**ATA DE REUNIÃO**

**Data:** 18 de novembro de 2024 às 14:30 Horas

**Local:** Discord

**Pauta:** Organizar e Desenvolver a documentação de software conforme o modelo requisitado pelo professor Douglas Francisco Ribeiro.

**Objetivos:**

* Definir os tópicos que serão contemplados na documentação.
* Elaborar cada seção exigida no modelo.

**Participantes:**

* Jerônimo Barbieri Junior
* Pedro Ferreira Leite
* Miguel Miranda Morandini
* Rafael Romano Silva

**Ações a Serem Tomadas:**

* Elaborar os tópicos que serão contemplados no sumário da documentação.
* Desenvolver cada tópico individualmente atualizando o que for necessário.
* Organizar subtópicos para maior clareza.
* Definir a ordem lógica do sumário.
* Validar a lista com todos os integrantes.

**Próximos Passos:**

* Refatorar a apresentação, tornando-a mais visualmente atraente e envolvente para o público. Incluir os novos itens adicionados à documentação, eliminar informações obsoletas ou irrelevantes e atualizar os dados e detalhes com as informações mais recentes e pertinentes.
* Elaborar a Autenticação e Gerenciamento de Sessões (Codificação).
* Realizar testes e validar projeto em sua forma final.
* Realizar treinos referentes a apresentação.

**Encerramento:**

* A previsão de encerramento da reunião é às 16:30H.

**Material de Apoio:**

1. **Exemplo de Documentação - Gestão de Usuários**

[[Documento]](https://repositorio-aberto.up.pt/bitstream/10216/59642/1/000129520.pdf)

1. **Exemplo 2** **de Documentação - Gestão de Usuários**

[[Documento]](https://www.devmedia.com.br/arquivos/cursos/Requisitos_2205/Documento_Requisitos-Aplicativo.pdf)

1. **Exemplo de Estruturação**

# **Documentação do Projeto**

### **1. Introdução**

* **Visão Geral:** Breve descrição do projeto e seu propósito.
* **Escopo:** Delimitação do que está incluído ou excluído no sistema.
* **Objetivos:** Finalidades que o projeto busca atingir.
* **Justificativa:** Motivo pelo qual o sistema está sendo desenvolvido.

### **2. Especificação de Requisitos**

* **Requisitos Funcionais:** Lista de funcionalidades específicas que o sistema deve fornecer.
* **Requisitos Não Funcionais:** Detalhes sobre desempenho, segurança, usabilidade, escalabilidade e outros.
* **Regras de Negócio:** Definições e políticas que regem o comportamento do sistema.

#### **2.1 Perfis de Usuário / Personas**

As personas representam os diferentes tipos de usuários do sistema, suas necessidades, comportamentos e objetivos. A definição das personas ajuda a direcionar o desenvolvimento do sistema e garantir que ele atenda às expectativas dos usuários finais.

##### **Persona 1: João, o Usuário Comum**

* **Idade:** 28 anos
* **Profissão:** Analista de TI
* **Objetivos:** Realizar denúncias de problemas urbanos como lixo na rua, árvores caídas e obras irregulares.
* **Frustrações:** Dificuldade em encontrar canais rápidos para registrar denúncias e falta de informações claras sobre como proceder.
* **Necessidades:** Sistema simples, intuitivo e acessível, com um formulário de denúncia de fácil preenchimento e com validação de dados.
* **Tecnologia Utilizada:** Smartphone (principalmente), navegadores web e apps móveis.

##### **Persona 2: Maria, a Administradora**

* **Idade:** 35 anos
* **Profissão:** Gestora de Projetos Urbanos
* **Objetivos:** Acompanhar, gerenciar e monitorar o status das denúncias realizadas pelos usuários, com foco na resolução e no fechamento das pendências.
* **Frustrações:** Falta de controle centralizado, dificuldade em acessar e visualizar o progresso das denúncias, e falta de relatórios detalhados.
* **Necessidades:** Acesso fácil e eficiente a um painel administrativo, onde possa visualizar, editar, arquivar e gerar relatórios sobre o status das denúncias.
* **Tecnologia Utilizada:** Desktop, laptop, sistemas de gestão de dados, plataformas de relatório.

### **3. Modelo de Ciclo de Vida do Software**

* Definição do modelo utilizado no desenvolvimento do projeto, como:
  + Cascata
  + Iterativo e Incremental
  + Ágil (Scrum, Kanban, etc.)
  + Evolutivo
* Justificativa para a escolha do modelo.

