



Universidade de Brasília

FCTE

Requisitos de Software
(FGA0313)

DeltaCore

Visão de Produto e Projeto

Grupo 2

Cibelly Lourenço Ferreira - 231026680

Eduardo Silva Waski - 231011284

Felipe Bressanelli Azevedo Filho - 222024579

Isabelle da Costa Figueiredo - 211039500

Luis Eduardo Castro Mendes de Lima - 221008285

Henrique Carvalho Neves - 222006801

Professor: George Marsciano

7 de Abril de 2025

1 CENÁRIO ATUAL DO CLIENTE E DO NEGÓCIO

1.1 Introdução ao Negócio e Contexto

Um dos segmentos que mais crescem é o meio fitness. Com o aumento do uso das redes sociais, esse segmento tem ganhado cada vez mais força, em especial o setor voltado ao bem-estar. Tendo isso em mente, nosso cliente é dono de duas franquias de studios fitness: Vidya Yoga e Race Bootcamp, ambas pertencentes ao grupo Smart Fit.

Vidya Yoga é um studio fitness que combina as práticas de yoga com uma sala aquecida a 40°C e iluminação que varia de acordo com os momentos e posições, utilizando paletas de cores quentes. Essa prática é conhecida como hot yoga. Sua principal missão é trazer tranquilidade à mente e ao corpo, alinhando ambos. Possui unidades em todas as regiões do Brasil, com foco em grandes capitais e áreas de alto poder aquisitivo. Seu público-alvo principal são mulheres jovens, de 20 a 40 anos, que se preocupam com sua saúde física e mental.

Race Bootcamp é um studio com equipamentos de academia, focado em esteiras e pesos livres. O controle da temperatura da sala e a iluminação são usados para incentivar os participantes a atingirem seu limite físico. É um ambiente controlado, onde os interessados podem usufruir dos equipamentos e orientação de professores, com o objetivo de descobrir e superar seus limites. Tem como principal propósito fornecer um ambiente ideal para que atletas amadores possam redefinir seus conceitos de limite físico. Assim como o Vidya Yoga, o Race Bootcamp está presente em todas as regiões do Brasil, com foco em grandes capitais e regiões de alto poder aquisitivo. Seu público-alvo são homens e mulheres jovens, de 25 a 40 anos, praticantes de corrida e atletas amadores interessados em conhecer seus próprios limites.

Ambas as franquias funcionam no sistema pay per use, sendo necessário comprar créditos pelos sites de cada studio fitness. Cada crédito equivale a uma determinada quantidade de aulas. Como ambos os studios possuem conceitos diferenciados dos studios fitness tradicionais, oferecem uma quantidade significativa de aulas experimentais, que podem ser agendadas pelo site, seguindo o mesmo processo da compra de créditos.

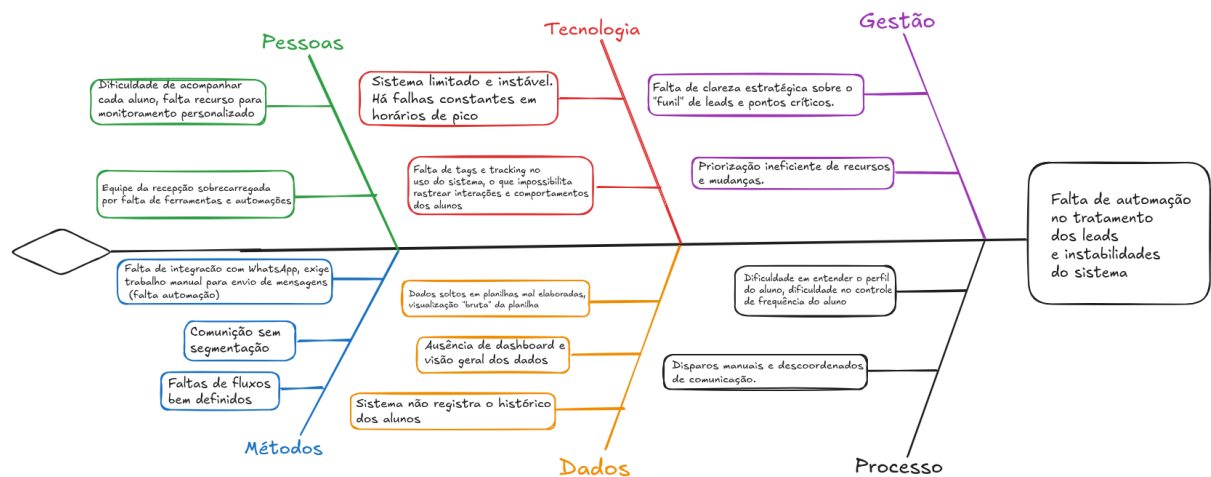
1.2 Identificação da Oportunidade ou Problema

