

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS CAJAZEIRAS
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

**UNIACCESS: UMA APLICAÇÃO QUE AUXILIA NO
GERENCIAMENTO DO ACESSO E CONSUMO DAS REFEIÇÕES DO
RESTAURANTE ESTUDANTIL DO IFPB - CAMPUS CAJAZEIRAS**

HÉLIO JOSÉ DA SILVA JÚNIOR

**Cajazeiras
2025**

HÉLIO JOSÉ DA SILVA JÚNIOR

**UNIACCESS: UMA APLICAÇÃO QUE AUXILIA NO GERENCIAMENTO DO ACESSO
E CONSUMO DAS REFEIÇÕES DO RESTAURANTE ESTUDANTIL DO IFPB -
CAMPUS CAJAZEIRAS**

Trabalho de Conclusão de Curso apresentado junto ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba - Campus Cajazeiras, como requisito à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador

Prof. Me. Fabio Abrantes Diniz.

**Cajazeiras
2025**

IFPB / Campus Cajazeiras
Coordenação de Biblioteca
Biblioteca Prof. Ribamar da Silva
Catalogação na fonte: Cícero Luciano Félix CRB-15/750

S586u Silva Júnior, Hélio José da.

Uniaccess : uma aplicação que auxilia no gerenciamento do acesso e consumo das refeições do restaurante estudantil do IFPB - campus Cajazeiras / Hélio José da Silva Júnior. – Cajazeiras, 2025.

83f. : il.

Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) – Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Cajazeiras, 2025.

Orientador: Prof. Me. Fabio Abrantes Diniz.

1. Desenvolvimento de sistemas. 2. Sistema de gestão. 3. Restaurante universitário. 4. Automação. I. Instituto Federal da Paraíba. II. Título.

IFPB/CZ

CDU: 004.4(043.2)



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

HELIO JOSE DA SILVA JUNIOR

**UNIACCESS: UMA APLICAÇÃO QUE AUXILIA NO GERENCIAMENTO DO ACESSO E
CONSUMO DAS REFEIÇÕES DO RESTAURANTE ESTUDANTIL DO IFPB - CAMPUS
CAJAZEIRAS**

Trabalho de Conclusão de Curso apresentado junto ao
Curso Superior de Tecnologia em Análise e
Desenvolvimento de Sistemas do Instituto Federal de
Educação, Ciência e Tecnologia da Paraíba - Campus
Cajazeiras, como requisito à obtenção do título de
Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador

Prof. Me. Fábio Abrantes Diniz

Aprovada em: **25 de Março de 2025.**

Prof. Me. Fábio Abrantes Diniz - Orientador

Prof. Dr. Fabio Gomes de Andrade - Avaliador
IFPB - Campus Cajazeiras

Prof. Me. Janderson Ferreira Dutra - Avaliador
IFPB - Campus Cajazeiras

Documento assinado eletronicamente por:

- **Fabio Abrantes Diniz**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 25/03/2025 15:55:51.
- **Fabio Gomes de Andrade**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 25/03/2025 17:42:35.
- **Janderson Ferreira Dutra**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 25/03/2025 23:29:08.

Este documento foi emitido pelo SUAP em 25/03/2025. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 688318
Verificador: cf7f3075f1
Código de Autenticação:



Rua José Antônio da Silva, 300, Jardim Oásis, CAJAZEIRAS / PB, CEP 58.900-000
<http://ifpb.edu.br> - (83) 3532-4100

RESUMO

A automatização dos processos do gerenciamento dos refeitórios universitários é uma tendência em crescimento no Brasil. O controle do acesso ao consumo em refeitórios de instituições de ensino ainda é um processo manual, ocasionando problemas como o aumento do tempo de espera e fraudes no acesso ao refeitório. Nesse contexto, a aplicação da tecnologia biométrica pode trazer uma série de benefícios para a gestão de refeitórios em instituições de ensino, garantindo um atendimento mais rápido, seguro e conveniente para os usuários. No entanto, Outro grande desafio enfrentado pelas Instituições de Ensino é o desperdício de alimentos, oriundos dos processos manuais da gestão do estoque de alimentos. Logo, a presença da Tecnologia da Informação na otimização da gestão administrativa do refeitório das Instituições é primordial, pois isso permite uma conexão automatizada entre os setores que gerem o refeitório. Aprimorando-se o monitoramento no uso do refeitório e controle do estoque das refeições. Portanto, para suprir essas demandas, neste trabalho foi desenvolvida uma aplicação contendo três serviços que vão auxiliar a Coordenação de Apoio ao Estudante (CAEST) do IFPB - Campus Cajazeiras no gerenciamento do consumo das refeições do Restaurante Estudantil (RE). A aplicação tem um serviço web, a ser utilizado pela CAEST, que vai gerenciar os registros dos discentes contemplados para o acesso ao RE. Gerencia um serviço biométrico para identificação dos discentes na entrada do RE. Por fim, disponibiliza um serviço *mobile* para uso dos discentes, para confirmar a pretensão diária da refeição. O último serviço é importante para que a equipe do RE possa se programar, diariamente, sobre a quantidade de refeições que devem ser preparadas.

Palavras-chave: Refeitório universitário, Automatização, Controle de acesso, Biometria, *Web*, *Mobile*.

ABSTRACT

The automation of university cafeteria management processes is a growing trend in Brazil. In this context, the application of biometric technology can bring a number of benefits to the management of cafeterias in educational institutions, ensuring faster, safer and more convenient service for users. However, another major challenge faced by educational institutions is food waste, stemming from the manual processes of food stock management. Therefore, the presence of Information Technology in optimizing the administrative management of cafeterias in institutions is essential, as it allows for an automated connection between the departments that manage the cafeteria, improving the monitoring of cafeteria usage and meal stock control. To meet these demands, this work developed an application containing three services that will assist the Student Support Coordination (CAEST) at IFPB - Cajazeiras Campus in managing meal consumption at the Student Restaurant (RE). The application includes a web service, to be used by CAEST, to manage the records of students authorized to access the RE. It also manages a biometric service for identifying students at the RE entrance. Finally, it offers a mobile service for students to confirm their daily meal intentions. This last service is crucial for the RE team to plan, on a daily basis, the number of meals that need to be prepared.

Keywords: School Cafeteria, Automation, Access Control, Biometry, Web, Mobile.

LISTA DE FIGURAS

Figura 1 – Fluxograma de controle do Refeitório Estudantil do IFPB Campus Cajazeira	24
Figura 2 – Padrão MVC	27
Figura 3 – Funcionamento da Arquitetura Limpa	29
Figura 4 – Alguns tipos de pontos característicos	31
Figura 5 – Tela principal do módulo administrativo do SISRU	34
Figura 6 – Tela de vendas filtrada por usuário e por período de tempo e tela de venda de ticket filtrada por período de tempo	35
Figura 7 – Tela para reserva ou compra de refeições e tela para cadastrar o cardápio	36
Figura 8 – Casos de uso	41
Figura 9 – Arquitetura do sistema	42
Figura 10 – Arquitetura do sistema	43
Figura 11 – Página de Login	46
Figura 12 – Página principal - Módulo de Relatórios	46
Figura 13 – Página principal - Módulo de Relatórios	47
Figura 14 – Módulo do Refeitório	48
Figura 15 – Módulo Controle de Listas - Lista selecionada	49
Figura 16 – Módulo Gerência de Alunos - Aluno selecionado	50
Figura 17 – Aplicativo UniAccess - Página inicial de <i>login</i>	51
Figura 18 – Aplicativo UniAccess - Página principal	52
Figura 19 – Aplicativo UniAccess - Modal para alterar senha	54
Figura 20 – Aplicativo UniAccess - Mensagem de refeitório indisponível	55
Figura 21 – Aplicativo UniAccess - Mensagem de presença confirmada	56
Figura 22 – Modelo conceitual do Banco de Dados	62
Figura 23 – Página de apresentação	65
Figura 24 – Menu lateral de navegação	66
Figura 25 – Módulo Controle de Usuários - Listagem	67
Figura 26 – Módulo Refeitório Interno	67
Figura 27 – Módulo Controle de Usuários - Cadastro de novo usuário	68
Figura 28 – Módulo Controle de Usuários - Dados do usuário	69
Figura 29 – Módulo Controle de Usuários - Editar usuário	69
Figura 30 – Módulo Controle de Usuários - Modal de exclusão do usuário	70
Figura 31 – Módulo Controle de Listas - Listagem	70
Figura 32 – Módulo Controle de Listas - Modal para criar lista	71
Figura 33 – Módulo Controle de Listas - Editar Lista alunos	72

Figura 34 – Módulo Controle de Listas - Editar Lista alunos extras	72
Figura 35 – Módulo Controle de Listas - Modal de exclusão de lista	73
Figura 36 – Módulo Gerência de Alunos - Listagem	73
Figura 37 – Módulo Gerência de Alunos - Cadastrar novo aluno	74
Figura 38 – Módulo Gerência de Alunos - Edição do aluno	74
Figura 39 – Módulo Gerência de Alunos - Modal de exclusão do aluno	75
Figura 40 – Módulo Perfil	76
Figura 41 – Módulo Perfil - Alterar foto de perfil	76
Figura 42 – Módulo Perfil - Editar perfil	77
Figura 43 – Módulo Template - Listagem	77
Figura 44 – Módulo Template - Criar novo template	78
Figura 45 – Módulo Template - Criar novo template extras	78
Figura 46 – Módulo Template - Dados do template	79
Figura 47 – Módulo Template - Editar	80
Figura 48 – Módulo Template - Editar extras	80
Figura 49 – Módulo Template - Modal de exclusão do template	81
Figura 50 – Mensagem de erro	81
Figura 51 – Mensagem de sucesso	81
Figura 52 – Aplicativo UniAccess - Tela Principal tema escuro	82
Figura 53 – Aplicativo UniAccess - Modal para alterar senha tema escuro	83
Figura 54 – Aplicativo UniAccess - Mensagem de presença confirmada tema escuro	84
Figura 55 – Aplicativo UniAccess - Mensagem de presença confirmada tema escuro	85

LISTA DE QUADROS

Quadro 1 – Comparação das características dos trabalhos relacionados	37
Quadro 2 – Requisitos Funcionais	40
Quadro 3 – <i>User Stories</i>	64

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
CAEST	Coordenação de Apoio ao Estudante
CSS	<i>Cascading Style Sheets</i>
CSV	<i>Comma-Separated values</i>
CRUD	<i>Create Read Update Delete</i>
CZ	Cajazeiras
DDI	Diretoria de Desenvolvimento Institucional
DTO	<i>Data Transfer Object</i>
HTML	<i>Hypertext Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
ID	Impressão Digital
IE	Instituição de Ensino
IF	Instituto Federal
IFPB	Instituto Federal da Paraíba
IVS	Índice de Vulnerabilidade Social
JPG	<i>Joint Photographic Experts Group</i>
JSON	<i>Javascript Object Notation</i>
MG	Minas Gerais
MVC	<i>Model-View-Controller</i>
PAPE	Programa de Apoio à Permanência do Estudante
PNAES	Plano Nacional de Assistência Estudantil
PNG	<i>Portable Network Graphics</i>
RE	Restaurante Estudantil
RF	Requisito Funcional
RFID	<i>Radio Frequency Identification</i>
RI	Restaurante Institucional
RU	Restaurante Universitário
SARI	Sistema de Acesso ao Restaurante Institucional
SDK	<i>Software Development Kit</i>

SISRU	Sistema de Automação de Restaurantes Universitários
SUAP	Sistema Unificado de Administração Pública
SVG	<i>Scalable Vector Graphic</i>
TCC	Trabalho de Conclusão de Curso
UFERSA	Universidade Federal Rural do Semi-Árido
US	<i>User Story</i>

SUMÁRIO

1	INTRODUÇÃO	15
1.1	JUSTIFICATIVA	16
1.2	OBJETIVOS	17
1.3	ORGANIZAÇÃO DO TRABALHO	18
2	METODOLOGIA	19
3	FUNDAMENTAÇÃO TEÓRICA	20
3.1	FATORES DO DESPERDÍCIO DE ALIMENTOS E O PAPEL DA EDUCAÇÃO NA REDUÇÃO	20
3.2	GESTÃO DA ASSISTÊNCIA ESTUDANTIL E REFEITÓRIO DAS INSTITUIÇÕES DE ENSINO	21
3.3	ABORDAGEM DE COMBATE AO DESPERDÍCIO NO IFPB CAMPUS CAJAZEIRAS	22
3.3.1	GESTÃO DAS REFEIÇÕES DO RESTAURANTE ESTUDANTIL DO IFPB - CAMPUS CAJAZEIRAS	23
3.4	PADRÕES E ARQUITETURAS DE DESENVOLVIMENTO	25
3.4.1	PADRÃO ARQUITETURAL MVC	26
3.4.2	ARQUITETURA LIMPA	28
3.5	SISTEMAS DE AUTENTICAÇÃO BIOMÉTRICOS	29
3.6	BIOMETRIA DE IMPRESSÃO DIGITAL (<i>FINGERPRINT</i>)	30
4	TRABALHOS RELACIONADOS	33
4.1	SISTEMA DE AUTOMAÇÃO DE RESTAURANTES UNIVERSITÁRIOS (SISRU)	33
4.2	SISTEMA PARA MODERNIZAÇÃO DO RESTAURANTE DO IF SUDESTE MG - CAMPUS JUIZ DE FORA. (SRE)	34
4.3	SISTEMA DE ACESSO AO RESTAURANTE INSTITUCIONAL (SARI)	36
4.4	ANÁLISE COMPARATIVA	37
5	UNIACCESS	39
5.1	ELICITAÇÃO DOS REQUISITOS	39

5.2	PROJETO ARQUITETURAL DO SISTEMA	42
5.2.1	DETALHAMENTO DA SOLUÇÃO ARQUITETURAL	43
5.3	IMPLEMENTAÇÃO	44
5.3.1	PRINCIPAIS FUNCIONALIDADES	45
6	CONSIDERAÇÕES FINAIS	58
6.1	TRABALHOS FUTUROS	58
	REFERÊNCIAS	59
	APÊNDICE A – MODELAGEM CONCEITUAL DO BANCO DE DADOS E <i>USER</i> <i>STORIES</i>	62
	APÊNDICE B – PÁGINAS E MÓDULOS DOS SISTEMAS	65

1 INTRODUÇÃO

O Refeitório das Instituições de Ensino (IE) desempenha um papel fundamental na implementação de políticas de apoio estudantil, oferecendo aos discentes uma alimentação equilibrada e acessível. Esse serviço visa criar condições favoráveis para a permanência dos estudantes na instituição, além de contribuir para o aprimoramento de seu desempenho acadêmico (HÄRTER et al., 2013). Entretanto, muitas vezes, a gestão desses espaços é ineficiente, trazendo problemas como filas extensas, erros no controle de estoque e desperdício de alimentos.

