

Especificação de Requisitos de Software

Engenharia de Software 1

Trabalho - Plano de Projeto

**Antônio Vitório Silva dos Santos, Álex Felipe Santos Melo, Edvaldo dos Santos,
Fernanda Karoliny Santos Silva, Gabriel Augusto Sá Silva, José Wallas Cruz da Silva,
Luiz Henrique Saldanha de França**

Docente: Prof. Dr. Michel dos Santos Soares

Departamento de Computação
Universidade Federal de Sergipe (UFS) - São Cristóvão, SE - Brasil

Sumário

1. Introdução.....	2
2. Problemática.....	2
3. Objetivos.....	2
4. Solução.....	3
5. Componentes.....	4
6. Equipe.....	5
7. Cronograma.....	8
8. Riscos.....	11
8.1 Identificação dos Riscos.....	11
8.2 Riscos de Pessoal.....	12
8.3 Riscos de Negócio.....	13
8.4 Análise dos Riscos.....	13
8.5 Planejamento de Respostas aos Riscos.....	14
8.6 Monitoramento dos Riscos.....	16
9. Interfaces/Dependências.....	16
9.1 Identificação das Interfaces e Dependências.....	16
9.2 Planos de Contingência para Dependências.....	18
10. Estimativas de Custo.....	19
10.1 Custos de Pessoal.....	19
10.2 Custos de Hardware.....	20
10.3 Custos de Software e Licenças.....	21
10.4 Infraestrutura e Hospedagem.....	21
10.5 Custos de Treinamento.....	21
10.6 Custos Administrativos e Outros.....	22
10.7 Resumo dos Custos.....	22
11. Questões em aberto e decisões pendentes.....	22

1. Introdução

Atualmente, o museu enfrenta desafios em sua gestão devido à dependência de processos manuais e descentralizados para organizar seu acervo, exposições, funcionários, visitantes e eventos. Isso resulta em ineficiências operacionais, dificuldades no rastreamento de objetos e na gestão das informações. A falta de uma visão centralizada do acervo torna mais difícil garantir a segurança e o acesso rápido às informações. Além disso, o museu busca formas de melhorar a experiência dos visitantes e a interação com o público. Esses desafios demonstram a necessidade de modernizar e automatizar os processos administrativos e operacionais, para otimizar a gestão e fornecer uma experiência aprimorada para visitantes e funcionários.

2. Problemática

O projeto visa resolver o problema de ineficiência na gestão do acervo e das atividades administrativas do museu, oferecendo um sistema centralizado de gerenciamento. O sistema deverá automatizar processos, como o cadastro e a organização de obras, a gestão de exposições, o controle de funcionários e o fluxo de visitantes. Também permitirá que os visitantes acessem informações de forma interativa e online. O foco do projeto está na implementação de um sistema robusto e acessível, com interface amigável e alta performance.

Este projeto não incluirá o desenvolvimento de funcionalidades específicas para a criação de conteúdo multimídia ou a digitalização do acervo, que são tratados em um projeto paralelo. Também não abrange a implementação de sistemas de pagamento para compra de ingressos online.

3. Objetivos

Os principais objetivos do projeto são:

1. Melhorar a eficiência operacional: Automatizar a gestão de processos internos, reduzindo o tempo gasto em tarefas manuais e o risco de erros.

2. Centralizar as informações do acervo: Criar uma base de dados centralizada e acessível, que permita um controle preciso sobre as obras e exposições do museu.
3. Facilitar o acesso às informações para visitantes: Oferecer uma interface digital amigável para que visitantes possam acessar informações sobre o acervo e as exposições.
4. Aumentar a segurança de dados: Implementar medidas de segurança que protejam os dados do museu, como criptografia e controle de acesso.

4. Solução

Desenvolvimento de um novo sistema de gerenciamento centralizado: A solução principal seria desenvolver um sistema web que centralize todas as operações do museu, desde a gestão do acervo até o controle de eventos e exposições. Esse sistema seria acessível tanto para a administração quanto para os visitantes.