O grupo Smart adquiriu recentemente uma startup chamada Angular, que desenvolve um sistema de mesmo nome para a gestão de franquias de academias no modelo pague pelo uso. Com o sistema, é possível visualizar quantos créditos cada aluno possui, quantas visitas ele realizou ao studio, além da agenda dos studios, com horários e profissionais responsáveis por cada aula.

Nosso cliente, franqueado de academias, utiliza esse sistema e relata inconsistências recorrentes, como erros na exibição das informações, além de abas e páginas que deixam de aparecer aleatoriamente, e, principalmente, instabilidade nos momentos de maior demanda — ou seja, durante os horários de pico da academia. Nessas ocasiões, o sistema fica fora do ar, obrigando os recepcionistas a anotarem informações de forma assíncrona e descentralizada, geralmente com papel e caneta, sem acesso aos dados necessários no site e possivelmente perdendo informação importante.

Outro ponto crítico é a total ausência de ferramentas de análise dos dados já concentrados no sistema. Não há nenhuma funcionalidade que permita o cruzamento de informações para identificar, por exemplo, os clientes mais inativos, aqueles que participaram de uma aula experimental e não retornaram, ou os que estão prestes a ficar sem créditos — entre outras categorias relevantes.

Com base nessas informações, é possível alinhar uma demanda já expressa pelo dono das franquias: o envio automatizado de mensagens aos clientes, com o objetivo de melhorar o atendimento, fidelizar e recuperar alunos. Isso eliminaria essa etapa essencial da rotina da recepção, atualmente feita de forma manual e, segundo o cliente, com baixa efetividade, já que os responsáveis frequentemente deixam de enviar as mensagens, e ele não consegue acompanhar essa tarefa de forma adequada.



1.3 Desafios do Projeto

Os principais desafios enfrentados pelo cliente envolvem a operacionalização manual de processos, que exige tempo da equipe para revisar dados de presença e comportamento dos alunos, além da falta de integração de dados, dificultando a análise e tomada de decisões. A baixa automação na comunicação impacta negativamente a experiência dos alunos e a escalabilidade do modelo atual compromete a padronização com o crescimento da franquia. Além disso temos o desafio técnico de fornecer constância para o sistema nos horários de pico.

1.4 Segmentação de Clientes

O sistema seria usado basicamente por apenas um grupo:

Usuários das franquias: Esse grupo não utilizará diretamente o sistema, mas sim as franquias, são homens e mulheres de alto poder aquisitivo entre 25 e 40 anos que possuem grande interesse em saúde e bem estar.

Recepcionistas (18 a 40 anos): grupo formado por pessoas de diferentes idades, com no mínimo um conhecimento básico em tecnologia. São capazes de operar sistemas de forma simples e funcional, mas não possuem domínio avançado sobre ferramentas mais complexas. . **Gerente/Dono da franquia:** Grupo formado pelos possíveis gerentes que podem vir a serem contratados caso o negocio escale e o Dono da franquia, onde são capazes de operar o sistema com desenvoltura e possuem ênfase em acompanhar e verificar o trabalho feito.

2 SOLUÇÃO PROPOSTA

2.1 Objetivos do Produto

O objetivo do produto é fornecer uma plataforma capaz de operar sem interrupções no funcionamento, mesmo durante os horários de pico. Além disso, agregar uma seção específica para o cruzamento e a análise de dados sobre os clientes, a academia e os funcionários, além de uma funcionalidade que envia mensagens personalizadas para cada tipo diferente de segmento de cliente, sem ser necessária a operação manual da equipe da recepção. A plataforma também manterá as funções essenciais que o sistema antigo já oferecia, como a visualização das marcações de aulas.

