

CENTRO UNIVERSITÁRIO FEI  
VITÓRIA CAROLYNE MARTINS SOUZA

**PROJETO FEIFOOD**

Relatório Final do Projeto apresentado ao Centro Universitário FEI, como parte dos requisitos da Disciplina “CCP110 – Fundamentos de Algoritmos” ministrada pelo Prof. Dr. Rafael Gomes Alves.

São Paulo  
2025

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>3</b>
<b>2 DESENVOLVIMENTO.....</b>	<b>4</b>
<b>3 RESULTADOS .....</b>	<b>6</b>
<b>4 CONCLUSÕES.....</b>	<b>8</b>
<b>REFERÊNCIAS.....</b>	<b>8</b>

## 1 INTRODUÇÃO

A proposta apresentada para o desenvolvimento do projeto FEIFood consiste na criação de uma plataforma para pedidos de comida, explorando as funcionalidades básicas de plataformas já conhecidas, como o IFood, por exemplo e utilizando a linguagem Python e arquivos de texto, conforme pré-requisitos.

- a) Objetivo do Projeto: O projeto FEIFood tem como principal objetivo a criação de uma plataforma cujo objetivo é comportar pedidos de comida.
- b) Tecnologias Utilizadas: Fora utilizada a linguagem de programação Python no desenvolvimento do projeto, além disso a persistência de dados foi implementada através do uso de arquivos “.txt”, conforme requisito do projeto.
- c) Escopo do Trabalho: O escopo do projeto abrange as funcionalidades essenciais de perfil de usuário, com o ciclo completo de gerenciamento de conta, busca por alimentos e gestão de pedidos, incluindo avaliação final.

## 2 DESENVOLVIMENTO

O projeto foi estruturado em módulos Python separados (*main.py*, *usuario.py*, *pedido.py*, *cardapio.py* e *config.py*), visando a organização e a separação clara de responsabilidades:

*config.py*: Módulo responsável por armazenar todas as variáveis de estado globais (como *lista\_usuarios*, *lista\_pedidos*, *espaco*) e os dicionários de navegação (*opcoes*, *opcao\_gerenc\_pedido*), eliminando a dependência circular entre os módulos.

*usuario.py*: Concentra toda a lógica de persistência (*ler\_usuarios* e *salvar\_cadastro\_usuario*) e autenticação (*cadastro\_usuario*, *login\_usuario* e *remover\_usuario*).

*cardapio.py*: Armazena a *lista\_alimentos* e as funções de visualização (*exibir\_cardapio*) e pesquisa (*buscar\_alimentos*).

*pedido.py*: Contém a lógica de gestão de pedidos (*criar\_pedido*, *editar\_pedido* e *excluir\_pedido*), incluindo funções de cálculo (*calcular\_total*), e sua manipulação (*adicionar\_item* e *remover\_item*).

*main.py*: Serve como ponto de entrada e orquestra o fluxo de navegação principal, chamando as funções dos outros módulos.

- a) Estrutura dos Arquivos de Dados: A persistência dos dados é realizada em arquivos de texto (*.txt*).
- I. *usuarios.txt*: Cada linha armazena os dados de um usuário, com os campos *nome*, *e-mail* e *senha* separados por vírgula. A função *salvar\_cadastro\_usuario* sobrescreve o arquivo a cada alteração.
  - II. *pedidos.txt*: Cada linha armazena um pedido completo. Os dados primários são separados por vírgulas, e os itens dentro da sacola do pedido são serializados com separadores específicos (*|* e *-*) para permitir a reconstrução precisa do dicionário de pedidos pela função *historico\_pedido*.
- b) Funcionalidades do Usuário: Todas as funcionalidades obrigatórias do perfil de usuário foram implementadas, com validação de entradas e persistência de dados, conforme abaixo:
- I. Cadastro e Login: As funções utilizadas no cadastro e login do usuário são *cadastro\_usuario()* e *login\_usuario()*. A função de cadastro valida a duplicidade do e-mail antes de adicionar o novo usuário à *lista\_usuarios*. O login faz a autenticação do usuário pela combinação do e-mail e senha.
  - II. Busca de Alimentos: A função utilizada na busca de alimentos é *buscar\_alimentos()*. Essa função permite buscar itens na *lista\_alimentos* por nome (*case-insensitive*), exibindo os resultados formatados.
  - III. Gestão de Pedidos: A função utilizada para a criação de pedidos é *criar\_novo\_pedido()*, permitindo ao usuário selecionar itens do cardápio. O total é calculado dinamicamente pela função *calcular\_total()* e o pedido é salvo com um *id\_pedido* sequencial. Outras funções utilizadas são *editar\_pedido()* e *excluir\_pedido()*, estas permitem modificar pedidos existentes. A função *editar\_pedido()* utiliza as funções auxiliares *adicionar\_item()* e *remover\_item()*, e chama *salvar\_pedido()* após a finalização.
  - IV. Avaliação de Pedidos: A função utilizada é *avaliar\_pedido()* e permite que o usuário atribua uma nota de 0 a 5 estrelas a um pedido concluído.

