

**JOSÉ HENRIQUE CAMARGO LEOPOLDO E SILVA
RAFAEL JARDIM PASTOR**

**SISTEMA DE MONITORAMENTO E CONTROLE
DA FERMENTAÇÃO DE CERVEJAS**

São Paulo
2020

**JOSÉ HENRIQUE CAMARGO LEOPOLDO E SILVA
RAFAEL JARDIM PASTOR**

**SISTEMA DE MONITORAMENTO E CONTROLE
DA FERMENTAÇÃO DE CERVEJAS**

Trabalho apresentado à Escola Politécnica
da Universidade de São Paulo para ob-
tenção do Título de Engenheiro Eletricista.

São Paulo
2020

**JOSÉ HENRIQUE CAMARGO LEOPOLDO E SILVA
RAFAEL JARDIM PASTOR**

**SISTEMA DE MONITORAMENTO E CONTROLE
DA FERMENTAÇÃO DE CERVEJAS**

Trabalho apresentado à Escola Politécnica
da Universidade de São Paulo para ob-
tenção do Título de Engenheiro Eletricista.

Orientador:

Prof. Dr. Carlos Eduardo Cugnasca

São Paulo
2020

Dedicatória

AGRADECIMENTOS

Thanks...

“Epígrafe”

-- Autor

RESUMO

Resumo...

Palavras-Chave – Palavra, Palavra, Palavra, Palavra, Palavra.

ABSTRACT

Abstract...

Keywords – Word, Word, Word, Word, Word.

LISTA DE FIGURAS

1	Diagrama de Casos de Uso da UML. Criado com a plataforma Visual Paradigm Online Diagrams Express Edition	22
2	Diagrama de Entidade Relacionamento.	25

LISTA DE TABELAS

LISTA DE SÍMBOLOS

Lista de símbolos...

SUMÁRIO

1	Introdução	13
1.1	Objetivo	13
1.2	Motivação	13
1.3	Organização do Trabalho	13
2	Contexto	14
2.1	História da Cerveja	14
2.2	Fabricação de Cervejas	14
2.2.1	Fermentação e Maturação	14
2.3	Estado da Arte de Sistemas de Produção Artesanal	14
2.3.1	Análise de Soluções Existentes	14
3	Metodologia do Trabalho	15
4	Projeto	16
4.1	Especificação de Requisitos Técnicos	16
4.1.1	Sistema Hardware	16
4.1.1.1	Requisitos funcionais de Hardware	16
4.1.1.2	Requisitos não funcionais de Hardware	17
4.1.2	Sistema Software	17
4.1.2.1	Requisitos funcionais de Software	17
4.1.2.2	Requisitos não funcionais de Software	18
4.2	Projeto de Controle	18
4.2.1	Modelagem Térmica do Sistema	18
4.3	Projeto de Hardware	18

4.3.1	Sensores e Atuadores	18
4.4	Projeto de Software	18
4.4.1	Especificação dos Casos de Uso	19
	UC - 1: Registro de Dispositivo	19
	UC - 2: Cadastro de Receitas	20
	UC - 3: Cadastro de Lotes	20
	UC - 4: Envio das informações do Lote para Dispositivo	21
	UC - 5: Envio das informações do Dispositivo para o Sistema	21
	UC - 6: Visualização das Informações dos Lotes	21
4.4.2	Modelo de Entidade-Relacionamento	22
4.4.3	Arquitetura de Software	23
	Camada de Interface	23
	Camada Intermediária	24
	Camada de Microserviços	24
4.4.4	Tecnologias escolhidas	24
4.4.5	Protótipos das Interfaces	24
5	Implementação	26
6	Testes e Avaliação do Protótipo	27
7	Considerações Finais	28
7.1	Conclusões do Projeto de Formatura	28
7.2	Contribuições	28
7.3	Perspectivas de Continuidade	28
	Apêndice A	29
	Apêndice B – Beta	30

Anexo A – Alpha	31
Anexo B	32

1 INTRODUÇÃO

1.1 Objetivo

O objetivo geral deste estudo é desenvolver um protótipo de um sistema que realize o monitoramento e controle do processo de fermentação de cervejas. O objetivo específico é possibilitar às cervejarias de pequeno e médio porte o desenvolvimento da capacidade de testes de novas receitas de cervejas, a fim de garantir a reprodutibilidade e qualidade das mesmas por meio do controle do processo de fermentação.

