

IFit: desenvolvimento de um aplicativo gratuito para praticantes de Crossfit acompanharem seu desempenho pessoal

¹Renan Campos Ribeiro, ²Alexandre Grotta (Orientador)

¹Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP)
Av. Thereza Ana Cecon Breda, 1896 - Vila Sao Pedro – Hortolândia – SP – Brazil

¹campos.renan@aluno.ifsp.edu.br, ²grotta@ifsp.edu.br

Abstract. *A sedentary lifestyle is a problem that causes obesity which in turn causes several health problems. In 2022 this problem increased due to the Covid-19. One of the ways to avoid a sedentary lifestyle is to practice sports, such as CrossFit. CrossFit involves a range of movements and exercises that have infinite combinations, allowing for different workouts every day. CrossFit gyms provide mobile apps for their monthly students to view their personal records and the day's training, however, because it is paid and the academies include this cost in the monthly fee, daily students do not have access to this app, which ends up being an excluding factor for these students, creating a barrier to entry into this sport. That's why IFit was developed to give daily students the basic features of this application for free. This allows more people to be able to practice this sport, which will make them more active and have a healthier life.*

Resumo. *O Sedentarismo é um problema que causa a obesidade que por sua vez causa diversos problemas de saúde. Em 2022 esse problema aumentou devido a pandemia de Covid-19. Uma das formas de evitar o sedentarismo é a prática de esportes, como o CrossFit. O CrossFit envolve uma gama de movimentos e exercícios que tem infinitas combinações, permitindo treinos diferentes todos os dias. As academias de CrossFit providenciam aplicativos móveis para seus alunos mensalistas visualizarem seus registros pessoais e o treino do dia, porém, por ser pago e as academias embutirem esse custo na mensalidade, os alunos diaristas não tem acesso a esse aplicativo, o que acaba sendo um fator excludente para esses alunos, criando uma barreira para a entrada nesse esporte. Por isso foi desenvolvido o IFit, para dar aos alunos diaristas as funcionalidades básicas desse aplicativo de forma gratuita. Isso permite que mais pessoas consigam praticar esse esporte, que fará com que elas se tornem mais ativas e tenham uma vida mais saudável.*

1. Introdução

Em 2022 o mundo se encontra em estado pós-pandemia de COVID -19, uma nova doença viral, causada pelo vírus SARS-COV-2 [Fiocruz, 2022]. No Brasil, até a data de publicação deste artigo, haviam sido constatadas 684 mil vítimas fatais decorrentes desta doença [Fioxrux, 2022]. Por outro lado, uma pessoa saudável tem menores chances de desenvolver casos mais graves desta doença. Uma boa imunidade contribui defesa do corpo contra

agentes infecciosos e vírus como o SARS-COV-2 que, apesar de não evitar esta doença, os impactos são menores. Pessoas com obesidade, por exemplo, são considerados grupo de risco em casos de COVID, mesmo um jovem de 20 anos obeso, tem os mesmos riscos de ter complicações como um idoso de 80 anos [SESA, 2020].

A atividade física é um dos principais pilares para uma boa imunidade e uma vida saudável. Dentre as atividades físicas existentes, destaca-se a modalidade CrossFit. Essa modalidade inspirada nos treinamentos de soldados americanos, tem sua origem na década de 2000, quando a marca CrossFit foi criada. Envolvendo exercícios aeróbios, movimentos ginásticos e levantamento de peso olímpico [CrossFit, 2021].

Os ginásios de CrossFit, ou *Box* como são chamados pelos praticantes, possuem professores certificados que podem prescrever treinos diários para os alunos. A maioria dos Boxes da região de Campinas - SP, faz a gestão e disponibiliza o acesso aos treinos por dois principais aplicativos, o Technofit e o CrossX [Cloud Gym, 2022]. Alunos que fazem aulas esporádicas usualmente ofertadas como benefício a trabalhadores, também conhecidas como aulas avulsas, não tem acesso ao aplicativo, que é pago por usuário, custo esse que está embutido na mensalidade dos alunos matriculados.

Esses alunos não têm acesso a materiais exclusivos e funcionalidades específicas da modalidade, tais como: levantamento de peso olímpico, movimentos ginásticos e treinos funcionais de resistência.

Diante desses desafios aos práticos esporádicos de CrossFit, foi desenvolvido neste trabalho um aplicativo móvel chamado IFit, que contribui para os mesmos, por ser um aplicativo gratuito, independente da caixa que o aluno frequenta, tendo acesso a como funcionalidades fundamentais para a evolução fazer atleta. O IFit é um aplicativo móvel com foco nessas pessoas e também pode ser utilizado pelas caixas para fornecer essas funcionalidades para seus alunos e ter o aplicativo com logomarca e identidade visual própria. Feito para as pessoas acessarem de qualquer aparelho celular, inclusive iOS e Android.

O aplicativo IFit foi desenvolvido com tecnologias React Native e Firebase, ambas as tecnologias líderes do segmento, utilizadas para o desenvolvimento rápido de aplicações móveis. A escolha das tecnologias foi baseada na facilidade de desenvolvimento multiplataforma para resolver um problema do mundo real da forma mais rápida e barata [Imasters 2022].

O objetivo principal, é o desenvolvimento de duas aplicações, sendo elas: IFit, o aplicativo móvel *whitelabel*, termo que será descrito na seção 4.5, com as funções visualizar treino do dia e registrar recordes pessoais. E o IFit BO, uma aplicação Web em que o professor submete uma planilha com o treino do dia para os alunos visualizarem no App. Para o desenvolvimento do aplicativo, tivemos os seguintes objetivos específicos: O desenvolvimento do backend, para gerenciar os dados dos usuários, como dados e cadastros. O desenvolvimento da interface com o aluno, em que o mesmo pode ver qual será o treino do dia e anotar seus recordes pessoais. E também a interface com o professor, em que o mesmo pode submeter uma planilha com o treino do dia.

Tendo esses objetivos, essa pesquisa assim segue:

- Seção referencial teórico, apresentamos as teorias e tecnologias utilizadas

- Seção metodologia, apresentamos a metodologia de desenvolvimento e metodologia de pesquisa.
- Seção desenvolvimento, apresentamos os passos das decisões técnicas e de desenvolvimento.
- Seção resultados, apresentamos as funcionalidades do aplicativo e da plataforma Web, assim como a aceitação pelos usuários.
- Seção de conclusão, apresentamos as conclusões finais da pesquisa e ideias de evolução desse trabalho.

2. Referencial teórico

Nesta seção são apresentados os referenciais teóricos utilizados para o desenvolvimento deste trabalho.

2.1. Aplicativos Multiplataforma

Aplicativos Multiplataforma são os aplicativos que estão acessíveis em múltiplas plataformas sem que tenham sido desenvolvidos especificamente para seus respectivos sistemas operacionais [React Native, 2022]. O aplicativo IFit está disponível em nas plataformas líderes de mercado Android e iOS e foi desenvolvido utilizando React Native, que será abordado a seguir.

