

Tópicos Avançado da Computação | Engenharia de Software

**Pdc.Ao**

Sistema de Gestão Para Clinica de Saúde Pdc.Ao

**Autores:**

Abel Gonçalo João

Frederico Kanda Fernando

Pedro Paulo Gomes Cunha

José Junqueira Faustino

**Docente:**

Mateus Padoca Calado, PhD.

Luanda, 2023

Sumário

[Introdução 4](#_Toc144967496)

[Parte I – Análise 4](#_Toc144967497)

[1. 1. Descrição do Projecto 4](#_Toc144967498)

[1. 2. C-Requisitos. 4](#_Toc144967499)

[1. 2.1. Requisitos Funcionais 4](#_Toc144967500)

[1. 2.1.1 Requisitos Não Funcionais 5](#_Toc144967501)

[1. 2.2. Casos de Uso 6](#_Toc144967502)

[1. 3. D-Requisitos: Requisitos Detalhados 11](#_Toc144967503)

[1. 3.1. Diagramas de Fluxos de Dados **Erro! Marcador não definido.**](#_Toc144967504)

[1. 3.2. Diagrama de Classes 13](#_Toc144967505)

[Parte II – Arquitectura 13](#_Toc144967506)

[1. 4. Arquitectura do Sistema 13](#_Toc144967507)

[Parte III – Planeamento 13](#_Toc144967508)

[1. 5. Recursos 16](#_Toc144967509)

[Tabela 20 Recursos de Hardware e Software 17](#_Toc144967510)

[1. 6. Estimação 17](#_Toc144967511)

[1. 6.1. Por Linhas de Código 17](#_Toc144967512)

[1. 6.2. Por Pontos de Função 18](#_Toc144967513)

[1. 6.3. Comparação 19](#_Toc144967514)

[1. 6.4. Modelos Empíricos (COCOMO) 20](#_Toc144967515)

[1. 7. Processo de Desenvolvimento de Software 21](#_Toc144967516)

[1. 8. Planeamento do Projecto 22](#_Toc144967517)

[1. 9. Gestão de Riscos 22](#_Toc144967518)

[1. 10. Análise Crítica 23](#_Toc144967519)

[Parte IV – Desenho 24](#_Toc144967520)

[1. 11. Desenho Detalhado 24](#_Toc144967521)

[Parte V – Implementação 24](#_Toc144967522)

[1. 12. Aplicação Final 24](#_Toc144967523)

[Parte VI – Testes 25](#_Toc144967524)

[1. 13. Plano de Testes 25](#_Toc144967525)

[Parte VII – Resultados 25](#_Toc144967526)

[1. 15. Medidas do Projecto 25](#_Toc144967527)

[1. 16. Planeado vs. Executado 25](#_Toc144967528)

[Parte VIII – Conclusões 26](#_Toc144967529)

[1. 17. Avaliação Post-Mortem e Síntese Crítica do Projecto 26](#_Toc144967530)

[Bibliografia 27](#_Toc144967531)

[Anexo 28](#_Toc144967532)

# 

# Introdução

O objectivo do trabalho é desenvolver um sistema baseado em Sistemas Web, utilizando o FrameWork Laravel, e como linguagem de programação o PHP, para a gestão e informações clinicas do grupo Pdc.Ao.

O Pdc.Ao consiste numa rede de instituições clinicas, distribuídas por em várias cidades. Os utilizadores do sistema Pdc.Ao, através de uma aplicação distribuída móvel, podem localizar instituições clinicas próximas com Exame/Consultas (ou vagas) disponíveis e, quando junto a uma instituição poder efetuar uma Consulta ou Exame para posteriormente a receber os resultados.

# Parte I – Análise

## 1. 1. Descrição do Projecto

O Pdc.Ao é composto por uma rede de instituições clinicas, distribuídas por uma área geográfica. Cada Instituição tem um conjunto de serviços disponíveis, bem como um conjunto de especialistas divididos por especialidades. Em cada momento, cada instituição pode ter horário livre ou ocupada para uma Consulta ou Exame.

## 1. 2. C-Requisitos: Requisitos dos usuários (Utente, P\_Admin, P\_Clinico)

### 1. 2.1. Requisitos Funcionais

**Utente**

* Efectua o Registo
* Efectua o Login
* Solicita Exame/Consulta
* Efectuar Exame/Consulta
* Entrega Exame/Consulta
* Lista Instituições
* Reserva Exame/Consulta
* Alterar dados pessoal no RCU
* Visualiza Especialidades, Médicos, e Diagnósticos

**P\_Admin**

* Define novas Instituições
* Define elimina Instituições
* Consultar estatísticas do sistema
* Efetuar criação do registo do pessoal Clinico
* Especifica os horários e as especialidades do pessoal clinico
* Efectua gestão das marcações de consultas e exames
* Enviar informações e notificações a utentes via email

**P\_Clinico**

* Logar no Sistema
* Verificar Registros de Consultas e Exames Vinculados a ele
* Emitir um diagnóstico
* Consulta e Insere dados do seu RCU