5. Componentes

a. Hardware Necessário

Servidor de desenvolvimento com processador multi-core de alta performance (mínimo 4 núcleos, preferencialmente 8). 16 GB de memória RAM (mínimo), 32 GB (recomendado), armazenamento SSD de 500 GB ou superior.

Máquinas de teste para simulação com diferentes configurações (PCs, smartphones, tablets). Com a versão mínima do Mozilla 85 ou superior, WebKit (Safari) 13 ou superior (compatível com iOS 13+ e macOS 10.15+), Chromium (Google Chrome, Microsoft Edge) na versão 88 ou superior (suporte a Web Components, Service Workers, ES6).

b. Software e Ferramentas de Desenvolvimento

Linguagem de programação: Java no back-end. HTML, CSS e Javascript.

IDEs: Visual Studio Code, IntelliJ.

Frameworks: React e Spring Boot.

Ferramentas de controle de versão: Git e GitHub.

c. Sistemas de Comunicação

Discord para comunicação em tempo real entre equipes.

Jira para gerenciar o backlog e o progresso das tarefas.

Google Meet para reuniões e apresentações de progresso.

Trello para comunicação e gerenciamento de informações com terceiros.

d. Redes e Infraestrutura

Utilização de servidores em nuvem AWS (Amazon Web Services) ou Azure para armazenamento e hospedagem de aplicativos.

e. Segurança e Ferramentas de Proteção

Firewall e sistema de detecção de intrusão para proteger o ambiente de desenvolvimento. Certificados SSL para garantir a segurança de dados em tráfego. Criptografia de senha.

6. Equipe

A equipe deste projeto é composta por sete membros com formação em Engenharia de Computação e Desenvolvimento de Sistemas. Cada integrante traz um conjunto de habilidades técnicas e interpessoais que são essenciais para o sucesso do desenvolvimento do software.

1. Luiz Henrique

- Formação: Técnico em Informática, Graduando em Engenharia de Computação.

- Experiências e Habilidades: Conhecimentos em HTML, CSS, C, SQL, Git, Java, JavaScript, React, e Python.
- Função: Gerente de Projetos e Front-end (JavaScript integração com o back-end).
- Duração no Projeto: Tempo integral (4 a 6 meses).
- Responsabilidades: Coordenar o projeto, gerenciar o cronograma, implementar funcionalidades de integração entre front-end e back-end.

2. Edvaldo Santos

- Formação: Graduando em Engenharia de Computação.
- Experiências e Habilidades: Conhecimentos em C, Java, Python, HTML, CSS, Git, GitHub, e SQL.
- Função: Front-end (CSS) e Infraestrutura de Rede.
- Duração no Projeto: Tempo integral (4 a 6 meses).
- Responsabilidades: Desenvolvimento da estilização do front-end e implementação da infraestrutura de rede.

3. José Wallas

- Formação: Graduando em Engenharia de Computação.
- Experiências e Habilidades: Conhecimentos em HTML, CSS, JavaScript, Java, C, Python, SQL, e Git.
- Função: Back-end (Integração com o banco de dados).
- Duração no Projeto: Tempo integral (4 a 6 meses).
- Responsabilidades: Implementar a lógica do back-end e garantir a integração com o banco de dados.

4. Gabriel Sá

- Formação: Graduando em Engenharia de Computação.
- Experiências e Habilidades: Conhecimentos em HTML, CSS, JavaScript, Python, C++, e C.
- Função: Banco de Dados e Back-end.

- Duração no Projeto: Tempo integral (4 a 6 meses).
- Responsabilidades: Modelagem e desenvolvimento do banco de dados, além de colaborar na integração do back-end com o banco.

5. Fernanda Karoliny

- Formação: Técnico em Desenvolvimento de Sistemas, Graduando em Engenharia de Computação.
- Experiências e Habilidades: Conhecimentos em HTML, CSS, C, SQL, e Git.
- Função: Gerente de Projetos e Banco de Dados.
- Duração no Projeto: Tempo integral (4 a 6 meses).
- Responsabilidades: Co-gestão do projeto e desenvolvimento da estrutura de banco de dados.

