

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
TRABALHO INTERDISCIPLINAR – APLICAÇÕES WEB

Arthur Crossy Mendes
Daniel Vitor de Oliveira Santos
Danielle Dias Vieira
Luiz Fernando Carneiro Rodrigues
Natã Oliveira Sousa
Rodolfo Sousa Lopes

PROJETO DIARY

Belo Horizonte
2020

Arthur Crossy Mendes
Daniel Vitor de Oliveira Santos
Danielle Dias Vieira
Luiz Fernando Carneiro Rodrigues
Natã Oliveira Sousa
Rodolfo Sousa Lopes

PROJETO DIARY

Trabalho apresentado à disciplina de Trabalho Interdisciplinar – Aplicações Web, da Pontifícia Universidade Católica de Minas Gerais.

Belo Horizonte
2020

LISTA DE FIGURAS

<i>Figura 1 - Kanban [Trello]</i>	14
<i>Figura 2 - Especificação de cores do Kanban</i>	14
<i>Figura 3 - Fluxo de tela do usuário</i>	15
<i>Figura 4 – Estrutura padrão do site</i>	16
<i>Figura 5 - Primeira parte do home page</i>	17
<i>Figura 6 - Segunda parte da home page</i>	18
<i>Figura 7 - Terceira parte da home page</i>	19
<i>Figura 8 - Última página da homepage</i>	20
<i>Figura 9 - Tela - Matérias</i>	20
<i>Figura 10 - Tela de Serviços</i>	21
<i>Figura 11 - Primeira parte da tela - Quem somos</i>	22
<i>Figura 12 - Segunda parte da tela Quem Somos</i>	23
<i>Figura 13 - Contatos</i>	24
<i>Figura 14 - Tela do Login</i>	25
<i>Figura 15 - Tela para recuperação da senha</i>	26
<i>Figura 16 – Tela de aplicação calculadora</i>	27
<i>Figura 17 - Tela da Agenda</i>	28
<i>Figura 18 - Estrutura das notícias do site</i>	29
<i>Figura 19 - Diagrama de componentes</i>	30
<i>Figura 20 - Layout</i>	31
<i>Figura 21 - Acesso a Calculadora</i>	32
<i>Figura 22 - Layout Dúvidas Frequentes</i>	32
<i>Figura 23 - Ofertas para clientes</i>	33
<i>Figura 24 – Layout Quem Somos</i>	33
<i>Figura 25 - Calculadora de investimentos</i>	34
<i>Figura 26 - Layout Contatos</i>	34
<i>Figura 27 – Layout Notícias</i>	35
<i>Figura 28 - Layout Footer</i>	35
<i>Figura 29 - Código fonte</i>	37
<i>Figura 30 - Lista de negociação</i>	38
<i>Figura 31 - Negociação JavaScript</i>	39
<i>Figura 32 - Negociação Controller</i>	40
<i>Figura 33 - Date Helper</i>	41
<i>Figura 34 - Classe adiciona evento</i>	42
<i>Figura 35 - Apaga e limpa formulário</i>	43
<i>Figura 36 – Modelo do Template</i>	43

<i>Figura 37 - Agenda</i>	44
<i>Figura 38 - Console log</i>	45

LISTA DE TABELAS

<i>Tabela 1 – Controle de Entregas Parciais</i>	<i>1</i>
<i>Tabela 2 - Ana Maria Pontes do Nascimento (Persona).....</i>	<i>4</i>
<i>Tabela 3 - Joaquim Barbosa dos Santons (Persona).....</i>	<i>5</i>

SUMÁRIO

1 INTRODUÇÃO.....	2
1.1 Problema.....	2
1.2 Objetivos.....	2
1.2.1 Objetivo Geral	2
1.2.2 Objetivos Específicos	3
2 ESPECIFICAÇÃO DO PROJETO	3
2.1 Personas	3
2.2 Histórias de Usuários.....	5
2.3 Requisitos do Projeto.....	7
2.4 Requisitos Funcionais.....	7
2.5 Requisitos não funcionais.....	8
2.6 Restrições	9
3 METODOLOGIA.....	10
3.1 Ambientes de Trabalho.....	10
3.2 Gestão de código fonte	11
3.3 Gerenciamento do Projeto	13
4 PROJETO DE INTERFACE.....	15
4.1 Fluxo do Usuário (UserFlow).....	15
4.2 Wireframe	16
4.2.1 Tela – Home – Page	17
4.2.2 Tela – Matérias	20
4.2.3 Tela – Serviços	21
4.2.4 Tela – Quem Somos	22
4.2.5. Tela – Contatos	24
4.2.6 Tela – Login	25
4.2.7 Tela – Esqueceu a Senha	26
4.2.8 Tela – Calculadora	27
4.2.9 Tela – Agenda	27
4.2.10 Tela – Notícias	28
5 IMPLEMENTAÇÃO DA SOLUÇÃO.....	30
5.1 Arquitetura de Solução	30
5.2 Template padrão do site.....	31
5.3 Layouts	31
5.3.1 Layout Navbar e Carousel	31

5.3.2 Layout acesso a calculadora (home)	31
5.3.3 Layout Dúvidas Frequentes	32
5.3.4 Layout Benefícios	32
5.3.5 Layout Quem Somos	33
5.3.6 Layout Calculadora (Serviços)	33
5.3.7 Layout Contatos:	34
5.3.8 Layout Notícias	34
5.3.9 Layout Footer	35
5.4 Orientações sobre a responsividade.....	36
5.5 Organização do código-fonte.....	36
5.6 Funcionalidades do Sistema	37
REFERÊNCIA BIBLIOGRÁFICA.....	46



PUC Minas

Portal de Notícias

Trabalho Interdisciplinar - Aplicações Web

Tabela 1 – Controle de Entregas Parciais

Participantes	Entrega inicial	Entrega Parcial 1	Entrega Parcial 2	Entrega Parcial 3	Entrega Final
	20 pts	15 pts	15 pts	15 pts	35 pts
Arthur	Grupo	Funcionalidade da Calculadora			Grupo
Daniel		Template Padrão do Site			
Danielle		Metodologia			
Luiz		Documentação de Contexto			
Natã		Projeto de Interface			
Rodolfo		Arquitetura de Solução			

Importante

Este documento é um modelo do projeto a ser criado para a disciplina de **Trabalho Interdisciplinar - Aplicações Web** da PUC Minas. Siga [este link](#) para mais detalhes sobre o processo de avaliação e orientações. Este [outro link](#) apresenta uma estrutura no GitHub de sugestão para o repositório do projeto.

1 INTRODUÇÃO

Em paralelo ao avanço do tempo, as finanças das pessoas continuam evoluindo cada vez mais, se tornando mais complexas, e com isso muitas pessoas estão perdendo cada vez mais dinheiro.

Esse fato pode vir a ocorrer devido à falta de conhecimentos simples, como matemática financeira e relação de taxa de juros. Além disso, as classes mais elevadas da pirâmide etária, sofrem com o denominado “analfabetismo digital”, aumentando cada vez mais essa bola de neve financeira. Devido a isso, avalia-se a necessidade de corrigir essas questões, pois uma grande parcela da população sofre com esse problema.

Sendo assim, uma aplicação web, com um mecanismo que permita ao usuário um acesso interativo, confiável, e além de tudo, que ofereça informações acerca de sua condição financeira e da condição financeira mundial, pode vir a ser a solução ideal para resolver esse problema.

1.1 Problema

A problemática que rodeia esse projeto é a questão de corrigir ou atenuar uma dificuldade muito grande presente em uma enorme parcela da população brasileira, a qual seria a má gestão financeira.

Devido a um ensino pouco prático no Brasil, que negligenciou o ensino de educação financeira e informática, fez-se com que esse grupo de pessoas percebesse a necessidade de uma plataforma web confiável e interativa, para gerir com segurança as finanças, calcular o verdadeiro montante financeiro, de compras e investimentos, atualizar os leitores com notícias a respeito da gestão de finanças, criar uma agenda financeira para facilitar a gestão. Tudo isso deverá ser gerado com confiabilidade, agilidade, eficiência e segurança.

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é a criação de um dispositivo web, que, realize cálculos financeiros para diversos fins, que oscilam desde compras do dia a dia até investimentos, que também tenha um portal de notícias atualizadas sobre gestão financeira e uma agenda para facilitar a gestão. Além disso, que seja interativa, e que permita o acesso a uma grande parte da população.

1.2.2 Objetivos Específicos

- Fornecer uma calculadora que realize cálculos financeiros, desde compras comuns, até cálculos financeiros um pouco mais robustos;
- Criação de uma agenda para gestão financeira, como fluxo de caixa;
- Portal de notícias atualizadas sobre finanças;

2 ESPECIFICAÇÃO DO PROJETO

A definição exata do problema e os pontos mais relevantes a serem tratados neste projeto foram consolidados com a participação dos usuários em um trabalho de imersão feita pelos membros da equipe a partir da observação dos usuários em seu local natural e por meio de entrevistas. Os detalhes levantados nesse processo foram firmados por das personas e das histórias de usuários.

2.1 Personas

As personas levantadas durante o processo de entendimento do problema são apresentadas nas tabelas que se seguem.

Tabela 2 - Ana Maria Pontes do Nascimento (Persona)


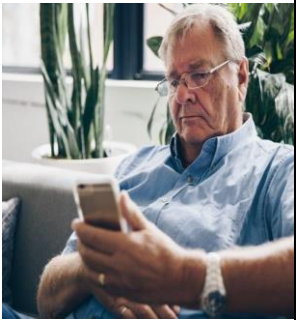
	Ana Maria Pontes Nascimento	
Motivações: <ul style="list-style-type: none"> • Viajar mais • Quitar suas dívidas • Trocar de carro • Gerenciar sua renda • Pagar os estudos dos netos 	Idade: 55 Ocupação: Desembargadora, trabalha no Tribunal de Justiça de Minas Gerais, se destacando nas áreas de ciência, educação, segurança e política.	Aplicativos: <ul style="list-style-type: none"> • Facebook • Araújo (farmácia) • Rádio Saudade • Agenda • WhatsApp
Personalidade: <ul style="list-style-type: none"> • Uma pessoa atenciosa, gentil, passiva e que adora conversar, principalmente sobre os netos. No tempo livre gosta de ir ao Shopping e assistir a canais no Youtube relacionados a finanças. 	Frustrações: <ul style="list-style-type: none"> • Perder o emprego • Ser refém da crise • Não ter controle de gastos • Medo de fracassar, devido as dívidas 	Hobbies: <ul style="list-style-type: none"> • Jardinagem • Assistir novela • Viajar para outro país • Gastar no Shopping • Tricô
	Sonhos: <ul style="list-style-type: none"> • Ter uma velhice tranquila e longa com o marido, e acompanhar a criação dos netos, e não menos importante, ter uma casa linda para decorá-la e trocar o seu carro atual. 	Organização financeira: <ul style="list-style-type: none"> • Apesar de bem sucedida profissionalmente, nunca teve controle de gastos, não recebeu educação financeira e desde muito jovem adquiriu diversas despesas, mesmo sem condição de paga-las.