1.2 Motivação

1.3 Organização do Trabalho

2 CONTEXTO

2.1 História da Cerveja

2.2 Fabricação de Cervejas

2.2.1 Fermentação e Maturação

2.3 Estado da Arte de Sistemas de Produção Artesanal

2.3.1 Análise de Soluções Existentes

3 METODOLOGIA DO TRABALHO

4 PROJETO

4.1 Especificação de Requisitos Técnicos

O protótipo deve ser capaz de monitorar e controlar o processo de fermentação de cervejas, seguindo as configurações de receita definidas pelo usuário. Todas as informações coletadas devem ser disponibilizadas ao usuário com a finalidade de possibilitar maior entendimento e reprodutibilidade do processo. O projeto foi dividido em dois sistemas: um Hardware, encarregado das medições e controle, e um Software, responsável por exibir informações e estabelecer uma interface com o usuário. A partir dessas premissas, foram determinados os seguintes requisitos para cada um dos sistemas.

4.1.1 Sistema Hardware

4.1.1.1 Requisitos funcionais de Hardware

HW-F-1) O sistema deve monitorar a temperatura (entre 0 e 100 °C), com precisão de 0,5 °C e intervalo de 1 minuto.

HW-F-2) O sistema deve monitorar o pH em intervalo de 1 minuto.

HW-F-3) O sistema deve monitorar a densidade relativa (entre 1,000 e 1,150), com precisão de 0,001 em relação à água a 20°C e intervalo de 1 minuto.

HW-F-4) O sistema deve controlar a temperatura de até 50 Litros de mosto em fermentação, com desvio máximo de 0,5°C em relação ao valor definido pelo usuário e diferença máxima de 10°C em relação ao ambiente.

HW-F-5) O sistema deve seguir o perfil de controle (temperatura x tempo) definido pelo usuário no Software.

HW-F-6) Os dados monitorados devem ser enviados para o Software a cada 5 minutos por meio de rede sem fio.

4.1.1.2 Requisitos não funcionais de Hardware

HW-NF-1) O sistema deve ser acoplável a fermentadores de até 50 Litros disponíveis no mercado.

HW-NF-2) Em caso de perda de conexão com o Software, o sistema deve tentar enviar os dados ainda não enviados a cada ciclo de envio.

HW-NF-3) Caso o sistema tenha uma oscilação no fornecimento de energia, ele deve ser capaz de voltar ao funcionamento normal.

HW-NF-4) Os dados monitorados devem ser armazenados temporariamente, por no mínimo 15 dias, no Hardware.

4.1.2 Sistema Software

4.1.2.1 Requisitos funcionais de Software

SW-F-1) O sistema deve fornecer acesso ao usuário após identificação com usuário e senha

SW-F-2) O sistema deve fornecer as informações instantâneas das fermentações em progresso.

SW-F-3) O sistema deve permitir acesso às informações históricas de fermentações já realizadas.

SW-F-4) O sistema deve permitir o cadastro de receitas. Uma receita é definida por: identificação, nome, estilo e campo livre para observações. O campo livre pode evoluir para um cadastro padronizado dos ingredientes e processos realizados.

SW-F-5) O sistema deve permitir o cadastro de lotes. Um lote é definido por: identificação, receita utilizada, instante de início da fermentação, instante de fim da fermentação, variáveis personalizadas, perfil de controle e observação.

SW-F-6) O sistema deve permitir o cadastro de perfis de controle. Um perfil de controle é definido por: identificação, nome e temperatura alvo, instante (em relação ao início da fermentação).

SW-F-7) O sistema deve permitir o cadastro de variáveis personalizadas. Uma variável personalizada é definida por: identificador, lote correspondente, chave, valor e instante (em relação ao início da fermentação).

SW-F-8) O sistema deve disponibilizar, para cada lote um gráfico com a evolução de cada variável monitorada ao longo do tempo de fermentação.

SW-F-9) Os dados recebidos pelo Hardware devem ser salvos em banco de dados

SW-F-10) Em caso de perda de conexão com o Hardware, o usuário deve ser notificado por e-mail.