Dentro desse contexto, a gestão do Instituto Federal da Paraíba (IFPB) - Campus Cajazeiras anseia automatizar processos do gerenciamento do consumo das refeições do Restaurante Estudantil (RE), a fim de reduzir custos e otimizar o atendimento aos discentes.

O programa de alimentação que utiliza o RE do IFPB campus Cajazeiras é coordenado pela Coordenação da Assistência ao Estudante (CAEST) (Resolução, 40/2011). A CAEST é um setor do IFPB CZ que gerencia o restaurante estudantil e as assistências estudantis. Ele tem como suas funções, planejar, mobilizar e discutir direitos a assistência estudantil no âmbito do IFPB - Campus Cajazeiras. De acordo com a CAEST (2023), todo o processo de acesso e consumo das refeições é realizado manualmente, o que resulta em atrasos significativos e prazos extensos para a conclusão dos editais. Além disso, ocorrem alterações frequentes na lista de alunos, o que exige que, após cada atualização, a CAEST imprima e entregue a versão atualizada no restaurante estudantil, gerando, assim, um alto consumo de papel.

Outro aspecto de grande preocupação para a Direção do IFPB Campus Cajazeiras é o desperdício de alimentos. As refeições são preparadas com base no número de estudantes presentes na lista diária elaborada pela CAEST. No entanto, quando um discente não comparece à refeição, ela acaba sendo desperdiçada. Esse problema ocorre com certa frequência. Além disso, existe o risco de discentes não contemplados pela CAEST tentarem se passar pelos contemplados, ocupando indevidamente vagas e burlando os procedimentos de controle estabelecidos. Outra questão é a possível imprecisão no preenchimento da lista de alunos, o que pode resultar na produção de alimentos além da necessidade.

Segundo Berbigier e Magalhães (2018), a diminuição do desperdício de alimentos é considerada um dos principais elementos para aprimorar a sustentabilidade

e eficiência no aproveitamento de recursos. Assim, por meio do controle preciso de acesso ao restaurante estudantil viabilizado pelo aplicativo *mobile*, é possível gerenciar com certa precisão a quantidade de refeições a serem preparadas, resultando em uma redução significativa do desperdício de alimentos. Portanto, neste trabalho foi desenvolvido uma aplicação contendo três serviços para solucionar os problemas gerenciais do acesso e consumo das refeições do RE do IFPB - Campus Cajazeiras. A aplicação tem um serviço *web* que gerencia os registros dos discentes contemplados para o acesso ao RE. Conta com um serviço biométrico para identificação dos discentes na entrada ao RE. E um serviço *mobile* utilizado pelos discentes para confirmar a pretensão diária da refeição.

A aplicação permite que os discentes acessem suas informações e visualizem os dias em que têm permissão para frequentar o refeitório por meio do aplicativo móvel. Os alunos contemplados deverão confirmar sua presença no aplicativo nos respectivos dias planejados pela CAEST. Essa confirmação é necessária antes do envio diário da lista ao refeitório, garantindo que a vaga seja temporariamente disponibilizada para outros alunos caso não seja confirmada. O serviço biométrico, com um leitor de impressão digital, assegura o acesso aos discentes autorizados que tenham previamente confirmado sua intenção de frequentar o refeitório na data especificada. Portanto, espera-se uma melhoria na eficiência do gerenciamento dos registros dos discentes contemplados pela CAEST, redução do tempo necessário para a elaboração das listas e controle preciso de acesso dos alunos, contribuindo para a diminuição do desperdício de alimentos.

1.1 JUSTIFICATIVA

Após uma entrevista com o responsável do setor da CAEST, foi observado que o controle do programa de alimentação, o qual garante aos discentes o consumo das refeições do RE do IFPB Campus Cajazeiras, enfrenta uma excessiva carga de trabalho manual. Existe, atualmente, nas atividades do gerenciamento do controle das refeições do refeitório, um uso diário de, em média, 10 folhas impressas de planilhas para controlar as listas de discentes cadastrados. E, conforme apontado por Oliveira (2019), o emprego de planilhas impressas para administrar o acesso a um ambiente acarreta uma limitada manipulação dos dados, bem como compromete o tempo de trabalho dos profissionais. Diante desse panorama, é de vital importância conceber um serviço *web* que permita à CAEST realizar o gerenciamento da relação dos discentes aprovados. Além disso, foi observado o compartilhamento de planilhas *Google Sheets* da Google, as quais muitas vezes sofrem a perda de informações, provocadas pelo mau gerenciamento do controle das versões dessas planilhas.

Diante dessa sobrecarga de trabalho que consome uma quantidade significativa de tempo, a aplicação foi desenvolvida com o propósito de auxiliar a CAEST do IFPB - Campus Cajazeiras no gerenciamento do acesso ao RE e no consumo das suas refeições contemplados pelos programas de alimentação. A aplicação tem um serviço *web*, a ser utilizado pela CAEST, com o propósito de gerenciar o cadastramento dos discentes contemplados no edital para o acesso ao RE. Essa aplicação faz a integração com um serviço biométrico de leitura digital para identificação dos discentes na entrada ao RE. Além disso, a aplicação possui uma versão *mobile* utilizada pelos discentes para confirmar a pretensão diária da refeição. Serviço importante para que a equipe do RE possa se programar diariamente sobre a quantidade de comida a ser preparada.

O sistema deve proporcionar uma série de benefícios, tanto no âmbito pessoal como para a CAEST e o restaurante estudantil do IFPB - CZ. Caso o sistema proposto seja implementado, espera-se uma significativa mudança de cultura, tanto na CAEST quanto no RE, abandonando métodos manuais e controles de acesso baseados em folhas impressas, em prol de sistemas totalmente automatizados e de fácil compreensão.

1.2 OBJETIVOS

Objetivo geral

Desenvolver uma aplicação que auxilie a CAEST do IFPB - Campus Cajazeiras no gerenciamento do acesso ao Restaurante Estudantil e no consumo das refeições contemplados pelos programas de alimentação.

Objetivos específicos

São objetivos específicos deste trabalho:

- Documentar os motivos que provocam o desperdício de alimentos no RE do IFPB - CZ;
- Explicar o gerenciamento dos processos relacionados ao refeitório e identificar oportunidades de melhorias;
- Pesquisar e avaliar os diferentes modelos de sensores biométricos disponíveis no mercado, para que seja integrado ao trabalho proposto;
- Implementar um sistema escalável e confiável para o gerenciamento e controle de acesso ao Refeitório.

- Reduzir o desperdício de alimentos no refeitório do IFPB - CZ.

1.3 ORGANIZAÇÃO DO TRABALHO

A organização deste trabalho foi estruturada para seguir a sequência lógica dos objetivos e temas abordados ao longo da pesquisa. O Capítulo 2 apresenta a metodologia utilizada, detalhando os procedimentos e abordagens adotadas para a elaboração do trabalho. No Capítulo 3, são discutidos os conceitos teóricos fundamentais, com ênfase nos fatores do desperdício de alimentos, a gestão da assistência estudantil, e as arquiteturas de desenvolvimento aplicadas. O Capítulo 4 trata dos trabalhos relacionados, incluindo uma análise comparativa entre diferentes sistemas de automação de restaurantes universitários.

No Capítulo 5, é apresentado o *UNIACCESS*, desde a elicitação dos requisitos até o projeto arquitetural detalhado, seguido da implementação das principais funcionalidades. Finalmente, o Capítulo 6 traz as considerações finais, abordando as conclusões da pesquisa e as possibilidades de futuras melhorias.

2 METODOLOGIA

Para a realização do trabalho proposto, foi estabelecido um fluxo de atividades. Nas subseções a seguir, encontram-se detalhadas as etapas desse fluxo.

- **Definição do tema e planejamento inicial. (A1)** Nesta etapa inicial do projeto foi delimitado o escopo do trabalho.
- **Realização de estudos embasados em literatura nacional e/ou internacional para compreender os fundamentos relacionados à construção desse ensino. (A2)** Esta etapa buscou dar embasamento para os argumentos apresentados na proposta do trabalho. Para a realização desta etapa foi feita uma pesquisa sobre trabalhos relacionados à educação para o combate ao desperdício de alimentos, gestão de refeitórios universitários, padrões arquiteturais escalonáveis e sistemas de autenticações biométricas.
- **Escolha do nome para o aplicativo proposto. (A3)** Etapa onde foi escolhido o nome para a aplicação.
- **Início da redação do trabalho. (A4)** Etapa onde o trabalho começou a ser escrito.
- **Levantamento dos requisitos para compreender e solucionar o problema inicialmente proposto. (A5)** Nesta etapa, os requisitos foram levantados após leitura das pesquisas bibliográficas e entrevistas com o setor da CAEST.
- **Visão geral dos requisitos para uma melhor compreensão das funcionalidades do sistema. (A6)** Nesta etapa é feita análise dos requisitos levantados com base nos estudos ao longo deste trabalho.
- **Definição do escopo do aplicativo e suas funcionalidades. (A7)** Nesta etapa é definido o escopo do sistema e suas funcionalidades com base nos requisitos analisados.
- **Implementação da aplicação. (A8)** Nesta etapa foram desenvolvidos a API, o Sistema *Web* e o Aplicativo *Mobile* e todas as suas funcionalidades.
- **Revisão e elaboração do documento Final. (A9)** Por fim, nesta etapa foram feitas as últimas correções e ajustes, para a confecção do documento final.

3 FUNDAMENTAÇÃO TEÓRICA

Nas subseções abaixo, serão apresentados os fatores que ocasionam o desperdício de alimentos, assim como a educação age como uma ferramenta para combater isso. Após isso, será abordado um pouco sobre como é feita a gestão da assistência estudantil e dos refeitórios em instituições de ensino. Por fim, será explicado como o IFPB-CZ lida com o desperdício de alimentos e como funciona o Restaurante Estudantil do Campus Cajazeiras.

3.1 FATORES DO DESPERDÍCIO DE ALIMENTOS E O PAPEL DA EDUCAÇÃO NA REDUÇÃO

Segundo Ricarte et al. (2021), o desperdício de alimentos ocorre quando não são utilizados ou quando pratos são preparados, mas não são consumidos e precisam ser descartados. A compreensão das causas e a adoção de estratégias para minimizar as perdas e desperdícios alimentares são preocupações centrais para profissionais envolvidos na área da alimentação em todas as suas formas (RICARTE et al., 2021). Essa abordagem é impulsionada pelos significativos impactos econômicos, ambientais e sociais que afetam globalmente a sustentabilidade do sistema alimentar (DELIBERADOR, 2019).

O padrão alimentar da população tem grande importância quando se trata de desperdício de comida. Segundo Firgulha et al. (2017), os padrões alimentares da população global são moldados por uma variedade de fatores, como valores culturais, convicções regionais, condições climáticas, práticas agrícolas, avanços tecnológicos e considerações econômicas, como resultado, observa-se uma diversidade de hábitos alimentares entre diferentes regiões e até mesmo dentro de um mesmo país.

Um exemplo de padrão alimentar é a dieta mediterrânea, amplamente adotada em países do Mediterrâneo, como Grécia, Itália e Espanha. Conforme explicado no artigo de Durão, Oliveira e Almeida (2008), essa dieta é caracterizada pelo consumo abundante de frutas, legumes, cereais integrais, leguminosas, nozes e azeite de oliva, além de incluir moderadamente peixes, aves, laticínios e vinho tinto.

De acordo com Silvério e Oltramari (2014), ressalta que a presença de sobras de comida está primordialmente ligada à ausência de planejamento adequado durante a produção de alimentos. Para evitar esse desperdício, é crucial adotar boas práticas

de fabricação em todas as etapas, desde a recepção dos alimentos até o seu preparo.

Além disso, o artigo 7º da Lei nº 9.605, de 1998, foi por muito tempo um obstáculo no combate ao desperdício de alimentos, pois impedia a doação de sobras de comida. Esse decreto atribui ao fornecedor a responsabilidade por qualquer eventualidade relacionada aos alimentos doados, o que cria uma insegurança jurídica e desencoraja as instituições e empresas a realizarem doações de alimentos excedentes (BRASIL, b).

No ano de 2020 foi aprovada a Lei nº 14.016, mais conhecida como Lei de Doação de Alimentos. Essa lei estabelece diretrizes e procedimentos para facilitar a doação de alimentos, visando reduzir a insegurança jurídica anteriormente imposta pela Lei nº 9.605, de 1998 (Brasil, 2020).

3.2 GESTÃO DA ASSISTÊNCIA ESTUDANTIL E REFEITÓRIO DAS INSTITUIÇÕES DE ENSINO

Liderar instituições empresariais representa um desafio contemporâneo, enquanto liderar universidades se configura como uma proposta consideravelmente mais exigente. Isso se evidencia diante das peculiaridades inerentes a essa instituição, bem como das múltiplas facetas que a permeiam (MARTENS, 2015). As instituições de ensino enfrentam o desafio da evasão escolar e da repetência devido à baixa condição financeira dos alunos, a falta de recursos financeiros muitas vezes impede que os estudantes tenham acesso a materiais didáticos adequados, transporte e alimentação adequada, o que pode afetar diretamente seu desempenho acadêmico (CERATTI, 2008).

Além disso, a necessidade de trabalhar para complementar a renda familiar pode levar à ausência frequente nas aulas, dificultando o acompanhamento das matérias e resultando em baixo rendimento escolar (CERATTI, 2008). Para enfrentar esse desafio, foi necessário implementar o Plano Nacional de Assistência Estudantil (PNAES), oferecendo bolsas de estudo, auxílio transporte e alimentação, além de investir em políticas educacionais inclusivas que garantam igualdade de oportunidades para todos os alunos, independentemente de sua condição financeira (GOMES; PASSOS, 2018).

O Programa Nacional de Assistência Estudantil (PNAES) desempenha um papel fundamental no apoio aos estudantes de instituições de ensino superior, especialmente no que se refere à alimentação (OLIVEIRA, 2017). Ao proporcionar acesso

a refeições balanceadas e de qualidade, o programa contribui diretamente para a melhoria do desempenho dos estudantes como evidenciado no trabalho de Oliveira e Freitas (2017), uma vez que uma alimentação adequada está diretamente relacionada ao bem-estar físico, mental e emocional.

3.3 ABORDAGEM DE COMBATE AO DESPERDÍCIO NO IFPB CAMPUS CAJAZEIRAS

Após várias consultas, por meio de entrevistas, ao setor da CAEST, verificou-se que o Campus Cajazeiras adota exclusivamente a abordagem de verificação de frequência no restaurante estudantil para combater o desperdício de alimentos. A abordagem inicia com a CAEST encaminhando a lista dos alunos regularmente cadastrados ao servidor responsável pela cozinha. O servidor fica responsável pela checagem manual do nome do aluno, a fim de permitir o acesso ao refeitório e registrar sua presença na relação.