Segundo a consultoria Gartner, o volume de vendas no mundo todo, apesar de ter decaído 5% em 2021, a Apple representa 12,6% de todas as vendas de smartfone do mundo, enquanto a Samsung, Huawei e Xiaomi representam juntas mais de 40% de vendas pelo mundo [Gartner, 2022]. Evidenciando assim a preferência do mercado por aparelhos com os sistemas operacionais Android e iOS, portanto o aplicativo IFit foi desenvolvido para ambas.

2.2. React Native

React Native é uma biblioteca de desenvolvimento de aplicativos de forma nativa para iOS e Android, criada pelo Facebook e publicada em 26 de março de 2015 durante a ReactJS conf. Atualmente é mantida pelo Facebook e pela comunidade, sendo assim uma plataforma de *software* aberto com licença MIT [React Native, 2022].

2.2.1. JSX

JSX é uma extensão de sintaxe para Javascript, é utilizada no React para descrever como a interface do usuário deve parecer [React Native, 2022].

```
<Text key={key}>
  {description}
</Text>
```

Figura 1. Exemplo de código JSX

2.2.2. Arquitetura do React Native

O React Native realiza uma integração entre o JavaScript (ReactJS) e o Nativo (Objective-C e Java), essa integração é responsável por toda a comunicação entre as plataformas e é conhecida como Bridge [Litslink, 2022].

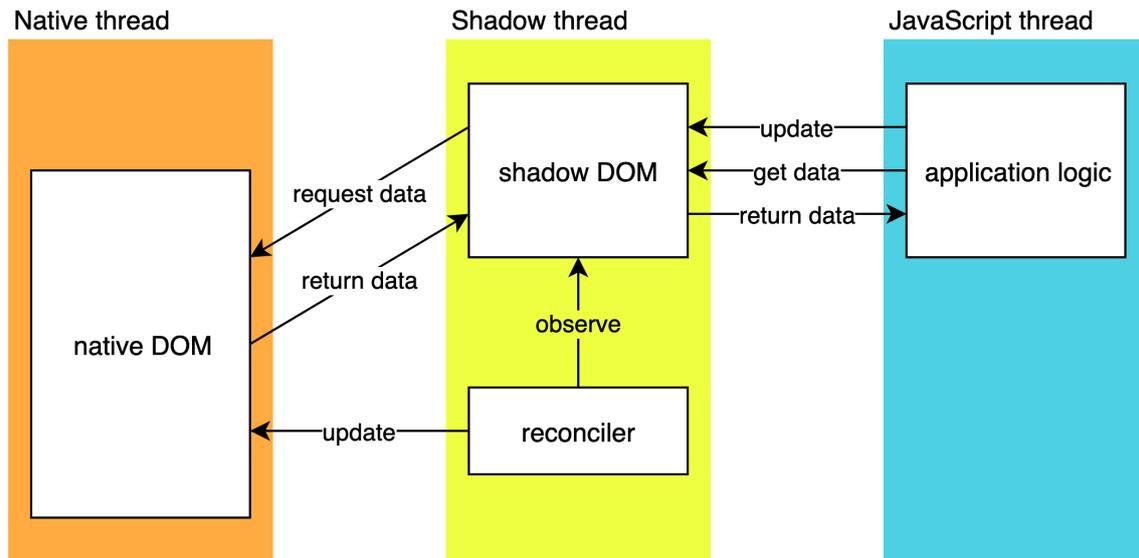


Figura 2. A arquitetura do React Native.

As aplicações são executadas nativamente no dispositivo do usuário, todo o código Javascript será executado na camada JSCore virtual machine, entretanto o código nativo terá suas respectivas variações em cada plataforma. Em dispositivos Android, o código Java será executado no android runtime e em dispositivos iOS o código Objective-c e Swift será executado no Native runtime.

Existem também três principais threads sendo executadas em uma aplicação React Native além das que já são executadas pelas aplicações nativas, sendo elas: Main thread, Exec Javascript e Shadow Queue.

A Main Thread é utilizada por qualquer aplicação nativa. Ela é responsável por tratar as requisições relacionadas a renderização de elementos na tela e também pelos gestos reproduzidos pelo usuário.

A Exec Javascript é exclusiva ao React Native, responsável por executar o código JavaScript.

O JavaScript é o responsável por toda lógica de negócio da aplicação. Além de definir a estrutura e as funcionalidades da nossa UI.

A *Shadow Queue* é a responsável pelos cálculos referentes ao *layout*.

A comunicação entre o Nativo e o Javascript se dá pela *bridge* da seguinte forma: A *Main thread* recebe uma informação de clique na tela que executa determinada função, essa informação é levada a *bridge* onde a *Thread* de execução do Javascript recebe essa informação e devolve uma resposta para a *Main thread* com o que deve ser renderizado na tela, sendo assim uma comunicação na sequência "N -> RN -> N". As Threads são

independentes e nunca se cruzam, todas as comunicações são feitas através da *Bridge*.

Componentes Nativos e JSX, o ReactJS implementam componentes JSX, que são componentes escritos em Javascript, semelhante ao XML, são traduzidos para componentes Nativos através de um processo de compilação transformando um código fonte em Javascript para um código fonte em código Java, Objective-C ou Swift [React Native, 2022].

Conclui-se assim que uma aplicação React Native é realmente nativa, uma vez que seus componentes são renderizados de forma nativa.

Por esses motivos, a facilidade em escrever o código uma única vez e ter o aplicativo disponível nas duas principais plataformas móveis de forma nativa, o React Native foi a escolha para a tecnologia do aplicativo móvel.

2.3. HTML, CSS e Javascript

HTML, CSS e Javascript são as principais linguagens da Web. [W3, 2022]. O HTML é a linguagem de marcação, o esqueleto de uma página Web. O CSS é responsável pelo estilo da página, suas cores e disposições como se fosse a "pele da sua página. O Javascript é responsável pela inteligência da sua página, suas ações, como se fosse o cérebro da sua página.

Por serem as principais linguagens da Web e serem o suficiente para o que é proposto como solução para a entrega do treino do dia do professor para disponibilizar no aplicativo, essas foram escolhidas para o desenvolvimento do IFit Web.

2.4. Firebase

Firebase é uma plataforma desenvolvida pelo Google para a criação de aplicativos móveis e da Web. Era originalmente uma empresa independente fundada em 2011. Em 2014, o Google adquiriu a plataforma e agora é sua oferta principal para o desenvolvimento de aplicativos [Firebase 2022].

Dentre as características do Firebase é ser multiplataforma, ou seja, suporta a integração com diversas plataformas, além de iOS e Android, possui integração com a Web e diversos *frameworks*, como o React Native. Também permite que todos os serviços utilizados para o desenvolvimento do IFit, que estão dentro do plano de utilização *Spark*, não tiveram custo algum. O custo vem com o crescimento, essa é uma das características mais importantes, por ser uma plataforma de *Back-End as a Service* [Sydle, 2022], ela permite que o foco seja no desenvolvimento do aplicativo e da página Web, os recursos passam a ser pagos pelo volume de requisições que são feitas a ele. Além disso, é simples, seguro e rápido, o desenvolvimento e todo o ciclo de vida do produto.

