



Instituto Federal do Maranhão
CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

Robson Luan do Nascimento de Sousa

**Desenvolvimento dirigido por comportamento aplicado em um
sistema simulado de travamento de portas veicular**

Santa Inês - MA
2025

Instituto Federal do Maranhão
CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

Robson Luan do Nascimento de Sousa

**Desenvolvimento dirigido por comportamento aplicado em um
sistema simulado de travamento de portas veicular**

Trabalho apresentado ao Instituto Federal
do Maranhão, Campus Santa Inês, como
requisito da obtenção do título de Bacharel
em Engenharia de Computação.

Orientador: Prof. Emanuel Cleyton Macedo
Lemos, Msc.

Coorientador: Prof. Aristoteles de Almeida
Lacerda Neto, Dr.

Santa Inês - MA

2025

ESTA SERÁ SUA FICHA CATALOGRÁFICA

ESTA SERÁ SUA FOLHA DE APROVAÇÃO

AGRADECIMENTOS

Agradeço o IFMA e ao curso de Engenharia da Computação, pela formação acadêmica e pelas oportunidades oferecidas ao longo da graduação.

Aos professores Msc. Emanuel Cleyton Macedo Lemos e Dr. Aristoteles de Almeida Lacerda Neto, pela orientação e por toda a ajuda durante a minha trajetória de aprendizado

Se pude enxergar a tão grande distância, foi subindo nos ombros de gigantes.

Isaac Newton
Carta à Robert Hooke, 1676

RESUMO

O presente trabalho tem como objetivo explorar a aplicação da metodologia de Desenvolvimento Orientado por Comportamento (BDD) no contexto de software embarcado, utilizando o mapeamento de exemplos e a linguagem Gherkin para definição de funcionalidades, geração de testes de aceitação automatizados e modelagem do sistema. Serão apresentados os principais benefícios decorrentes do uso dessa abordagem, como a ampla cobertura de testes, e a geração de documentação viva que valida o comportamento do sistema contra os resultados esperados. A metodologia será aplicada no desenvolvimento de um protótipo que simula um sistema automotivo de travamento de portas, atendendo a todas as especificações de comportamento previamente definidas.

Palavras-chave: *BDD. Gherkin. teste-automatizado. engenharia-automotiva. desenvolvimento-ágil.*

ABSTRACT

This study aims to explore the application of the Behavior Driven Development (BDD) methodology in the context of embedded software, using example mapping and the Gherkin language for defining functionalities, generating automated acceptance tests and modeling the system. The main benefits of this approach will be presented, such as extensive test coverage and the creation of living documentation that validates the system's behavior against the expected results. The methodology is going to be applied on the development of a prototype that simulates an automotive door locking system, fulfilling all previously defined behavioral specifications.

Key-words: *BDD. Gherkin. automated-tests. automotive-engineering. agile-development..*

LISTA DE ILUSTRAÇÕES

Figura 1 – História de Usuário 1: Travamento de todas as portas	18
Figura 2 – História de Usuário 2: Destravamento de todas as portas	19
Figura 3 – História de Usuário 3: Feedback de travamento	20
Figura 4 – História de Usuário 4: Keyless Access	21
Figura 5 – História de Usuário 5: Travamento automático	22

LISTA DE TABELAS

Tabela 1 – Tipos de aplicações e abordagens preferenciais.	20
Tabela 2 – Comandos seriais suportados pela estação meteorológica <i>Vantage VueTM</i>	29