Tabela 3 - Joaquim Barbosa dos Santos (Persona)

	Joaquim Barbosa dos Santos	
	Idade: 63 Ocupação: Servidor Público, trabalha como professor de biologia marinha na UFRJ, se destacando nos principais campos da biologia.	Aplicativos: <ul style="list-style-type: none"> ● Agenda ● WhatsApp ● Rádio FM ● Cartola FC ● Facebook
Motivações: <ul style="list-style-type: none"> ● Comprar um sítio ● Melhorar a qualidade de vida ● Conhecer mais sobre investimentos ● Conquistar o reconhecimento da família 	Frustrações: <ul style="list-style-type: none"> ● Sente dificuldade com a tecnologia ● Gastos inesperados ● Perder o cargo ● Não conseguir apoio na família ● Falta de compromisso 	Hobbies, História: <ul style="list-style-type: none"> ● Pescar ● Futebol ● Churrasco ● Jogo do Bicho ● Bingo
Personalidade: <ul style="list-style-type: none"> ● Típico Baby Boomer que adora churrasco e ama sertanejo raiz, possui temperamento calmo e amável. É também um exemplar pai de família, e adora estar informado sobre a situação política e econômica do país. 	Sonhos: <ul style="list-style-type: none"> ● Sonha em aumentar seu patrimônio, ter uma aposentadoria tranquila e comprar um sítio em Piracema. Não menos importante, deseja ajudar a sua família em questões financeiras e dívidas. 	Organização financeira: <ul style="list-style-type: none"> ● Nunca teve educação financeira, cresceu com a ideia de “dinheiro foi feito para gastar”, mas ao longo da vida foi vendo que não é assim, hoje deseja se organizar financeiramente.

2.2 Histórias de Usuários

A partir da compreensão do dia a dia das personas identificadas para o projeto, foram registradas as seguintes histórias de usuários.

Eu como ... [PERSONA]	... quero/desejo ... [O QUE]	... para [POR QUE]
Joaquim Barbosa	Ter acesso rápido e fácil a uma aplicação de gestão financeira	Aprender mais sobre como investir corretamente
Joaquim Barbosa	Ter uma ferramenta capaz de me ajudar a planejar e controlar meus gastos	Gastar menos tempo anotando em minha agenda física
Ana Maria	Aprender gerenciar os meus investimentos	Ter a minha independência financeira
Ana Maria	Uma plataforma didática e simples, voltada ao público idoso	Sentir menos dificuldade ao acessar o site
Joaquim Barbosa	Compartilhar as informações que achar relevante no site	Poder discutir com a família e amigos
Joaquim Barbosa	Ler e ter acesso as fontes de notícias	Ter certeza de que a informação é confiável
Ana Maria	Anotar os meus gastos em uma plataforma digital	Poupar o tempo e agilizar o acesso a qualquer lugar
Ana Maria	Comentar nos posts e pontuar minhas dúvidas	Ter um maior entendimento sobre o assunto
Joaquim Barbosa	Acessar tanto no desktop quanto no mobile	Estudar sobre os assuntos do meu interesse em qualquer horário
Joaquim Barbosa	Utilizar uma plataforma segura e confiável	Não cair em golpes ou ser vítima de hackers
Ana Maria	Um Sistema não burocrático, ágil e simples	Facilitar a procura por informações
Ana Maria	Aumentar a minha renda	Realizar os meus sonhos

2.3 Requisitos do Projeto

O escopo funcional do projeto é definido por meio dos requisitos funcionais que descrevem as possibilidades interação dos usuários, bem como os requisitos não funcionais que descrevem os aspectos que o sistema deverá apresentar de maneira geral. Estes requisitos são apresentados a seguir.

2.4 Requisitos Funcionais

A tabela a seguir apresenta os requisitos do projeto, identificando a prioridade em que os mesmos devem ser entregues.

ID	Descrição	Prioridade
RF-01	O site deve apresentar uma plataforma didática, sem muitos conceitos técnicos para facilitar o entendimento para os idosos.	Alta
RF-02	O site deve apresentar, para cada notícia, uma ou mais imagem correspondente ao assunto apresentado.	Baixa
RF-03	O site deve permitir o compartilhamento das informações expostas para os usuários discutirem fora da plataforma.	Baixa
RF-04	O site deve fornecer ao usuário uma ferramenta capaz de ajudar a gerenciar e organizar seus gastos.	Média
RF-05	O site deve disponibilizar os links das fontes de pesquisa utilizadas para montar as informações mostradas	Média
RF-06	O site tem que demonstrar segurança ao público, para que ele se sinta confortável ao utilizá-lo.	Média
RF-07	O site precisa ter um menu de fácil acesso aos usuários, para assim conseguirem achar o que necessitam facilmente.	Média
RF-08	O site deve exibir os comentários feitos por outros usuários em cada post.	Baixa
RF-09	O site deve se atentar a experiência do usuário.	Média
RF-10	O site deve ter um design atrativo, para transmitir confiança no nosso	Média

	produto.	
RF-11	O site deve evitar assuntos polêmicos, para não afastar o público do foco principal.	Baixa
RF-12	O site deve ser focado na administração das finanças, ensinando os usuários e dando dicas de como melhorar e alcançar seus objetivos.	Alta
RF-13	O site deverá disponibilizar uma área de cadastro para os usuários.	Média
RF-14	O site precisa garantir a segurança dos dados dos usuários.	Baixa
RF-15	O site precisa ter um bom desempenho em qualquer aplicativo, seja ele em mobile ou desktop, priorizando a rapidez.	Média

2.5 Requisitos não funcionais

A tabela a seguir apresenta os requisitos não funcionais que o projeto deverá atender. São eles, desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas.

ID	Descrição	Prioridade
RNF-01	O site deve ser publicado em um ambiente acessível publicamente na Internet (GitHub Pages);	Alta
RNF-02	O site deverá ser responsivo permitindo a visualização em um celular de forma adequada	Alta
RNF-03	O site deve ser implementado na linguagem JavaScript, HTML e CSS	Alta
RNF-04	O site deve ser executável em qualquer plataforma e navegadores, como Google Chrome, Microsoft Edge, Mozilla Firefox e Internet Explorer.	Alta

RNF-05	O site deve atender a alguns requisitos de acessibilidade na web, como, fontes mais avantajadas, contrastes de cor mais nítidos e botões de boa visibilidade.	Alta
RNF-06	O site não deverá em hipótese alguma utilizar os dados dos usuários para fins maliciosos.	Baixa

2.6 Restrições

As questões que limitam a execução desse projeto e que se configuram como obrigações claras para o desenvolvimento do projeto em questão são apresentadas por meio desse tópico na tabela a seguir.

ID	Descrição
RE-01	O projeto deverá ser entregue, não podendo extrapolar a data de 14/09/2020.
RE-02	O aplicativo deve se restringir às tecnologias básicas da Web no Frontend.
RE-03	A equipe não pode subcontratar o desenvolvimento do trabalho, tudo deve ser feito apenas pelos integrantes do grupo.
RE-04	Nenhum integrante da equipe deve realizar mudanças sem o consentimento de todos.
RE-05	Utilizar plataformas de trabalho em equipe online, como o Miro para organizar as ideias referentes ao projeto.
RE-06	Reunir no mínimo 3 vezes por semana para discutir sobre o andamento do projeto.
RE-07	O trabalho deve ser distribuído em parcelas iguais, para não sobrecarregar nenhum dos integrantes.

3 METODOLOGIA

A metodologia contempla as definições ferramentais que serão utilizados pela equipe tanto para a manutenção dos códigos e demais artefatos, quanto para a organização do time na execução das tarefas do projeto.

3.1 Ambientes de Trabalho

Os artefatos do projeto são desenvolvidos a partir de diversas plataformas e a relação dos ambientes com seu respectivo propósito é apresentada na tabela que se segue.

Ambiente	Plataforma	Link de Acesso
Repositório de código fonte	GitHub	https://github.com/ICEI-PUC-Minas-PMGCC-TI/tiaw-pmg-cc-t-20202-gestao-de-investimentos
Documentos do projeto	GitHub	https://github.com/ICEI-PUC-Minas-PMGCC-TI/tiaw-pmg-cc-t-20202-gestao-de-investimentos/tree/master/docs
Gerenciamento do Projeto	Trello	https://trello.com/b/dn1hSkir/projeto-diary
Projeto de Interface e Wireframes	MarvelApp	https://marvelapp.com/prototype/25j896dg/section/1150219
Editor de código-fonte	Visual Studio Code	https://code.visualstudio.com/

Foi escolhido o GitHub para Repositório de código fonte e dos documentos do projeto, pois é uma plataforma que pode ser utilizada por qualquer usuário de forma grátis e porque tem seu controle de versão baseado no Git. Ela também é uma plataforma que alguns membros da equipe já tinham familiaridade para trabalhar. E como são 6 contribuidores ao mesmo tempo, com o GitHub e o Git, é possível várias pessoas trabalhando em suas tarefas, tendo menos risco de conflitos no código.

Foi escolhido o aplicativo Trello para gerenciar as tarefas do projeto, pois também é grátis e tem cartões com as tarefas de forma intuitiva. É uma ferramenta que se pode utilizar com facilidade.

Foi escolhido o aplicativo MarvelApp, pois para criar o design e Wireframe não é necessário nenhum código, é possível realizar designs interativos de forma intuitiva e também é grátis a plataforma.

Foi escolhido o editor de código fonte Visual Studio Code, pois ele tem controle de Git já incorporado, é possível instalar várias extensões de linguagens, edição de temas e idiomas, conectar serviços adicionais, entre outros. Todos da equipe já utilizam este software facilitando o trabalho e a interação de todos.

É utilizado também as redes sociais Whatsapp e Microsoft Teams para a comunicação da equipe, pois são aplicativos que toda a equipe já conhece e tem facilidade no uso do dia a dia.

As ferramentas utilizadas para construir os diagramas foi o Miro e o Dia (GNOME), porque os membros já conheciam estes softwares, são intuitivos e gratuitos.

3.2 Gestão de código fonte

A ferramenta de controle de versão adotada no projeto foi o Git, sendo que o Github foi utilizado para hospedagem do repositório.

Para gestão do código fonte do software desenvolvido pela equipe, o grupo utiliza um processo baseado no Gitflow, que é um design de fluxo de trabalho Git que foi criado por Vincent Driessen (2010).

Como há 6 contribuidores neste projeto Diary, é importante o uso do Gitflow para se ter o controle do que está sendo produzido pela equipe, como são corrigidas as falhas e o implemento de funcionalidades. Desta forma, tudo se torna mais organizado. Abaixo há a descrição da forma padrão do projeto:

O projeto segue a seguinte convenção para o nome de branches:

- **Master:** Versão estável já testada do software. É onde teremos o produto final para entregar para o cliente.
- **Develop:** Versão de desenvolvimento do software. É para integração das outras branches e funcionalidades, de forma que haja manutenção e verificação se o código

está funcionando de acordo com o desejo do cliente. Está em paralelo com o branch master.

- **Release:** Prepara o projeto para uma nova versão e quando o branch develop estiver com as funcionalidades prontas para entrega de uma sprint, fazemos o merged com a develop e a master.