Por esses motivos, o Firebase foi a escolha para o back-end do aplicativo IFit e do IFit Web. Os serviços utilizados serão abordados na seção 4 Desenvolvimento.

2.5. Trabalhos Correlatos

Existem aplicativos disponíveis para os Boxes de CossFit, fazerem a gestão dos treinos alunos, e disponibilizar esse controle para os alunos matriculados, através de uma assinatura mensal paga pelo Box conforme mencionado na seção introdução e motivação.

Esta Seção mostra alguns aplicativos existentes que serviram como base de pesquisa para a proposta desse trabalho, suas principais funcionalidades e contrasta os recursos já existentes com os propostos para o aplicativo deste artigo.

2.5.1. Tecnofit

Esse aplicativo, criado pela empresa Tecnofit Tecnologia e Sistemas, tem o objetivo de fazer uma gestão completa das rotinas diárias de academias em geral [Tecnofit, 2021].

O aplicativo do aluno tem seu custo baseado na quantidade de alunos ativos, quanto mais alunos, menor o custo por aluno [Tecnofit, 2022]. Por ser de uma empresa que fornece essa plataforma de gestão para todo tipo de academia, não apenas Boxes de CrossFit, tem um aplicativo pesado com diversas funcionalidades que não tem utilidade para os praticantes de CrossFit. Na Playstore sua avaliação é de 4,9 estrelas, considerado excelente, avaliado por 6 mil pessoas até o momento da consulta em 2022.

2.5.2. CrossX

Esse aplicativo criado pela empresa homônima, tem o objetivo de fazer a gestão dos alunos de um Box de CrossFit [CrossX, 2021]. A empresa CrossX não divulga os preços de maneira pública, e sua nota na Playstore é de 3,3 estrelas, considerado baixo por ser um aplicativo de nicho e ter apenas mil avaliações até o momento da consulta em 2022.

2.5.3. Diferenciais do aplicativo IFit

O objetivo do aplicativo IFit, é fornecer funcionalidades necessárias aos alunos, como evolução de cargas, treino do dia e recordes pessoais, sem custo adicional ao Box ou ao aluno, independente do Box que estiver treinando. Por ser um aplicativo *whitelabel*, o Box de CrossFit que tiver interesse, poderá solicitar um aplicativo personalizado com nome, cor e identidade visual do Box.

3. Metodologia

Nesta seção, serão abordadas as metodologias utilizadas para a pesquisa e o desenvolvimento do IFit App e IFit Web.

3.1. Pesquisa de Design e Método Sombra

Pesquisa de *design* é um termo amplo para o processo que os *designers* usam para entender melhor os desejos, necessidades e desafios subjacentes e às vezes ocultos dos usuários finais, também conhecidos como o público-alvo [UxDdesign, 2022].

Ao contrário das formas mais tradicionais de pesquisa, a pesquisa de *design* não trata de confirmar uma hipótese ou encontrar uma solução. Trata-se de entender motivações e necessidades não atendidas. *Design Science Research* é usualmente empregado em prol da resolução de problemas da sociedade, melhorar o mundo exterior, através de novas perspectivas e percepções gratificantes, reduzindo o risco de projetar a solução errada e criando espaço para soluções inovadoras centradas no usuário.

Segundo Seonaidh McDonald [McDonald, 2005], *shadowing* é um método de pesquisa qualitativo onde o pesquisador acompanha o usuário enquanto ele resolve um problema à ser explorado ou enquanto ele utiliza um produto ou serviço. Em alguns casos, essa pesquisa tem o mesmo objetivo das entrevistas com usuários.

O *shadowing* acaba trazendo informações mais confiáveis sobre o comportamento dos usuários. Isso porque você está observando a pessoa fazer, ao invés de ouvir uma descrição dos acontecimentos.

Baseado nas características de pesquisa de *design*, consideramos o cenário em que as pessoas treinam dia a dia com os colegas de treino, que se queixavam por não saberem seus recordes pessoais, perguntarem qual seria o treino do dia, para se prepararem e levarem os equipamentos necessários.

Antes do desenvolvimento, uma pesquisa informal com os alunos do Box e foram encontrados alguns pontos relevantes durante essa observação. Os alunos desejam um histórico de recordes pessoais. Os professores necessitam de uma maneira fácil de submeter a planilha com o treino para os alunos. O Box gostaria de ter um aplicativo com a logomarca e as cores do Box.

Dois artefatos foram apresentados como solução, o IFit e o IFit Back Office, que estão descritos na introdução.

3.2. Metodologia de desenvolvimento

As metodologias de desenvolvimento de *software* consistem, no conjunto de abordagens utilizadas durante o planejamento e desenvolvimento dos artefatos propostos pela pesquisa de *design*.

3.2.1. Kanban

O termo *Kanban* é de origem japonesa, que significa sinalização ou cartão, e propõe o uso de cartões para indicar e acompanhar o andamento da produção dentro da indústria. Trata-se de um sistema visual que busca gerenciar o trabalho conforme ele se move pelo processo.

Um quadro *Kanban*, é uma ferramenta que ajuda a controlar o fluxo de trabalho e ter uma melhor visualização do andamento do projeto. Esse quadro consiste em 3 colunas onde as tarefas são movimentadas sempre para frente. As colunas são:

Backlog, é a coluna onde ficam todas as tarefas a serem realizadas ficam no início do projeto. Novas tarefas podem ser adicionadas conforme surgem durante o desenvolvimento, isso é natural, uma vez que algumas demandas surgem no decorrer do desenvolvimento.

In Progress, é a coluna onde ficam as tarefas que estão sendo realizadas. O ideal é que apenas uma tarefa esteja nessa coluna por vez.

Done, é a coluna onde ficam as tarefas que já foram desenvolvidas. Todas as tarefas estarão nessa coluna no fim do projeto.

O quadro *kanban* foi escolhido para auxiliar a visualização do andamento do de-

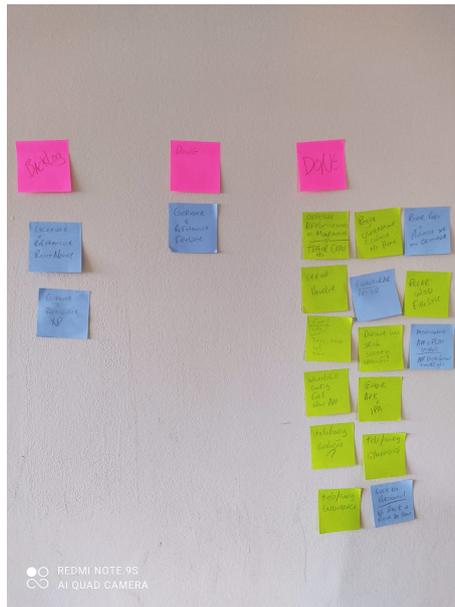


Figura 3. Esta figura representa o quadro kanban do projeto IFit.

envolvimento do projeto IFit app e IFit Web foi escolhido por sua simplicidade e utilidade para a proposta do projeto.

