

Um dos maiores desafios foi fazer com que todos os sensores funcionassem ao mesmo tempo sem que um atrapalhasse o outro, o que exigiu uma reorganização profunda do código ao longo do projeto. A célula de carga apresentou um problema específico: o componente responsável por converter o peso em sinal elétrico exige uma temporização muito precisa, incompatível com a forma como o sistema gerencia múltiplas tarefas simultaneamente. A solução foi transferir esse controle para um recurso interno do

próprio microcomputador, capaz de executá-lo de forma independente. Garantir que a gravação contínua de áudio não conflitasse com a transmissão de dados pela internet também exigiu atenção: a equipe precisou separar os canais de comunicação internos do dispositivo para evitar que as operações se bloqueassem mutuamente.

Ao final, todos os módulos do sistema foram validados em laboratório. A contagem de abelhas funcionou corretamente mesmo em cenários de movimento intenso. Os sensores de qualidade do ar responderam de forma coerente a estímulos controlados. A pesagem mostrou-se adequada para acompanhar tendências ao longo do tempo. A captura de áudio foi validada com gravação contínua e armazenamento organizado em ciclos. O painel de visualização exibiu dados em tempo real durante os testes, confirmando que todo o caminho — da leitura do sensor à tela do usuário — funciona de ponta a ponta. A Figura 2 apresenta o protótipo físico desenvolvido.

Figura 2: Protótipo físico do sistema *ApiSSense*



Fonte: Os autores.

Para a equipe, a experiência foi transformadora. O projeto levou os integrantes a dominar ferramentas e conhecimentos que vão além do que a graduação costuma oferecer, e ensinou a lidar

com a frustração dos testes que falham e a importância de manter o foco diante das limitações. A Residência Tecnológica mostrou que construir tecnologia é muito mais do que chegar a um resultado final — é sobre aprender a adaptar ideias às restrições reais de tempo, recursos e do próprio mundo físico. A equipe saiu do programa tendo vivido o ciclo completo de desenvolvimento de um produto de *hardware*, da concepção ao protótipo testado.

O próximo passo é testar o dispositivo em colmeias reais, com apicultores parceiros, para calibrar os algoritmos de contagem e detecção a partir do comportamento real das abelhas. Em paralelo, a equipe pretende melhorar a fixação da célula de carga para torná-la menos sensível a vibrações, avançar na análise dos sons da colmeia para identificar eventos específicos como enxameação, e desenvolver uma versão alimentada por painel solar para operação autônoma em campo. O objetivo final permanece firme: transformar o protótipo em um produto acessível que leve inovação sustentável diretamente à realidade do apicultor brasileiro.

Projeto 3: Desenvolvendo a Inteligência do Futuro: Sistemas Embarcados para Eficiência Energética

Ariagildo Souza Lins
(Residente)

Guilherme Miller Gama Cardoso
(Residente)

Maic de Oliveira Santos
(Residente)

Richard Lima Ribeiro
(Residente)

A residência tecnológica em sistemas embarcados surgiu como uma oportunidade concreta de formação aplicada na área, conectando profissionais e estudantes a desafios reais da indústria. Para os quatro integrantes deste grupo, o programa representou um ponto de inflexão na trajetória de cada um: um espaço onde a teoria encontra o *hardware*, onde o erro faz parte do processo e onde o aprendizado acontece, muitas vezes, no momento em que algo não funciona como esperado.

O grupo foi composto por pessoas com históricos distintos, algumas com experiência prévia em eletrônica, outras com foco em *software*, todas com o mesmo ponto em comum: a vontade de compreender melhor como os sistemas embarcados funcionam e como podem ser aplicados a problemas reais. A diversidade de perfis contribuiu para a complementaridade de conhecimentos no desenvolvimento das atividades.

A residência foi estruturada em duas fases. A primeira, de caráter introdutório, serviu como base para nivelar o conhecimento do grupo e apresentar os fundamentos da programação em C voltada a sistemas embarcados. A plataforma utilizada ao longo de todo o programa foi a *BitDogLab*, fornecida pelo próprio programa,

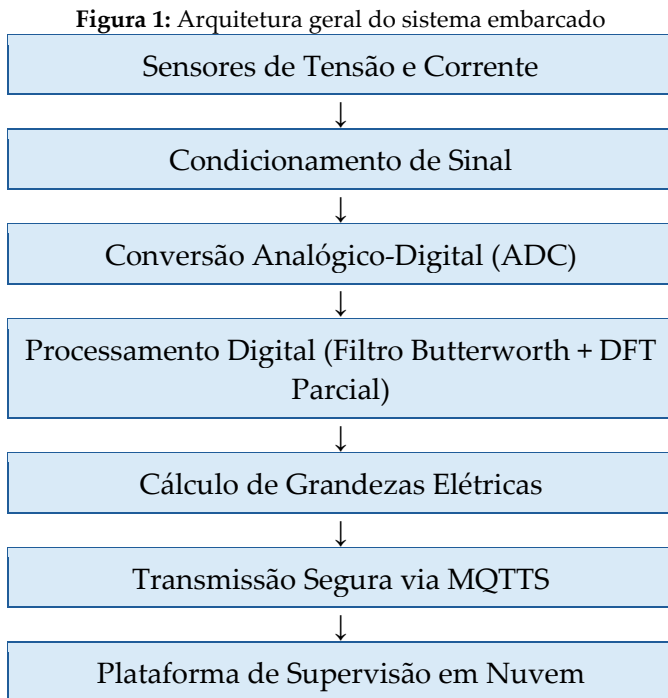
baseada na *Raspberry Pi Pico W*. Contar com um *hardware* próprio desde o início tornou o aprendizado mais acessível e prático, permitindo que cada conceito novo fosse testado de imediato na bancada.

Os encontros presenciais semanais no laboratório combinavam mentorias, aulas síncronas com conteúdos mais densos, como *multithread*, sensores, atuadores e protocolos de comunicação, e tempo livre para experimentação. Esse formato criou uma rotina de imersão que foi fundamental para o amadurecimento técnico do grupo. Não havia apenas instrução: havia espaço para tentar, errar, perguntar e debater com colegas que enfrentavam desafios semelhantes.

O ponto de virada da segunda fase foi o início do projeto aplicado, desenvolvido em parceria com uma empresa do setor de eficiência energética. A proposta era construir um medidor de energia elétrica próprio, capaz de monitorar parâmetros de redes monofásicas e trifásicas e transmitir os dados de forma segura para uma plataforma em nuvem. Nesse contexto, o grupo se organizou de forma mais estruturada, com dois integrantes assumindo maior foco nas atividades relacionadas ao *hardware*, enquanto os outros dois se dedicaram mais ao desenvolvimento de *software*, o que contribuiu para maior eficiência na execução das tarefas. Diferente dos exercícios anteriores, aqui havia um cliente real, requisitos reais e a pressão saudável de entregar algo que funcionasse além da bancada do laboratório.

O sistema resultante do projeto é um medidor embarcado capaz de aferir tensão, corrente, potência ativa, potência reativa, potência aparente e fator de potência em redes monofásicas e trifásicas. Os sinais são capturados por sensores dedicados, condicionados para compatibilidade com os conversores analógico-digitais dos microcontroladores e processados localmente por algoritmos de filtragem digital e extração da componente fundamental do sinal. Os dados são então transmitidos via protocolo *MQTT* (*Message Queuing Telemetry Transport*) com criptografia *TLS* (*Transport Layer Security*) para a

infraestrutura em nuvem da empresa parceira. A figura abaixo sintetiza essa arquitetura.



Fonte: Os autores.

Ao longo do projeto, a equipe enfrentou limitações concretas de *hardware*, como a restrição de pinos analógicos em uma das placas, que obrigou a adoção de dois microcontroladores comunicando-se via protocolo serial. Essa solução, que não estava no plano original, exigiu do grupo o desenvolvimento de um protocolo próprio de troca de mensagens, com atenção especial à sincronização e ao tratamento de erros.

A implementação da comunicação criptografada foi outro ponto de fricção relevante. Configurar certificados digitais e gerenciar uma pilha *TLS* em um microcontrolador com memória restrita não é tarefa trivial, e o processo envolveu várias rodadas de tentativas, erros e ajustes. Da mesma forma, a qualidade das

leituras dos sensores só atingiu um nível satisfatório após refinamento nos algoritmos de filtragem e nos circuitos de condicionamento de sinal, evidenciando a importância do alinhamento entre *hardware* e *software* para garantir medições confiáveis.

Além dos desafios técnicos, a própria dinâmica de trabalho em grupo em um projeto real trouxe aprendizados importantes. Tomar decisões técnicas coletivamente, documentar o que foi feito e comunicar resultados com clareza são habilidades que o laboratório ajudou a desenvolver tanto quanto a escrita de código ou a montagem de circuitos.

Ao final da residência, o protótipo desenvolvido pelo grupo atingiu estágio funcional completo para o sistema trifásico. As medições de tensão apresentaram erros inferiores a 1% em quase toda a faixa testada, confirmadas por comparação com instrumentos de referência. A comunicação segura via *MQTT* foi validada com dados transmitidos em tempo real para a plataforma da empresa parceira. Para além dos resultados quantitativos, destaca-se a evolução técnica do grupo ao longo do projeto, partindo de conceitos introdutórios até a integração de eletrônica de potência, processamento digital de sinais e mecanismos de comunicação segura em uma solução funcional.

A residência deixou marcas que vão além das competências técnicas adquiridas. Aprender a programar microcontroladores em C, entender o funcionamento interno de filtros digitais, configurar protocolos de comunicação seguros: tudo isso faz parte do legado técnico do programa. Mas igualmente importante foi o que aconteceu nos intervalos das aulas e nos momentos de dificuldade compartilhada: a construção de uma rede de pessoas com interesses e desafios comuns.

O evento *Bahia Innovate Summit 2025* (BIS), presencial em Salvador, reuniu residentes de diferentes polos e parte dos professores que sustentaram o programa, foi um momento que sintetizou bem esse espírito. Encontrar pessoalmente colegas com quem se havia trocado experiências ao longo de meses, conhecer as

soluções desenvolvidas por outros grupos e perceber a dimensão coletiva do que foi construído reforçou o sentido de pertencer a algo maior do que o próprio projeto.

O protótipo desenvolvido apresenta como etapas futuras a implementação de mecanismos de detecção de anomalias, o encapsulamento em uma placa de circuito impresso dedicada e a realização de testes em campo. Mas as perspectivas abertas pela residência vão além do projeto em si. Para cada integrante do grupo, a experiência consolidou uma base técnica e um modo de trabalhar que será levado para os próximos desafios profissionais, seja no desenvolvimento de novos sistemas embarcados, seja na aplicação dos conceitos aprendidos em outros contextos.

Projeto 4: Sistema de Coleta e Envio de Dados de Temperatura de Sistemas de Refrigeração Usando *Raspberry Pi Pico* e Protocolo RS485

João Felipe Teles Monte
(Residente)

Marco Antonio de Jesus Saturnino
(Residente)

Pedro Henrique Carvalho Felix
(Residente)

Thalles Inácio Araújo
(Residente).

