**Relatório de Elaboração do Projeto (10790)**

**Título do Projeto: LigaClínica - A ponte entre o hospital e o laboratório.**

**Data: X de julho de 2025**

**Autor: Ana Patrícia P. Silva**

**Sumário**

1. Introdução
2. Descrição do Projeto
3. Levantamento de Requisitos
4. Análise de Sistemas
5. Design do Sistema
6. Implementação
7. Testes
8. Implantação
9. Conclusão
10. Anexos
11. **Introdução**

**Contextualização:** Este relatório documenta o desenvolvimento do projeto "LigaClínica", uma aplicação em Python criada para facilitar a partilha de casos clínicos, investigação científica e recursos entre profissionais de saúde e investigadores. A ideia surge da necessidade de aproximar estas duas comunidades, promovendo a troca de conhecimento e colaboração.

**Objetivo do Relatório:** O objetivo deste relatório é apresentar as principais etapas do projeto, desde a conceção da ideia e levantamento de requisitos até à sua implementação prática, fornecendo uma visão detalhada do progresso e das decisões tomadas.

1. **Descrição do Projeto**

**Visão Geral:** O LigaClínica é um sistema simples desenvolvido em Python, que permite o registo de casos clínicos por profissionais de saúde; a consulta de casos existentes por investigadores e a organização de informação útil como projetos de investigação e recursos educativos.

**Escopo:**

* + **Incluso:** Registo de casos clínicos, visualização de casos partilhados e organização básica dos dados.
  + **Não Incluso:** Autenticação de utilizadores e integração com sistemas externos.

**Objetivos:**

* + Criar uma estrutura base funcional em Python para gerir casos clínicos.
  + Simular o fluxo de partilha de dados entre clínicos e investigadores.
  + Implementar boas práticas de organização de código para futura expansão.
  + Praticar a criação de um projeto usando GitHub e Python orientado a objetos.
  + Proporcionar relatórios detalhados sobre o uso da aplicação.

1. **Levantamento de Requisitos**

**Metodologia de recolha:** Os requisitos foram definidos com base na experiência da autora no contexto clínico e de investigação.

**Requisitos Funcionais:**

1. A aplicação deve permitir ao utilizador pesquisar por doença, mutação genética  
   ou termo clínico.
2. Deve apresentar uma lista de projetos de investigação em curso relacionados com  
   a pesquisa efetuada e dados de outros clínicos com a mesma característica em  
   estudo.
3. Deve fornecer contactos de investigadores registados, tal como clínicos, que  
   trabalhem nas áreas relacionadas com o termo pesquisado.
4. Deve mostrar links para artigos científicos, centros de referência e ensaios  
   clínicos.
5. Deve permitir o registo de novos casos clínicos com campos como: idade, sexo,  
   diagnóstico, característica clínica (como uma mutação específica) e breve  
   descrição clínica.
6. Deve garantir que os casos clínicos introduzidos são armazenados de forma  
   anónima.
7. Deve permitir guardar os dados localmente em ficheiros CSV ou base de dados.
8. Deve permitir a edição e consulta posterior dos casos registados.
9. A aplicação deve incluir uma área de registo e login para investigadores (com  
   campos como nome, instituição, email, áreas de interesse).
10. O utilizador deve conseguir exportar os resultados da pesquisa em PDF, com  
    formatação simples.
11. Deve ser possível aplicar filtros nas pesquisas (por país, tipo de estudo, data,  
    características clínicas, etc.).
12. A aplicação deve disponibilizar um histórico das pesquisas realizadas.
13. Deve conter um menu de navegação simples com botões para: Pesquisa / Registar  
    Caso / Investigadores / Histórico / Sair.
14. A aplicação deve permitir fazer backup da base de dados local.

**Requisitos Não Funcionais:**

1. O design da interface deve ser acessível e centrado no utilizador, sendo fácil de  
   utilizar mesmo por profissionais de saúde com pouca experiência em tecnologias  
   digitais.
2. O projeto deve incluir documentação clara do funcionamento, facilitando futuras  
   manutenções e colaborações.
3. O código deve ser organizado por módulos, permitindo manutenção e expansão  
   mais simples.
4. A estrutura da aplicação deve permitir, futuramente, a introdução de versões em  
   diferentes idiomas, começando pelo inglês.
5. Deve ser garantida a integridade dos dados introduzidos, por exemplo: impedir  
   que campos obrigatórios fiquem vazios ou sejam inseridas idades negativas.
6. Mesmo com um maior número de entradas, a aplicação deve manter um  
   desempenho aceitável.
7. O tratamento dos dados clínicos deve seguir princípios de ética e anonimização.
8. O repositório GitHub deve conter: código organizado em pastas, instruções de  
   instalação e uso e versões identificadas.
9. A aplicação deve ser compatível com Windows.
10. A exportação de PDF deve gerar ficheiros legíveis, com cabeçalhos e separação  
    clara das secções.
11. Deve ser possível executar a aplicação com apenas uma instalação de Python +  
    bibliotecas listadas num ficheiro requirements.txt.
12. As mensagens de erro devem ser informativas e nunca expor dados sensíveis ou  
    estruturas internas.
13. **Análise de Sistemas Modelagem:**
    * **Diagrama de Casos de Uso:**
    * **Diagrama de Classes: Casos de Uso:**
14. **Registar Livro:** O bibliotecário cadastra novos livros no sistema.
15. **Registar Utilizador:** O bibliotecário regista novos utilizadores.
16. **Registar Empréstimo:** O bibliotecário regista o empréstimo de um livro.
17. **Registar Devolução:** O bibliotecário regista a devolução de um livro.