3.2.2. Programação Extrema individual

Esta pesquisa é uma atividade curricular obrigatória a ser desenvolvida individualmente com carga horária prevista de 80 horas. Portanto este projeto foi feito por apenas um desenvolvedor, no qual utilizei a Programação Extrema Individual gerenciada por *Kanban*, baseado na pesquisa "*Extreme Programming for a Single Person Team*" da universidade de Auburn feita pelo professor PHD David Umphress.

A Programação extrema individual (PEI), é uma adaptação da metodologia Extreme Programming (XP) para um único desenvolvedor. A metodologia XP, foca em situações envolvendo pequenos times, onde o desenvolvimento é feito em pares, onde um desenvolvedor deve revisar o código do outro desenvolvedor. Umphress em seu artigo sugere um roteiro para o desenvolvimento do projeto as fases de planejamento, desenvolvimento e *Post Mortem*.

Durante a fase de planejamento, abordamos a definição do problema, definição das tecnologias, definição da arquitetura dos sistemas, desenho das telas das aplicações, a criação das tarefas no *board Kanban* e o alinhamento de expectativas com o cliente, sendo nesse projeto o professor orientador, do qual tive que apresentar pequenas entregas, coletar feedbacks e fazer melhorias.

Com os resultados da pesquisa de *design*, a problemática ficou muito clara, assim como o quê precisaria ser desenvolvida e quais tecnologias utilizar. A próxima parte do planejamento passou a ser a criação das tarefas no quadro *Kanban*. As tarefas foram divididas em três principais frentes, arquitetura da aplicação, *design* da aplicação, desenvolvimento de código e escrita da documentação, que nesse projeto é este mesmo

artigo.

Tendo a arquitetura e tecnologias definidas, pequenas tarefas bem escritas e priorizadas, *design* das funcionalidades definidas, coleta de feedbacks com melhorias semanais em reuniões com o cliente, o desenvolvimento fica extremamente fluído, com pequenos ajustes e novas histórias surgindo. Se a funcionalidade não está como o esperado, uma nova tarefa é escrita e o *Kanban* é repriorizado, do contrário a tarefa é movida para done, e a próxima tarefa da coluna *backlog* deve ser desenvolvida e testada.

Quando todas as tarefas de desenvolvimento estão prontas, os artefatos devem ser movidos para produção, no caso deste projeto, o aplicativo está funcionando e disponível para o *download* dos usuários. O último passo do roteiro de desenvolvimento se dá com a aceitação completa dos usuários. O *Post Mortem* é o passo onde todas as funcionalidades propostas estão entregues e testadas.

4. Desenvolvimento

Nesta seção estão os detalhes do desenvolvimento do IFit app e IFit Web, assim como suas arquiteturas e definições técnicas de estrutura.

A arquitetura do projeto IFit se divide em 3 principais componentes, o aplicativo para o aluno, a aplicação Web para o professor disponibilizar o treino do dia, e o Firebase, que faz todo o controle dos dados e acessos dos usuários.

IFit Web é uma página em HTML, CSS e Javascript, com as configurações do Firebase em que o professor executa a página *index.html* e pode fazer o *upload* de uma planilha com a extensão *.CSV*. Essa planilha é enviada para o Firebase no serviço de banco de dados que será descrito na seção 4.2.1, esse registro é o mesmo que o IFit App vai buscar para mostrar o treino do dia para o aluno.

IFit App é um aplicativo React Native com o Firebase configurado em que o aluno pode visualizar o treino do dia e registrar seus recordes pessoais.

Firebase é o serviço em nuvem que faz a comunicação do IFit Web e IFit app e também é responsável por distribuir os instaladores do aplicativo APK para Android e IPA para iOS.

O Diagrama da arquitetura está disponível no ANEXO I.

4.1. Arquitetura cliente servidor

O modelo cliente-servidor, em computação, é uma estrutura de aplicação distribuída que distribui as tarefas e cargas de trabalho entre os fornecedores de um recurso ou serviço, designados como servidores, e os requerentes dos serviços, designados como clientes.

4.1.1. Cliente

Neste projeto temos dois clientes, são as duas aplicações, IFit BO e IFit App. IFit app consome informações dos clientes, treino do dia e autenticação dos usuários. IFit BO envia a planilha com o treino do dia disponibilizado pelo professor.

4.1.2. Servidor

Firebase provê os serviços de autenticação e banco de dados, através de seus serviços, *authentication* e *real time database*.

4.1.3. Authentication

Serviço responsável pela autenticação, criação, gerenciamento de conta dos usuários, serviço de envio de e-mails e SMS, conforme figura 4.

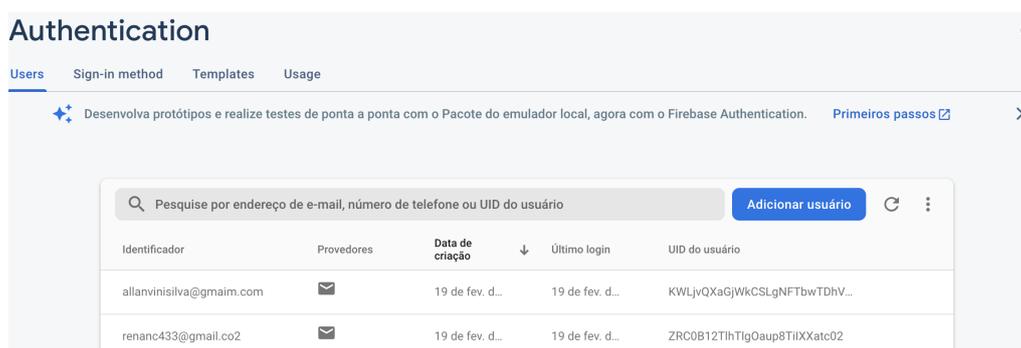


Figura 4. Painel de controle de usuários do Firebase authentication

4.1.4. Real time database

O banco de dados em tempo real é um sistema de banco de dados do Firebase que usa processamento em tempo real para lidar com cargas de trabalho cujo estado está em constante mudança, diferente dos bancos de dados tradicionais que contêm dados persistentes, principalmente não afetados pelo tempo.

4.1.5. App distribution

Serviço de gerenciamento de grupos de testadores e disponibilização do aplicativo fora das lojas oficiais.