Quanto à gerência de issues, o projeto adota a seguinte convenção:

- **Hotfix:** É criado quando necessária correção de algum erro e conflito durante a criação do produto. Terá o merged feito no branch ou branches onde apresentarão os erros.
- **Feature:** Quando uma nova funcionalidade precisa ser introduzida.

Quanto à gerência de commits, o projeto segue a seguinte convenção:

- **Fix:** Quando é necessário a correção de um problema (bug).
- **Feat:** Quando inclui um novo código no projeto para uma nova funcionalidade.
- **Docs:** Quando se inclui ou altera algo na documentação do projeto.
- **Style:** Quando se altera ou inclui algo somente visual (CSS) no sistema.
- **Refactor:** Quando se altera o código com a finalidade da melhoria de qualidade do código, sem alterar as funcionalidades ou aparência do projeto.

Quanto à gerência de tags, utilizaremos as tags anotadas, pois elas são um armazenamento completo de objetos no banco de dados do Git. Elas contém marcações de nome, e-mail, data, entre outras informações importantes para verificar quando preciso, aquela versão.

Quanto à gerência de mergeds:

- **Develop:** Quando o código desse branch chegar ao nível estável, todas as alterações devem ser mergeadas de volta ao master juntamente com a branch release.
- **Feature:** Assim que concluída, é mesclada (merged) com a branch develop. Nunca irá interagir diretamente com o branch master.
- **Release:** Ao final da sprint, quando é feito a tag para a descrição da versão do projeto, este branch é mesclado com a branch master e depois com a develop.
- **Hotfix:** É também mesclada com o branch master e develop, após a correção do erro da funcionalidade.

3.3 Gerenciamento do Projeto

A equipe utiliza metodologias ágeis, tendo escolhido o Scrumban como base para definição do processo de desenvolvimento. Scrumban (termo cunhado pelo Corey Ladas em 2008), é uma metodologia de desenvolvimento Agile que é um híbrido de Scrum e Kanban. Usa a natureza prescritiva do Scrum para ser Ágil. Usa a melhoria de processo do Kanban para permitir que a equipe melhore continuamente seu processo.

Foi escolhido o Scrumban por ter o melhor das metodologias de desenvolvimento Agile, por trabalharmos com o que antecede o desenvolvimento da sprint e com o desenvolvimento da mesma. É um processo que garante o trabalho contínuo e reduz o tempo ocioso dos membros da equipe. Se necessário, pode-se fazer melhorias e revisão da sprint. Todos da equipe trabalham juntos para atingir o objetivo do projeto Diary.

A equipe está organizada da seguinte maneira:

Scrum Master: Danielle Dias Vieira

Equipe de Desenvolvimento:

- o Arthur Crossy Mendes;
- o Daniel Vitor de Oliveira Santos;
- o Danielle Dias Vieira;
- o Luiz Fernando Carneiro Rodrigues;
- o Natã Oliveira Sousa;
- o Rodolfo Sousa Lopes.
- Equipe de Design:
 - o Daniel Vitor de Oliveira Santos;
 - o Natã Oliveira Sousa;

Para organização e distribuição das tarefas do projeto, a equipe está utilizando o Trello estruturado com as seguintes listas:

- Recursos: Esta lista contém tarefas que são como legendas das cores de cada cartão e também serão utilizadas para cópia de tarefas, agilizando assim a criação de novos cartões.
- Backlog: Todos os artefatos identificados por todo o projeto. Representa o Product Backlog.
- To Do: Esta lista é o Sprint atual que estamos trabalhando.
- Doing: Quando uma tarefa tiver sido iniciada, ela é movida para esta lista.

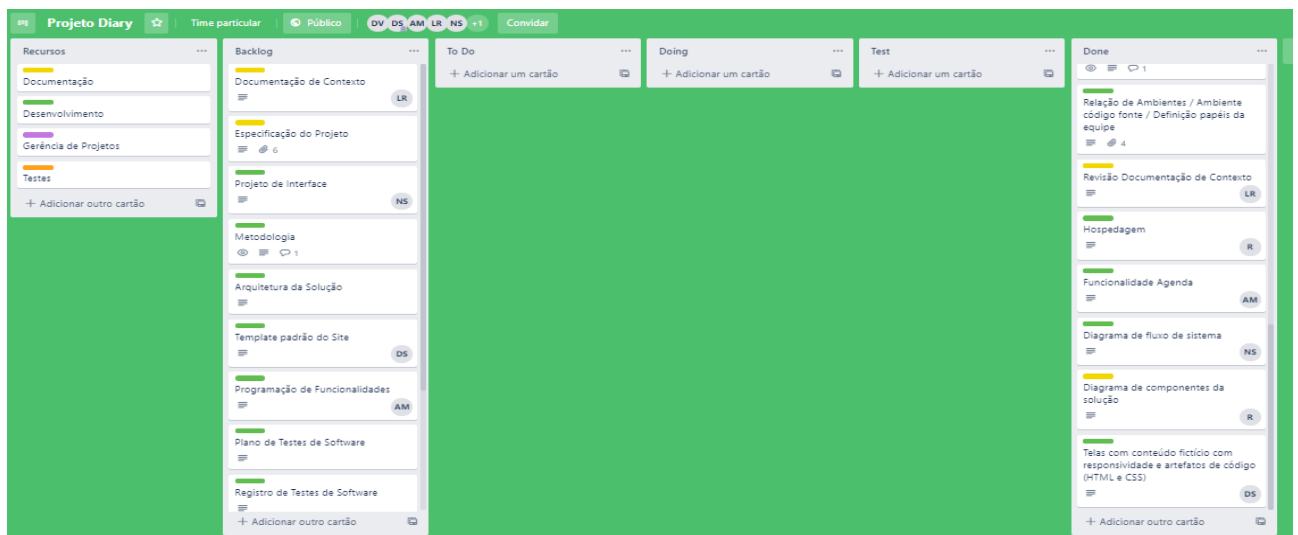
- Test: Quando as tarefas são concluídas são movidas para esta lista, são testadas e é feito a revisão.
- Done: Nesta lista são colocadas as tarefas concluídas, as que já foram revisadas e estão prontas para serem entregues para o cliente.

O quadro do Kanban do grupo está disponível no Trello, através da URL a seguir:

- Trello: <https://trello.com/b/dn1hSkir/projeto-diary>

A definição desta estrutura teve como referência a proposta feita por Andrew Littlefield em 2016. Ela está representada, no estado atual, pela imagem abaixo:

Figura 1 - Kanban [Trello]



As tarefas seguem o seguinte esquema de cores:

Figura 2 - Especificação de cores do Kanban



4 PROJETO DE INTERFACE

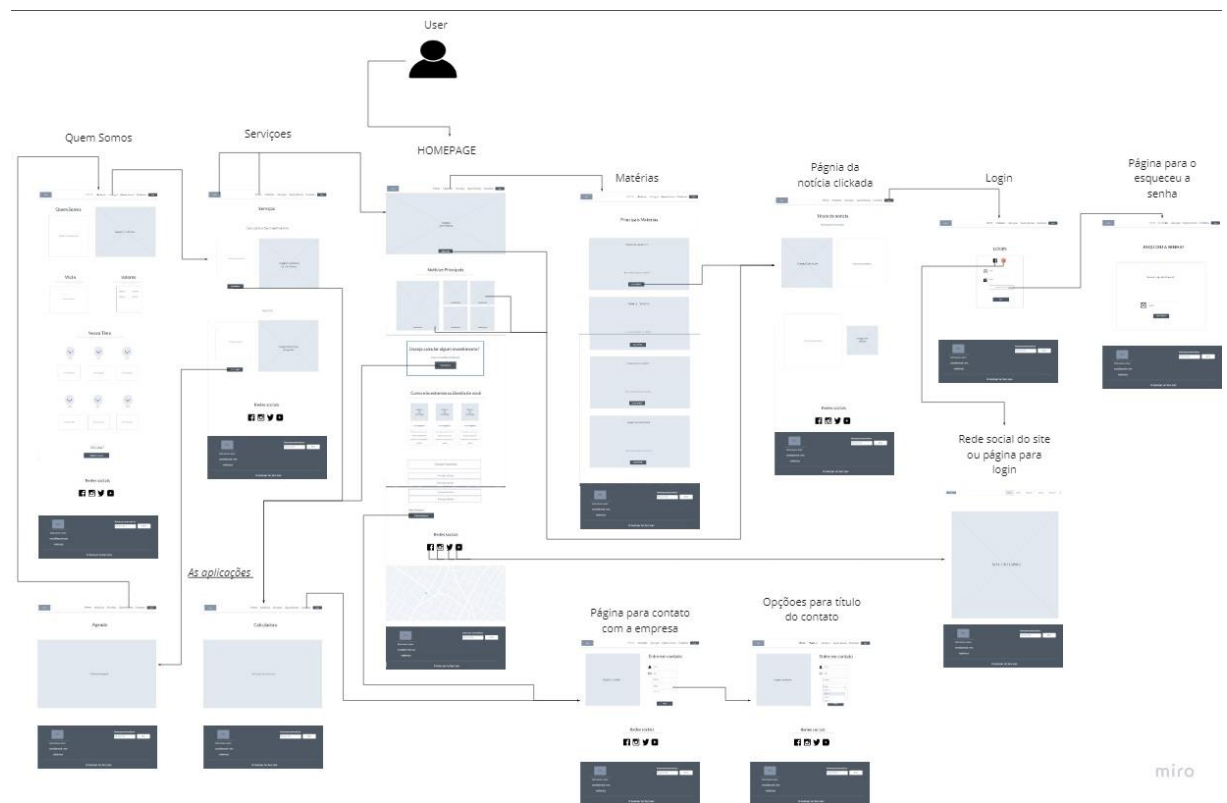
Em consequência do público-alvo que se pretende alcançar por meio dessa aplicação, Mirou-se, principalmente, em requisitos que ampliassem seu uso por usuário leigos na internet. Tais requisitos podem ser listados como usabilidade e acessibilidade, esqueletos fundamentais na montagem da interface do sistema do projeto em questão. Para essa realização, fez-se necessário uma padronização das páginas, para que a aplicação seja utilizada não apenas no ambiente desktop, mas também no mobile.

4.1 Fluxo do Usuário (UserFlow)

O fluxo do usuário é um dos tópicos no projeto de interface, e tem como principal objetivo mostrar a interação do usuário com a interface do site. Para que isso seja possível, é válido a utilização de wireframes na intenção de demonstrar os caminhos e as finalidades do usuário ao acessar determinada página. Vale ressaltar que, para que sejam indicados os diversos caminhos que o usuário pode ter no âmbito do site, foi utilizado setas que partem de um ponto (na qual o usuário pode clicar), e chegam até uma página ou ação, que representa a finalidade do usuário.

A figura abaixo (Figura 3) mostra o fluxo de usuário do projeto.

Figura 3 - Fluxo de tela do usuário



Para visualizar o fluxo do usuário com maior definição e liberdade, acesse: https://miro.com/app/board/o9J_kiuetwU=/. É possível também visualizar as telas do wireframe do projeto de maneira interativa, todas com a aplicação do fluxo. Para isso, acesse: <https://marvelapp.com/prototype/25j896dg/section/1150219>.

