**BreadTrack**: Trabalho Integrado dos Componentes de Banco de Dados II, Engenharia de Software I e Programação II

Arthur Borger Kochem **[[1]](#footnote-1)**

Arthur Costa Gruber[[2]](#footnote-2)\*[[3]](#footnote-3)\*

Bruno Gabriel Konzen[[4]](#footnote-4)\*\*[[5]](#footnote-5)\*

Franciele Petry\*\*[[6]](#footnote-6)\*[[7]](#footnote-7)\*

Leonardo Agostini Costa[[8]](#footnote-8)\*[[9]](#footnote-9)\*[[10]](#footnote-10)\*\*[[11]](#footnote-11)\*

Otília Donato Barbosa\*\*[[12]](#footnote-12)\*[[13]](#footnote-13)\*[[14]](#footnote-14)\*\*

Roberson Junior Fernandes Alves\*\*[[15]](#footnote-15)\*[[16]](#footnote-16)\*\*\*\*

**Resumo**

Este artigo apresenta um estudo de caso sobre a implementação de um Sistema de Gerenciamento de Banco de Dados (SGBD) em uma padaria, com o objetivo de aprimorar o gerenciamento de informações e processos relacionados ao estabelecimento. Inicialmente, foi realizada uma análise das necessidades da padaria, identificando os principais requisitos para o SGBD. Os requisitos incluíam o armazenamento e a organização de dados de clientes, produtos, estoque, vendas, pedidos de fornecedores e funcionários. A modelagem do banco de dados foi realizada utilizando-se a abordagem relacional, onde foram definidas as entidades, atributos e relacionamentos relevantes para a padaria. Com o banco de dados modelado, procedeu-se o estudo arquitetando o banco com a base da engenharia de software, seguidos de modelos ágeis que facilitam o entendimento do software em geral. Por fim, foi realizado a implementação do SGBD, criando as tabelas, estabelecendo as chaves primárias e estrangeiras e populando o banco com os dados existentes na padaria. Ao longo da implementação, foram realizados testes para verificar a eficácia do SGBD. Foram utilizadas ferramentas como Trello para planejar o projeto juntamente com o PostgreSQL, Eclipse, Visual Paradigm e DBeaver para modelagem e estruturação e o versionamento foi feito via GitHub. É crucial enfatizar a necessidade de uma abordagem robusta em engenharia de software, que compreenda aspectos como o controle de versionamento, para satisfazer as demandas em constante evolução de uma agência de eventos de renome.

**Palavras-chave**: Desenvolvedores. Engenharia. Software. Programação. Dados. Informações. Padaria. SGBD. Implementação.

# **INTRODUÇÃO**

Com o avanço tecnológico, diversas áreas têm se beneficiado da automação e otimização de processos, em que a eficiência na gestão de informações desempenha um papel crucial, a implementação de soluções tecnológicas torna-se essencial para o sucesso e reconhecimento de estabelecimentos especializados. Este artigo discute o desenvolvimento de um Sistema Gerenciador de Banco de Dados para o controle de uma padaria. Neste contexto, surge a proposta de desenvolvimento do software "BreadTrack", dedicado a atender às demandas específicas de uma padaria renomada.

A padaria moderna não se limita apenas à produção de pães e doces; ela representa um ambiente dinâmico onde a eficiência operacional é crucial. O sistema visa oferecer uma solução integrada, otimizando desde o controle de estoque e gestão de fornecedores até o acompanhamento de pedidos e preferências dos clientes.

Assim, diante desse contexto, o artigo apresentará a abordagem do BreadTrack, destacando sua proposta inovadora para a gestão de pedidos em padarias, abrangendo desde a modelagem inicial até os resultados obtidos durante o desenvolvimento do sistema.

Uma solução abrangente para otimizar a gestão da padaria "BreadTrack" seria a implementação de um sistema integrado que englobe não apenas o controle de estoques, pedidos e vendas, mas também considere aspectos como a experiência do cliente, a eficiência operacional e a adaptação às demandas dinâmicas do setor alimentício, tudo isso integrado em uma gestão de pedidos eficiente juntamente com uma análise de relatórios em um sistema integrado e intuitivo.

# **MATERIAIS E MÉTODOS**

# Para o desenvolvimento desse sistema, foi utilizado o banco de dados relacional PostgreSQL, já desenvolvido anteriormente, juntamente com a ferramenta Dbeaver. Na parte de programação foi utilizada a linguagem Java juntamente com a ferramenta Eclipse como ambiente de desenvolvimento para fazer a parte do back-end com Spring Boot, com o intuito de fazer a conexão com o banco de dados . Já na parte de front-end foi ultilizado HTML com Bootstrap. No desenvolvimento dos modelos foram utilizados os requisitos levantados anteriormente junto com a ferramenta Visual paradigm. O versionamento do projeto foi realizado através do GitHub. E por fim na divisão e planejamento das atividades e tarefas foi utilizado a ferramenta Trello.