### 1. 2.1.1 Requisitos Não Funcionais

|  |  |
| --- | --- |
| # | Descrição |
| RNF01 | SEGURANÇA: O Sistema é acessível a qualquer pessoa, exigindo uma gestão de identidade e acesso para disponibilizar a informação. |
| RNF02 | OPERACIONALIDADE: No Sistema é garantida a continuidade das operações. Os recursos estarão disponíveis durante a utilização do sistema. |
| RNF03 | MANUTENIBILIDADE: O sistema vai ser facilmente atualizado e modificado, permitindo a correção de eventuais Bugs, e a adição de novas funcionalidades, bem como a documentação adequada para facilitar a manutenção. |
| RNF04 | ESCALABILIDADE: O sistema é capaz de processar várias requisições num dado intervalo de tempo sem comprometer os serviços. |
| RNF05 | CONFIABILIDADE: O sistema pode processar e apresentar correção e recuperação de erros. |
| RNF06 | SEGURANÇA: O sistema utilizará uma comunicação segura baseando-se no protocolo de segurança HTTPs. |
| RNF07 | COMPATIBLIDADE: O sistema é compatível com sistemas operativos Windows, Linux, Android, IOs, sendo acessado via Navegador Browser. |
| RNF08 | USABILIDADE: O sistema é fácil de usar, tem uma interface obedecendo os princípios de IPM bem como a experiência do utilizador com outros sistemas. |
| RNF09 | TESTABILIDADE: O sistema vai ser capaz de permitir que o software, quando modificado, seja validado. |

### 1. 2.2. Casos de Uso

Identificação e descrição dos actores. Diagramas de Casos de Uso (UML). Descrição textual dos casos de uso.

Fazer os desenhos

**Ilustração 1**: Diagrama de Caso de Uso (UML).

**1. 2.2.1 Definção dos Actores**

|  |  |
| --- | --- |
| **Actores** | **Função** |
| Utente | Efetua registro, procurar Instituições clinicas Disponíveis,  Efectuar Exame/Consulta, levantar Exame/Consulta, Consulta seu estado. |
| Admin | Consultar Estatísticas, Registrar Instituição, Elimina Instituição,Invoca serviços de Efectuamento de Exame/Consultas. |

**Tabela 1**: Descrição dos Actores.

* + - 1. **Des crição dos casos de uso – Utente.**

**Descrição do caso de uso Realizar Registo de Utente.**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Realizar Registo Utente** |
| Caso de Uso Geral |  |
| Ator Principal | Utente |
| Atores Secundários |  |
| Resumo | Este caso de uso descreve as etapas de registo do Utente, para que ele tenha a oportunidade de fazer marcar Exame/Consulta. |
| Pré-Condição | O Utente deve aceder o sistema online |
| Pós-Condição |  |
| Fluxo Principal | |
| **Ações do Actor** | **Ações do Sistema** |
| 1. Selecionar a registo; | 2. Exibe o formulário de registo. |
| **Fluxo Alternativo** | |
| **Ações do Actor** | **Ações do Sistema** |
| ------------------------------------ | ----------------------------------------------------------------------------- |

**Tabela 2:** Descrição textual do Caso de Uso Registo.

**Descrição do caso de uso Marcar Exame/Consulta.**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Marcar Exame/Consulta** |
| Caso de Uso Geral |  |
| Ator Principal | Utente |
| Atores Secundários |  |
| Resumo | Este Caso de Uso descreve as etapas percorridas por um Utente para realizar a marcação de Exame/Consulta. |
| Pré-Condição | O Utente deve estar Registado ou ñ no Pdc.Ao. |
| Pós-Condição |  |
| Fluxo Principal | |
| **Ações do Actor** | **Ações do Sistema** |
|  |  |
| 1. Selecionar a opção instituições clinicas disponíveis; |  |
|  | 2. Exibe as instituições clinicas com Exame/Consultas disponíveis e os médicos |
| 3. Selecionar as instituições clinicas com Exame/Consultas disponíveis; |  |
| 4. Selecionar a opção Marcar Consulta/Exame; |  |
|  | 5. Exame/Consulta Marcada com sucesso. |
| Fluxo de Exceção | |
| **Ações do Actor** | **Ações do Sistema** |
|  | 1. Caso já tivera Efectuarda uma Exame/Consulta emitir mensagem apropriada. |
| **Restrições/validações** |  |

**Tabela 3:** Descrição textual do Caso de Uso Reservar Exame/Consulta.

**Descrição do caso de uso Entregar Exame/Consulta.**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Entregar Exame/Consulta** |
| Caso de Uso Geral |  |
| Ator Principal | Utente |
| Atores Secundários |  |
| Resumo | Este Caso de Uso descreve as etapas percorridas por um Utente para realizar a Entrega de Exame/Consulta. |
| Pré-Condição | O Utente deve estar Registado no sistema. |
| Pós-Condição |  |
| Fluxo Principal | |
| **Ações do Actor** | **Ações do Sistema** |
| 1. Selecionar a opção instituições clinicas disponíveis; |  |
|  | 2. Exibe as instituições clinicas com Exame/Consultas disponíveis; |
| 3. Selecionar as instituições clinicas disponíveis; |  |
| 4. Selecione a opção Entregar; |  |
|  | 5. Exame/Consulta entregada com sucesso. |
| **Fluxo Alternativo** | |
| **Ações do Actor** | **Ações do Sistema** |
|  | 1. Caso já tivera Entregado uma Exame/Consulta emitir mensagem apropriada. |
| **Restrições/validações** |  |