SW-F-11) O sistema deve permitir que o usuário realize o download de seus dados em formato de planilha.

SW-F-12) O sistema deve permitir que o usuário registre seus dispositivos.

4.1.2.2 Requisitos não funcionais de Software

SW-NF-1) As informações de cada usuário são, por padrão, particulares de cada usuário e devem seguir padrões de segurança.

SW-NF-2) As informações instantâneas devem estar disponíveis em até 1 minuto após o recebimento dos dados pelo HW.

SW-NF-3) O sistema deve ser desenvolvido na forma de Web-App, e ser responsivo a dispositivos mobile e computadores.

4.2 Projeto de Controle

4.2.1 Modelagem Térmica do Sistema

4.3 Projeto de Hardware

4.3.1 Sensores e Atuadores

4.4 Projeto de Software

A partir dos Requisitos Técnicos levantados, foi realizado o projeto de software, consistindo na definição de casos de uso a serem implementados e na elaboração de diagramas para estabelecer em maior detalhe a organização e interações dos componentes do sistema. O Diagrama de Casos de Uso ilustra os atores do sistema suas ações, o Diagrama Entidade Relacionamento define como as informações são organizadas em banco de dados, e o Diagrama de Componentes especifica a arquitetura de software adotada. Ao final

da apresentação e discussão desses tópicos, as a escolha de cada ferramenta e tecnologia escolhida é justificada.

4.4.1 Especificação dos Casos de Uso

Na definição dos casos de uso do projeto foram definidos dois atores que interagem com o sistema de software a ser desenvolvido, identificados como usuário e dispositivo. O usuário representa o utilizador humano do sistema a ser desenvolvido, responsável por todas as interações humanas necessárias. O usuário se comunica com o sistema por duas interfaces, uma aplicação web, que se comunica diretamente com o sistema, e um aplicativo para smartphone, necessário para configurações iniciais do dispositivo. O dispositivo representa o sistema hardware de controle e monitoramento, também desenvolvido neste projeto. Em relação ao sistema de software, ele é tratado como um ator com suas devidas interações; seu projeto e especificações são discutidos na seção destinado ao projeto de hardware.

Segue a especificação dos casos de uso em si, contendo a identificação de cada caso de uso, sua breve descrição, enumeração dos passos que o definem e listagem dos requisitos técnicos relacionados ao caso de uso. Com caráter ilustrativo, o Diagrama de Casos de Uso da UML é apresentado na figura 1.

UC - 1: Registro de Dispositivo

Descrição: ao obter um novo dispositivo, o usuário deve configurar seu acesso à rede Wi-Fi e registrá-lo, de modo que o sistema reconheça que aquele dispositivo pertence ao usuário.

1. Usuário acessa aplicativo em seu smartphone
2. Sistema autentica acesso do usuário
3. Aplicativo se conecta ao dispositivo
4. Usuário informa configurações da rede Wi-Fi
5. Aplicativo envia informações da rede para dispositivo
6. Dispositivo se conecta na rede e se prepara para receber mensagens do sistema
7. Aplicativo envia informações do dispositivo para o sistema

8. Sistema cadastra informações do dispositivo e usuário

Requisito relacionado: SW-F-12

UC - 2: Cadastro de Receitas

Descrição: fluxo de cadastro de receitas.

1. Usuário acessa tela de listagem de receitas
2. Sistema exibe todas as receitas referentes ao usuário
3. Usuário seleciona opção “Criar Receita” e acessa tela de cadastro de receita
4. Usuário informa nome, estilo e observações da receita e clica em “Salvar”
5. Sistema cadastra a receita no banco de dados

Requisito relacionado: SW-F-4

UC - 3: Cadastro de Lotes

Descrição: fluxo de cadastro de lotes, perfis de controle e associação de lote a um dispositivo.