LISTA DE QUADROS

Quadro 1 – Exemplo de matriz de confusão	21
Quadro 2 – Cronograma	21

LISTA DE CÓDIGOS

LISTA DE ABREVIATURAS E SIGLAS

LI	Lorem Ipsum
LII	Lorem Ipsum Ipsum

SUMÁRIO

1	INTRODUÇÃO	15
1.1	PROBLEMÁTICA	15
1.2	OBJETIVOS GERAIS E ESPECÍFICOS	15
1.3	METODOLOGIA	16
1.4	ORGANIZAÇÃO DO TRABALHO	17
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	SEÇÃO DE EXEMPLO 1 - COMO FAZER CITAÇÕES	18
2.2	SEÇÃO DE EXEMPLO 2 - COMO INSERIR FIGURAS	18
2.3	SEÇÃO DE EXEMPLO 3 - SOBRE TABELAS	19
2.4	SUBSEÇÃO DE EXEMPLO 4 - SEÇÕES	20
3	MATERIAIS E MÉTODOS	23
3.1	SEÇÃO DE EXEMPLO 1	23
3.1.1	Subseção de exemplo 1 - Referenciando seções	23
4	RESULTADOS	24
4.1	SEÇÃO DE EXEMPLO 1 - CÓDIGOS	24
4.1.1	Subseção de exemplo 1 - Inserindo trechos de códigos	24
4.2	SEÇÃO DE EXEMPLO 2 - LISTAS	24
4.2.1	Subseção de exemplo 2 - Lista de itens	24
4.2.1.1	Subsubseção de exemplo 1 - Lista sem numeração	24
4.2.1.2	Subsubseção de exemplo 2 - Lista enumerada	24
4.2.1.3	Subsubseção de exemplo 3 - Lista mista	25
5	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	27
5.1	TRABALHOS FUTUROS	27
	REFERÊNCIAS	28
ANEXO A	COMANDOS SERIAIS DA ESTAÇÃO METEOROLÓGICA <i>VANTAGE VUE™</i>	29

1 INTRODUÇÃO

A indústria automotiva tem experimentado um crescimento constante, acompanhado por um aumento significativo na modernização dos veículos e pela intensa competição entre montadoras para entregar produtos cada vez mais robustos e de alta qualidade. Esse cenário traz consigo um aumento na complexidade do desenvolvimento, demandando soluções mais eficientes e seguras para sistemas cada vez mais sofisticados e que sejam capazes de atender às necessidades específicas de cada cliente.

Diante desse contexto, a Engenharia Automotiva enfrenta o desafio de projetar e desenvolver produtos com um nível altíssimo de qualidade, ao mesmo tempo em que busca reduzir erros e retrabalho. A aplicação de metodologias ágeis, como o Behavior Driven Development (BDD) (NORTH, 2006), oferece uma abordagem promissora, pois foca na definição do comportamento esperado do sistema a partir da perspectiva do usuário. Essa abordagem permite uma colaboração mais efetiva entre os membros de uma equipe ágil (Atlassian, n.d.), facilita a descoberta de funcionalidades e possibilita que o desenvolvimento seja guiado por testes de aceitação automatizados, promovendo entregas de valor de forma mais rápida e com maior confiabilidade.

1.1 PROBLEMÁTICA

O problema central deste trabalho é: “como é possível aplicar o processo de BDD para facilitar a definição do sistema e guiar seu desenvolvimento no contexto da Engenharia de Sistemas Automotivos?”. O BDD incorpora diversas premissas que surgem das lições aprendidas na aplicação de outras metodologias ágeis, definindo o desenvolvimento do produto a partir do comportamento que gera valor ao usuário. Além disso, ele busca promover a colaboração entre as diferentes etapas do desenvolvimento, garantindo que toda a equipe compreenda o produto e consiga assumir a perspectiva do usuário. Por outro lado, a Engenharia Automotiva apresenta uma complexidade significativa, o que torna desafiadora a entrega de produtos de alta qualidade. A definição de um sistema veicular envolve muito mais do que software ou eletrônica (Bosch, 2022), sendo necessário integrar componentes mecânicos, considerar falhas potenciais e contemplar casos de uso extremos. O principal desafio deste trabalho está em identificar técnicas capazes de capturar essas complexidades e torná-las compatíveis com as premissas do BDD.