**Tabela 4:** Descrição textual do Caso de Uso Entregar Exame/Consulta.

* + - 1. **Descrição dos casos de uso Consultar Saldo**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Consultar Diagnóstico** |
| Caso de Uso Geral |  |
| Ator Principal | Utente |
| Atores Secundários |  |
| Resumo | Este Caso de Uso descreve as etapas percorridas por um Utente para consultar o diagnóstico de um exame ou Consulta. |
| Pré-Condição | Estar cadastrado no Sistema |
| Pós-Condição |  |
| Fluxo Principal | |
| **Ações do Actor** | **Ações do Sistema** |
| 1. Selecionar a opção Meu Perfil; |  |
|  | 2. Exibe o diagnóstico disponível. |
| **Fluxo Alternativo** | |
| **Ações do Actor** | **Ações do Sistema** |
| ------------------------------------ | ----------------------------------------------------------------------------- |

**Tabela 5:** Descrição textual do Caso de Uso Consular Saldo.

**Descrição do caso de uso Listar Instituição.**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Listar Instituição** |
| Caso de Uso Geral |  |
| Ator Principal | Utente |
| Atores Secundários |  |
| Resumo | Este Caso de Uso descreve a lista de instituições clinicas do grupo pdc.ao |
| Pré-Condição |  |
| Pós-Condição |  |
| Fluxo Principal | |
| **Ações do Actor** | **Ações do Sistema** |
| 1. Selecionar a opção Minha trajectoria |  |
|  | 2. Exibe na tela a trajetória |
| **Fluxo Alternativo** | |
| **Ações do Actor** | **Ações do Sistema** |
| ------------------------------------ | ----------------------------------------------------------------------------- |

**Tabela 6:** Descrição textual do Caso de Uso Listar Instituiçãos.

**Tabela 7:** Descrição textual do Caso de Uso Recolocar Instituição.

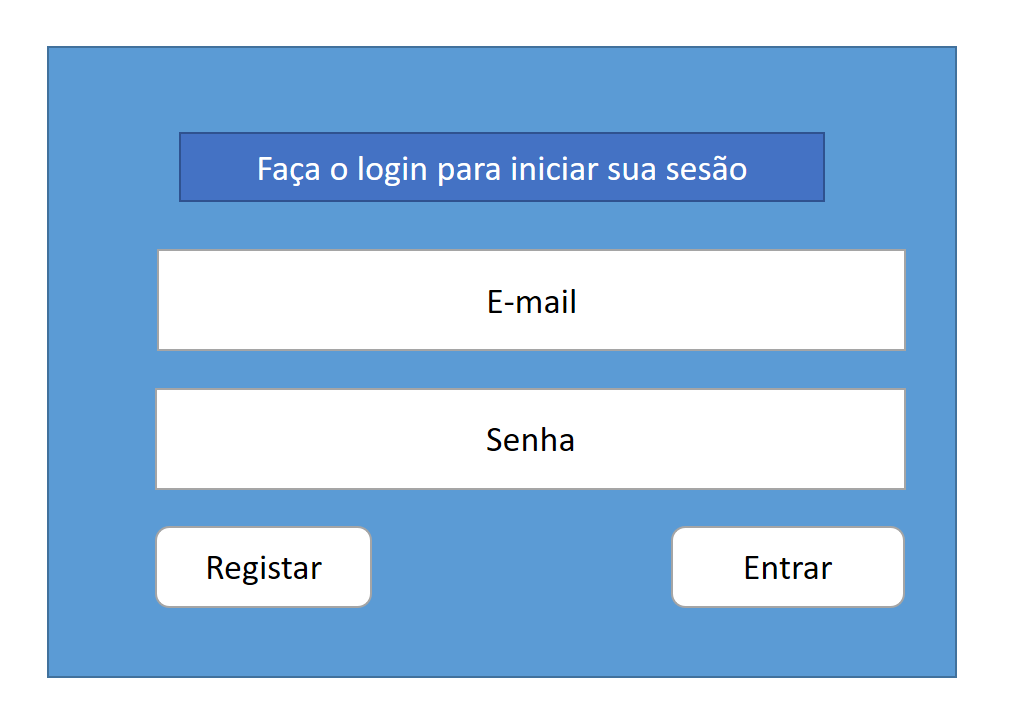
* + - 1. **Descrição dos casos de uso Eliminar Instituição**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Eliminar Instituição** |
| Caso de Uso Geral |  |
| Ator Principal | Admin |
| Atores Secundários |  |
| Resumo |  |
| Pré-Condição |  |
| Pós-Condição |  |
| Fluxo Principal | |
| **Ações do Actor** | **Ações do Sistema** |
| 1. Selecionar a opção Eliminar Instituição; |  |
| 2. Selecione a opção confirmar; |  |
|  | 3. Instituição eliminada com sucesso |
| **Fluxo Alternativo** | |
| **Ações do Actor** | **Ações do Sistema** |
|  | 1. Caso os dados não sejam validos emitir mensagem apropriada. |
| **Restrições/validações** | Nenhum campo de estar vazio |