2.2 Características da Solução

A solução proposta contará com um conjunto de funcionalidades voltadas à automatização da gestão da franquia, à melhoria da comunicação com os alunos e à organização das operações internas. Dentre as principais características previstas, destacam-se:

- Automação da comunicação com alunos, incluindo mensagens de boas-vindas, lembretes de aula e incentivo para retorno em caso de faltas.
- Painel analítico com relatórios, estatísticas e indicadores de desempenho, contando com diversos filtros para personalização
- Interface simples e intuitiva
- Sistema livre de interrupções em momentos de alta demanda
- Painel de visualização dos agendamentos

2.3 Tecnologias a Serem Utilizadas

A solução será desenvolvida com tecnologias modernas, amplamente adotadas no mercado e de fácil integração, visando garantir escalabilidade, desempenho e uma boa experiência para os usuários. As principais tecnologias previstas são:

- **Frontend:** Next.js, para criação de interfaces amigáveis, intuitivas e responsivas.
- **Backend:** Node.js, com desenvolvimento de APIs REST para comunicação entre os sistemas.
- **Banco de dados:** PostgreSQL, para armazenamento dos dados.
- **Serviços de mensageria:** Integração com WhatsApp API, envio de e-mail e SMS para automação da comunicação com os alunos.
- **Hospedagem:** Heroku, mas passível de mudança ao longo do projeto, dependendo das necessidades de escalabilidade e facilidade de implementação.

2.4 Pesquisa de Mercado e Análise Competitiva

No Brasil, existem outras plataformas semelhantes, inclusive a já utilizada atualmente pelo cliente, a AngularE E-commerce. Também contamos com a IFitness e a Evo (W12).

Evo (W12): é um sistema relatado por usuários como extremamente complexo, exige treinamento da equipe e muito esforço e comprometimento para a aprendizagem. Além disso, não é focado no modelo de academias que funcionam com compra e marcação de aulas por meio de créditos. Também não oferece um cruzamento e análise de dados eficientes. Já foram relatados problemas como quedas do sistema em horários de pico.

iFitness: focado em academias convencionais, com ênfase em treinos de musculação. Não foi desenhado para academias com modelo de créditos por aula e agendamento online. A interface é considerada complicada de usar e exige treinamento da equipe.

2.5 Análise de Viabilidade

A solução é viável tecnicamente, pois utiliza tecnologias maduras e conhecidas pelo grupo, permitindo o desenvolvimento modular e escalável. O prazo estimado para entrega da versão inicial (MVP) é de 3 a 4 meses. Financeiramente, a proposta inicial será de código aberto para a franquia, com potencial de monetização futura via licenciamento. A economia de tempo e recursos, aliada ao aumento na conversão e retenção de alunos, torna o projeto atrativo. O mercado demonstra demanda crescente por soluções desse tipo, especialmente entre franquias em expansão.

2.6 Impacto da Solução

A implementação do sistema trará impactos significativos na gestão da franquia, com redução de custos operacionais, maior controle das informações e suporte ao crescimento do negócio. A automação permitirá um atendimento mais ágil e personalizado, melhorando a experiência dos alunos e a rotina da equipe. Além disso, a comunicação inteligente deve aumentar o engajamento, a taxa de retorno e fortalecer a imagem da marca por meio da modernização dos serviços, contribuindo para uma expansão estruturada da franquia.

3 ESTRATÉGIAS DE ENGENHARIA DE SOFTWARE

3.1 Estratégia Priorizada

Abordagem de Desenvolvimento de Software: O projeto adotará a abordagem ágil de desenvolvimento de software. Esta abordagem foi escolhida por promover entregas incrementais, priorizar a colaboração contínua com o cliente e permitir adaptações rápidas às mudanças de requisitos, características essenciais para o contexto da franquia de academias, que está em expansão e requer soluções flexíveis e evolutivas.

Ciclo de Vida: O ciclo de vida adotado será o ciclo de vida Ágil, com foco em entregas frequentes e iterativas de valor ao cliente, promovendo colaboração constante, resposta rápida a mudanças e melhoria contínua. Esse modelo favorece a validação antecipada de funcionalidades, como a automação da comunicação com os alunos e os painéis analíticos, garantindo alinhamento com as necessidades do usuário final ao longo do desenvolvimento.

Processo de Engenharia de Software: O processo de desenvolvimento escolhido é o **ScrumXP**, uma junção do framework Scrum com práticas do Extreme Programming (XP). O ScrumXP oferece um equilíbrio entre a organização dos papéis, eventos e artefatos do Scrum e as práticas técnicas do XP, como integração contínua, refatoração e desenvolvimento orientado a testes (TDD), o que garante qualidade de código desde o início e agilidade na adaptação das funcionalidades.

