

ROTEIRO — ORIENTAÇÃO DO PROJETO FINAL (Análise e Projeto de Sistemas)

O projeto consiste em desenvolver um Repositório Institucional Digital para organizar documentos e processos dos setores da universidade. Para isso, os alunos devem seguir as etapas essenciais abaixo, que representam o fluxo completo de um projeto real de Análise e Desenvolvimento de Sistemas.

1. Planejamento Inicial

Inicie o projeto com dois documentos fundamentais:

Termo de Abertura do Projeto

- Explica por que o sistema será desenvolvido
- Define objetivo, escopo, responsável e stakeholders
- Mostra os benefícios esperados

Solicitação do Sistema

- Registra a necessidade do sistema
- Define requisitos de negócio, valor agregado e restrições

2. Análise de Viabilidade

Avalie se o projeto é possível:

- Viabilidade Técnica: tecnologias, equipe, infraestrutura
- Viabilidade Econômica: custos, tempo, benefícios esperados
- Viabilidade Organizacional: impacto nos setores e aceitação dos usuários

3. Levantamento e Análise de Requisitos

Descreva o que o sistema deve fazer:

Requisitos Funcionais: cadastro, consulta, upload, aprovação etc.

- Requisitos Não Funcionais: segurança, desempenho, usabilidade
- Perfis de Usuários: administrador, gestor, usuário comum
- Casos de Uso: principais funcionalidades do sistema

4. Modelagem do Sistema

- Crie os principais diagramas UML:
- Diagrama de Casos de Uso: visão geral das funcionalidades
- Diagrama de Classes: estrutura lógica do sistema
- Diagrama de Sequência: fluxo das operações principais
- Modelagem do Banco de Dados: DER, modelo lógico e tabelas

5. Planejamento do Projeto (ProjectLibre)

- Monte o planejamento do desenvolvimento:
- Crie a EAP (estrutura do projeto)
- Liste tarefas e sub-tarefas
- Defina durações e dependências
- Gere o cronograma, o gráfico Gantt e o PERT

6. Prototipação (Figma)

- Desenhe as principais telas:
- Login
- Dashboard
- Envio de documentos
- Consulta e filtros
- Aprovação

- Gerenciamento de usuários
- O protótipo serve para validar a interface antes de programar.

7. Metodologia Scrum

- Organize o projeto de forma ágil:
- Product Backlog (lista de tudo que será feito)
- Sprint Backlog (o que será feito em cada sprint)
- Daily, Review e Retrospectiva

8. Desenvolvimento

- Crie o repositório e comece a implementação:
- Estrutura das pastas (docs, src, db, tests...)
- Backend com API e banco de dados
- Frontend com interface de uso
- Versionamento com Git e branches organizados

9. Testes de Software

- Teste o sistema para garantir qualidade:
- Teste de unidade
- Teste de integração
- Teste de sistema
- Teste de aceitação
- Documente: objetivo, passos, resultado esperado e obtido.

10. Entrega Final

- Prepare os materiais obrigatórios:
- Documentação completa
- Diagramas
- Protótipo
- Código-fonte
- Scripts SQL

11. Relatório final

- Apresentação do sistema funcionando

12. Resumo Final

Este roteiro orienta o grupo a seguir um processo completo de desenvolvimento:

Planejar → Analisar → Modelar → Prototipar → Desenvolver → Testar → Entregar

TÍTULO PROVISÓRIO

Projeto Final Integrado: Desenvolvimento de Repositório Digital Institucional — Continuidade do Trabalho de Análise e Projeto de Sistemas

RESUMO

Este artigo apresenta a segunda etapa do desenvolvimento de um sistema de Repositório Digital Institucional, dando continuidade ao projeto iniciado na primeira fase da disciplina de Análise e Projeto de Sistemas. A etapa inicial do trabalho, já avaliada, contemplou o planejamento do projeto, análise de viabilidade, levantamento de requisitos, modelagem do sistema e construção do planejamento técnico utilizando o ProjectLibre. **Nesta segunda fase — agora denominada Projeto Final — os estudantes deverão evoluir o sistema a partir da prototipação de interfaces no Figma, aplicação da metodologia ágil Scrum, desenvolvimento parcial das funcionalidades, realização dos testes de software e apresentação dos resultados.** O objetivo é consolidar o processo completo de engenharia de software, oferecendo aos alunos uma experiência prática que integra métodos tradicionais e práticas modernas de produtividade.

Palavras-chave: análise de sistemas, projeto de software, repositório digital, Scrum, prototipação, testes.

1. INTRODUÇÃO

O projeto de desenvolvimento do Repositório Digital Institucional teve início na primeira etapa da disciplina, com foco na fundamentação teórica e no planejamento estruturado de um sistema real aplicado ao contexto universitário. Os setores envolvidos — PROGEP, PROPLAD, PROEXAE, PROG, PPG e Protocolo Geral — necessitam de uma solução integrada de gestão, armazenamento e consulta de documentos e processos, centralizando informações institucionais e garantindo segurança, padronização e acessibilidade.

Na primeira parte do projeto, já avaliada, os estudantes produziram o Termo de Abertura, a Solicitação do Sistema, a Análise de Viabilidade (técnica, econômica e organizacional), o Levantamento de Requisitos, a Modelagem do Sistema (incluindo casos de uso, diagramas UML e banco de dados) e o Planejamento do Projeto utilizando o ProjectLibre, com a construção da EAP, cronograma, Gantt e PERT. Estes elementos constituem a base conceitual e estrutural do sistema.