**Tabela 8:** Descrição textual do Caso de Uso Eliminar Instituição.

**1. 2.3. Esboços das Interfaces de baixa fidelidade**

**2.3.1 Esboço login**

****

**Ilustração 2:** Esboço login.

## 1. 3. D-Requisitos: Requisitos Detalhados

Diagrama de Actividade Reservar Exame/Consulta

Diagrama de Sequencia Realizar registo de Utente

**Ilustração 7:** Diagrama de Sequência Registar.

Diagrama de Sequência Login

**Ilustração 8:** Diagrama de Sequência Login.

Diagrama de Sequência Reservar Exame/Consulta

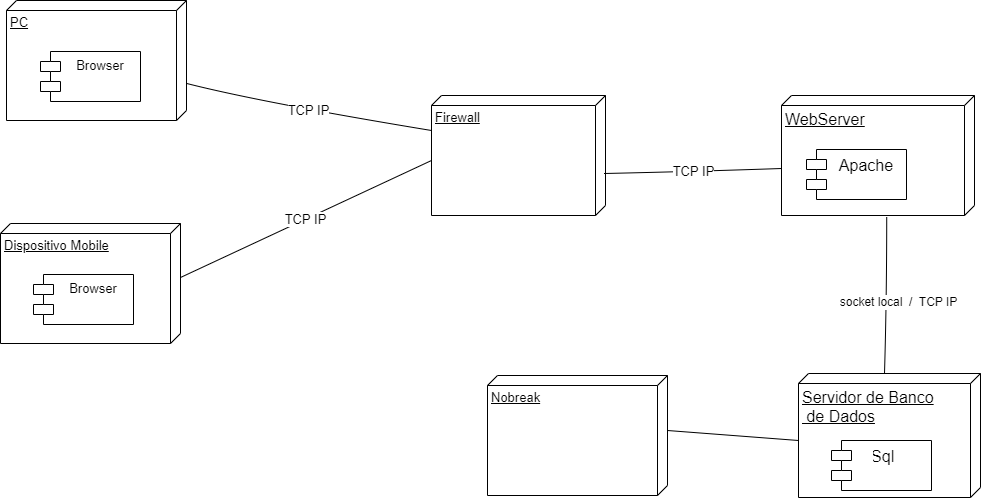
### 

### 1. 3.2. Diagrama de Classes

Diagrama de classes (UML) da base de dados, com apresentação das respectivas RIAS. Deve estar legível a correlação com a especificação de requisitos.

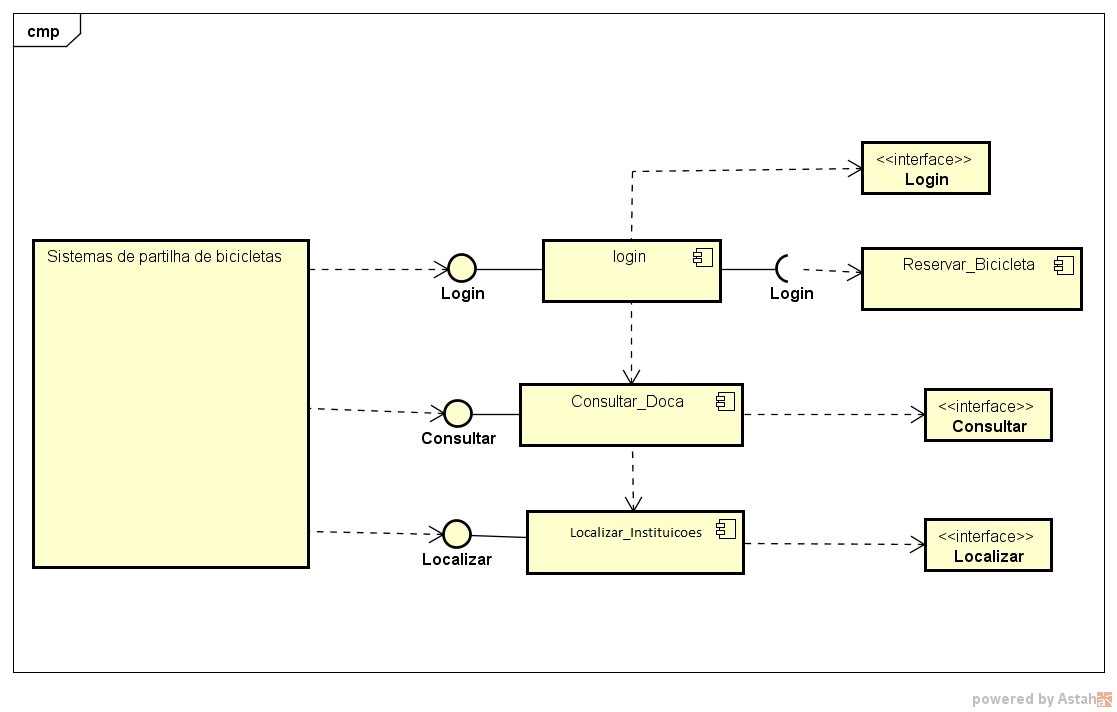
# Parte II – Arquitectura

## 1. 4. Arquitectura do Sistema

**Tabela :** Diagrama de Implantação

# 

**Tabela :** Diagrama de Componentes.



# Parte III – Planeamento

## 1. 5. Recursos

**1.5.1. Recursos humanos:**

|  |  |
| --- | --- |
| **Nome do Membro** | **Capacidades** |
| Abel João | Designer e documentação associada |
| Frederico Kanda | Fron-end, gestão do projecto e documentação associada |
| Pedro Paulo Gomes Cunha | Back-end, base de dados e intregração. |
| José Junqueira | Analise de sistema e documentação associada |