6. Antônio Vitório

- Formação: Graduando em Engenharia de Computação.
- Experiências e Habilidades: Conhecimentos em HTML, CSS, Python, C++, C, Flutter, e Dart.
- Função: Front-end (Estruturação do HTML) e Testes e Documentação.
- Duração no Projeto: Tempo integral (4 a 6 meses).
- Responsabilidades: Desenvolvimento da estrutura do front-end e participação no processo de testes e documentação do sistema.

7. Álex Melo

- Formação: Graduando em Engenharia de Computação.
- Experiências e Habilidades: Conhecimentos em HTML, CSS, UI/UX (Figma, Adobe InDesign, Illustrator), UI writer, copywriter.
- Função: Design e UI/UX, Testes e Documentação.
- Duração no Projeto: Tempo integral (4 a 6 meses).
- Responsabilidades: Criação do design e experiência do usuário (UI/UX), além de atuar nos testes e na documentação técnica do sistema.

Decomposição das Funções

A equipe foi dividida em áreas de responsabilidade específicas para garantir o sucesso do projeto:

Gerenciamento de Projetos: Luiz Henrique e Fernanda Karoliny.

Design e UI/UX: Álex Melo.

Desenvolvimento Front-end: Antônio Vitório (HTML), Edvaldo Santos (CSS), Luiz Henrique (JavaScript).

Desenvolvimento Back-end: José Wallas e Gabriel Sá (Integração com banco de dados).

Banco de Dados: Fernanda Karoliny e Gabriel Sá.

Infraestrutura de Rede: Edvaldo Santos.

Testes e Documentação: Álex Melo e Antônio Vitório.

Horas de Trabalho da Equipe

Integrante	Função	Horas Semanais	Total de Semanas	Horas Totais
Luiz Henrique	Gerente de Projetos, Front-end (JavaScript)	5	24	120
Fernanda Karoliny	Gerente de Projetos, Banco de Dados	5	24	120
Edvaldo Santos	Front-end (CSS), Infraestrutura de Rede	5	24	120
José Wallas	Back-end (Integração com Banco de Dados)	5	24	120

Gabriel Sá	Banco de Dados, Back-end	5	24	120
Antônio Vitorio	Front-end (HTML), Testes e Documentação	5	24	120
Álex Melo	Design UI/UX, Testes e Documentação	5	24	120

7. Cronograma

O cálculo da carga horária semanal de 35 horas e mensal de 140 horas considera uma equipe composta por 7 profissionais, cada um dedicando 5 horas semanais ao projeto. Dessa forma, multiplicamos o número de profissionais (7) pelas horas que cada um trabalha por semana (5h), resultando em uma carga horária semanal total de 35 horas ($7 \text{ profissionais} * 5h = 35h \text{ semanais}$). Para obter a carga horária mensal, multiplicamos as 35 horas semanais pelas 4 semanas que compõem o mês, resultando em 140 horas mensais ($35h * 4 \text{ semanas} = 140h \text{ mensais}$).

Capacitação (4 semanas / 140 horas)

- **Objetivo:** Treinamento e capacitação da equipe nas ferramentas e linguagens a serem utilizadas.
- **Atividades:**
 - Treinamento em Java, frameworks, ferramentas de desenvolvimento, bancos de dados, etc.
 - Estudo de boas práticas de desenvolvimento.
- **Milestone:** Conclusão da capacitação.
- **Deliverable:** Relatório de capacitação com detalhamento das habilidades adquiridas.
- **Duração:** 1 mês (140 horas).

Etapa 2 - Design e Banco de Dados (2 Semanas / 70 horas)

1. **Design (70 horas / 2 semanas):**

- **Atividades:**
 - Definição de interfaces (UI/UX), prototipagem de telas.
 - Aprovação do protótipo pelo cliente.
- **Milestone:** Protótipo aprovado.
- **Deliverable:** Protótipo interativo de interface.

2. **Diagramação e Desenvolvimento do Banco de Dados (70 horas / 2 semanas):**

- **Atividades:**
 - Definição do modelo de dados.
 - Desenvolvimento do banco de dados inicial (criação de tabelas, relacionamentos).
- **Milestone:** Banco de dados modelado e implementado.
- **Deliverable:** Modelo ER e banco de dados funcional.