Na era da informação e da Indústria 4.0, a conectividade à internet é um requisito estratégico para a eficiência dos mais variados sistemas industriais. Todavia, muitos sistemas de refrigeração ainda não dispõem de tecnologia para aquisição estruturada de dados e conectividade; isso se consolida como um gargalo para o monitoramento remoto e a análise de performance em tempo real. Este problema foi apresentado à equipe *4K Energy*, formada por quatro estudantes das áreas de Engenharia Elétrica, da Computação e Mecânica no contexto da residência em sistemas embarcados.

Assim, a equipe teve como desafio o desenvolvimento de uma solução de baixo custo e não invasiva capaz de transformar estes aparelhos ligados em dispositivos *IoT* (*internet of things*). O foco deste trabalho está na coleta de variáveis térmicas críticas, especificamente a temperatura na entrada e na saída do compressor. O dispositivo desenvolvido, operando em conjunto com ferramentas de outras equipes, compõe o arsenal necessário para que ferramentas modernas de análise, modelos de aprendizado de máquina e algoritmos de Inteligência Artificial

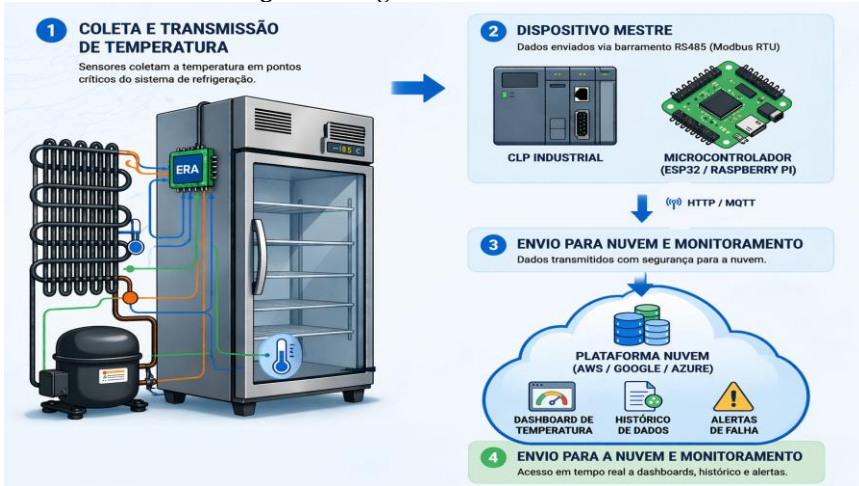
possam ser aplicados na otimização do consumo energético e manutenção preditiva.

No enfrentamento a este problema, poder contar com uma abordagem multidisciplinar tendo em vista a área de formação de cada integrante se mostrou como um desafio à parte, mas também como uma vantagem, uma vez que isso possibilitou um olhar mais holístico para o tema. Outro ponto importante foi o viés prático da residência no CEPEDI, onde os conceitos teóricos de sistemas embarcados eram acompanhados de desafios nas áreas de agricultura, indústria, sustentabilidade e outras.

Foi justamente a partir dessa vivência, da convergência de competências e da necessidade de construir uma solução de alta confiabilidade que surgiu o Projeto ERA (*Energy Refrigeration Analyzer*). O sistema foi projetado com vista a ser uma resposta robusta para a coleta, o tratamento e o envio de dados térmicos em tempo real.

O módulo *ERA* atua como um dispositivo de instrumentação de borda (*edge*), sendo instalado próximo aos sistemas que se deseja monitorar. Por meio do protocolo *Modbus RTU*, o módulo disponibiliza os dados a um dispositivo mestre, servindo como a interface primária para a digitalização de sistemas, conforme ilustrado no fluxo de dados da Figura 1.

Figura 1: Diagrama Geral do Sistema



Fonte: Os autores.

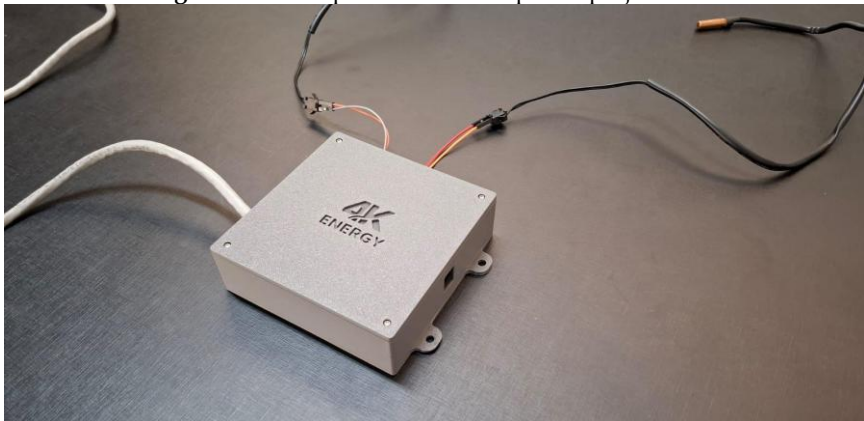
Tecnicamente, o Projeto ERA fundamenta-se na arquitetura *dual-core* do microcontrolador (MCU) RP2040. Essa estratégia permitiu o isolamento físico das tarefas críticas: o núcleo 1 dedica-se exclusivamente às rotinas de aquisição e processamento digital dos dados térmicos, enquanto o núcleo 0 gerencia a pilha do protocolo *Modbus RTU* e a comunicação via *RS485*. Essa divisão é o que garante o determinismo temporal do sistema, impedindo que o processamento de dados afete a responsividade da rede industrial ou cause latência nas respostas ao mestre. Embora o uso de sistemas operacionais de tempo real, como o *FreeRTOS*, fosse uma alternativa viável para o gerenciamento dessas tarefas, a equipe optou pela execução direta em ambos os núcleos para maximizar a eficiência e reduzir o consumo de memória, garantindo um código mais enxuto e otimizado para a *hardware*.

A aquisição de temperatura é executada via sensores *NTC* (*Negative Temperature Coefficient*) de 10 k Ω , posicionados em pontos estratégicos do sistema de refrigeração: próximos à entrada e saída do compressor. Para mitigar ruídos eletromagnéticos inerentes ao ambiente industrial e aumentar a precisão das leituras analógicas,

o *firmware* implementa a técnica de *oversampling*, realizando a captura de 200 amostras consecutivas por ciclo de leitura, seguida de filtragem por média aritmética. O objetivo com esta medida foi melhorar a resposta do sistema em ambientes críticos e foi possibilitada pela arquitetura *dual-core* comentada anteriormente.

Para consolidar a solução em um produto apto para o uso industrial, a equipe desenvolveu uma Placa de Circuito Impresso (PCI) dedicada utilizando uma máquina CNC e uma *case* personalizada em polímero **PLA**, modelada no Fusion 360 e fabricada por impressão 3D. Esta etapa teve como objetivo assegurar ao módulo maior robustez elétrica e mecânica. O protótipo final está apresentado na Figura 2.

Figura 2: Protótipo desenvolvido para o projeto ERA



Fonte: Os autores.

Cabe citar que o desenvolvimento do protótipo envolveu ajustes de escopo e a adequação à infraestrutura laboratorial disponível. A validação da comunicação industrial, por exemplo, esbarrou na ausência de um Controlador Lógico Programável (CLP) no polo de capacitação. Para suprir essa lacuna, implementou-se um dispositivo mestre customizado para verificar a integridade da rede *Modbus* do módulo *ERA*.

Sobre o dispositivo mestre desenvolvido, é importante dizer que o *hardware* foi concebido a partir da placa *Bitdoglab* e o módulo *MAX485*, um conversor de lógica *TTL* (*Transistor-Transistor Logic*) para *RS485*.

Quanto ao *firmware* do dispositivo mestre, a equipe optou pela linguagem de programação *MicroPython* devido à necessidade do projeto de garantir uma comunicação segura via *MQTT* (*Message Queuing Telemetry Transport*) para o monitoramento remoto e a indisponibilidade de bibliotecas maduras em *C/C++* para essa finalidade específica na plataforma. Essa decisão permitiu a rápida implementação de um “*gateway*” de testes, possibilitando a validação da comunicação *Modbus RTU* do módulo *ERA* e a rastreabilidade dos dados antes mesmo da integração final com sistemas comerciais.

Por fim, o projeto desenvolvido junto à empresa parceira nos levou a adotar rigorosos padrões de documentação via *Doxygen* para assegurar a continuidade do trabalho. Como perspectivas futuras, visamos integrar o *ERA* a sistemas de medição de potência elétrica, para, por meio do cruzamento de dados, obter de forma mais precisa o perfil e o comportamento dos sistemas de refrigeração. Estabelece-se também, como objetivos futuros, realizar testes em ambientes operacionais complexos, garantindo que o projeto evolua de um protótipo acadêmico para uma solução comercial de alto impacto na eficiência energética de aparelhos de refrigeração.

Todos estes aspectos do projeto e das próprias atividades semanais, onde fomos instigados a superar limitações técnicas e propor soluções para problemas complexos, tiveram como ponto positivo a multidisciplinaridade da equipe. A integração técnica possibilitou um olhar mais amplo dos problemas, o que de certo modo enriqueceu as reuniões semanais e as entregas.

Assim, o programa *EmbarcaTech* consolidou-se como uma oportunidade para aplicação da engenharia como ferramenta para solução de problemas técnicos e otimização de processos, ampliando a visão de mundo e a postura dos residentes frente aos problemas contemporâneos.

Polo Vitória da Conquista

Auerê Vasconcelos Veras

(Mentor)

Cléia Santos Libarino

(Mentora)

O programa *EmbarcaTech* desenvolveu parte de suas atividades no Polo de Vitória da Conquista, com o objetivo de capacitar profissionais e estudantes de nível superior e técnico em Tecnologias da Informação e Comunicação (TIC) e áreas correlatas, com foco específico em Sistemas Embarcados.

O programa foi estruturado em duas etapas. A primeira, de capacitação inicial, ocorreu de forma online e sem encontros presenciais nos polos. A segunda etapa consistiu em uma Residência Tecnológica voltada aos alunos aprovados na fase anterior. Esta fase foi realizada em parceria com o Instituto Federal da Bahia (IFBA) — *campus* Vitória da Conquista, instituição estratégica que atende cerca de 2 mil discentes e oferece cursos que vão desde a qualificação básica até o nível superior. O *campus* Vitória da Conquista é uma instituição estratégica no estado, oferecendo para a sociedade local, do Sudoeste da Bahia e do Norte de Minas Gerais, cursos de Educação Tecnológica Profissional em diversos níveis do sistema educacional. Esses cursos são estabelecidos de acordo com a legislação pertinente: básico (cursos de qualificação, requalificação e re-profissionalização de jovens, adultos e trabalhadores em geral, com qualquer nível de escolarização), técnico (habilitação profissional de nível médio) e superior (graduação).

Durante a residência, os participantes aplicaram os conhecimentos adquiridos em estudos de caso e projetos práticos,

enfrentando desafios em setores como educação, segurança, indústria e saúde, especialmente voltados à Internet das Coisas (*IoT*). No Polo de Vitória da Conquista, a equipe foi composta por 20 residentes: 16 na trilha de Sistemas Embarcados e 4 na trilha de *FPGA*. O grupo destacou-se pela heterogeneidade, reunindo diferentes níveis de formação e áreas de atuação, o que favoreceu a troca de conhecimentos.

Os residentes foram organizados em quatro equipes para trilha de Sistemas Embarcados e uma equipe para *FPGA*, porém o acompanhamento direto com os mentores só ocorreu com os residentes de Sistemas Embarcados, e a equipe de *FPGA* tendo suporte sob demanda. As equipes foram divididas de modo a mesclar da melhor forma os residentes, de acordo com suas formação e experiências.