1.2 OBJETIVOS GERAIS E ESPECÍFICOS

Para responder à problemática levantada, o presente estudo tem como objetivo geral aplicar o levantamento e definição de funcionalidades na análise de um sistema

de travamento de portas veicular, seguido do desenvolvimento guiado por testes de aceitação automatizados, demonstrando como essas práticas podem contribuir para o desenvolvimento de software automotivo de forma mais organizada e eficiente. Para isso, os objetivos específicos serão focados em:

- Abordar as complexidades dos sistemas automotivos de maneira compatível com o processo de BDD;
- Empregar técnicas de desenvolvimento que são amplamente utilizadas na indústria automotiva;
- Realizar o levantamento e definição de funcionalidades considerando possíveis limitações decorrentes da dependência de hardware e de sistemas mecânicos;
- Definir um escopo de desenvolvimento que possibilite a realização do design do sistema forma clara e compreensível;
- Empregar testes de aceitação automatizados para possibilitar o desenvolvimento do sistema de maneira iterativa, consistente com o design definido;
- Possibilitar a adaptação às descobertas surgidas durante a aplicação do processo, garantindo a melhoria contínua do design do sistema e do valor entregue ao usuário.

1.3 METODOLOGIA

A metodologia utilizada neste estudo baseia-se na revisão da bibliografia disponível, com o objetivo de esclarecer o processo de BDD e investigar como ele pode ser adaptado para incorporar práticas consolidadas na Engenharia Automotiva. Para tanto, serão aplicadas técnicas como o mapeamento de exemplos (GOSLING; JOHNSON, 2014; Cucumber.io, n.d.) , voltado ao levantamento e definição de funcionalidades, e testes de aceitação automatizados (A... ,) , que servirão para guiar o desenvolvimento do sistema. A implementação do sistema simulado também fará uso de técnicas típicas da Engenharia Automotiva, incluindo o Model Based Design (SANGIOVANNI-VINCENTELLI; MARTIN, 2001; MathWorks, 2021) e a geração automática de código segundo o padrão de software AUTOSAR (AUTOSAR,). Conceitos adicionais do domínio automotivo serão considerados para assegurar a adequada abstração nas diferentes etapas do desenvolvimento. Espera-se que os resultados obtidos evidenciam os benefícios da integração do BDD no contexto automotivo, destacando sua contribuição para a qualidade e confiabilidade do sistema. A validação será realizada por meio da execução dos testes de aceitação e da análise do uso do sistema final sob a perspectiva do usuário.

1.4 ORGANIZAÇÃO DO TRABALHO

Este trabalho está estruturado da seguinte forma: o capítulo 2 apresenta a fundamentação teórica sobre BDD e testes automatizados, o capítulo 3 descreve a metodologia aplicada, o capítulo 4 detalha a implementação e os resultados obtidos, e o capítulo 5 apresenta as conclusões e considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 SEÇÃO DE EXEMPLO 1 - COMO FAZER CITAÇÕES

Existem vários tipos de citações...

2.2 SEÇÃO DE EXEMPLO 2 - COMO INSERIR FIGURAS

Neste trabalho iremos exemplificar duas formas de se inserir figuras no Latex. O primeiro método insere, no documento, uma figura simples por meio do comando:

```
\imagem{ Escala }{ Arquivo sem extensão }{ Descrição }{ Fonte }
```

Obs.: A fonte pode ser uma citação do tipo `\citeonline{}`.