4.2 Wireframe

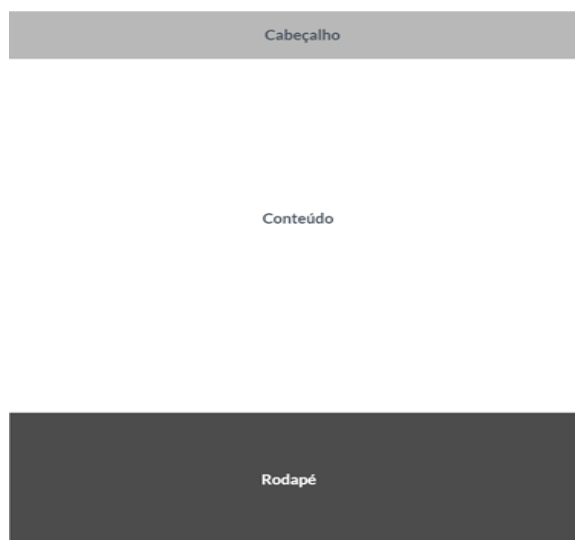
Os Wireframes, são protótipos de tela de baixa fidelidade que tem como objetivo fazer um “esboço” do site. Ademais, sua importância transcende uma simples imagem de uma tela, e pode evitar grandes problemas futuros na organização da interface das páginas.

As telas do projeto, como dito no item anterior, seguem um padrão bem definido. Esse padrão é constituído por três itens, são eles:

- Cabeçalho: Local inicial da página onde se encontra os elementos de identidade, como, a logo do site/empresa, e, os elementos de navegação que facilitam no fluxo entre as telas do site.
- Conteúdo: Local onde está organizado as informações e serviços de uma determinada página do site.
- Rodapé: Local específico que se encontra no final da tela e tem como objetivo passar ao usuário informações complementares sobre o site.

A figura a seguir (Figura 4) exemplifica com exatidão, como, os três itens relacionados anteriormente, estão dispostos na tela das páginas do site do projeto.

Figura 4 – Estrutura padrão do site



4.2.1 Tela – Home – Page

A tela home-page é a tela principal do site. Será nela que o usuário, uma vez que tenha acessado o site, irá ser direcionado. Por esse motivo, essa página é a que apresenta mais informações. Logo, é a mais extensa. Pensado nisso, foi adicionado na página mais elementos gráficos do que textuais, fazendo com que o usuário se canse menos ao rolar o conteúdo da tela.

Em decorrência do tamanho, e das limitações na representação (baixa qualidade ao representar a tela inteira), A tela será dividida em 4 parte.

A primeira figura a ser exemplificada (Figura 5) diz respeito a primeira parte da tela home. Nela, contém o elemento inicial (cabeçalho) e a parte inicial do conteúdo, na qual será a primeira a ser visualizada pelo usuário quando acessar a home-page.

Figura 5 - Primeira parte do home page



Pode-se observar que na primeira parte da página do home-page, não houve presença de elementos puramente textuais. Isso acontece porque o objetivo da primeira parte da página é de prender o usuário ao site. Para isso, é fundamental o uso de imagens ao invés de textos, pois além de serem elementos gráficos, são muito mais fáceis de se ler, o que se é buscado constantemente para cumprir o requisito de acessibilidade.

Nessa primeira parte é colocado uma notícia mais importante no centro e com um tamanho considerável, junto a um botão que redirecionará o usuário para uma página interna, onde ele poderá ler mais sobre essa notícia. Abaixo tem a presença do tópico “Notícias Principais” nela também são encontradas mais notícias sobre o mercado financeiro, gerenciamento de finanças, Ações, Investimentos e relacionados, e, quando clicados, também redireciona o usuário para a página específica daquela notícia.

A segunda parte do home-page é mostrada por intermédio da figura 6.

Figura 6 - Segunda parte da home page



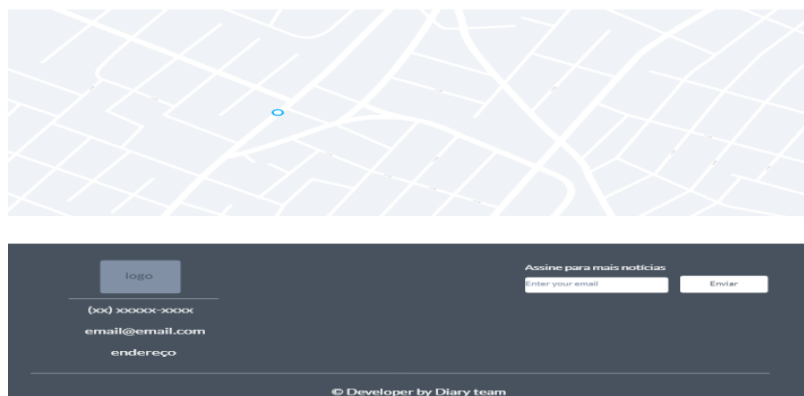
Aqui vemos um atalho para acessar a calculadora mais facilmente (novamente instigando a acessibilidade como alvo principal do site). Logo abaixo a página apresenta o tópico “como cuidamos de você”, na qual mostrará uma breve introdução das tecnologias que estão sendo usadas na confecção do site. Esse elemento se fez necessário pois na pesquisa de campo realizada, descobriu-se que há uma desconfiança muito grande por parte da sociedade no que diz respeito a site de finanças.

Figura 7 - Terceira parte da home page



Na imagem acima é possível observar a presença do tópico “Dúvidas Frequentes”, na qual apresenta alguns dropdowns com as principais dúvidas do usuário. Logo abaixo, em “Mais Dúvidas”, o botão “Fale Conosco”, redireciona para a página “Contatos”. Ainda abaixo, apresentasse as redes sócias do site, que, quando clicados, redirecionam o usuário a uma página externa (nesse caso a rede social em questão). Esse elemento e o anterior (“Mais Dúvidas”) aparecerão em outras telas distribuídas pelo site.

Figura 8 - Última página da homepage



Nesta figura, pode-se ver o elemento do rodapé e a localização da empresa. Esse elemento intenciona trazer também mais conforto ao usuário.

4.2.2 Tela – Matérias

Essa tela consiste em uma série de notícias reunidas e pode ser acessada facilmente pelo cabeçalho do site.

Por representar um nível de complexidade de elementos baixa, pode-se dizer que essa página contém basicamente três elementos, são eles, cabeçalho, principais matérias e o rodapé.

Segue-se abaixo a (Figura 9), a qual demonstra o que foi dito.

Figura 9 - Tela - Matérias



As notícias distribuídas no tópico “Principais Matérias” Seguem o padrão, imagem da notícia, na qual o exemplificará graficamente, a descrição da notícia, que dará ao usuário uma breve explicação da matéria, e o botão “Ir para notícia”, na qual redireciona o usuário para uma página (interna) com a notícia em um grau de detalhamento maior.

4.2.3 Tela – Serviços

A tela de serviços é mais uma das telas do site, e é acessível pelo cabeçalho. Nela será possível acessar as aplicações que serão oferecidas. São elas, a calculadora e a agenda.

Retirando o cabeçalho e o rodapé, que são fixos e constantes no site, a página de serviços se apresenta figurado na imagem da figura 10 a seguir.

Figura 10 - Tela de Serviços



Na tela acima temos um elemento texto, que norteia a página onde o usuário se encontra. Abaixo, vemos dois subtítulos, ambos com os nomes das aplicações oferecidas pelo site. Mais abaixo, estão dispostos elementos de texto, que descrevem brevemente as aplicações. Rolando

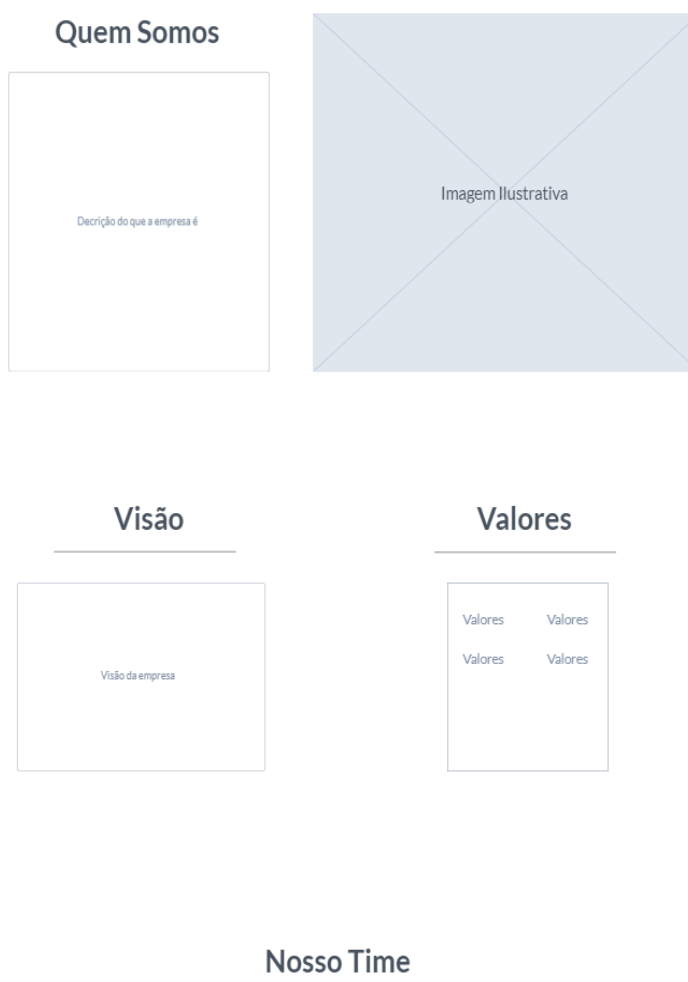
a tela da página em questão, é visível um elemento botão, na qual redireciona o usuário para a aplicação escolhida. Ao lado, vemos uma imagem ilustrativa que se relaciona graficamente com a aplicação descrita pelo subtítulo. Ao final do conteúdo da tela, é apresentado novamente o acesso às redes sociais da página.

4.2.4 Tela – Quem Somos

Nesse tópico, é mostrado uma das mais importantes telas de qualquer página web de uma empresa. É aqui que o usuário terá acesso a informações sobre quem ou o que gerencia o site no intuito de trazer mais confiança a quem utiliza a página.

Na figura abaixo, é mostrado como isso foi aplicado no site da página.