Como a dinâmica da residência era em formato híbrido, os residentes tinham aulas remotas semanais com os instrutores, sendo as mentorias remotas e presenciais no polo. As mentorias presenciais ocorreram no Laboratório de Sistemas Mecatrônicos, pertencente ao curso de Engenharia Elétrica, o qual dispõe de infraestrutura para realização de atividades propostas nas aulas. Os residentes, apesar de estarem divididos em equipes, nos encontros presenciais contavam com o apoio dos dois mentores, bem como das outras equipes, aproveitando assim a experiência e expertise de cada um. Esse formato, no Polo de Vitória da Conquista, surtiu efeito positivo, então mantemos durante toda a residência.

Nesse início da segunda fase, os residentes e mentores acompanharam as aulas dos instrutores. A mentoria semanal era voltada para rever o que foi visto na aula, tirando possíveis dúvidas, bem como o acompanhamento das tarefas solicitadas pelos instrutores. Em alguns momentos, foram necessárias aulas complementares, para nivelamento dos residentes em determinados conteúdos. Quando necessário, além da mentoria remota semanal, nos reunimos no Polo para realização das atividades da semana. Cada atividade deveria ser entregue pelo *GitHub*, onde deveria constar toda documentação, desde o relatório

referente à atividade, bem como código fonte, e vídeo de demonstração de funcionamento. Esta plataforma de colaboração (*GitHub*), foi utilizada em todas as fases da residência.

Os encontros presenciais tornaram-se mais frequentes após a entrega dos kits de sensores e atuadores. Devido à insuficiência de componentes disponíveis para cada residente, foi necessário o uso do laboratório para o compartilhamento desses recursos. Essa prática se revelou altamente benéfica, pois estimulou a troca de conhecimento e a consolidação das informações entre as equipes, promovendo um ambiente colaborativo de aprendizado.

No decorrer da segunda fase, chegou o momento em que cada equipe iria focar em um problema concreto, disponibilizado por uma empresa parceira do CEPEDI. Deste modo, cada equipe ficou com um projeto específico, e teve um representante (*PO*) da empresa alocado ao projeto. Foram realizadas reuniões entre os residentes, mentores e *PO* para que cada equipe conseguisse compreender o problema, fazer o levantamento de requisitos, e identificação de componentes a serem utilizados no desenvolvimento do projeto.

As quatro equipes do Polo de Vitória da Conquista, desenvolveram os seguintes projetos:

1. Desenvolvimento de um veículo guiado automaticamente (*AGV*) para navegação autônoma em ambientes industriais;
2. Desenvolvimento de um protótipo com controle XYZ para um armazém automatizado que utiliza uma máquina *CNC 3018* e o microcontrolador *Raspberry Pi Pico W* para movimentar, identificar e gerenciar *pallets* em seis posições de armazenamento;
3. Carga e teste do módulo do *pack* com 2 baterias de Li-Ion;
4. Bateria de *backup* para *CODAX-M*.

Assim, como cada equipe estava com um problema distinto, o acompanhamento foi mais individualizado, continuando as mentorias semanais, porém com cada equipe em separado, para focar nos problemas específicos de cada projeto. Nesse contexto, além das mentorias semanais, incluímos pequenas reuniões diárias (*Daily*) para o melhor acompanhamento.

Nesse contexto, foi incluído o *Airtable*, uma plataforma em nuvem que facilita o acompanhamento do projeto, organização de informações, fluxo de trabalho, divisão de tarefas entre os residentes, e a documentação do projeto. Essa plataforma foi alimentada tanto pelos mentores, como pelos residentes, de forma contínua.

Durante a fase de desenvolvimento de projetos, os residentes passaram por algumas dificuldades, que vão desde conhecimento teóricos/práticos, como a compreensão dos requisitos do projeto, além da construção da parte física (Protótipo). Porém essas dificuldades foram sanadas no decorrer do desenvolvimento do projeto. Antes da conclusão dos projetos, tiveram a oportunidade de apresentar no *BIS (Bahia Innovate Summit)* organizado pelo CEPEDI, onde puderam apresentar o andamento dos trabalhos, trocar experiências e, principalmente, captar sugestões para o projeto.

Projeto 1: Controle XYZ: Automação de *Warehouses*

Carlos Henrique

(Residente)

Guilherme Aguiar Silva

(Residente)

Maria Luiza Brasil Medrado

(Residente)

Muriel da Costa Santos

(Residente)

A gestão de estoques é um pilar crítico para a eficiência de qualquer indústria. O projeto surgiu da parceria com o Centro de Pesquisa, Desenvolvimento Tecnológico e Inovação (CEPEDI), que identificou desafios reais na gestão manual de armazéns. Processos manuais frequentemente resultam em erros de contagem, baixa produtividade devido ao tempo gasto na localização de itens e falta de visibilidade dos dados em tempo real. O objetivo central foi construir uma solução que automatizasse a movimentação e o registro de produtos, reduzindo falhas humanas e otimizando o fluxo operacional.

A equipe foi formada por integrantes com diferentes níveis de experiência: desde estudantes de Engenharia Elétrica e Sistemas de Informação até profissionais já atuantes no mercado. Essa diversidade trouxe o desafio de nivelar conhecimentos técnicos em eletrônica e programação C.

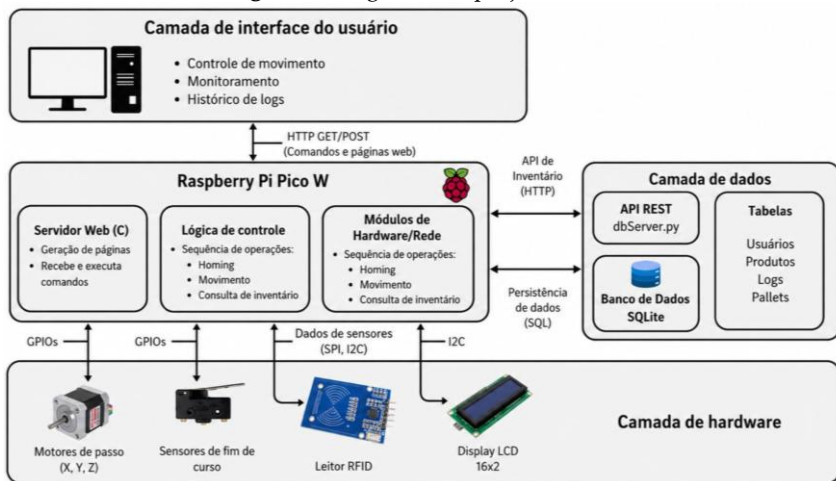
A evolução do grupo foi marcada pela transição do aprendizado teórico para a aplicação prática. A organização seguiu metodologias ágeis, com ciclos de desenvolvimento (*sprints*) de duas semanas e reuniões diárias para alinhamento. Um ponto marcante da trajetória foi o amadurecimento na gestão de equipe e na comunicação, permitindo que o grupo superasse as dificuldades

iniciais de integração entre o desenvolvimento do *software* e a montagem física do protótipo.

O Controle XYZ é um sistema de armazém automatizado que utiliza uma estrutura de máquina *CNC 3018* modificada como base para operação em escala reduzida. A arquitetura do projeto é composta por um controle central baseado no microcontrolador *Raspberry Pi Pico W*, executando o sistema operacional de tempo real *FreeRTOS*. O mecanismo de movimentação é realizado por motores de passo *NEMA 17*, responsáveis por garantir a precisão nos três eixos de deslocamento.

Para identificação e manipulação dos itens, o sistema emprega um leitor *RFID*, que reconhece os paletes por meio de etiquetas eletrônicas, e um eletroímã, utilizado na coleta e no posicionamento dos objetos. A interação com o operador ocorre por meio de uma interface *web*, enquanto um banco de dados é responsável pelo registro e monitoramento das operações realizadas.

Figura 1: Diagrama do projeto Controle XYZ



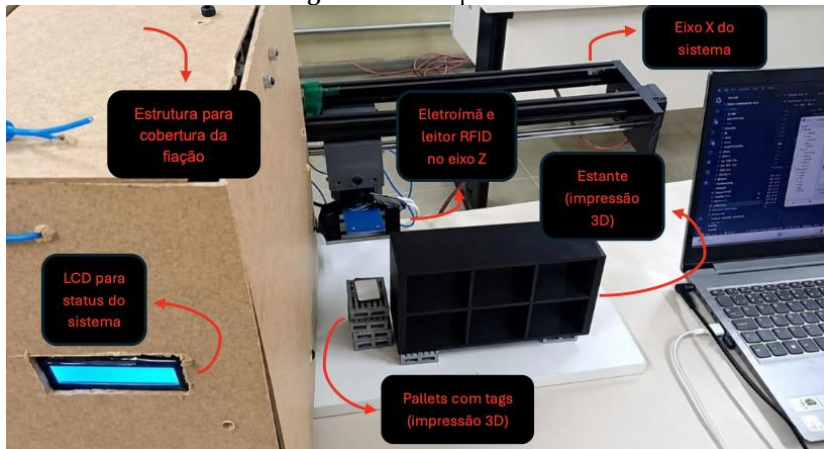
Fonte: Os autores.

O projeto resultou em um protótipo funcional capaz de realizar ciclos completos de armazenamento e retirada de paletes em seis posições distintas. Os testes realizados confirmaram a

precisão no posicionamento dos eixos e a confiabilidade na leitura das etiquetas de identificação, evidenciando a consistência do sistema de controle e sensoriamento.

A integração com a interface *web* possibilitou o controle remoto do sistema sem atrasos perceptíveis, garantindo o registro adequado das operações em banco de dados. Além disso, foram implementados mecanismos de segurança baseados em sensores de fim de curso, que impedem a execução de movimentos fora dos limites físicos e evitam operações em posições vazias, contribuindo para a confiabilidade e integridade do sistema.

Figura 2: Protótipo Final.



Fonte: Os autores.

O percurso não foi linear e apresentou obstáculos significativos:

- **Integração Físico-Lógica:** Muitas falhas de código só apareciam no protótipo real, não sendo detectadas em simulações. Isso exigiu que a equipe antecipasse os testes físicos no cronograma.

- **Ajustes de Rota:** Foi necessário simplificar algumas funcionalidades inicialmente planejadas para garantir que o núcleo principal de controle dos motores funcionasse com total estabilidade.

A experiência no *EmbarcaTech* foi transformadora para todos os membros. Tecnicamente, o grupo aprendeu a programação de sistemas de tempo real e a integração complexa de periféricos de *hardware*. No aspecto humano, o trabalho colaborativo sob pressão e a interação direta com uma empresa parceira (CEPEDI) trouxeram uma visão de mercado essencial para a formação profissional. O maior aprendizado foi compreender que, em sistemas embarcados, o *hardware* e o *software* devem caminhar juntos desde o primeiro dia de projeto.

Para a evolução do Controle XYZ, prevê-se o desenvolvimento de uma placa de circuito impresso (PCB) dedicada, com o objetivo de centralizar as conexões e aumentar a robustez elétrica e mecânica do sistema. Além disso, pretende-se expandir a interface *web* para permitir o cadastro detalhado de produtos e a gestão de informações associadas. Essas melhorias visam aproximar o sistema de uma solução com características industriais, ampliando sua escalabilidade, organização operacional e capacidade de integração em ambientes mais complexos.

