

# ***Bootcamp Accenture Residência de Software 2021.2***

***UNIT - Universidade Tiradentes***

***Curso: Análise e Desenvolvimento de Sistemas***

***4º Período – Turno: Noite***

***Equipe:***

***Arthur de Oliveira Ávila***

***Auriclecio Marques Pereira***

***Edson Ramos da Silva Filho***

***Recife  
2021***

## Sumário

1. Descrição do MVP (Mínimo Produto Viável)	3
1.1 Problema	3
1.2 Solução	3
1.3 Regras de negócio	3
1.4 Melhorias futuras	4
2. Wireframe	4
3. Estrutura do banco de dados	6
3.1 Dados persistidos	6
3.2 Relação entre as entidades em diagrama	6
3.3 Tecnologias utilizadas	7
4. Noções da arquitetura da solução	7
4.1 Tecnologias utilizadas no desenvolvimento	9
5. Ferramentas utilizadas no desenvolvimento do produto	12
6. Estratégia e artefatos	14

## ***1. Descrição do MVP (Mínimo Produto Viável)***

O Mínimo Produto Viável – ou Minimum Viable Product (MVP) – é a versão simplificada de um produto final de uma startup. A partir dela, o empreendedor vai oferecer o mínimo de funcionalidades com o objetivo de testar a aceitação do produto no mercado.

Ao contrário do desenvolvimento tradicional de produtos, que geralmente envolve um longo e pensativo período de incubação e busca a perfeição do produto, o objetivo do MVP é começar o processo de aprendizagem, e não finalizá-lo. Ao contrário de um teste de protótipo ou conceito, um MVP foi concebido não apenas para responder questões sobre o design do produto e questões técnicas. Seu objetivo é testar hipóteses fundamentais do negócio.

Um MVP possui três características principais:

- Tem valor suficiente para que as pessoas comecem a utilizá-lo;
- Demonstra benefícios suficientes para reter usuários iniciais;
- Fornece um ciclo de feedback para orientar o desenvolvimento futuro.

Esta técnica de desenvolvimento assume que os usuários iniciais podem visualizar o produto final a partir do MVP, e que o produto deixa abertura para receber comentários e sugestões ajudando no desenvolvimento de versões futuras.

### ***1.1 Qual o problema?***

Com a problemática da pandemia instaurada pelo Coronavírus, as empresas, após início gradual das atividades presenciais perceberam que suas instalações físicas, como salas e escritórios, eventualmente podiam encontrar-se com quantidade de funcionários acima do permitido pelo protocolo de convivência e sanitário definido pela OMS, ou com quantidades bem abaixo. Ambas as situações são indesejadas pelas empresas gerando transtornos, como ociosidade dos espaços físicos, deslocamento desnecessário de funcionários para o trabalho, ou mesmo, paliativos com permuta de funcionários entre salas e setores.

### ***1.2 Qual a Solução?***

Desenvolver uma plataforma que gerencie os agendamentos das salas das empresas de forma que os problemas citados anteriormente possam ser evitados. A aplicação deve ser intuitiva e controlar todo o fluxo de alocações dos funcionários entre salas, departamentos e dias da semana, conforme regras de negócio definidas pela empresa utilizadora da ferramenta e pelos protocolos de convivência e sanitários definidos pela OMS.

### ***1.3 Regras de Negócio***

Foram definidas para este momento inicial de implementação as seguintes regras de negócio:

- Somente será permitido ao funcionário realizar no máximo 02 (dois) agendamentos por semana;
- Será limitada a quantidade de funcionários que poderão ocupar uma sala em um determinado dia de trabalho em até um funcionário por 3,14m<sup>2</sup> da área da sala, conforme orientação do protocolo de convivência e sanitário da OMS;

- Cada agendamento cadastrado terá validade até às 18:00h de cada dia útil, sendo excluído da aplicação após este horário;
- Somente será permitido ao funcionário agendar uma sala do departamento ao qual ele pertence.

## 1.4 Melhorias futuras

- Implementar acesso via login do funcionário;
- Implementar um ícone reservado para a foto de identificação do funcionários;
- Implementar registro de histórico e gráficos estatísticos da frequência dos funcionários.