**Tabela 8:** Capacidades.

|  |  |  |  |
| --- | --- | --- | --- |
| **Função** | **Nome do Membro** | **Disponibilidade** | **Responsabilidades Específicas/Comentários** |
| Admin do Sistema | Abel João | 100% | Gere a equipa, Atribui tarefas, Supervisiona |
| Desenvolvedores | Abel João Pedro Cunha Frederico Kanda | 100% | Codifica as regras de negócio do sistema |
| Administração de Banco de Dados / Gerenciador de Banco de Dados | Pedro Cunha, Abel João | 100% | Assegura que os activos e o ambiente de dados de teste (Banco de dados) são gerenciados e mantidos, adminstrar os dados de teste m |
| Designer | Abel João | 100% | Prototipa as telas do sistema |
| Product Owner | Frederico Kanda | 98% | Analisa as regras de negócios e transforma em diagramas, executa testes, regista resultados e documenta anomalias(defeitos). |

**Tabela 9**: Recursos humanos.

**1.5.2. Recursos de Hardware e Software**

|  |  |  |
| --- | --- | --- |
| **Recursos** | **Ferramentas** | **Versão/Feito** |
|  | Boostrap | Versão 5.0 |
| Gerenciamento de Projeto | Draw.io, Microsoft Word, Asth Profissional, | Online |
| 2016 |
| Versão 3.30 |
| Ferramentas de Base de Dados (SGBD) | SQLSERVER | Versão 12 |
| Framework | Laravel, Asp.Net | Versão 9, 7 |
| Editor de Código | Visual Studio code | Versão 1.74.3 |
| Linguagem | C#, Php | Versão 7 |

# Tabela 20 Recursos de Hardware e Software

## 1. 6. Estimação

### 1. 6.1. Por Linhas de Código

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Componente** | **Estimativa Optimista** | **Estimativa Provável** | **Estimativa Pessimista** | **Valor Esperado** |
| Interface com o utilizador | **2200LOC** | **3400LOC** | **5000LOC** | **3467LOC** |
| Suporte das regras de negócio | **2800LOC** | **3000LOC** | **4500LOC** | **3217LOC** |
| Controlo de periféricos | **200LOC** | **245LOC** | **280LOC** | **243LOC** |
| **Número total esperado de linhas de código do software** | | | | **6921LOC** |

**Tabela 11**: Linhas de código.

### 1. 6.2. Por Pontos de Função

Tabelas com os pontos de função não ajustados. Tabelas com os factores de ajuste de Complexidade. Cálculo do valor final.

Para os processos, apenas é necessário apresentar os cálculos das complexidades médias e altas.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Componentes** | **Peso Baixo** | **Peso Medio** | **Peso Alto** | **Total** |
| Entrada Externas (EE) | 3 \* 3 = 9 | 0 \* 4 = 0 | 0 \* 6 = 0 | 9 |
| Saídas Externas (SE) | 0 \* 4 = 0 | 0 \* 5 = 0 | 0 \* 7 = 0 | 0 |
| Ficheiros Internos (ALI) | 3 \* 7 = 21 | 0 \* 10 = 0 | 0 \* 15 = 0 | 21 |
| Interfaces Externas (AIE) | 0 \* 5 = 0 | 0 \* 7 = 0 | 0 \* 10 = 0 | 0 |
| Total de pontos de função não ajustados (PFNA) = 30 | | | | |

**Tabela 13**: Os pontos de função não ajustados.

###### **Tabelas com os factores de ajustes de complexidade**

|  |  |  |  |
| --- | --- | --- | --- |
| Necessárias copias de segurança? | **0** | Os ficheiros originais são atualizados online? | **0** |
| Necessárias redes de comunicação | **3** | As entradas, saídas e ficheiros são complexos? | **3** |
| Necessário processamento distribuído | **1** | O processamento interno é complexo | **2** |
| Desempenho é  critico | **4** | O código deve ser reutilizável | **4** |
| Pode ocorrer num ambiente já em uso? | **3** | Necessária conversão de formatos antigos e utilitários de instalação? | **0** |
| Necessária entrada de dados online? | **2** | Suporte para múltiplas instalações em diversas organizações | **0** |
| Necessários múltiplos écrans para a introdução de dados | **0** | Facilidade de alteração pelo utilizador | **3** |
| Factor de ajuste de complexidade (FAC) | | | **25** |

**Tabela 14:** Factores de ajustes de complexidade.

Escala subjectiva (0= nenhuma importância, até 5 = essencial)

FAC = SUN (NI)

PF= PFNA\* (0, 65+ 0, 01\*FAC)

PF= 30\*(0, 65 +0, 01\*25)

PF= 30\*0,9

PF=27

### 1. 6.3. Comparação

A linguagem é JAVA e na tabela de complexidade diz que 1 PF na linguagem JAVA equivale a 58 Loc.

**Passo 1:** converter LOC em pf

1PF -------------------58Loc

X -------------------------6921 Loc

X= (6921Loc \* 1pf )/ 58Loc

X=119,32pf

**Passo 2:** Calcular Esforço

Albrecht e Yaffney

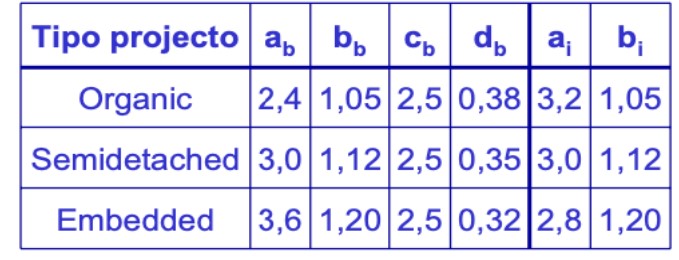
E= -13,39 + 0,0545 \* pf

E= -13,39(0,65+0,01\*25) + 0,0545 \* 119,32

E= -6 p.mes

Logo é diferente do modelo de COCOMO onde obteve um esforço de 19 p.mes.