Trata-se de uma abordagem rudimentar de controle de acesso, realizada manualmente no campus Cajazeiras. Isso pode resultar em fraudes, já que alguns alunos tentam se passar por outros. O ruído causado pela grande quantidade de alunos na fila pode gerar incertezas ao registrar a presença. Muitos discentes também não têm acesso ao refeitório a tempo de retornar às aulas no horário, o que leva alguns a perderem as refeições para assistir às aulas, gerando desperdício de comida.

Essa abordagem também afeta a elaboração do planejamento das estimativas de refeições diárias no RE do Campus Cajazeiras. O servidor responsável não tem uma estimativa média precisa de quantas refeições devem ser produzidas a cada dia. Embora os profissionais mais experientes do refeitório façam estimativas baseadas na intuição, que, por vezes, estão corretas devido à sua longa permanência no local, essas previsões contribuem, de certa forma, para a redução do desperdício de alimentos no campus.

Embora os profissionais se esforcem para minimizar o desperdício de comida, devido à normas internas de higiene e normativas federais, o instituto fica impossibilitado de realizar um aproveitamento consciente das sobras, sendo obrigado a descartá-las diretamente no lixo por ainda seguir as restrições impostas pela Lei nº 9.605, de 1998.

3.3.1 GESTÃO DAS REFEIÇÕES DO RESTAURANTE ESTUDANTIL DO IFPB - CAMPUS CAJAZEIRAS

Como mencionado anteriormente, foram realizadas entrevistas que confirmaram que o setor da CAEST é responsável pela coordenação da gestão de acesso e consumo das refeições do RE do IFPB - Campus Cajazeiras.

Essa gestão, conforme ilustrado na Figura 1 abaixo, começa com a elaboração de um formulário na plataforma *Google Forms*. O formulário tem como objetivo selecionar previamente os discentes interessados nas refeições, contendo informações como os dias em que o aluno ficará no contraturno e se ele participa do Programa de Apoio à Permanência do Estudante (PAPE), um programa do campus que oferece auxílios monetários a alunos em situação de vulnerabilidade social, ou apresenta o Índice de Vulnerabilidade Social (IVS), que avalia o grau de vulnerabilidade do aluno. Após o período de inscrições, é gerada uma lista de acesso provisória.

Figura 1 – Fluxograma de controle do Refeitório Estudantil do IFPB Campus Cajazeira



Fonte: Elaborado pelo autor (2023)

Em seguida, a CAEST aplica os processos editoriais para obtenção do IVS e PAPE para os novos alunos do semestre. Devido à quantidade limitada de refeições diárias, somente uma certa quantidade de discentes podem ter acesso ao consumo

das refeições do RE, portanto, após os processos citados acima, a CAEST elabora um processo seletivo que vai utilizar o IVS e outros critérios para selecionar os discentes para a lista final, que vai ser armazenada no Google Sheets como visto no fim da Figura 1.

Um dos principais problemas no uso do refeitório estudantil do IFPB Campus Cajazeiras é ele ser controlado manualmente por meio de listas de almoço e janta. Uma profissional do refeitório é designada, no horário de abertura do refeitório estudantil, para controlar o acesso ao refeitório, os alunos se dirigem a porta e dizem seu nome e qual modalidade está cursando, com isso a profissional verifica se o aluno está naquela lista da sua modalidade e marca a presença do aluno manualmente no papel.

Este controle de acesso e presença feitos de forma totalmente manual, serve para controlar o histórico de presenças de cada aluno durante o mês, portanto fazer o controle de quantos dias um aluno específico foi em um mês, requer considerável esforço por ser necessário averiguar manualmente todas as listas do mês onde o aluno está inserido e verificar se o mesmo compareceu em cada dia, dificultando bastante o controle de alunos que tenham baixo comparecimento. O sistema proposto pretende acabar com os processos manuais do restaurante estudantil citados acima, automatizando o controle de acesso e consumo.

Além disso, o cancelamento do acesso dos discentes no sistema vai ocorrer em duas ocasiões. A primeira delas é o aluno ter uma frequência baixa de uso do restaurante, fazendo-o assim perder o direito à sua vaga e à passando a outro aluno que aguarda na lista de espera. A segunda ocasião ocorre a cada começo de período onde todas as listas são refeitas integralmente por meio de todo o processo explicado acima.

3.4 PADRÕES E ARQUITETURAS DE DESENVOLVIMENTO

As arquiteturas de desenvolvimento de *software* referem-se à estrutura geral e organização de um sistema de *software* (MEDVIDOVIC; TAYLOR, 2010). Fornecem diretrizes e padrões para o projeto, construção e manutenção de aplicações, visando aprimorar a eficiência, escalabilidade, manutenibilidade e reusabilidade do *software* (MEDVIDOVIC; TAYLOR, 2010).

Existem diversas arquiteturas de desenvolvimento de *software* (RICHARDS, 2015). Alguns exemplos notáveis incluem a arquitetura em camadas (*Layered architecture*), que separa o sistema em camadas hierárquicas; a arquitetura orientada a eventos (*Event-driven architecture*), que se baseia em eventos e suas respectivas

reações; a arquitetura *microkernel* (*Microkernel architecture*), que busca manter o núcleo do sistema mínimo e extensível; a arquitetura de microsserviços (*Microservices architecture*), que divide o sistema em componentes menores e independentes; e a arquitetura baseada em espaço (*Space-based architecture*), que utiliza a abstração de espaço para comunicação entre componentes (RICHARDS, 2015).

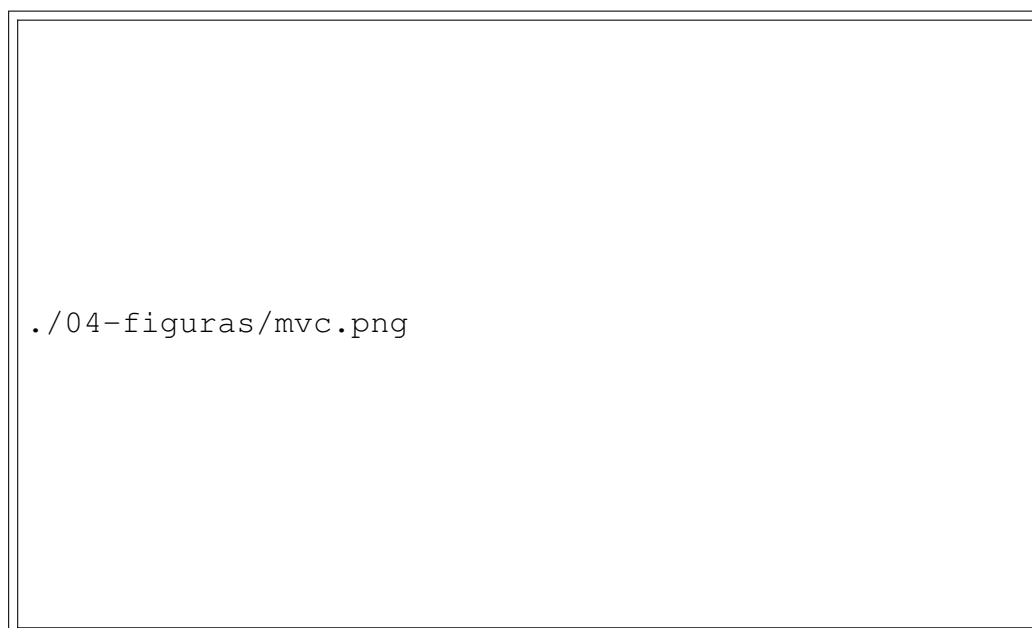
Nas próximas seções, serão detalhadas duas arquiteturas específicas: o padrão arquitetural *Model-View-Controller* (MVC) e a arquitetura limpa (*Clean Architecture*). Essas arquiteturas foram escolhidas com base na experiência prévia do autor, que possui conhecimento e prática em sua implementação.

3.4.1 PADRÃO ARQUITETURAL MVC

O padrão arquitetural MVC é uma abordagem para organizar e separar as diferentes partes de um aplicativo em três componentes principais: o Modelo (Model), a Visualização (View) e o Controlador (Controller) (LE WAGON, 2020). A Figura 2 ilustra uma representação visual do padrão MVC, destacando a interação entre os componentes.

O usuário interage com a *View*, geralmente por meio de um navegador web, realizando ações como cliques ou preenchimento de formulários. Essas ações são capturadas pela *View*, que as envia ao *Controller*. O *Controller*, por sua vez, processa a solicitação, interage com o *Model* para buscar ou atualizar dados, e então repassa os dados atualizados à *View*. A *View*, por sua vez, exibe as informações para o usuário. Os dados podem ser retornados em formatos como *JSON* ou *XML*, especialmente em aplicações web.

Figura 2 – Padrão MVC



Fonte: Elaborado pelo autor (2023)

A seguir, os componentes do padrão MVC serão detalhados para explicar suas responsabilidades e como cada um contribui para a arquitetura do sistema.

O modelo (*Model*) representa a estrutura de dados e a lógica de negócios da aplicação, sendo responsável pela manipulação, validação e armazenamento dos dados. Segundo Qureshi e Sabir (2013), o *Model* lida com a lógica interna da aplicação, sem se preocupar com a interface do usuário.

A visualização (*View*) é responsável pela apresentação dos dados ao usuário, exibindo a interface gráfica ou a representação visual dos dados. Ela interage com o usuário e atualiza a exibição com base nas mudanças feitas no *Model*. Como afirma Qureshi e Sabir (2013), a *View* trabalha em harmonia com o *Model*, refletindo seu estado atual.

O controlador (*Controller*) atua como intermediário entre o *Model* e a *View*. Ele recebe as solicitações do usuário, processa as ações e atualiza tanto o *Model* quanto a *View* conforme necessário (QURESHI; SABIR, 2014).

O padrão MVC foi escolhido para a estruturação do projeto *web* e *mobile* devido à separação de responsabilidades que facilita o desenvolvimento e a manutenção

do aplicativo. Essa abordagem permite que cada componente seja alterado independentemente, promovendo a reutilização de código, escalabilidade e flexibilidade no sistema.

3.4.2 ARQUITETURA LIMPA

A *Clean Architecture*, também conhecida como Arquitetura Limpa, é um padrão arquitetural proposto por Robert C. Martin, mais conhecido como *Uncle Bob*, renomado engenheiro de *software*, autor e palestrante, reconhecido por suas contribuições para o desenvolvimento de boas práticas e padrões de arquitetura de *software* (SANTOS, 2022).

Conforme observado por Martin (2012), ao longo dos anos, surgem várias arquiteturas de sistemas e todas compartilham o objetivo comum de separar conceitos. Essas Arquiteturas alcançam essa meta ao dividir o *software* em camadas, geralmente possuindo uma última camada para as regras de negócios e outra para interfaces de usuário (SANTOS, 2022).

Na arquitetura limpa, os componentes do *software* são organizados em camadas concêntricas, onde cada camada possui uma responsabilidade específica como pode ser visto na Figura 3. Conforme observado por Vasconcelos (2022), na parte mais interna, encontram-se as camadas de políticas, onde temos as regras de negócio da empresa, que são políticas que determinam como a empresa opera e toma decisões, e as regras da aplicação que são diretrizes que ditam como a aplicação deve funcionar em termos de processamento de dados, validações, etc.

Nas terceira e quarta camadas, encontram-se os adaptadores de interface, componentes responsáveis pela integração entre diferentes sistemas. Por fim, os *Frameworks* e *drivers* englobam conjuntos de ferramentas e bibliotecas que fornecem a estrutura necessária para o desenvolvimento do *software*.

Figura 3 – Funcionamento da Arquitetura Limpa

Fonte: DEV Community, Rubem Vasconcelos, 2022

A *Clean Architecture* foi escolhida para o desenvolvimento por ter como objetivo principal separar as preocupações e responsabilidades dentro de um sistema, promovendo a modularidade, a testabilidade e a manutenção facilitada. Ela propõe uma abordagem que coloca o foco no domínio do problema e nas regras de negócio, mantendo essas partes centrais independentes de detalhes de implementação e *frameworks* externos.

3.5 SISTEMAS DE AUTENTICAÇÃO BIOMÉTRICOS

Os sistemas de autenticação biométrica são tecnologias que utilizam características únicas e distintivas dos indivíduos, como por exemplo, impressão digital, reconhecimento facial, íris ou voz, para verificar e autenticar sua identidade. A finalidade desses sistemas é garantir a segurança e a precisão no processo de identificação de pessoas, substituindo ou complementando métodos tradicionais, como senhas, cartões de identificação ou chaves (MAGALHÃES; SANTOS, 2003).

Existem alguns métodos de autenticação biométrica que são mais populares como o RFID (*Radio Frequency Identification*) que é uma tecnologia que permite

a identificação e o rastreamento de objetos usando ondas de rádio (MONTEIRO PACHECO, 2010). O Reconhecimento Facial (*Facial Recognition*) é uma tecnologia que permite identificar e verificar a identidade de uma pessoa por meio da análise de suas características faciais (BRAGA et al., 2013). E a biometria por impressão digital que verifica dobras da pele encontradas nos dedos das mãos e dos pés para autenticar um usuário (MAZI; JÚNIOR, 2008).

Na próxima subseção será abordada a biometria por impressão digital, suas características e funcionamentos específicos. Este foi o tipo de sistema de autenticação biométrica escolhido para o uso na aplicação por principalmente possuir uma alta segurança contra fraudes, dentre outros motivos que serão explorados no decorrer da subseção.

3.6 BIOMETRIA DE IMPRESSÃO DIGITAL (*FINGERPRINT*)

As impressões digitais (ID) são características únicas e distintas presentes na pele dos dedos humanos. Essas impressões são formadas pelas papilas dérmicas, que são elevações encontradas na camada mais profunda da pele, conhecida como derme (MAZI; JÚNIOR, 2008).

Essas papilas são responsáveis por criar padrões específicos de cristas e sulcos na pele, que só se tornam visíveis através da camada superficial e transparente chamada epiderme (MAZI; JÚNIOR, 2008). Segundo Mazi e Pino. 2009, esses padrões únicos de cristas e sulcos nas impressões digitais são determinados geneticamente e permanecem inalterados ao longo da vida de uma pessoa. Devido sua singularidade e estabilidade, as impressões digitais (ID) são amplamente utilizadas para identificação e autenticação biométrica, sendo consideradas uma forma confiável e segura de reconhecimento individual.

Figura 4 – Alguns tipos de pontos característicos

Fonte: Identificação biométrica através da impressão digital usando redes neurais artificiais, Renan Corio Mazi e Arnaldo Dal Pino Júnior, 2009

Os padrões de cristas e sulcos possuem variações de suas cristas que ajudam na identificação da pessoa. Esses padrões podem ser observados na formação das cristas, onde podem ocorrer alguns tipos de cristas como, cruzamentos, bifurcações, esporas, etc. Como é ilustrado pela Figura 4, e essas variações são chamadas de minúcias ou pontos característicos (MAZI; JÚNIOR, 2008).