1. Usuário acessa tela de listagens de receitas
2. Sistema exibe todas as receitas referentes ao usuário
3. Usuário escolhe uma receita e seleciona a opção “Criar Lote”, e acessa a tela de cadastro de lote
4. Sistema carrega listagem de perfis de controle já cadastrados e dispositivos do usuário
5. Usuário informa identificação e observações do lote
6. Usuário escolhe um perfil de controle já existente ou cria um novo perfil, informando uma identificação e cada um dos passos de controle (instante e valor alvo de temperatura)
7. Usuário seleciona qual dispositivo irá controlar a produção do lote

8. Usuário clica em “Salvar”
9. Sistema cadastra lote, perfil de controle (caso novo), associação de lote e perfil de controle, e associação de lote e dispositivo
10.)Sistema envia informações do lote para dispositivo selecionado

Requisitos relacionados: SW-F-5 e SW-F-6.

UC - 4: Envio das informações do Lote para Dispositivo

Descrição: fluxo de envio das informações do lote para o dispositivo associado ao controle daquele lote.

1. Dispositivo recebe informações do lote que foi associado por tópico de mensagens
2. Dispositivo salva informações localmente
3. Quando pronto, dispositivo inicia rotina de monitoramento e controle

Requisito relacionado: HW-F-5

UC - 5: Envio das informações do Dispositivo para o Sistema

Descrição: fluxo de envio das informações obtidas pelo monitoramento do processo pelo dispositivo para o sistema.

1. Dispositivo envia dados coletados para o sistema
2. Sistema processa dados e salva informações no banco de dados

Requisitos relacionados: HW-F-6, SW-F-9

UC - 6: Visualização das Informações dos Lotes

Descrição: fluxo para visualização das informações gerais e de evolução dos lotes correntes e passados

1. Usuário acessa tela de listagem dos lotes
2. Sistema exibe todas os lotes referentes ao usuário

3. Usuário escolhe um lote e seleciona a opção “Ver Informações”, acessando a tela de informações do lote
4. Sistema exibe informações gerais sobre o lote, como identificação, receita, observações, status, data de início, data de término, densidades relativas inicial e final/atual, estimativa de teor alcoólico, pH final/atual, temperatura final/atual.
5. Sistema exibe um gráfico com as variáveis monitoradas em relação ao tempo
6. Caso usuário clique em “Baixar Dados”, sistema efetua o download dos dados do gráfico em arquivo de texto