Os novos quadros

3.2 Quadro Comparativo

| Critério | Scrum/XP | FDD - Desenvolvimento Orientado a Funcionalidades |
|---------------------------|---|---|
| Visão Geral | Iterativo e incremental, com foco em agilidade e baixa exigência documental | Focado na implementação de funcionalidades específicas, dividindo o sistema em partes menores |
| Tratamento da Arquitetura | Arquitetura evolui conforme as necessidades; sem planejamento detalhado inicial | Uso de diagramas UML para modelar a estrutura, com apoio de diagramas de sequência |
| Organização do Processo | Ciclos curtos (sprints) de 2 a 4 semanas, com entregas contínuas | Iterações de 2 a 10 dias, com planejamento baseado em funcionalidades |
| Adaptação de Requisitos | Alta – requisitos mudam a cada ciclo conforme feedback do cliente | Moderada – mudanças são possíveis, mas há foco em funcionalidades específicas |
| Participação do Cliente | Alta – cliente acompanha, revisa e sugere melhorias a cada entrega | Menor – comunicação é feita via documentação detalhada entre equipe e cliente |
| Nível de Complexidade | Processo enxuto, com foco em colaboração e menor dependência de documentação | Mais estruturado, com uso intenso de modelagem e documentação técnica |
| Técnicas Utilizadas | Programação em dupla, TDD, integração contínua, refatoração, design simples | Criação e uso de diagramas UML, documentação detalhada e refinamento incremental |
| Garantia de Qualidade | Qualidade garantida por testes automatizados e TDD durante o desenvolvimento | Avaliação no final de cada iteração, com testes e inspeções dos incrementos |
| Suporte às Equipes | Equipes autônomas com comunicação direta e colaborativa | Suporta equipes maiores com estruturação formal do processo |

3.3 Justificativa

A escolha do **ScrumXP** como processo de engenharia de software se baseia na necessidade do projeto de adaptar-se constantemente às demandas do cliente e ao contexto de uma franquia em crescimento. A flexibilidade na priorização de funcionalidades e a entrega contínua de valor são aspectos fundamentais para validar rapidamente as soluções propostas, especialmente em um cenário onde a experiência do usuário e a eficiência operacional são decisivas para o sucesso do produto.

Além disso, as práticas técnicas oriundas do XP garantem uma base de código mais confiável e sustentável, enquanto o framework Scrum permite uma organização clara das tarefas da equipe, promovendo transparência, colaboração e melhoria contínua. Esses fatores tornam o ScrumXP a escolha mais alinhada com os objetivos e desafios do projeto.

4 Cronograma e Entregas

O projeto DeltaCore seguirá a metodologia ágil, com sprints quinzenais e entregas incrementais. A duração total será de aproximadamente três meses, dividida em 6 sprints.

4.1 Cronograma de Sprints

| Sprint | Período | Entrega Principal | Descrição das Atividades |
|----------|---------------|---|---|
| Sprint 1 | 08/04 – 21/04 | Levantamento de Requisitos e Protótipo de Telas | Alinhamento com o cliente, entrevistas e criação do backlog. Wireframes e protótipos de baixa fidelidade da interface do sistema. |
| Sprint 2 | 22/04 – 05/05 | MVP da Interface e Arquitetura do Backend | Implementação inicial do frontend com Next.js e definição da arquitetura de APIs com Node.js. Início da configuração do banco PostgreSQL. |
| Sprint 3 | 06/05 – 19/05 | Módulo de Agendamentos e Autenticação | Entrega do painel de agendamento de aulas com visualização e autenticação de usuários. Integração básica com o banco de dados. |
| Sprint 4 | 20/05 – 02/06 | Módulo de Comunicação Automatizada | Desenvolvimento do sistema de envio de mensagens (WhatsApp, e-mail). Configuração inicial da integração com a API de mensagens. |
| Sprint 5 | 03/06 – 16/06 | Módulo de Análise de Dados e Painel Analítico | Geração de relatórios, visualização de estatísticas sobre alunos e uso do sistema. Criação de filtros e gráficos básicos. |
| Sprint 6 | 17/06 – 30/06 | Refino, Testes Finais e Documentação | Refatoração, correções de bugs, testes com usuários, entrega da documentação técnica e manual de uso. Preparação para deploy. |

4.2 Reuniões e Revisões

- **Daily meetings:** Adaptaremos o conceito de reuniões diárias, para reuniões todas terças e quintas com duração de no máximo 15 minutos.
- **Sprint Planning:** ao início de cada sprint.
- **Sprint Review e Retrospective:** ao final de cada sprint com demonstração das entregas ao cliente e discussão de melhorias.

