**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**

**PUC Minas Virtual**

**Pós-graduação *Lato Sensu* em Engenharia de *Software***

Projeto Integrado

Relatório Técnico

PUCfit

Lucas Ribeiro Cardoso

Belo Horizonte

Maio de 2023.

# Projeto Integrado

**Sumário**

Projeto Integrado 3

1. Cronograma de Trabalho 4

2. Introdução 5

2.1 Objetivos do Trabalho 5

2.2 Apresentação do problema 5

2.3 Descrição geral do software 5

3. Definição Conceitual da Solução 7

3.1 Diagrama de Casos de Uso 7

3.2 Requisitos Funcionais 7

3.3 Requisitos Não-funcionais 9

4. Protótipo Navegável do Sistema 9

5. Diagrama de Classes de Domínio 10

6. Arquitetura da Solução 10

6.1 Padrão Arquitetural 10

6.2 C4 model - Diagrama de Contexto 11

## Cronograma de Trabalho

|  |  |  |  |
| --- | --- | --- | --- |
| **Datas** | | **Atividade / Tarefa** | **Produto / Resultado** |
| **De** | **Até** |
| 03/04/2023 | 05/04/2023 | 1. Revisão de todo o material disponibilizado na disciplina do Projeto Integrado (textos e vídeos). | Entendimento geral do Projeto Integrado. |
| 06/04/2023 | 07/04/2023 | 2. Mapeamento da documentação de escopo para identificação de atores, requisitos e outras informações necessárias para o início do Projeto. | Lista de atores, requisitos e outras informações importantes. |
| 07/04/2023 | 08/04/2023 | 3. Desenvolvimento dos Objetivos do Trabalho, Apresentação do Problema, Descrição Geral do Software. | Definição conceitual / introdução. |
| 08/04/2023 | 09/04/2023 | 4. Identificação de ferramenta e desenvolvimento de diagrama de Caso de Uso. | Diagrama de Caso de Uso. |
| 11/04/2023 | 13/04/2023 | 5. Criação do Protótipo de Interface navegável e seus itens relacionados. | Protótipo navegável. |
| 24/04/2023 | 25/04/2023 | 6. Identificação de ferramenta e desenvolvimento de diagrama de Classes de Domínio. | Diagrama de Classes de Domínio. |
| 26/04/2023 | 26/04/2023 | 7. Definição do Padrão Arquitetural do Projeto. | Descrição do Padrão Arquitetural definido para o Projeto. |
| 27/04/2023 | 28/04/2023 | 8. Identificação de ferramenta e desenvolvimento de diagrama de Contexto. | Diagrama de Contexto. |

## Introdução

***2.1. Objetivos do trabalho***

O objetivo geral deste Trabalho é desenvolver uma aplicação que forneça um serviço completo às academias - mais especificamente às academias de CrossFit.

Inicialmente, a aplicação será desenvolvida somente para mobile (Android), onde tem-se uma única interface que pode ser acessada, pelos Empregados do estabelecimento e seus Clientes (Aluno), essa interface fornecerá todos os recursos necessários para ambos.

***2.2. Apresentação do problema***

Com o avanço tecnológico, a popularização dos Smartphones e o acesso à Internet, aplicativos e softwares vêm ganhando espaço para facilitar o cotidiano de empresas e usuários.

Pensando nessas resoluções criadas para facilitar o dia a dia, foi identificada a necessidade da criação de uma solução para facilitar tanto os processos internos de quanto o atendimento ao público de academias de CrossFit. A aplicação tem em vista facilitar as rotinas de trabalho, especialmente dos Professores, no quesito de ter um controle de seus alunos e aulas de maneira mais ágil, possibilitando assim, um maior contato com seus alunos (Clientes) durante as aulas.

Pelo lado do cliente, a disponibilização de uma plataforma simples e objetiva, onde é possível visualizar as aulas que serão ministradas para ele e, caso não tenha tempo disponível para se deslocar à academia e realizar as atividades, possa reproduzir os exercícios mais simples em sua casa. Será possível, também, visualizar sua presença, realizar pagamentos, consultas etc.

***2.3. Descrição geral do software***

Esse software tem por objetivo fornecer uma aplicação que possa conectar o estabelecimento ao Empregado (mais especificamente, Professor) e a um Cliente (Aluno), facilitando assim todo o processo.

O Empregado (Recepcionista) será o responsável pelo cadastramento do Empregado (Professor) em eventual contratação e Cliente (aluno) quando aderir ao plano da academia. Registrará os períodos de férias dos funcionários, pagamentos etc.

Os Empregado (Professor), além de outras funcionalidades, terá o foco mantido em: realizar o cadastro de alunos (para o controle pessoal) e disponibilizar os scripts de suas aulas.

O Cliente (Aluno) poderá fazer o cadastro na plataforma, criando seu perfil e poderá fazer consultas de suas aulas, presenças, pagamentos etc.