Projeto 2: Fonte Inteligente de *Backup* para Sistema de Monitoramento

Luiz Filipe Ribeiro
(Residente)

Heitor Rodrigues Lemos Dias
(Residente)

Paulo César de Jesus Di Lauro
(Residente)

Roberto Souza Cardoso
(Residente)

Sistemas de monitoramento exigem operação contínua, mas sua confiabilidade é frequentemente comprometida pela dependência exclusiva da rede elétrica convencional. Durante interrupções de energia, a inoperância imediata dos equipamentos causa perda de dados críticos. A motivação deste projeto foi transpor essa vulnerabilidade. O objetivo central foi desenvolver uma "Fonte Inteligente" que garanta a transição ininterrupta de energia e, simultaneamente, atue como um nó *IoT*, fornecendo diagnósticos vitais e telemetria remota sobre o status da rede e da reserva da bateria.

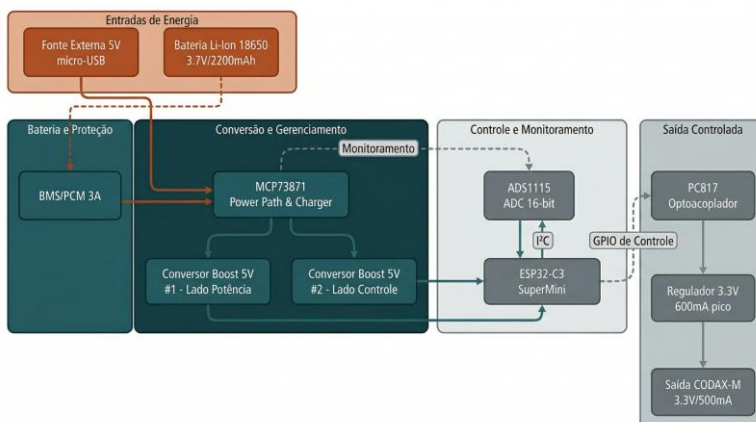
A equipe adotou o *framework* ágil *SCRUM*, estruturando o desenvolvimento em sete *Sprints* iterativas. O trabalho foi organizado em duas frentes simultâneas e integradas: *hardware* e *software* embarcado. No *hardware*, a evolução partiu de pesquisas de componentes e simulações virtuais, avançando para testes práticos em *protoboard* até o desenho e refino da Placa de Circuito Impresso (*PCB*). Paralelamente, o *firmware* do *ESP32* foi arquitetado utilizando Programação Orientada a Objetos em C++, garantindo um código modular para a leitura de sinais, armazenamento de *logs* em memória *flash* e conectividade *Wi-Fi*. Durante o processo, a comunicação frequente entre as frentes foi estritamente necessária

para resolver conflitos práticos e ajustar o código às restrições elétricas do *hardware*, o que viabilizou a integração do sistema de *backup* com alinhamento entre as duas frentes.

A solução desenvolvida consiste em uma fonte de alimentação com bateria integrada, capaz de manter o funcionamento do sistema durante interrupções de energia. O dispositivo opera de forma automática, utilizando a rede elétrica quando disponível e acionando a bateria em caso de falha. Além da função de *backup*, o sistema incorpora monitoramento do estado de funcionamento, permitindo identificar quando está operando em modo normal ou em modo de emergência. Também foram incluídos recursos para registro de eventos e acompanhamento do comportamento ao longo do tempo.

De forma geral, a solução combina alimentação ininterrupta com funcionalidades de observação do sistema, ampliando sua utilidade para além do fornecimento de energia. A Figura 1 apresenta o fluxo de energia e informação no sistema, evidenciando a separação entre o caminho de potência e o domínio de controle lógico.

Figura 1: Diagrama de blocos da arquitetura de hardware do sistema desenvolvido



Fonte: Os autores.

O projeto resultou na entrega de um protótipo físico montado e funcional (Figura 2), acondicionado em um *case* impresso em 3D desenhado pela equipe. O sistema atua como um *nobreak* inteligente e gerencia a transição ininterrupta de energia para o dispositivo externo. Em validações de laboratório com o uso de osciloscópio, a tensão de saída demonstrou grande estabilidade na faixa dos 3,3 V, o que garante a operação segura do dispositivo externo. Com a implementação de estratégias de eficiência energética no *firmware*, a autonomia da bateria em testes práticos alcançou aproximadamente 1 hora de operação contínua, superando com folga a meta inicial de 30 minutos.

Figura 2: Protótipo físico da Fonte Inteligente de *Backup*



Fonte: Os autores.

Além da confiabilidade elétrica, a camada de telemetria (*IoT*) foi validada na prática: o equipamento identifica de forma autônoma as quedas e retornos de energia, armazena esses *logs* internamente e os sincroniza com um servidor *web* externo. Isso resultou em um *dashboard* funcional que comprova a viabilidade técnica do monitoramento remoto do dispositivo com alinhamento ao ciclo de desenvolvimento proposto.

O projeto enfrentou desafios críticos de *hardware*, como a incompatibilidade do módulo *buck-boost* com os picos de corrente e quedas de tensão na *PCB*, resolvidas com o redesenho da

alimentação e o estanhamento manual das trilhas. No *software*, a instabilidade do sistema *SPIFFS* exigiu a migração para *LittleFS* e o uso de *POO* para organizar a complexidade.

As principais limitações foram o cronograma apertado, devido à transição tardia para este projeto, e o atraso na chegada de componentes, que restringiu o tempo de testes físicos. A mudança de rota foi técnica e logística: a equipe abandonou a perfeição teórica inicial para focar em um *MVP* funcional, utilizando o *framework Scrum* para garantir a entrega e a integração do sistema dentro do prazo disponível.

A experiência no projeto permitiu ao grupo consolidar o domínio em *ESP32*, aprendendo a integrar *hardware* e *software* sob rigor industrial. O principal ganho técnico foi converter teoria em uma solução real de telemetria e *backup* de energia, superando em 100% a meta de autonomia e garantindo a integridade dos dados com o uso de *LittleFS*.

Organizacionalmente, a vivência com o *Scrum* e a empresa parceira amadureceram a gestão de prazos e a priorização de requisitos. A experiência foi o diferencial na formação dos integrantes ao exigir a entrega de um *MVP* funcional em curto prazo, desenvolvendo a capacidade de resolver problemas críticos de engenharia sob pressão e com visão de mercado.

Os próximos avanços visariam otimizar o *MVP* e escalar a arquitetura. No *hardware*, projeta-se a fabricação industrial da *PCB* com trilhas redimensionadas, eliminando adaptações manuais, e a otimização termodinâmica do *case* impresso em 3D. No domínio de *software*, planeja-se integrar algoritmos de Inteligência Artificial para realizar a análise preditiva da vida útil da bateria, antecipando sua degradação estrutural. Além disso, estuda-se a adoção de comunicação redundante (via *GSM/LTE* ou *LoRa*) para assegurar o envio dos *logs* de telemetria mesmo em cenários onde a infraestrutura *Wi-Fi* local se torne inoperante.

Projeto 3: Inovação em Sistemas Embarcados: Módulo Inteligente para Diagnóstico e Carga de Baterias

Aulo Cezar Ferreira dos Anjos Filho
(Residente)

Leonardo Romão Queiroz Araújo
(Residente)

Marcus Vinicius Magalhães França Filho
(Residente)

Matheus Nepomuceno Souza
(Residente)

O uso de dispositivos móveis no comércio e na logística depende da confiabilidade e da autonomia energética, especialmente em equipamentos que utilizam baterias de íon-lítio com sistemas internos de monitoramento. Esses sistemas acompanham informações como carga e temperatura, sendo essenciais para o funcionamento seguro. No entanto, na manutenção, muitos técnicos ainda utilizam carregadores convencionais que não acessam esses dados, dificultando o diagnóstico e podendo levar ao descarte prematuro de baterias.

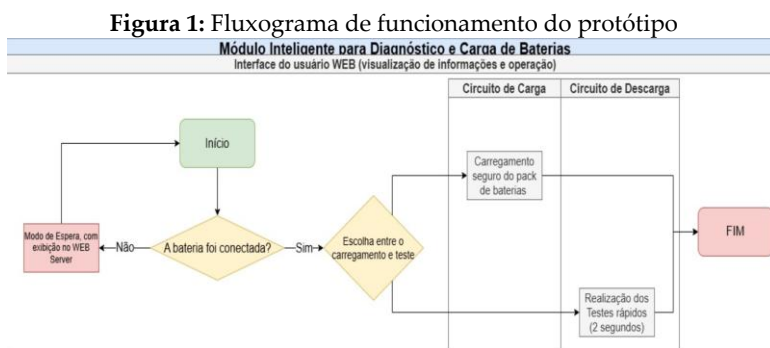
Diante desse cenário, o projeto buscou desenvolver um módulo capaz de realizar o carregamento e o diagnóstico dessas baterias em uma única solução. A proposta visa tornar o processo mais eficiente e confiável, contribuindo para diagnósticos mais precisos e para a redução de desperdícios, além de incentivar o uso mais sustentável de recursos eletrônicos.

O grupo de Vitória da Conquista uniu competências de Engenharia Elétrica, Sistemas de Informação e Ciência da Computação para abordar o problema de forma multidisciplinar. Inicialmente focado apenas no carregamento de baterias, o projeto evoluiu ao longo do processo para incluir funcionalidades de

diagnóstico, ampliando seu escopo. O desenvolvimento ocorreu interativamente, com testes frequentes, divisão de tarefas conforme as habilidades da equipe e ajustes contínuos para garantir o avanço do projeto dentro do prazo.

A solução desenvolvida consiste em um módulo embarcado capaz de realizar o carregamento e o diagnóstico de baterias de forma integrada. O sistema foi projetado para operar de maneira confiável, organizando as etapas de teste e garantindo a segurança durante o uso.

De forma geral, o sistema é composto por um microcontrolador responsável pelo controle das operações. Este componente central orquestra todo o processo, desde a comunicação com a bateria até a coleta e o envio de dados. Integrado ao sistema, há um módulo de carregamento dedicado a realizar a recarga da bateria com perfis otimizados, e um sistema de descarga, crucial para a execução dos testes de capacidade e saúde das baterias. A arquitetura se completa com um conjunto de *software* que coordena o funcionamento dos módulos de *hardware*, realiza a coleta de dados de tensão e temperatura, e permite o acompanhamento do diagnóstico por meio de uma interface simples e intuitiva, garantindo a rastreabilidade das informações. O fluxograma completo de funcionamento do protótipo pode ser visualizado na Figura 1.



Fonte: Os autores.

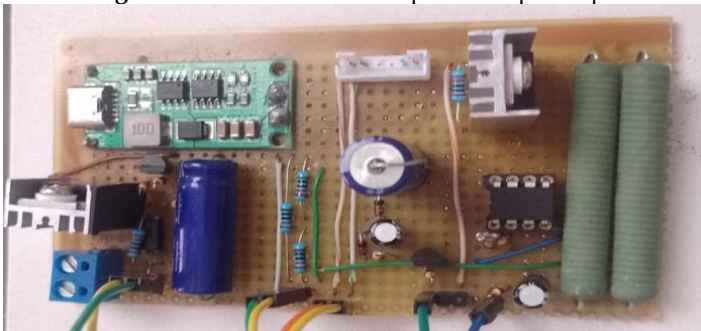
Durante o desenvolvimento, a equipe enfrentou desafios técnicos e de integração que exigiram análise e adaptação. Um dos principais problemas ocorreu no circuito de descarga, onde a tensão disponível não era suficiente para o funcionamento adequado de um dos componentes, sendo necessário ajustar o projeto para garantir o desempenho do sistema.