### 1. 6.4. Modelos Empíricos (COCOMO)



**Tabela 15:** Modelo Empíricos.

1-Cálculo do Esforço e da Duração do projecto por ponto de Linha de código 6921 na linguagem Java

1. Calcular Esforço

KLOC=6921/1000

KLOC=6921 = 7

Esforço= ab \* KLOC^(bb)

Esforço= 2,4\* 7^(1,05)

Esforço= 18,5 = 19 p.mes

1. Calcular Duração

Duração = cb\* Esforço^(db)

Duração = 2,5\* 19^(0,38)

Duração = 7,65= 8 meses

## 1. 7. Processo de Desenvolvimento de Software

O Modelo de Processo de Desenvolvimento de Software escolhido é o **RUP** (Rational Unified Process) é um processo Unificado de desenvolvimento de software. Porque Processo Unificado Racional, uma metodologia completa aplicada ao desenvolvimeno de software. Encontra-se fortemente integrado com o UML. O RUP segue um modelo iterativo e incremental, centrado numa arquitectura e conduzido por casos de uso. É uma metodologia que se baseia no paradigma da orientação por objectos, e abrangência das suas propostas contempla a possiblidade de adaptação à realidade de cada projecto ou organização.

O modelo foi implementado por fases:

* Concepção: Delimita o âmbito do projecto e se enquadra o produto nos outros sistemas com que vai interagir;
* Elaboração: Fez-se a análise do domínio do problema, estabelece-se a arquitectura do sistema que é microserviços;
* Construção: Concluir o sistema para os testes exaustivos ao produto;
* Transição: Está fase tem como objectivo a transferência do produto para a comunidade de utilizadores.

## 1. 8. Planeamento do Projecto

Decomposição de tarefas e sua breve descrição. Afectação das tarefas aos recursos humanos.

* Planeamento: Identificar as necessidades, selecionar e definir o plano de trabalho. (Duração 11 dias) ;
* Análise dos Requisitos: O estudo detalhado do domínio do problema. (Duração 7 dias);
* Desenho do sistema: O desenho procede à especificação formal das características que a implementação do sistema deverá apresentar. (Duração 5 dias);
* Desenvolvimento do sistema: A tarefa de desenvolvimento inclui todas as actividades de implementação do sistema. (Duração 35 dias);
* Testes e Integração: avaliar a adequada correcção e funcionamento de todos os componentes do sistema. (Duração 10 dias);
* Instalação: Esta fase consiste na preparação e instalação do sistema na infraestrutura computacional destino. (Duração 1 dias);
* Ponto critico: São todas as tarefas que impedem o início da outra tarefa. (Duração 11 dias) ;

## 1. 9. Gestão de Riscos

Identificação e descrição dos riscos e suas consequências. Tabela com os riscos, a sua categoria, a sua probabilidade e o seu impacto, com escalas devidamente definidas. Plano RMMM.

**Impacto:** 1 – Sem gravidade; 2 – pouca grave; 3 – grave; 4 – Muito grave; 5-Extremamente Grave.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Riscos** | **Categoria** | **Probalidade** | **Impacto** | **RMMM** |
| Número de usuários maior que o planeado | Tecnologia | 10% | 5 | Verificar se o provedor de hospedagem permite lhe dar com grande volumes de trafego |
| O sistema não atinge as expectativas do Utente | Pessoal | 50% | 2 | Comunicação ou reuniões constates entre a equipa e o Utente |
| - Não comprimento do prazo de entrega | Cronograma | 35% | 5 | Desenvolver um cronograma para o desenvolvimento do projecto |
| Pessoal sem experiência;  Equipa insuficiente | Pessoal | 30% | 3 | Melhorar a comunicação na equipa entendendo as possíveis causas |
| Falta de energia electrica | Factor social | 98% | 5 | Ter um gerador, como alternativa |