## Definição Conceitual da Solução

## Diagrama de Casos de Uso

|  |
| --- |
|  |

## Requisitos Funcionais

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Descrição Resumida** | **Dificuldade (B/M/A)\*** | **Prioridade**  **(B/M/A)\*** |
| RF01 | Um usuário (Recepcionista) deve ser capaz de cadastrar um novo usuário (Aluno, Professor) | B | A |
| RF02 | Um usuário deve ser capaz de realizar o login em seu perfil como (Empregado\*\*) ou (Aluno) | B | A |
| RF03 | Um usuário (Recepcionista) deve ser capaz de registrar o período de férias dos funcionários | A | A |
| RF04 | Um usuário (Aluno) deve ser capaz de visualizar seus pagamentos | M | M |
| RF05 | Um usuário (Professor) deve ser capaz de cadastrar novos alunos para seu controle | B | A |
| RF06 | Um usuário (Professor) deve preencher todas as informações para cadastrar alunos em seu controle | B | M |
| RF07 | Um usuário (Professor) deve ser capaz de listar todos os alunos em seu controle | B | M |
| RF08 | Um usuário (Professor) deve ser capaz de excluir um aluno em seu controle | M | A |
| RF09 | Um usuário (Professor) deve ser capaz de ver todas as informações de seus alunos em seu controle | M | B |
| RF10 | Um usuário (Professor) deve ser capaz de cadastrar um script da aula | M | A |
| RF11 | Um usuário (Professor) deve ser capaz de listar todos os scripts de suas aulas | B | M |
| RF12 | Um usuário (Aluno) deve ser capaz confirmar a presença na aula | M | M |
| RF13 | Um usuário (Aluno) deve ser capaz de cancelar a presença na aula | B | B |
| RF14 | Um usuário (Professor) deve ser capaz de verificar a presença dos alunos por aula | A | M |
| RF15 | Um usuário (Aluno) deve ser capaz de ver suas presenças | B | M |
| RF16 | Um usuário (Aluno) deve ser capaz de ver os scripts de aula | B | M |
| RF17 | Um usuário (Recepcionista), ao receber a confirmação de pagamento de mensalidade do usuário (Aluno), deve ser capaz de registrar o pagamento no sistema | M | A |
| RF18 | Um usuário (Aluno) deve ser capaz de marcar consultas | A | A |
| RF19 | Um usuário (Aluno) deve ser capaz de ver a data da próxima consulta que está agendada | B | B |
| RF20 | Um usuário (Recepcionista) deve ser capaz de gerar um aviso/comunicado | M | M |

\* B = Baixa, M = Média, A = Alta.

\*\* Empregado = Recepcionista, Instrutor/Professor e Gerente.

## Requisitos Não-funcionais

|  |  |  |
| --- | --- | --- |
| **ID** | **Descrição** | **Prioridade**  **B/M/A** |
| RNF01 | O sistema deve apresentar um tempo de resposta abaixo de 2 segundos no processamento de todas as operações, com exceção do login | M |
| RNF02 | O sistema deverá executar em plataforma Android | A |
| RNF03 | O sistema deve estar disponível em qualquer período, funcionando em um regime 24/7 | A |
| RNF04 | O sistema deve ser dimensionado para suportar, pelo menos, 70 usuários conectados simultaneamente | M |
| RNF05 | O sistema deve garantir a segurança das senhas dos usuários (Alunos), criptografando-as ao serem inseridas no Banco de Dados | A |
| RNF06 | O sistema deve ser de fácil usabilidade, possibilitando assim a agilidade do manuseio | B |

## Protótipo Navegável do Sistema

**Link do vídeo (Google Drive):**

<https://drive.google.com/drive/folders/1AI3_vGEiu525WaY6OCDKVuE8e_iQsTYy?usp=sharing>

**Link do protótipo navegável:**

<https://www.figma.com/proto/DcBsEr2SjKyRykNklY3QS2/PUCfit?node-id=27-1341&starting-point-node-id=27%3A1341>

**Link das telas (draft):**

<https://www.figma.com/file/DcBsEr2SjKyRykNklY3QS2/PUCfit?type=design&node-id=0-1&t=8lD2FCYrwpvoSYR6-0>

## Diagrama de Classes de Domínio

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente com confiança média

## Arquitetura da Solução

## Padrão Arquitetural

O Padrão arquitetural escolhido para o projeto foi o MVC, sendo que a arquitetura foi modularizada utilizando TDD, em alguns Domínios não foi necessária a criação da camada de View, e a camada Model foi mantida com a mesma nomenclatura. Por fim, foi-se utilizado Adapter para auxiliar as Views.

## C4 model - Diagrama de Contexto

Tem-se abaixo, o Diagrama de Contexto da Aplicação, onde é apresentado a macro arquitetura desta.

Os usuários, podendo ser tanto um Empregado como um Cliente, que acessa o aplicativo Mobile através de seus dispositivos Android, se comunica com uma API que realiza a autenticação do usuário e, ambos por sua vez, se comunicam com a Bases de Dados, onde são persistidas todas a informações referentes a Aplicação.

Diagrama

Descrição gerada automaticamente