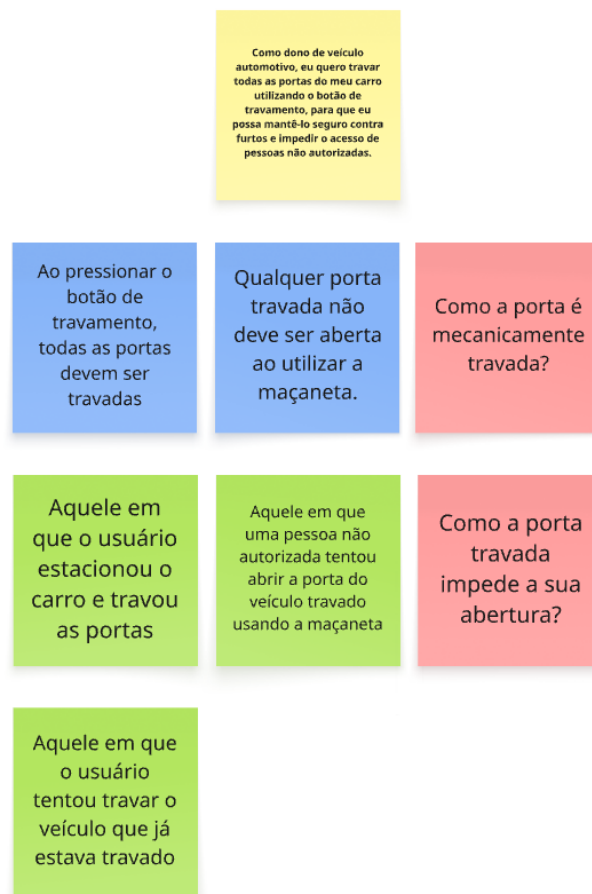


Figura 1 – História de Usuário 1: Travamento de todas as portas

Para referenciar uma figura deve ser usada comando `\ref{img:<label ou nome do arquivo>}`, como exemplo, estamos referenciando a figura `??`. Isso vale tanto para figuras simples quanto para as compostas, como por exemplo as figuras `??` e `??`. Ao inserir uma

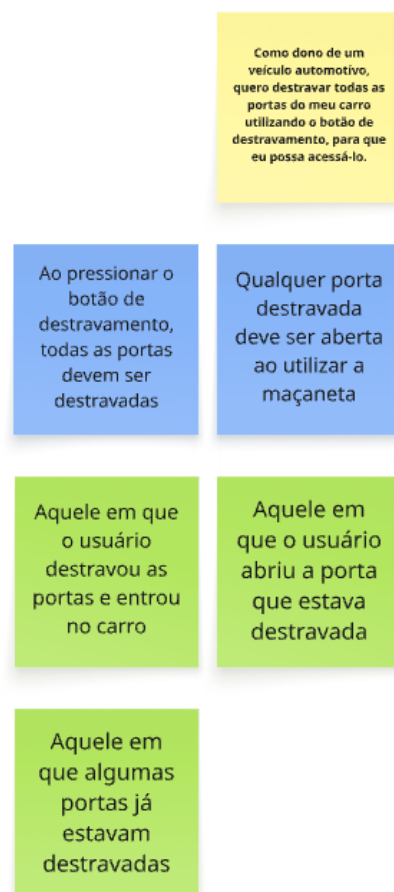


Figura 2 – História de Usuário 2: Destravamento de todas as portas

figura, ela é automaticamente identificada e incluída no elemento pré-textual da lista de figuras.

2.3 SEÇÃO DE EXEMPLO 3 - SOBRE TABELAS

As tabelas em Latex são deveras capciosas, por isso não serão abordadas em sua completude neste documento.

Há um site que possui uma ferramenta interessante para ser utilizada na construção tabelas em Latex.

O Tables Generator <– Isto é um *link* :D

Contudo, busquem entendimento sobre o assunto, pois tabelas são elementos textuais importantes e enriquecem muito o texto, quando bem construídas.

A tabela 1 é um exemplo de como uma tabela pode ser construída, assim como a tabela do anexo A.

Também é possível criar quadros, que são ligeiramente diferente de tabelas. Acompanhe o exemplo no Quadro 1