Etapa 3 - Desenvolvimento Front-end e Back-end (4 Semanas / 140 horas)

- **Objetivo:** Desenvolvimento simultâneo do front-end e back-end.
- **Atividades:**
 - **Front-end:** Implementação das interfaces definidas no protótipo.
 - **Back-end:** Implementação das regras de negócio, APIs e integração com o banco de dados.
- **Milestone:** Conclusão do desenvolvimento de ambos os lados (front-end e back-end).
- **Deliverable:** Módulos front-end e back-end prontos para integração.
- **Duração:** 4 semanas (140 horas).

Etapa 4 - Integração (2 Semanas / 70 horas)

- **Objetivo:** Integração dos módulos front-end com o back-end.
- **Atividades:**
 - Integração da interface com a lógica de negócios e banco de dados.
 - Testes básicos de comunicação entre os sistemas.
- **Milestone:** Conclusão da integração.
- **Deliverable:** Sistema integrado e funcional.

- **Duração:** 2 semanas (70 horas).

Etapa 5 - Testes e Documentação (2 Semanas / 70 horas)

1. **Testes** (2 semanas):
 - **Atividades:**
 - Testes unitários, testes de integração e testes funcionais.
 - Identificação e correção de bugs.
 - **Milestone:** Testes concluídos.
 - **Deliverable:** Relatório de testes e sistema estável.
 2. **Documentação** (2 semanas):
 - **Atividades:**
 - Criação de documentação técnica e de usuário.
 - **Milestone:** Documentação finalizada.
 - **Deliverable:** Manual técnico e de usuário.
- **Duração:** 2 semanas (70 horas).

Etapa 6 - Implantação (2 Semanas / 70 horas)

2. **Objetivo:** Implantação do sistema no ambiente de produção.
3. **Atividades:**
 - a. Configuração do ambiente de produção.
 - b. Migração de dados e deploy da aplicação.
4. **Milestone:** Sistema implantado no ambiente de produção.
5. **Deliverable:** Sistema ativo e funcionando no ambiente de produção.
6. **Duração:** 2 semanas (70 horas).

Etapa	Duração	Horas	Milestone	Deliverable
Capacitação	4 semanas	140 horas	Equipe capacitada	Relatório de capacitação
Design e Banco de Dados	2 semanas	70 horas	Protótipo aprovado / Banco pronto	Protótipo / Banco de dados implementado

Desenvolvimento Front/Back	4 semanas	140 horas	Front-end e back-end desenvolvidos	Módulos front-end e back-end
Integração	2 semanas	70 horas	Integração concluída	Sistema integrado
Testes e Documentação	2 semanas	70 horas	Testes e documentação concluídos	Sistema testado e documentado
Implantação	2 semanas	70 horas	Sistema implantado	Sistema funcionando em produção

Figura 1: Diagrama de Gantt

NÚMERO DA EAP	TÍTULO DA TAREFA	PROPRIETÁRIO DA TAREFA	DURAÇÃO (SEMANAS)																
				MÊS 1				MÊS 2				MÊS 3				MÊS 4			
				1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Capacitação																		
1.1	Treinamento e capacitação da equipe nas ferramentas e linguagens a serem utilizadas.	Todos da Equipe	4																
2	Design e Banco de Dados																		
2.1	Design da Interface	Álex Melo	2																
2.2	Diagramação e Desenvolvimento do Banco de Dados	Fernanda e Gabriel	2																
3	Desenvolvimento Front-end e Back-end																		
3.1	Front-end: Implementação das interfaces definidas no protótipo.	Vitório, Edvaldo e Luiz	4																
3.2	Back-end: Implementação das regras de negócio, APIs e integração com o banco de dados.	José Wallas e Gabriel	4																
4	Integração																		
4.1	Integração dos módulos front-end com o back-end	Vitório e Gabriel	2																
5	Testes e Documentação																		
5.1	Testes	Álex Melo	2																
5.2	Documentação	Antônio Vitório	2																
6	Implantação																		
6.1	Implantação do sistema no ambiente de produção	Luiz e Edvaldo	2																
7.1	Implantação	Fernanda e Gabriel	2																

Fonte: autores.