4.2. Estrutura de arquivos do IFit app

A estrutura de arquivos de um projeto é fundamental para sua manutenção, quanto mais organizado e desacoplado, mais fácil é sua modificação, o código deve ser como argila, ser modificado com certa frequência para manter sua flexibilidade [código limpo, 2008]. Algumas das coisas descritas podem parecer óbvias se tiver a referência do nome das pastas e arquivos, mas a simplicidade e a obviedade é importante para que seja dedicado raciocínio às coisas mais complexas e de maior valor da aplicação.

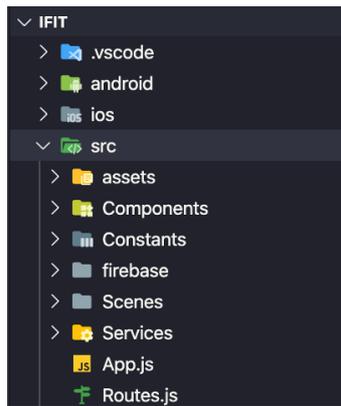


Figura 5. Esta figura representa a estrutura das pastas do IFit App.

Na figura 5 apresentamos algumas pastas e arquivos importantes deste projeto:

- **Android e iOS**
Android, código nativo em Java para os dispositivos Android, essa pasta é criada automaticamente pelo React Native no momento da criação do projeto, então não foi uma escolha. Aqui dentro é onde configuramos ícones, cores da Top Bar do celular no momento em que o app está aberto e *splash screen*. ios, o mesmo citado acima, mas com o código swift e objective-C.
- **SRC**
Código Javascript e as configurações do React native e onde a estrutura do código está definida, é onde você vai trabalhar de fato no desenvolvimento dos componentes, *assets*, rotas, dependências e logica do aplicativo.
- **assets**
Onde estão as imagens e ícones da aplicação.
- **Components**
Aqui estão os componentes do React Native, como botões, menus e listas.
- **Constants**
Todos os objetos com listas de nomes de exercícios que se repetem em vários lugares do projeto, se precisarem ser alterados, basta alterar no arquivo de constantes que serão refletidas em todo o projeto.
- **Firebase**
Arquivos de configuração do Firebase.
- **Scenes** Aqui estão as telas do aplicativo.
- **Services**
Aqui está a lógica do aplicativo, todas as chamadas em bancos de dados do Firebase e algumas funções utilitárias, como formatação de datas.
- **App.js e Routes.js**
Arquivos de inicialização e navegabilidade do aplicativo.

4.3. Design da aplicação

IFit foi desenhado para ser fácil, prático, intuitivo e ser uma aplicação *whitelabel*.

4.3.1. Aplicação Whitelabel

Uma aplicação *whitelabel*, é uma aplicação em que é possível gerar uma nova aplicação com uma identidade visual diferente, preservando suas funcionalidades, possibilitando assim o uso pro diversos Boxes que quiserem ter um aplicativo customizado com suas cores e logomarcas [FIA, 2022].

4.3.2. Figma

Figma é um editor *online* de *design* gráfico focado na criação de interfaces gráficas e experiência de usuário com ênfase colaborativa. A ferramenta permite que uma equipe de *designers* tenha acesso simultâneo e trabalhe em cima de um mesmo projeto ao mesmo tempo, facilitando a vida de times com profissionais em trabalho remoto em diversas localidades diferentes. [Figma, 2022]

Por uma das melhores ferramentas de prototipação do mercado e ter sua versão gratuita disponível, ela foi escolhida para a definição das telas e o diagrama de fluxo do usuário que pode ser visto com mais detalhes no ANEXO II

5. Resultados

Esta seção demonstra as funcionalidades do aplicativo IFit e do IFit Web, assim como sua aceitação pelos usuários.

5.1. Telas e funcionalidades

Principais funcionalidades e telas do aplicativo IFit estão disponíveis a seguir. As demais telas estão disponíveis no ANEXO III, devido a limitação de tamanho.

5.1.1. Criação de conta, Entrar e Tela principal

Após realizar a criação de conta e Entrar, o usuário acessará a tela principal, onde há um cumprimento com o nome do aluno e estão todos os menus do aplicativo. Há também dois menus que poderão ser utilizados numa futura evolução do aplicativo, são eles "Store" para o Box de CrossFit vender itens diretamente no aplicativo. E também o "Open 2023", que é um ranqueamento dos alunos na competição anual de CrossFit, o CrossFit Open.

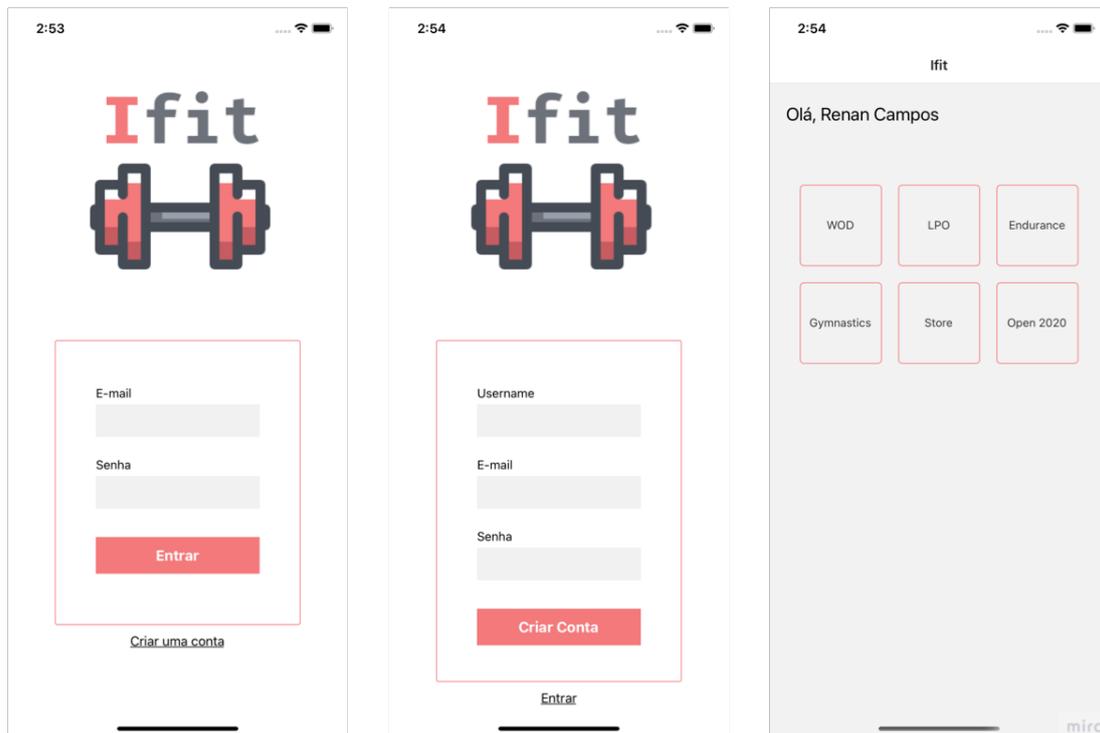


Figura 6. Figura com imagens da criação de conta, Entrar e Tela principal

5.1.2. Treino do dia, exercícios e Recordes Pessoais

A tela "treino do dia", exibe o treino que do dia enviado pelo treinador no IFit Web. Note que o treino se divide em 3 partes, aquecimento, técnica e treino do dia, essas informações, são muito valiosas para o aluno, pois ele consegue se planejar e levar os equipamentos necessários para o treino.

