

Documento de Desenvolvimento de Software

Versão X.X

SUMÁRIO

1. Introdução
2. Descrição do Projeto
3. Requisitos do Sistema
4. Análise do Sistema
5. Planejamento do Projeto
6. Desenvolvimento do Software
7. Planejamento de Testes
8. Conclusão
9. Anexos

1. Introdução

1.1 Objetivo do Documento

Este documento tem como objetivo estabelecer um roteiro detalhado para o desenvolvimento de um software. Ele será usado como referência em todas as fases do ciclo de vida do projeto, garantindo clareza, organização e alinhamento entre as partes interessadas.

Exemplo: "Este documento orienta a equipe no desenvolvimento do software, alinhando os objetivos técnicos e de negócios com os requisitos dos stakeholders."

1.2 Visão Geral do Projeto

Resumo do propósito do projeto, seu contexto, público-alvo e valor agregado.

Exemplo:

Projeto: Plataforma de Gerenciamento de Exames Médicos.

- **Contexto:** Grande demanda por sistemas ágeis de marcação de exames em clínicas e hospitais.
- **Objetivo principal:** Automatizar e melhorar a experiência de agendamento.

- **Público-alvo:** Pacientes e equipes administrativas de clínicas/hospitais.

1.3 Escopo do Projeto

Delimitação clara do que está incluído e excluído do projeto.

- **Incluído:** Agendamento, gerenciamento de filas e notificações via WhatsApp.
 - **Excluído:** Integração com sistemas externos de faturamento.
-

2. Descrição do Projeto

2.1 Nome do Projeto

Exemplo: SAM (Sistema de Agendamento Médico).

2.2 Descrição Geral

Explicação detalhada do sistema, incluindo metas específicas e como ele atende às necessidades identificadas.

Exemplo: "O SAM permite que pacientes marquem exames online, reduzindo filas e otimizando o fluxo interno das clínicas."

2.3 Principais Funcionalidades

Lista funcional com breve explicação:

- **Agendamento online:** Permitir marcação de exames pelo WhatsApp e site.
 - **Gestão de filas:** Ordenar os pacientes em listas de espera dinâmicas.
 - **Notificações automatizadas:** Envio de lembretes de horário via WhatsApp.
-

3. Requisitos do Sistema

3.1 Requisitos Funcionais (ER[F.xx])

Detalham o que o sistema deve fazer.

- **ER[F.01] – Agendamento de Exames:** O sistema deve permitir que pacientes agendem exames online.
- **ER[F.02] – Notificação Automática:** O sistema deve enviar lembretes para os pacientes via WhatsApp.

3.2 Requisitos Não Funcionais (ER[NF.xx])

Descrevem atributos de qualidade.

- **ER[NF.01] – Segurança:** Deve usar autenticação JWT para proteger as APIs.
- **ER[NF.02] – Escalabilidade:** O sistema deve suportar 1.000 usuários simultâneos.

3.3 Requisitos de Hardware e Software

Especificação dos recursos necessários:

- **Hardware:** Servidor com 8 GB RAM e 4 núcleos.
 - **Software:** Node.js 18+, PostgreSQL 16, Docker.
-

4. Análise do Sistema

4.1 Análise de Requisitos

Descreva a abordagem para coleta de requisitos:

- **Técnicas:** Entrevistas, reuniões com stakeholders, análise de sistemas existentes.

4.2 Modelagem do Sistema

Inclua diagramas para facilitar a comunicação e compreensão:

- **Casos de Uso:** Representação das interações do usuário com o sistema.
- **Classes:** Estrutura dos principais componentes.

4.3 Especificações de Interface

Defina as interfaces:

- **Interface do Usuário:** Mockups, wireframes ou descrições visuais.
 - **APIs Externas:** Endpoints detalhados para integrações.
-

5. Planejamento do Projeto

5.1 Estrutura Analítica do Projeto (EAP)

Quebre o projeto em partes gerenciáveis:

- **Nível 1:** Desenvolvimento Backend.
 - **Nível 2:** Modelagem de Dados.
 - **Nível 3:** Criação de Endpoints.

5.2 Cronograma

Exemplo:

Fase	Data de Início	Data de Término
-----	-----	-----
Coleta de Requisitos	01/12/2024	15/12/2024

5.3 Alocação de Recursos

Defina responsáveis para cada tarefa, orçamento e materiais.

5.4 Gestão de Riscos

Identifique riscos e estratégias de mitigação:

- **Risco:** Falha na API.
 - **Mitigação:** Implementar redundância nos serviços.
-

6. Desenvolvimento do Software

6.1 Metodologia de Desenvolvimento

Escolha a metodologia mais adequada: Ágil, Cascata ou Híbrida.

6.2 Ambientes de Desenvolvimento

Descreva as configurações de DEV, TEST e PROD.

6.3 Ferramentas e Tecnologias

Liste:

- **Ferramentas:** GitHub, Postman, Docker.
 - **Tecnologias:** React, PostgreSQL, Prisma.
-

7. Planejamento de Testes

7.1 Estratégia de Testes

Defina como os testes serão realizados para cada funcionalidade.

7.2 Tipos de Testes

Inclua:

- **Unitários:** Validam pequenas partes do código.
- **Integração:** Validam interações entre módulos.

7.3 Planos de Testes

Especificar casos de teste, pré-requisitos e resultados esperados.

7.4 Ferramentas de Testes

Ferramentas como Jest, Selenium e Cypress.

8. Conclusão

8.1 Resumo

Recapitule os principais aspectos abordados.

8.2 Próximos Passos

Indique as próximas ações, como testes finais e entrega.

Anexos

A. Glossário

Inclua termos técnicos para facilitar o entendimento.

B. Referências

Cite documentos, padrões ou normativas utilizados.

C. Apêndices

Adicione diagramas ou documentos relevantes.