**PROJETO DASHBOARD INTELLIGENT HOUSE**

**Requisitos da disciplina Modelagem de Software e Arquitetura de Sistemas**

**INTEGRANTES DO PROJETO e RA’S**

Victor Lopes Domingues - 23024580

Enzo Mancio Vaz - 25027739

Melissa Julia Lecona Lequipe - 23025611

São Paulo

2025

Sumário

[1 INTRODUÇÃO 3](#_Toc195214459)

[2. DOCUMENTO DE ABERTURA DO PROJETOS 4](#_Toc195214460)

[2.1 – Project Charter 4](#_Toc195214461)

[2.2 – Histórias do Usuário 5](#_Toc195214462)

[3. DESIGN SPRINT – Ideação e prototipação do desafio 6](#_Toc195214463)

[3.1 Desafio 6](#_Toc195214464)

[3.2 Entender Mapear 6](#_Toc195214465)

[3.3 Ideação – desenho da solução (trilha do usuário) 6](#_Toc195214466)

[3.4 Prototipagem 6](#_Toc195214467)

[4.REQUISITOS DE SISTEMA 6](#_Toc195214468)

[4.1 REQUISITOS FUNCIONAIS DE SOFTWARE 6](#_Toc195214469)

[4.2 REQUISITOS NÃO FUNCIONAIS DE SOFTWARE 7](#_Toc195214470)

[5. CASOS DE USO 8](#_Toc195214471)

[6. DIAGRAMA DE CLASSE 8](#_Toc195214472)

[7. ARQUITETURA DO SISTEMA 8](#_Toc195214473)

[8. REFERÊNCIAS BIBLIOGRÁFICAS 8](#_Toc195214474)

# 1 INTRODUÇÃO

Tendo.......

Incluir o documento do Projeto do PI – Problemas de Smart Cities

**Smart Cities/Smart House**

**Nome da Instituição**: Flex Automation

**Objetivo da Aplicação:**

ADS1 O objetivo do desafio é gerar um dashboard de uma cidade/casa inteligente que permita o controle de sensores e atuadores.

Este desafio busca, de forma modular, introduzir como uma cidade/casa inteligente pode ser controlada, tratando seus dados de forma a aprimorar o sistema e otimizando a sustentabilidade.

Seu dashboard deverá receber e enviar sinais de/para um simulador de casa/cidade inteligente, provenientes da rede/internet. O servidor será fornecido pelos professores.

**Desafio:**

O projeto da Flex Automation, assim como outras iniciativas, trabalha para poder criar cidades inteligentes buscando a sustentabilidade, o melhor uso dos recursos planetários e o menor impacto na natureza. Para que isso ocorra é necessário ter uma alta capacidade de mensuração e controle para a otimização da vida na cidade, desde recursos até o tráfego de pedestres. Também, a conscientização da população de como uma cidade inteligente funciona e/ou é controlada, de forma a instruir sobre as melhores maneiras para a cidade a ser sustentável.

**Personas a Serem Atendidas**:

-**Usuário final** do sistema, que deseja controlar sua casa de forma a gastar menos e otimizar os recursos da cidade. Considere que o usuário possui conhecimento básico para utilizar dispositivos mobile.

-**Controlador da cidade**, um funcionário da cidade que deve acompanhar um dashboard/mapa/painel informativo da cidade, tratando situações inesperadas, acompanhando os dados dos sensores e acionando os programas da cidade. Considere que ele tem um conhecimento médio para avançado de tecnologia.

**Recursos:**

https://store.steampowered.com/app/949230/Cities\_Skylines\_II/ https://store.steampowered.com/app/2741560/SimCity\_3000\_Unlimited/ https://planetsmartcity.com/ https://flexautomation.com.br

# 2. DOCUMENTO DE ABERTURA DO PROJETOS

## 2.1 – Project Charter

**Prefácio**

Deve definir os possíveis leitores do documento e descrever seu histórico de versões, incluindo uma justificativa para a criação de uma nova versão e um resumo das mudanças feitas em cada versão.

**Introdução**

Deve descrever a necessidade para o sistema. Deve descrever brevemente as funções do sistema e explicar como ele vai funcionar com outros sistemas. Também deve descrever como o sistema atende aos objetivos globais de negócio ou estratégicos da organização que encomendou o software.

**Glossário**

Deve definir os termos técnicos usados no documento. Você não deve fazer suposições sobre a experiência ou o conhecimento do leitor.