Figura 11 - Primeira parte da tela - Quem somos

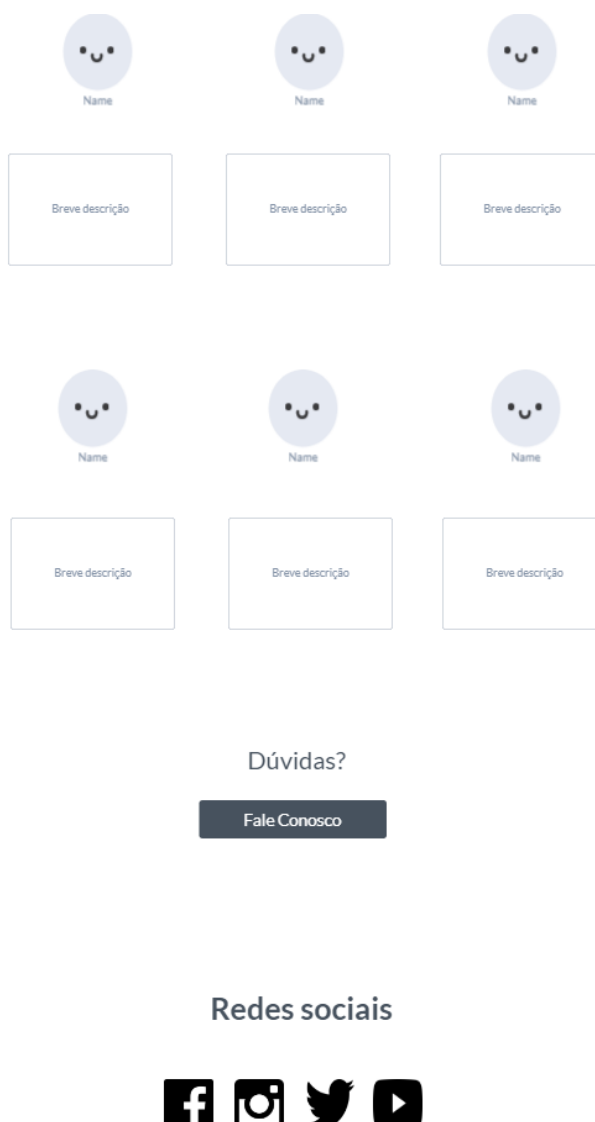


Em consequência do nível de complexidade dos elementos presentes na tela “Quem Somos” do site, não foi possível visualizar a tela toda em apenas uma imagem. Por esse motivo, ela será dividida em duas partes, semelhante à tela do home-page.

Novamente retirando o cabeçalho e o rodapé, por serem constantes em todas as telas, é descrito nessa primeira parte o título da página e logo abaixo um elemento de texto que descreve quem ou o que é responsável pelo conteúdo daquele site. Ao lado, vemos um elemento gráfico para exemplificar o tópico da página. Rolando a tela, o usuário irá se deparar com os tópicos visão e valores, ambos com elementos de texto. No tópico visão, o elemento texto mostrará ao visitante do site quais são os objetivos da empresa. Já no tópico valores, o conteúdo do elemento texto será os valores mais relevantes da empresa. Logo abaixo, vemos o tópico “Nosso time”, na qual será mostrado na parte dois dessa página.

A imagem a seguir mostra a parte dois da tela “quem somos”.

Figura 12 - Segunda parte da tela Quem Somos



Na segunda parte da tela, vemos o conteúdo do tópico “Nosso time”. Este é mais um dos tópicos acrescentados no site na intenção de trazer confiança ao usuário. Nele pode-se ver um elemento imagem, e dois elementos textuais. No elemento imagem será adicionado a foto dos

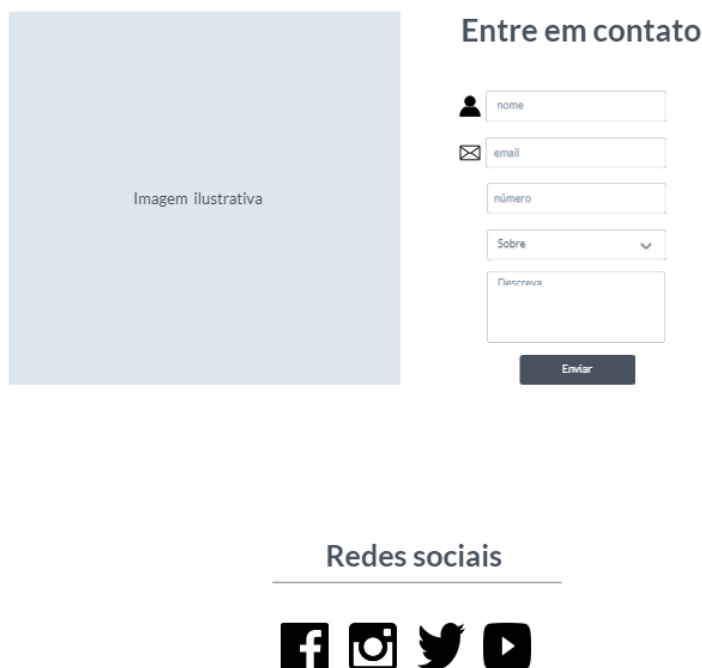
integrantes da empresa. Logo abaixo o nome, e mais abaixo descrições como, o que faz e sua contribuição na confecção do site. Rolando a página vemos novamente o elemento Dúvidas, com a presença de um botão que redireciona o usuário a página “Contatos”. Abaixo disso, o elemento redes sociais, na qual foi descrito anteriormente.

4.2.5. Tela – Contatos

A partir dessa página, o usuário poderá fazer contato com a equipe do site para tirar dúvidas. Ademais, essa página pode ser acessada por duas maneiras, a primeira é pelo cabeçalho, e a segunda é pelo botão “Fale Conosco”.

A seguir a página demonstrada pela figura 13, ilustra a página contatos.

Figura 13 - Contatos



Entre em contato

nome

email

número

Sobre ▾

Mensagem

Enviar

Redes sociais

f i t y

Podemos observar que, além do título da página, a tela ainda apresenta ao lado uma imagem que ilustra o tópico em questão. A página ainda apresenta quatro inputs, que são, nome, e-mail, número e comentário, além de um dropdown para facilitar na classificação do contato. O botão enviar foi colocado para concluir a tarefa de contatar ao grupo. Mais abaixo, apresentasse as redes sociais da empresa.

4.2.6 Tela – Login

A próxima tela é extremamente importante para o projeto. Ela é responsável em identificar o usuário, guardando suas informações nas aplicações para otimizar seu uso, intensificando o requisito de usabilidade do site.

Figura 14 - Tela do Login

A imagem mostra a interface de login de um sistema web. No topo, há uma barra de navegação com links: Home, Matérias, Serviços, Quem Somos, Contatos e um botão Login. Abaixo, o título "LOGIN" é exibido em letras maiúsculas. O formulário de login contém: ícones para login rápido via Facebook e Google+; campos de entrada para "email" e "senha" com ícones de envelope e cadeado, respectivamente; um link "esqueceu a senha?"; e um botão "Logar".

Pode-se perceber que a tela de login, além do título da página que norteia o usuário, abaixo tem as opções de login rápido, que são pelo facebook e pelo google. Ao serem clicados, o usuário é redirecionado para uma página externa para fazerem login por esses meios. Abaixo estão os inputs de e-mail e senha que são grandes para auxiliar na acessibilidade. Mais abaixo o botão apresentam-se os botões, “esqueceu a senha”, que redireciona o usuário para uma página de recuperação de senha, e o botão “logar” que confirma a ação.

4.2.7 Tela – Esqueceu a Senha

Essa tela tem como o objetivo recuperar a senha do usuário e só pode ser acessada clicando no botão “esqueceu a senha” no login.

A figura abaixo (Figura 15) mostra um protótipo da tela descrita acima.

Figura 15 - Tela para recuperação da senha



O protótipo da tela de recuperação de senha apresenta uma interface limpa. No topo, há uma barra de navegação com links para Home, Matérias, Serviços, Quem Somos e Contatos, além de botões para Login e Logout. O título principal da tela é "ESQUECEU A SENHA?". Abaixo dele, há um formulário contendo o texto "Guia breve de como prosseguir", um campo de entrada para o e-mail (com um ícone de envelope) e um botão "CONFIRMAR".

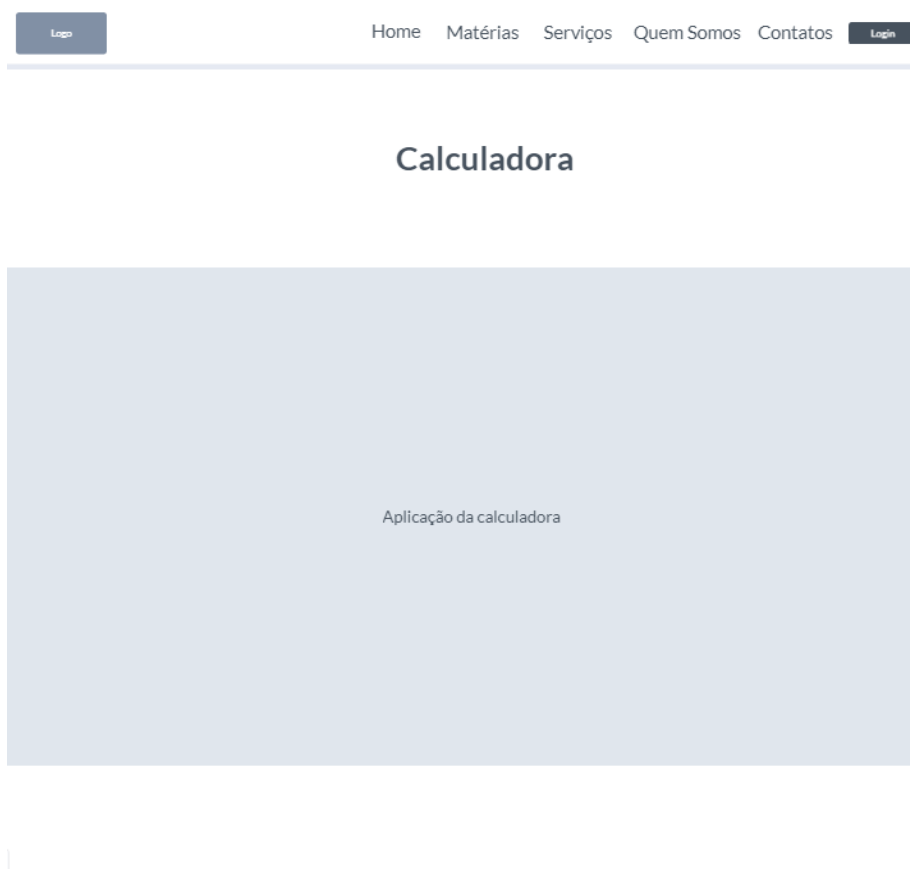
Como pode ser visto acima, a tela apresenta o título da página, e um quadrado, que delimita um elemento texto, um elemento de input e um elemento botão. O elemento texto explica ao usuário como ele deve prosseguir para que ele consiga recuperar sua senha. Isso se deve pois o usuário que se pretende alcançar é leigo na internet.