No *software*, surgiram dificuldades relacionadas ao envio simultâneo de dados, gerando inconsistências no armazenamento. Esse problema foi resolvido com mecanismos de controle de acesso. Além disso, interferências nos contatos elétricos afetaram a confiabilidade das leituras, exigindo ajustes para estabilização dos sinais. Esses desafios contribuíram para o amadurecimento do projeto ao longo do processo.

Ao final do desenvolvimento, foi possível construir um protótipo funcional para carregamento e teste de baterias. O sistema apresentou funcionamento estável durante os ciclos de operação, sendo capaz de coletar e armazenar dados relevantes, além de operar de forma autônoma mesmo em situações de falha de conexão.

Os testes indicaram que o módulo realiza diagnósticos de forma mais ágil em comparação a métodos tradicionais, fornecendo informações úteis sobre o estado das baterias. Assim, o projeto atingiu um estágio de protótipo validado em ambiente de testes, evidenciando a viabilidade da solução e seu potencial de aplicação prática. A Figura 2 mostra a placa de circuito impressa do protótipo.

Figura 2: Placa de Circuito Impressa do protótipo



Fonte: Os autores.

A participação no desenvolvimento do projeto proporcionou aprendizados relevantes para todos os integrantes, tanto no aspecto técnico quanto no trabalho em equipe. Ao longo do processo, houve aprofundamento em áreas como eletrônica aplicada, sistemas embarcados e organização de *software*, sempre a partir de situações práticas que exigiram análise, adaptação e tomada de decisão.

Além disso, a experiência contribuiu para o desenvolvimento de habilidades importantes, como colaboração, divisão de responsabilidades e comunicação entre diferentes áreas de conhecimento. O contato com um problema real também ampliou a compreensão sobre o papel da tecnologia na redução de desperdícios e no uso mais eficiente de recursos, reforçando o impacto positivo que soluções bem planejadas podem gerar.

Como continuidade do projeto, a equipe identificou oportunidades de melhoria tanto no *hardware* quanto no *software*, incluindo ajustes no circuito para reduzir interferências durante os testes e a adaptação do sistema para suportar diferentes tipos de baterias, ampliando sua aplicabilidade. Além disso, há perspectivas de evolução na interface de uso, tornando-a mais intuitiva, e na integração com sistemas externos para análise mais avançada dos dados coletados, indicando caminhos para tornar a solução mais robusta e preparada para aplicações em contextos mais amplos.

Projeto 4: Sistema de Controle AGV com *Dashboard* em Tempo Real

Jeová Cosme de Jesus Pinheiro
(Residente)

Jonas Souza Pinto
(Residente)

Leonardo Rodrigues Luz
(Residente)

Wilton Lacerda Silva Jr.
(Residente)

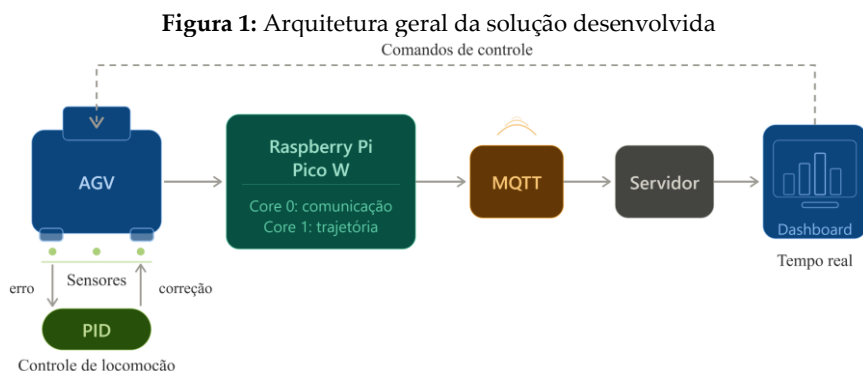
O projeto foi desenvolvido no contexto da residência *EmbarcaTech*, a partir da necessidade de automatizar o transporte interno de itens em ambientes logísticos. A proposta buscou responder a um problema prático: como criar uma solução acessível, funcional e capaz de demonstrar, em pequena escala, recursos importantes para a automação, como navegação guiada, identificação de cargas e monitoramento em tempo real. A escolha de um *AGV* (*Automated Guided Vehicles*) com *dashboard* permitiu trabalhar uma aplicação próxima de situações reais, unindo sistemas embarcados, sensoriamento e comunicação de dados em uma solução objetiva, viável para o tempo disponível e relevante para a formação da equipe.

Ao longo do programa, a equipe manteve reuniões noturnas praticamente todos os dias úteis, em formato de *dailys*, para atualizar o que havia sido feito, alinhar os próximos passos e, quando necessário, redistribuir tarefas entre os integrantes. As decisões e encaminhamentos dessas reuniões eram registrados em Atas, o que ajudou a acompanhar a evolução do trabalho e a reduzir retrabalho. Além disso, ocorreram reuniões constantes com o *Product Owner* (*PO*), importantes para revisar requisitos, discutir

dificuldades e validar ajustes no caminho adotado pela equipe. Com esse acompanhamento frequente, o projeto avançou de forma mais organizada, integrando as frentes de *software*, eletrônica, modelagem e testes.

A solução desenvolvida consiste em um AGV seguidor de linha com monitoramento em tempo real. No protótipo, o veículo lê a pista por sensores, ajusta a locomoção para manter a trajetória e envia informações para acompanhamento externo. O *Raspberry Pi Pico W* atua como núcleo do sistema, concentrando o processamento embarcado e a comunicação da solução. Além do deslocamento do AGV, a proposta integra leitura, resposta e supervisão em um mesmo fluxo de operação. Isso permite apresentar o projeto como uma solução coesa, na qual navegação, controle e visualização trabalham de forma articulada.

Como mostra a Figura 1, o projeto foi organizado como um fluxo simples e integrado: o AGV realiza a leitura da linha, o controle de locomoção corrige o movimento, o processamento *dual-core* gerencia trajetória e comunicação, e os dados seguem por *MQTT* até o servidor e o *dashboard* em tempo real. Essa organização torna a solução mais clara, funcional e adequada à proposta de automação desenvolvida pela equipe.



Como resultado, a equipe construiu um protótipo funcional capaz de seguir a linha da pista, responder a eventos de parada e enviar dados para monitoramento remoto em tempo real. O sistema integrou de forma consistente os elementos principais do projeto, conectando sensores, controle embarcado, comunicação via *MQTT* e *dashboard* de acompanhamento.

Como se observa na Figura 2, o protótipo operou de forma articulada com a interface de visualização, permitindo acompanhar o deslocamento do *AGV* e o comportamento dos dados exibidos no *dashboard*. Nos testes realizados em ambiente controlado, a solução demonstrou estabilidade em trechos retos e curvas, além de confirmar a viabilidade da proposta como base concreta para aplicações futuras de transporte automatizado em pequena escala.

Figura 2: Protótipo do *AGV* em operação e visualização do sistema no *dashboard*



Fonte: Os autores.

Um dos principais desafios foi ajustar o escopo do projeto para algo compatível com o prazo. A proposta inicial precisou ser redirecionada para uma solução mais viável, o que levou à adoção do *AGV* seguidor de linha. Houve também dificuldades com a estrutura mecânica do chassi, que precisou suportar melhor o peso dos próprios sensores, componentes eletrônicos e da carga transportada, exigindo ajustes para alcançar maior estabilidade.

Também foi necessário revisar algumas escolhas técnicas ao longo do desenvolvimento. Inicialmente, foi utilizado um sensor a

laser para medição de distância, mas os testes mostraram instabilidades nas leituras. Por esse motivo, optou-se pela substituição por um sensor ultrassônico, que apresentou funcionamento mais estável e mais adequado à aplicação. Somaram-se a isso os desafios de calibração dos sensores e de integração dos componentes, que demandaram testes sucessivos até atingir um funcionamento confiável.

O projeto trouxe aprendizados importantes tanto no plano técnico quanto no trabalho em equipe. A equipe aprofundou conhecimentos em sistemas embarcados, integração de sensores, controle, comunicação *IoT* e desenvolvimento de interfaces para monitoramento. Também houve ganho na leitura de documentação técnica, na organização de tarefas e na condução de um projeto com etapas iterativas e decisões baseadas em testes. A composição multidisciplinar do grupo favoreceu a troca de conhecimentos entre áreas diferentes e ampliou a compreensão sobre como soluções reais exigem integração entre *software*, eletrônica, organização e comunicação. Como impacto formativo, o projeto fortaleceu a autonomia técnica e a maturidade do grupo.

Como continuidade, o projeto pode evoluir com melhorias na robustez mecânica do protótipo, no tratamento de sinais e na confiabilidade das leituras em operação contínua. Também há espaço para ampliar o *dashboard*, refinar a lógica de navegação e tornar a solução mais modular e fácil de replicar. Em uma etapa futura, a proposta pode ser validada em cenários mais próximos de uso real, com rotas maiores, novas rotinas de transporte e maior maturidade de integração entre *hardware* e *software*.

Outra possibilidade de evolução está no aprimoramento da estrutura física do AGV, com melhor distribuição de peso e maior resistência para suportar componentes adicionais e cargas mais estáveis. Do ponto de vista funcional, a solução também pode incorporar novas rotinas de parada, maior precisão de sensoriamento e uma interface ainda mais completa para acompanhamento remoto e configuração do sistema.

Trilha *FPGA*

Amanda Costa Martinez

(Mentora)

Pedro Henrique Andrade Trindade

(Mentor)

O programa de residência tecnológica *EmbarcaTech* se consolidou como uma iniciativa relevante na formação de profissionais em sistemas embarcados no Brasil. Dentro desse contexto, a trilha de *FPGA* destacou-se por abordar uma área de alta complexidade técnica, caracterizada por uma curva de aprendizado acentuada e forte demanda no setor industrial. Diferentemente de outras trilhas, a formação em *FPGA* exige não apenas domínio conceitual, mas também compreensão arquitetural e capacidade de desenvolvimento em nível de *hardware*.

A execução da trilha ocorreu de forma integralmente remota, envolvendo encontros síncronos e atividades assíncronas. Esse formato foi adotado considerando a distribuição geográfica da equipe e dos participantes, permitindo a integração de diferentes contextos regionais em um ambiente único de formação. Ao todo, 16 residentes participaram da trilha, com perfis majoritariamente iniciantes na área de *FPGA*, o que representou um desafio adicional na condução pedagógica do programa.

A estrutura da trilha foi organizada em dois eixos complementares: aulas teóricas e mentorias técnicas. As aulas tinham como objetivo introduzir os fundamentos de eletrônica digital e computação reconfigurável, enquanto as mentorias focavam na aplicação prática desses conceitos, aproximando os participantes da realidade do desenvolvimento profissional. Essa separação mostrou-se funcional na condução das atividades,

permitindo que os momentos teóricos e práticos se complementassem ao longo da trilha.