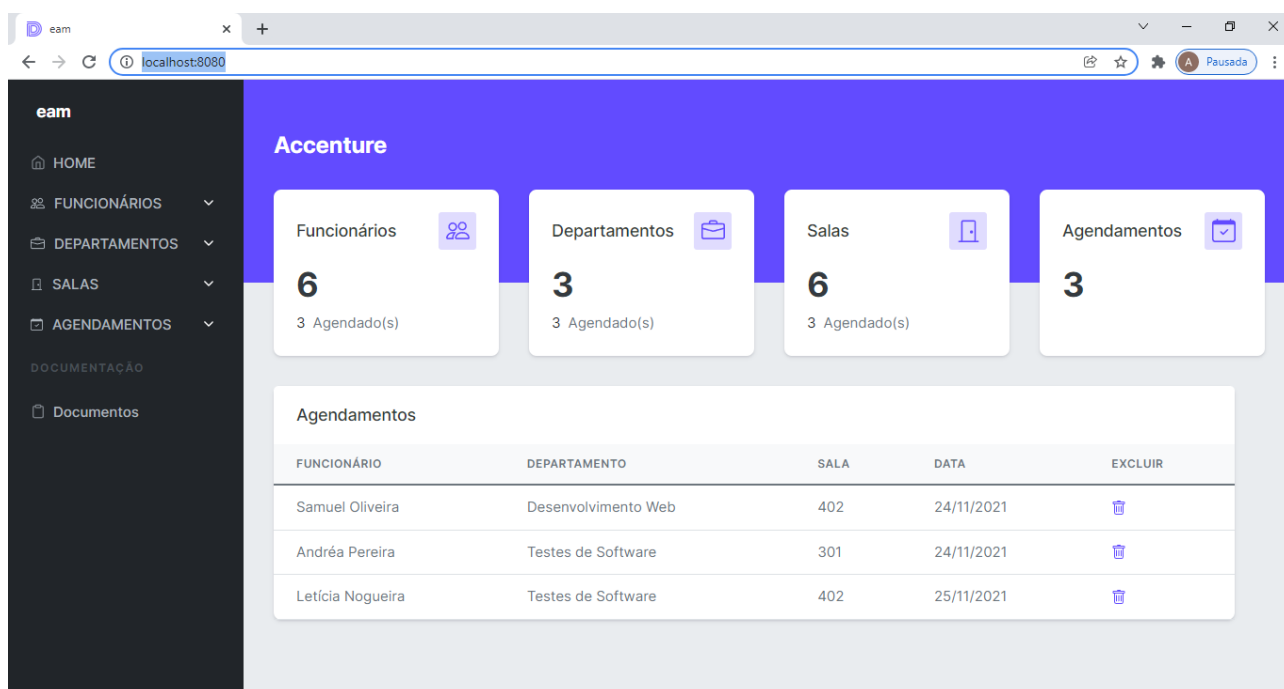
## 2. Wireframe

Wireframes são esboços simples de telas de produtos digitais, como sites e aplicativos. O intuito é estruturar e validar ideias, por isso os wireframes não contam com detalhes como cores, fontes, ícones e imagens.