**Tabela 17**: Gestão de Riscos.

## 1. 10. Análise Crítica

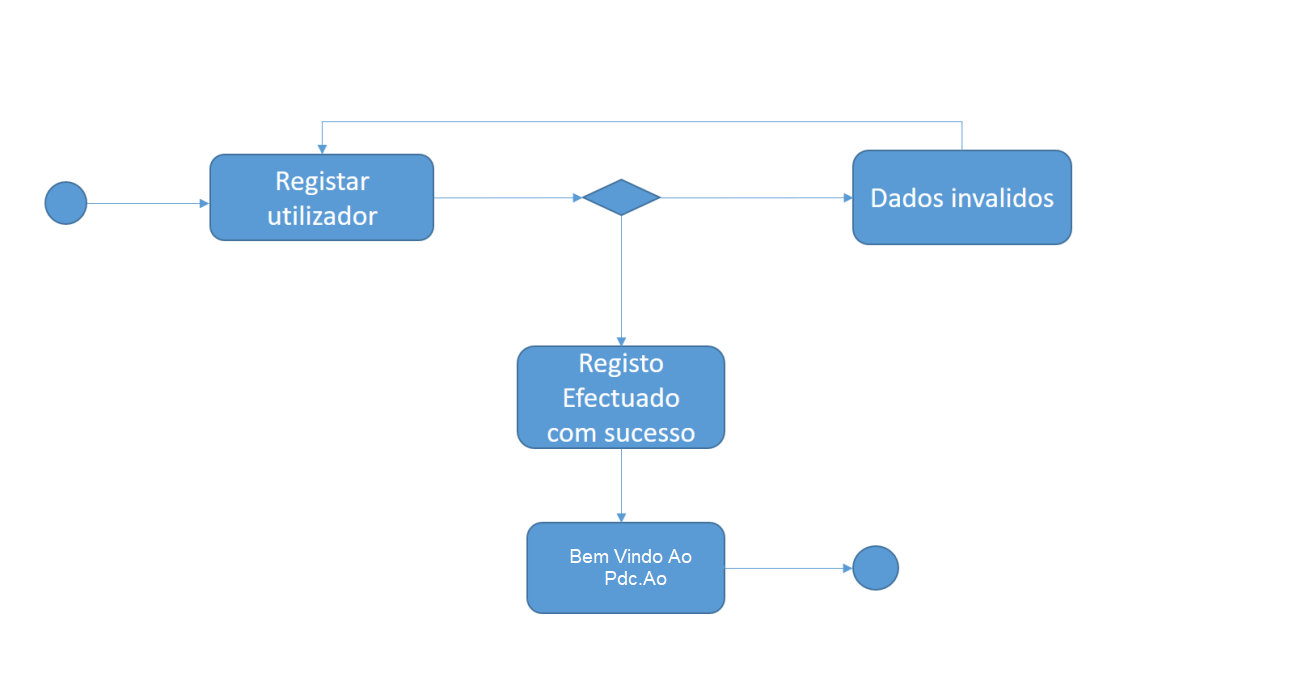
Em ponto de partida para o desenvolvimento de software foi de interesse chamar atenção para a questão de implementação do um sistema .

Em analise aos aspectos negativos do sistema referisse ao facto de não haver internet disponível sempre que necessário pós e um software website e carece sempre de acesso online.

Em análise aos aspectos positivos esse sistema garantira aos Admin obterem informações sobre o sistema de gestão de uma plataforma de Exame/Consultas partilhadas a qualquer hora, momento e a qualquer lugar.

# Parte IV – Desenho

## 1. 11. Desenho Detalhado



# Parte V – Implementação

## 1. 12. Aplicação Final

Imagens com a interface do sistema, exemplificando as principais funcionalidades. Comentários sobre os aspectos considerados mais significativos na sua implementação.

# Parte VI – Testes

## 1. 13. Plano de Testes

Teste funcional ou Teste da caixa Preta:

-No sistema de gestão de uma plataforma de partilha de Exame/Consultas.

**Classes equivalência**

Classe Pessoa: Pessoa pode ser Utente e Admin;

* Se pessoa for Utente deve informar os seguintes dados:.
* Utente - idade 18-75 (idade)
* Admin 18-75(idade)

**1. 14. Resultado dos Testes**

Descrição dos resultados obtidos. Comentários.

# Parte VII – Resultados

## 1. 15. Medidas do Projecto

O esforço real despendido por realização do projecto foi aproximadamente de 2 mês e duas semanas. Alcançou-se um total ou aproximadamente 6921 de linhas de códigos codificadas. Em relação o que havia sido estimado não há uma grande diferença, pois os riscos e até mesmo o RMMM foram alcantilados bem antes de acontece-lo.

## 1. 16. Planeado vs. Executado

Nível de implementação de cada funcionalidade prevista para o sistema. Por exemplo, autenticação foi implementada a 100%.

|  |  |
| --- | --- |
| **Funcionalidade (Planeadas)** | **Nível de implementação em % (Executado)** |
| Enviar/receber pontos entre dispositivos moveis | 100% |
| Enviar/receber mensagens da texto entre dispositivos | A Definir |
| Registar utilizador | 100% |
| Registar Utilizador | 100% |
| Enviar Nova trajectoria | A Definir |
| Mostrar a trajectoria passada e mais recente no mapa | A Definir |
| Obter informações do utilizador | 100% |
| Obter lista de instituições clinicas com Exame/Consultas disponíveis | 100% |
| Reserve a Exame/Consulta na Instituição especifica | A Definir |
| Lidar com enventos de Efectuarmento/ devolução de Exame/Consultas | 100% |
| Detectar qual Exame/Consulta está em uso por cada utilizador | A Definir |
| Segurança | 100% |
| Robustez | 100% |
| Outras funcionalidades | 100% |

**Tabela 20**: Planeado vs. Executado.

# Parte VIII – Conclusões

## 1. 17. Avaliação Post-Mortem e Síntese Crítica do Projecto

Para concluir, podemos observar que a **avaliação** de um **projeto** depende da formulação de indicadores de performance e de outras métricas para sabermos se realmente houve sucesso ou não.

Em analise global da duração do projecto precisamos de 41 dias uteis distribuídos em tarefas de desenvolvimento de software que na sua maioria não tivemos êxito na conclusão ao prazo dos dias estipulados e em linhas de codigo refletiu-se em uma framework que não era de domínio de todos elementos do grupo que causou uma mínima dificuldade na codificação.

