

SoftFit

VISÃO DO PRODUTO E PROJETO

Versão [\[1.0\]](#)

Histórico de Revisão

Data	Versão	Descrição	Autor
31/01/2022	1.0	Primeira versão do documento.	Victor Hugo Oliveira Leão; Felipe Alef Pereira Rodrigues; Luan Vasco Cavalcante; Arthur Heleno do Couto Silva; Marcos Well Neres Silva Pereira

Sumário

1	<i>VISÃO GERAL DO PRODUTO</i>	<i>4</i>
1.1	Declaração do Problema.....	4
1.2	Declaração de Posição do Produto	4
1.3	Objetivos do Produto.....	5
2	<i>ABORDAGEM DE DESENVOLVIMENTO DE SOFTWARE</i>	<i>5</i>
2.1	Metodologia	5
2.2	Processo.....	6
2.3	Procedimentos	6
3	<i>LIÇÕES APRENDIDAS</i>	<i>6</i>
3.1	Unidade 1	6
3.2	Unidade 2	6
3.3	Unidade 3	7
3.4	Unidade 4	7
3.5	Unidade 5	7
4	<i>REFERÊNCIAS BIBLIOGRÁFICAS.....</i>	<i>7</i>

VISÃO DO PRODUTO E PROJETO

1 VISÃO GERAL DO PRODUTO

1.1 Declaração do Problema

O problema	É a falta de uma solução digital e consolidada para o gerenciamento das questões humanas da academia. A partir disso, foi identificada a oportunidade de desenvolvimento de um software para o nicho de academias.
Afeta	O administrador, os alunos e os profissionais (professores/Personal Trainers).
Cujo impacto é	Para o administrador: a falta de concentração dos dados financeiros e de controle de frequência de seus alunos matriculados em uma única plataforma; a dificuldade de comunicação imediata e automática de avisos para com os alunos. Para os alunos: terem suas avaliações e treinos registrados em fichas físicas, que podem ser facilmente perdidas ou danificadas; a necessidade de se dirigir a uma unidade física da academia para resolver problemas financeiros ou solucionar dúvidas ou consultas simples. Para os profissionais: não possuírem sua rotina e ferramentas de gerenciamento e auxílio de alunos em um único lugar.
Uma solução de sucesso seria	Um software em que os alunos pudessem administrar seus treinos e avaliações, resolver questões que envolvam seu estado financeiro e entrar em contato com a academia. Para o administrador, uma solução de sucesso seria aquela em que os principais dados dos alunos e professores estivessem concentrados e que o possibilitaria com poucos cliques enviar avisos para os alunos por e-mail. Já para os professores, ter sua rotina em uma única tela juntamente com a possibilidade de auxiliar os alunos de forma on-line.

1.2 Declaração de Posição do Produto

Para	Academias
Que	Buscam uma solução em que possam realizar o gerenciamento de questões humanas e entregarem uma experiência digital aos alunos e professores
O SoftFit	É uma aplicação Web responsiva
Que	Entrega soluções digitais para as 3 partes: administrador, aluno e professor.
Ao contrário	Das academias tradicionais em que: <ol style="list-style-type: none">1. os dados, treinos e avaliações do aluno são disponíveis de forma física;2. Os alunos precisam se locomover até uma unidade física para solucionar problemas financeiros e tirarem dúvidas;3. Os professores só conseguem criar e administrar treinos se estiverem com os alunos ao seu lado;4. Os administradores não tem os principais dados dos alunos concentrados e com a disponibilidade de entrarem em contato de forma imediata e automática acerca de avisos importantes;
Nosso produto	Possui as seguintes funções: <ol style="list-style-type: none">1. Para o administrador, há a disposição dos dados, tanto pessoais e financeiros, quanto de frequência de seus alunos matriculados.

	<p>Além disso, o administrador também consegue enviar avisos importantes e entrar em contato por e-mail com os frequentadores da academia.</p> <p>2. Para os alunos, haverá seus treinos e avaliações (medidas) em uma única página (com a possibilidade de vídeos demonstrativos), também é possível entrar em contato com a academia por e-mail e chats para poderem tirar suas dúvidas e controlarem seu estado financeiro.</p> <p>3. Para os professores terão sua rotina disposta e ferramentas de criação e auxílio de treinos a partir dos dados dos alunos.</p>
--	---

1.3 Objetivos do Produto

O principal objetivo do produto é proporcionar uma experiência digital intuitiva e consolidada para os alunos, professores e administrador da academia.