Requisitos relacionados: SW-F-2, SW-F-3, SW-F-8, SW-F-11

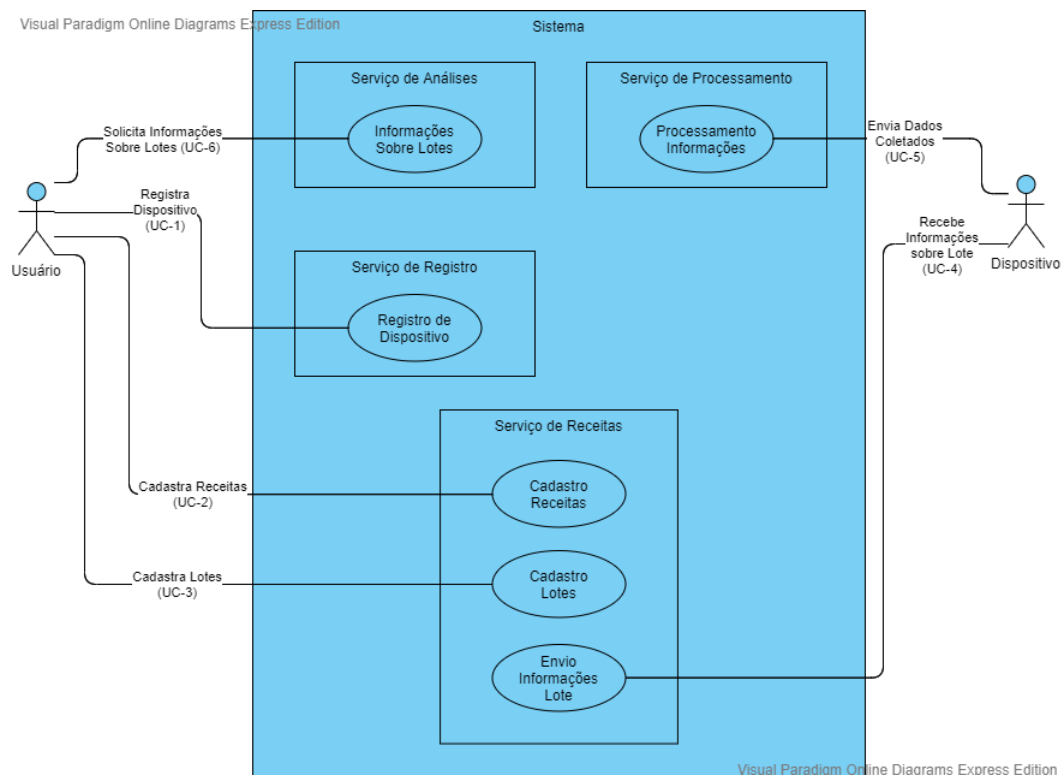


Figura 1: Diagrama de Casos de Uso da UML. Criado com a plataforma Visual Paradigm Online Diagrams Express Edition

4.4.2 Modelo de Entidade-Relacionamento

O Modelo Entidade Relacionamento descreve como as informações são organizada no sistema e como o banco de dados do sistema é estruturado. Cada entidade no modelo representa um tipo de informação que é produzida e consumida pelo sistema, na execução dos casos de uso.

Todas as informações são armazenadas em um banco de dados relacional, dividido em schemas que atuam como partições de dados em diferentes domínios. Foram definidos três domínios, traduzidos em schemas no banco de dados, para os dados: user, receita e control. O schema user contém as informações referentes aos usuários e dispositivos do sistema; o domínio receita contempla as entidades relacionadas às receitas e lotes de produção, assim como os dados coletados em cada execução de uma receita; e o schema control, por fim, os perfis de controle que são seguidos pelo dispositivo durante seu funcionamento. A separação das entidades em domínios é importante na arquitetura de microsserviços para assegurar que cada serviço tenha controle apenas às informações de sua competência.

A definição das entidades e seus relacionamentos é ilustrada pelo Diagrama Entidade Relacionamento da figura 2, com destaque em cor para cada um dos schemas determinados.

4.4.3 Arquitetura de Software

A arquitetura de software do projeto foi estruturada tendo como base o trajeto da informação especificado nos casos de uso e os conceitos gerais de arquitetura de microsserviços. Dessa forma, foram definidos quatro camadas: Camada de Interface, contemplando as interfaces utilizadas diretamente pelos atores; Camada Intermediária, contendo proxy reverso isolando os microsserviços e um message broker para intermediar a comunicação entre dispositivos e sistema; Camada de Microsserviços, contendo todos os microsserviços; e Camada de Banco de Dados, com os serviços de armazenamento de dados. Em sequência, os componentes de cada camada são explicitados e descritos quanto a suas responsabilidades e detalhes de implementação. A arquitetura é representada visualmente pelo Diagrama de Componentes da Figura X3.

Camada de Interface

Aplicação Front End - Implementação: HTML + CSS + React.js - Responsabilidades: disponibilizar interfaces com o usuário para gerenciamento e visualização de suas informações

Aplicativo em SmartPhone - Implementação: aplicativo em Java para Android - Responsabilidades: envio de configurações iniciais ao dispositivo por bluetooth e registro do dispositivo no sistema

Dispositivo - Implementação: linguagem Arduino - Responsabilidades: monitora-

mento e controle do processo de fermentação

Camada Intermediária

A camada intermediária acomoda um serviço de API Gateway, que age como um proxy reverso que roteia as requisições externas ao sistema para quais microsserviços sejam necessários e retorna ao cliente; e um serviço de Message Broker que controla o fluxo de mensagens que são transmitidas entre o dispositivo e o sistema.

API Gateway - Implementação: aplicação flask (Python), servido com Gunicorn e Nginx - Responsabilidades: Centralizar as requisições ao sistema e traduzi-las em requisições ao microsserviços necessários para o completar a ação solicitada

Message Broker - Implementação: Mosquitto (Protocolo MQTT) - Responsabilidades: Receber e disponibilizar mensagens em tópicos