4.2.8 Tela – Calculadora

É nesta tela do site onde situará a aplicação da calculadora, como sugere o nome. Ela pode ser acessada tanto pelo home-page, quanto pela tela de serviços

A imagem abaixo exemplifica como é a estrutura dessa tela

Figura 16 – Tela de aplicação calculadora

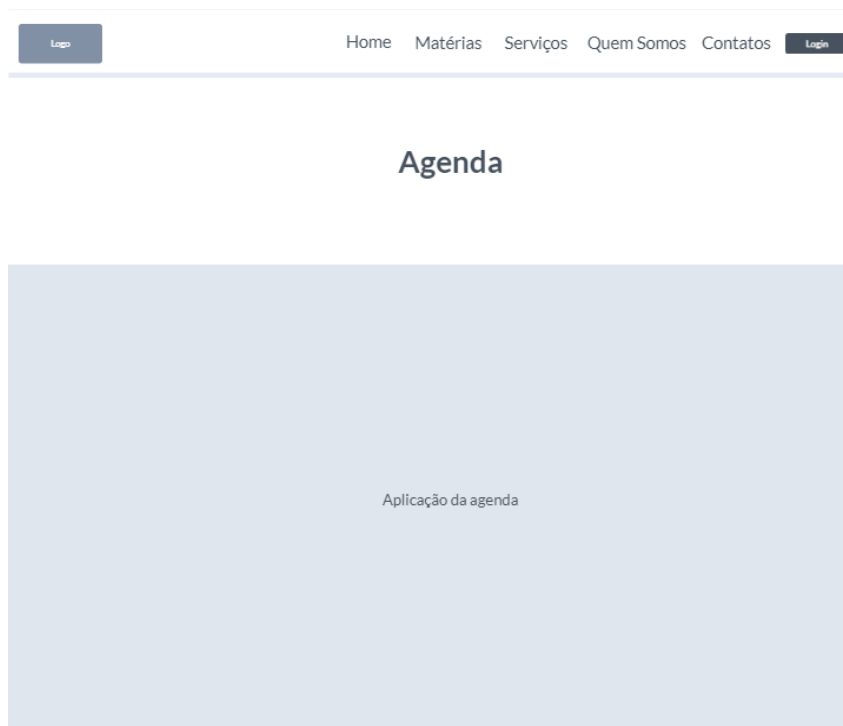


A estrutura de site acima pode ser classificada simplesmente por título da página, e aplicação, ambos tomando grande parte da tela para que possa ser visível a públicos que necessitam de mais acessibilidade na internet.

4.2.9 Tela – Agenda

Semelhante a tela da calculadora, a tela da agenda tem como objetivo acomodar a aplicação da agenda, mas, só pode ser acessada pela página serviços.

A figura abaixo relaciona a tela em questão.

Figura 17 - Tela da Agenda

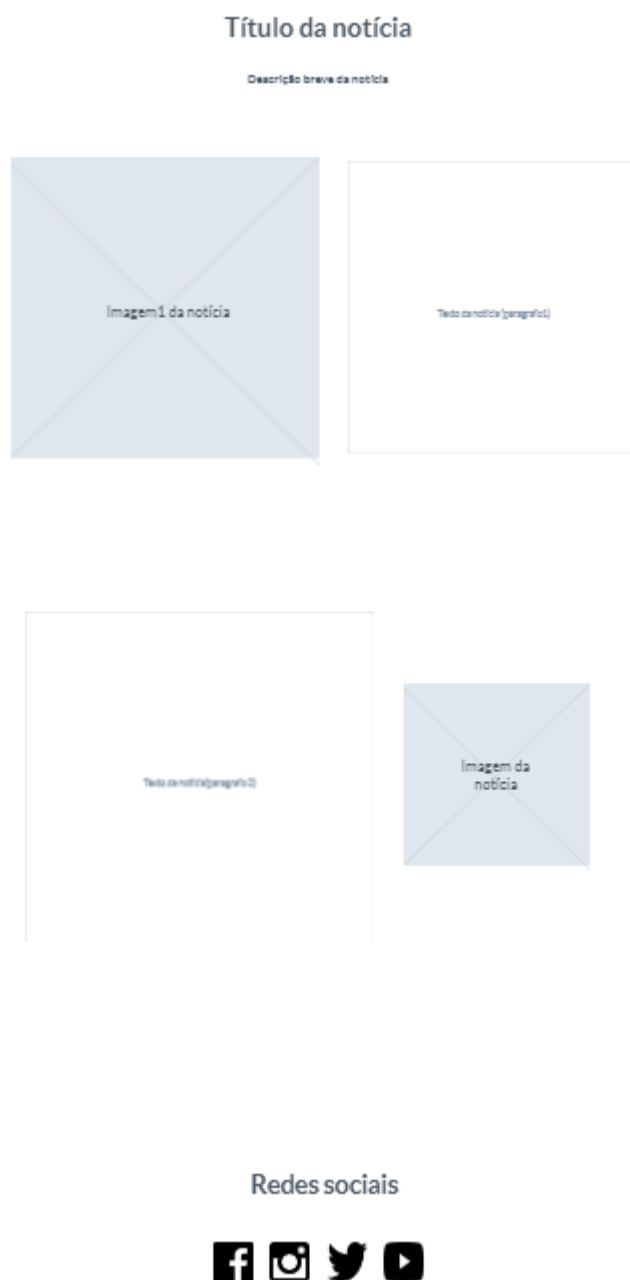
A tela da aplicação da Agenda segue os mesmos itens e os mesmos objetivos aplicados a tela anterior, porém, relacionados a agenda.

4.2.10 Tela – Notícias

Essa tela é uma das únicas que não pode ser acessada pelo cabeçalho, pois ela não é uma tela independente. Ela é a representação estrutural das notícias distribuídas pelo site, e só pode ser acessada ao clicar em saiba mais nas notícias.

A figura abaixo exemplifica essa estrutura.

Figura 18 - Estrutura das notícias do site



A estrutura acima busca mostrar as notícias em um grau de minuciosidade relativamente grande. Por isso, a tela apresenta dois elementos de texto e dois de imagem, para que a notícia muito bem explicada. Além disso, os elementos em questão, apesar da má resolução na representação nesse documento, procuram um tamanho que seja de fácil leitura tanto visual quanto gráfico para o usuário. Ao descer a página, encontra-se novamente as redes sócias do site.

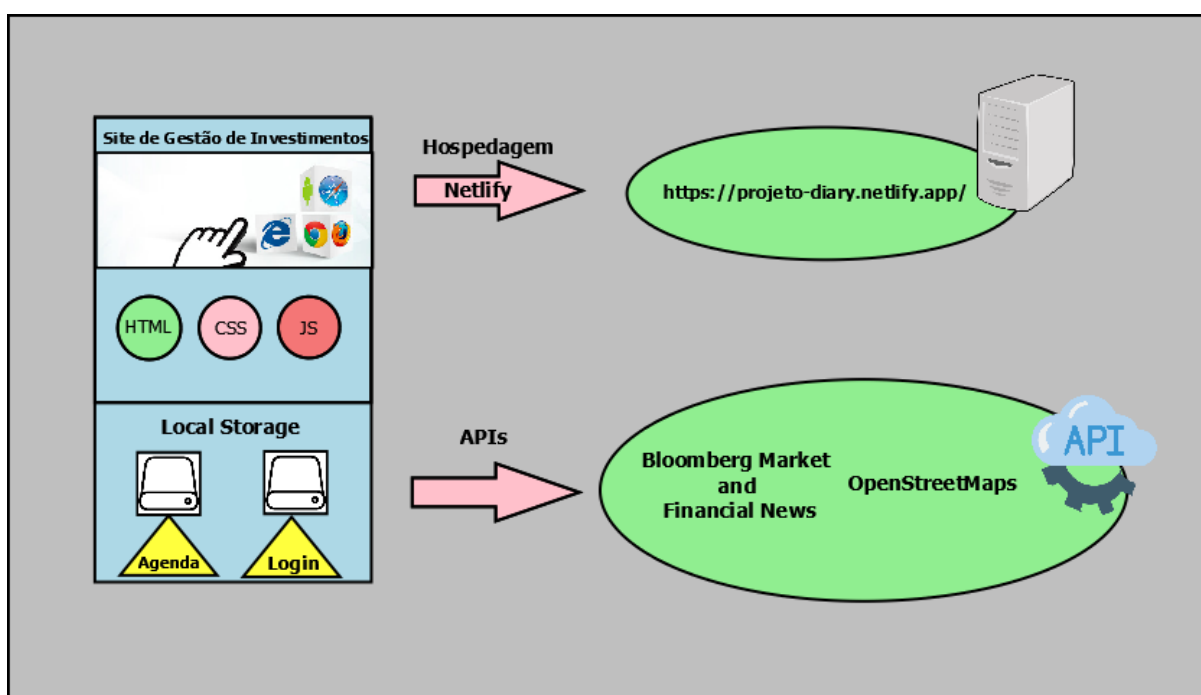
5 IMPLEMENTAÇÃO DA SOLUÇÃO

Nesta seção são apresentados os detalhes técnicos da solução criada pela equipe, tratando da Arquitetura da Solução, Template Padrão do Site e as estruturas de dados e as telas já implementadas.

5.1 Arquitetura de Solução

Diagrama de Componentes - Gestão de Investimentos

Figura 19 - Diagrama de componentes



A solução pensada para o desafio em questão é composta de:

Navegador: Meio pelo qual nossa solução será acessada.

Consiste de:

Website: Conjunto de páginas Web, feitas a partir de arquivos HTML, CSS E JavaScript.

Local Storage/Cookies: O armazenamento será mantido no navegador no projeto final pelo Local Storage, no entanto, nessa entrega inicial será salvo por cookies.

Agenda: Salva os investimentos colocados no site pelo usuário.

Login: Necessário para entrar na agenda na entrega final.

APIs: Aplicativos feitos por terceiros que serão utilizados na solução. São eles:

Bloomberg Market and Financial News: Trará informações para serem utilizadas na calculadora do site.

OpenStreetMaps: Projeto Open-Source para mostrar a localização do escritório da empresa.

Hospedagem: O website usa a plataforma netlify nas releases iniciais, mas na final, será utilizado o serviço Heroku.

Link: <https://projeto-diary.netlify.app/>

5.2 Template padrão do site

Em Desenvolvimento, apresenta todos os layouts base propostos pela equipe, onde vamos trabalhar em cima desse modelo.

5.3 Layouts

5.3.1 Layout Navbar e Carousel

Esse é um modelo de navbar e carousel que optamos por utilizar na nossa aplicação, ela apresenta um design moderno e com destaque aos botões. Todas as imagens são fictícias.

Figura 20 - Layout



5.3.2 Layout acesso a calculadora (home)

Apresenta uma área exclusiva de acesso a calculadora, que será a nossa principal aplicação no site. Todas as imagens são fictícias.

Figura 21 - Acesso a Calculadora



5.3.3 Layout Dúvidas Frequentes

Como o público em sua maioria será leigo, a equipe decidiu fazer uma área de destaque para responder algumas das suas possíveis dúvidas.

Figura 22 - Layout Dúvidas Frequentes



5.3.4 Layout Benefícios

Os principais benefícios da aplicação a serem oferecidos ao público serão publicados nessa área, abordados em pequenas frases para ser de fácil entendimento.