Na tela "exercícios" estão listados todos os movimentos, subdivididos em três categorias, *Endurance*, *Gymnastics* e *LPO* e suas variações utilizados no CrossFit. Ao selecionar o exercício o usuário pode registrar e modificar suas informações de recordes pessoais.

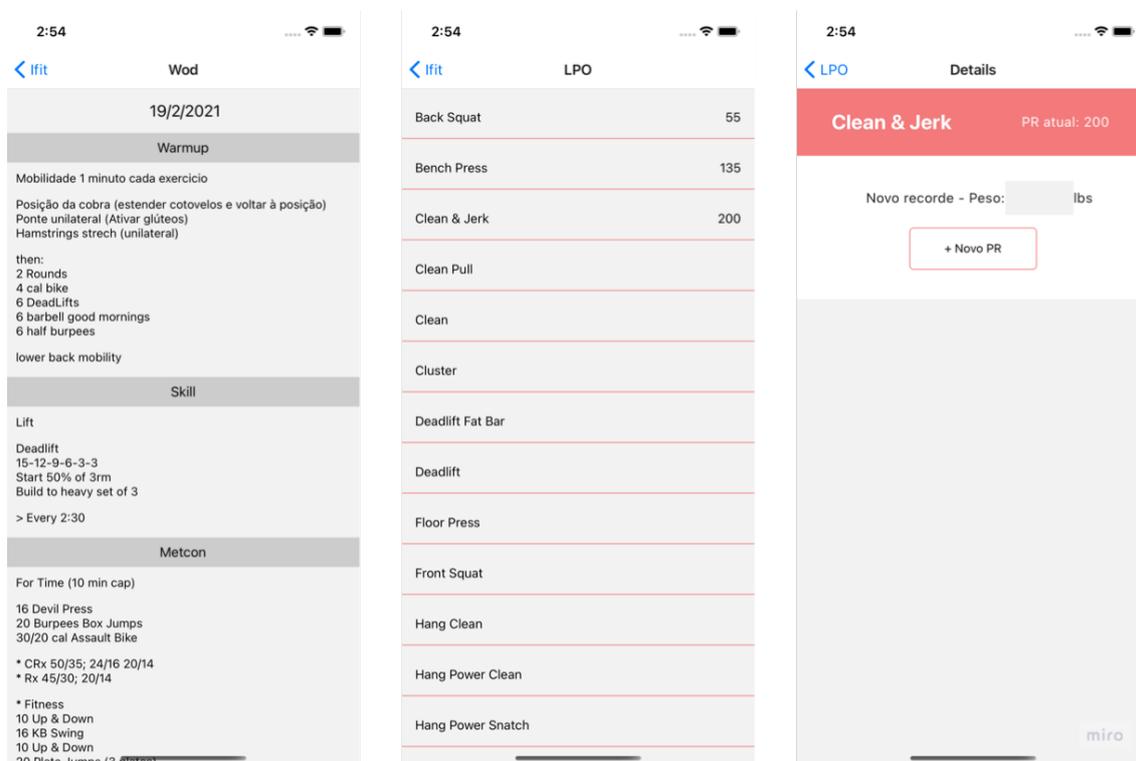


Figura 7. Figura com imagens do treino do dia, exercícios e recordes pessoais

5.1.3. IFit Web

Essa aplicação é muito simples e possui apenas uma tela, onde o treinador faz o *upload* da planilha, que pode ser conferida no ANEXO IV, com o treino do dia. Essa planilha se torna um registro no Firebase database e é consumido pelo aplicativo, na tela de treino do dia.

5.2. Disponibilização do APP

O aplicativo IFIT não será disponibilizado via lojas oficiais da Google e da Apple por envolver questões de custo. Será utilizado apenas via App Distribution [Firebase, 2022], conforme detalhado na subseção 4.2. Na data de publicação deste artigo, o custo para poder disponibilizar um aplicativo no Google Play Store era de US\$ 25,00 e na Apple Store era de US\$ 99,00.

5.2.1. Play Store

O custo para poder disponibilizar um aplicativo no Google play store é de 25 dólares por ano. 139,81 reais na cotação atual do dólar em real em 26 de outubro de 2021 + IOF de 4,08%. [Android developer, 2021]

5.2.2. App Store

O custo para poder disponibilizar um aplicativo no Apple store é de 99 dólares por ano. 553,66 reais na cotação atual do dólar em real em 26 de outubro de 2021 + IOF de 4,08%. [Apple developer, 2021]

5.3. Validação da aplicação

Apesar de não ter o aplicativo disponível nas lojas oficiais, os alunos aceitaram utilizar o aplicativo IFit, apresentaram boa receptividade, assim como o professor que disponibilizou os treinos, propondo ideias de futuras versões do App com mais funcionalidades.

Dos 7 alunos que utilizaram o aplicativo, 4 se sentiram muito satisfeitos com a experiência e as funcionalidades, 3 se sentiram satisfeitos e sugeriram novas funcionalidades.

O único professor que submeteu o treino todos os dias ficou satisfeito, gostou da experiência simples e sugeriu novas funcionalidades.

6. Conclusão

Nesta pesquisa, apresentou-se o desafio que praticantes de CrossFit encontram caso não possuam um aplicativo para acompanhar as aulas. Dentre os desafios, eles podem não conseguir acompanhar o treino do dia, seu progresso ou mesmo se preparar corretamente para as aulas por não saberem os equipamentos pessoais que serão necessários.

Na ausência de um aplicativo gratuito que auxiliasse estes praticantes, e inspirada pelo incentivo à prática de esportes para uma vida mais saudável, nesta pesquisa foi desenvolvido um aplicativo gratuito para auxiliar tais alunos, o IFit. Este aplicativo disponibiliza as funcionalidades essenciais de maneira intuitiva. O IFit foi desenvolvido para ser disponibilizado tanto nas plataformas Google e Apple. também foi desenvolvido um aplicativo web que possibilita aos professores de CrossFit atualizarem os treinos e disponibilizarem antecipadamente para os alunos. Em complemento, o IFit foi desenvolvido de maneira whitelabel, o que significa que esta solução pode ser adaptada para o contexto de outras academias. Assim, o objetivo principal desta pesquisa foi concluído.

Como funcionalidades futuras, o IFit poderia ter:

- i) funcionalidade de gestão de competições internas, como registro de resultados e rankings;
- ii) funcionalidade de reserva de aula em horários desejados;
- iii) funcionalidade de controle de frequência pelo professor.