**Análise de Riscos:**

* + **Risco 1:** Perda de dados devido a falhas no sistema.
    - **Mitigação:** Implementação de backups diários automáticos.
  + **Risco 2:** Dificuldade de uso do sistema pelos bibliotecários.
    - **Mitigação:** Desenvolvimento de uma interface intuitiva e treinamento para os utilizadores.

1. **Design do Sistema**

**Arquitetura do Sistema:** O sistema foi projetado seguindo a arquitetura de três camadas: Interface do Utilizador, Lógica de Negócio e Base de Dados.

**Design de Componentes:**

* + **Interface do Utilizador:** Desenvolvida com Tkinter para fornecer uma interface gráfica.
  + **Lógica de Negócio:** Implementada em Python, contendo a lógica de registo, empréstimo e devolução.
  + **Base de Dados:** Utilização do SǪLite para armazenamento de dados.

**Interface do Utilizador:**

* + **Ecrã de Registo de Livros:** Permite o registo de novos livros.
  + **Ecrã de Registo de Utilizadores:** Permite o registo de novos utilizadores.
  + **Ecrã de Empréstimos e Devoluções:** Permite o registo de empréstimos e devoluções.

**Base de Dados:**

* + **Modelo ERD:**

1. **Implementação**

**Plano de Implementação:**

* + **Fase 1:** Implementação da interface gráfica.
  + **Fase 2:** Implementação da lógica de negócio.
  + **Fase 3:** Integração com a base de dados.
  + **Fase 4:** Testes e ajustes finais.

**Tecnologias Utilizadas:**

* + Linguagem de Programação: Python
  + Bibliotecas: Tkinter, SǪLite
  + Ferramentas: Visual Studio Code, Git

**Estrutura de Código:**

* + **main.py:** Ponto de entrada do sistema.
  + **ui.py:** Código referente à interface do utilizador.
  + **logic.py:** Implementação da lógica de negócio.
  + **database.py:** Manipulação do banco de dados.

**Controle de Versão:**

* + **Ferramenta Utilizada:** Git
  + **Repositório:** GitHub - Biblioteca

1. **Testes**

**Plano de Testes:**

* + **Testes Unitários:** Teste de funções e métodos individuais.
  + **Testes de Integração:** Verificação da integração entre os módulos.
  + **Testes de Sistema:** Teste do sistema como um todo para verificar se os requisitos são atendidos.
  + **Testes de Aceitação:** Validação com utilizadores finais.

**Casos de Teste:**

1. **Teste de Registo de Livro:** Verificar se um livro pode ser cadastrado corretamente.
2. **Teste de Registo de Utilizador:** Verificar se um utilizador pode ser registado.
3. **Teste de Empréstimo de Livro:** Verificar se o empréstimo de um livro é registado corretamente.
4. **Teste de Devolução de Livro:** Verificar se a devolução de um livro é registada corretamente.

**Resultados dos Testes:**

* + **Teste de Registo de Livro:** Sucesso
  + **Teste de Registo de Utilizador:** Sucesso
  + **Teste de Empréstimo de Livro:** Sucesso
  + **Teste de Devolução de Livro:** Sucesso

1. **Implantação**

**Plano de Implantação:**

* + **Data de Implantação:** 1 de janeiro de 2024
  + **Atividades:**
    - Configuração do ambiente de produção.
    - Migração dos dados existentes.
    - Formação dos utilizadores finais.
    - Monitorização inicial e suporte.

**Formação:**

* + **Sessões de formação:** 2 sessões de 2 horas cada para bibliotecários.
  + **Materiais de Apoio:** Manual do Utilizador e Vídeos Tutoriais.

**Suporte:**

* + **Suporte Técnico:** Disponível durante o horário comercial para resolução de problemas.
  + **Contato:** [suporte@biblioteca.com](mailto:suporte@biblioteca.com)

1. **Conclusão**

**Resumo dos Resultados:** O Sistema de Gestão de Biblioteca foi desenvolvido conforme os requisitos especificados, com todas as funcionalidades principais implementadas e testadas com sucesso. A interface amigável e o desempenho satisfatório foram alcançados, proporcionando uma ferramenta eficiente para a gestão de bibliotecas.

**Próximos Passos:**

* + Monitorização contínua do desempenho do sistema.
  + Recolha de feedback dos utilizadores para futuras melhorias.
  + Planeamento de novas funcionalidades, como a gestão de multas.

**Reflexão:** O projeto proporcionou uma valiosa experiência em desenvolvimento de software, destacando a importância de um planeamento detalhado e de uma comunicação eficaz entre a equipa. Lições aprendidas incluem a necessidade de testes rigorosos e a importância de um design centrado no utilizador.

1. **Anexos**
   * **Documento de Requisitos Funcionais e Não Funcionais**
   * **Diagrama de Casos de Uso**
   * **Diagrama de Classes**
   * **Modelo ERD**
   * **Planos de Teste e Resultados**
   * **Código de desenvolvimento**

Este relatório fornece uma visão abrangente do processo de desenvolvimento do Sistema de Gestão de Biblioteca em Python, documentando todas as fases desde o levantamento de requisitos até a implantação e suporte.