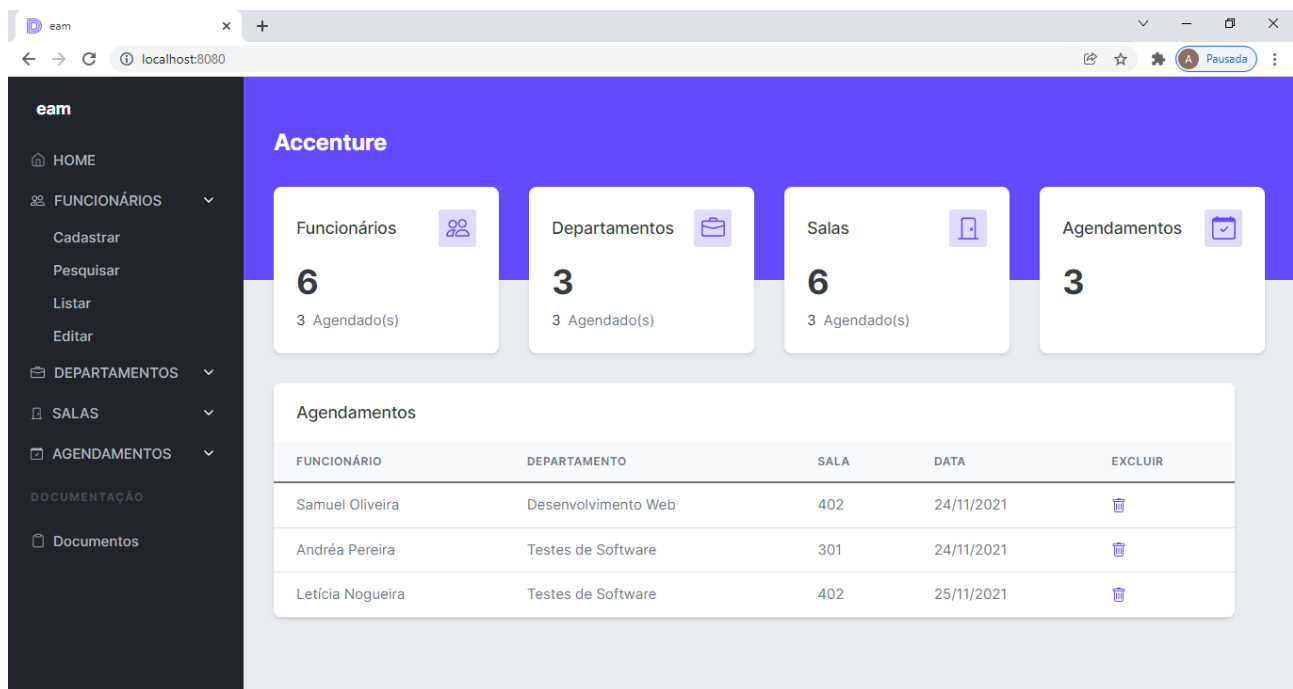
Desta forma, os dois objetivos principais do wireframe são o auxílio ao designer na hora da diagramação dos conteúdos e na aplicação da identidade visual, e também ser a principal ferramenta em relação a um alinhamento inicial da expectativa do cliente quanto ao visual do projeto contratado.

Para elaboração do wireframe foi escolhida a ferramenta Figma por ser web, ou seja, não precisa de instalação no computador, também por ser completa em seus recursos e por permitir criar uma quantidade de wireframes gratuitamente.

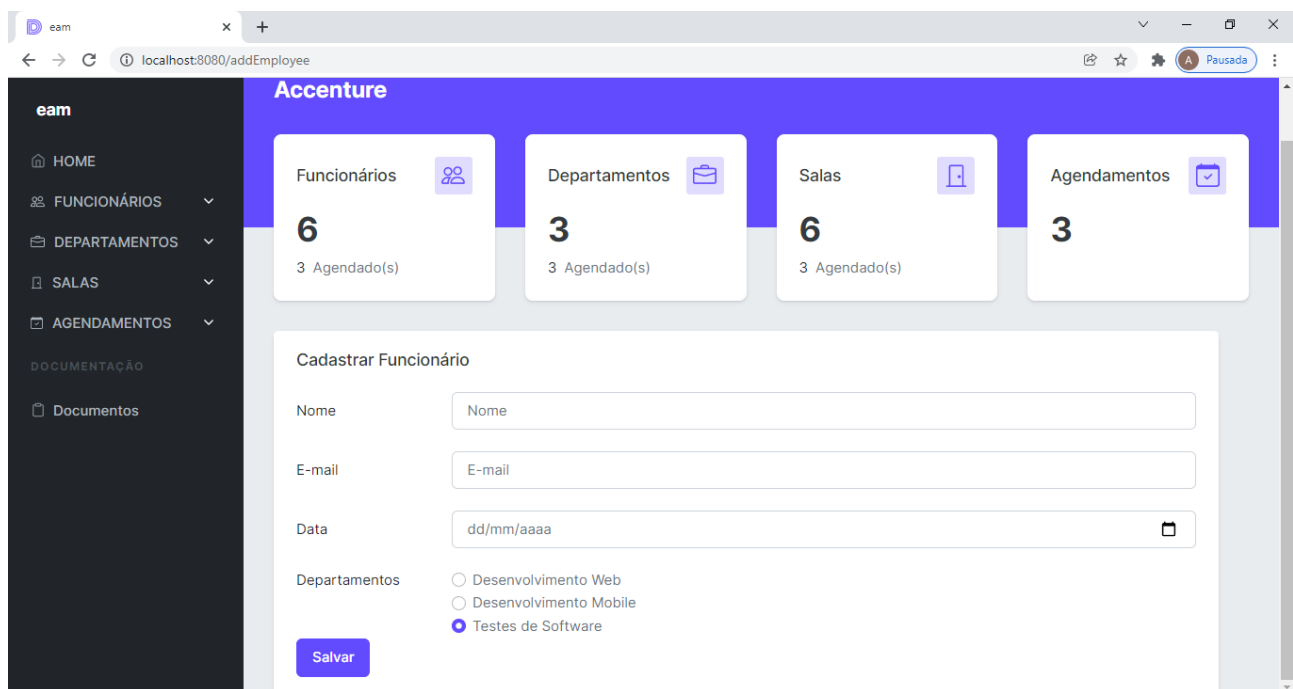
Segue abaixo algumas telas da aplicação:



**Tela 1.** Dashboard inicial



**Tela 2.** Menu de acesso em castata



**Tela 3.** Formulário de cadastro de funcionários

The screenshot displays a web browser window with the URL `localhost:8080/addScheduling`. The application, titled 'eam', features a dark sidebar with navigation links: HOME, FUNCIONÁRIOS, DEPARTAMENTOS, SALAS, AGENDAMENTOS, DOCUMENTAÇÃO, and Documentos. The main content area is titled 'Cadastrar Agendamento' and contains the following form elements:

- ID Agendamento:** A text input field with the placeholder 'Agendamento'.
- Matrícula:** A text input field with the placeholder 'Matrícula'.
- Data Criação:** A date picker field showing 'dd/mm/aaaa'.
- Data Agendamento:** A date picker field showing 'dd/mm/aaaa'.
- Salas Disponíveis:** A list of radio buttons corresponding to room numbers: 301, 302, 402, 404, 603, and 605. The option 605 is currently selected.
- Salvar:** A blue button to submit the form.

At the top of the main area, there are four summary cards, each displaying '3 Agendado(s)'.

**Tela 4.** Formulário de cadastro de agendamentos

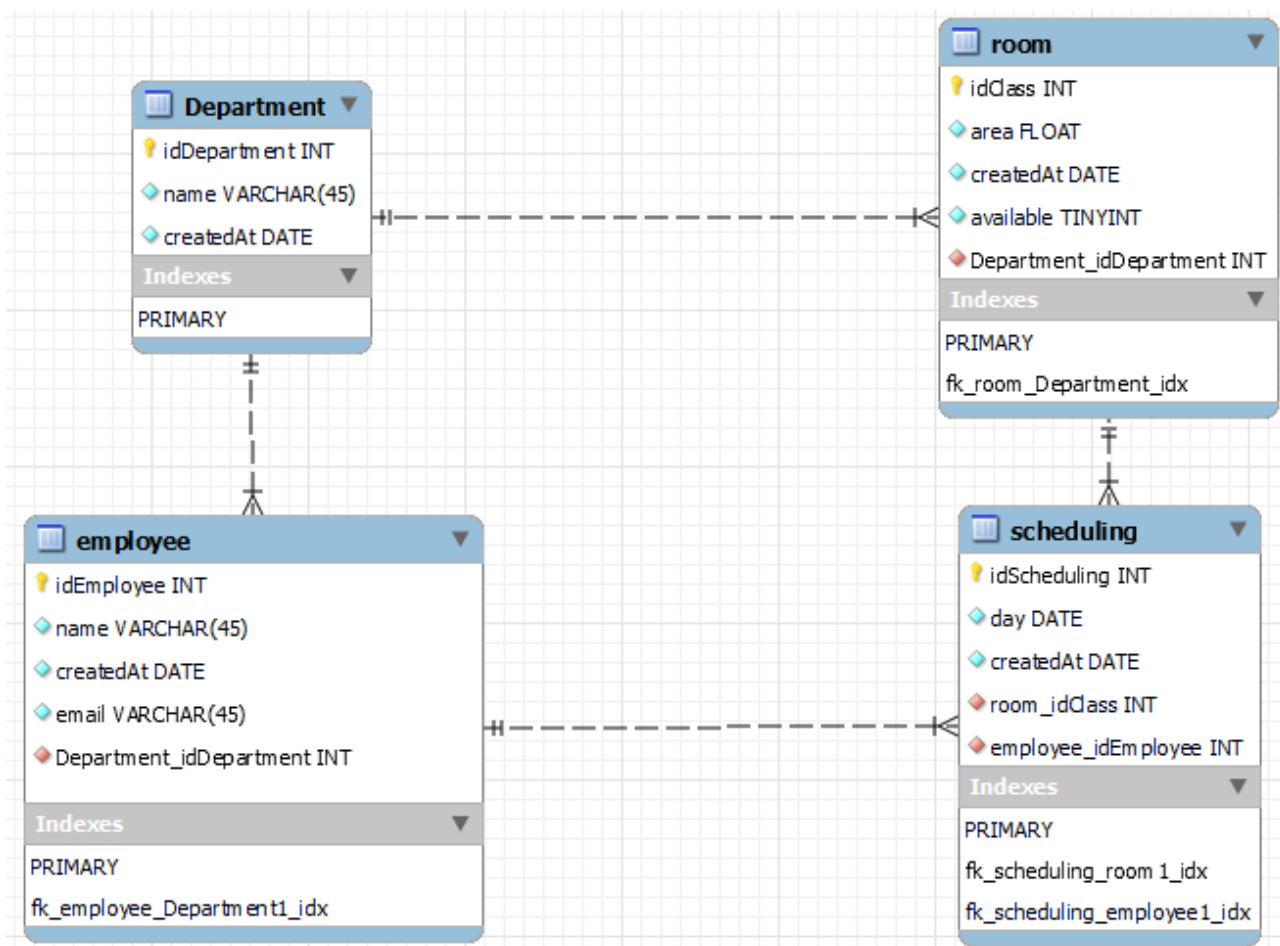
### 3. Estrutura de banco de dados

#### 3.1 Dados persistidos

- Dados dos funcionários;