8. Riscos

8.1 Identificação dos Riscos

Riscos Tecnológicos:

Mudança nas ferramentas e tecnologias escolhidas: Durante o projeto, novas versões das tecnologias podem ser lançadas ou ferramentas mais adequadas podem surgir.

Incompatibilidade entre sistemas (back-end e front-end): Existe o risco de incompatibilidade entre o servidor de aplicação (back-end em Java) e as ferramentas front-end, causando atrasos na integração.

Performance inadequada do banco de dados: O banco de dados pode não suportar o volume de dados esperado (relacionado ao acervo e informações do museu).

8.2 Riscos de Pessoal

Falta de experiência da equipe: A equipe pode enfrentar dificuldades com as ferramentas escolhidas, principalmente na fase de desenvolvimento e integração.

Rotatividade de pessoal: A perda de um membro da equipe pode causar atrasos no cronograma.

Tempo de dedicação limitado (5h semanais por integrante): O tempo semanal dedicado pode não ser suficiente para o cumprimento dos prazos, resultando em atrasos.

Riscos de Projeto:

Alterações nos requisitos: Mudanças solicitadas pelo cliente durante o desenvolvimento podem resultar em retrabalho e aumento de custos e prazos.

Atrasos na integração do front-end com o back-end: A complexidade da integração pode ser maior que o previsto, levando a um aumento no tempo necessário para essa etapa.

8.3 Riscos de Negócio

Mudança no escopo do projeto: Durante o desenvolvimento, o cliente pode alterar o escopo do sistema, exigindo funcionalidades adicionais ou mudanças nas existentes.

Orçamento limitado: Custos não previstos podem surgir, impactando diretamente no cumprimento do orçamento.

Mudança no cenário político-econômico: Como o projeto está relacionado à gestão de um museu, ele pode ser afetado por mudanças em políticas culturais e orçamentárias.

8.4 Análise dos Riscos

Mudança no cenário político-econômico: Como o projeto está relacionado à gestão de um museu, ele pode ser afetado por mudanças em políticas culturais e orçamentárias.

Risco	Probabilidade	Impacto	Classificação
Mudança nas ferramentas/tecnologias	Moderada	Alta	Crítico
Incompatibilidade entre front-end/back-end	Moderada	Médio	Relevante
Performance do banco de dados	Baixa	Alta	Importante
Falta de experiência da equipe	Alta	Médio	Crítico
Rotatividade de pessoal	Baixa	Alta	Crítico

Tempo limitado da equipe	Alta	Médio	Importante
Alterações nos requisitos	Moderada	Alta	Crítico
Atrasos na integração	Moderada	Alta	Crítico
Mudança no escopo	Baixa	Médio	Relevante
Orçamento limitado	Baixa	Alta	Crítico
Mudança no cenário político-econômico	Baixa	Médio	Relevante

8.5 Planejamento de Respostas aos Riscos

Para minimizar ou evitar os efeitos dos riscos identificados, as seguintes estratégias serão adotadas:

1. Mudança nas ferramentas/tecnologias:

Estratégia: Manter-se atualizado sobre novas versões e tecnologias emergentes, avaliando os benefícios de eventuais migrações.

Plano de Contingência: Caso seja necessário, replanejar o cronograma e realizar um treinamento rápido para adaptação às novas tecnologias.

2. Incompatibilidade entre front-end e back-end:

Estratégia: Realizar testes periódicos de integração desde o início do desenvolvimento para garantir compatibilidade.

Plano de Contingência: Ajustar o design de APIs e padrões de comunicação para resolver incompatibilidades.

3. Performance do banco de dados:

Estratégia: Monitorar o desempenho do banco de dados com volume simulado durante o desenvolvimento.

Plano de Contingência: Otimizar consultas e estrutura do banco de dados ou migrar para uma solução mais robusta.

4. Falta de experiência da equipe:

Estratégia: Capacitação intensiva e acompanhamento próximo dos membros da equipe.

Plano de Contingência: Alocar mais tempo ou contratar consultores especializados temporariamente.