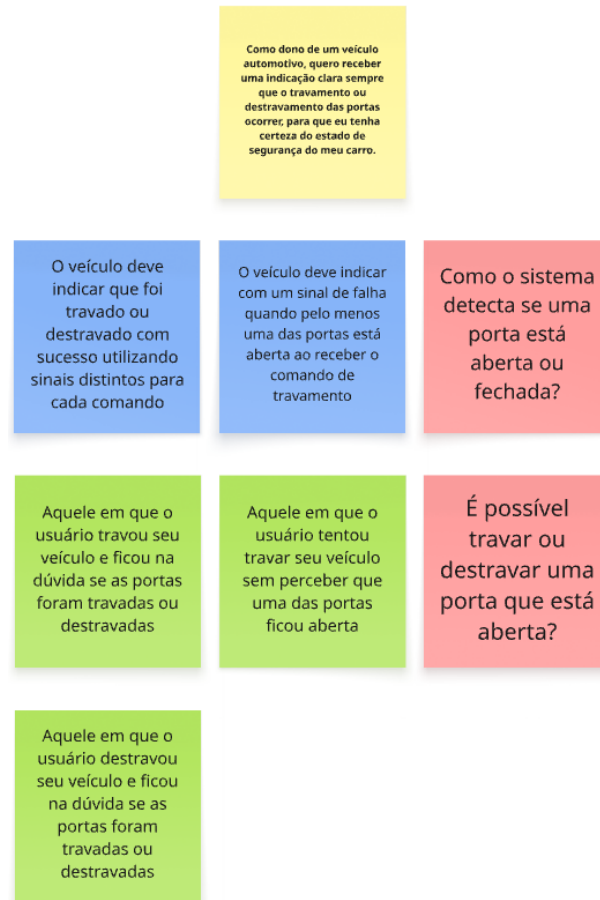


Figura 3 – História de Usuário 3: Feedback de travamento

Tabela 1 – Tipos de aplicações e abordagens preferenciais.

Código da Aplicação	Web	Híbrida	Interpretada / Compilação Cruzada
Aplicações baseadas em dados providos por um servidor	3	2	1
Aplicações independentes	1	2	3
Aplicações baseadas em sensores e processamento de dados no dispositivo	1	2	3
Aplicações baseadas em sensores e processamento de dados no servidor	1	3	2
Aplicações Cliente-Servidor	1	3	2

Fonte: ??) (Traduzido)

2.4 SUBSEÇÃO DE EXEMPLO 4 - SEÇÕES

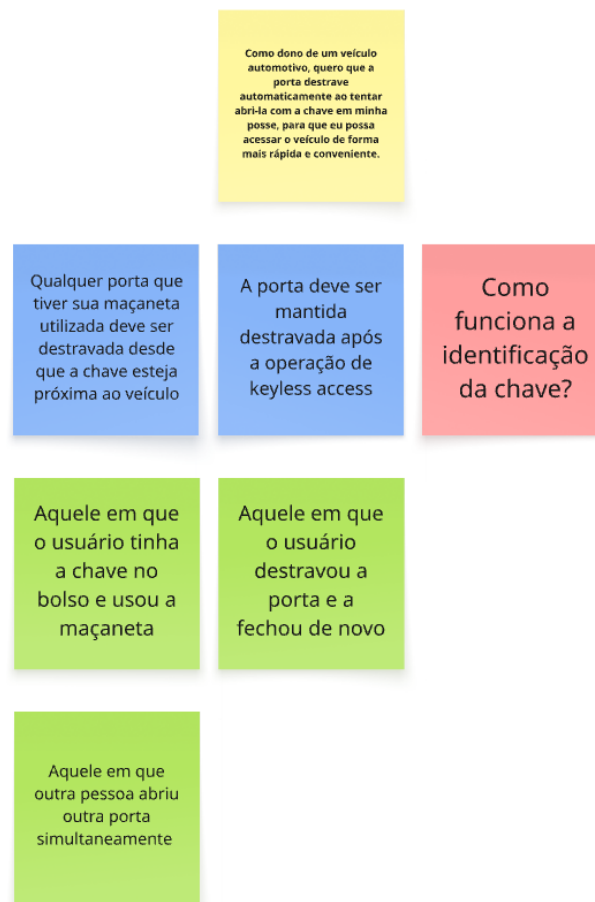


Figura 4 – História de Usuário 4: Keyless Access

Quadro 1 – Exemplo de matriz de confusão

		Classe prevista	
		Classe = 1	Classe = 0
Classe real	Classe = 1	f_{11}	f_{10}
	Classe = 0	f_{01}	f_{00}

Fonte: O autor

Quadro 2 – Cronograma

Atividade	Set/19	Out/19	Nov/19	Dez/19	Jan/20
1	x				
2		x			
3		x			
4			x	x	
5		x	x	x	
6					x