**Definição de requisitos de usuário**

Deve descrever os serviços fornecidos ao usuário. Os requisitos não funcionais de sistema também devem ser descritos nessa seção. Essa descrição pode usar a linguagem natural, diagramas ou outras notações compreensíveis para os clientes. Normas de produto e processos que devem ser seguidos devem ser especificados.

**Arquitetura do sistema**

Deve apresentar uma visão geral em alto nível da arquitetura do sistema previsto, mostrando a distribuição de funções entre os módulos do sistema. Componentes de arquitetura que são reusados devem ser destacados.

**Especificação de requisitos do sistema**

Deve descrever em detalhes os requisitos funcionais e não funcionais. Se necessário, também podem ser adicionados mais detalhes aos requisitos não funcionais. Interfaces com outros sistemas podem ser definidas.

**Modelos do sistema**

Pode incluir modelos gráficos do sistema que mostram os relacionamentos entre os componentes do sistema, o sistema e seu ambiente. Exemplos de possíveis modelos são modelos de objetos, modelos de fluxo de dados ou modelos semânticos de dados.

**Evolução do sistema**

Deve descrever os pressupostos fundamentais em que o sistema se baseia, bem como quaisquer mudanças previstas, em decorrência da evolução de hardware, de mudanças nas necessidades do usuário etc. Essa seção é útil para projetistas de sistema, pois pode ajudá-los a evitar decisões capazes de restringir possíveis mudanças futuras no sistema.

**Apêndices**

Deve fornecer informações detalhadas e específicas relacionadas à aplicação em desenvolvimento, além de descrições de hardware e banco de dados, por exemplo. Os requisitos de hardware definem as configurações mínimas ideais para o sistema. Requisitos de banco de dados definem a organização lógica dos dados usados pelo sistema e os relacionamentos entre esses dados.

## 2.2 – Histórias do Usuário

Alguns detalhes sobre a casa inteligente que cujos dados estão no arquivo anexo:  
  
-2 Pessoas vivem nesta casa

-A casa possuí 2 quartos, 1 sala, 1 cozinha e 1 piscina e são identificados respectivamente pelos sensores de ID: 1, 2, 3, 4, 5.

-O gasto energético médio para deixar cada local ligado é:

Quartos (ID 1 e 2) – 1,5KWatts/Hora (Considerando 1 TV,1 lâmpada e um ar-condicionado)

Sala (ID 3) – 50Watts/Hora (Considerando 1 TV e 5 lâmpadas)

Cozinha (ID 4) – 3KWatts/Hora (Considerando 1 Micro-ondas, 1 máquina de lavar louça e 3 lâmpadas)

Piscina (ID 5) – 7KWatts/Hora (Bomba + Aquecedor)

Você tem a possibilidade de adicionar comandos separados para controlar cada um dos elementos descritos acima.

**EXEMPLO DA BASE DOS SENSORES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TimeStamp | ID\_Sensor | Temperatura | Umidade | Movimento |
| 28/4/25 0:18 | 3 | 39 | 71 | 0 |
| 22/5/25 4:43 | 4 | 19 | 82 | 0 |
| 20/4/25 20:38 | 3 | 24 | 71 | 0 |
| 12/2/25 0:03 | 1 | 22 | 22 | 0 |
| 14/4/25 1:33 | 2 | 19 | 46 | 1 |
| 27/1/25 14:21 | 2 | 37 | 27 | 0 |
| 30/5/25 7:19 | 1 | 10 | 87 | 0 |
| 21/7/25 6:17 | 1 | 34 | 88 | 0 |
| 21/1/25 9:20 | 3 | 39 | 28 | 0 |
| 2/2/25 23:55 | 4 | 28 | 33 | 0 |
| 22/6/25 14:15 | 3 | 17 | 32 | 0 |
| 24/6/25 15:22 | 2 | 38 | 29 | 0 |
| 30/4/25 0:32 | 2 | 18 | 88 | 1 |
| 26/6/25 2:00 | 2 | 26 | 63 | 0 |
| 26/6/25 10:09 | 2 | 21 | 50 | 0 |
| 1/3/25 7:15 | 5 | 40 | 30 | 1 |
| 27/6/25 7:02 | 3 | 15 | 28 | 1 |