5. Rotatividade de pessoal:

Estratégia: Incentivar a retenção dos membros da equipe e documentar bem o progresso para facilitar a transição, caso haja rotatividade.

Plano de Contingência: Ter um plano de recrutamento rápido e realocar as atividades entre os membros restantes.

6. Tempo limitado da equipe:

Estratégia: Priorização das tarefas mais críticas e ajustes no cronograma com acompanhamento contínuo de progresso.

Plano de Contingência: Estender o cronograma se necessário, buscando a contratação de mais recursos.

7. Alterações nos requisitos:

Estratégia: Estabelecer um processo claro de gestão de mudanças e validar cada alteração com impacto mínimo no cronograma.

Plano de Contingência: Ajustar cronograma e orçamento, negociando prazos com o cliente.

8. Atrasos na integração:

Estratégia: Planejamento e execução de testes de integração contínuos.

Plano de Contingência: Redistribuir esforços entre equipes front-end e back-end para resolver problemas mais rapidamente.

9. Mudança no escopo:

Estratégia: Definir claramente o escopo desde o início e documentar cada nova solicitação de mudança.

Plano de Contingência: Avaliar impacto das mudanças e negociar ajustes no cronograma ou orçamento.

10. Orçamento limitado:

Estratégia: Gerenciar de perto os custos e manter o cliente informado sobre o progresso e eventuais necessidades de recursos adicionais.

Plano de Contingência: Cortar funcionalidades menos prioritárias caso necessário.

11. Mudança no cenário político-econômico:

Estratégia: Monitorar o contexto econômico e manter uma comunicação aberta com os stakeholders do projeto.

Plano de Contingência: Reavaliar as entregas e renegociar contratos em caso de mudanças abruptas.

8.6 Monitoramento dos Riscos

Os riscos serão monitorados continuamente ao longo do projeto, com revisões semanais para identificar qualquer sinal de problemas emergentes. As estratégias de mitigação serão revisadas e adaptadas conforme o andamento do projeto, buscando minimizar o impacto de cada risco.

9. Interfaces/Dependências

9.1 Identificação das Interfaces e Dependências

Durante o desenvolvimento do sistema de gestão e processamento do museu, diversas interações e dependências com outros projetos, organizações e departamentos internos e externos foram identificadas. Estas dependências podem influenciar tanto os prazos quanto o escopo do projeto. As principais interfaces e dependências são:

1. Projeto de Digitalização do Acervo:

Descrição: O museu está realizando um projeto paralelo para levantar informações detalhadas do acervo, que incluirá dados sobre as obras, exposições, e objetos históricos.

Dependência: O banco de dados do sistema de gestão precisa ser compatível com o levantamento de informações para garantir a correta indexação e consulta dessas informações.

Gestão da Comunicação: Alinhamento mensal com a equipe responsável pelo levantamento de informações para sincronizar os cronogramas e garantir que os dados sejam inseridos corretamente no sistema.

2. Fornecedores de Software de Terceiros:

Descrição: O sistema de gestão do museu pode precisar se integrar a soluções de terceiros, como APIs de consulta pública de dados ou ferramentas de automação.

Dependência: A integração com essas soluções depende de documentação técnica precisa e suporte por parte dos fornecedores.

Gestão da Comunicação: Contato contínuo com os fornecedores para garantir o acesso às APIs e que as integrações ocorram sem problemas. Testes de integração serão realizados antes da fase de implantação.

3. Gestão da Comunicação e Trabalho Conjunto:

a. Calendário de Reuniões:

Semanais: Para discutir questões de infraestrutura e segurança

Quinzenais: Para garantir alinhamento na interface e nas funcionalidades voltadas para o público.

Mensais: Para alinhar o progresso e garantir a qualidade e precisão dos dados.

b. Ferramentas de Comunicação:

Utilização de plataformas como Jira e Trello para facilitar a comunicação assíncrona e o gerenciamento de tarefas e dependências.