Camada de Microsserviços

4.4.4 Tecnologias escolhidas

4.4.5 Protótipos das Interfaces

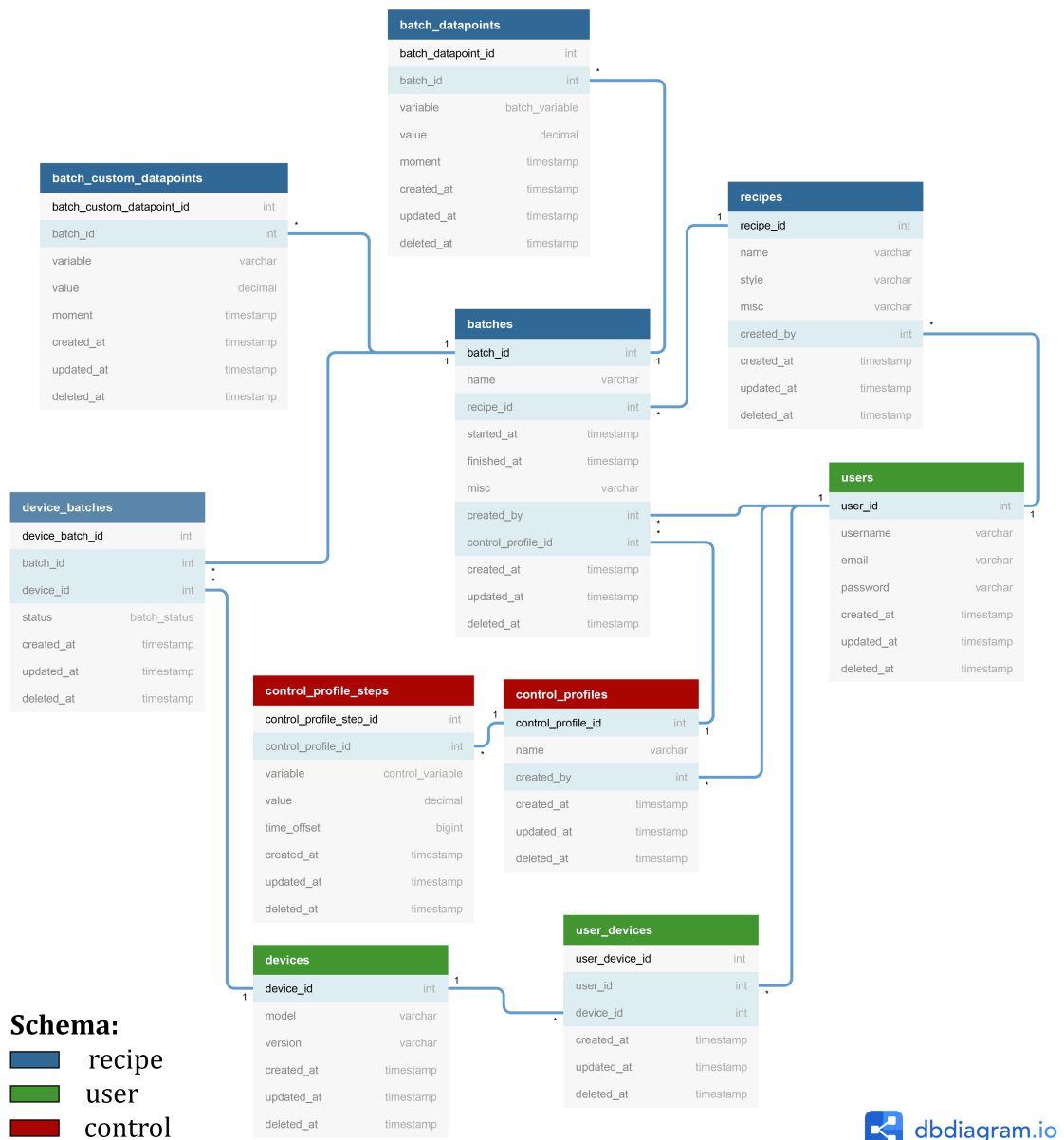


Figura 2: Diagrama de Entidade Relacionamento.

5 IMPLEMENTAÇÃO

6 TESTES E AVALIAÇÃO DO PROTÓTIPO

7 CONSIDERAÇÕES FINAIS

7.1 Conclusões do Projeto de Formatura

7.2 Contribuições

7.3 Perspectivas de Continuidade

APÊNDICE A

APÊNDICE B – BETA

ANEXO A – ALPHA

ANEXO B