O projeto central da trilha foi concebido como o principal instrumento de aprendizado, adotando uma abordagem orientada a projeto, amplamente reconhecida como eficaz em contextos de formação técnica. O desafio proposto consistiu no desenvolvimento de um sistema de processamento de imagens em *FPGA*, com comunicação realizada via interface *UART* entre o computador e o *hardware* reconfigurável.

A arquitetura do projeto foi estruturada em duas macroetapas. Na primeira, os residentes implementaram a comunicação serial *UART* em *VHDL*, integrando os módulos por meio do protocolo *AXI Stream*. Essa etapa teve como objetivo validar a infraestrutura de comunicação, utilizando um mecanismo de *loopback* para garantir a integridade dos dados transmitidos e recebidos. A abordagem incremental permitiu que os participantes consolidassem conceitos fundamentais antes de avançar para níveis mais elevados de complexidade.

Na segunda etapa, o foco deslocou-se para o processamento digital de imagens diretamente em *hardware*. Os residentes desenvolveram algoritmos capazes de operar sobre imagens em escala de cinza, realizando desde operações simples, como inversão de tons, até técnicas mais avançadas, como filtragem por janelas e uso de *buffers* de linha. A introdução de *pipelines* e paralelismo destacou características intrínsecas ao desenvolvimento em *FPGA*, diferenciando-o de abordagens tradicionais baseadas em *software*.

Ao longo do projeto, diversas competências técnicas foram desenvolvidas. Entre elas, destacam-se a descrição de *hardware* em *VHDL*, o uso de máquinas de estados finitos, a implementação de interfaces padronizadas, a integração *hardware-software* e a gestão eficiente de memória e recursos da *FPGA*. Além disso, os residentes foram expostos a práticas profissionais como versionamento de código, organização modular e validação incremental.

Os desafios enfrentados ao longo da trilha foram consistentes com o perfil dos participantes e com a natureza da área. A ausência

de conhecimento prévio em *FPGA* para a maioria dos alunos exigiu uma abordagem pedagógica baseada em progressão gradual e constante adaptação do ritmo. Dificuldades iniciais com *VHDL*, necessidade de maior quantidade de exemplos práticos e percepção de ritmo acelerado foram aspectos recorrentes.

Apesar dos desafios, os resultados obtidos foram expressivos. A totalidade dos participantes avaliou o conteúdo como tecnicamente relevante, e a grande maioria considerou adequada a profundidade abordada. O projeto prático foi amplamente reconhecido como um elemento central no aprendizado, com destaque para a divisão em etapas, que facilitou a compreensão e execução das atividades.

No que se refere ao impacto formativo, observa-se um aumento significativo na confiança dos participantes para atuar em projetos com *FPGA*, sendo possível identificar casos de inserção no mercado de trabalho na área em um curto intervalo após a conclusão da residência. A experiência proporcionou não apenas o aprendizado técnico, mas também uma compreensão mais clara do funcionamento do mercado e das aplicações reais da tecnologia. A aproximação com práticas industriais foi um dos aspectos mais valorizados, reforçando o papel do programa como ponte entre academia e setor produtivo.

Os pontos fortes da mentoria concentraram-se no domínio técnico apresentado, na experiência prática compartilhada e na abordagem orientada a projeto. Esses elementos foram percebidos como diferenciais em relação ao ensino tradicional, contribuindo para uma formação mais alinhada às demandas contemporâneas.

Por outro lado, observou-se, ao longo da execução, oportunidade de aprimoramento na integração entre os momentos teóricos e práticos, especialmente no que se refere ao alinhamento mais explícito entre conteúdos e cronograma. Esse aspecto, comum em iniciativas em fase inicial, não comprometeu o desenvolvimento das atividades, mas aponta para ganhos potenciais de eficiência pedagógica em futuras edições.

Em síntese, a trilha de *FPGA* do *EmbarcaTech* demonstrou ser uma experiência formativa robusta e alinhada às necessidades do mercado tecnológico nacional. A combinação entre teoria e prática, associada a um projeto estruturante, possibilitou o desenvolvimento de competências técnicas e metodológicas essenciais para atuação profissional na área.

A experiência evidencia que, mesmo em formato remoto e com participantes majoritariamente iniciantes, é possível conduzir uma formação de alto nível em *FPGA*, desde que baseada em metodologia adequada, acompanhamento próximo e desafios progressivos. O aprendizado institucional obtido nesta edição fornece base sólida para aprimoramento contínuo e consolidação da trilha como referência em formação na área de *hardware* reconfigurável.

Projetos Desenvolvidos ao Longo da Trilha

Camila Ramos Gomes
(Residente, Vitória da Conquista)
Daniel Alencar Penha Carvalho
(Residente, Juazeiro)
Gabriel Cavalcanti Coelho
(Residente, Juazeiro)
Gustavo Oliveira Alves
(Residente, Vitória da Conquista)
Jacson Souza dos Santos
(Residente, Bom Jesus da Lapa)
Kaian de Souza Gonçalves
(Residente, Vitória da Conquista)
Lucas Almeida Pereira
(Residente, Vitória da Conquista)
Lucas Fialho Xavier
(Residente, Bom Jesus da Lapa)
Luiz Felipe do Rosário Alves Silva
(Residente, Ilhéus)
Maryana Souza Silveira
(Residente, Ilhéus)
Matheus Santos Souza
(Residente, Bom Jesus da Lapa)
Theógenes Gabriel Araújo de Andrade
(Residente, Juazeiro)
Thiago Roberto de Lima Ribeiro
(Residente, Juazeiro)
Vitor Coutinho Lima
(Residente, Ilhéus)
Yago Rodrigo Morato Marques Guirra
(Residente, Juazeiro)
Yohanan Santos de Santana
(Residente, Ilhéus)

O processamento de imagens digitais está presente em quase tudo ao nosso redor, desde os efeitos visuais aplicados em fotos nas redes sociais até os sistemas de visão de carros autônomos, exigindo soluções capazes de operar com alto desempenho e confiabilidade. Essa é uma tarefa naturalmente pesada, pois manipular grandes volumes de dados de forma ininterrupta requer um alto poder computacional. Para aplicar um simples filtro espacial, como destacar as bordas ou suavizar uma foto, milhares de operações matemáticas precisam ser feitas repetidamente sobre cada ponto da imagem e toda a sua vizinhança.

Quando tentamos realizar essa tarefa em arquiteturas convencionais, como nos processadores de propósito geral que encontramos em computadores e notebooks, esbarramos em um limite de processamento sequencial. Nessas arquiteturas, os processos ocorrem de forma linear, executando uma instrução de cada vez. As operações acabam disputando os mesmos recursos internos da máquina, o que gera atrasos indesejados, alto consumo de energia e, muitas vezes, inviabiliza aplicações que exigem respostas instantâneas.

Diante desse cenário, a equipe buscou uma solução nas Field Programmable Gate Arrays (FPGAs). A grande vantagem desses componentes é a capacidade de realizar o trabalho em paralelo que, ao invés de fazer um cálculo por vez, nos permite construir circuitos onde milhares de operações matemáticas ocorrem simultaneamente, oferecendo um desempenho muito superior ao de um programa de computador comum. Portanto, o objetivo geral do projeto foi desenvolver, implementar e validar um sistema de processamento digital de imagens direto no *hardware*. Com isso, consolidamos nosso domínio sobre as linguagens de descrição de *hardware*, mais especificamente a linguagem *VHDL*, que é uma das ferramentas utilizadas para a construção dessas soluções.

Esse projeto serviu como um instrumento de capacitação prática e como uma prova de conceito que validou a arquitetura elaborada, garantindo que o fluxo contínuo de dados fosse recebido, armazenado, calculado e transmitido de volta sem

apresentar gargalos, erros ou interrupções. Dessa forma, foi feito um sistema de processamento de imagens funcional que é escalável para sistemas reais de visão computacional, telecomunicações e controle industrial. Como resultado, o nosso aprendizado foi muito além da teoria, pois assimilamos boas práticas de engenharia, de validação e de documentação que são referências na indústria. Todo esse conjunto de competências técnicas consolidou a nossa habilidade de projetar sistemas modulares e estabeleceu uma base sólida para enfrentarmos os desafios concretos do mercado de tecnologia, marcando a transição definitiva entre o conhecimento acadêmico e a prática real.

No início da trilha de FPGA, a equipe consolidou os conceitos fundamentais de sistemas digitais, partindo de operações com números binários até a lógica combinacional e sequencial. Essa base permitiu a progressão para o uso de ferramentas profissionais, como o *Vivado* e o *Visual Studio Code*, com a extensão *Icarus Verilog*, focando inicialmente na linguagem *Verilog* e *SystemVerilog*, mas expandindo o domínio para *VHDL*.

A integração dessas ferramentas viabilizou a implementação de protocolos de comunicação essenciais, como *UART*, *SPI*, *I²C* e *HDMI*. Além da teoria, a trajetória foi marcada pela aplicação prática em *hardware* real, utilizando a placa *Colorlight i9+* em conjunto com diversos sensores, como o acelerômetro. Essa etapa inicial foi crucial para alinhar o nível técnico do grupo e preparar o terreno para os desafios de desenvolvimento mais complexos que viriam a seguir.

A equipe atuou em conjunto durante todo o processo de aprendizagem e desenvolvimento, de modo que todos realizaram as mesmas atividades, consolidando, ao final, todo o caminho da trilha. Essa estratégia foi importante, pois manteve os residentes em um mesmo nível de conhecimento ao serem atribuídos às mesmas atividades e projetos a todos, visto que a trilha de FPGA tinha como objetivo o desenvolvimento teórico-técnico dos participantes. Ao fim, consolidou-se um projeto final com a

contribuição de todos, visto que os integrantes já haviam realizado tais atividades anteriormente.

A evolução da equipe durante o *Embarcatech* foi não só a nível técnico, mas uma jornada de adaptação à programação a nível de *hardware*. As etapas a seguir consolidam os marcos que definiram essa maturidade no programa:

Domínio da Lógica Programável (VHDL/Verilog): A transição inicial foi o desafio de sair da programação sequencial (*Python/C*) para as *FPGAs*. O domínio de *VHDL* e *Verilog* permitiu à equipe entender como gerenciar sinais em nanossegundos, essencial para projetos de alta performance.

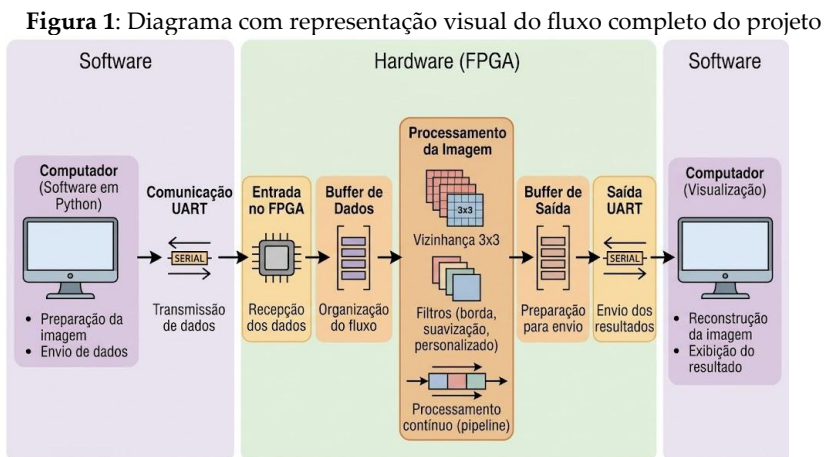
Implementação de Sistemas em FPGA: Um marco técnico divisor de águas foi a integração de processadores *soft-core* com lógica customizada. Aprender a interfacear o processamento de propósito geral com aceleradores de *hardware* específicos na *FPGA* deu à equipe a base para sistemas complexos e robustos.