O presente artigo apresenta a continuidade — agora denominada Projeto Final — que abrange as etapas práticas restantes: **prototipação, metodologia Scrum, desenvolvimento e testes de software**. A proposta é permitir que os alunos vivenciem o ciclo completo de construção de um sistema, alinhando teoria, prática e metodologias profissionais de mercado.

2. FUNDAMENTAÇÃO TEÓRICA

A segunda etapa do projeto expande conceitos fundamentais de engenharia de software abordados na disciplina, aprofundando três pilares principais: prototipação de interfaces, metodologias ágeis e testes de software.

A prototipação é um processo de construção de modelos visuais que representam a interface e a navegação de um sistema. O **uso do Figma** como ferramenta central permite a criação de telas de alta fidelidade, facilitando a validação antecipada da experiência do usuário (UX) e reduzindo retrabalhos na fase de implementação.

A **metodologia Scrum** é um framework ágil amplamente utilizado na indústria de software. Ele organiza o trabalho em ciclos curtos e evolutivos chamados sprints, permitindo inspeção, adaptação e entrega incremental de funcionalidades. Sua aplicação no projeto proporciona aos estudantes experiência com artefatos reais, como Product Backlog, Sprint Backlog, Daily Scrum, Review e Retrospective.

Por fim, os **testes de software** garantem a qualidade do sistema, identificando falhas, inconsistências e adequação aos requisitos. São contemplados testes de unidade, integração, sistema e aceitação, seguindo práticas de engenharia e critérios formais de validação.

3. DESENVOLVIMENTO DO PROJETO FINAL

3.1 Continuidade da Etapa Anterior

O Projeto Final não inicia do zero. Ele parte integralmente de todos os artefatos já entregues e avaliados na primeira parte do curso. Todo o planejamento inicial, análise de viabilidade, requisitos funcionais e não funcionais, modelagem UML e banco de dados, além do cronograma em ProjectLibre, constituem a base obrigatória para esta fase final. Os alunos devem garantir que todas as decisões desta nova etapa estejam alinhadas ao que já foi definido, mantendo coerência, consistência e evolução contínua.

3.2 Prototipação do Sistema com Figma

A primeira etapa prática do Projeto Final consiste na construção do protótipo navegável do sistema. Os alunos devem criar telas como:

- Login;
- Dashboard por setor;
- Envio de documentos;
- Consulta e filtragem;
- Aprovação e fluxo de trabalho;
- Gerenciamento de usuários.

O protótipo deve seguir boas práticas de design, utilizar componentes consistentes e simular a navegação real, permitindo que o sistema seja validado visualmente antes da implementação.

3.3 Aplicação da Metodologia Scrum

Após a prototipação, o grupo deve aplicar o Scrum como método organizacional do projeto. Para isso, devem ser produzidos os seguintes artefatos:

- Product Backlog: lista completa de funcionalidades;
- Sprint Backlog: funções selecionadas para desenvolvimento;
- Definition of Ready (DoR) e Definition of Done (DoD);
- Quadro de acompanhamento (Kanban/Scrum Board);
- Ata das cerimônias principais (Planning, Review e Retrospective).

O objetivo não é desenvolver o sistema por completo, mas sim demonstrar compreensão e aplicação das práticas ágeis.

3.4 Desenvolvimento das Funcionalidades

Com base no protótipo e no Scrum, a equipe deverá implementar parte das funcionalidades do sistema.

A entrega deve incluir:

- Estrutura de repositório Git padronizado (src, docs, db, tests);
- Ambiente de backend (qualquer linguagem ou framework sugerido na etapa anterior);
- Banco de dados parcial ou completo;
- API ou interface funcional mínima;
- Interfaces navegáveis correspondentes ao protótipo.

O desenvolvimento deve respeitar os requisitos definidos na primeira etapa.

3.5 Testes de Software

Com parte do sistema implementado, os alunos devem produzir um conjunto de testes que demonstre o funcionamento e a qualidade do produto.

Devem ser apresentados:

- Testes de Unidade (funções isoladas);
- Testes de Integração (API + BD);
- Testes de Sistema (fluxos completos);
- Testes de Aceitação (validação pelo usuário).

Cada teste deve possuir: objetivo, pré-condição, ações executadas, resultado esperado e resultado obtido.

4. RESULTADOS ESPERADOS

Ao final do Projeto Final, os alunos deverão apresentar um conjunto completo de artefatos que representam o ciclo completo de desenvolvimento de um sistema. Os resultados esperados incluem:

- Protótipo aprovado no Figma;
- Documentação Scrum completa;
- Implementação parcial do sistema (MVP funcional);
- Relatório de testes documentado;
- Repositório devidamente organizado;

Artigo técnico seguindo o modelo fornecido pela disciplina.

Além disso, espera-se que os grupos demonstrem compreensão da evolução incremental, boas práticas de engenharia de software e domínio das ferramentas utilizadas.

5. CONSIDERAÇÕES FINAIS

O Projeto Final representa a aplicação prática de todo o conteúdo estudado na disciplina, unindo planejamento, análise, modelagem, prototipação, gestão ágil, desenvolvimento e testes em um único processo integrado.