Fonte: Elaborado pela autora (2019)

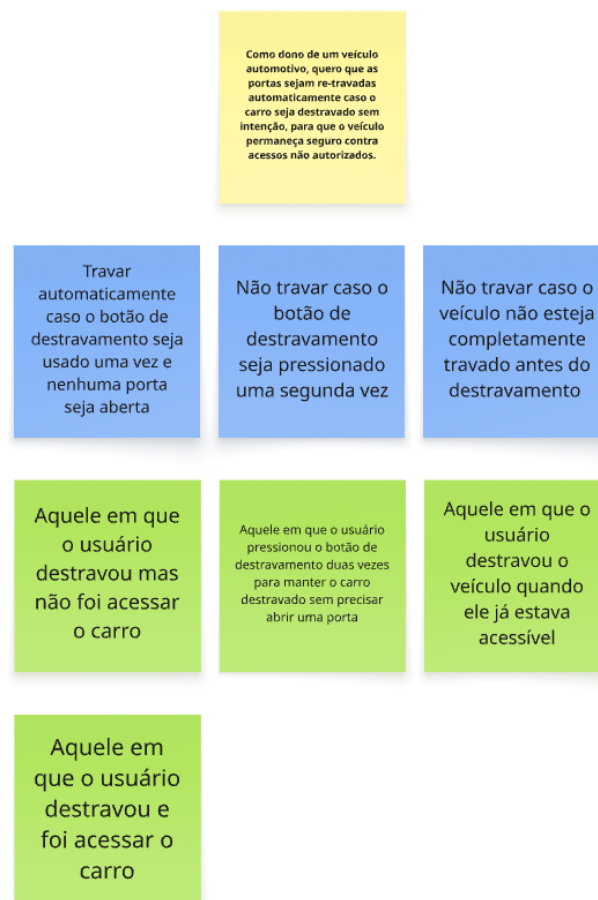


Figura 5 – História de Usuário 5: Travamento automático

3 MATERIAIS E MÉTODOS

3.1 SEÇÃO DE EXEMPLO 1

3.1.1 Subseção de exemplo 1 - Referenciando seções

4 RESULTADOS

4.1 SEÇÃO DE EXEMPLO 1 - CÓDIGOS

4.1.1 Subseção de exemplo 1 - Inserindo trechos de códigos

O nosso querido Leonardo Cavalcante providenciou um comando que deixa nossos trechos de códigos bonitinhos e gera um elemento pré-textual de Lista de Códigos.

Os códigos são adicionados através do comando seguinte:

```
\sourcecode{ Descrição }{Label}{Linguagem}{Arquivo com extensão}
```

Um exemplo pode ser visto no código ?? abaixo.

4.2 SEÇÃO DE EXEMPLO 2 - LISTAS

4.2.1 Subseção de exemplo 2 - Lista de itens

Existem alguns tipos de listas no Latex, iremos exemplificar a lista sem numeração (seção 4.2.1.1), a lista enumerada (seção 4.2.1.2) e a lista mista (seção 4.2.1.3). As listas podem ser encadeadas de diversas maneiras, de acordo com a necessidade do autor.

4.2.1.1 Subsubseção de exemplo 1 - Lista sem numeração

Este é um exemplo de lista sem numeração.

- **Cadastrar usuário**

- Atores
 - * Usuário
- Fluxo de eventos primário
 - * o usuário deve se cadastrar informando seu nome, *e-mail* e senha;
 - * a API armazena os dados do usuário;
 - * o usuário é liberado para realizar o *login*.
- Fluxo alternativo
 - * o usuário desiste de se cadastrar e cancela o caso de uso clicando no botão voltar.

4.2.1.2 Subsubseção de exemplo 2 - Lista enumerada

Este é um exemplo de lista enumerada.