A captação da impressão digital envolve o uso de um dispositivo, como um *scanner* de impressão digital, para registrar os padrões únicos das cristas e sulcos, presentes na pele dos dedos de uma pessoa, e convertê-los em informações digitais (ZAPAROLI, 2002). Em relação ao armazenamento, os dados biométricos da impressão digital passam por um tratamento de criptografia antes de serem armazenados em um banco de dados (ZAPAROLI, 2002). Quando uma pessoa precisa ser autenticada, os dados biométricos captados dessa pessoa são comparados com os dados biométricos previamente cadastrados no banco de dados, utilizando algoritmos que buscam correspondências entre os padrões da impressão digital capturada e a impressão digital armazenada (MAZI; JÚNIOR, 2008).

Além disso, este tipo de sistema de autenticação biométrica possui algumas bibliotecas de fácil implementação como as bibliotecas da empresa *Fingerprint*¹, que disponibiliza vários SDK (*Software Development Kit*) para utilização de sensores de impressão biométrica.

¹ <https://fingerprint.com/sdk-libraries/>

Um exemplo de biblioteca a ser citada seria a *Javascript Agent*² da *Fingerprint* que atua do lado do cliente por meio de um navegador *web* e coleta dados de sensores e do navegador e os manda à API para ser feito o processamento deles.

Em relação aos sensores utilizados para a leitura das impressões digitais (ID), há poucas marcas que possuem seus sensores devidamente homologados, tais como a Suprema e a Nitgen³. Exemplificando, a Suprema oferece sensores capazes de identificar até mesmo se o dedo verificado é uma prótese ou um dedo real, como o Sensor Suprema LFD. No entanto, devido à incorporação dessa tecnologia avançada, o custo desse sensor é elevado. Uma alternativa mais acessível seria o Sensor de Led Verde, embora seja importante mencionar que sua precisão não é tão alta, pois trata-se de um sensor mais básico.

Neste projeto, optou-se pela utilização da biometria de impressão digital, não apenas pelo seu alto nível de segurança, mas também porque sua implementação é mais simples, considerando a experiência técnica adquirida pelo autor deste trabalho.

² <https://dev.fingerprint.com/docs/js-agent>

³ <https://simpax.com.br/qual-a-diferenca-entre-os-sensores-biometricos-de-impressao-digital-por-que-existem-3-opcoes-de-modelos/>

4 TRABALHOS RELACIONADOS

Foi realizada uma pesquisa inicial na literatura sobre sistemas destinados a auxiliar no gerenciamento do acesso e consumo das refeições do RE em Instituições de Ensino. A partir dessa pesquisa, foram identificados trabalhos que desenvolveram sistemas com esse objetivo. Em seguida, fez-se uma seleção dos estudos mais relevantes para o trabalho proposto, com base nos seguintes critérios de busca:

- Trabalhos científicos de origem brasileira no período de 2018 a 2023;
- Implementações de aplicações de biometria, *mobile* ou *web*, que atendem o objetivo proposto.

A partir desses critérios, foram selecionados os seguintes trabalhos: Sistema de Automação de Restaurantes Universitários (SISRU) de Duarte e Silva. (2020), o Sistema para Modernização do Restaurante do IF Sudeste MG - Campus Juiz de Fora de França e Moraes (2021) e o Sistema de Acesso ao Restaurante Institucional (SARI) do Instituto Federal do Piauí - Campus Floriano que teve experiência relatada por Lopes e Lima (2019).

4.1 SISTEMA DE AUTOMAÇÃO DE RESTAURANTES UNIVERSITÁRIOS (SISRU)

É um sistema desenvolvido para automatizar o processo de pagamento e liberação de refeições do restaurante universitário da Universidade Federal Rural do Semi-Árido (UFERSA) - Campus Pau dos Ferros (DUARTE; SILVA, 2020). O SISRU se divide em dois subsistemas, um subsistema *WEB* que disponibiliza uma API que permite a autorização de refeições, realização de depósito em conta virtual, acompanhamento das últimas refeições solicitadas, acompanhamento dos últimos depósitos efetuados e dá a possibilidade de gerar relatórios (ver Figura 5).

Figura 5 – Tela principal do módulo administrativo do SISRU



Fonte: Sistema de automação de refeitórios universitários - SISRU, DUARTE E SILVA, 2020.

O outro subsistema é um embarcado por meio de um arduino que faz a leitura dos cartões RFID com um leitor conectado e realiza a comunicação com a API por um módulo *Ethernet* (DUARTE; SILVA, 2020).

4.2 SISTEMA PARA MODERNIZAÇÃO DO RESTAURANTE DO IF SUDESTE MG - CAMPUS JUIZ DE FORA. (SRE)

É um sistema *WEB* desenvolvido para a Diretoria de Desenvolvimento Institucional (DDI) do campus Juiz de Fora no estado de Minas Gerais, que controla a comercialização do *ticket* rápido, criando uma carteira virtual para cada discente e disponibilizando para a gestão o fluxo de pessoas e venda de *tickets* do refeitório em tempo real (FRANÇA; MORAES, 2022).

Foi desenvolvido utilizando *Hypertext Markup Language* (HTML), *Cascading Style Sheets* (CSS), *Javascript* e o *framework Bootstrap* em seu *Front-End* (FRANÇA; MORAES, 2022). No *Back-End* foi utilizada a linguagem PHP com o *framework CodeIgniter* e no banco de dados utilizaram MySQL (FRANÇA; MORAES, 2022).

Figura 6 – Tela de vendas filtrada por usuário e por período de tempo e tela de venda de ticket filtrada por período de tempo



Fonte: Um Sistema para Modernização do Restaurante do IF Sudeste MG - Campus Juiz de Fora, FRANÇA E MORAES, 2021.

Na Figura 6 ilustra as funcionalidades do sistema que permitem verificar os dados de um usuário selecionado, e abaixo podemos observar um relatório de *tickets* vendidos pelo caixa ou fornecidos pela assistência.

4.3 SISTEMA DE ACESSO AO RESTAURANTE INSTITUCIONAL (SARI)

O sistema do restaurante institucional do campus Floriano controla o acesso dos alunos ao Restaurante Institucional (RI) do campus, proporcionando a possibilidade dos discentes realizarem reservas das refeições de forma gratuita e dos servidores efetuarem compra das refeições no RI (LOPES; LIMA, 2019).

O SARI foi desenvolvido para a *web* com responsividade para funcionar tanto em telas de *smartphones* quanto em computadores e *notebooks* (LOPES; LIMA, 2019). No campus Floriano o sistema foi implementado, em um terminal de acesso, para ler o código de barras que é impresso na carteira de estudante de cada discente (LOPES; LIMA, 2019).

Figura 7 – Tela para reserva ou compra de refeições e tela para cadastrar o cardápio



Fonte: O Sistema de Acesso ao Restaurante Institucional do Instituto Federal do Piauí - Campus Floriano, LOPES E LIMA, 2018.

Na Figura 7 podemos observar duas telas do sistemas, sendo a primeira responsável por permitir a aquisição ou reserva das refeições, mostrando o cardápio,

o horário de funcionamento do refeitório e a quantidade de *tickets* disponíveis, e a segunda mostrando o cadastro do cardápio (LOPES; LIMA, 2019).

4.4 ANÁLISE COMPARATIVA

Com a finalidade de encontrar funcionalidades que representam os requisitos implementados e destacar as especificidades de cada sistema, foi realizada uma análise com os projetos estudados anteriormente, comparando as características dos mesmos em relação ao sistema que está sendo proposto neste trabalho.

No Quadro 1, são apresentados os principais aspectos do serviço de gerenciamento de acesso ao RE, destacando as tecnologias empregadas por cada sistema, as funcionalidades oferecidas e os diferentes tipos de relatórios gerados.

Quadro 1 – Comparação das características dos trabalhos relacionados



Fonte: Elaborado pelo autor (2023)

Entre os pontos positivos dos sistemas analisados, destacam-se:

- **SISRU:** O SISRU possui uma interface intuitiva que facilita a reserva antecipada das refeições. Além disso, integra funcionalidades de notificação e atualizações

em tempo real, contribuindo para a eficiência no controle do consumo e na gestão dos registros, o que simplifica o acompanhamento dos alunos.

- **SRE:** O SRE diferencia-se por separar as reservas em listas distintas para alunos e servidores. Essa segmentação possibilita uma gestão mais precisa dos acessos, facilitando a alocação de recursos e a identificação das demandas específicas de cada grupo, o que é um diferencial importante na organização do sistema.
- **SARI:** O SARI se destaca por oferecer uma solução enxuta e altamente eficiente, concentrando-se na reserva ou compra antecipada de refeições e na geração de relatórios gerais. Sua abordagem simples e robusta permite uma implementação rápida, fácil manutenção e integração com outros sistemas, garantindo confiabilidade e precisão na gestão do acesso ao refeitório.

No caso do *UNIACCESS*, o sistema integra de forma inovadora as funcionalidades presentes nos trabalhos relacionados. Além de permitir a reserva das refeições por meio do serviço *mobile*, o *UNIACCESS* incorpora um leitor biométrico de impressão digital, que eleva significativamente a segurança contra fraudes.

O sistema possibilita a geração de relatórios individualizados para cada usuário, promovendo um acompanhamento detalhado e personalizado, e conta com um Serviço de Notificação que assegura a comunicação em tempo real.

Outro diferencial importante é a sua interface simples e intuitiva, cuidadosamente projetada para maximizar a usabilidade e facilitar a interação dos usuários, independentemente do nível de familiaridade com tecnologias digitais.

Essas características ressaltam que, embora os sistemas SISRU, SRE e SARI apresentem limitações – com o SISRU e o SARI operando com uma única lista de controle e o SRE oferecendo uma segmentação específica para reservas – cada um possui pontos positivos que atendem a contextos particulares.

Em contrapartida, o *UNIACCESS* unifica essas funcionalidades com inovações que aprimoram tanto a segurança quanto a usabilidade, consolidando-se como uma solução mais abrangente e eficaz para o controle de acesso e o gerenciamento do consumo em Restaurantes Estudantis.

5 UNIACCESS

UNIACCESS é uma aplicação contendo serviços com o propósito de auxiliar o gerenciamento de acesso e consumo de refeições no RE do IFPB - Campus Cajazeiras. Tendo como objetivo principal a automatização de gerenciamento dos registros de discentes no setor da CAEST. Além da redução do desperdício dos alimentos produzidos pelo RE por meio de um serviço de acesso mais preciso.

A aplicação conta com serviço *Mobile* para uso exclusivo dos discentes, em que é possível ver os dados do discente e modificá-los. E sua principal funcionalidade é disponibilizar para o discente a funcionalidade de confirmação de utilização da refeição do dia atual no refeitório. Caso o discente não realize essa confirmação, o sistema notificará a CAEST. Então, a CAEST, por meio do serviço *web*, irá remanejar as vagas não confirmadas para outros discentes que necessitem o acesso ao restaurante.

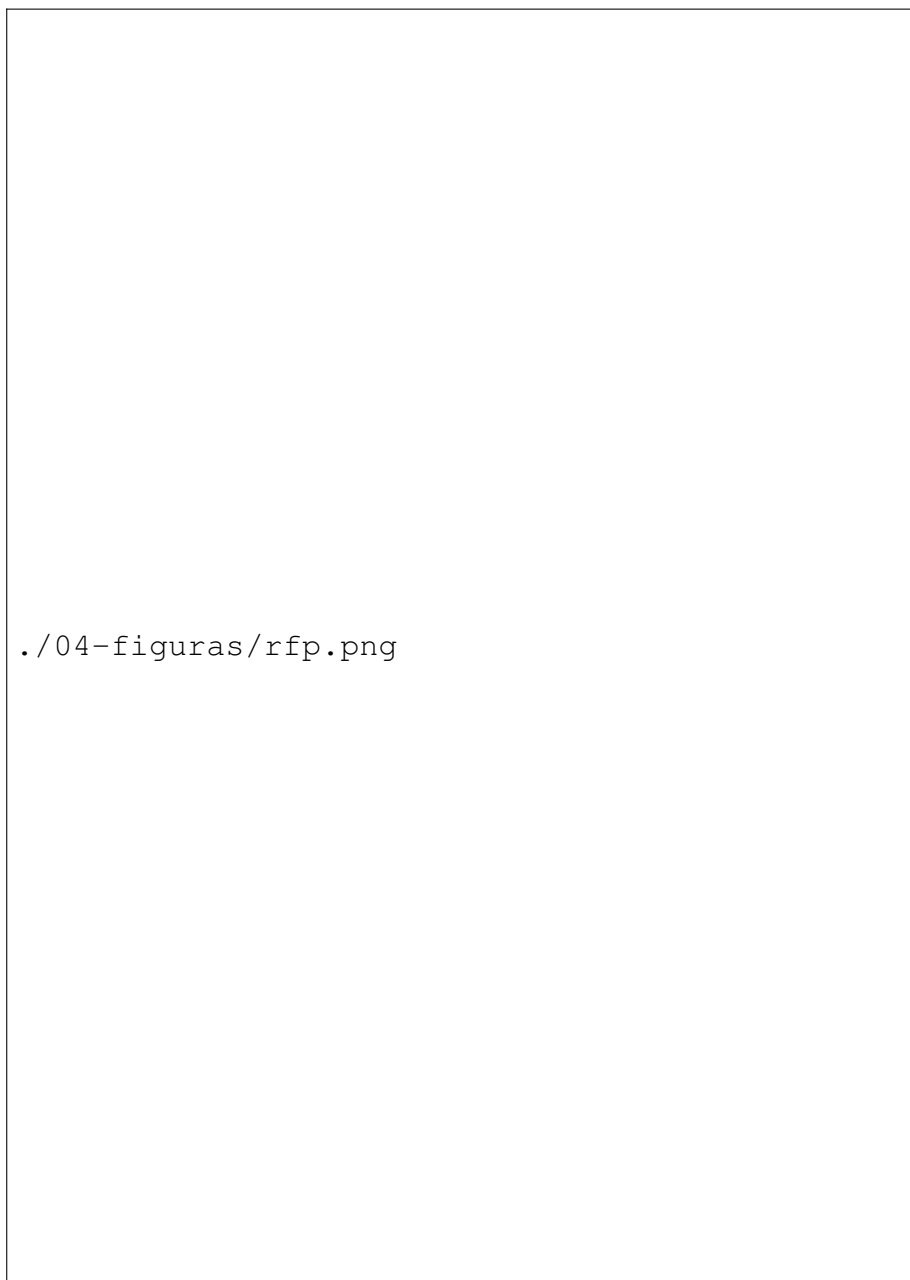
O serviço *web* da aplicação *UNIACCESS* tem como propósito coordenar o registro dos discentes e controle do acesso às refeições do RE. Por meio deste serviço é possível cadastrar os discentes na lista de alunos contemplados com a refeição do dia da semana. Além disso, verificar os dados de cada discente individualmente, relatórios do uso diário, mensal e semestral do RE que os discentes frequentaram.