A documentação foi um ponto de maior dificuldade para poder preencher todos os itens da documentação principalmente os diagramas, os testes, as estimativas de métricas do projecto.

As funcionalidades que foram mais requerentes são: Efectuar Exame/Consulta, Devolver Exame/Consulta, exigiu mais criatividade da equipa.

Avaliação da equipa foi confiada ao supervisor em que cada elemento tinha um prazo para entregar as tarefas feitas e assim quem não fizesse seria punido e tornou a equipa mais produtiva. Mas no meio do processo tivemos a desistência de dois elementos que abalou e levou um tempo até que a equipa se recompões. Distribuição de tarefas na equipa: 2 Programadores, 1- Design, 1- Documentação.

Os factores positivos no desenvolvimento do software foi o gerenciamento do projecto na distribuição de tarefas e o trabalho em equipa com dias de reuniões para abortar assuntos relevantes para o avanço do projecto. Os factores negativo foi a falta de compromisso da equipa, equipa incompleta, e algumas vezes o incumprimento de prazos.

As mudanças que faríamos estão relacionadas ao gerenciamento de equipa aplicado mais rigor na cobrança de tarefas e no desenvolvimento do código do software e reuniões constantes de equipa.

O interessante foi análise de requisito aplicando as regras de negócio e o aprendizado na area de Aplicações e Serviço na Web, Desenvolvimento de Aplicações Moveis e Sistemas Distribuídos, fazer a documentação do sistema informação com número avultado de passos a cumprir que não levaríamos muito em conta no decorrer do desenvolvimento do sistema.

O fim do projeto foi sem dúvida tão claro quanto se gostaria.

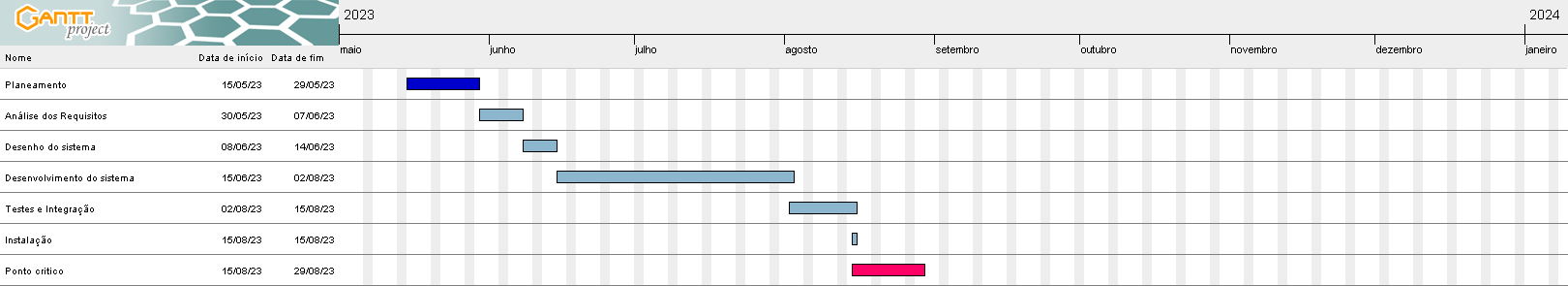
## Bibliografia

[0] UML2. Abordagem Pratica.

1. Braude, Eric J., *Software Engineering: An Object-Oriented Perspective*, John Wiley & Sons, 2001, ISBN 0471322083.
2. .https://express.adobe.com/pt-BR/sp/design/post/urn:aaid:sc:EU:0a75284f-b07c-4e23-858e-b36ec527edfc.
3. BOOCH GradyUML : Guia do Usuário, O mais avançado tutorial  
   sobre Unified Modeling Language.
4. Ragget, D., *Getting started with HTML*, W3C, May 2005.

http://www.w3.org/MarkUp/Guide/ .

# Anexo



**GLOSSÁRIO**

* **PFNA**- pontos de função não ajustados
* **EE-** entradas externas
* **CE-** consultas externas
* **SE-**saídas externas
* **ALI**-arquivos lógicos internos
* **AIE**-arquivos de interface externas
* **PF**-Ponto de função (Ajustado)
* **FAC** - Factor de ajuste de complexidade
* **SqlServer** - É um sistema de gerenciamento de banco de dados, que utiliza a linguagem SQL como interface.
* **Visual Studio Code** é um editor de código-fonte desenvolvido pela Microsoft para Windows, Linux e macOS. Ele inclui suporte para depuração, controle de versionamento Git incorporado, realce de sintaxe, complementação inteligente de código, snippets e refatoração de código.
* **UML** é uma linguagem-padrão para a elaboração da estrutura de projetos de software. Ela poderá ser empregada para a visualização, a especificação, a construção e a documentação de artefatos que façam uso de sistemas complexos de software.
* **API** Interface de Programação de Aplicações, é um conjunto de rotinas e padrões estabelecidos por um software para a utilização das suas funcionalidades por aplicativos que não pretendem envolver-se em detalhes da implementação do software, mas apenas usar seus serviços.
* **HTML** é uma linguagem de marcação utilizada na construção de páginas na Web. Documentos HTML podem ser interpretados por navegadores.
* **CSS** Cascading Style Sheets é um mecanismo para adicionar estilo a um documento web. O código CSS pode ser aplicado directamente nas tags ou ficar contigo dentro das tags <style>.