1. O Usuário deseja ver o histórico das variáveis climáticas, então através da interface de usuário escolhe o período ao qual o histórico se refere;
2. A aplicação solicita à API através de uma requisição HTTP contendo o momento de início e o momento do fim do período em seus parâmetros;
3. A API recebe a solicitação e se comunica com a base de dados, então requiere as informações quem possuem a data de leitura no intervalo escolhido;
4. A base de dados retorna os dados em formato Json para a API;
5. A API responde à requisição retornando os dados, também em formato Json, para a aplicação cliente;
6. A aplicação cliente renderiza os gráficos utilizando o conjunto de dados obtidos.

4.2.1.3 Subsubseção de exemplo 3 - Lista mista

Este é um exemplo de lista mista.

- **Cadastrar usuário**

- Atores
 - * Usuário
- Fluxo de eventos primário
 1. o usuário deve se cadastrar informando seu nome, *e-mail* e senha;
 2. a API armazena os dados do usuário;
 3. o usuário é liberado para realizar o *login*.
- Fluxo alternativo
 - * o usuário desiste de se cadastrar e cancela o caso de uso clicando no botão voltar.

- **Visualizar dados atuais**

- Atores
 - * Usuário
- Pré-condições
 - * o usuário deve estar autenticado
- Fluxo de eventos primário
 1. o usuário deve efetuar o *login* informando o *e-mail* e a senha;

2. caso o usuário não seja autenticado, o sistema informa a respeito de credenciais inválidas e encerra o caso de uso;
3. a API autentica o usuário;
4. o usuário é liberado para visualizar os dados atuais dos sensores da estação;
5. após a visualização o usuário pode finalizar o caso de uso ou efetuar uma nova consulta se desejar.

– Fluxo alternativo

- * o usuário desiste de visualizar os dados atuais e cancela o caso de uso clicando no botão voltar.

• **Visualizar histórico**

– Atores

- * Usuário

– Pré-condições

- * o usuário deve estar autenticado

– Fluxo de eventos primário

1. o usuário deve efetuar o *login* informando o *e-mail* e a senha;
2. caso o usuário não seja autenticado, o sistema informa a respeito de credenciais inválidas e encerra o caso de uso;
3. a API autentica o usuário;
4. o usuário é liberado para escolher qual período cujo histórico será exibido;
5. o usuário seleciona as variáveis a serem exibidas no gráficos de linhas;
6. após a visualização do histórico o usuário pode finalizar o caso de uso se desejar.

– Fluxo alternativo

1. após a escolha do período de exibição do histórico o usuário pode voltar para a tela anterior e escolher um novo período;
2. o histórico é exibido para o usuário;
3. após a visualização do histórico o usuário pode finalizar o caso de uso ou efetuar uma nova consulta se desejar.

– Fluxo alternativo

1. o usuário desiste de visualizar o histórico e cancela o caso de uso clicando no botão voltar.

5 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

5.1 TRABALHOS FUTUROS

Nunc velit. Nullam elit sapien, eleifend eu, commodo nec, semper sit amet, elit. Nulla lectus risus, condimentum ut, laoreet eget, viverra nec, odio. Proin lobortis. Curabitur dictum arcu vel wisi. Cras id nulla venenatis tortor congue ultrices. Pellentesque eget pede. Sed eleifend sagittis elit. Nam sed tellus sit amet lectus ullamcorper tristique. Mauris enim sem, tristique eu, accumsan at, scelerisque vulputate, neque. Quisque lacus. Donec et ipsum sit amet elit nonummy aliquet. Sed viverra nisl at sem. Nam diam. Mauris ut dolor. Curabitur ornare tortor cursus velit.