Seguindo as etapas do processo de desenvolvimento de *software* (MONITORA TEAM, 2021), nas seções seguintes, serão detalhadas as etapas de levantamentos dos requisitos, análise dos requisitos, bem como a etapa do projeto do sistema. Detalhando a arquitetura do sistema com suas tecnologias que serão abordadas no trabalho, além do projeto do banco de dados.

5.1 ELICITAÇÃO DOS REQUISITOS

Para ajudar nos processos internos da CAEST, foi pensado o desenvolvimento do *UNIACCESS* que automatiza o gerenciamento e controle das relações diárias de alunos e disponibilizará métricas de frequência ao RE e quantidade de alunos diários, para tomada de decisão estratégica do setor.

Para a elicitação dos requisitos, foi realizada uma reunião com a CAEST, pesquisas bibliográficas e estudos de trabalhos relacionados ao tema. Essa etapa foi essencial para a definição clara e precisa dos requisitos do sistema, apresentados no Quadro 2 e detalhados no Apêndice A.

Quadro 2 – Requisitos Funcionais

Fonte: Elaborado pelo autor (2023)

Na Figura 8, podemos observar os casos de uso do sistema, que envolve quatro atores. Um desses atores é o Aluno, que tem acesso a três funcionalidades do serviço *mobile*, sendo a principal delas a confirmação da pretensão diária, além da possibilidade de visualizar e atualizar seus dados cadastrados.

No serviço *web*, que é o principal, os servidores da CAEST e do Restaurante podem acessar diversas funcionalidades, sendo as principais, a autorização do aluno por parte do servidor CAEST e a visualização de autorização que se dá pelo servidor do Restaurante após o serviço *web* receber a confirmação do leitor biométrico de impressão digital.

Figura 8 – Casos de uso



Fonte: Elaborado pelo autor (2023)

5.2 PROJETO ARQUITETURAL DO SISTEMA

O *UNIACCESS* segue a estruturação proposta pela *Clean Architecture* no *back-end* em conjunto com o padrão arquitetural MVC. A adoção da estrutura da Arquitetura Limpa promove uma definição mais clara e precisa das responsabilidades de cada componente do sistema por meio do estabelecimento de níveis de abstração, tornando simples a manutenção do projeto a longo prazo.

O MVC proporciona uma estrutura organizada e modular ao sistema, facilitando ainda mais sua manutenção, reutilização de componentes, testabilidade e escalabilidade. A Figura 9 ilustra a solução da infraestrutura proposta pelo *UNIACCESS*, onde pode se observar a interação entre cada componente do sistema.

Figura 9 – Arquitetura do sistema



Fonte: Elaborado pelo autor (2023)

Ao analisar a Figura 9, pode-se identificar os elementos que compõem a arquitetura utilizada no desenvolvimento do *UNIACCESS*. A aplicação possui dois *Front-End's*: um para a versão *Mobile*, que foi desenvolvida utilizando *React Native* (EISENMAN, 2015) e é exclusiva para o uso dos discentes.

A versão *WEB* utiliza o framework *ReactJS* (BANKS; PORCELLO, 2017) e é utilizada pelos servidores da CAEST e do Restaurante Estudantil. O acesso ao sistema *WEB* e a troca de dados com o *Back-End* requerem a utilização de um navegador *web*.

A comunicação entre o *Front-End* e o *Back-End* é realizada por meio de requisições HTTP (GOURLEY et al., 2002) (*Hypertext Transfer Protocol*), utilizando JSON (MARRS, 2013) (*Javascript Object Notation*) como formato padrão para troca de dados. Ao chegar no *Back-End*, os dados são recebidos pelo *framework Express* (HAHN, 2016) por meio do objeto *Request* e são processados utilizando a linguagem *Javascript* (FLANAGAN, 2020) no lado do servidor, fazendo uso do *NodeJS* (CANTELON et al., 2013).

Por fim, o *UNIACCESS* utiliza o banco de dados *MongoDB* (CHODOROW et al., 2019) por ser não relacional e possuir como característica uma estrutura de armazenamento altamente adaptável e escalável, permitindo que os dados sejam manipulados de maneira flexível e eficiente, sem a necessidade de definir esquemas ou tabelas rígidas como no caso de bancos de dados relacionais. O detalhamento do modelo conceitual adotado pelo sistema pode ser visto no Apêndice A.

5.2.1 DETALHAMENTO DA SOLUÇÃO ARQUITETURAL

A Figura 10 ilustra a estrutura e o funcionamento do *UNIACCESS*, seguindo os princípios da *Clean Architecture*. Esse modelo organiza as responsabilidades do sistema de forma clara, permitindo uma melhor visualização do fluxo de dados e facilitando tanto a manutenção quanto a escalabilidade da aplicação.

Figura 10 – Arquitetura do sistema



Fonte: Elaborado pelo autor (2023)

A Figura 10 permite visualizar com clareza o trajeto dos dados dentro da camada de *Back-End*, evidenciando o fluxo no contexto da *Clean Architecture*. Essa abordagem organizou os serviços da aplicação em camadas, garantindo a separação de responsabilidades, como a distinção entre a lógica de negócios, a apresentação e o controle de fluxo. O fluxo de dados segue a seguinte estrutura:

1. O usuário realiza uma requisição HTTP para os serviços do UniAccess a partir de uma aplicação web ou mobile que esteja integrada à API (Camada externa de Visualização).
2. A UNIACCESS API recebe a solicitação e a direciona ao controlador correspondente (interface *IController*). Os controladores são responsáveis por disponibilizar *endpoints* para serviços específicos, como a geração de relatórios, além de aplicar validações por meio de DTOs¹. Em seguida, os controladores encaminham os dados processados para a camada de serviços (interface *IService*).
3. A camada de serviços (Regras de Negócio) processa as informações recebidas dos controladores, aplicando as regras necessárias. Essa camada pode utilizar entidades da Camada de Domínios para construir DTOs. Após esse processamento, os dados são enviados ao repositório (interface *IRepository*), de acordo com o método HTTP correspondente à solicitação.
4. A camada de repositório executa as operações de CRUD, conforme definido pela camada de serviços, estabelecendo a conexão direta com o banco de dados por meio do *Repository Mongoose* para realizar as operações necessárias.
5. Por fim, uma resposta é gerada e percorre todas as camadas até retornar ao controlador, que a envia de volta à aplicação do usuário no formato JSON.

5.3 IMPLEMENTAÇÃO

O projeto foi desenvolvido utilizando uma arquitetura *API RESTful*, garantindo a comunicação entre o *Back-End*, o *Front-End Web* e o aplicativo *Mobile*.

Para o *Back-End*, foi utilizado *Node.js*, uma plataforma baseada no motor V8 do *Google Chrome*, que permite a execução de código *JavaScript* no servidor, juntamente com o *Express.js*, um framework minimalista para criação de *APIs* eficientes e escaláveis. Além disso, optou-se pelo uso de *TypeScript*, uma linguagem que adiciona

¹ Objeto de Transferência de Dados, um padrão de projeto utilizado para transmitir informações entre diferentes partes de um sistema.

tipagem estática ao *JavaScript*, proporcionando maior segurança e organização no desenvolvimento.

Para o gerenciamento dos dados, foi utilizado o banco de dados *MongoDB*, um banco de dados *NoSQL* orientado a documentos, que oferece escalabilidade e flexibilidade na modelagem dos dados. Como Sistema de Gerenciamento de Banco de Dados (SGBD), utilizou-se o *MongoDB Atlas*, uma plataforma em nuvem que fornece um ambiente gerenciado, seguro e altamente disponível, reduzindo a complexidade da administração do banco. A comunicação entre a aplicação e o banco foi realizada por meio do *Mongoose*, uma biblioteca para *Node.js* que facilita a interação com o *MongoDB* ao fornecer uma camada de abstração baseada em esquemas, garantindo mais controle sobre a estrutura dos dados e validação das informações armazenadas.

No *Front-End Web*, utilizou-se *React.js*, uma biblioteca *JavaScript* declarativa para construção de interfaces dinâmicas e reativas. Para a estilização e estruturação da interface, foi adotado o *Material-UI*, uma biblioteca baseada no *Material Design*, que oferece componentes prontos e acessíveis para um desenvolvimento mais ágil. Assim como no *Back-End*, o *TypeScript* foi empregado para garantir melhor manutenção e escalabilidade do código.

Para o aplicativo *Mobile*, foi utilizado *React Native*, um framework que permite o desenvolvimento de aplicativos nativos para *Android* e *iOS* utilizando *JavaScript* e componentes reutilizáveis. O uso de *TypeScript* no *React Native* ajudou a melhorar a previsibilidade do código, reduzir erros em tempo de desenvolvimento e proporcionar uma melhor experiência na integração com *APIs* e bibliotecas externas.

A seguir, serão apresentadas as principais páginas das plataformas *Web* e *Android*, bem como a forma de utilização por parte do usuário. Todas as páginas e funcionalidades de ambos os sistemas, incluindo a *API RESTful*, foram devidamente implementadas e operam de maneira correta, garantindo uma experiência consistente e eficiente para os usuários.

5.3.1 PRINCIPAIS FUNCIONALIDADES

Na página de *Login*, ilustrada na Figura 11, o usuário deve preencher os campos com seu email e senha previamente cadastrados pelo administrador do sistema, o tipo do usuário define quais módulos o mesmo tem acesso, administradores tem acesso a todos os módulos, funcionários não possuem acesso ao módulo de Refeitório (Figura 14) e Controle de Usuários (Figura 25), por fim, usuários do refeitório possuem acesso apenas ao módulo de Refeitório (Figura 14).

Figura 11 – Página de Login

Fonte: Elaborado pelo autor (2025)

Ao realizar o *Login* o usuário é redirecionado para a página principal onde, por padrão, é mostrado o módulo de Relatórios, como pode ser visto nas Figuras 12 e 13.

Figura 12 – Página principal - Módulo de Relatórios

Fonte: Elaborado pelo autor (2025)

No Relatório de Presença por Aluno, mostra os alunos e quantas presenças e ausências cada um tem, podendo buscar o aluno por nome ou matrícula. No segundo relatório, de Presenças ao longo do semestre, mostra todas as listas e quantos alunos compareceram ao refeitório em cada dia da semana. Ambos os relatórios podem ser exportados nos formatos SVG, PNG e CSV clicando no menu de opções de cada relatório.

Na Figura 13 mostra o Relatório detalhado de presenças, onde é possível pesquisar as presenças pelo nome do aluno, matrícula ou pelo dia da lista.

Figura 13 – Página principal - Módulo de Relatórios



Fonte: Elaborado pelo autor (2025)

O Módulo para o refeitório foi desenvolvido levando em conta acessibilidade e simplicidade, como mostra a Figura 14. Este módulo apresenta a lista do dia com o nome completo e matrícula de cada aluno, e ao lado de cada um, botões para confirmar sua presença ou ausência ao adentrar o refeitório. Ao confirmar que o aluno está presente ou ausente, seu nome na lista muda visualmente para melhor visualização.

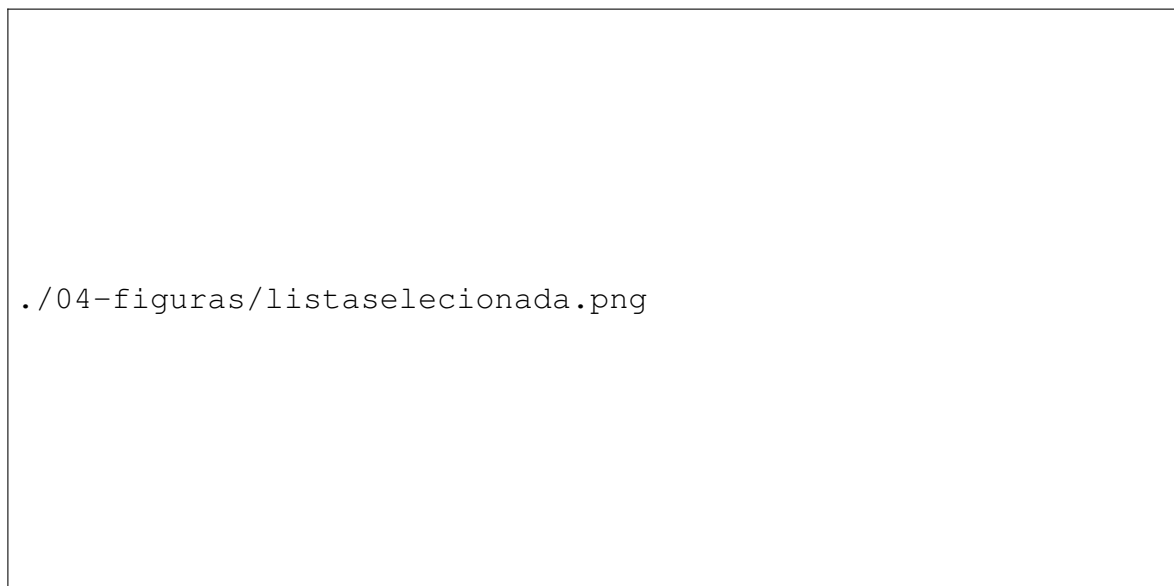
Figura 14 – Módulo do Refeitório

Fonte: Elaborado pelo autor (2025)

No módulo de controle de listas (Figura 15), é possível realizar buscas pelo dia da lista. Além disso, pode-se editá-la, incluindo ou removendo alunos sem alterar o template, e também excluí-la.

Para facilitar o controle e a gestão, foi desenvolvida uma função para gerar as listas semanais. Essas listas são criadas com base no dia da semana e utilizam os dados do template da lista correspondente, iniciando já preenchidas com as informações dos alunos e alunos extras. Essa função pode ser executada ao clicar no botão "Gerar listas da semana".

Figura 15 – Módulo Controle de Listas - Lista selecionada



Fonte: Elaborado pelo autor (2025)

Neste módulo, como apresenta a Figura 16, podemos ver os dados do aluno selecionado previamente na lista de alunos e os dias em que possui acesso autorizado ao refeitório, também é possível tanto editar os dados do aluno quanto autorizar ou desautorizar o acesso à dias específicos, poder excluir o mesmo.

Figura 16 – Módulo Gerência de Alunos - Aluno selecionado



Fonte: Elaborado pelo autor (2025)

O Aplicativo é de uso exclusivo do aluno, ao abri-lo o aluno pode realizar o *login* como indica a Figura 17 que o redirecionará para a página principal (Figura 18).

Figura 17 – Aplicativo UniAccess - Página inicial de *login*



Fonte: Elaborado pelo autor (2025)

Na página principal o aluno tem acesso aos dias autorizados em que pode

frequentar o refeitório, nos dias em que o mesmo tem acesso, é possível fazer a pré confirmação da presença como também pode ser visto na Figura 18.