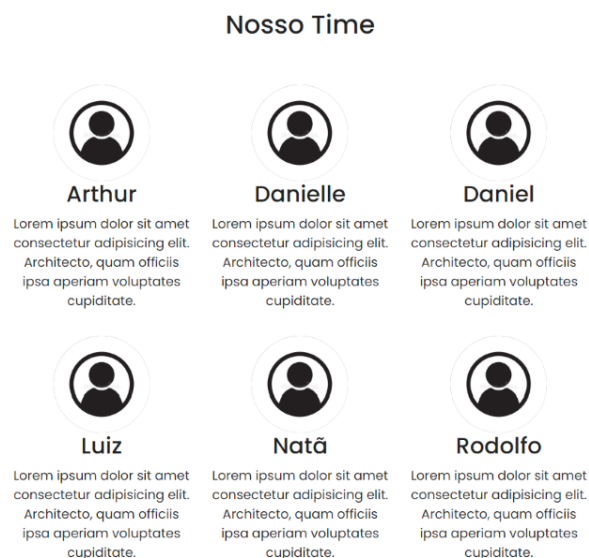
Figura 23 - Ofertas para clientes



5.3.5 Layout Quem Somos

Dentre as preocupações expostas pelo público-alvo, destaca-se a insegurança com relação a tecnologia, e com base nisso optamos por criar uma página exclusiva para falar um pouco sobre nossos interesses e objetivos com essa aplicação, a fim de levar mais confiança ao público. Vale destacar que cada um dos membros da equipe vai ter uma pequena descrição para se familiarizar com os usuários.

Figura 24 – Layout Quem Somos



5.3.6 Layout Calculadora (Serviços)

A imagem abaixo representa o local no qual as aplicações da calculadora e da agenda vão ficar disponíveis para o público acessar.

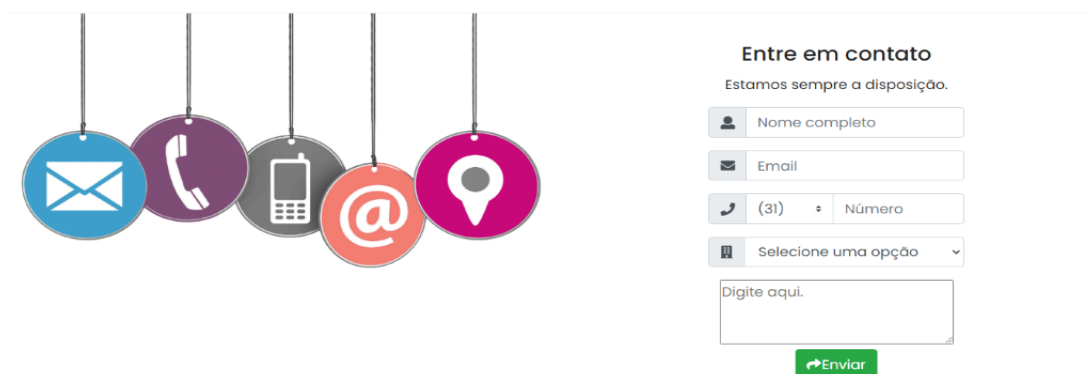
Figura 25 - Calculadora de investimentos



5.3.7 Layout Contatos:

Com o mesmo intuito do “quem somos”, essa área foi criada com o objetivo de confortar o usuário, levar mais segurança sabendo que ele pode encontrar em contato conosco e tirar qualquer dúvida ou problema rapidamente.

Figura 26 - Layout Contatos



5.3.8 Layout Notícias

Conforme foi citado nos requisitos funcionais, nossa aplicação pretende levar informação ao usuário, por isso torna-se essencial uma área na qual ela vai poder ler algumas matérias, artigos e conteúdos relacionados a gestão de investimentos.

Figura 27 – Layout Notícias

Renda fixa

Lorem ipsum dolor, sit amet consectetur adipisicing elit. At sunt recusandae aliquid, molestiae facilis dignissimos?

Lorem ipsum dolor sit amet consectetur adipisicing elit. Amet ratione id possimus veritatis illum aperiam nihil expedita, eaque dolor minima magni fugiat officia facere beatae, illo numquam voluptas! In odio obcaecati, aperiam exercitationem iusto porro corporis nulla vero. Doloribus sunt dicta tempora voluptates obcaecati quisquam atque hic est, numquam tenetur perspiciatis reiciendis optio iste, eius ducimus repellat a, iusto sit commodi nihil! Soluta ratione sint, facilis incidunt praesentium natus. Mollitia, fugit rem voluptates voluptatum quaerat incidunt eius exercitationem aut harum!



Lorem ipsum dolor sit amet consectetur adipisicing elit. Quod repudiandae accusamus at maxime praesentium voluptates harum natus, veniam ipsa! Iure similique aspernatur, debitis sunt cum quaerat harum? Soluta dolores nihil nulla dolore aliquid rerum molestiae, veritatis, cum nobis ut, amet obcaecati libero quaerat illum ab quas dignissimos ullam commodi autem?



Lorem ipsum dolor, sit amet consectetur adipisicing elit. Molestiae deserunt, doloremque nemo tempore aliquam optio, esse tempora vero sint laboriosam velit neque, eligendi quas atque reiciendis. Nam ipsam eveniet doloremque voluptas autem quia quisquam reiciendis, eligendi quibusdam ipsa praesentium minima rerum consectetur, eum, laudantium impedit commodi. Ullam dicta ab iste possimus fugit repudiandae amet placeat laborum nam totam, deserunt animi eveniet, quae maiores corporis vitae sint assumenda, officiis temporibus? Labore aut repudiandae explicabo animi neque in commodi, obcaecati dolorum.

5.3.9 Layout Footer

O footer apresenta um design simples, com alguns dos temas principais abordados no site, uma pequena área para e-mail, logo do site e o endereço fictício.

Figura 28 - Layout Footer

(31) 9 9999-9999
email@hotmail.com
Av. Nephelus 784, Jardim Miragem, Contagem
CEP: 19567-456

Principais Temas

- Renda fixa
- Ações
- Calculadora
- CDI

Assine e Receba Notícias

© Developer by Diary Team

5.4 Orientações sobre a responsividade

A responsividade foi feita por meio do sistema grid do Bootstrap e da media queries do CSS3, focando principalmente nos dispositivos Desktop, mas também atendendo perfeitamente aos dispositivos Mobile. Sobre o Grid System do Bootstrap, utilizou-se principalmente o sistema de classes, pois é totalmente responsivo e construído com flexbox. Em relação ao CSS, a parte de responsividade foi dividida nos tamanhos 576px, 768px e 992px, para ser adaptável tanto em celulares quanto tablets.

Alguns destaques são para a área do template “Como nós cuidamos de você”, no qual na versão Desktop ele divide os itens em 3 por linha, e na versão Mobile é dividido em 1 item por linha. Vale ressaltar a parte das "Redes Sociais", que sofre uma adaptação tanto em tamanho como em organização dependendo do tamanho da tela do dispositivo. Outras áreas onde a responsividade foi aplicada são:

- **Navbar :** Encolhe para o formato "Menu Hambúrguer" na versão mobile.
- **Carousel:** Os botões e o tamanho da fonte sofrem adaptações de acordo com o tamanho da tela do dispositivo.
- **Footer:** Retira a parte do email quando em dispositivos mobile.
- **Quem somos:** Todos os itens da tela se adaptam em tamanho e organização de acordo com o tamanho do dispositivo.
- **Notícias Principais:** Sofre adaptações de tamanho e organização de acordo com a tela do dispositivo.

5.5 Organização do código-fonte

Conforme a imagem abaixo, o código foi separado em partes de acordo com o conteúdo, apresentando um comentário antes de cada sessão para ser de fácil compreensão. Apresenta boas práticas de programação tanto na parte de HTML quanto em CSS.

- Link onde o código-fonte pode ser encontrado:
- <https://github.com/ICEI-PUC-Minas-PMGCC-TI/tiaw-pmg-cc-t-20202-gestao-de-investimentos>

Figura 29 - Código fonte

```

87         <a class="carousel-control-next" href="#slides" role="button" data-slide="next">
88             <span class="carousel-control-next-icon" aria-hidden="true"></span>
89             <span class="sr-only">Next</span>
90         </a>
91     </div>
92 </div>
93 </section>
94 <!--Titulo Noticias-->
95 <section class="titulo-noticias">
96     <div class="container-fluid padding">
97         <div class="row welcome text-center">
98             <div class="col-12">
99                 <h1 class="display-4">Noticias Principais</h1>
100             </div>
101             <hr>
102         </div>
103     </div>
104 </section>
105 <!--Noticias-->
106 <section class="noticias-principais">
107     <div class="container-fluid padding">
108         <div class="row">
109             <div class="col-12 pb-5">
110                 <section class="row">
111                     <div class="col-12 col-md-6 pb-0 pb-md-3 pt-2 pr-md-1">
112                         <div id="featured" class="carousel slide carousel" data-ride="carousel">
113                             <ol class="carousel-indicators top-indicator">
114                                 <li data-target="#featured" data-slide-to="0" class="active"></li>

```

5.6 Funcionalidades do Sistema

O objetivo primário seria a criação de uma agenda que registras a data, a quantidade, e o valor para deixar o usuário mais satisfeito, e conseguir registrar diversas coisas como os seus custos, e datas.

Primeiramente, criamos uma array cada campo será usado com o objeto `document.querySelector`, no caso, uma API do DOM que nos permite buscar o elemento para usar um elemento um seletor CSS ou id no elemento. Além disso, usa se o `forEach`, para iterar diversos dados como a data, quantidade e o valor. Após isso, criaremos um `td` dinamicamente que não conterá nenhuma informação, com o `appendChild`, adicionaremos á `td` como filho. Logo após, cadastrarmos precisa se limpar os campos. Então, depois de preencher os dados em toda `array.value` coloca se ou aspas vazia, 1, e, 0 para melhorar a experiência do usuário.

Figura 30 - Lista de negociação

```
document.querySelector('.form').addEventListener('submit', function(event) {
    event.preventDefault();

    var tr = document.createElement('tr');

    campos.forEach(function(campo) {
        var td = document.createElement('td');
        td.textContent = campo.value;
        tr.appendChild(td);
    });

    var tdVolume = document.createElement('td');
    tdVolume.textContent = campos[1].value * campos[2].value;
    tr.appendChild(tdVolume);

    tbody.appendChild(tr);

    campos[0].value = '';
    campos[1].value = 1;
    campos[2].value = 0;

    campos[0].focus();
});
```

Esse trabalho utilizara o modelo MVC(model-view-controller) com o objetivo de criar um modelo de negociação, logo, utiliza se as classes. Dessa forma, com a class Negociacao ela adicionará um método em Negociacao.js. No caso, com o constructor que vai receber os insumos data, quantidade, valor. Logo após, cria-se o método obterVolume, no caso, seria a multiplicação da quantidade * valor. Além disso, para bons métodos de programação, todas as propriedades que conter _ só pode ser acessada pelos próprios métodos da classe. Diante disso, vou usar o getData, getQuantidade, e o getValor com esse objetivo.

Para proteger o código, para isso, congelaremos o objetivo, portanto, qualquer alteração nas suas propriedades será ignorada. Isso é positivo, porque nem mesmo os métodos da classe que são as funções dentro da classe vão poder alterar as propriedades da negociação.

Figura 31 - Negociação JavaScript

```

class Negociacao {
  constructor(data, quantidade, valor) {
    this._data = new Date(data.getTime());
    this._quantidade = quantidade;
    this._valor = valor;
    Object.freeze(this);
  }

  get volume() {
    return this._quantidade * this._valor;
  }

  get data() {
    return new Date(this._data.getTime());
  }

  get quantidade() {
    return this._quantidade;
  }

  get valor() {
    return this._valor;
  }
}

```

Diante disso, quando usarmos `new Negociacao`, receberemos uma instância congelada. Outro ponto muito importante para a execução dessa parte foi a substituição das variáveis de trocar a `var` pela `let`, que faz com que as variáveis permitam ficar no escopo de bloco e evita que corrompa para o escopo global.