REFERÊNCIAS

- A Study of the Characteristics of Behaviour Driven Development. Citado na página 16.
- Atlassian. **Agile Teams**. n.d. Acesso em: 12 jul. 2025. Disponível em: <<https://www.atlassian.com/agile/teams>>. Citado na página 15.
- AUTOSAR. **Classic Platform Standards**. Disponível em: <<https://www.autosar.org/standards/classic-platform>>. Citado na página 16.
- Bosch. **Automotive Handbook**. 9. ed. [S.l.: s.n.], 2022. Citado na página 15.
- Cucumber.io. **Mapeamento de exemplos**. n.d. Disponível em: <<https://cucumber.io/blog/bdd/example-mapping-introduction/>>. Citado na página 16.
- GOSLING, A.; JOHNSON, B. **Behavior-Driven Development with Cucumber: Better collaboration for better software**. [S.l.]: Packt Publishing, 2014. Citado na página 16.
- MathWorks. **Model-Based Design for Embedded Systems**. 2021. Citado na página 16.
- NORTH, D. **Introducing Behavior-Driven Development**. 2006. Acesso em: 12 jul. 2025. Disponível em: <<https://dannorth.net/introducing-bdd/>>. Citado na página 15.
- SANGIOVANNI-VINCENNELLI, A.; MARTIN, G. Platform-based design and software design methodology for embedded systems. 2001. Citado na página 16.

ANEXO A – COMANDOS SERIAIS DA ESTAÇÃO METEOROLÓGICA VANTAGE VUE™

Tabela 2 – Comandos seriais suportados pela estação meteorológica *Vantage Vue™*

Instrução	Descrição
Comandos de teste	
TESTE	Envia a <i>string</i> "TEST\n" de volta
WRD	Responde com o tipo de estação meteorológica
RXCHECK	Responde com o diagnóstico do Console
RXTEST	Muda a tela do console de " <i>Receiving from</i> " para tela de dados atuais
VER	Responde com a data do <i>firmware</i>
RECEIVERS	Responde com a lista das estações que o console "enxerga"
NVER	Responde com a versão do <i>firmware</i>
Comandos de dados atuais	
LOOP	Responde com a quantidade de pacotes especificada a cada 2s
LPS	Responde a cada 2s com a quantidade de pacotes diferentes especificada
HILOWS	Responde com todo os dados de <i>high/low</i>
PUTRAIN	Seta a quantidade anual de precipitação
PUTET	Seta a quantidade anual de evapotranspiração
Comandos de <i>download</i>	
DMP	Faz o <i>download</i> de todo o arquivo de memória
DMAFT	Faz o <i>download</i> de todo o arquivo de memória após a data especificada
Comandos da EEPROM	
GETEE	Lê toda a memória EEPROM
EEWR	Escreve um <i>byte</i> de dados à partir do endereço especificado
EERD	Lê a quantidade de dados especificada iniciando no endereço especificado
EEBWR	Escreve os dados na EEPROM
EEBRD	Lê os dados da EEPROM
Comandos de calibração	
CALED	Envia os dados da temperatura e umidade corrente para atribuir à calibração
CALFIX	Atualiza o <i>display</i> quando os números de calibração mudam
BAR	Seta os valores da elevação e o <i>offset</i> do barômetro quando a localização é alterada
BARDATA	Mostra os valores atuais da calibração do barômetro

Continua na próxima página

Tabela 2 – Continuação da página anterior

Instrução	Descrição
Comandos de limpeza	
CLRLOG	Limpa todo o arquivo de dados
CLRALM	Limpa todos os limiares dos alarmes
CLRCAL	Limpa todos os <i>offsets</i> da calibração da temperatura e da umidade
CLRGRA	Limpa o gráfico do console
CLRVAR	Limpa o valor da precipitação ou da evapotranspiração
CLRHIGHS	Limpa todos os valores de pico diários, mensais ou anuais
CLRLOWS	Limpa todos os valores de mínimos diários, mensais ou anuais
CLRBITS	Limpa os <i>bits</i> de alarme ativos
CLRDATA	Limpa todos os dados atuais
Comandos de configuração	
BAUD	Atribui o valor do <i>baudrate</i> do console
SETTIME	Define a data e a hora do console
GAIN	Define o ganho do receptor de rádio
GETTIME	Retorna a hora e a data atual do console
SETPER	Define o intervalo de arquivamento
STOP	Desabilita a criação dos registros
START	Habilita a criação dos arquivos
NEWSETUP	Reinicia o console após alguma configuração nova
LAMPS	Liga ou desliga as lâmpadas do console

Fonte: ??) (Traduzido).