Referências

Agarwal, Ravikant Umphress, David. (2008). Extreme programming for a single person team. 82-87. 10.1145/1593105.1593127.

APLICATIVOS E DESENVOLVIMENTO MOBILE HÍBRIDO X NATIVO. Imasters. Disponível em: <https://imasters.com.br/desenvolvimento/aplicativos-e-desenvolvimento-mobile-hibrido-x-nativo>, Acesso em: 10 de Janeiro de 2022.

BAAS: O QUE É BACKEND AS A SERVICE E POR QUE UTILIZAR?. Sydle. Disponível em: <https://www.sydle.com/br/blog/baas-o-que-e-backend-as-a-service-e-por-que-utilizar-6081de239b901904c48b323e/> Acesso em: 10 de Janeiro de 2022.

CORE COMPONENTS AND NATIVE COMPONENTS. React Native. Disponível em: <https://reactnative.dev/docs/intro-react-native-components>. Acesso em: 19 de Fevereiro de 2020.

EXTREME PROGRAMING ENABLING CHART. Disponível em: <http://wiki.c2.com/?ExtremeProgrammingEnablingChart>. Acesso em 19 de Fevereiro de 2020.

Firestore AUTHENTICATION. Firebase. Disponível em: <https://firebase.google.com/docs/auth>, Acesso em: 20 de Fevereiro de 2020.

Firestore APP DISTRIBUTION. Firebase. Disponível em: <https://firebase.google.com/docs/app-distribution>, Acesso em: 20 de Fevereiro de 2020.

Firestore CLOUD FIRESTORE. Firebase. Disponível em: <https://firebase.google.com/docs/firestore>, Acesso em: 20 de Fevereiro de 2020.

GARTNER SAYS WORLDWIDE SMARTPHONE SALES DECLINED FIVE PER CENT IN FOURTH QUARTER OF 2020. Gartner. Disponível em: <https://www.gartner.com/en/newsroom/press-releases/2021-02-22-4q20-smartphone-market-share-release>, Acesso em: 10 de Janeiro de 2022.

HTML, CSS AND JS, THE Web TRIAD. W3 SCHOOLS. Disponível em: <https://www.w3schools.com/> Acesso em: 10 de Janeiro de 2022.

McDonald, Seonaidh. (2005). Studying Actions in Context: A Qualitative Shadowing Method for Organizational Research. Qualitative Research. 5. 10.1177/1468794105056923.

NEW REACT NATIVE ARCHTECTURE. Litslink. Disponível em: <https://litslink.com/blog/new-react-native-architecture>, Acesso em: 10 de Janeiro de 2022.

O QUADRO KANBAN EXPLICADO EM 5 PASSOS SIMPLES. Kanbanize. Disponível em: <https://kanbanize.com/pt/recursos-kanban/primeiros-passos/o-que-e-quadro-kanban>. Acesso em: 21 de Fevereiro de 2020.

O QUE É WHITELABEL?. FIA. Disponível em: <https://fia.com.br/blog/white-label>. Acesso em: 10 de Janeiro de 2022.

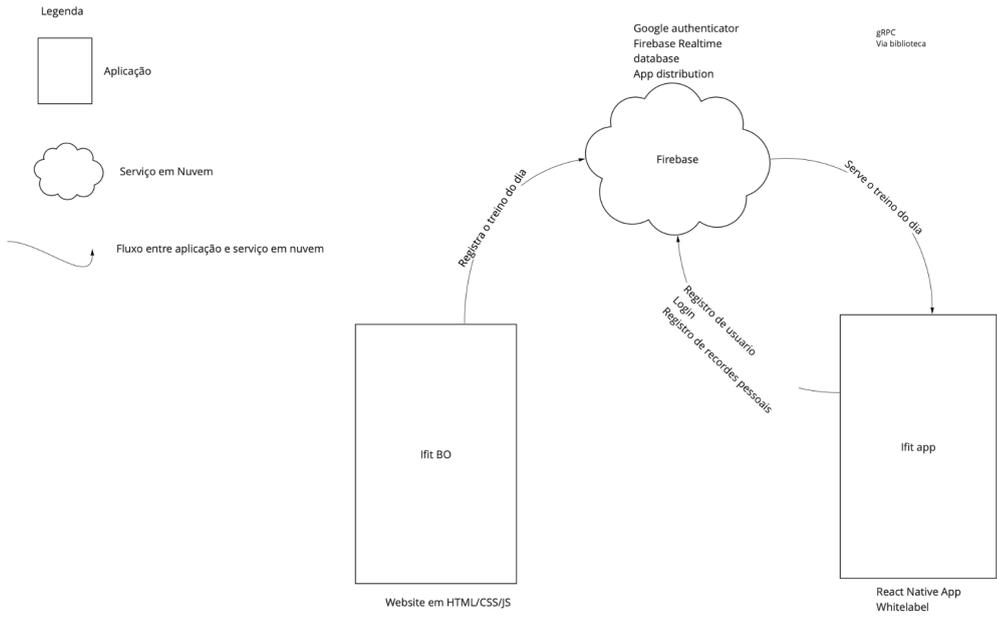
RASMUSSEN, Ana, OS NÚMEROS NÃO MENTEM: ANDROID OU iOS, QUAL É O MELHOR?. Tecnoblog. Disponível em:

<https://tecnoblog.net/247956/referencia-site-abnt-artigos/>. Acesso em: 19 de Fevereiro de 2020.

REACT NATIVE ARCHITECTURE. Disponível em <https://reactnative.dev/docs/xplat-implementation>. Acesso em 25 de Fevereiro de 2020.

SO, WHAT EXACTLY IS DESIGN RESEARCH?. Disponível em <https://uxdesign.cc/so-what-exactly-is-design-research>. Acesso em 29 de Dezembro de 2021.

ANEXO I



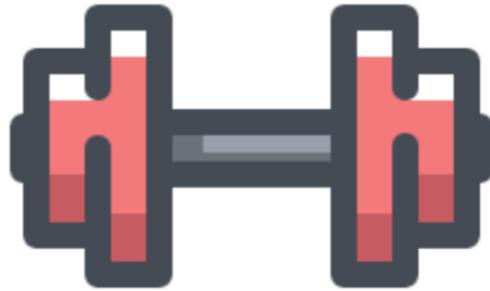
Anexo II

Anexo III

2:53



I fit



E-mail

Senha

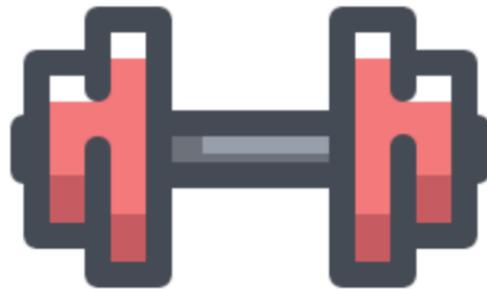
Entrar

[Criar uma conta](#)

2:54



I fit



Username

E-mail

Senha

Criar Conta

[Entrar](#)