- Dados dos departamentos;
- Dados das salas dos departamentos;
- Dados dos agendamentos.

#### 3.2 Relação entre as entidades em diagrama

Abaixo temos o diagrama do modelo conceitual que trata da descrição da semântica de aplicativo de software em um alto nível de abstração. Sua principal finalidade é capturar os requisitos de informação e regras de negócio sob o ponto de vista do negócio.



### 3.3 Tecnologias utilizadas



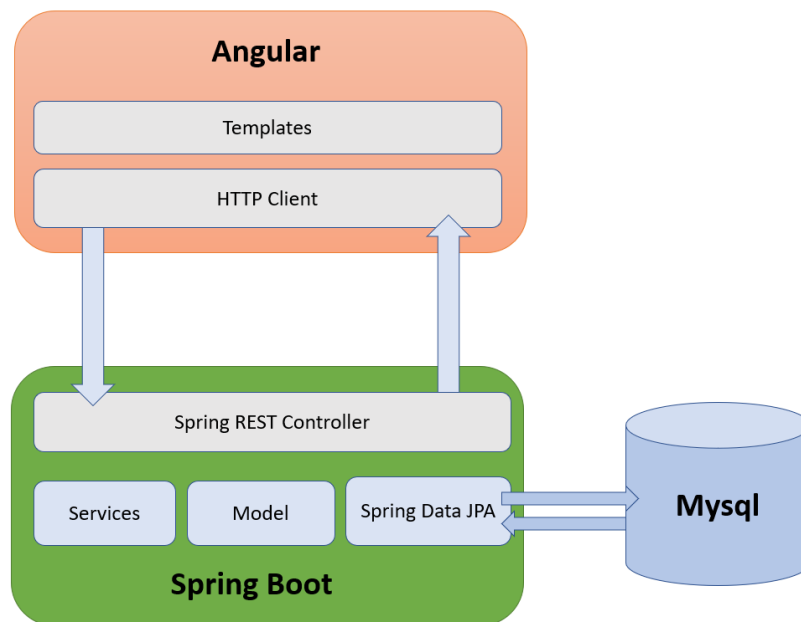
O MySQL é um sistema gerenciador de banco de dados relacional de código aberto usado na maioria das aplicações gratuitas para gerir suas bases de dados. O serviço utiliza a linguagem SQL (Structure Query Language – Linguagem de Consulta Estruturada), que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados. Na criação de aplicações web abertas e gratuitas, o conjunto de aplicações mais usado é o LAMP, um acrônimo para Linux, Apache, MySQL e Perl/PHP/Python. Nesse conjunto de aplicações, inclui-se, respectivamente, um sistema operacional, um servidor web, um sistema gerenciador de banco de dados e uma linguagem de programação. Assim, o MySQL é um dos componentes centrais da maioria das aplicações públicas da Internet.