Projeto Final: O projeto final de processamento de imagem consolidou o aprendizado sobre paralelismo real em *FPGA*. Deste modo, implementaram-se filtros convolucionais diretamente em *hardware*, utilizando matrizes de *buffers* para gerenciar o fluxo de dados em tempo real. O maior aprendizado foi lidar com o gargalo de memória e a latência, em que foram projetadas arquiteturas responsáveis por realizar operações matemáticas simultâneas em cada *pixel* sem que haja o travamento do sistema, transformando conceitos abstratos de filtros e convoluções em uma solução física de alta performance.

A principal mudança ao longo do caminho, foi superar a barreira da programação sequencial para uma linguagem voltada para as *FPGAs*, de tal modo que o design é ditado pelos limites físicos do *hardware* e não apenas pela lógica em si. O maior aprendizado foi a integração entre o processamento de alto nível e aceleradores customizados em *VHDL/Verilog*, garantindo que o tempo real fosse atingido em projetos complexos. Essa jornada ensinou à equipe a ser resiliente diante de erros de *timing* e a otimizar recursos finitos para criar soluções de engenharia viáveis.

A solução desenvolvida pela equipe consistiu na construção de um sistema de processamento de imagens em *FPGA* integrado a um computador, responsável pelo envio e recebimento dos dados. A comunicação entre esses dois elementos é realizada por meio de uma interface serial *UART*, permitindo que a imagem seja transmitida *pixel a pixel* para processamento no *hardware* e posteriormente reconstruída no computador.

Para viabilizar essa interação, também foi desenvolvida uma ferramenta em *Python*, responsável por preparar os dados da imagem no formato adequado para transmissão, além de receber e reorganizar os *pixels* processados. Essa ferramenta permitiu não apenas visualizar os resultados, mas também configurar dinamicamente o tipo de filtro aplicado, como desfoque (*blur*), realce de nitidez (*sharpening*) e detecção de bordas.



Fonte: Os autores.

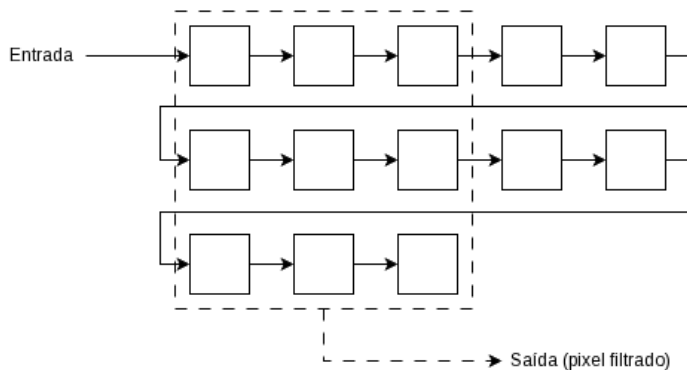
Através deste diagrama, é possível a visualização a nível geral do projeto e seu funcionamento. O computador, através de um programa auxiliar escrito em *Python*, e conectado à *FPGA* via um cabo *USB*, consegue estabelecer um meio de comunicação *UART* com a placa, devido ao *chip* adaptador *CH347* presente na mesma.

Através desta conexão, o módulo de recepção de dados da *FPGA* capta os *pixels* recebidos continuamente pelo computador e os entrega para processamento, com a assistência de duas filas *FIFO* (*First-In, First-Out*), que servem como margens para evitar a perda de *pixels* caso a velocidade de processamento não seja a mesma que a velocidade de transferência. Os *pixels* resultantes são então devolvidos para o computador pelo módulo de saída *UART*.

A fim de evitar o desperdício de memória e possibilitar o processamento contínuo em alta velocidade, a arquitetura da unidade de processamento de imagem consiste em uma *pipeline*, onde *pixels* são movidos como em uma linha de montagem, consistindo na visão de parte da imagem.

Como requisito da implementação, foi necessário um modo de obter uma janela de visão de 3×3 *pixels* da imagem, para cálculo da convolução do *kernel* (filtro) com a imagem, que gera a imagem filtrada.

Figura 2: Arquitetura de *pipeline* utilizado para obtenção do *pixel* processado
 Janela de amostragem
 (3x3)



Fonte: Os autores.

Além desta arquitetura, também foi utilizado o formato de ponto fixo para representação e cálculos numéricos, resultando em circuitos mais simples que uma implementação alternativa

utilizando ponto flutuante. Este design possibilitou o processamento contínuo de *pixels* da imagem, com baixa latência, permitindo o uso em aplicações de tempo real.

A concepção do projeto foi orientada pela busca de uma aplicação que explorasse plenamente as capacidades de processamento paralelo das *FPGAs*. Partindo de estudos prévios sobre protocolos de comunicação com ênfase na interface *UART*, amplamente explorada pela equipe, definiu-se a implementação do sistema de processamento de imagens já descrito. Para garantir a viabilidade e a organização da execução, o desenvolvimento foi estruturado nas etapas fundamentais detalhadas a seguir:

1ª etapa – Planejamento e modelagem do fluxo de dados

O estágio inicial concentrou-se na elaboração de um diagrama de blocos capaz de representar com precisão o fluxo de execução do sistema. O objetivo foi projetar uma arquitetura interna para a *FPGA* que aliasse eficiência no processamento à integridade dos dados. Este planejamento visual permitiu mapear o trajeto completo da informação, desde a origem no computador até o retorno após o processamento.

O fluxo de dados estabelecido segue a lógica: *PC (UART TX) → UART RX (FPGA) → Fila FIFO (Entrada) → Processamento de Imagem → Fila FIFO (Saída) → UART TX (FPGA) → PC (UART RX)*.

Este mapeamento foi crucial para a correta codificação do sistema, servindo como guia para a integração dos módulos de recepção, armazenamento temporário e transmissão.

2ª etapa – Implementação e validação da interface de comunicação

Com a arquitetura definida, iniciou-se o desenvolvimento dos módulos de transmissão (TX) e recepção (RX) do protocolo *UART* (*Universal Asynchronous Receiver Transmitter*). Para assegurar uma

troca de dados estável e confiável entre o computador e a *FPGA*, configurou-se uma taxa de transmissão (*baud rate*) de 9600 bps.

Um aspecto crítico desta implementação foi a integração da **paridade par** (*even parity*) como mecanismo de detecção de erros. O *bit* de paridade foi adicionado ao *frame* de dados com o objetivo de garantir a integridade da informação durante o trajeto serial: o sistema calcula e insere um *bit* adicional para que a soma dos *bits* em nível lógico alto ("1") seja sempre um número par. Caso o receptor detecte uma inconsistência nessa contagem, o dado é identificado como corrompido, mitigando falhas causadas por ruídos na comunicação.

A descrição do protocolo foi fundamentada em uma análise técnica rigorosa dos *datasheets* dos componentes. A validação desta etapa envolveu testes exaustivos e a observação de formas de onda (*Waveforms*) em simuladores, garantindo que o comportamento temporal do *hardware*, incluindo a lógica de paridade, estivesse estritamente alinhado aos requisitos do projeto antes de avançar para as funções de processamento de imagem.

3ª etapa – Implementação e integração de *buffers FIFO*

A terceira fase do projeto consistiu no desenvolvimento e integração dos módulos *FIFO* (*First-In, First-Out*), componentes essenciais para a sincronização do fluxo de dados entre os diferentes domínios de tempo do sistema. A principal função desses *buffers* é atuar como elementos de armazenamento temporário, permitindo que a transmissão via *UART* e o processamento interno ocorram de forma coordenada, evitando o transbordamento de dados (*overflow*) ou a perda de *pixels* durante o ciclo de cálculo.

A arquitetura foi projetada para gerenciar a latência inerente ao processamento: enquanto a *FIFO* de entrada armazena os *pixels* brutos recebidos, a *FIFO* de saída organiza os resultados processados antes de sua retransmissão para o computador. O controle de fluxo foi implementado por meio de um mecanismo de

sinalização (*handshaking*) que monitora os estados de "cheio" (*full*) e "vazio" (*empty*) de cada *buffer*. Dessa forma, o sistema só libera o envio de um novo dado quando o módulo subsequente sinaliza prontidão para o recebimento.

A validação desta etapa envolveu testes de integração rigorosos com os protocolos *UART* desenvolvidos na etapa anterior. Através de simulações e análise de cronogramas de tempo, garantiu-se que a alternância entre escrita e leitura nos *buffers* ocorresse sem conflitos, estabelecendo uma base sólida para a implementação da unidade de processamento de imagem.

4ª etapa – Implementação da unidade de processamento de imagem

Esta etapa compreende o desenvolvimento do núcleo funcional do sistema, responsável pelo processamento aritmético dos pixels recebidos e pela geração da imagem filtrada em tempo real. Com a infraestrutura de comunicação (*UART*) e de armazenamento temporário (*FIFOs*) devidamente validadas, o foco voltou-se para a arquitetura da unidade de processamento.

Para viabilizar a aplicação de filtros baseados em matrizes 3x3, foi necessária a implementação de uma estrutura de *line buffers* (*buffers* de linha). Esta técnica consiste em armazenar internamente três linhas consecutivas da imagem, permitindo que o sistema acesse simultaneamente uma vizinhança de 9 *pixels* (3x3). Sem essa estratégia, o processamento seria limitado a operações ponto a ponto, impossibilitando cálculos de vizinhança como desfoque ou detecção de bordas.

A lógica opera através de uma janela deslizante que percorre a imagem da esquerda para a direita e de cima para baixo. À medida que novos pixels são recebidos pela *UART* e organizados nos *buffers*, a unidade de processamento realiza o cálculo da convolução, multiplicando os valores dos *pixels* pelos coeficientes do *kernel* do filtro selecionado. O resultado deste processamento é

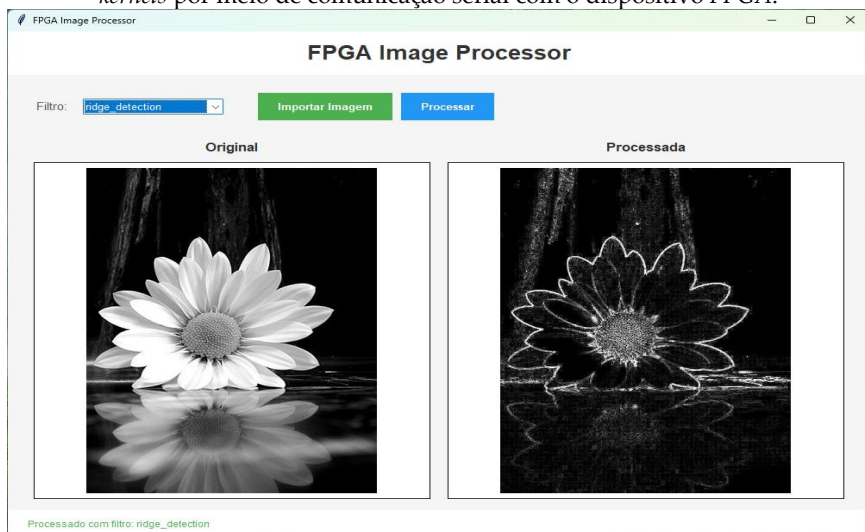
então encaminhado para a *FIFO* de saída, garantindo um fluxo contínuo e sincronizado de dados.