# 3. DESIGN SPRINT – Ideação e prototipação do desafio

## 3.1 Desafio

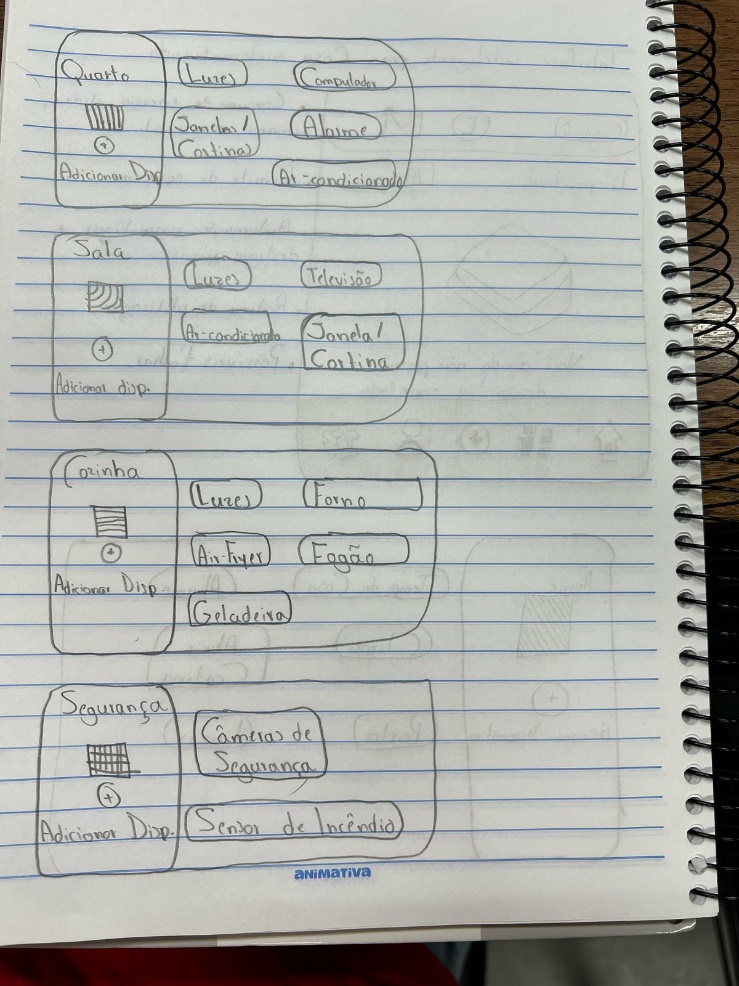
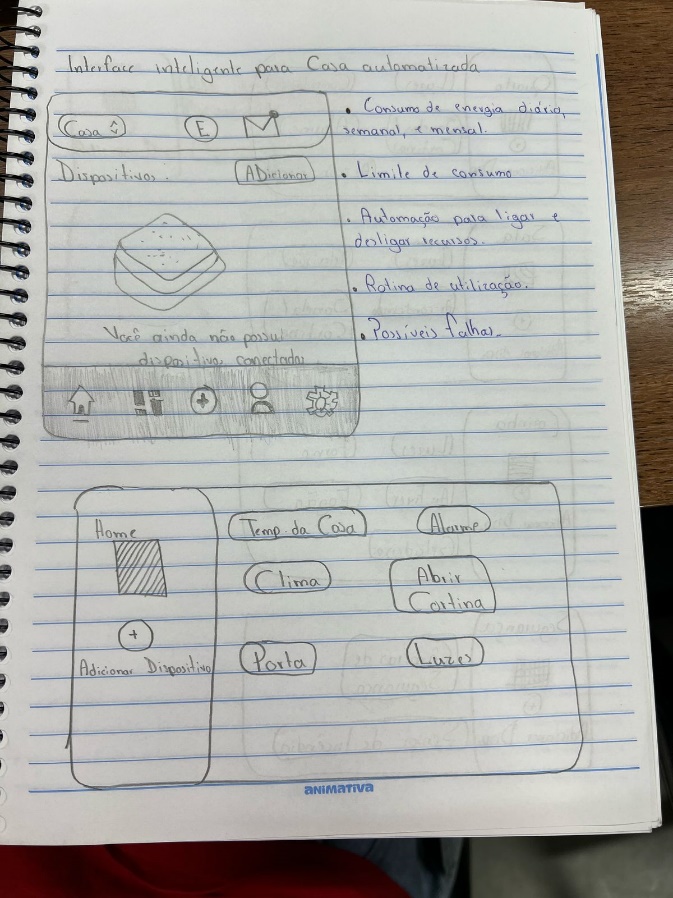
Criar Dashboard onde o usuário tenha uma visualização de fácil e simples, onde demostre os gasto de energia em KWh.