### 4. Noções da arquitetura da solução

A arquitetura de soluções desenha e implementa recursos e ferramentas de TI para atender as necessidades do negócio. Na prática, isso quer dizer que os objetivos da empresa devem estar alinhados com os softwares, plataformas e infraestruturas que serão criados ou que já estão em uso.

Entre as soluções desenvolvidas, podem estar:

- Manutenção dos serviços;
- Implantação de novos sistemas;
- Manuseio de dados estruturados, entre outros.



De maneira simples, a arquitetura fornece uma perspectiva holística da organização em vez de, simplesmente, uma perspectiva da TI. Assim sendo, quais são as melhores práticas para chegar à transformação digital? Fundamentalmente, tem a ver com os dados. Portanto, há uma necessidade geral por uma Competência de Dados da Empresa.

Sob essa visão, agrupamos as melhores práticas em três categorias de competência:

- Competência de integração: A habilidade de conectar fluxos de dados diversos de toda a organização de modo ágil, eficiente e com melhoria contínua;
- Competência de informação: A capacidade de gerenciar o significado e o contexto e, portanto, o valor de negócios, os dados;
- Competência de transformação: A capacidade de realizar alterações complexas entre funções em resposta às condições do mercado, aos avanços das tecnologias e às oportunidades de negócios, não apenas uma vez, mas como uma disciplina contínua.

### **Mecanismos Arquiteturais**

Os Mecanismos Arquiteturais são soluções comuns para problemas normalmente encontrados que você pode usar durante o desenvolvimento para minimizar a complexidade. O valor na definição de mecanismos de arquitetura é que eles:

- Mostram explicitamente aspectos dos mecanismos da solução que são comuns em todo o sistema. Isto lhe ajuda a planejar;



- Coloca marcadores para que os desenvolvedores construam os aspectos do sistema uma vez e reuse-os. Isto reduz a carga de trabalho;
- Promove o desenvolvimento de um conjunto consistente de serviços. Isto faz com que o sistema fique mais fácil de manter.

Um mecanismo arquitetural pode ter três estados: análise, design e implementação. Estas categorias refletem a maturidade da descrição do mecanismo.

O estado muda à medida que níveis sucessivos de detalhes são descobertos quando você refina os requisitos arquiteturalmente significativo em software funcional. As categorias estão resumidas na tabela a seguir.

### Estados de um Mecanismo Arquitetural

Estado	Descrição
Análise	Uma solução conceitual para um problema técnico comum. Por exemplo, a persistência é uma solução abstrata para o requisito comum de armazenamento de dados. A finalidade desta categoria é simplesmente identificar a necessidade de projetar e implementar um mecanismo arquitetural, e para capturar atributos básicos para esse mecanismo.
Design	Design refinando de um mecanismo de análise em uma tecnologia concreta (por exemplo, um <i>RDBMS</i> ). A finalidade desta categoria é guiar a seleção precisa do produto ou da tecnologia.
Implementação	Um refinamento adicional de um mecanismo de design em uma especificação para o software. Isto pode ser apresentado como um padrão de design ou código de exemplo.

## 4.1 Tecnologias utilizadas no desenvolvimento

### Html e CSS



HTML é uma linguagem usada para a publicação de conteúdo (texto, imagens, vídeos, áudio etc.) na web. Para que você possa entender bem, o HTML é baseado no conceito de hipertexto, que são conjuntos de elementos ligados por conexões, que podem ser palavras, imagens, vídeos, áudio, documentos etc. que quando conectados, formam uma grande rede de informação. A conexão feita em um hipertexto é algo imprevisto que permite a comunicação de dados, organizando conhecimentos e guardando informações relacionadas.