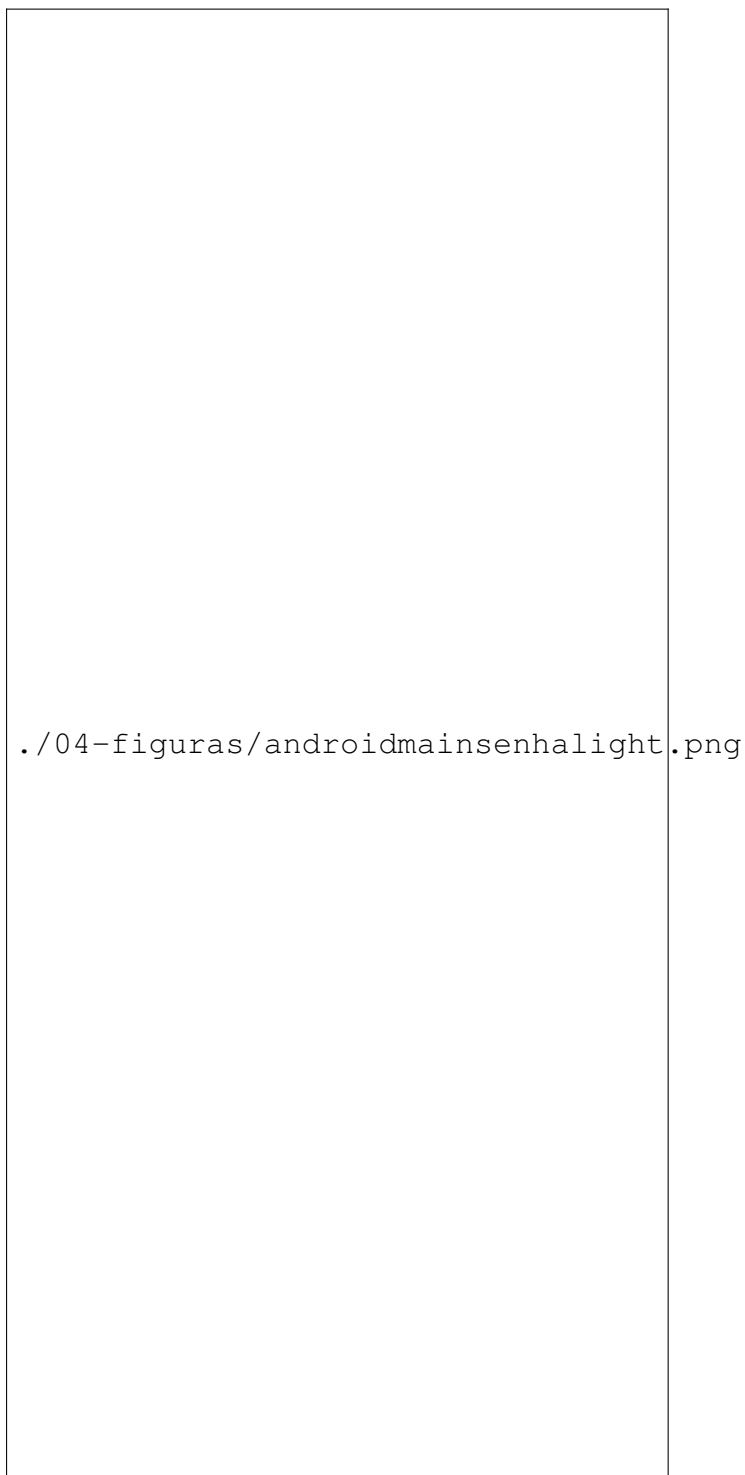
Figura 18 – Aplicativo UniAccess - Página principal



Fonte: Elaborado pelo autor (2025)

Cada aluno, ao ser cadastrado, recebe uma senha padrão correspondente à sua matrícula. No entanto, é possível alterá-la por meio do aplicativo (Figura 19). Todas as funcionalidades críticas são processadas no servidor, garantindo que o aluno não consiga burlar horários ou datas simplesmente modificando o aplicativo ou ajustando o horário do dispositivo.

Figura 19 – Aplicativo UniAccess - Modal para alterar senha



Fonte: Elaborado pelo autor (2025)

Nos dias que o aluno não precisar enviar sua pré confirmação, o aplicativo

mostrará uma mensagem avisando que o refeitório não poderá ser acesso naquele dia como pode ser visto na Figura 20.

Figura 20 – Aplicativo UniAccess - Mensagem de refeitório indisponível



./04-figuras/androidpresencanaoconfirmadalight.png

Fonte: Elaborado pelo autor (2025)

Entretanto, quando for o dia de confirmar presença, será mostrado uma mensagem de pendente e o aluno pode apertar no botão de de Confirmar, como indica a Figura 18, para enviar sua pré confirmação e o aplicativo mostrará uma confirmação e a data do dia pré confirmado como pode ser observado na Figura 21.

Figura 21 – Aplicativo UniAccess - Mensagem de presença confirmada



Fonte: Elaborado pelo autor (2025)

O aplicativo conta com modo claro e escuro para melhor legibilidade dos alunos. Estes modos e demais informações e explicações sobre cada módulo e página do sistema podem ser vistos no Apêndice B.

6 CONSIDERAÇÕES FINAIS

Com este Trabalho foi possível determinar os problemas que estão ocorrendo na gestão do Restaurante Estudantil do Instituto Federal da Paraíba - Campus Cajazeiras e propor soluções para os mesmos por meio do desenvolvimento do *UNIACCESS*. E como podemos ver no decorrer de todo o trabalho proposto, o sistema desenvolvido pode gerar um grande impacto na Coordenação de Apoio ao Estudante, assim como causar uma mudança de cultura aos servidores do Restaurante Estudantil e aos alunos contemplados, de modo a facilitar o controle de acesso e consumo das refeições e reduzir bastante o desperdício de alimentos no Campus Cajazeiras.

6.1 TRABALHOS FUTUROS

Em trabalhos futuros, o *UNIACCESS* poderá ser expandido para oferecer novas funcionalidades, como a implementação de um sistema de licenciamento, permitindo sua distribuição para outras instituições de ensino. Além disso, o sistema poderá passar a contar com mais opções de controle e relatórios, ampliando as possibilidades de análise de dados e oferecendo novas formas de exportação de resultados. Também poderá ser integrado a outras plataformas de gestão estudantil, facilitando a comunicação e o controle de acesso. Outra possibilidade é aprimorar as funcionalidades de gerenciamento de listas, com novos tipos de relatórios e comparações, oferecendo um maior leque de opções para a gestão do refeitório e o controle de desperdício de alimentos.

REFERÊNCIAS

ARQUITETURA Limpa: O conceito por trás do código. Disponível em: <<https://dev.to/rubemfsv/arquitetura-limpa-o-conceito-por-tras-do-codigo-4kdf>>. Acesso em: 09 Jul. 2023.

BANKS, A.; PORCELLO, E. **Learning React: Modern Patterns for Developing React Apps**. [S.l.]: O'Reilly Media, 2017.

BERBIGIER, M. C.; MAGALHÃES, C. R. Estado nutricional e hábito alimentar de estudantes universitários em instituição pública do Brasil. **Saúde e Pesquisa**, v. 14, n. 1, 2021.

BRAGA, L. F. Z. et al. **Sistemas de Reconhecimento Facial**. Tese (Doutorado) — UNIVERSIDADE DE SÃO PAULO, 2013.

BRASIL. Lei nº 14.016, de 23 de junho de 2020. Disponível em: <https://www.planalto.gov.br/ccivil_03/leis/l9605.htm>.

_____. Lei nº 9.605, de 12 de fevereiro de 1998. Disponível em: <https://www.planalto.gov.br/ccivil_03/leis/l9605.htm>.

_____. Resolução nº 40, de maio de 2011. Disponível em: <<https://www.ifpb.edu.br/orgaoscolegiados/consuper/resolucoes/2011/resolucao-no-40>>.

CANTELON, M.; HARTEK, M.; HOLOWAYCHUK, T.; RAJLICH, N. **Node.js in Action**. [S.l.]: Manning Publications, 2013.

CERATTI, M. R. N. Evasão escolar: causas e consequências. **Curitiba/PR**, 2008.

CHODOROW, K.; BRADSHAW, S.; BRAZIL, E. **MongoDB: The Definitive Guide**. [S.l.]: O'Reilly Media, 2019.

DELIBERADOR, L. R. Desperdício de alimentos em restaurantes: uma análise em uma instituição universitária. Universidade Federal de São Carlos, 2019.

DUARTE, F. L. C.; SILVA, V. M. L. Sistema de automação de restaurantes universitários-sisru. **Anais do Encontro de Computação do Oeste Potiguar ECOP/UFERSA (ISSN 2526-7574)**, n. 4, 2020.

DURÃO, C.; OLIVEIRA, J.; ALMEIDA, M. D. V. d. Portugal e o padrão alimentar mediterrânico. 2008.

EISENMAN, B. **Learning React Native: Building Native Mobile Apps with JavaScript**. [S.l.]: O'Reilly Media, 2015.

FIRGULHA, L. C.; COSTA, M. D. O.; MARIANO, F. D. O.; PIANZOLA, S. P. U.; CEZARIO, J. S. F. Desperdício de alimentos: Um estudo em restaurante em venda nova do imigrante-es.

FLANAGAN, D. **JavaScript: The Definitive Guide**. [S.l.]: O'Reilly Media, 2020.

FRANÇA, F. D. O. T.; MORAES, E. A. P. Um sistema para modernização do restaurante do if sudeste mg-campus juiz de fora. **Seminários de Trabalho de Conclusão de Curso do Bacharelado em Sistemas de Informação**, v. 7, n. 1, 2022.

GOMES, A. M. de O.; PASSOS, G. de O. A implementação do programa nacional de assistência estudantil (pnaes) nos institutos federais. **Revista de Políticas Públicas**, Universidade Federal do Maranhão, v. 22, n. 1, p. 416–442, 2018.

GOURLEY, D.; TOTTY, B.; SAYER, M.; AGGARWAL, A. **HTTP: The Definitive Guide**. [S.l.]: O'Reilly Media, 2002.

HAHN, E. **Express in Action: Writing, building, and testing Node.js applications**. [S.l.]: Manning Publications, 2016.

HÄRTER, A. C.; SILVA, C. E. S. da; SNEYDER, D.; SIQUEIRA, R. F. Estudo de caso sobre a satisfação dos usuários do restaurante universitário-ru. **Revista de Estudos Sociais**, Faculdade de economia, v. 15, n. 30, p. 33–42, 2013.

LOPES, O. S.; LIMA, M. V. A. O sistema de acesso ao restaurante institucional do instituto federal do piauí–campus floriano. **Livros**, p. 18–39, 2019.

MAGALHÃES, P. S. T.; SANTOS, H. D. d. Biometria e autenticação. Associação Portuguesa de Sistemas de Informação (APSI), 2003.

MARRS, T. **JSON at Work: Practical Data Integration for the Web**. [S.l.]: O'Reilly Media, 2013.

MARTENS, A. E. Os caminhos da gestão universitária. INPEAU/UFSC, 2015.

MARTIN, R. C. **The Clean Code Blog**. Disponível em: <<https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html>>. Acesso em: 09 Jul. 2023.

MAZI, R. C.; JÚNIOR, A. D. P. Identificação biométrica através da impressão digital usando redes neurais artificiais. **Anais do XIV ENCITA**, p. 19–22, 2008.

MEDVIDOVIC, N.; TAYLOR, R. N. Software architecture: foundations, theory, and practice. In: **Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 2**. [S.l.: s.n.], 2010. p. 471–472.

MONTEIRO PACHECO, L. Rfid radio frequency identification. **Universidade Federal do Rio de Janeiro**, 2010.

O que é padrão MVC Entenda arquitetura de sistemas Disponível em: <<https://blog.lewagon.com/pt-br/skills/o-que-e-padrao-mvc>>. Acesso em: 09 Jul. 2023.

OLIVEIRA. Gr-huol: Sistema web para gerenciamento do acesso ao refeitório do hospital universitário onofre lopes. 2019.

OLIVEIRA, F. Avaliação do impacto dos benefícios pnaes sobre o desempenho acadêmico: O caso da universidade federal do rio grande. **AVALIES, 2017**, 2017.

PROCESSO de desenvolvimento de software: veja os principais modelos. Disponível em: <<https://www.monitoratec.com.br/blog/processo-de-desenvolvimento-de-software/>>. Acesso em: 09 Jul. 2023.

QUAL a diferença entre os Sensores Biométricos de Impressão Digital? Por Que existem 3 opções de Modelos? Disponível em: <<https://simpax.com.br/qual-a-diferenca-entre-os-sensores-biometricos-de-impressao-digital-por-que-existem-3-opcoes-c>> Acesso em: 09 Jul. 2023.

QURESHI, M.; SABIR, F. A comparison of model view controller and model view presenter. **arXiv preprint arXiv:1408.5786**, 2014.

RICARTE, M. P. R.; FÉ, M. A. B. M.; SANTOS, I. H. V. da S.; LOPES, A. K. M. Avaliação do desperdício de alimentos em uma unidade de alimentação e nutrição institucional em fortaleza-ce. **Saber Científico (1982-792X)**, v. 1, n. 1, p. 158–175, 2021.

RICHARDS, M. **Software architecture patterns**. [S.l.: s.n.], 2015.

SANTOS. Uma introdução a robert cecil martin “uncle bob”. 2022.