2 ABORDAGEM DE DESENVOLVIMENTO DE SOFTWARE

2.1 Metodologia

Para o estabelecimento da metodologia de desenvolvimento de software, algumas questões foram previamente respondidas:

- **Questões Técnicas:**
 - **Qual é o tamanho do sistema que está sendo desenvolvido?** Relativamente extenso, uma vez que envolve funções para o aluno, o professor e o administrador da academia.
 - **Que tipo de sistema está sendo desenvolvido?** Aplicação Web responsiva.
 - **Qual a vida útil prevista para o sistema?** 4 meses (tempo da disciplina).
 - **O sistema está sujeito a controle externo?** Sim, o dono/administrador da academia adquiriria o software, logo o produto não estaria somente sob influência dos desenvolvedores.
- **Questões Humanas:**
 - **Qual é o nível de competência dos projetistas e programadores do time de desenvolvimento?** A princípio, todos que compõem o time possuem um nível de competência igualmente básico-intermediário.
 - **Como está organizado o time de desenvolvimento?** Não há uma divisão concreta além de Product Owner e desenvolvedores.
 - **Quais são as tecnologias disponíveis para apoiar o desenvolvimento do sistema?** Trello, Discord, Microsoft Teams e GitHub.
- **Questões Organizacionais:**
 - **É importante ter uma especificação e um projeto (design) bem detalhados antes de passar para a implementação - talvez por motivos contratuais?** De certa forma, não é necessário uma detalhação de especificações e design, apesar de haver um prévio conhecimento do que será construído e algumas funções do projeto.
 - **É realista uma estratégia de entrega incremental, na qual o software é entregue aos clientes ou outros stakeholders e um rápido feedback é obtido?** Considerando que teremos um Product Owner, isto é, alguém que trabalharia analisando as entregas do grupo e ver se está de acordo com o pensado, é sim realista.
 - **Os representantes do cliente estarão disponíveis e dispostos a participar do time de desenvolvimento?** Basicamente, a equipe terá um “representante de cliente”, então sim.
 - **Existem questões culturais que possam afetar o desenvolvimento do sistema?** Não

A partir das respostas desenvolvidas, considera-se que por não haver uma necessidade da criação de documentos formais sendo criados dentro das atividades e, uma vez que os requisitos e o projeto serão desenvolvidos conjuntamente, uma **abordagem ágil** seja ideal.

Sendo assim, a metodologia ágil **Kanban** foi escolhida a partir dos seguintes motivos:

- Seguir uma filosofia contínua de desenvolvimento e entrega e, portanto, admitir uma maior flexibilidade e leveza para os desenvolvedores;
- Ter uma organização visível clara que possibilita ao grupo, sem a necessidade de reunião constantes, entender os processos e em que etapa o desenvolvimento está como um todo;
- Por não ter a necessidade de uma divisão de funções específicas, o grupo pode, como um todo, se auto-organizar e trabalhar com maior autonomia;
- A priorização de atividades é bem intuitiva com o quadro e com a limitação de WIP (trabalhos em andamento).

Dessa forma, apesar dos horários rígidos que os integrantes do grupo, como alunos, possam ter, com o Kanban, o grupo pode trabalhar de forma ágil, mas sem a necessidade de comprometer diários de horários.

2.2 Processo

<i>Atividade</i>	<i>Objetivo</i>	<i>Papel</i>
<i>Criar quadro Kanban</i>	<i>Organizar o quadro que a equipe utilizará para se orientar durante o desenvolvimento</i>	<i>Product Owner e Equipe de Desenvolvimento</i>
<i>Estabelecer escopo do produto</i>	<i>Definir as características que devem estar presentes no produto de software</i>	<i>Product Owner e Equipe de Desenvolvimento</i>
<i>Estudar tecnologias (Python e Django)</i>	<i>Adquirir o conhecimento necessário para o desenvolvimento do sistema</i>	<i>Equipe de Desenvolvimento</i>
...

2.3 Procedimentos

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>
<i>Criar quadro Kanban</i>	<i>Requisitos – Em desenvolvimento - Entregue</i>	<i>Trello</i>
<i>Estabelecer escopo do produto</i>	<i>EAP – Estrutura Analítica de Projeto(não definitivo, ainda)</i>	<i>Trello</i>
<i>Estudar tecnologias (Python e Django)</i>	<i>Individual</i>	<i>Materiais Digitais (cursos, vídeos e artigos)</i>
...

3 LIÇÕES APRENDIDAS

3.1 Unidade 1

[Liste as lições aprendidas na retrospectiva, com ênfase especial nas ações a serem tomadas para melhorar, por exemplo: o ambiente de desenvolvimento, o processo ou a colaboração da equipe.]

3.2 Unidade 2

[Liste as lições aprendidas na retrospectiva, com ênfase especial nas ações a serem tomadas para melhorar, por exemplo: o ambiente de desenvolvimento, o processo ou a colaboração da equipe.]

3.3 Unidade 3

[Liste as lições aprendidas na retrospectiva, com ênfase especial nas ações a serem tomadas para melhorar, por exemplo: o ambiente de desenvolvimento, o processo ou a colaboração da equipe.]

3.4 Unidade 4

[Liste as lições aprendidas na retrospectiva, com ênfase especial nas ações a serem tomadas para melhorar, por exemplo: o ambiente de desenvolvimento, o processo ou a colaboração da equipe.]

3.5 Unidade 5

[Liste as lições aprendidas na retrospectiva, com ênfase especial nas ações a serem tomadas para melhorar, por exemplo: o ambiente de desenvolvimento, o processo ou a colaboração da equipe.]

4 REFERÊNCIAS BIBLIOGRÁFICAS

1. Sommerville, Ian Engenharia de software/ Ian Sommerville; tradução Luiz Cláudio Queiroz; revisão técnica Fábio Levy Siqueira. -- 10. ed. -- São Paulo: Pearson Education do Brasil, 2018. Título original: Software engineering ISBN 978-65-5011-048-2 1. Engenharia de software I. Siqueira.