O HTML5 é a nova versão do HTML4 e um dos seus principais objetivos é facilitar a manipulação dos elementos, possibilitando o desenvolvedor modificar as características dos objetos de forma não intrusiva, fazendo com que isso fique transparente para o usuário final.

Para se ter uma ideia disso, diferente das versões anteriores, o HTML5 fornece ferramentas para o CSS e o Javascript fazerem seu trabalho da melhor possível de forma que um web site ou aplicação continue leve e funcional.

CSS é a sigla para Cascading Style Sheets, ou seja, Folhas de Estilo em Cascatas. É uma maneira de dar estilo ao código criado por linguagens como HTML, XML ou XHTML, por exemplo. De forma prática, ela funciona como uma camada de personalização ao conteúdo visível. Estes tipos de linguagem citados foram idealizados para serem funcionais e, por serem de marcação, precisaram ser aperfeiçoados. O CSS, no caso, veio como uma boa solução para aprimorar a estética dos sites.

## **Bootstrap**



O Bootstrap é um framework front-end gratuito e de código-fonte aberto, que disponibiliza componentes prontos para você utilizar em seu projeto. Com ele, você consegue criar e personalizar sites responsivos para dispositivos móveis, desktops e notebooks, com componentes pré-construídos e com plugins JavaScript poderosos. Tudo isso porque toda a estrutura do HTML, CSS e Javascript já se encontram prontas, bastando apenas que o desenvolvedor importe suas dependências em seu projeto.

## **JavaScript**



JavaScript é uma linguagem de programação de alto nível criada, a princípio, para ser executada em navegadores e manipular comportamentos de páginas web. Com seus scripts é possível incluir, em uma página estática, elementos dinâmicos como mapas, formulários, operações numéricas, animações, infográficos interativos e muito mais.

O JavaScript é uma das mais importantes tecnologias voltadas para o front-end e, unindo-se ao trio HTML, CSS e PHP, formam um grupo de linguagens que abrangem praticamente todas as exigências do desenvolvimento de uma página completa, dinâmica e com boa performance.

## Java



Java é uma linguagem de programação orientada a objetos desenvolvida na década de 90 por uma equipe de programadores chefiada por James Gosling, na empresa Sun Microsystems. Em 2008 o Java foi adquirido pela empresa Oracle Corporation. Diferente das linguagens de programação modernas, que são compiladas para código nativo, a linguagem Java é compilada para um bytecode que é interpretado por uma máquina virtual (Java Virtual Machine, mais conhecida pela sua abreviação JVM). A linguagem de programação Java é a linguagem convencional da Plataforma Java, mas não é a sua única linguagem. J2ME Para programas e jogos de computador, celular, calculadoras, ou até mesmo o rádio do carro.

## Spring Boot



O Spring é um framework open source para a plataforma Java criado por Rod Johnson. Trata-se de um framework não intrusivo, baseado nos padrões de projeto inversão de controle (IoC) e injeção de dependência. No Spring o container se encarrega de "instanciar" classes de uma aplicação Java e definir as dependências entre elas através de um arquivo de configuração em formato XML, inferências do framework, o que é chamado de auto-wiring ou ainda anotações nas classes, métodos e propriedades. Dessa forma o Spring permite o baixo acoplamento entre classes de uma aplicação orientada a objetos. O Spring possui uma arquitetura baseada em interfaces e POJOs (Plain Old Java Objects), oferecendo aos POJOs características como mecanismos de segurança e controle de transações. Também facilita testes unitários e surge como uma alternativa à complexidade existente no uso de EJBs. Com Spring, pode-se ter um alto desempenho da aplicação. Esse framework oferece diversos módulos que podem ser utilizados de acordo com as necessidades do projeto, como módulos voltados para desenvolvimento Web, persistência, acesso remoto e programação orientada a aspectos.

## AngularJS



AngularJS é um framework JavaScript de código aberto, mantido pelo Google, que auxilia na execução de single-page applications. Seu objetivo é aumentar aplicativos que podem ser acessados por um navegador web, foi construído sob o padrão model-view-controller (MVC), em um esforço para facilitar tanto o desenvolvimento quanto o teste dos aplicativos. A biblioteca lê o HTML que contém atributos especiais e então executa a diretiva na qual esta tag pertence, e faz a ligação entre a apresentação e seu modelo, representado por variáveis JavaScript comuns. O valor dessas variáveis JavaScript podem ser setadas manualmente, ou via um recurso JSON estático ou dinâmico.