# **2. 1 MODELAGEM**

A partir dos requisitos já levantados, foi dado início a construção do diagrama de casos de uso e dos diagramas de sequência, de atividade, de estado e de classes utilizando a ferramenta Visual Paradigm. Abaixo as figuras 1, 2, 3, 4 e 5 mostram os diagramas.

Figura 1: Diagrama de sequência.

Linha do tempo

Descrição gerada automaticamente com confiança média

Figura 2: Diagrama de casos de uso.

Diagrama

Descrição gerada automaticamente

Figura 3: Diagrama de estado

Diagrama

Descrição gerada automaticamente

Figura 4: Diagrama de atividade.

Diagrama

Descrição gerada automaticamente

Figura 5: Diagrama de classes

Diagrama

Descrição gerada automaticamente

# **DESENVOLVIMENTO**

Inicialmente foi criado um KANBAN na ferramenta Trello, para organizar as tarefas e decidir quais atividades cada integrante ficou responsável. A Figura 1 mostra o KANBAN inicialmente em seus primeiros dias.

Figura 1 - Estado inicial do KANBAN

Interface gráfica do usuário

Descrição gerada automaticamente

Fonte: Autoria própria (2023).

É perceptível que no começo as atividades eram básicas e com o objetivo de organização do projeto, somente mais tarde que as modelagens e implementações foram adicionadas como atividades como mostra a Figura 2 e 3.

Figura 2 - KANBAN das atividades

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Fonte: Autoria própria (2023).

Figura 3 - Estágio final do KANBAN

Tela de celular com aplicativo aberto

Descrição gerada automaticamente

Fonte: Autoria própria (2023).

# **SISTEMA BREADTRACK**

# **CÓDIGOS**

# **CONCLUSÃO**

# **REFERENCIAS**

1. \* Discente do Curso de Ciência da Computação

   Unoesc - Campus de São Miguel do Oeste

   Rua Oiapoc, 2011. São Miguel do Oeste-SC

   arthurkochem12@gmail.com

   \*\* Discente do Curso de Ciência da Computação

   Unoesc - Campus de São Miguel do Oeste

   Rua Oiapoc, 2011. São Miguel do Oeste-SC

   arthurcg21@gmail.com

   \*\*\* Discente do Curso de Ciência da Computação

   Unoesc - Campus de São Miguel do Oeste

   Rua Oiapoc, 2011. São Miguel do Oeste-SC

   brunogkonzen@hotmail.com

   \*\*\*\* Mestre em Informática

   Docente do Curso de Ciência da Computação

   Unoesc - Campus de São Miguel do Oeste

   Rua Oiapoc, 211. São Miguel do Oeste-SC

   franciele.petry@unoesc.edu.br

   \*\*\*\*\* Discente do Curso de Ciência da Computação

   Unoesc - Campus de São Miguel do Oeste

   Rua Oiapoc, 2011. São Miguel do Oeste-SC

   Laccosta1242@gmail.com

   \*\*\*\*\*\* Mestre em Engenharia Biomédia e Informática Industrial

   Docente do Curso de Ciência da Computação

   Unoesc - Campus de São Miguel do Oeste

   Rua Oiapoc, 2011. São Miguel do Oeste-SC

   otilia.barbosa@unoesc.edu.br

   \*\*\*\*\*\*\* Mestre em Computação Aplicada (UEPG)

   Professor do Curso de Bacharelado em Ciência da Computação

   Unoesc - Campus de São Miguel do Oeste

   Rua Oiapoc, 211 – São Miguel do Oeste – SC

   roberson.alves@unoesc.edu.br [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)
3. [↑](#footnote-ref-3)
4. [↑](#footnote-ref-4)
5. [↑](#footnote-ref-5)
6. [↑](#footnote-ref-6)
7. [↑](#footnote-ref-7)
8. [↑](#footnote-ref-8)
9. [↑](#footnote-ref-9)
10. [↑](#footnote-ref-10)
11. [↑](#footnote-ref-11)
12. [↑](#footnote-ref-12)
13. [↑](#footnote-ref-13)
14. [↑](#footnote-ref-14)
15. [↑](#footnote-ref-15)
16. [↑](#footnote-ref-16)