2:54



Ifit

Olá, Renan Campos

WOD

LPO

Endurance

Gymnastics

Store

Open 2020

2:54



Wod

19/2/2021

Warmup

Mobilidade 1 minuto cada exercicio

Posição da cobra (estender cotovelos e voltar à posição)

Ponte unilateral (Ativar glúteos)

Hamstrings strech (unilateral)

then:

2 Rounds

4 cal bike

6 DeadLifts

6 barbell good mornings

6 half burpees

lower back mobility

Skill

Lift

Deadlift

15-12-9-6-3-3

Start 50% of 3rm

Build to heavy set of 3

> Every 2:30

Metcon

For Time (10 min cap)

16 Devil Press

20 Burpees Box Jumps

30/20 cal Assault Bike

* CRx 50/35; 24/16 20/14

* Rx 45/30; 20/14

* Fitness

10 Up & Down

16 KB Swing

10 Up & Down

20 Plate Jumps (3 plates)

2:54



< Ifit

LPO

Back Squat	55
Bench Press	135
Clean & Jerk	200
Clean Pull	
Clean	
Cluster	
Deadlift Fat Bar	
Deadlift	
Floor Press	
Front Squat	
Hang Clean	
Hang Power Clean	
Hang Power Snatch	

2:54



[← LPO](#)

Details

Clean & Jerk

PR atual: 200

Novo recorde - Peso: lbs

+ Novo PR

2:54



Endurance

Air Bike (100 cal)

3 min

Air Bike (50 cal)

Air Bike (Max cal 1)

Lateral Squats Hops

Row 100m

Row 10km

Row 1km

Row 2km

Row 500m

Row 5km

Run 1 mile

Run 10 km

Run 1200m

2:54

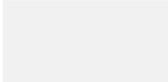


[Endurance](#)

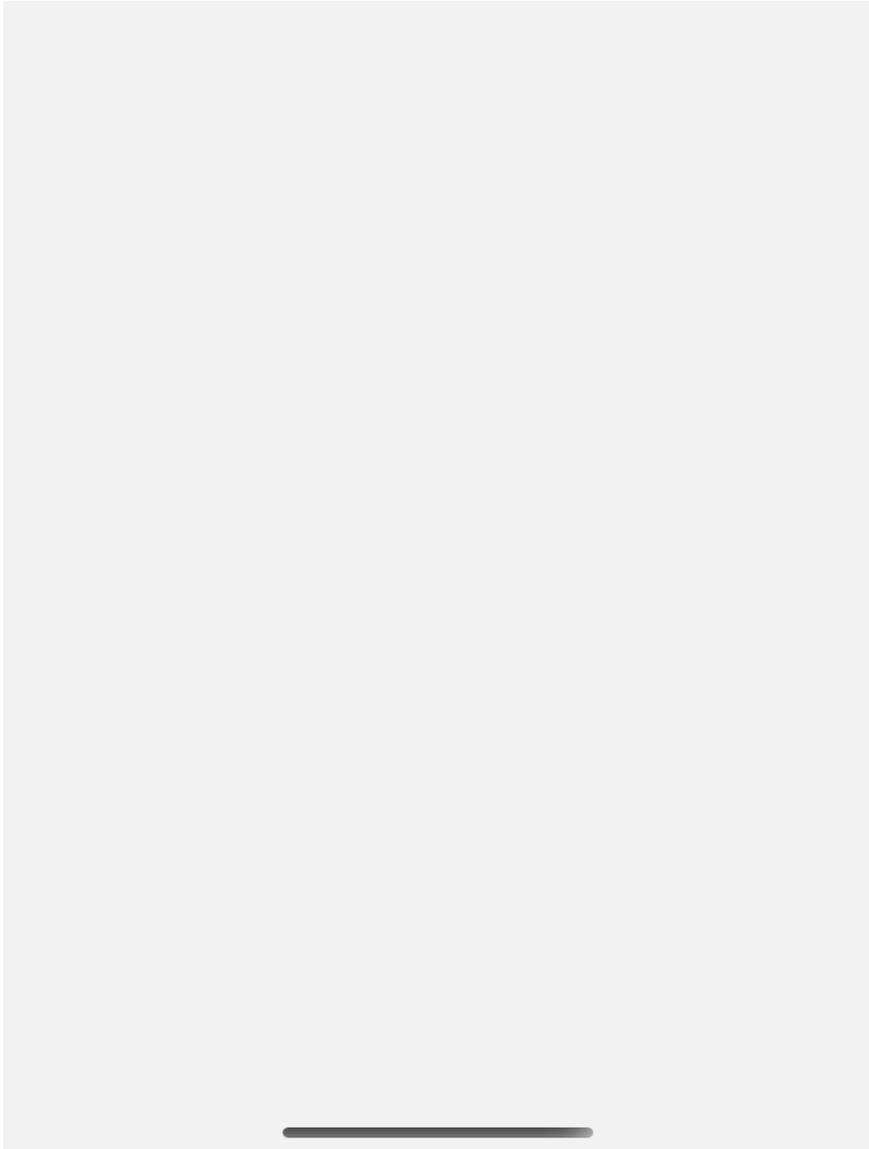
Details

Air Bike (50 cal)

PR atual: 0

Novo recorde - 

+ Novo PR



2:55



Gymnastics

Abmat Sit-up Max Reps	500
Box Jump Max Height	
DU Max Reps	
HSPU Max Reps	
Handstand Walk max distance	
L-sit Max Hold	
MU - 30 Reps for time	
MU Max Reps	
Pull Up Weighted 1 Rep Mx	
Pull Up chest to bar Max Reps	
Pull Up strict Max Reps	
Push Up Max Reps	
Ring Dip Max Reps	

2:55



[< Gymnastics](#)

Details

Box Jump Max Height

PR atual: 0

Novo recorde -

+ Novo PR

2:57



Gymnastics

Abmat Sit-up Max Reps

500

Box Jump Max Height

DU Max Reps

HSPU Max Reps

Handstand Walk max distance

L-sit Max Hold

MU - 30 Reps for time

MU Max Reps

Pull Up Weighted 1 Rep Mx

Pull Up chest to bar Max Reps

Pull Up strict Max Reps

Push Up Max Reps

Ring Dip Max Reps

Anexo IV



Olá Treinador, qual o treino de hoje?

Escolher arquivo Nenhum arquivo selecionado

Enviar

Documento Digitalizado Público

Versão final TCC de Renan Campos Ribeiro

Assunto: Versão final TCC de Renan Campos Ribeiro
Assinado por: Alexandre Grotta
Tipo do Documento: Relatório
Situação: Finalizado
Nível de Acesso: Público
Tipo do Conferência: Documento Digital

Documento assinado eletronicamente por:

- **Alexandre Grotta, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 11/09/2022 21:17:32.

Este documento foi armazenado no SUAP em 11/09/2022. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifsp.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1096225

Código de Autenticação: 7fa2336bf8