### *5. Ferramentas utilizadas no desenvolvimento do produto*

#### Eclipse



Eclipse é uma IDE para desenvolvimento Java, porém suporta várias outras linguagens a partir de plugins como C/C++, PHP, ColdFusion, Python, Scala e Kotlin. Ele foi feito em Java e segue o modelo open source de desenvolvimento de software.

O projeto Eclipse foi iniciado na IBM que desenvolveu a primeira versão do produto e doou-o como software livre para a comunidade. O gasto inicial da IBM no produto foi de mais de 40 milhões de dólares. Hoje, o Eclipse é o IDE Java mais utilizado no mundo. Possui como característica marcante o uso da SWT e não do Swing como biblioteca gráfica, a forte orientação ao desenvolvimento baseado em plug-ins e o amplo suporte ao desenvolvedor com centenas de plugins que procuram atender as diferentes necessidades de diferentes programadores.

#### GitHub



Git é um sistema distribuído de controle de versões, que permite que os usuários acompanhem mudanças no código-fonte durante o desenvolvimento de software.

Um sistema de controle de versões é bastante útil para manter registro das alterações que são promovidas na base do código do projeto. Ainda dá para ver quem fez cada mudança e recuperar linhas de código que foram deletadas ou alteradas.

A plataforma permite que você faça gratuitamente o upload de seus projetos. Também é possível seguir e acompanhar os principais projetos de desenvolvimento open-source do mundo com bibliotecas e frameworks completos para você conferir. Dependendo do seu interesse e do projeto, dá até para reportar bugs para os desenvolvedores e até mesmo enviar sua própria contribuição com uma correção para o código.

## **Figma**



Prototipagem de software é um processo iterativo de geração de modelos de software que faz parte da análise do ciclo de vida do desenvolvimento do sistema. É a atividade de desenvolvimento de uma versão inicial do sistema baseada no atendimento dos requisitos ainda pouco definidos. Um protótipo simula a aparência e funcionalidade do software permitindo que os clientes, analistas, desenvolvedores e gerentes percebam os requisitos do sistema podendo interagir, avaliar, alterar e aprovar as características mais marcantes na interface e funções.

Figma é um editor gráfico de vetor e prototipagem de projetos de design baseado principalmente no navegador web, com ferramentas offline adicionais para aplicações desktop para GNU/Linux, macOS e Windows. O Figma se destaca pelos seus recursos colaborativos em tempo real. Designers podem trabalhar juntos remotamente. Programadores podem adicionar/escrever seu próprio código em um projeto em tempo real enquanto a equipe vai dando feedback.

### **Vantagens:**

- Configurar determinadas características para um tipo específico e atribui essas características a algum texto no seu layout. A partir dessa configuração é possível modificar todos de uma só vez;
- O auto-layout é uma funcionalidade sobre espaçamento e distribuição de objetos. O layout é adaptado automaticamente. Ele leva em consideração as distâncias entre os elementos e espaçamentos internos horizontais e verticais. Isto é, não importa o tamanho dos elementos, o espaçamento se mantém;
- Os componentes são úteis para definir um componente master para que todas suas “instâncias” sejam subordinadas a ele. Isto é, as alterações feitas no componente master, afetam diretamente às instâncias;
- É possível, através do Figma, a instalação de plugins que agilizam e tornam mais fluido o nosso workflow. Existe uma infinidade de plugins na plataforma.

## ***6. Estratégia e artefatos***

A estratégia do projeto está norteadada na dor das empresas em alocar seus funcionários durante o processo de retorno gradual do trabalho presencial durante a pandemia causada pelo Coronavírus.

Procuramos fazer com que nossa aplicação atenda demandas específicas no mercado. Nosso Benchmark se deu a partir das ideias tiradas nas reuniões, tais como iniciar com uma aplicação para web. Cada nova versão do projeto, sinaliza uma nova pesquisa utilizada no mesmo com acréscimos perceptíveis no andamento dele. Foram usados como ferramentas de gerenciamento, o GoogleMeet para comunicação, o GoogleDrive para troca de documentos e arquivos, e o GitHub para controle de versionamento de código e para armazenamento de dados, documentos e etc.