Resultados alcançados

A equipe chegou a um sistema funcionando de ponta a ponta: uma imagem sai do computador, é processada diretamente pelo *hardware* da *FPGA* e retorna transformada, com baixa demora entre o envio e a resposta. Esse ciclo completo, visto funcionando pela primeira vez, foi um marco importante para o grupo.

O sistema processa imagens de 256×256 *pixels*, aplicando filtros matemáticos diretamente no *hardware* de forma contínua e estável. Para facilitar os testes e a operação do sistema, a equipe desenvolveu também uma ferramenta em *Python* que funciona como interface entre o computador e a placa: por meio dela, é possível enviar a imagem, escolher o tipo de filtro a aplicar e visualizar o resultado processado, tudo sem a necessidade de reprogramar o *hardware* a cada alteração.

Figura 3: Interface desenvolvida para realizar o envio e recepção de imagens e *kernels* por meio de comunicação serial com o dispositivo *FPGA*.



Fonte: Os autores.

A validação do sistema ocorreu em duas etapas. Primeiro, o comportamento do *hardware* foi verificado por meio de simulações, confirmando que os cálculos estavam corretos antes de qualquer teste físico. Em seguida, os testes foram realizados na placa real, onde o sistema precisou demonstrar estabilidade ao longo de transmissões completas, sem perda de dados. Essa etapa revelou ajustes que não eram visíveis nas simulações, e resolvê-los foi parte importante da maturidade técnica que a equipe desenvolveu ao longo do projeto.

Com o sistema estabilizado, foram testados três filtros clássicos de processamento de imagens: o *Sobel*, que identifica e destaca as bordas presentes na imagem; o *Gaussiano*, que aplica uma suavização uniforme; e o *sharpening*, que aumenta o contraste e realça os detalhes. Em todos os casos, o *hardware* respondeu de forma precisa, produzindo resultados limpos e fiéis ao efeito esperado de cada filtro. As imagens a seguir apresentam, lado a lado, a entrada original e o resultado após o processamento em cada caso.

Figura 4: Resultado da aplicação do filtro *Sobel* para detecção de bordas utilizando o sistema implementado em *FPGA*. Imagem original à esquerda.



Fonte: Os autores.

O resultado do filtro *Sobel* evidencia a precisão com que o sistema identificou os contornos da imagem, destacando as bordas sem distorcer as demais regiões, o que confirma o correto funcionamento dos cálculos que consideram a vizinhança de cada *pixel*.

Figura 5: Resultado da aplicação do filtro *Gaussiano*, demonstrando o efeito de suavização obtido pelo processamento em hardware. Imagem original à esquerda.



Fonte: Os autores.

O filtro *Gaussiano* produziu uma suavização equilibrada em toda a extensão da imagem, reduzindo variações bruscas entre regiões. O resultado confirma que os pesos do filtro foram aplicados de maneira uniforme e consistente pelo *hardware*.

Figura 6: Resultado da aplicação do filtro de realce de nitidez (*sharpening*), destacando o aumento do contraste em regiões de detalhe da imagem.



Fonte: Os autores.

O filtro de realce de nitidez demonstrou a capacidade do sistema de amplificar diferenças entre *pixels* próximos, tornando os detalhes da imagem mais pronunciados. A ausência de distorções ou exageros no resultado indica que o processamento interno estava bem calibrado.

Desafios enfrentados

Um dos principais obstáculos enfrentados foi a transposição de conceitos teóricos complexos de Sistemas Digitais para uma aplicação prática e acessível. Como o grupo era bastante heterogêneo, surgiu o desafio de nivelar o aprendizado: havia participantes que não eram da área de computação, outros que ainda não tinham cursado disciplinas básicas e alguns que sequer possuíam graduação, o que exigiu estratégias didáticas que permitissem o avanço coletivo.

Além das questões pedagógicas, a organização do fluxo de trabalho trouxe desafios logísticos e técnicos significativos. A leitura de materiais técnicos e a execução das tarefas exigiam não apenas domínio do conteúdo, mas também clareza na comunicação interna, o que promoveu apoio mútuo nas reuniões síncronas e interações em outras plataformas. Somado a isso, compreender como diferentes partes do sistema se comunicavam entre si foi outro desafio, visto que muitas vezes algo funcionava isoladamente, mas apresentava falhas quando conectado ao sistema completo.

No que diz respeito aos aspectos práticos, a maioria conseguiu desenvolver os exercícios propostos, apesar de alguns percalços. Houve atrasos pontuais devido à disponibilidade de placas e sensores necessários para as atividades, além de dificuldades técnicas relacionadas aos sistemas operacionais dos computadores de cada residente. Esses problemas foram superados com o apoio dos professores e, principalmente, pela colaboração entre os integrantes do grupo, que se mostraram muito solícitos ao auxiliar na configuração dos ambientes de desenvolvimento e na resolução das dificuldades encontradas.

Por fim, o grupo enfrentou o desafio de manter um engajamento contínuo ao longo das atividades. Construir esse ambiente de aprendizado colaborativo foi essencial para o êxito da trilha. Esse processo funcionou como um catalisador para o aprimoramento das competências interpessoais, consolidando a capacidade coletiva de resolver problemas em equipe e de se adaptar a cenários educacionais desafiadores.

Aprendizados da equipe

Ao longo do desenvolvimento do projeto, a equipe adquiriu conhecimentos técnicos relevantes relacionados ao uso de *FPGAs*, especialmente na implementação de sistemas digitais descritos em *hardware*. Foram consolidados conceitos como a comunicação serial via *UART*, controle de fluxo inspirado em protocolos como o *AXI-*

Stream, uso de *buffers* de linha (*line buffers*) e aplicação de filtros espaciais por meio de convoluções 3x3. Além disso, os integrantes aprofundaram a compreensão sobre arquiteturas em *pipeline* e processamento em fluxo contínuo (*streaming*), fundamentais para aplicações de alto desempenho em *hardware*.

No aspecto colaborativo, o projeto exigiu organização e divisão de tarefas entre os membros da equipe, promovendo a troca constante de conhecimentos e a integração entre diferentes etapas do sistema. A necessidade de alinhar interfaces, testar módulos individualmente por meio de *testbenches* e, posteriormente, integrá-los evidenciou a importância da comunicação clara e do trabalho em grupo, especialmente em projetos de maior complexidade.

Também houve um significativo desenvolvimento pessoal e profissional, uma vez que os desafios enfrentados ao longo do projeto exigiram autonomia, persistência e capacidade de resolução de problemas, ao mesmo tempo em que fortaleceram competências como pensamento crítico, organização, gestão do tempo e adaptabilidade frente a imprevistos.

Situações como erros de sincronização, perda de dados na comunicação serial e ajustes no processamento da imagem favoreceram o amadurecimento técnico da equipe, exigindo ciclos iterativos de testes, análise de sinais e refinamento do código em *HDL*. Esse processo permitiu compreender, na prática, a importância do timing, das interações reais de engenharia. O projeto permitiu compreender, na prática, como os sistemas digitais são projetados, descritos, simulados, integrados e validados, utilizando ferramentas e fluxos semelhantes aos do ambiente profissional. Essa vivência contribuiu diretamente para a preparação dos integrantes para desafios futuros na área de sistemas embarcados e processamento digital, fortalecendo tanto a base técnica quanto às competências exigidas pelo mercado.

Por fim, a trilha FPGA teve papel fundamental na formação dos estudantes, proporcionando uma experiência prática que vai além do conteúdo teórico, aproximando-os de aplicações reais de engenharia. O projeto permitiu compreender, na prática, como os

sistemas digitais são projetados, descritos, simulados, integrados e validados, utilizando ferramentas e fluxos semelhantes aos do ambiente profissional. Essa vivência contribuiu diretamente para a preparação dos integrantes para desafios futuros na área de sistemas embarcados e processamento digital, fortalecendo tanto a base técnica quanto às competências exigidas pelo mercado.

Perspectivas futuras

O sistema de análise de imagens em *FPGA*, resultado deste projeto, é mais um começo do que uma conclusão. A estrutura adaptável e expansível criada pela equipe facilita diversas melhorias, tanto na área técnica quanto nas aplicações práticas.

Em termos de *hardware*, um avanço rápido seria suportar imagens com mais detalhes, ultrapassando o limite atual de 256×256 *pixels*. Isso precisaria de um controle melhor dos recursos de memória interna da *FPGA*, mas a base já criada – com *buffers* de linha e processamento contínuo – oferece o suporte necessário para essa expansão. Incluir cores (*RGB*), em vez de apenas tons de cinza, permitiria usar o sistema em muitas outras situações.

Ainda na parte técnica, é possível adicionar filtros mais complexos e redes neurais convolucionais (*CNNs*) diretamente no *hardware*, seguindo a tendência de usar inteligência artificial em dispositivos que gastam pouca energia. Trocar a interface *UART* por opções mais rápidas, como *USB 3.0* ou *Ethernet*, também resolveria o problema de comunicação identificado durante o desenvolvimento, permitindo processar vídeos em tempo real.

Nas aplicações, o sistema tem potencial para ser usado em áreas como inspeção industrial automática, visão em veículos autônomos, telecomunicações e equipamentos científicos – onde a rapidez e a precisão são essenciais e onde as soluções baseadas em *FPGA* já são melhores que as opções tradicionais.

Além disso, uma evolução natural do projeto envolve a melhoria da modularidade e reutilização dos blocos desenvolvidos, permitindo que a arquitetura seja facilmente

adaptada para diferentes aplicações e escalas de complexidade. A adoção de interfaces padronizadas, como *AXI-Stream*, já utilizada no projeto, reforça esse potencial ao facilitar a integração com outros sistemas. Outro ponto relevante é a possibilidade de desenvolvimento de ferramentas de validação e testes mais robustos, como bancos de teste automatizados e simulações mais completas, aumentando a confiabilidade do sistema.

Finalmente, o caminho percorrido pela equipe no *EmbarcaTech* oferece uma visão que vai além do projeto. Aprender a programar *hardware*, lidar com problemas de tempo e criar soluções que funcionam bem preparou os participantes para uma forma de engenharia que raramente se aprende apenas na teoria. Essa experiência técnica e de trabalho em equipe capacita a equipe a contribuir em projetos mais complexos, seja na universidade, em pesquisas ou na indústria de sistemas embarcados.

Este projeto foi apoiado pelo Ministério da Ciência, Tecnologia e Inovações,
com recursos da Lei nº 8.248, de 23 de outubro de 1991,
no âmbito do PPI-SOFTEX, coordenado pela Softex e publicado
Residência em TIC 37, DOU 01245.001849/2024-76.

Executores:



INSTITUTO FEDERAL
Ceará



INSTITUTO FEDERAL
Rio Grande do Norte



INSTITUTO FEDERAL
Piauí



INSTITUTO FEDERAL
Maranhão



INSTITUTO
HARDWARE BR

Coordenação:



MINISTÉRIO DA
CIÊNCIA, TECNOLOGIA
E INOVAÇÃO