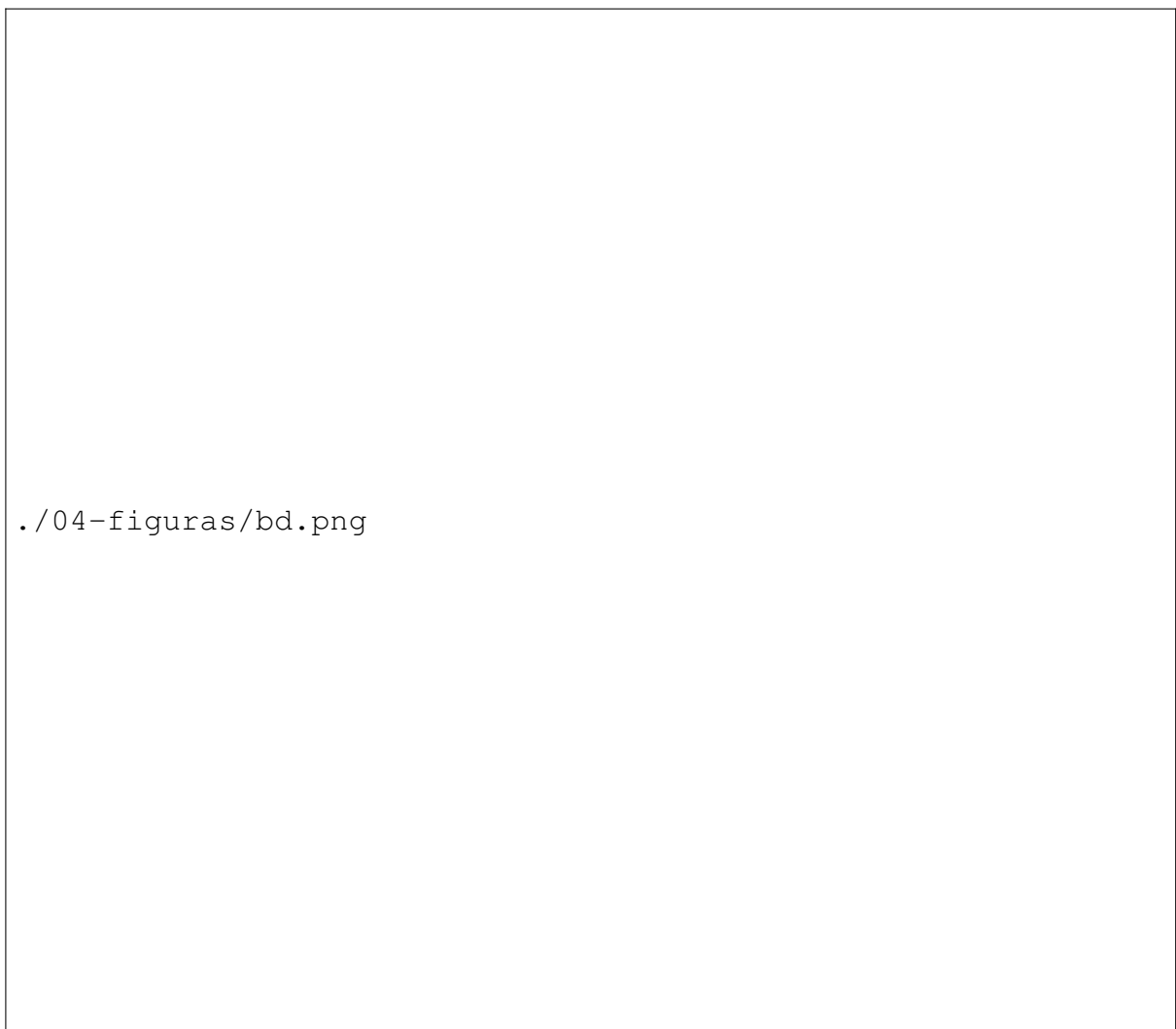
SILVÉRIO, G. de A.; OLTRAMARI, K. Desperdício de alimentos em unidades de alimentação e nutrição brasileiras food waste in brazilian units food and nutrition. **Ambiência**, v. 10, n. 1, p. 125–133, 2014.

ZAPAROLI, A. **Protótipo de Software para Controle de Acesso de Funcionários utilizando Redes Neurais para Identificação de Impressão Digital**. Tese (Doutorado) — UNIVERSIDADE REGIONAL DE BLUMENAU, 2002.

APÊNDICE A – MODELAGEM CONCEITUAL DO BANCO DE DADOS E *USER STORIES*

Como podemos observar na Figura 22, o banco de dados possui cinco documentos, sendo eles, *User*, que armazena as informações dos usuários sendo eles servidores da CAEST e Restaurante Estudantil; *Student*, que armazena as informações dos alunos cadastrados no sistema; *Presence*, que armazena o histórico do aluno de comparecimento ao restaurante; *List Instance*, que guarda os dados da lista de alunos autorizados de cada dia da semana; E por fim a *List Template*, que armazena os dados do template das listas do semestre.

Figura 22 – Modelo conceitual do Banco de Dados



Já os Casos de uso e Requisitos Funcionais serão desenvolvidos neste apêndice em forma de *User Stories* (Quadro 3). Para acessá-los os usuários precisam estar autenticados.

Quadro 3 – *User Stories*



./04-figuras/usp.png

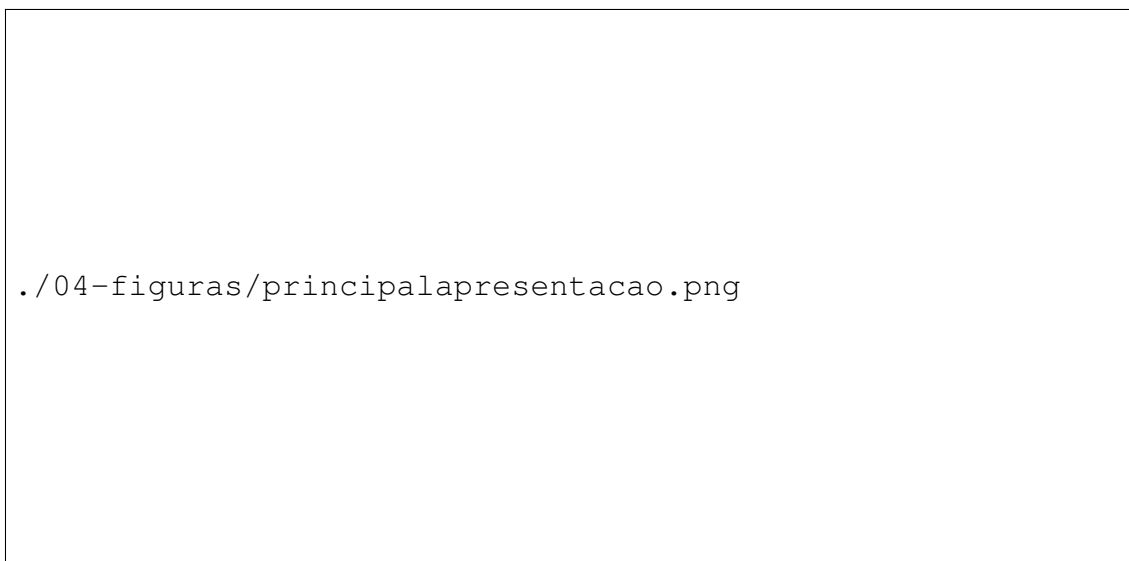
Fonte: Elaborado pelo autor (2023)

APÊNDICE B – PÁGINAS E MÓDULOS DOS SISTEMAS

As Páginas e Módulos restantes serão apresentadas abaixo.

A página de apresentação, mostrada na Figura 23, introduz a plataforma ao usuário e fornece acesso à página de *Login*. Para prosseguir, o usuário deve apenas clicar no botão *Login*.

Figura 23 – Página de apresentação



Fonte: Elaborado pelo autor (2025)

O usuário pode navegar entre os módulos utilizando-se do Menu lateral, que pode ser acessado ao clicar no ícone de Menu ou no UniAccess, como indica a Figura 24.

Figura 24 – Menu lateral de navegação



Fonte: Elaborado pelo autor (2025)

O usuário que tem permissões de administrador do sistema pode acessar os módulos de Controle de Usuários (Figura 25) e Painel do Refeitório (Figura 26), além da página de uso externo do refeitório (Figura 14).

Figura 25 – Módulo Controle de Usuários - Listagem



Fonte: Elaborado pelo autor (2025)

Figura 26 – Módulo Refeitório Interno

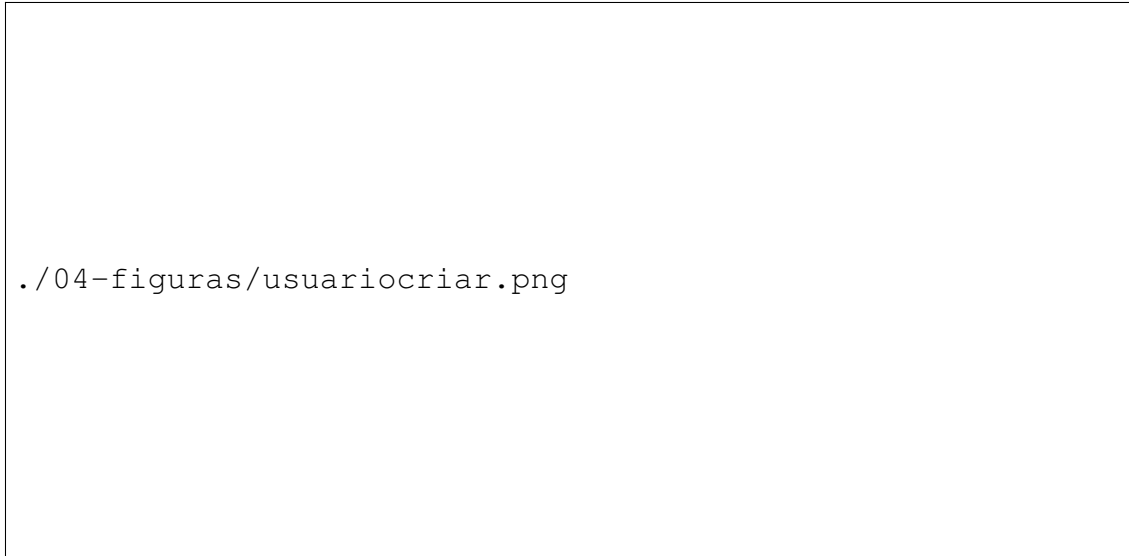


Fonte: Elaborado pelo autor (2025)

No Módulo de controle de usuários, o administrador tem acesso à todos os usuários do sistema, como mostra na Figura 25, podendo cadastrar novos usuários

como na Figura 27 e também buscar um usuário específico utilizando seu nome ou seu email cadastrado.

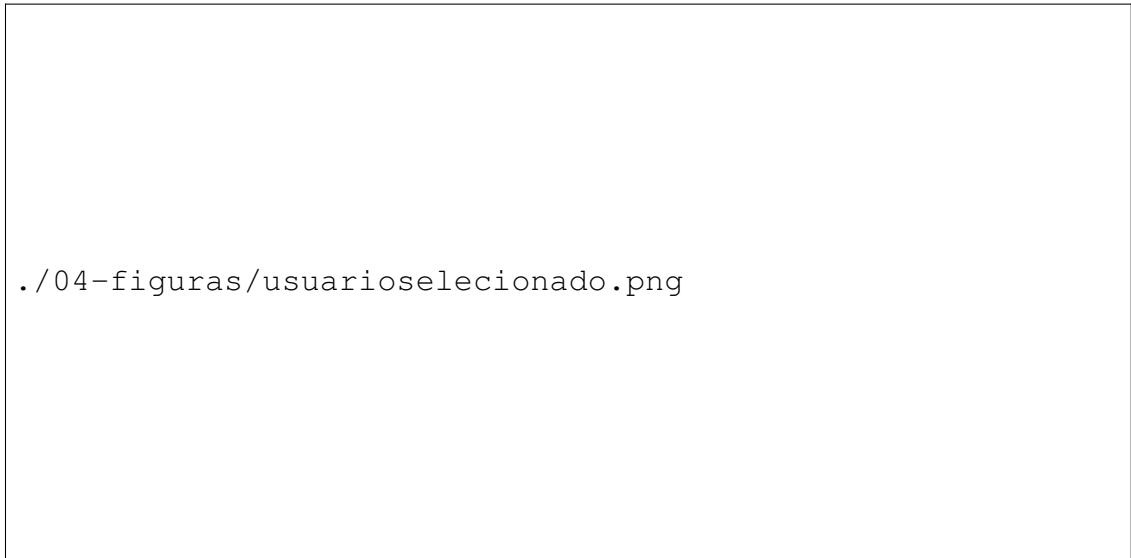
Figura 27 – Módulo Controle de Usuários - Cadastro de novo usuário



Fonte: Elaborado pelo autor (2025)

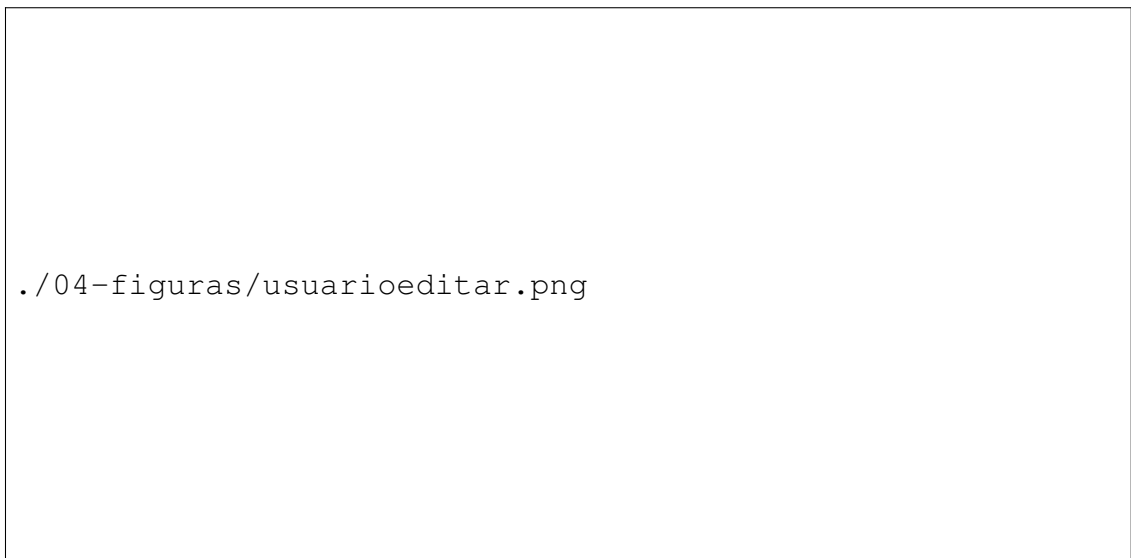
Ao selecionar um usuário, o administrador pode editá-lo ou excluí-lo, ao Editar é possível trocar o nome, email, atribuir um tipo de usuário diferente e mudar sua senha, como visto nas Figuras 28 e 29.

Figura 28 – Módulo Controle de Usuários - Dados do usuário



Fonte: Elaborado pelo autor (2025)

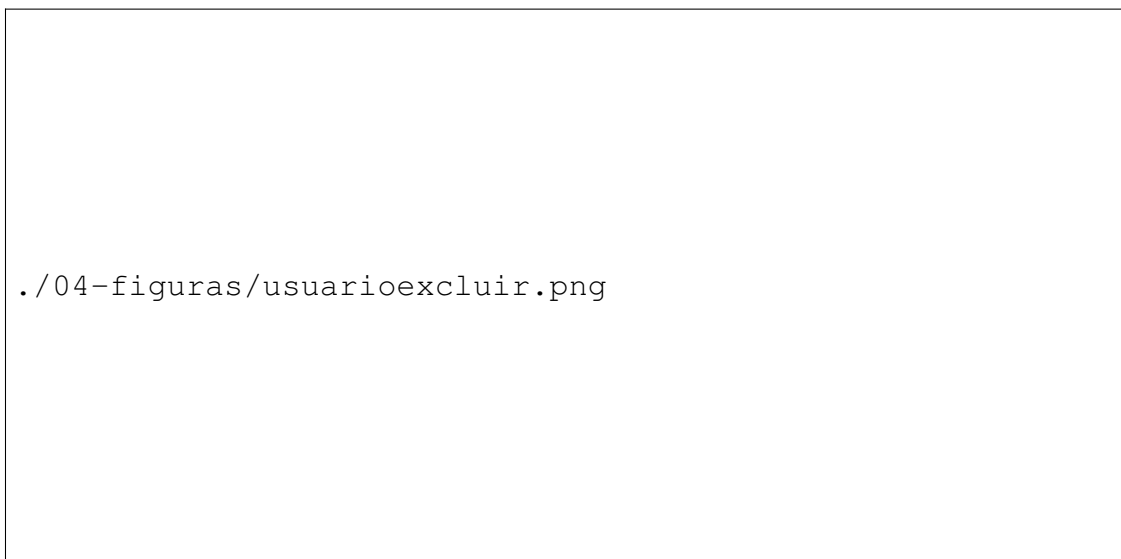
Figura 29 – Módulo Controle de Usuários - Editar usuário



Fonte: Elaborado pelo autor (2025)

Ao tentar apagar um usuário, é exibido um modal de exclusão (Figura 30).