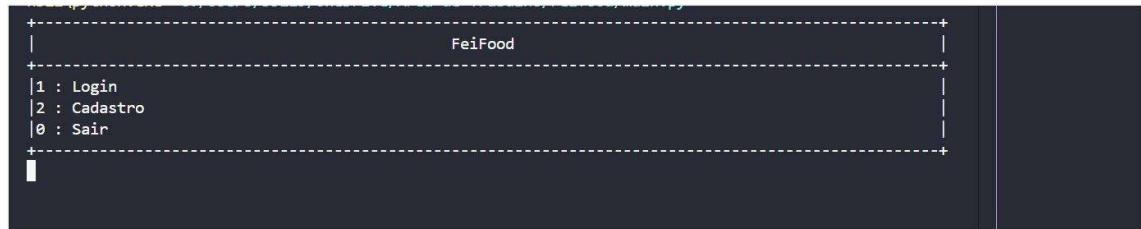
A nota é armazenada no campo *avaliação* do dicionário do pedido e persistida no arquivo.

### 3 RESULTADOS

Neste capítulo será demonstrado o êxito nas execuções das funcionalidades desenvolvidas neste projeto da plataforma criada, a FEIFood.

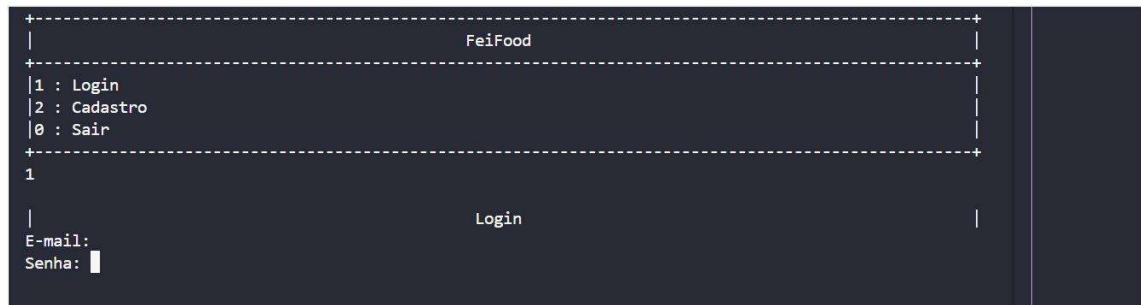
- a) Apresentação das Funcionalidades: As imagens abaixo exemplificam as principais funcionalidades da plataforma FEIFood, dentre elas estão as telas de cadastro e login do usuário, a busca por alimentos, a gestão e a avaliação dos pedidos:

*Tela inicial*



```
+-----+
|                               FeiFood
+-----+
|1 : Login
|2 : Cadastro
|0 : Sair
+-----+
```

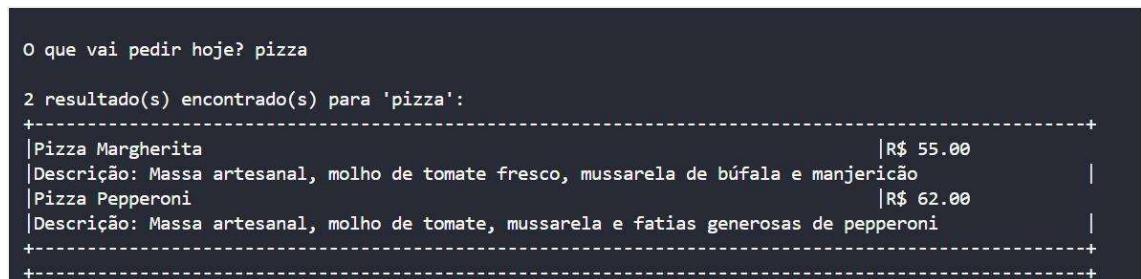
*Tela de login*



```
+-----+
|                               FeiFood
+-----+
|1 : Login
|2 : Cadastro
|0 : Sair
+-----+
1

|                               Login
|-----+
| E-mail: [REDACTED]
| Senha: [REDACTED]
```

*Tela de busca por alimentos*



O que vai pedir hoje?	pizza
2 resultado(s) encontrado(s) para 'pizza':	
Pizza Margherita	R\$ 55.00
Descrição: Massa artesanal, molho de tomate fresco, mussarela de búfala e manjericão	
Pizza Pepperoni	R\$ 62.00
Descrição: Massa artesanal, molho de tomate, mussarela e fatias generosas de pepperoni	