5 Interação entre Equipe e Cliente

5.1 Composição da Equipe

A equipe de desenvolvimento será composta pelos seguintes papéis e responsáveis:

| Papel | Descrição | Participante(s) |
|------------------------|---|---|
| Gerente de Projeto | Coordena o projeto, garante a comunicação entre cliente e equipe, controla prazos e entregas. | Luis Edardo Lima |
| Desenvolvedor Frontend | Responsável pela interface do usuário, design e implementação das funcionalidades no lado do cliente. | Isabelle de Costa e Cibelly Lourenço |
| Desenvolvedor Backend | Implementa a lógica de negócios, integração com banco de dados e APIs. | Eduardo Silva Waski e Henrique Carvalho |
| Analista de QA | Garante a qualidade do produto, executando testes de funcionalidade, performance e usabilidade. | Felipe Bressanelli |
| Analista de Requisitos | Define os requisitos funcionais e não funcionais do sistema e garante que eles sejam atendidos. | Toda a equipe |

5.2 Comunicação

Ferramentas de Comunicação:

- **WhatsApp:** Será utilizado para comunicação diária entre os membros da equipe. Permite envio rápido de mensagens, compartilhamento de arquivos e criação de canais temáticos. Também será usado para interações com o cliente.
- **Google Meet:** Utilizados para reuniões semanais de revisão e planejamento com o cliente, permitindo validação das entregas e discussão de atividades futuras.
- **Miro/ZenHub:** Usado para gerenciar o backlog, controlar tarefas e acompanhar o progresso de cada sprint. Permite transparência e participação do cliente nas prioridades.
- **Gather Town:** Ferramenta que simula um escritório, onde cada participante tem seu espaço de trabalho, facilita o trabalho remoto, pois emula o trabalho em ambientes presencial, além de ter gráficos parecidos com jogo.
- **Discord:** Usada pela equipe para Reuniões internas, guardar informações importantes e troca de mensagens

Métodos e Frequência de Reuniões:

- **Reuniões Diárias (Daily Scrum):** Adaptaremos o conceito de reuniões diárias para reuniões presenciais todas terças e quintas, e ao longo da semana mandaremos atualizações do andamento do trabalho via mensagens de forma assíncrona.
- **Reunião de Revisão de Sprint (quinzenal):** Apresentação das funcionalidades implementadas ao cliente para validação e coleta de feedback.
- **Reunião de Planejamento de Sprint:** Definição das atividades do próximo sprint com base no feedback da revisão.
- **Reunião de Retrospectiva:** Discussão interna ao final de cada sprint sobre melhorias de processo e lições aprendidas.

Frequência de Interações com o Cliente:

- **Revisões de Sprint (quinzenal):** Envolvimento direto do cliente para validação e feedback.
- **Interações Adicionais por Slack:** Acesso direto do cliente ao canal da equipe para esclarecimentos e solicitações rápidas.

5.3 Processo de Validação

O processo de validação da solução será dividido em três etapas principais:

1. **Definition of Ready (DoR):** Antes de iniciar o desenvolvimento, será verificado se os requisitos estão bem definidos, documentados e com critérios de aceitação claros.
2. **Definition of Done (DoD):** Uma funcionalidade só será considerada pronta após passar por testes unitários e de integração, com aprovação visual e funcional pela equipe e pelo cliente.
3. **Teste de Aceitação:** Após a validação interna, o cliente realizará testes de aceitação baseando-se nos critérios definidos na DoR para confirmar o atendimento aos requisitos.

6 Lições Aprendidas

6.1 Unidade 1

Durante o desenvolvimento inicial do projeto da DeltaCore, várias lições importantes foram aprendidas que irão influenciar as próximas fases do projeto. Esta seção apresenta essas lições, organizadas por desafios enfrentados, ações de melhoria e impactos esperados.

Lições Aprendidas e Melhorias

1. Personalização na Solução de Problemas

Desafio: A equipe inicialmente utilizava uma abordagem padronizada para resolução de problemas, o que se mostrou ineficiente em alguns casos devido às particularidades de cada cliente, como nível de conhecimento técnico e disponibilidade para reuniões.

Ação de Melhoria: Passou-se a adotar uma análise mais individualizada de cada problema, considerando o contexto específico do cliente. Isso incluiu a adaptação da linguagem técnica utilizada, o ritmo de trabalho e a forma de comunicação com base na disponibilidade do cliente para interações síncronas ou assíncronas.

Resultado Esperado: Maior eficiência na resolução de problemas, redução de mal-entendidos e aumento da satisfação do cliente com o processo de desenvolvimento e suporte.