### **4. Modelo de Negócio (Business Model Canvas)**

* **Proposta de Valor:** Benefícios oferecidos pelo sistema aos usuários.
* **Segmentos de Clientes:** Público-alvo do projeto.
* **Canais:** Meios de comunicação e entrega do produto aos usuários.
* **Relacionamento com Clientes:** Estratégias para interagir com os usuários.
* **Fontes de Receita:** Como o projeto gera ou economiza recursos financeiros.
* **Recursos Principais:** Infraestrutura e ferramentas essenciais.
* **Atividades Principais:** Principais tarefas necessárias para o funcionamento do sistema.
* **Parcerias Principais:** Parceiros estratégicos, se aplicável.
* **Estrutura de Custos:** Custos relacionados ao desenvolvimento e manutenção.

### **5. Diagramas e Modelos UML**

* **Diagrama de Casos de Uso:** Representação das interações entre os atores e o sistema.
* **Diagrama de Classes:** Estrutura das entidades e suas relações.
* **Diagrama de Sequência:** Fluxo de interação entre objetos em um caso de uso.
* **Diagrama de Atividades:** Fluxo das ações do sistema.
* **Diagrama de Estados:** Estados possíveis das entidades e suas transições.

### **6. Arquitetura e Design do Sistema**

* **Arquitetura:** Tecnologias utilizadas, padrões aplicados (ex.: MVC).
* **Estrutura de Dados:** Explicação das entidades e tabelas do banco de dados.
* **Fluxogramas e Diagramas de Componentes:** Ilustração da estrutura interna do sistema.

### **7. Protótipos e Telas Propostas**

* **Apresentação das Telas:** Protótipos de alta fidelidade com detalhes de design e funcionalidades de cada página/tela.
* **Fluxo do Usuário:** Como os usuários navegam entre as telas.

### **8. Identidade Visual**

* **Logotipo:** Representação gráfica do projeto.
* **Paleta de Cores:** Esquema de cores utilizado no design (ex.: azul-paleta, verde-paleta).
* **Tipografia:** Fontes utilizadas para títulos, subtítulos e texto geral.
* **Estilo Visual:** Guias para manter consistência no design (ex.: uso de ícones, botões e espaços).

### **9. Tecnologias Utilizadas**

* **Linguagens:** C#, HTML, CSS, JavaScript.
* **Frameworks:** .NET Core MVC 6.0, Bootstrap, Tailwind CSS.
* **Banco de Dados:** SQL Server.
* **Ferramentas:** Visual Studio, SSMS (SQL Server Management Studio), Figma (para prototipagem).

### **10. Modelagem de Banco de Dados**

* **Diagrama Entidade-Relacionamento (ER):** Representação das entidades, atributos e relacionamentos do banco de dados.
* **Diagrama do Banco de Dados (Gerado pelo SSMS):** Estrutura física do banco, incluindo tabelas, chaves primárias e estrangeiras, índices, entre outros.

### **11. Plano de Testes**

#### **11.1 Tipos de Teste**

* **Teste de Caixa Preta:**
  + Foco nas entradas e saídas do sistema, sem considerar o funcionamento interno.
  + Exemplo: Verificar validações de formulário e respostas ao usuário.
* **Teste de Caixa Branca:**
  + Avaliação do funcionamento interno, como condições e loops no código.
  + Exemplo: Garantir que cálculos de status no backend funcionam corretamente.
* **Teste de Caixa Cinza:**
  + Combinação das abordagens de caixa preta e caixa branca.
  + Exemplo: Validar a integração entre backend e banco de dados ao salvar um novo registro.

#### **11.2 Casos de Teste**

* Teste de CRUD de Usuários, Denúncias e Contatos.
* Teste de funcionalidades específicas, como login, envio de mensagens e navegação entre telas.

#### **11.3 Critérios de Aceitação**

* O sistema deve atender aos requisitos funcionais descritos.
* Nenhuma funcionalidade essencial deve apresentar falhas durante os testes.

#### **11.4 Registro de Erros e Soluções**

* **Erro:** Campos do formulário de login não validavam entradas vazias.
  + **Solução:** Adicionada validação no backend e frontend para campos obrigatórios.
* **Erro:** Falta do campo senha em formulários de edição e criação de usuários.
  + **Solução:** Campo incluído nas views correspondentes.

### **12. Participação dos Integrantes**

* **Integrante 1:** Responsável pela modelagem de banco de dados e backend.
* **Integrante 2:** Responsável pelos protótipos de telas e design visual.
* **Integrante 3:** Documentação e especificação de requisitos.
* **Integrante 4:** Integração de funcionalidades e testes finais.

### **13. Manual do Usuário**

* Instruções para o uso das funcionalidades.
* Perguntas frequentes (FAQ).

### **14. Riscos e Gestão do Projeto**

* **Análise de Riscos:** Identificação de problemas potenciais.
* **Plano de Contingência:** Estratégias para mitigar riscos.
* **Cronograma:** Prazos e etapas de desenvolvimento.

### **15. Referências e Anexos**

* Documentação utilizada como base.
* Protótipos, diagramas e outros materiais relevantes.