Figura 30 – Módulo Controle de Usuários - Modal de exclusão do usuário



Fonte: Elaborado pelo autor (2025)

Na tela de Listas o usuário pode selecionar uma lista clicando sobre ela, pode filtrar as listas pela data e pode gerar as listas da semana como já foi explicado anteriormente e é mostrado na Figura 31.

Figura 31 – Módulo Controle de Listas - Listagem



Fonte: Elaborado pelo autor (2025)

No modal de criação de lista (Figura 32) o usuário deve selecionar a data para a nova lista e escolher o template a ser utilizado.

Figura 32 – Módulo Controle de Listas - Modal para criar lista



Fonte: Elaborado pelo autor (2025)

Como pode ser visto nas Figuras 33 e 34, o usuário pode editar as listas adicionando ou removendo alunos que estão autorizados a frequentar o refeitório no dia da lista selecionada e pode adicionar alunos extras em eventuais contratemplos.

Figura 33 – Módulo Controle de Listas - Editar Lista alunos



Fonte: Elaborado pelo autor (2025)

Figura 34 – Módulo Controle de Listas - Editar Lista alunos extras



Fonte: Elaborado pelo autor (2025)

Ao tentar excluir uma lista é mostrado um Modal para evitar exclusões acidentais, como indica a Figura 35.

Figura 35 – Módulo Controle de Listas - Modal de exclusão de lista



Fonte: Elaborado pelo autor (2025)

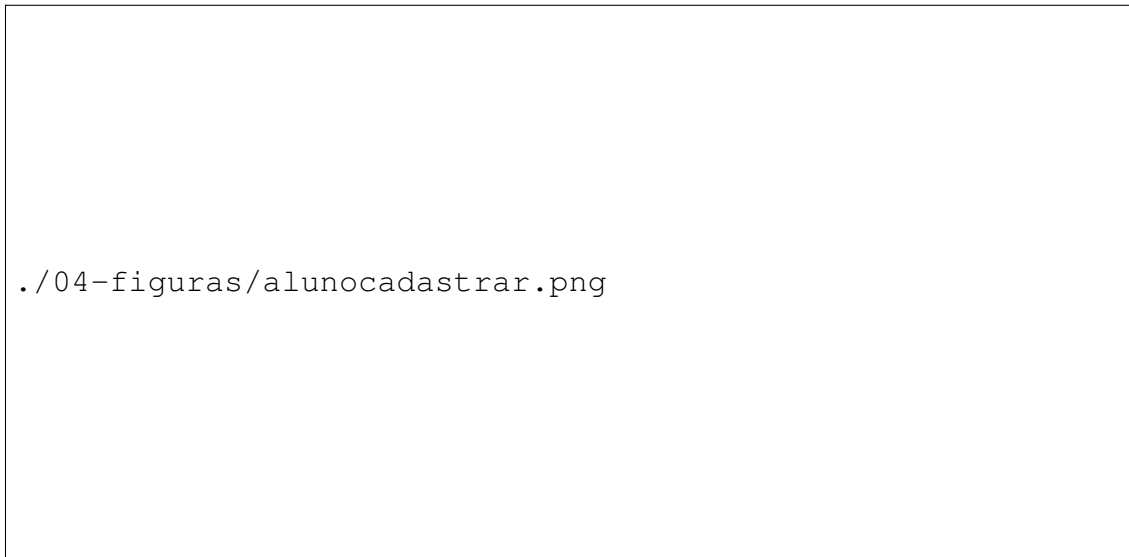
No Módulo de gerência de alunos, Figura 36, é possível filtrar os alunos por nome ou matrícula e cadastrar um novo aluno informando seu nome completo e sua matrícula como pode ser visto na Figura 37.

Figura 36 – Módulo Gerência de Alunos - Listagem



Fonte: Elaborado pelo autor (2025)

Figura 37 – Módulo Gerência de Alunos - Cadastrar novo aluno



Fonte: Elaborado pelo autor (2025)

Na edição do aluno, Figura 38, o usuário pode alterar seu nome, matrícula e autorizar ou desautorizar os dias de acesso ao refeitório.

Figura 38 – Módulo Gerência de Alunos - Edição do aluno



Fonte: Elaborado pelo autor (2025)

Ao tentar excluir um aluno, é mostrado ao usuário um modal para confirmar sua exclusão, Figura 39.

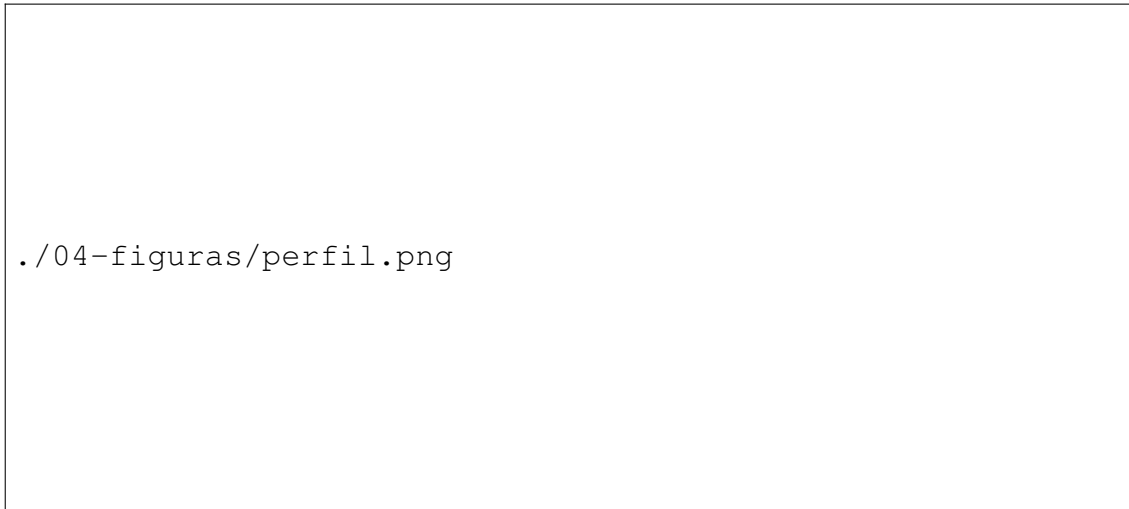
Figura 39 – Módulo Gerência de Alunos - Modal de exclusão do aluno



Fonte: Elaborado pelo autor (2025)

O usuário pode acessar seu perfil clicando em sua foto selecionando a opção "Perfil", na opção "Sair" o usuário se desconecta do sistema, onde será exibido seus dados (Figura 40) juntamente com sua foto, que pode ser alterada ao clicar sobre ela e enviar uma nova foto nos formatos *.png* ou *.jpg*, observado na Figura 41.

Figura 40 – Módulo Perfil



Fonte: Elaborado pelo autor (2025)

Figura 41 – Módulo Perfil - Alterar foto de perfil



Fonte: Elaborado pelo autor (2025)

Ao clicar em Editar perfil, o usuário pode mudar seu nome ou sua senha, como apresenta a Figura 42, para mudar o nome não é preciso preencher a senha.

Figura 42 – Módulo Perfil - Editar perfil



Fonte: Elaborado pelo autor (2025)

No Módulo dos templates das listas do semestre (Figura 43) é possível realizar uma busca por semestre ou pelo dia, criar novos templates fornecendo o semestre, o dia que o template irá representar, selecionar os alunos que possuem acesso permitido no dia selecionado e adicionar alunos extras em casos especiais (Figuras 44 e 45).

Figura 43 – Módulo Template - Listagem



Fonte: Elaborado pelo autor (2025)

Figura 44 – Módulo Template - Criar novo template



Fonte: Elaborado pelo autor (2025)

Figura 45 – Módulo Template - Criar novo template extras



Fonte: Elaborado pelo autor (2025)

Para ver os os dados de um template, o usuário deve selecionar o mesmo na lista. Também é possível editar ou excluir o template selecionado, como mostra a Figura 46.

Figura 46 – Módulo Template - Dados do template



Fonte: Elaborado pelo autor (2025)

Ao editar é possível adicionar ou remover alunos que possuem permissão para acessar o refeitório no dia do template e adicionar alunos extras como visto nas Figuras 47 e 48.

Figura 47 – Módulo Template - Editar



Fonte: Elaborado pelo autor (2025)

Figura 48 – Módulo Template - Editar extras



Fonte: Elaborado pelo autor (2025)

Se o usuário desejar apagar um template, é mostrado um modal para confirmar a exclusão e evitar exclusões acidentais (Figura 49).

Figura 49 – Módulo Template - Modal de exclusão do template



Fonte: Elaborado pelo autor (2025)

Sempre que é feita alguma operação em qualquer tela, é exibida uma mensagem de sucesso (Figura 51) para avisar ao usuário que tudo ocorreu bem, ou uma mensagem de erro (Figura 50) para orientar o usuário sobre o que aconteceu de errado.

Figura 50 – Mensagem de erro



Fonte: Elaborado pelo autor (2025)

Figura 51 – Mensagem de sucesso



Fonte: Elaborado pelo autor (2025)

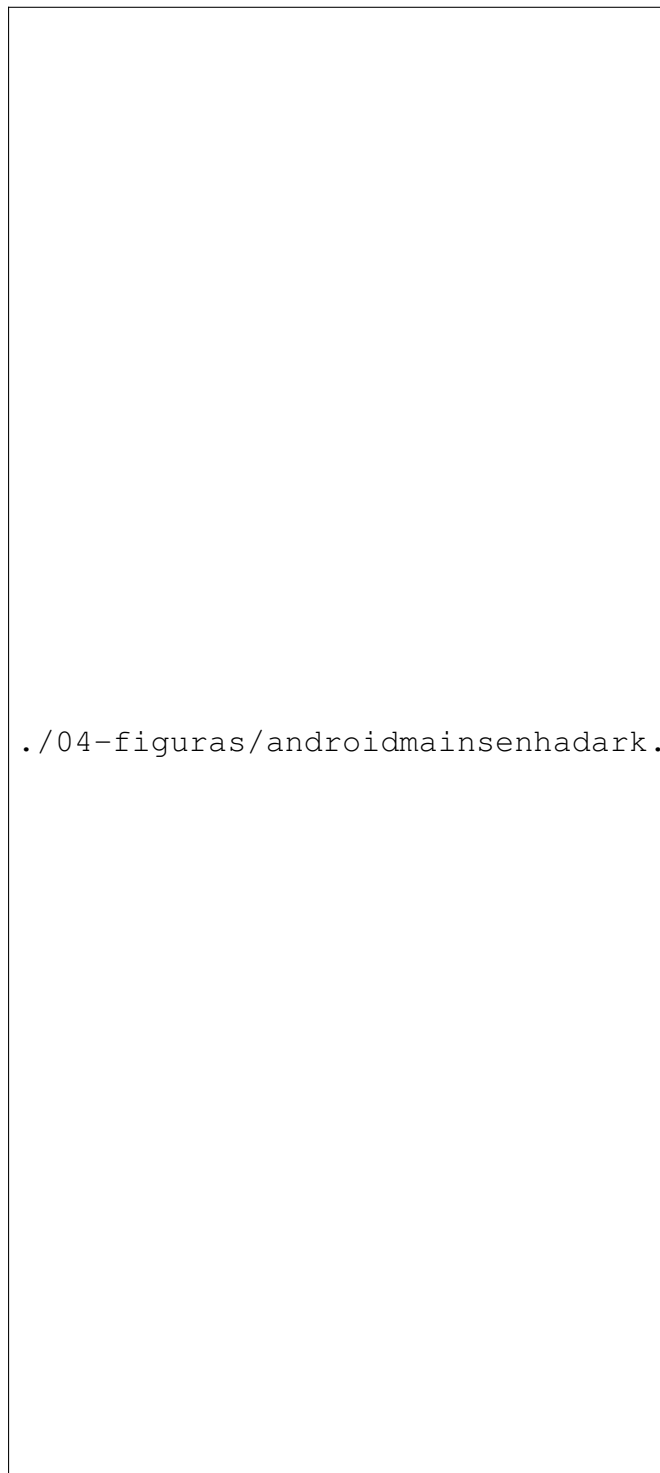
As Figuras 52, 53, 54 e 55 mostram as telas do aplicativo android utilizando tema escuro.

Figura 52 – Aplicativo UniAccess - Tela Principal tema escuro



Fonte: Elaborado pelo autor (2025)

Figura 53 – Aplicativo UniAccess - Modal para alterar senha tema escuro



Fonte: Elaborado pelo autor (2025)

Figura 54 – Aplicativo UniAccess - Mensagem de presença confirmada tema escuro



./04-figuras/androidpresencaconfirmadadark.png


Fonte: Elaborado pelo autor (2025)

Figura 55 – Aplicativo UniAccess - Mensagem de presença confirmada tema escuro



./04-figuras/androidpresencanaoconfirmadadark.png

Fonte: Elaborado pelo autor (2025)

	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
	Campus Cajazeiras - Código INEP: 25008978
	Rua José Antônio da Silva, 300, Jardim Oásis, CEP 58.900-000, Cajazeiras (PB)
	CNPJ: 10.783.898/0005-07 - Telefone: (83) 3532-4100

Documento Digitalizado Ostensivo (Público)

Trabalho de Conclusão de Curso

Assunto:	Trabalho de Conclusão de Curso
Assinado por:	Junior Silva
Tipo do Documento:	Anexo
Situação:	Finalizado
Nível de Acesso:	Ostensivo (Público)
Tipo do Conferência:	Cópia Simples

Documento assinado eletronicamente por:

- **Helio José da Silva Júnior, ALUNO (201812010014) DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - CAJAZEIRAS**, em 31/03/2025 10:16:25.

Este documento foi armazenado no SUAP em 31/03/2025. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1440898

Código de Autenticação: 7b23e78f4e