*Tela de gestão de pedidos*

```

+-----+
|                               Gerenciamento de Pedidos
+-----+
|1 : Criar
|2 : Editar
|3 : Excluir
|0 : Voltar
+-----+
Digite o número que corresponde a opção desejada: 1

```

*Tela de avaliação do pedido*

```

+-----+
|                               Avaliar Pedido
+-----+
|                               Seus Pedidos
+-----+
ID: 1|Valor: R$ 56.50|          Avaliação: 0.0|Status da Avaliação: PENDENTE
+-----+
Digite o ID do pedido que deseja avaliar (ou 0 para cancelar): 1
Atribua a nota ao pedido 1 (0 a 5 estrelas): 4.5

Pedido 1 avaliado com sucesso! Nota: 4.5 estrelas.
+-----+

```

- b) Demonstração da Persistência de Dados: Neste item será demonstrado o salvamento correto dos dados dos usuários cadastrados no arquivo de texto:

*Arquivo de texto antes do cadastro do usuário*

*Arquivo de texto depois do cadastro do usuário*

- c) Validação das Regras: Como exemplo, será demonstrado o que ocorre se um usuário procurar por um alimento inexistente na plataforma ou se o usuário errar seu e-mail ou senha no momento do login à plataforma, garantindo a experiência do usuário:

### *Busca por alimento não cadastrado*

```

|                               Busca por Alimentos
|
|                               Cardápio Completo
+-----+
|ITEM          |PREÇO      |INFORMAÇÕES
|Spaghetti Carbonara |R$ 38.50  |Macarrão spaghetti, molho pomod...
|Lasanha Bolonhesa   |R$ 45.00  |Massa artesanal com molho pomod...
|Pizza Margherita    |R$ 55.00  |Massa artesanal, molho de tomate...
|Pizza Pepperoni     |R$ 62.00  |Massa artesanal, molho de tomate...
|Tiramisu           |R$ 23.00  |feita com queijo mascarpone, bala...
|Panna Cotta com Frutas |R$ 18.50  |feita com creme de leite cozido...
|Suco de Laranja      |R$ 12.00  |suco de laranja 100% natural, e...
|Soda Italiana       |R$ 18.00  |feita com xarope de frutas, águ...
+-----+
O que vai pedir hoje? tapioca
Nenhum alimento encontrado para: 'tapioca'
+-----+

```

### *E-mail ou senha incorreta*

```

|                               Login
|
|E-mail: feispmv@edu.br
|Senha: 1234
|E-mail ou senha inválidos.
+-----+

```

## 4 CONCLUSÕES

O desenvolvimento da plataforma FEIFood demonstrou a aplicação prática dos conceitos de Estruturas e Dados (listas e dicionários) e Manipulação de Arquivos abordados na disciplina. O desafio da persistência de dados foi superado com a criação de funções de serialização e desserialização (*salvar\_pedido* e *historico\_pedido*).

A modularização do código, realizada nas etapas finais, garantiu a organização lógica das funcionalidades, resolvendo o problema de dependência circular através da criação de um módulo central de configuração. O projeto está completo dentro do escopo definido para o perfil de usuário.

Desta forma, a plataforma FEIFood respeitou todos os requisitos e se tornou funcional, conforme demonstrado no capítulo RESULTADOS, onde restou evidente o total sucesso das funcionalidades.

## REFERÊNCIAS

BIONDI, Gabriela. Fundamentos de Algoritmos – Aula 12: Arquivos. São Bernardo do Campo: Centro Universitário FEI, 1º semestre de 2025.

BIONDI, Gabriela. Fundamentos de Algoritmos – Aula 13: Manipulação de Strings, Módulos/Pacotes e GUI. São Bernardo do Campo: Centro Universitário FEI, 1º semestre de 2025.

CURSO EM VÍDEO. Curso Python #21 – Funções (Parte 2). YouTube, 22 nov. 2018. Disponível em: [https://www.youtube.com/watch?v=etjJ\\_4Eqrk8](https://www.youtube.com/watch?v=etjJ_4Eqrk8). Acesso em: 9 nov. 2025.

CURSO EM VÍDEO. Curso Python #22 – Módulos e Pacotes. YouTube, 26 ago. 2019. Disponível em: [https://www.youtube.com/watch?v=s3r8\\_Aug4y8](https://www.youtube.com/watch?v=s3r8_Aug4y8). Acesso em: 9 nov. 2025.

CURSO EM VÍDEO. Curso Python #23 - Tratamento de Erros e Exceções. YouTube, ago. 2019. Disponível em: <https://www.youtube.com/watch?v=xz2B3bfNjEk>. Acesso em: 9 nov. 2025.

PYTHON SOFTWARE FOUNDATION. A Referência da Linguagem Python. Disponível em: <https://docs.python.org/pt-br/3/reference/index.html>. Acesso em: 07 nov. 2025.