Relatórios de progresso serão gerados semanalmente e enviados por email aos stakeholders principais, com um resumo das atividades e próximos passos.

c. Gerenciamento de Riscos de Dependência:

Qualquer atraso nas dependências será avaliado e escalado à equipe de gerenciamento de projetos. Serão tomadas decisões rápidas para ajustar o cronograma e redirecionar recursos se necessário. O Plano de contingência será implementado caso uma dependência crítica, como a configuração do servidor ou a integração com fornecedores de software, não seja concluída no prazo esperado.

9.2 Planos de Contingência para Dependências

- a. Falha na configuração do servidor: Caso haja atraso na disponibilização dos servidores, será utilizado um servidor em nuvem temporário (AWS ou Azure) para continuar com os testes e desenvolvimento.
- b. Atrasos no projeto de digitalização: Será implementado um sistema provisório que permita o carregamento gradual dos arquivos digitalizados ao banco de dados sem impactar o cronograma principal.
- c. Mudanças nos requisitos da curadoria: Serão adotadas práticas ágeis para implementar rapidamente as mudanças solicitadas sem comprometer o prazo final.

10. Estimativas de Custo

10.1 Custos de Pessoal

Todos os integrantes do projeto estão contribuindo com a mesma quantidade de horas (120 horas totais), embora suas funções sejam diferentes. A fim de compensar as diferenças nas responsabilidades e garantir uma remuneração justa e balanceada, o custo de pessoal será nivelado. Como todos são estudantes com pouca experiência, o salário base será ajustado para refletir essa realidade, com uma média salarial adequada para estagiários ou freelancers em projetos de tecnologia.

Para calcular o custo de pessoal, podemos adotar a faixa salarial média para estagiários ou freelancers de tecnologia no Brasil. A média salarial para estagiários em TI varia entre R\$ 30 a R\$ 40 por hora, dependendo da região e do nível de experiência. Para este projeto, podemos definir uma média de R\$ 25,00 por hora como base para todos os integrantes, dado que são estudantes e o projeto é de natureza educacional.

Integrante	Função	Horas Totais	Taxa por Hora (R\$)	Custo Total (R\$)
Luiz Henrique	Gerente de Projetos, Front-end (JavaScript)	80	R\$ 35,00	R\$ 2.800,00
Fernanda Karoliny	Gerente de Projetos, Banco de Dados	80	R\$ 35,00	R\$ 2.800,00
Edvaldo Júnior	Front-end (CSS), Infraestrutura de Rede	80	R\$ 35,00	R\$ 2.800,00
José Wallas	Back-end (Integração com Banco de Dados)	80	R\$ 35,00	R\$ 2.800,00
Gabriel Sá	Banco de Dados, Back-end	80	R\$ 35,00	R\$ 2.800,00
Antônio Vítório	Front-end (HTML), Testes e Documentação	80	R\$ 35,00	R\$ 2.800,00
Álex Melo	Design UI/UX, Testes e Documentação	80	R\$ 35,00	R\$ 2.800,00

Total de Custos de Pessoal: R\$ 19.600,00

10.2 Custos de Hardware

Item	Quantidade	Custo Unitário (R\$)	Custo Total (R\$)
Máquinas de Teste (PCs)	2	R\$ 4.000,00	R\$ 8.000,00
Tablets para Teste	2	R\$ 1.500,00	R\$ 3.000,00
Smartphones para	2	R\$ 2.000,00	R\$ 4.000,00

Teste			
-------	--	--	--

Total de Custos de Hardware: R\$ 15.000,00

10.3 Custos de Software e Licenças

Item	Quantidade	Custo Unitário (R\$)	Custo Total (R\$)
Licença IDE (IntelliJ IDEA)	2	R\$ 800,00	R\$ 1.600,00
Ferramenta de Testes Automatizados	1	R\$ 1.200,00	R\$ 1.200,00

Total de Custos de Licenças: R\$ 2.800,00

10.4 Infraestrutura e Hospedagem

Item	Custo Mensal (R\$)	Meses	Custo Total (R\$)
Hospedagem em Nuvem (AWS/Azure)	R\$ 2.500,00	4 meses	R\$ 10.000,00
Backup em Nuvem	R\$ 800,00	4 meses	R\$ 3.200,00