Dar dicas de economia de energia cada item que sesta sendo monitorado.

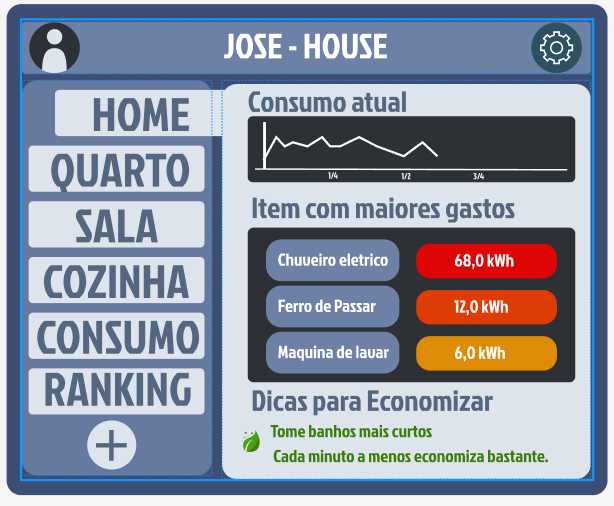
Rankear todos os usuários com base em suas economias mensais

## 3.2 Entender Mapear

## 3.3 Ideação – desenho da solução (trilha do usuário)



## 3.4 Prototipagem



# 4.REQUISITOS DE SISTEMA

## 4.1 REQUISITOS FUNCIONAIS DE SOFTWARE

Necessários 6 requisitos

## 1 Controle de Dispositivos Conectados O sistema deve permitir que o usuário ligue, desligue ou ajuste dispositivos inteligentes (como luzes, ar-condicionado, tomadas, etc.) diretamente pelo dashboard.

## Monitoramento em Tempo Real O sistema deve exibir informações em tempo real, como temperatura ambiente, consumo de energia, status de portas e janelas, ou presença de pessoas.

## Automação de Rotinas O sistema deve possibilitar a criação e gerenciamento de rotinas automatizadas (ex: “desligar todas as luzes às 23h” ou “ligar o ar-condicionado quando a temperatura passar de 28°C”).

## Gerenciamento de Perfis de Usuário O sistema deve permitir múltiplos perfis de usuários com diferentes níveis de permissão (por exemplo, administrador, visitante, criança).

## Histórico de Acessos O sistema deve manter um registro das atividades dos dispositivos conectados, como quando foram ligados/desligados, quando tiveram o último checkup de energia, etc.

## Notificações e Alertas O sistema deve enviar notificações (push, e-mail ou SMS) em caso de eventos importantes, como detecção de fumaça, movimento inesperado ou consumo anormal de energia.

## 4.2 REQUISITOS NÃO FUNCIONAIS DE SOFTWARE

Necessários 6 requisitos

**Segurança**

O sistema deve garantir a proteção contra acessos não autorizados, utilizando autenticação forte (ex: biometria, senha, autenticação de dois fatores).

**Manutenibilidade**

O software deve ser fácil de manter e atualizar, permitindo correções de bugs, melhorias e atualizações de segurança sem impactar negativamente o funcionamento da casa.

**Disponibilidade**

O sistema deve estar disponível 99,9% do tempo, com tolerância a falhas e funcionamento contínuo mesmo durante quedas de energia (com backup ou energia reserva).

**Escalabilidade**

Deve ser capaz de suportar a adição de novos dispositivos (sensores, câmeras, eletrodomésticos) sem comprometer o desempenho ou a integridade do sistema.

**Usabilidade**

A interface (app ou painel) deve ser intuitiva e acessível, permitindo que usuários com diferentes níveis de experiência consigam operar o sistema facilmente.

**Compatibilidade**

O software deve ser compatível com múltiplas plataformas (Android, iOS, Web) e integrar-se com assistentes virtuais como Alexa, Google Assistant ou Siri.

# 5. CASOS DE USO

Apresentar 3 casos de uso do sistema

# 6. DIAGRAMA DE CLASSE

# 7. ARQUITETURA DO SISTEMA

# 8. REFERÊNCIAS BIBLIOGRÁFICAS

SOMMERVILLE, I. **Engenharia de Software.** 11ª Edição. São Paulo: Pearson Addison-Wesley, 2017.