Diante disso, ao terminar a negociação deve-se capturar as ações do usuário e moldar este modelo através de um controller. Primeiramente, deve-se colocar o `event.preventDefault()` para evitar a submissão automática do formulário, e logo, atrapalhar a submissão do projeto.

Dessa forma, usaremos o método `adiciona` para submeter o formulário, e com isso, a instância de `negociacaoController`, chama o método `adiciona(event)`. Além disso, o `document.querySelector()` vai fazer a função de buscar pelo DOM que é a interface da programação para documentos. Dessa forma, ele vai buscar os elementos pelo id `#data`, `#quantidade`, `#valor`. Além disso, substituiremos o `$ = document.querySelector` dessa forma, o código fica menos verboso. Para funcionar essa artimanha do Jquery deve-se usar também a função `bind` que serve como uma ligação para o documento.

Figura 32 - Negociação Controller

```
class NegociacaoController {  
  constructor() {  
    let $ = document.querySelector.bind(document);  
    this._inputData = $('#data');  
    this._inputQuantidade = $('#quantidade');  
    this._inputValor = $('#valor');  
    this._listaNegociacoes = new ListaNegociacoes();  
    this._negociacoesView = new NegociacoesView($('#negociacoesView'));  
    this._negociacoesView.update(this._listaNegociacoes);  
    // this._mensagemView = new MensagemView($('#mensagemView'));  
    // this._mensagemView.update(this.mensagem);  
  }  
}
```

Além disso, deve-se criar a negociação com os dados do formulário, vamos criar o formulário negociação e utilizará três insumos o input.data, input.quantidade, input.value. Dessa forma, dá erro e encontramos nosso primeiro grande problema conseguir transformar a string date em uma forma que seja melhor visível.

O objetivo é transformar em ano, mês, dia, dessa forma cada um desses itens tem que ser parâmetros dessa construção. Um ponto interessante, é que o mês tem que ser decrescido por menos um já que janeiro é igual a 0, logo pela array. Além disso, deve-se usar o Split(-) para retirar os dd-mm-aaaa, e o ... spread operator, com o objetivo de desmembrar essa array. Lembre-se que para o código ficar mais visível, e para obedecer as pedidas do MCV (model-control-view) deve sempre separar os códigos para ficar mais fácil possíveis reparações.

Figura 33 - Date Helper

```
class DateHelper {  
    constructor() {  
        throw new Error('Esta classe não pode ser instanciada');  
    }  
  
    static dataParaTexto(data) {  
        return `${data.getDate()}/${data.getMonth()+1}/${data.getFullYear()}`;  
    }  
  
    static textoParaData(texto) {  
  
        // data.getDate()  
        // + '/' + (data.getMonth() +1)  
        //+ '/' + data.getFullYear();  
  
        if(!/\d{4}-\d{2}-\d{2}/.test(texto))  
            throw new Error('Deve estar no formato aaaa-mm-dd');  
  
        return new Date(...texto.split('-').map((item, indice) => item - indice % 2));  
    }  
}
```

O próximo objetivo é quando clicar no botão incluir, conseguir submeter o formulário, e dessa forma, fazer com que inclua essas arrays de negociação com a tela do navegador. No entanto, primeiramente deve se resolver a questão das datas para ficar dia/mês/ano, logo, criaremos uma variável let diaMesAno e no caso, logo após usar o .getMonth, getFullYear e o getDate + / para separar cada um desses valores. Também lembrar de somar +1 graças o problema do mês.

Além disso, criaremos uma instância para exibir a data, cujos quais: um para converter texto para a data e outro convertendo data para texto. Utilizaremos os métodos ágeis para se for falso jogar erro ou se for verdadeiro esse if vai retornar e utilizar todas aquelas parafernalias ditas acima como o spread operator, o Split para separar as strings além do map para construir um novo array.

Figura 34 - Classe adiciona evento

```
adiciona(event) {  
    event.preventDefault();  
    this._listaNegociacoes.adiciona(this._criaNegociacao());  
    this._negociacoesView.update(this._listaNegociacoes);  
    this._limpaFormulario();  
}  
  
_criaNegociacao() {  
    return new Negociacao(  
        DateHelper.textoParaData(this._inputData.value),  
        this._inputQuantidade.value,  
        this._inputValor.value);  
}
```

Logo após, de resolver a questão das datas. Agora o objetivo de adicionar as demais negociações cadastradas para a lista. Ao trabalhar com array de negociações, utilizaremos o tipo de controle e podemos fazer várias operações com o array, ao utilizar o modelo `model` ele serve para encapsular a regra de uma lista de negociação, dentro da pasta `model`, e para criar o arquivo `ListaNegociacoes.js` assim cria-se a nova classe.

Além disso, observa-se um novo problema porque ao criar os dados precisa – se uma forma de limpar os dados para que essa ação não seja realizada pelo usuário uma forma de aumentar o conforto do usuário. Dessa forma, necessita-se criar um método chamado `limpaFormulário`, que apenas vai zerar todos os campos.

Figura 35 - Apaga e limpa formulário

```

apaga() {
    this._listaNegociacoes.esvazia();
    this._negociacoesView.update(this._listaNegociacoes);

    // this._mensagem.texto = 'Negociações apagadas ';
    // this._mensagemView.update(this._mensagem);
}

_limpaFormulario() {
    this._inputData.value = '';
    this._inputQuantidade.value = 1;
    this._inputValor.value = 0.0;

    this._inputData.focus();
}
}

```

Além disso, para conseguir adicionar os elementos para a lista utilizaremos o método `Adiciona()` que vai ser o caminho para adicionarmos à negociação, ao passarmos usaremos o `this._negociacoes` a importância `this` se dá por ser uma variação implícita logo o `criaNegociacao` vai receber pela `ListaNegociacao` adiciona e logo após, vamos usar a função `limpaFormulario` para apagar os cabecários. Diante disso, se vê a importância é as qualidades de se usar esses métodos que automaticamente após fazerem uma função ele chama outra função, dessa forma, organizando o código, e faz todos os seus objetivos.

Figura 36 – Modelo do Template

```

template(model) {
    return `
    <table class="table table-hover table-bordered">
      <thead>
        <tr>
          <th>DATA</th>
          <th>QUANTIDADE</th>
          <th>VALOR</th>
          <th>VOLUME</th>
        </tr>
      </thead>
      <tbody>
        ${model.negociacoes.map(n => `
          <tr>
            <td>${DateHelper.dataParaTexto(n.data)}</td>
            <td>${n.quantidade}</td>
            <td>${n.valor}</td>
            <td>${n.volume}</td>
          </tr>
        `).join('')}
      </tbody>
      <tfoot>
        <td colspan="3"></td>
        <td>
          ${model.negociacoes.reduce((total, n) => total + n.volume, 0.0)}
        </td>
      </tfoot>
    </table>
    `;
}

```

Logo após, fazer o modelo da negociação, e ele cumprir todas aquelas funções iremos para o próximo passo do modelo MVC, dessa forma, o objetivo é usar a pasta view para encapsular a apresentação que o usuário vai receber. Dessa forma, para exibir a tabela para o usuário necessita criar a função template que vai retornar um template, depois ao jogar esse conteúdo em uma função return.

Além disso, precisa construir uma tabela div para associar com o DOM, dessa forma necessita de usar um construtor que vai receber um elemento. Além do mais, utilizaremos um update deve ser usado para pegar o elemento do DOM e acessar a propriedade INNER HTML, que vai retornar o template.

Essa função INNER HTML, será responsável para converter as strings em elementos visíveis e vão ser inseridas nas divs dessa forma aparecendo para o usuário. Dessa forma, conseguiremos a renderização.

Figura 37 - Agenda

Agenda de Registros

Data

Quantidade

Valor

DATA	QUANTIDADE	VALOR	VOLUME
7/10/2020	2	33	66
15/10/2020	23	123	2829
23/10/2020	11	33	363
8/6/2023	22	3333	73326
			75584

Figura 38 - Console log

```

net::ERR_FILE_NOT_FOUND
console.log(negociacaoController)
VM60:1
NegociacaoController {_inputData:
  input#data.form-control, _inputQuantidade:
  input#quantidade.form-control, _inputValor:
  input#valor.form-control, _listaNegociacoes:
  ListaNegociacoes, _negociacoesView:
  NegociacoesView}
  ► _inputData: input#data.form-control
  ► _inputQuantidade: input#quantidade.form-c
  ► _inputValor: input#valor.form-control
  ▼ _listaNegociacoes: ListaNegociacoes
    ▼ _negociacoes: Array(4)
      ► 0: Negociacao {data: Wed Oct 07 202...
      ▼ 1: Negociacao {data: Thu Oct 15 2020 00:00:00 GM...
        ► _data: Thu Oct 15 2020 00:00:00 GM...
        ► _quantidade: "23"
        ► _valor: "123"
        data: (...)
        quantidade: (...)
        valor: (...)
        volume: (...)
        ► __proto__: Object
      ▼ 2: Negociacao {data: Fri Oct 23 2020 00:00:00 GM...
        ► _data: Fri Oct 23 2020 00:00:00 GM...
        ► _quantidade: "11"
        ► _valor: "33"
        data: (...)
        quantidade: (...)
        valor: (...)
        volume: (...)

```

REFERÊNCIA BIBLIOGRÁFICA

<https://www.ecommercebrasil.com.br/noticias/idosos-tem-renda-superior-a-media-da-populacao-e-aumentam-consumo-de-bens-massivos/>

https://assinaturaglobo.globo.com/o-globo/vitrine/valor-economico-materia-exclusiva?tp=SP&utm_source=valor&utm_medium=barreira&utm_campaign=valor_barreira&url_retorno=https%3A%2F%2Fvalor.globo.com%2Fbrasil%2Fnoticia%2F2020%2F04%2F08%2Frenda-mais-alta-e-acesso-a-servicos-privados-de-saude-favorecem-idoso-brasileiro.ghhtml&interno_origem=sitevalor

<https://idec.org.br/idec-na-imprensa/superendividados-30-milhoes-ja-nao-podem-mais-pagar-suas-dividas>

<https://exame.com/tecnologia/alfabetizacao-digital-segura-avanco-do-acesso-a-internet-no-brasil/>

<https://imasters.com.br/agile/fluxo-de-desenvolvimento-com-gitflow>

<https://blog.pmtech.com.br/scrumban/#:~:text=Scruban%20%C3%A9%20uma%20metodologia%20de,projetos%20de%20desenvolvimento%20e%20manuten%C3%A7%C3%A3o>

<https://blog.trello.com/br/scrumban-metodologia-agil>

<https://www.conventionalcommits.org/pt-br/v1.0.0-beta.4/#resumo>

<https://www.git-scm.com/book/pt-br/v2/Fundamentos-de-Git-Criando-Tags>

<https://medium.com/@luizcarvalho/modelo-de-ger%C3%Aancia-de-branches-de-sucesso-para-git-54955f876c7>

<https://blog.pmtech.com.br/scrumban/>