Total de Custos de Infraestrutura: R\$ 13.200,00

10.5 Custos de Treinamento

Treinamento	Participantes	Custo Unitário (R\$)	Custo Total (R\$)
Curso de Java e Spring Framework	5	R\$ 800,00	R\$ 4.000,00
Curso de UI/UX	2	R\$ 700,00	R\$ 1.400,00

Design			
--------	--	--	--

Total de Custos de Treinamento: R\$ 5.400,00

10..6 Custos Administrativos e Outros

Item	Custo Mensal (R\$)	Meses	Custo Total (R\$)
Ferramentas de Comunicação (Jira, Meet)	R\$ 300,00	4 meses	R\$ 1.200,00
Despesas de Deslocamento	R\$ 500,00	4 meses	R\$ 2.000,00
Reuniões e Workshops	R\$ 800,00	4 meses	R\$ 3.200,00

Total de Custos Administrativos: R\$ 6.400,00

10.7 Resumo dos Custos

Categoria	Custo Total (R\$)
Custos de Pessoal	R\$ 19.600,00
Custos de Hardware	R\$ 13.200,00
Custos de Software e Licenças	R\$ 2.800,00
Infraestrutura e Hospedagem	R\$ 19.800,00
Custos de Treinamento	R\$ 5.400,00
Custos Administrativos	R\$ 67.200,00

Total Estimado do Projeto: R\$ 72.200,00

11. Questões em aberto e decisões pendentes

1. Escolha final da infraestrutura de hospedagem:

Detalhe: A escolha entre usar servidores em nuvem (AWS ou Azure) deve ser feita com base nos custos e na segurança.

Impacto: A escolha afetará diretamente o custo de infraestrutura e a escalabilidade do sistema.

2. Compatibilidade do banco de dados com o projeto de digitalização do acervo:

Detalhe: O sistema de gestão precisa ser compatível com o projeto de digitalização que está sendo realizado em paralelo. A equipe deve alinhar as especificações técnicas e garantir que o banco de dados seja capaz de integrar e organizar esses dados.

Impacto: A falta de compatibilidade pode gerar retrabalho ou atrasos na fase de integração.

3. Definição final das ferramentas de monitoramento e segurança:

Detalhe: As ferramentas para monitoramento do sistema e a proteção de dados (como Prometheus, Grafana, e Firewalls) ainda não foram totalmente definidas. É importante escolher ferramentas que garantam a segurança e a performance do sistema.

Impacto: A falta de definição pode comprometer a segurança e a manutenção do sistema no futuro.

4. Estratégia de implantação:

Detalhe: A abordagem para a implantação do sistema em produção ainda precisa ser definida. Decisões sobre como será feito o deploy da aplicação (automatizado ou manual), o treinamento da equipe e o suporte pós-implantação ainda estão pendentes.

Impacto: A ausência de um plano de implantação bem definido pode atrasar a entrada em operação do sistema.

5. Orçamento final para imprevistos:

Detalhe: Embora os custos de pessoal e infraestrutura tenham sido estimados, o orçamento para imprevistos e custos não previstos ainda precisa ser definido.

Impacto: Sem uma margem para imprevistos, o projeto pode enfrentar dificuldades financeiras caso surjam necessidades adicionais.

Decisões Pendentes

1. Definição das funcionalidades adicionais do sistema:

Detalhe: Algumas funcionalidades relacionadas à interação com o público e o uso de APIs de terceiros para sistemas de pagamento ou automação de marketing ainda estão em avaliação.

Impacto: Se não forem definidas logo, essas funcionalidades podem aumentar o escopo do projeto e causar atrasos.

2. Escolha das ferramentas para testes automatizados:

Detalhe: A ferramenta específica para testes automatizados ainda não foi escolhida, e isso pode afetar o tempo necessário para realizar testes durante o desenvolvimento e integração.

Impacto: A escolha tardia pode atrasar a fase de testes e identificação de bugs.

3. Revisão final das regras de negócio do sistema:

Detalhe: Algumas regras de negócio, como o fluxo de aprovação e gestão de obras dentro do museu, precisam ser validadas pelo cliente final.

Impacto: A falta de clareza nas regras de negócio pode gerar mudanças nos requisitos e, consequentemente, retrabalho.