**Documento de Requisitos de Software**

**Site de instruções para a Academia**

Versão Beta 0.1

**Desenvolvedores/Analistas**

Gabriel Batista dos Santos

Raní de Souza Bezerra

Lucas Chaves de Souza

**Rio Branco – AC**

**2022**

**Histórico de Alterações**

| **Data** | **Versão** | **Descrição** | **Autor** |
| --- | --- | --- | --- |
| 26/05/2022 | 0.1 | Criação do documento | Gabriel Batista, Raní de Souza, Lucas Chaves |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. **Análise do Problema**

Tendo em visto o pouco uso de tecnologia para instruir alunos ingressantes e residentes da academia, a quantidade pequena de instrutores, os horários de pico, o uso de papel como lembrete das instruções e avaliações biomecânicas, todos esses problemas somados juntos cultivavam a ideia de criar uma solução.

1. **Necessidades Básicas do Cliente**

O cliente necessita que haja uma interface onde poderá administrar os treinos e modificá-los, sendo assim tendo um controle maior dos ingressantes e residentes.

1. **Estudo de Viabilidade**

**3.1. Viabilidade Técnica**

Visto que o cenário atual da academia não possui uma estrutura de software adequada para suas atividades, é proposto um sistema para simplificar/agilizar o processo de gerenciamento de treinos dos clientes

O sistema será desenvolvido de forma simples e objetiva, utilizando tecnologia já existente (Django). Iremos focar exclusivamente na resolução principal do problema.

**3.2. Viabilidade Econômica**

O custo será o tempo necessário para desenvolvimento do software, tendo em vista as etapas e os processos que o grupo irá enfrentar na hora de desenvolvê-lo, vale ressaltar o custo de manter o software atualizado e suporte que ele precisa.

Já o custo benefício para empresa será a redução do uso dos papéis, a perda de tempo para produzir os treinos.

**3.3. Viabilidade Legal**

O software será desenvolvido com ferramentas open source.

1. **Missão do Software**

O software a ser desenvolvido deverá agilizar o processo de cadastro/consulta de clientes, armazenamento de características do treino e/ou do cliente; além de automatizar processos triviais da academia.

1. **Limites do Sistema**

| **ID** | **Funcionalidade** | **Justificativa** |
| --- | --- | --- |
| **L1** | O sistema irá computar apenas os treinos dos clientes. | Visto que irão existir dificuldades e por conta do tempo, iremos focar no principal problema que são os treinos estarem no papel. |

1. **Benefícios Gerais**

| **ID** | **Benefício** |
| --- | --- |
| **B1** | Facilidade na mudança e criação dos treinos para o instrutor. |
| **B2** | Controle dos treinos e dos alunos que frequentam a academia. |
| **B3** | Anulação de perdas de treino. |
| **B4** | Facilidade para os clientes e o instrutor conferirem os treinos. |

1. **Restrições**

| **ID** | **Restrição** | **Descrição** |
| --- | --- | --- |
| **R1** | Não irá computar o pagamento. | O sistema não irá se associar ao controle de pagamento do cliente. |
| **R2** | Não será um site de divulgação da empresa. | O sistema irá apenas servir como um suporte para a empresa. |

1. **Atores**

| **ID** | **Atores** | **Descrição** |
| --- | --- | --- |
| **A1** | Usuário | Vai poder visualizar o treino. |
| **A2** | Funcionário | Vai poder modificar/visualizar o treino. |

1. **Requisitos Funcionais**

| **ID** | **Funcionalidade** | **Necessidades** | **Classificação** |
| --- | --- | --- | --- |
| **RF1** | Interface do Cliente | O cliente terá a possibilidade de acessar seus próprios treinos | Alta |
| **RF2** | Interface do Instrutor | O instrutor poderá acessar fazendo uma pesquisa pelo nome do cliente, podendo assim alterá-la | Alta |

1. **Requisitos Não-Funcionais**

| **ID** | **Requisitos** | **Categoria** | **Classificação** |
| --- | --- | --- | --- |
| **NRF1** | Haverá espaço do usuário e do cliente | Apresentação | Normal |
| **NRF2** | Sistema será desenvolvido usando o plugin de Python: Django | Codificação | Alto |

1. **Requisitos de Hardware**

**11.1. Configuração Mínima**

**Sistema Operacional:** Windows, Linux, MAC OS

**Processador:** Dual Core 1.2 Ghz

**Ram:** 2 GB

**11.2. Configuração Recomendada**

**Sistema Operacional:** Windows, Linux, MAC OS

**Processador:** Dual Core 1.2 Ghz

**Ram:** 4 GB

1. **Ferramentas de Desenvolvimento e Licença de Uso**
2. **Pycharm 2022.1.1** - Community:
3. **Visual Studio Code** - Community:
4. **SqlLite3** - Opens Source:
5. **GitHub** - Open Source: