

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS GUARAPUAVA
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET

HELBERT DA ROCHA

WEGOAL: SISTEMA PARA GERENCIAMENTO E
COMPARTILHAMENTO DE OBJETIVOS FINANCEIROS
INDIVIDUAIS E COLETIVOS

PROJETO DO TRABALHO DE CONCLUSÃO DE CURSO

GUARAPUAVA

2015

HELBERT DA ROCHA

**WEGOAL: SISTEMA PARA GERENCIMENTO E
COMPARTILHAMENTO DE OBJETIVOS FINANCEIROS
INDIVIDUAIS E COLETIVOS**

Projeto de Trabalho de Conclusão de curso de graduação, apresentado a disciplina de Trabalho de Conclusão 1 do Curso Superior de Tecnologia em Sistemas para Internet – TSI – da Universidade Tecnológica Federal do Paraná – UTFPR – Câmpus Guarapuava, como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador: Prof. Dr. Roni Fabio Banaszewski

GUARAPUAVA

2015

RESUMO

Rocha, Helbert da. WeGoal: Sistema para Gerenciamento e Compartilhamento de Objetivos Financeiros Individuais e Coletivos. 59 f. Projeto do Trabalho de Conclusão de Curso - Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuava, 2015.

As pessoas são movidas por objetivos que desejam alcançar, os quais podem ser individuais ou coletivos. Basicamente, um objetivo individual representa o objetivo de apenas uma pessoa e um objetivo coletivo representa um objetivo de um grupo de pessoas. Com a alta da inflação, aumento nos preços dos produtos e aumento do desemprego, está cada vez mais difícil para as pessoas alcançarem seus objetivos financeiros. Neste sentido, surge a necessidade de um sistema para gerenciar objetivos financeiros individuais e coletivos. Este sistema possibilitaria o compartilhamento dos objetivos financeiros de um indivíduo ou grupo, com intuito de que pessoas ou organizações possam ajuda-los a atingir seus objetivos. Também, este sistema permitiria fazer um controle dos gastos e a divisão dos mesmos por um indivíduo ou membros de um grupo. Os usuários poderiam acessar o sistema *web* ou um aplicativo móvel com os dados sempre atualizados. Deste modo, o sistema buscaria auxiliar seus usuários no alcance de seus objetivos financeiros por meio da exposição de alguns métodos econômicos e psicológicos, tais como: motivação subjetiva ou externa, lembretes sobre metas periódicas, controle de gastos, e apoio por meio de redes sociais como, por exemplo, o Facebook.

Palavras-chave: Objetivo financeiro. Meta financeira. Compartilhamento de objetivos financeiros. Divisão de gastos. Controle de objetivos e metas financeiros.

ABSTRACT

Rocha, Helbert da. WeGoal: System Management and Sharing of Financial Goals Individual and Collective. 59 f. Projeto do Trabalho de Conclusão de Curso - Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuava, 2015.

People are driven by the goals that they want to achieve, the goals can be individual or collective. Basically, an individual goal is a goal of only from a person and a collective goal is a shared goal with a group of people. Because of high inflation, increase in product prices and rising unemployment, it is becoming increasingly difficult for people to reach their financial goals. In this regard, the need arises for a system to manage individual and collective financial goals. The system would enable the goal's sharing with external people or organizations that can help an individual or group to achieve their financial goal. Also, this system would make a control of spending and the division thereof by an individual or members of a group. Users could access the web system or a mobile application with the data up to date. Thus, the seek system assist its members in achieving their financial goals through exposure of some economic and psychological methods, such as subjective or external motivation, reminders for periodic goals, cost control, and support through social networks as, for example, Facebook.

Keywords: Financial objective. Financial goals. Financial goal sharing. Division of costs. Control financial objectives and financial goals.

LISTA DE FIGURAS

Figura 1 - Sistema Mobills Financial Education.....	13
Figura 2 - Sistema GoalTracker.....	14
Figura 3 - Sistema GoalsOnTrack.....	16
Figura 4 - Sistema de Financiamento Coletivo Catarse.....	17
Figura 5 - Projeto em Andamento no Catarse.....	17
Figura 6 - Visão do produto - frente e verso.....	38
Figura 7 - Protótipo da tela inicial.....	40
Figura 8 - Protótipo da tela de cadastro de objetivo.....	41
Figura 9 - Protótipo da tela das metas dos objetivos.....	41
Figura 10 - Protótipo da tela compartilhamento de meta ou objetivo.....	41
Figura 11 - Diagrama de classes do sistema.....	45
Figura 12 - Modelagem do banco de dados do sistema.....	46
Figura 13 - Formulário da Pesquisa.....	54
Figura 14 - Gráfico da Questão "Você conhece algum aplicativo ou sistema para controle de objetivos financeiros?".....	55
Figura 15 - Gráfico da Questão "Usaria um aplicativo que te ajudaria a atingir metas e objetivos financeiros?".....	55
Figura 16 - Gráfico da Questão "Compartilharia suas metas e objetivos para receber ajuda de amigos?".....	55
Figura 17 - Gráfico da Questão "Gostaria de receber ofertas e descontos nos produtos que procura como objetivo financeiro?".....	56
Figura 18 - Gráfico da Questão "Gostaria de dividir objetivos e metas financeiros com grupos de amigos, familiares, entre outros grupos de pessoas?".....	56
Figura 19 - Gráfico da Questão "Gostaria de dividir e controlar gastos a fim de ajudá-lo a atingir o objetivo financeiro?".....	56
Figura 20 - Gráfico da Questão "Gostaria de acompanhar a partir de gráficos o andamento de suas metas e objetivos financeiros?".....	56
Figura 21 - Protótipo da tela de <i>login</i>	57
Figura 22 - Protótipo da tela de cadastro.....	57
Figura 23 - Protótipo da tela de cadastro de grupo.....	58
Figura 24 - Protótipo da tela das opções do grupo.....	58
Figura 25 - Protótipo da tela de seleção de pessoas.....	58
Figura 26 - Protótipo da tela das opções do objetivo.....	58
Figura 27 - Protótipo da tela de criação de uma meta.....	59
Figura 28 - Protótipo da tela de seleção da data da meta.....	59
Figura 29 - Protótipo da tela do acompanhamento por meio de gráficos do objetivo.....	59
Figura 30 - Protótipo da tela da linha do tempo dos objetivo.....	59

LISTA DE TABELAS

Tabela 1 - Cronograma de desenvolvimento do trabalho.	30
--	----

LISTA DE SIGLAS

API	Interface de Programação de Aplicativos (do inglês Application Programming Interface)
ASF	Apache Software Foundation
CSS	Folha de estilo em cascata (do inglês Cascading Style Sheets)
GAE	Google App Engine
GWT	GWT Kit de Ferramentas Web (do inglês GWT Web Toolkit)
HTML	Linguagem de Marcação de Hipertexto (do inglês HyperText Markup Language)
JSF	JavaServer Faces
JSP	JavaServer Pages
JSON	Notação de Objeto JavaScript (do inglês JavaScript Object Notation)
MVC	Modelo-Visão-Controlador (do inglês Model-View-Controller)
OOP	Paradigma Orientada a Objetos (do inglês Object-Oriented Programming)
PaaS	Plataforma como Serviço (do inglês Platform as a Service)
SDK	Ferramentas de Desenvolvimento de Software (do inglês Software Development Kit)
SPA	Aplicação de Página Única (do inglês Single Page Application)
SQL	Linguagem de Consulta Estruturada (do inglês Structured Query Language)
TDD	Desenvolvimento Orientado a Testes (do inglês Test Driven Development)
W3C	World Wide Web Consortium

SUMÁRIO

1 INTRODUÇÃO	8
1.1 OBJETIVOS	10
1.1.1 Objetivo Geral	10
1.1.2 Objetivos Específicos	10
1.2 DIFERENCIAL TECNOLÓGICO	11
2 RESENHA LITERÁRIA	12
2.1 ESTADO DA ARTE	12
2.1.1 Mobills Financial Education	12
2.1.2 GoalTracker	13
2.1.3 GoalOnTracker	14
2.1.4 Catarse	16
2.1.5 Comparativo entre as ferramentas	18
2.2 FUNDAMENTAÇÃO TEÓRICA	18
2.2.1 Linguagem de marcação HTML	19
2.2.2 Linguagem de estilização CSS	19
2.2.3 Linguagem de programação JavaScript	20
2.2.4 Framework JavaScript AngularJS	21
2.2.5 Frameworks para desenvolvimento móvel Apache Cordova e PhoneGap	22
2.2.6 SDK para desenvolver aplicações híbridas Ionic	24
2.2.7 Linguagem de programação Java	24
2.2.8 Plataforma de desenvolvimento Google App Engine	25
2.2.9 Metodologia de desenvolvimento ágil Scrum	26
2.3 CONCLUSÃO	27
3 METODOLOGIA	29
4 DESENVOLVIMENTO	31
4.1 ANÁLISE DO SISTEMA	33
4.2 PROJETO DO SISTEMA	38
4.2.1 Projeto do Protótipo das Telas do Sistema	39
4.2.2 Projeto da Arquitetura do Sistema	42
4.2.3 Projeto da Base de Dados	42
4.3 CONCLUSÃO	47
5 CONSIDERAÇÕES FINAIS	48
REFERÊNCIAS	49
APÊNDICE A - QUESTIONÁRIO SOBRE A ACEITABILIDADE DO SISTEMA	53
APÊNDICE B - PROTÓTIPO DAS TELAS DA APLICAÇÃO	57

1 INTRODUÇÃO

As pessoas são movidas por metas a serem atingidas a fim de alcançarem seus objetivos, os quais podem ser de natureza individual ou coletiva. Neste contexto, objetivo consiste na descrição daquilo que alguém pretende alcançar e metas se refere ao objetivo definido em termos quantitativos envolvendo obrigatoriamente um determinado prazo (POLITI, 2015). De fato, a definição e perseguição de objetivos e metas é essencial para o alcance do sucesso em diferentes esferas da vida, tal como na pessoal, financeira e profissional.

Atualmente, está cada vez mais difícil atingir o sucesso econômico e financeiro por conta das dificuldades econômicas causadas pela atual crise que afeta o nosso país. Esta crise gera consequências indesejáveis, tais como a flutuação de preços causada pela alta da inflação. De forma mais precisa, The American Heritage Dictionary (2015) define inflação como: “Um aumento persistente do nível de preços ao consumidor ou um declínio persistente de poder de compra do dinheiro”. Como resultado desta crise, está ocorrendo o aumento dos juros, desemprego e redução dos salários. Segundo o Instituto Brasileiro de Geografia e Estatística (2015), por meio da Pesquisa Nacional por Amostra de Domicílios Contínua, o número de desempregados no Brasil foi estimado em 8,4 milhões de pessoas no segundo trimestre de 2015.

Portanto, mais do que nunca, para atingir um objetivo financeiro é imprescindível poupar recursos, evitando gastos com bens descartáveis ou não emergências. Principalmente nesse momento de dificuldade, mas também em qualquer outro, o importante é apresentar metas financeiras para ficar mais claro o foco do que se pretende. Desse modo, é possível reduzir o tempo para chegar a concretização da meta, minimizando a ocorrência de fatos que atrasem este atingimento.

Deste modo, o foco deste trabalho é o estudo e a materialização de práticas para facilitar o atingimento de metas e objetivos financeiros, as quais envolvem a aquisição de

bens de consumo e serviços. Segundo o dicionário Michaelis (2015), bens de consumo são “produtos industriais destinados diretamente ao consumo” (e.g. computador, carro, casa) e serviços são “execução de trabalho ou desempenho de funções, ordenados ou pagos por outrem” (e.g. ato médico, serviço telefônico, serviço de transporte).

Uma prática utilizada para o atingimento de metas e objetivos financeiros é a divisão e controle dos gastos entre as pessoas de um grupo (e.g. os integrantes de uma família). Dessa maneira, é possível estabelecer como cada pessoa irá ajudar quantitativamente e periodicamente para o alcance dos objetivos financeiros. Segundo Gail (2007, apud Dommert, 2015), a partir de um estudo¹ que envolveu 267 pessoas de diferentes idades, países e carreiras, foi observado que as pessoas que escreveram seus objetivos aumentaram em 42% as chances de alcançá-los. E ainda, aqueles que transformaram os objetivos em ações, assumiram um compromisso com um amigo e que acompanharam os relatórios do progresso semanal, aumentaram as chances de alcançar seus objetivos para 78%. Segundo Meek (2015), definir e esforçar-se para atingir um objetivo, mesmo que o indivíduo não consiga atingi-lo, faz com que ele se sinta feliz.

Também, outra prática que intensifica a motivação de um indivíduo ou grupo para o cumprimento de metas financeiras é a cobrança exercida internamente (e.g. aspirações próprias) ou externamente (e.g. família, amigos, conhecidos, superiores em uma hierarquia empresarial) ao indivíduo. Ademais, este envolvimento externo também pode servir como meio de contribuir para reduzir as dificuldades para o atingimento do objetivo. Ao perseguir um objetivo, o indivíduo adota mentalidades que facilitam o sucesso em cada fase da busca (VANDERDRIFT e AGNEW, 2014). Segundo Barker (2015), motivação é poder, ela provê sucesso mais que a inteligência, habilidade ou o salário.

Deste modo, considerando a grande adoção e dependência dos indivíduos aos dispositivos digitais (e.g. *smartphones* e computadores) e às relações interpessoais mais virtuais ocasionadas pela expansão cultural de redes sociais e outros meios similares de comunicação (e.g. mensagens instantâneas), uma maneira moderna de implantar estas boas práticas seria por meio da materialização digital em forma de uma aplicação *web* e móvel. Este sistema permitiria a realização de anotações e definições de estratégias, tais como o controle de gastos para o atingimento de metas e objetivos financeiros individuais ou

¹ Estudo sobre objetivos: <http://www.dominican.edu/academics/ahss/undergraduate-programs/psych/faculty/fulltime/gailmatthews/researchsummary2.pdf>

coletivos. Ainda, o sistema permitiria notificações recorrentes a fim de incentivar a perseguição de tais objetivos. Além de gerenciar os gastos individuais e coletivos. Ademais, o sistema também permitiria o compartilhamento de metas e objetivos em redes sociais ou similares a fim de externalizar a intenção e sentido que orienta o indivíduo ou grupo naquele momento, tendo como propósito o recebimento de apoio sentimental ou material de indivíduos próximos, desconhecidos ou mesmo de órgãos empresariais para a concretização dos objetivos financeiros almejados.

Com a concepção e disponibilização deste sistema ao público, espera-se que os usuários se beneficiem das práticas e estratégias implementadas para que possam definir, compartilhar e acompanhar o progresso da realização de seus objetivos financeiros de forma mais eficiente e eficaz. Com isto, espera-se contribuir para a disseminação da cultura relacionada ao ato de economizar, de separar responsabilidades econômicas em um grupo que buscam o mesmo objetivo e compartilhar as aspirações de cada qual, com indivíduos ou organizações que possuem recursos para apoio.

1.1 OBJETIVOS

Nesta seção serão apresentados o objetivo geral e objetivos específicos do presente trabalho.

1.1.1 Objetivo Geral

O objetivo geral deste trabalho é desenvolver um sistema, nas versões *web* e móvel, que possibilite que pessoas ou grupos de pessoas possam registrar e compartilhar objetivos financeiros, afim de auxilia-las a atingir os mesmos.

1.1.2 Objetivos Específicos

- Estudar e analisar as ferramentas de objetivos, metas e de controle financeiro individual;
- Criar um sistema para gerenciar objetivos financeiros individuais e coletivos e

disponibilizar a opção de dividir os objetivos em metas;

- Dividir despesas com as pessoas de um determinado grupo;
- Possibilitar várias formas de compartilhamento dos objetivos para que qualquer pessoa ou organização possa ajudar a alcançar o objetivo;
- Apresentar relatórios, gráficos e históricos dos objetivos alcançados e em progresso.

1.2 DIFERENCIAL TECNOLÓGICO

O projeto proposto neste trabalho tem como intenção criar um sistema *web* e uma aplicação móvel que possibilite o gerenciamento e compartilhamento de objetivos financeiros individuais e coletivos. O sistema poderá ser usado por um indivíduo que deseja adquirir algum bem de consumo ou por um grupo de indivíduos que deseja adquirir um bem de consumo ou serviço de forma coletiva.

O sistema possibilitará o controle e a divisão de despesas individualmente e/ou coletivamente para possibilitar o atingimento de metas e objetivos financeiros. Os usuários poderão ou não compartilhar publicamente seus objetivos individuais ou coletivos, sendo que o compartilhamento online pode incentivar o recebimento de ajuda para que o objetivo seja mais facilmente alcançado.

Por ser um sistema *web* e um aplicativo, este pode ser usado em qualquer lugar que possua acesso à Internet. Caso não haja sinal de Internet, os dados serão gravados localmente e sincronizados com o servidor assim que a conexão for reestabelecida.

2 RESENHA LITERÁRIA

Neste capítulo serão apresentados os trabalhos correlatos que constituem o atual estado da arte do problema proposto. Então, será exposta a fundamentação teórica, onde serão abordados os principais conceitos a serem utilizados para realização deste trabalho, assim como as principais tecnologias.

2.1 ESTADO DA ARTE

Nesta seção serão apresentadas referências para que se tenha um melhor entendimento da proposta de trabalho. Como nas pesquisas realizadas não foram encontrados softwares com objetivos semelhantes ao deste projeto, serão referenciados dados que dizem respeito a projetos de software de cunho diferentes, mas com os temas próximo ao deste.

2.1.1 Mobills Financial Education

Mobills Financial Education é um aplicativo para controlar finanças que permite ao usuário registrar suas despesas e seus rendimentos usando seu computador, *tablet* ou *smartphones* (MOBILLS, 2015). O aplicativo é desenvolvido pela empresa Mobills Labs.

O usuário pode interagir com outros usuários dentro da *Mobills Community* e compartilhar experiências e dicas financeiras. Ainda, o usuário pode registrar e categorizar gastos em qualquer lugar escolhendo uma opção referente ao gasto.

O Mobills é gratuito durante um período de avaliação, após este período é necessário assinar a versão *premium* para habilitar funcionalidades adicionais. O Mobillis conta com mais de 1 milhão de pessoas que fazem uso da tecnologia para controlar suas finanças

peçoais e mesmo interagir em sua rede social. Mais precisamente, o usuário pode curtir, compartilhar e convidar amigos a utilizar o aplicativo e assim ganhar MB\$² que podem ser trocados pela versão *premium* do Mobills. Os dados de uma conta podem ser acessados pela *web* ou pelos sistemas operacionais móveis iOS, Android ou Windows Phone. A Figura 1 mostra a tela de cadastro de despesas do sistema Mobills Financial Education em sua versão móvel e *web*.

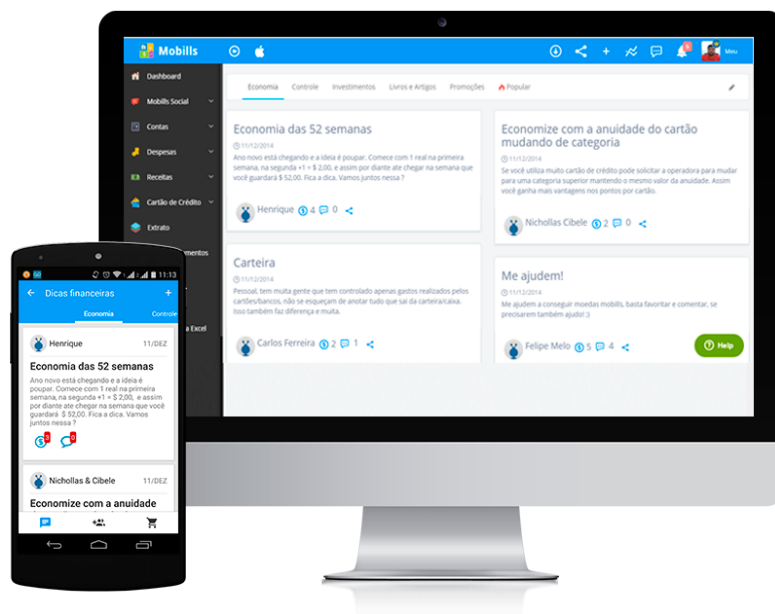


Figura 1 - Sistema Mobills Financial Education.
Fonte: Mobills (2015).

2.1.2 GoalTracker

GoalTracker é um aplicativo simples que ajuda o usuário a alcançar sucesso rastreando os seus objetivos de vida. Desenvolvido pela empresa Mobillion, o aplicativo permite estabelecer metas semanais, mensais ou ainda por um longo período customizado em diversos domínios, tais como: família, crescimento pessoal, finanças, saúde, socialização, hobbies, carreira e outros (MOBILLION, 2015).

Com o GoalTracker é possível estabelecer alarmes para cada meta e o alarme lembrará o usuário quando necessário. O aplicativo possui suporte para colocar uma imagem para cada meta, podendo ser selecionado uma imagem da galeria de imagens ou capturar uma foto usando a câmera. Caso o ajuste dinâmico esteja ativado, o aplicativo irá calcular

² Moeda utilizada no sistema Mobills Financial Education

dinamicamente o progresso esperado a partir de uma unidade de tempo.

A navegação no GoalTracker é simples. O usuário consegue acessar *views* de maneira fácil e mudar de *view* em apenas um clique. Ainda é possível criar e modificar as metas passo a passo, sendo possível criar uma nova meta em cinco passos. Entretanto, é difícil para um usuário começar a utilizar o sistema por não ser muito intuitivo, pois não mostra claramente como realizar o cadastro dos objetivos. Com a funcionalidade *Goal List* é possível acompanhar o progresso das metas diferenciando o estado do progresso por três cores diferentes para indicar a distância entre o progresso esperado e o progresso atual. Por fim, o GoalTracker fornece o progresso das metas por meio de gráficos e a revisão das metas por meio de um calendário. Pode-se acompanhar o menu das funcionalidades do sistema GoalTracker na Figura 2.

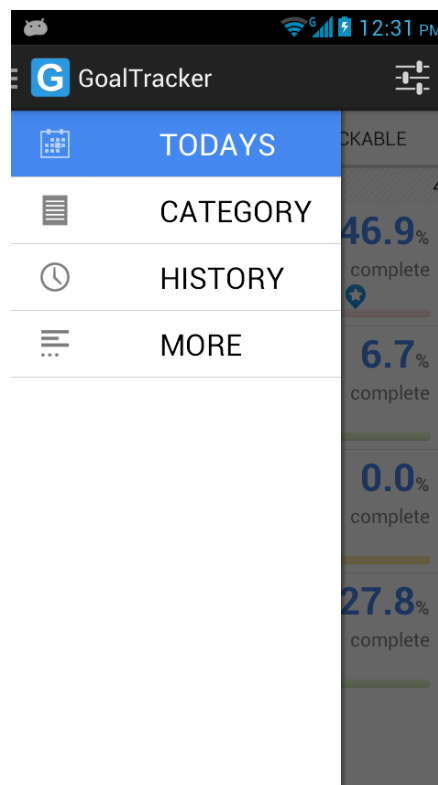


Figura 2 - Sistema GoalTracker.
Fonte: Mobillion (2015).

2.1.3 GoalsOnTrack

GoalsOnTrack é um software que permite ao usuário alcançar seus objetivos fazendo

menos (VANCOUVER IT SERVICES, 2015). O software permite que o usuário crie e especifique metas como específica, mensurável, atingível, realística e oportuna. É possível quebrar metas muito longas, complexas, sobrecarregadas em metas pequenas e em pedaços mensuráveis. Ainda permite acompanhar o progresso em tempo real das metas quando uma tarefa é completada.

O software permite que os usuários criem seus próprios *templates* de acordo com suas metas. Além disso, o software permite definir as tarefas por meta e também organizar as metas por categorias e dias. O software oferece suporte a ação de interface gráfica *drag-and-drop* (arrastar e largar) para priorizar as metas diárias. Há ainda a possibilidade de envio de e-mails para lembrar o usuário sobre o estado de cada meta.

Por meio do software, também é possível visualizar as metas em um calendário, no qual as metas podem ser realocadas facilmente pela ação *drag-and-drop*. Ainda, o sistema permite que o usuário imprima uma folha de papel com suas metas diárias para levar a qualquer lugar. Quando uma meta é cumprida, é possível fazer o *upload* de imagens para aquela meta e colocar em um mural de conquistas.

Segundo Vancouver It Services (2015), o software possibilita criar um modelo de bons hábitos por meio da ferramenta *Habits Builder Tool*. Com essa ferramenta, é possível acompanhar os hábitos por meio de um calendário. Há a opção de criar pequenos parágrafos por meio da ferramenta *Goal Journal Tool*. Essa ferramenta permite ao usuário acompanhar todos os passos das suas metas e aprender com suas falhas. GoalsOnTrack permite ao usuário compartilhar suas metas entre usuários ou com o mundo, uma vez que o usuário pode definir as suas metas como privadas ou públicas.

O software permite ao usuário “Ver suas metas e tarefas em sua ferramenta de calendário favorita, como Outlook, Google Calendar, Yahoo Calendar etc. O usuário pode também importar tarefas do Outlook e exportar dentro de arquivos CSV³” (VANCOUVER IT SERVICES, 2015). É possível criar relatórios no formato PDF de metas e tarefas, acompanhar seu progresso e visualizar gráficos 3D online das mesmas. Os sistemas operacionais móveis suportados pelo software são Iphone, Android, BlackBerry e Windows Phone 7.

Apesar de fornecer muitas ferramentas, o GoalsOnTrack não possui uma interface gráfica intuitiva, amigável e completa para o usuário. Como exemplo, não é possível fazer

³ Comma-Separated Values, é um tipo de formato regulamentado pelo RFC 4180 usado para trocar dados entre aplicações diferentes.

cadastro a partir do aplicativo móvel. Assim, apenas é possível fazer uso do aplicativo após fazer o cadastro no site da empresa. Por fim, não há versão de grátis do sistema e a licença custa \$68 anuais.

A Figura 3 mostra a tela inicial do sistema GoalsOnTracker na sua versão *web*, versão para *tablet* e versão móvel.

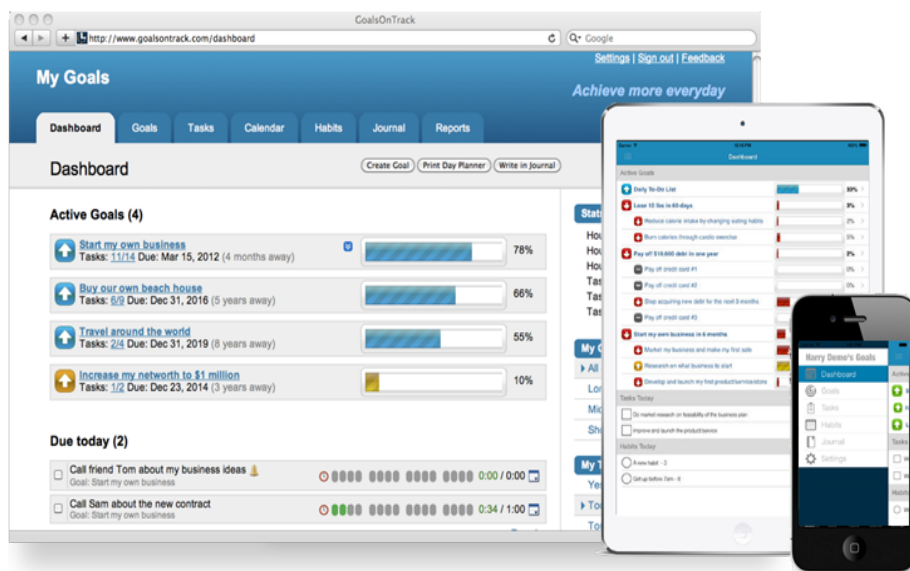


Figura 3 - Sistema GoalsOnTracker.
Fonte: Vancouver It Services (2015).

2.1.4 Catarse

Catarse é um site que fornece o serviço de *crowdfunding*. *Crowdfunding* ou financiamento coletivo em português corresponde a um ambiente em que pessoas contribuem com pequenas quantias a fim de viabilizarem projetos (CATARSE, 2015).

O modelo adotado é o “tudo ou nada”, que tem como objetivo bater a meta no prazo ou devolver o dinheiro caso a meta não seja cumprida. Os realizadores são as pessoas que querem fazer um projeto e eles têm 60 dias para alcançar a meta financeira necessária para que o projeto aconteça.

O Catarse cobra uma taxa de 13% sobre o valor total de projetos que atingiram ou ultrapassaram a meta de arrecadação (CATARSE, 2015). O site fica com 9% e o restante é passado ao site Pagar.me, que é o meio de pagamento utilizado pelo Catarse. Catarse é um site independente, sem investidores e com código aberto (CATARSE, 2015). **A Error! Reference source not found.** mostra a tela de seleção de projetos no sistema de financiamento coletivo Catarse. A divisão dos projetos é feita pela categoria que cada um pertence.



Figura 4 - Sistema de Financiamento Coletivo Catarse.
Fonte: Catarse (2015).

A Figura 5 mostra a página de um projeto em andamento no Catarse.



Figura 5 - Projeto em Andamento no Catarse.
Fonte: Catarse (2015).

2.1.5 Comparativo entre as ferramentas

Os aplicativos, o software e site descritos são de áreas específicas e não apresentam relação entre si. O trabalho proposto visa unir as funcionalidades dos projetos acima apresentados e ainda adicionar outras funcionalidades para que os usuários possam criar e compartilhar objetivos financeiros individuais ou coletivos.

A Quadro 1 apresenta uma comparação entre as ferramentas citadas e o sistema a ser desenvolvido. Na primeira coluna estão os sistemas, sendo o último item da tabela o sistema a ser desenvolvido. Nas demais colunas estão os sistemas suportados e as funcionalidades implementadas.

Sistema	WEB	Android	iOS	Windows Phone	Objetivos individuais	Objetivos coletivos	Compartilhar objetivos	Dividir gastos	Receber ajuda externa	Foco em bens de consumo
Mobills	X	X	X		X					
GoalTracker		X			X					
GoalsOnTrack	X	X	X	X	X					
Catarse	X				X	X			X	
WeGoal	X	X	X	X	X	X	X	X	X	X

Quadro 1 - Comparação entre os sistemas.
Fonte: Autor.

2.2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção serão apresentadas as tecnologias utilizadas no desenvolvimento do sistema que constitui o projeto. Tem-se como intenção realizar um estudo sobre as linguagens de programação, linguagem de marcação e banco de dados a fim de acumular maior conhecimento teórico sobre as mesmas. Este estudo também contribui para obter uma visão mais concreta do projeto e conhecer como ele será colocado em prática.

2.2.1 Linguagem de marcação HTML

Segundo Prescott (2015), a Linguagem de Marcação para Hipertexto (HTML) pode ser usado para criar páginas *web* independentes do tipo de conteúdo. Esse conteúdo pode ser um conjunto de imagens, gráficos, texto, áudio ou vídeo. O navegador é um programa destinado a visualizar documentos desenvolvidos com linguagem de marcação, ou seja, páginas para a *web* (SILVA, 2008). Atualmente, os navegadores mais usados são o Google Chrome, Mozilla Firefox, Internet Explorer, Safari e Opera.

Por ser uma linguagem de marcação de fácil aprendizado, que inclusive pode ser escrita em um editor simples de texto, a linguagem HTML tornou-se um padrão para o desenvolvimento de *websites* na Internet (QUIERELLI, 2012). O HTML é escrito de forma declarativa por meio de comandos chamados *tags*, tais como as *tags* de navegação. As *tags* de navegação permitem que linhas de texto ou similares apontem para informações na Internet (PRESCOTT, 2015). Esses apontamentos formam interligações entre os documentos *web* que possibilitam a um internauta acessar as páginas na Internet através de tais interligações.

O HTML já evoluiu por sete vezes e atualmente se encontra em sua quinta versão, popularmente conhecida como HTML 5. Segundo Prescott (2015, p. 7), “O HTML 5 permite criar aplicações *web* que podem interagir com os dados locais e com servidores mais facilmente do que nunca”. A principal diferença do HTML 5 em relação às suas versões anteriores está na desnecessidade de APIs e *plug-ins* proprietários, evitando com isso prováveis problemas de incompatibilidade. Também, o carregamento das páginas fica mais fácil e leve.

Por conta de sua essencialidade para a *web*, a linguagem HTML será usada neste projeto para criar as páginas que apresentarão informações para os usuários. Essas páginas irão renderizar formulários, imagens, tabelas, gráficos, entre outros elementos que serão utilizados na interação dos usuários com o sistema.

2.2.2 Linguagem de estilização CSS

A melhor e mais precisa definição sobre CSS (Cascading Style Sheet) ou em português, folha de estilo em cascata, pode ser encontrada no site da World Wide Consortium

(W3C)⁴. Nesta definição, CSS consiste em um mecanismo simples para adicionar estilos (e.g., fontes, cores, espaçamento) aos documentos *web* (W3C, 2015).

Segundo Silva (2012), a linguagem CSS possui a finalidade de devolver ao HTML/XML o propósito inicial da linguagem, ou seja, a marcação e estruturação de conteúdo. Portanto, não cabe ao HTML fornecer informações ao agente do usuário⁵ sobre a apresentação dos elementos (e.g. cores de fonte, tamanho de textos). Segundo Silva (2012, p. 25), “[...] o aspecto visual de um documento não deve ser responsabilidade da HTML”. Logo, cabe a linguagem CSS todas as definições da estilização para a apresentação do documento.

A primeira proposta de implementação das CSS aconteceu em 1994. Atualmente, o CSS se encontra em sua terceira versão (CSS3). Nesta versão, o CSS é dividido em módulos, diferentemente das versões anteriores que formavam um único documento. Segundo Silva (2012), cada modulo do CSS3 é desenvolvido de forma independente e segue seu cronograma próprio. Assim, os fabricantes podem começar a implementar funcionalidades de módulos em fases de desenvolvimento mais adiantadas. Este desmembramento contribui para que a maioria dos navegadores *web* já apresentem suporte às propriedades do CSS3.

2.2.3 Linguagem de programação JavaScript

Como a linguagem de marcação HTML não possui funcionalidades para adicionar interatividades avançada às páginas, fica a cargo das linguagens de programação promover a interatividade. Deste modo, a linguagem “JavaScript foi criada pela Netscape em parceria com a Sun Microsystems com a finalidade de fornecer um meio de adicionar interatividade a uma página web” (SILVA, 2010, p. 23). Segundo Silva (2010), atualmente o padrão oficial da linguagem JavaScript é ECMAScript e sua versão é a ECMA-262 v5.

Segundo Crockford (2008), o JavaScript é uma linguagem importante porque é a linguagem dos navegadores *web*. Ela também é uma das linguagens de programação mais populares no mundo. A linguagem JavaScript foi criada para executar no lado do cliente, ou seja, é utilizado um interpretador JavaScript presente no navegador do usuário para fazer a interpretação e promover o funcionamento da linguagem. Em tese, é necessário apenas um

⁴ W3C: <http://www.w3.org/>

⁵ O termo agente de usuário é empregado para se fazer referência a qualquer dispositivo capaz de interpretar um documento escrito em linguagem de marcação.

navegador para fazer funcionar os scripts desenvolvidos com a linguagem JavaScript (SILVA, 2010).

Com a linguagem JavaScript é possível definir, alterar e controlar de forma dinâmica a apresentação de um documento HTML, tal como alterar a cor de fundo, de textos e *links*, alterar o posicionamento dos elementos HTML no documento, manipular as folhas de estilo associada ao documento HTML e criar novas regras CSS, alterar ou anula-las. Ainda, é possível manipular o comportamento do navegador, como criar janelas *pop-up*, apresentar mensagens ao usuário, alterar as dimensões do navegador, entre outros comportamentos. Também é possível realizar validações instantâneas em formulários, realizar cálculos e fornecer dicas para o preenchimento dos mesmos. Todas estas operações realizadas pelo JavaScript executam dentro de um evento de *loop* no navegador, o que permite ver alterações nos documentos instantaneamente.

Uma outra característica importante da linguagem JavaScript é o suporte ao paradigma de orientação a objetos, em inglês, Object-Oriented Programming (OOP). A linguagem simula muitos fundamentos de OOP, mas não está plenamente alinhada com todos os conceitos de orientação a objetos. Também, possui uma notação literal poderosa, que permite a criação de objetos, apenas listando seus componentes. Foi essa notação que serviu de inspiração para a concepção do formato de dado JSON⁶.

Neste projeto, a linguagem JavaScript será utilizada no desenvolvimento das funcionalidades principais do projeto, tais como: comunicação dos documentos HTML com as APIs dos *frameworks* de desenvolvimento móvel, interatividade dinâmica nas páginas (e.g. validar formulários, exibir mensagens e realizar animações), persistência de dados localmente no dispositivo móvel, criação e atualização de gráficos, entre outras funcionalidades.

2.2.4 Framework JavaScript AngularJS

O AngularJS consiste em um *framework* JavaScript que implementa o padrão arquitetural MVC (Model-View-Controller), o qual separa unidades lógicas e responsabilidades no desenvolvimento de aplicações de grande porte (SESHADRI e GREEN, 2014). Com isso, o *framework* facilita a criação rápida de aplicações que executam adequadamente em qualquer plataforma *desktop* ou móvel (WILLIAMSON, 2015). O

⁶ Notação de Objeto JavaScript (do inglês JavaScript Object Notation): formato leve para troca de dados.

AngularJS é considerado um “super-herói” porque permite que o desenvolvedor foque no núcleo da aplicação e ele cuida do resto. Ainda, no lado do servidor, o *framework* permite ao desenvolvedor aplicar boas práticas de engenharia de software e no lado do cliente, o *framework* aumenta a produtividade dos desenvolvedores.

Em uma definição mais técnica, “Esse *framework* oferece uma estrutura consistente e escalável que facilita desenvolver aplicações complexas e de grande porte como parte de uma equipe” (SESHADRI e GREEN, 2014, p. 20). Além disso, os *templates* no AngularJS são escritos utilizando HTML puro, o que auxilia os profissionais de *design* na tarefa de criar as páginas. Comumente, as aplicações desenvolvidas com AngularJS seguem a tendência do mercado no que se refere a SPA (Single Page Application, ou Aplicações de página única) principalmente por conta do uso intenso de requisições assíncronas possibilitados pela tecnologia Ajax.

O AngularJS é um *framework* de código aberto mantido pela empresa Google e atualmente está em sua versão 1.4.7. Vale salientar que o AngularJS passou rapidamente de um *framework* desconhecido para um dos mais populares e utilizados na atualidade. Devido a esta popularidade, o *framework* possui um rico conjunto de documentação de usuários e um ótimo suporte da comunidade.

Devido a praticidade oferecida, o AngularJS será usado no corrente projeto. Esta escolha se deve pelo fato de que criar e manter uma aplicação utilizando apenas JavaScript é uma tarefa difícil e trabalhosa. Com o AngularJS, o desenvolvedor pode focar apenas no *core* da aplicação, sem se preocupar com sua arquitetura e ainda, criar arquivos de código menores do que usando o JavaScript puro ou mesmo o JQuery⁷ (SESHADRI e GREEN, 2014). Todas estas vantagens oferecidas contribuem para a popularidade deste *framework*.

2.2.5 Frameworks para desenvolvimento móvel Apache Cordova e PhoneGap

O Apache Cordova é um *framework* de código aberto para o desenvolvimento móvel. Este *framework* permite usar tecnologias padrão da *web*, como HTML5, CSS3 e JavaScript para o desenvolvimento multi-plataforma, evitando a necessidade de uso das linguagens nativas de cada plataforma (APACHE CORDOVA, 2015). Isto é possível porque os aplicativos são executados dentro de um invólucro particular de cada plataforma.

⁷ Biblioteca JavaScript client-side desenvolvida para interagir com o HTML.

O Apache Cordova fornece um conjunto de APIs para que os aplicativos possam acessar recursos (e.g. sensores, câmera, rede) de cada dispositivo. Portanto, este *framework* é recomendado para os desenvolvedores que desejam criar aplicativos para mais de uma plataforma, inclusive com distribuição do aplicativo via as respectivas lojas de aplicativos de cada plataforma. Por fim, o *framework* também é recomendado aos desenvolvedores que procuram produzir aplicativos de maneira híbrida, ou seja, utilizando uma *WebView* para acessar componentes nativos por meio de plug-ins. Atualmente, as plataformas suportadas são Android, iOS, Windows Phone, BrackBerry, Ubuntu, Firefox OS, WebOS e FireOS. No corrente projeto, serão utilizadas as APIs fornecidas pelo Apache Cordova, principalmente às que se referem à câmera e rede.

Similar ao Apache Cordova, o PhoneGap é um *framework open source*, sob a licença do Apache Software Foundation (ASF), que permite ao desenvolvedor criar aplicativos móveis utilizando um conjunto de APIs para a plataforma desejada (ADOBE SYSTEMS, 2015). Este *framework* permite a criação de aplicativos móveis para diversas plataformas de maneira rápida, uma vez que faz uso das populares linguagens HTML5, JavaScript e CSS. Mais precisamente, o desenvolvedor cria uma aplicação *web* para dispositivos móveis e faz uso de uma ferramenta especial provida pelo PhoneGap para transformar a aplicação *web* em aplicativos nativos para as plataformas suportadas (WARGO, 2012). As plataformas suportadas atualmente pelo PhoneGap são as mesmas suportadas pelo Apache Cordova. Portanto, o PhoneGap faz uso do mesmo processo de execução utilizado pelo *framework* Apache Cordova.

Desde a criação do PhoneGap em 2011, o uso deste *framework* é estimado em mais de 400.000 desenvolvedores ao redor do planeta, havendo mais de 1 milhão de *downloads*. Atualmente, grandes empresas contribuem com o projeto, incluindo a IBM, RIM, Microsoft e outras. Por conta deste apoio e dos benefícios oferecidos por esta tecnologia, há uma grande comunidade em volta do PhoneGap.

Basicamente, o que difere o *framework* PhoneGap do *framework* Apache Cordova é o seu serviço em nuvem. O PhoneGap Build é um serviço em nuvem que permite ao desenvolvedor compilar seus aplicativos móveis diretamente na nuvem sem a necessidade de SDKs, compiladores e *hardware* na máquina local (ADOBE SYSTEMS, 2015). Entretanto, o conjunto de API's que ambos utilizam é o mesmo, uma vez que o PhoneGap utiliza o mesmo núcleo do Apache Cordova.

2.2.6 SDK para desenvolver aplicações híbridas Ionic

O Ionic é um SDK *open source* para desenvolver aplicativos híbridos utilizando tecnologias *web* como HTML, CSS e JavaScript (DRIFTY, 2015). O Ionic permite desenvolver aplicações com *native-feeling*, ou seja, aplicativos que parecem nativos mais utilizam tecnologias *web* para o seu desenvolvimento.

Em uma definição simplista, aplicativos híbridos são essencialmente pequenos *websites* que executam em um navegador e têm acesso à plataforma nativa. Esses aplicativos têm algumas vantagens sobre aplicativos nativos, como variedade de plataformas suportadas, velocidade de desenvolvimento e acesso a códigos de terceiros. O Ionic tem acesso nativamente às plataformas por meio das APIs do Apache Cordova ou PhoneGap. Da mesma forma que há esta dependência do Ionic com estas APIs, ele também depende do AngularJS para o desenvolvimento de aplicações.

Neste projeto, o SDK Ionic será responsável pela parte gráfica utilizando HTML, CSS e JavaScript. Por sua vez, o *framework* Apache Cordova ou PhoneGap serão utilizados para acessar os periféricos dos dispositivos móveis.

2.2.7 Linguagem de programação Java

O Java é uma linguagem de programação orientada a objetos diferenciada e apresenta várias características que justificam o seu uso em sistemas de diferentes portes. Algumas características desta linguagem são: *multithread*, interpretada, neutra de arquitetura, portátil, robusta e oferece alto desempenho (MENDES, 2009). Basicamente, o Java é uma tecnologia composta da linguagem de programação propriamente dita e de uma plataforma composta de uma API e máquina virtual (Java Virtual Machine - JVM).

Uma das principais características do Java é a portabilidade de aplicações entre sistemas operacionais possibilitado pelo uso de JVMs. Esta vantagem se aplica tanto para aplicações desktop quanto para aplicações *web*. Em particular, a plataforma Java oferece um conjunto completo de tecnologias para o desenvolvimento de sistemas *web*, tais como Servlet, JavaServer Pages (JSP), JavaServer Faces (JSF) e GWT (GWT Web Toolkit) (MENDES, 2009). Por conta desta abrangência, o Java possui uma grande comunidade de desenvolvedores e uma vasta documentação formal e informal para o saneamento de dúvidas.

A tecnologia Java foi criada em 1991 pela Sun Microsystems e atualmente é usada em 97% dos *desktops* corporativos e em 3 bilhões de celulares (ORACLE, 2015). Por conta dos benefícios oferecidos, a linguagem Java será adotada como linguagem do lado servidor por meio da tecnologia Servlet.

2.2.8 Plataforma de desenvolvimento Google App Engine

O Google App Engine (GAE) é considerada uma plataforma como serviço, em inglês Platform as a Service (PaaS), que permite ao desenvolvedor criar e executar aplicações na infraestrutura do Google (GOOGLE, 2015). Utilizando o GAE não há a necessidade de se preocupar com questões como manutenção, segurança ou escalabilidade dos servidores, basta o usuário da plataforma fazer o *upload* do código-fonte da aplicação para imediatamente a aplicação estar disponível na *web*.

Vale salientar que a plataforma GAE foi projetada para hospedar aplicações com muitos usuários simultâneos (SANDERSON, 2009), ou seja, aplicações escaláveis. Deste modo, os recursos para as aplicações hospedadas são alocados de acordo com a quantidade de usuários correntes de tais aplicações. Neste sentido, o dono da aplicação paga somente e exatamente pelos recursos utilizados, tais como o uso do CPU⁸, espaço para armazenamento do código-fonte ou dados, banda e outros recursos ou serviços específicos (e.g., Channel API, envio de emails, chamadas a recursos externos ao servidor) (SANDERSON, 2009). Apesar do GAE cobrar pelos recursos e serviços utilizados, ele é gratuito para o desenvolvimento inicial. O GAE oferece cotas gratuitas dos recursos e serviços, os quais somente passarão a ser cobrados quando tais cotas forem exercidas. Como exemplo, o GAE disponibiliza gratuitamente 1 GB⁹ para armazenamento e tráfego diário (GOOGLE, 2015).

Em relação à persistência de dados, a plataforma GAE oferece duas arquiteturas diferentes para escolha de seus usuários: uma baseada no modelo relacional e outra não-relacional. Certamente, o modelo não-relacional é mais apropriado para aplicações escaláveis e, portanto, é a arquitetura mais indicada para uso com o GAE. Mais precisamente, o banco não-relacional ofertado pela plataforma, o DataStore, armazena os dados em forma de entidades, que basicamente são objetos estruturados em qualquer das linguagens suportadas

⁸ Unidade de Processamento Central (do inglês Central Processing Unit).

⁹ Gigabyte corresponde a 1 bilhão de bytes.

pela plataforma (i.e. Java, PHP, Python e Go). Neste trabalho, a linguagem utilizada será o Java tanto para o tratamento das requisições e respostas no GAE como para estruturar o modelo de dados para persistência no DataStore.

2.2.9 Metodologia de desenvolvimento ágil Scrum

O Scrum é uma metodologia de gestão e controle que tem como intenção cortar a complexidade dos processos de desenvolvimento para que as equipes possam focar na construção de softwares que atendam às necessidades de negócios (SCRUM.ORG, 2015). Em resumo, o Scrum oferece uma estrutura simples para incentivar a colaboração de equipes em projetos de software simples a complexos (SCRUM.ORG, 2015).

O termo Scrum surgiu primeiramente em 1986 por meio de uma citação no artigo de Hirotaka Takeuchi e Ikujiro Nonaka publicado na Harvard Business Review (PHAM e PHAM, 2012). Após 9 anos, em 1995, Jeff Surtherland e Ken Schwaber criaram a metodologia ágil Scrum baseada em suas experiências em diferentes níveis de projetos em que trabalharam.

Uma das principais características do Scrum é o incentivo à formação de equipes enxutas e multifuncionais. Basicamente, uma equipe Scrum é composta pelos papéis de ScrumMaster, Product Owner e a própria equipe de desenvolvimento (PHAM e PHAM, 2012). É importante que a equipe Scrum possua as habilidades necessárias para desenvolver um produto de software, principalmente as habilidades para coletar os requisitos, projetar e codificar o sistema e boas relações interpessoais (PHAM e PHAM, 2012).

Por se tratar de uma metodologia ágil, o Scrum se diferencia em vários fatores das metodologias mais tradicionais de desenvolvimento de software. Como exemplo, os requisitos funcionais são coletados em reuniões que contam com a presença de toda a equipe e o cliente, quando é formado o conjunto de requisitos denominado *backlog* do produto. Na verdade, os requisitos são descritos como histórias dos usuários, ou seja, são os clientes que escrevem as funcionalidades desejadas em um formato especificado e texto curto denotando uma história. O termo histórias do usuário vem do inglês *user stories* (PHAM e PHAM, 2012). Basicamente, uma história deve conter a informação do papel do usuário requisitante, da funcionalidade desejada e da serventia desta funcionalidade. Ademais, a história pode vir acompanhada do grau de importância que o cliente entende para tal funcionalidade. Este

atributo é importante para que a equipe possa ordenar a execução das histórias dentro do processo de desenvolvimento.

Outra característica evidente do Scrum e das próprias metodologias ágeis é a existência de ciclos ou iterações curtas da equipe com os clientes. Em Scrum, um ciclo de desenvolvimento é chamado de Sprint e normalmente consiste em uma ou duas semanas. Neste sentido, as histórias inicialmente levantadas e que fazem parte do *backlog* são analisadas de acordo com os seus graus de importância e velocidade de desenvolvimento da equipe para definir um conjunto de histórias que irá formar o *backlog* da Sprint. Estas histórias podem ser divididas em tarefas e então podem ser alocadas aos respectivos membros da equipe. Como uma outra característica do Scrum é a transparência do processo de desenvolvimento, as histórias e tarefas são colocadas em um quadro de tarefas, o kanban, o qual geralmente ocupa um lugar de destaque no ambiente de desenvolvimento. Após a finalização do ciclo do Sprint, uma parte do sistema implementada e testada é entregue ao cliente para verificação e validação. Nesta reunião de entrega, novas histórias podem ser levantadas pelo cliente com o objetivo de correção, manutenção ou mesmo para novas funcionalidades não observadas na reunião inicial. Novamente, novas histórias serão selecionadas para começar um novo ciclo de Sprint.

De acordo com o descrito e ainda a outros benefícios que levaram a metodologia Scrum à popularidade e importância que se tem na área de desenvolvimento de software, esta será utilizada neste trabalho com o intuito de melhor organizar os requisitos para execução, de haver um feedback rápido sobre as implementações, de evitar burocracias relacionadas a documentação por conta da natureza volátil dos projetos e também para ajudar no acompanhamento do projeto.

2.3 CONCLUSÃO

Com base nos sistemas similares apresentados na Seção 2.1, conclui-se que ainda não foi encontrado um aplicativo que una as funcionalidades de um sistema de gerenciamento de finanças individuais e de grupos de pessoas e que também possibilite o compartilhamento de objetivos e metas financeiras por meios das redes sociais. Como apresentado, os sistemas do estado da arte focam somente em um indivíduo ou então em projetos, como é o caso do Catarse. Por isso, como apresentado na Tabela 1 da Seção 2.1, o sistema proposto neste trabalho tem como intenção unir as práticas de gerenciamento de despesas de um indivíduo ou

grupo de indivíduos com acompanhamento e apoio material ou sentimental de pessoas conhecidas ou desconhecidas e empresas. Ademais, entende-se que a automotivação é primordial para o atingimento de objetivos, e para despertar esta vontade que está dentro de cada indivíduo, a aplicação proposta também faz uso de algumas técnicas motivacionais, tais como o envio de notificações de lembretes sobre as metas, o próprio desmembramento de um objetivo em meta para que o indivíduo perceba pequenas vitórias e a análise do progresso do atingimento do objetivo.

Para cumprir com a promessa de desenvolver esta aplicação que certamente será de grande utilidade às pessoas, foram pesquisadas e estudadas várias tecnologias a fim de elencar as mais apropriadas para o objetivo que se tem. As tecnologias selecionadas para aplicação no projeto são desenvolvidas, utilizadas e atualizadas frequentemente por grandes cooperações tecnológicas, tais como: Google, Microsoft, Apache, entre outras. Também, estas tecnologias possuem uma vasta documentação formal e informal e contam com uma grande comunidade de usuários que contribuem para o melhoramento das tecnologias e também para sanar dúvidas sobre resoluções de problemas ou melhores práticas para utilização de tais tecnologias. Portanto, com o estudo das tecnologias selecionadas, espera-se se aprofundar em suas particularidades no decorrer do projeto a fim continuar aprendendo e principalmente, fazer uso das mesmas da melhor maneira possível no desenvolvimento do projeto.

3 METODOLOGIA

Este Capítulo apresenta as fases de metodologia de trabalho usada para o desenvolvimento deste trabalho. A metodologia é dividida em 6 etapas, sendo que estas estão relacionadas e descritas a seguir:

- **Levantamento dos requisitos do sistema:** primeiramente será levantado alguns requisitos que o sistema deve atender a partir da técnica de *brainstorming* com embasamento científico e por meio de um questionário disponibilizado na Internet. A partir dessas informações, espera-se ter o conhecimento necessário para arquitetar o sistema.
- **Estudo e definição das tecnologias à serem utilizadas:** estudo das tecnologias e ferramentas que serão utilizadas para o desenvolvimento do projeto, tal como a linguagem de programação, uso de *frameworks*, definição do sistema gerenciador de banco de dados.
- **Idealização da arquitetura do sistema e prototipagem do sistema:** a partir dos requisitos levantados por meio de histórias do usuário conforme a metodologia Scrum, o próximo passo consiste em definir a arquitetura do sistema, tanto do lado cliente quanto do lado do servidor. Também, outra atividade importante desta fase consiste em criar o protótipo das telas do sistema, tanto da versão *web* quanto da versão móvel. Por fim, uma atividade com a mesma importância consiste em criar o modelo de dados.
- **Desenvolvimento do sistema e testes do sistema:** nesta fase se dará início ao desenvolvimento e teste do sistema, sendo que os testes ocorrerão concomitantemente ao desenvolvimento por conta do uso da técnica de Desenvolvimento Orientado a Testes (TDD), comum às metodologias ágeis.

4 DESENVOLVIMENTO

Nessa seção serão apresentadas as funcionalidades que demonstram a viabilidade do projeto juntamente com resultados preliminares alcançadas até o presente momento.

Basicamente, o sistema a ser desenvolvido permitirá que uma pessoa ou um grupo de pessoas crie, gerencie e compartilhe objetivos financeiros. Para isso, o sistema deve exigir que o usuário insira sua renda mensal e opcionalmente as suas despesas mensais no momento em que realizará seu cadastro. Por meio da renda, o sistema poderá gerenciar as despesas do usuário e sugerir um valor mensal que este possa economizar para o alcance de seu objetivo.

Para um objetivo coletivo, um usuário deve criar um grupo e adicionar os demais participantes ao grupo. Há a previsão de implementação de três formas para convidar um indivíduo a participar de um grupo: através de um convite realizado por meio do aplicativo ou pelo sistema *web* ou então, por meio de redes sociais. O convite conterá as diretrizes para que o convidado possa fazer uso da aplicação, tal como os apontamentos para realizar o *download* do aplicativo ou então para acessar o *site* do sistema (para fins de cadastro, caso ainda não esteja cadastrado) a fim de concretizar sua inserção ao grupo.

Além de definir os participantes de um grupo, o atributo principal a ser definido é o objetivo coletivo a ser alcançado. Para que um objetivo seja criado, primeiramente é preciso verificar se cada membro do grupo possui a renda mensal cadastrada a fim de permitir ao sistema conhecer o poder aquisitivo de cada qual e realizar as tarefas dependentes de tal informação, tal como a divisão de despesas. Caso um usuário não possua renda mensal cadastrada no sistema, este não poderá ser alocado em um grupo.

Para criar um objetivo individual, o processo é mais simples do que a criação de um objetivo coletivo. Neste caso, basta definir o objetivo a ser alcançado, uma vez que há apenas um indivíduo envolvido.

Conceitualmente, um objetivo é composto por uma ou mais metas, sendo que ambos são definidos manualmente pelo usuário. Como as metas são divididas manualmente pelo

usuário, logo, para dividir um objetivo em metas, o usuário deve informar a descrição de cada meta, o valor que se deve alcançar para concluí-la e a data de conclusão da mesma. Quando o objetivo é individual, o usuário que definiu a meta fica responsável pela mesma, mas quando o objetivo é coletivo, se faz necessário selecionar um usuário que ficará responsável pela concretização desta. Certamente, para que uma meta seja atribuída a um usuário, tal usuário precisa pertencer ao grupo correspondente ao objetivo.

Após o cadastro de objetivos e metas, os usuários envolvidos podem desejar compartilhar tais eventos a fim de publicidade do fim que lhes norteiam ou então com o intuito de receber apoio de qualquer natureza (material ou sentimental) de pessoas próximas, desconhecidas ou organizações. O projeto do aplicativo proposto tem como intenção tornar simples o ato de compartilhamento e principalmente assegurar que tais compartilhamentos atinjam os resultados esperados. Para este fim, o usuário precisaria selecionar o objetivo ou meta a ser compartilhado e escolher o meio de compartilhamento desejado, que em princípio seriam as redes sociais e aplicações de mensagens instantâneas mais populares (e.g., Facebook, Google+, WhatsApp). O compartilhamento pode ser público ou privado (particular a um indivíduo) a fim de melhor atender os diferentes graus de sigilo ou privacidade pretendidos pelos usuários aos seus objetivos cadastrados.

Certamente, o compartilhamento de objetivos e metas pode ajudar bastante um indivíduo ou grupo a atingir seus objetivos, principalmente se for de organizações ou pessoas que possuem o objeto almejado financeiramente. Neste sentido, organizações que fornecem o bem de consumo desejado por muitos dos usuários do aplicativo podem se beneficiar de tais informações para realizar ofertas a fim de aumentar seus faturamentos e também conquistar novos clientes ou fidelizar já existentes. Para tanto, além de tais organizações poderem ser notificadas dos interesses por meio da funcionalidade de compartilhamento de objetivos, elas também poderão realizar o processo inverso ao que comumente ocorre no comércio tradicional, ou seja, elas poderão buscar no sistema proposto por potenciais clientes e consequentemente, vendas.

Neste sentido, como poderá haver grande interesse das organizações em realizar ofertas para os usuários cadastrados, o sistema precisa adotar uma política para garantir um bom mercado para as empresas fornecedoras e também para evitar que usuários sejam demasiadamente assediados por ofertas falsas e por empresas sem credibilidade. Para tanto, o sistema deve selecionar a empresa que irá ofertar o bem de consumo aos usuários interessados. Esta seleção pode ocorrer automaticamente por meio de uma função matemática que considere os atributos mais pertinentes para uma melhor seleção de produto e empresa,

tais como: preço do produto, taxa de frete, satisfação dos clientes em negociações passadas e outros. Com isso, a intenção é permitir que a empresa selecionada possa ofertar o produto por um tempo determinado e após este período, uma nova seleção possa ser realizada a fim de incentivar a rotatividade.

Um fato importante que pode impactar na adesão das empresas ao aplicativo é a quantidade de usuários interessados em um determinado bem de consumo. Portanto, é importante que além da divulgação do sistema para atrair mais usuários, também haja a compatibilidade do mesmo com múltiplas plataformas de dispositivos a fim de incentivar o seu uso. Neste sentido, o sistema proposto apresentará uma versão móvel e uma versão *web*, sendo que ambas as versões terão como base o mesmo código-fonte. Assim, o sistema estará disponível à todas as pessoas interessadas em usá-lo, bastando ter acesso à Internet para tal.

Mesmo que o compartilhamento dos objetivos e ofertas recorrentes de empresas possam aumentar a motivação dos usuários para o atingimento dos objetivos, ainda a aplicação pretende incentivar a automotivação de cada usuário. Para tal, o sistema propõe a aplicação de algumas técnicas psicológicas consideradas eficazes para o atingimento de objetivos. Entre estas, citam-se o envio de notificações periódicas para o usuário lembrar de seus objetivos e a visualização do progresso do usuário em relação aos seus objetivos. Outra técnica utilizada consiste na própria divisão de um objetivo em metas a fim do usuário apresentar várias sensações de vitórias durante a concretização de cada meta pertencente a um objetivo. Também, há a intenção de apresentar os objetivos concluídos pelo usuário ou grupo em forma de linha do tempo a fim de que possa ser visto o que foi conquistado e isto inspire novas conquistas.

As particularidades do sistema proposto poderão ser melhor entendidas nas seções seguintes. Mais precisamente, a Seção 4.1 apresenta os artefatos gerados durante a análise do sistema; a Seção 4.2 apresenta os artefatos gerados durante o projeto do sistema e por fim, a Seção 4.3 apresenta as considerações finais sobre o capítulo.

4.1 ANÁLISE DO SISTEMA

Nesta seção será apresentado toda documentação referente à análise do projeto em questão e a visão do produto. Esta etapa é importante para que haja o claro entendimento das funcionalidades previstas para o sistema.

Para determinar as funcionalidades do sistema foi realizado um *brainstorming* com base em subjeções, alguns artigos científicos sobre o assunto e respostas do questionário submetido a uma centena de usuários, o qual está presente no Apêndice A. Como resultado desta reunião de planejamento do sistema, alguns requisitos funcionais foram levantados no formato de histórias, conforme a metodologia Scrum. As histórias que formam o *backlog* do produto são apresentadas no Quadro 2, formadas pelos seus respectivos números de identificação, descrição, grau de importância e critérios de aceitação.

ID	Histórias	Imp	Crítérios de aceitação
1	Como um usuário, eu gostaria de realizar o cadastro para ter acesso ao sistema.	20	- O usuário não pode solicitar o cadastro se não informar corretamente todos os dados requeridos para o cadastro. - O sistema ao encontrar um usuário já cadastrado com os dados informados (usuário, email), deve pedir para o usuário informar dados diferentes. - O sistema ao encontrar um usuário cadastrado a partir dos dados informados, deve informar que o usuário foi cadastrado com sucesso.
2	Como um usuário, eu devo fazer o <i>login</i> para acessar o sistema.	25	- O usuário não pode solicitar acesso ao sistema se não informar os dados para o <i>login</i> no sistema. - O sistema ao encontrar as informações para o <i>login</i>, deve redirecionar o usuário à página inicial do sistema. - O sistema ao não encontrar as informações corretas para o <i>login</i>, deve informar que o usuário ou senha não foram encontrados.
3	Como um usuário, eu gostaria de criar grupos para adicionar pessoas.	38	- O usuário não pode solicitar a criação de um grupo se não informar o nome e tipo do grupo. - O sistema ao não encontrar um grupo com o mesmo nome, deve informar que o grupo foi cadastrado com sucesso. - O sistema ao encontrar um grupo já cadastrado com o mesmo nome, deve informar que o grupo já foi cadastrado e deve pedir para o usuário informar um novo nome.
4	Como um usuário,	70	O usuário não pode solicitar o registro de um

	eu quero cadastrar um objetivo para poder alcançá-lo.		objetivo se não informar o título do objetivo. • O sistema ao não encontrar um objetivo já cadastrado com o mesmo título, deve informar que o grupo foi cadastrado com sucesso. • O sistema ao encontrar um objetivo com o mesmo título, deve informar que o objetivo já foi registrado e deve pedir para o usuário informar um novo título.
5	Como um usuário, eu gostaria de dividir os objetivos em metas para ter mais controle sobre eles.	65	• O usuário não pode solicitar a criação de uma meta se não informar a qual objetivo aquela meta pertence. • O sistema ao encontrar o objetivo, deve exibir informações do objetivo e cadastrar a meta para o devido objetivo. • O sistema, não encontrando o objetivo, deve informar que o objetivo não foi encontrado e a meta não foi cadastrada.
6	Como um usuário, eu gostaria de compartilhar metas e objetivos para receber ajuda externa.	60	• O usuário não pode solicitar o compartilhamento de objetivos ou metas sem selecionar qual objetivo ou meta deseja compartilhar. • O sistema, realizando com sucesso o compartilhamento, deve informar que o objetivo ou meta foi compartilhado com sucesso. • O sistema, não compartilhando o objetivo ou meta, deve informar que o objetivo ou meta não pode ser compartilhado naquele momento e por que não pode ser compartilhado.
7	Como um usuário, eu gostaria de dividir gastos individualmente para controlar minhas despesas.	50	• O usuário não pode solicitar a divisão de despesas individualmente se não cadastrar a sua renda mensal no momento do cadastro. • O sistema, encontrando a renda do usuário, deve permitir a divisão de suas despesas, e informar a quantia que pode ser utilizada no objetivo. • O sistema, não encontrando a renda do usuário, deve informar que a renda não foi encontrada.

8	Como um grupo de usuários, nós queremos dividir os gastos coletivamente para definir a quantia que cada membro irá fornecer para alcance do objetivo.	45	- Os usuários pertencentes a um grupo não podem solicitar a divisão das despesas coletivamente se cada membro do grupo não estiver com seu salário cadastrado. - O sistema, encontrando as rendas mensais dos integrantes do grupo, deve permitir a divisão das despesas coletivamente dos mesmos, e informar a quantia que cada membro pode fornecer para o alcance do objetivo. - O sistema, não encontrando as rendas mensais dos integrantes do grupo, deve informar que há pessoas no grupo que não informaram a sua renda.
9	Como um usuário, eu quero acessar gráficos dos objetivos para saber a porcentagem alcançada, atrasada e faltante dos objetivos.	40	- O usuário não pode solicitar a porcentagem dos objetivos se não selecionar o objetivo. - O sistema, deve informar um gráfico com as porcentagens que foi alcançada, que está atrasada e que está faltante para completar o objetivo.
10	Como um usuário, eu gostaria de acessar o histórico dos objetivos para ter acesso à informações sobre os objetivos alcançados.	35	- O usuário não pode solicitar o histórico dos objetivos se não alcançou nenhum objetivo. - O sistema, encontrando objetivos alcançados, deve apresentar todos os objetivos alcançados em forma de linha do tempo. - O sistema, não encontrando objetivos alcançados, deve informar que nenhum objetivo foi alcançado até o momento.
11	Como um fornecedor, eu gostaria de realizar o cadastro para ter acesso ao sistema.	55	- O fornecedor não pode solicitar o cadastro se não informar corretamente todos os dados requeridos para o cadastro. - O sistema, encontrando um fornecedor já cadastrado com os dados informados (CPF/CNPJ, email) deve pedir para o fornecedor informar

			dados diferentes. • O sistema, não encontrando um fornecedor cadastrado a partir dos dados informados, deve informar que o fornecedor foi cadastrado com sucesso.
12	Como um fornecedor, preciso fornecer as informações sobre os produtos as pessoas cadastradas no aplicativo.	43	• O fornecedor não pode oferecer produtos e descontos em produtos se não estiver cadastrado no sistema. • O sistema, encontrando pessoas interessadas no produto, deve mostrar ao fornecedor que uma ou mais pessoas estão interessadas em seus produtos. • O sistema, não encontrando pessoas interessadas nos produtos, deve informar que nenhuma pessoa ou grupo está interessado nos produtos no momento.

Quadro 2 - Histórias e critérios de aceitação.
Fonte: Autor.

Por se tratar de uma metodologia ágil, o *backlog* apresentado bem como os atributos de cada história podem ser alterados durante o desenvolvimento do projeto. De acordo com o avanço do processo de desenvolvimento, algumas previsões vão se tornando realidade e com isso consegue-se um maior conhecimento sobre as particularidades do projeto. Por conta disto, algumas histórias poderão ser adicionadas, alteradas ou até mesmo removidas do *backlog* a partir do início das execuções das *Sprints*.

Após a obtenção de uma visão geral sobre as funcionalidades do sistema a partir do levantamento das histórias, se fez necessário sintetizar e especializar a visão que os envolvidos têm sobre produto a ser desenvolvido. A definição de uma visão única do produto serve tanto para a equipe compartilhar a mesma visão quanto para o cliente validar que realmente é esta a imagem que ele tem do produto ou para ele melhor entender o escopo do problema a ser tratado. Com isto, todos pactuam sobre o que será desenvolvido.

Dentre as técnicas de elaboração da visão do produto, a escolhida foi a Técnica da Caixa do Produto. Essa técnica consiste em criar a visão do produto como se o *software* fosse vendido em prateleira, devendo ser criada uma caixa para o produto. O resultado da aplicação da técnica para o projeto corrente é ilustrado na Figura 6, onde a frente apresenta o nome e logotipo do produto acompanhado do que o sistema se destina a fazer e no verso, estão

elencadas as principais funcionalidades do sistema. Como a equipe para o desenvolvimento deste projeto é composta por apenas um membro, esta imagem representa a visão que o autor tem sobre o produto a ser materializado. Em equipes maiores, todas as imagens deveriam ser analisadas para chegar a um consenso e assim definir a visão que melhor representa a de toda a equipe e cliente.

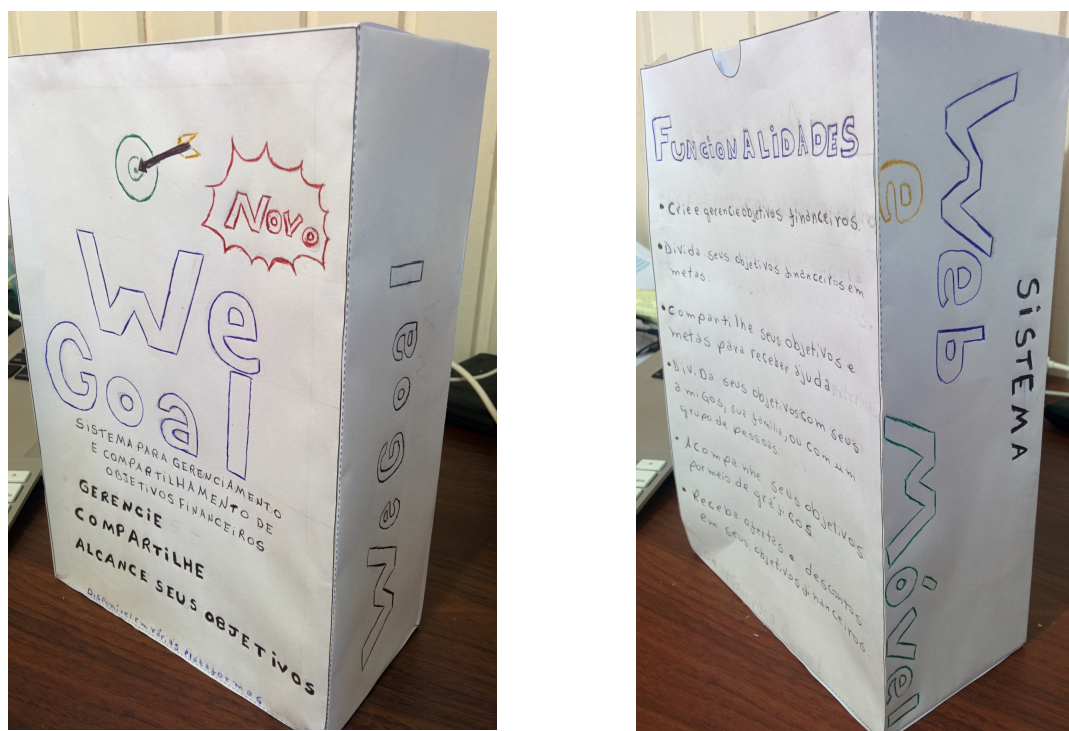


Figura 6 - Visão do produto - frente e verso.
Fonte: Autor.

Por meio do que foi apresentado, as principais histórias foram levantadas e sintetizadas para o melhor entendimento do que precisa ser feito. Este entendimento é suficiente para a criação dos artefatos partes do projeto do sistema, os quais facilitarão o entendimento de como o sistema será feito.

4.2 PROJETO DO SISTEMA

Nesta seção serão apresentados os artefatos gerados na elaboração do projeto do sistema. Mais precisamente, a Seção 4.2.1 apresenta o projeto das principais telas do sistema por meio de protótipos e suas respectivas descrições de funcionamento; a Seção 4.2.2

apresenta a arquitetura do sistema e a Seção 4.2.3 apresenta o projeto dos modelos de dados a serem aplicados no lado cliente e servidor.

4.2.1 Projeto do Protótipo das Telas do Sistema

Os protótipos das telas do sistema foram construídas utilizando a ferramenta *online* gratuita draw.io¹⁰. Esta ferramenta permitiu construir os protótipos de uma maneira que mais se aproxima da realidade da aplicação a ser desenvolvida, oferecendo componentes próximos do *design* do *framework* Material Design. Porém, mesmo havendo esta proximidade, isto não significa que a aplicação a ser desenvolvida respeitará cada detalhe dos protótipos, uma vez que as bibliotecas de componentes a serem utilizadas contém mais recursos do que as ferramentas de prototipação.

Como a tela inicial da aplicação é a mais importante e certamente a mais visualizada pelos usuários, esta precisa ser simples e apresentar uma boa usabilidade. A tela inicial da versão móvel do sistema proposto é ilustrada na Figura 7. Basicamente, a tela inicial mostra os objetivos em forma de grade e separados nas categorias individual e coletivo. Também, para cada objetivo, são mostrados o nome e o percentual concretizado. A tela apresenta mais opções de acesso rápido por meio do botão com o símbolo de “+”, tais como o cadastro de um objetivo ou meta, finalização de um objetivo ou meta, entre outras opções.

¹⁰ Draw.io: <https://www.draw.io/>

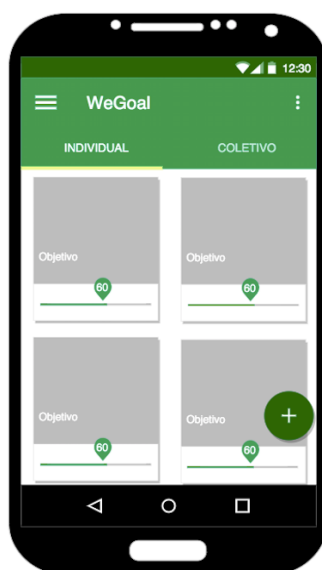


Figura 7 - Protótipo da tela inicial.
Fonte: Autor.

A Figura 8 mostra o protótipo da tela de cadastro de um objetivo. Os campos representados com cores claras representam campos de entrada de dados para um dado tipo de objetivo, individual ou coletivo. Mais precisamente, os campos se referem a imagem representativa do objetivo, título do objetivo e valor monetário correspondente. Para um objetivo coletivo, uma nova janela será aberta pedindo para o usuário selecionar o grupo que aquele objetivo pertence. Como um grupo é composto por um ou mais participantes, ficará a cargo de alguém do grupo definir as metas pertencentes a determinado objetivo. Estas por sua vez devem ser aceitas pelo administrador do grupo. Para que se possa dividir as metas é necessário que todos presentes no grupo estejam com suas rendas mensais cadastradas, caso contrário o membro que não tiver este dado não poderá ter uma meta atribuída.

A Figura 9 mostra o protótipo de tela das metas cadastradas para um objetivo. Basicamente, a meta conterá uma descrição, o tempo restante para ser completada e algumas opções de acesso rápido, tais como: conclusão, compartilhamento, edição e exclusão da meta.

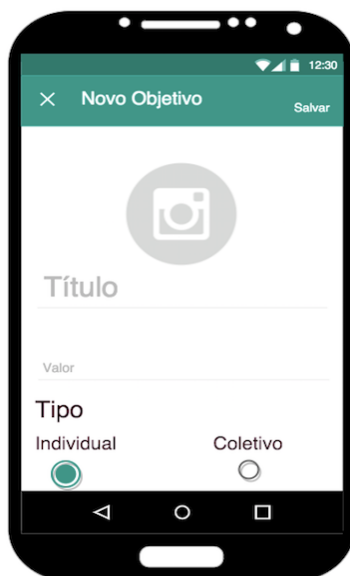


Figura 8 - Protótipo da tela de cadastro de objetivo.

Fonte: Autor.



Figura 9 - Protótipo da tela das metas dos objetivos.

Fonte: Autor.

A Figura 10 mostra o protótipo de tela do compartilhamento das metas e objetivos. Por meio dessa tela, os usuários poderão escolher a rede social ou aplicação de mensagem instantânea nas quais irão compartilhar suas metas ou objetivos.

Por meio destas telas, já é possível vislumbrar o funcionamento e materialização de algumas histórias descritas na Seção 4.1. No entanto, devido à complexidade do sistema a ser desenvolvido, há outras telas que também implementam funcionalidades importantes do sistema, mas que não serão apresentadas no corpo desta Seção. Os protótipos destas telas e suas respectivas descrição são apresentadas no Apêndice B.



Figura 10 - Protótipo da tela compartilhamento de meta ou objetivo.

Fonte: Autor.

4.2.2 Projeto da Arquitetura do Sistema

O modelo de arquitetura Cliente-Servidor será utilizado para o desenvolvimento do projeto. Os dados guardados no aplicativo móvel serão sincronizados com os dados presentes no servidor quando o aplicativo se conectar à Internet. Desse modo, será possível manter os dados locais sincronizados com os dados do servidor.

Nos últimos anos, houve um certo avanço e incentivo para a utilização de boas práticas no desenvolvimento de software, sendo que muitas destas são expressas em padrões de projeto. Um padrão arquitetural muito popular e indispensável para o desenvolvimento de um sistema desacoplado é o MVC. Este padrão é utilizado tanto no lado cliente quanto no lado servidor. Devido a sua importância, o mesmo já se apresenta implementado em forma de *frameworks* a fim de incentivar o seu uso. Como exemplo, há o próprio AngularJS no lado cliente e o JSF no lado servidor. No entanto, mesmo sem o uso de um *framework*, a implementação do padrão é considerada simples. Portanto, a arquitetura de todos o sistema a ser desenvolvido será baseada neste padrão, mesmo que no lado servidor não seja utilizado um *framework* MVC pré-implementado.

Ademais, outros padrões serão usados para compor a arquitetura do sistema. Entre estes, vale salientar o uso do DAO e Service. Basicamente, o DAO teria como responsabilidade abstrair e encapsular os mecanismos de acesso ao banco de dados a fim de esconder os detalhes da execução para a origem dos dados, enquanto que o Service teria como responsabilidade incluir lógica para operar os dados enviados ou recebidos do DAO.

4.2.3 Projeto da Base de Dados

Uma das camadas mais importantes de um sistema certamente é a camada de dados. Como o sistema a ser desenvolvido é praticamente separado em cliente e servidor, ambos os lados apresentam a necessidade de persistência de dados. No lado cliente, o modelo de dados utilizado é o relacional, enquanto que no lado servidor, por questões de eficiência e escalabilidade, o modelo escolhido é o não relacional. Portanto, como a linguagem de programação adotada no lado servidor é Java, o modelo de dados será representado por meio de um diagrama de classes.

Neste sentido, a Figura 11 ilustra as entidades (por meio de um diagrama de classe) que comporão o domínio do sistema. O diagrama tem como classe principal a entidade *Entity*. Esta classe contém os atributos e relacionamentos que serão herdados por meio do princípio da herança pelas classes filhas *User* e *Group*. Um relacionamento herdado pelas classes filhas é com a classe *Expense* que representa a despesa. Desse modo, Os usuários e os grupos de usuários podem conter despesas cadastradas no sistema. A classe *Expense* conta com os atributos descrição da despesa representado pelo campo *description*, valor da despesa representado pelo campo *value*, número de parcelas ou vezes que a despesa está dividida representado pelo campo *times*, datas de início e fim da despesa representados pelos campos *initiatedAt* e *finishedAt* respectivamente. Para saber se a despesa é individual ou coletiva, o campo *type* é consultado. A classe *Entity* também está relacionada com a classe *Transaction* que representa as transações. O usuário ou grupos podem não tem transações no sistema como pode ter várias transações. A classe *Transaction* contém os atributos *date* que representa a data da transação e *balance* que representa o valor da transação. A classe *Entity* está relacionada com a classe *User* por meio do princípio da herança.

A classe *User* por sua vez, contém os atributos referentes aos usuários do sistema, tais como nome, salário, email, senha, uma imagem para o perfil do usuário, o número de telefone do mesmo e o balanço atual. Esses atributos são representados pelos campos *name*, *income*, *email*, *password*, *profileImage*, *phone* e *currentBalance* respectivamente. O campo *currentBalance* que representa o balanço, será consultado para saber o montante que o usuário possui a seu dispor para auxiliar uma meta ou um objetivo. O usuário pode conter ou não um endereço. Este por sua vez é representado pela classe *Address*. O usuário também, pode ou não estar em um grupo representado pela classe *Group*. Por sua vez, um grupo deve ter ao menos um usuário. O grupo tem um título representado pelo campo *title*, uma lista de usuários representados pelo campo *users* e um tipo representado pelo campo *type*.

Em relação aos objetivos, estes são representados pela tabela *Goal*, sendo que um usuário ou grupo de usuários podem ter um ou mais objetivos financeiros. Os atributos contidos na classe *Goal* são *title* que representa o título do objetivo, *value* representa o valor do objetivo, *image* representa uma imagem para o mesmo, *percentageCompleted* esse atributo representa a porcentagem do progresso de cada objetivo e *type* que representa o tipo do objetivo que pode ser compartilhado ou privado. Como acontece com as despesas, para saber se o objetivo é individual ou coletivo, o campo *type* é consultado.

Um objetivo pode não ter ou ter mais de uma meta. As metas são representadas pela tabela *Objective*. As metas são relacionadas a classe *Goal* pelo princípio da composição, uma

vez que não faz sentido existir metas sem um objetivo. Os atributos presentes na classe *Objective* são a descrição da meta, data de início, data de encerramento, a data em que a meta foi completada, a data que a meta foi alterada, a data da criação da meta, o percentual da meta que foi alcançado e o tipo da meta. Esses atributos são representados pelos campos *description*, *beginDate*, *endDate*, *completedAt*, *changedAt*, *createdAt*, *objectivePercentage* e *type* respectivamente. Como uma meta pode ser compartilhada, o campo *type* é usado para saber se a meta é compartilhável ou não. Vale destacar que a palavra *goal* em inglês tem o mesmo significado que a palavra objetivo em português, enquanto que a palavra *objective* em inglês tem o mesmo significado que a palavra meta em português.

Ambos os produtos e serviços são representados pela tabela *Product*. Deste modo, um objetivo pode ser composto por um ou mais produtos ou serviços. Os atributos representados na classe *Product* são o nome do produto representado pelo campo *name*, a descrição do mesmo representado pelo campo *description*, o valor do produto representado pelo campo *value*, a quantidade de produtos cadastrados pelo fornecedor representados pelo campo *quantity* e a data que o produto foi oferecido no sistema representado pelo campo *date*. Todo o produto ou serviço tem um fornecedor ou prestador de serviço, o qual é representado pela entidade *Provider*. Os atributos contidos na classe *Provider* são o nome do provedor, cpf/cnpj, *email*, senha, data de registro, os telefones, a reputação e uma imagem para o logo do provedor. Esses atributos são representados pelos campos *name*, *cpf/cnpj*, *email*, *password*, *registredAt*, *phones*, *reputation* e *logo* respectivamente. Um fornecedor contém um ou mais números de telefone, sendo que cada número será armazenado em uma lista de telefone representados pelo atributo *phone* sendo este uma lista do tipo *String*. De forma semelhante, um fornecedor também possui um ou mais endereços, sendo cada endereço representado pela entidade *Address*.

Basicamente, são estas as entidades que formarão o modelo de dados do sistema do lado servidor. Na verdade, praticamente as mesmas entidades apresentadas comporão a base de dados do lado cliente. Porém, a diferença principal é o modelo de dados utilizado, pois no lado cliente os dados precisam ser persistidos em tabelas, ou seja, precisa ser usado o modelo relacional. Portanto, o diagrama de dados apresentado na Figura 12 representa as tabelas e suas relações que serão usadas para persistência dos dados manipulados por meio da linguagem JavaScript.

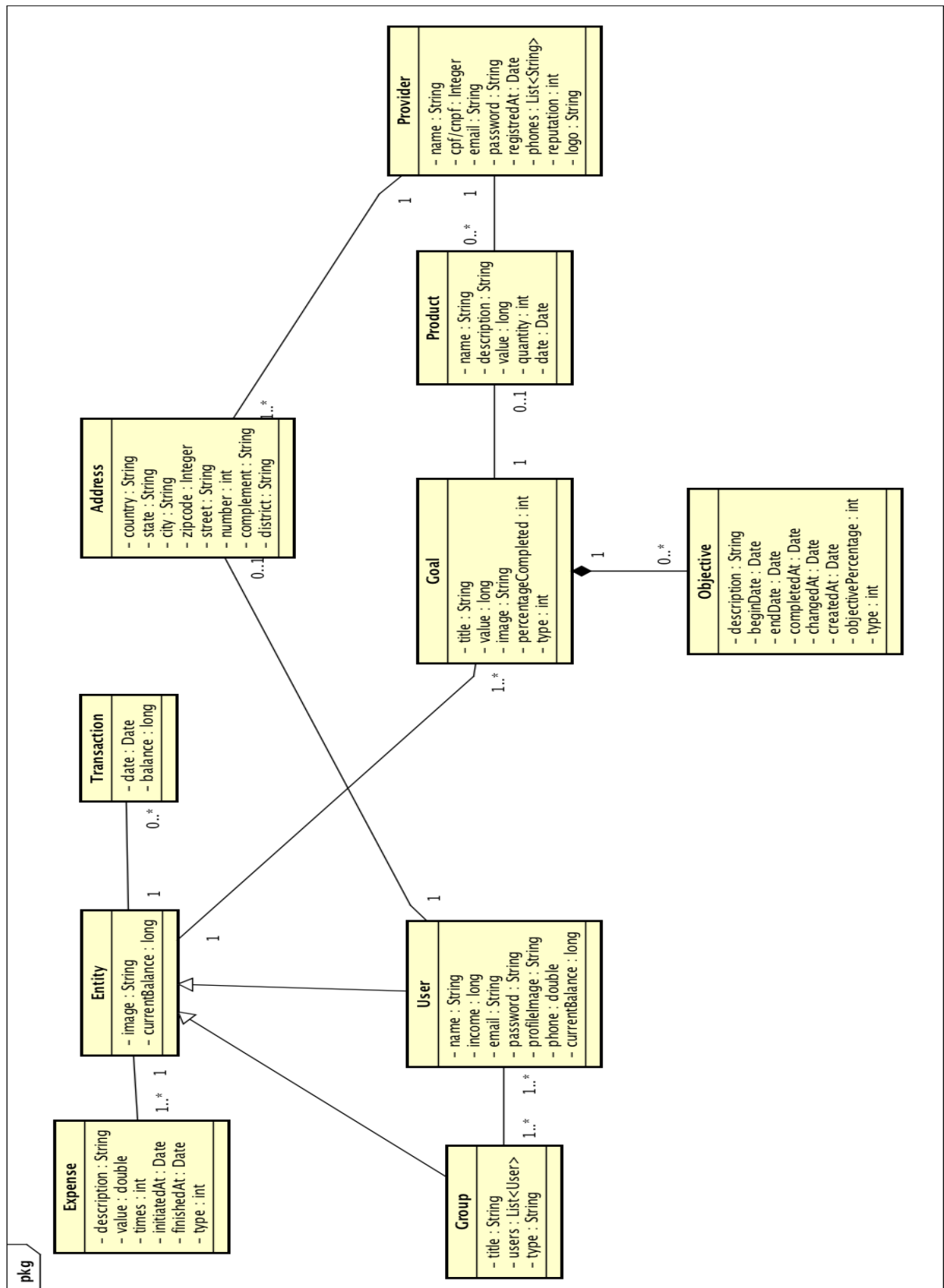


Figura 11 - Diagrama de classes do sistema.

Fonte: Autor.

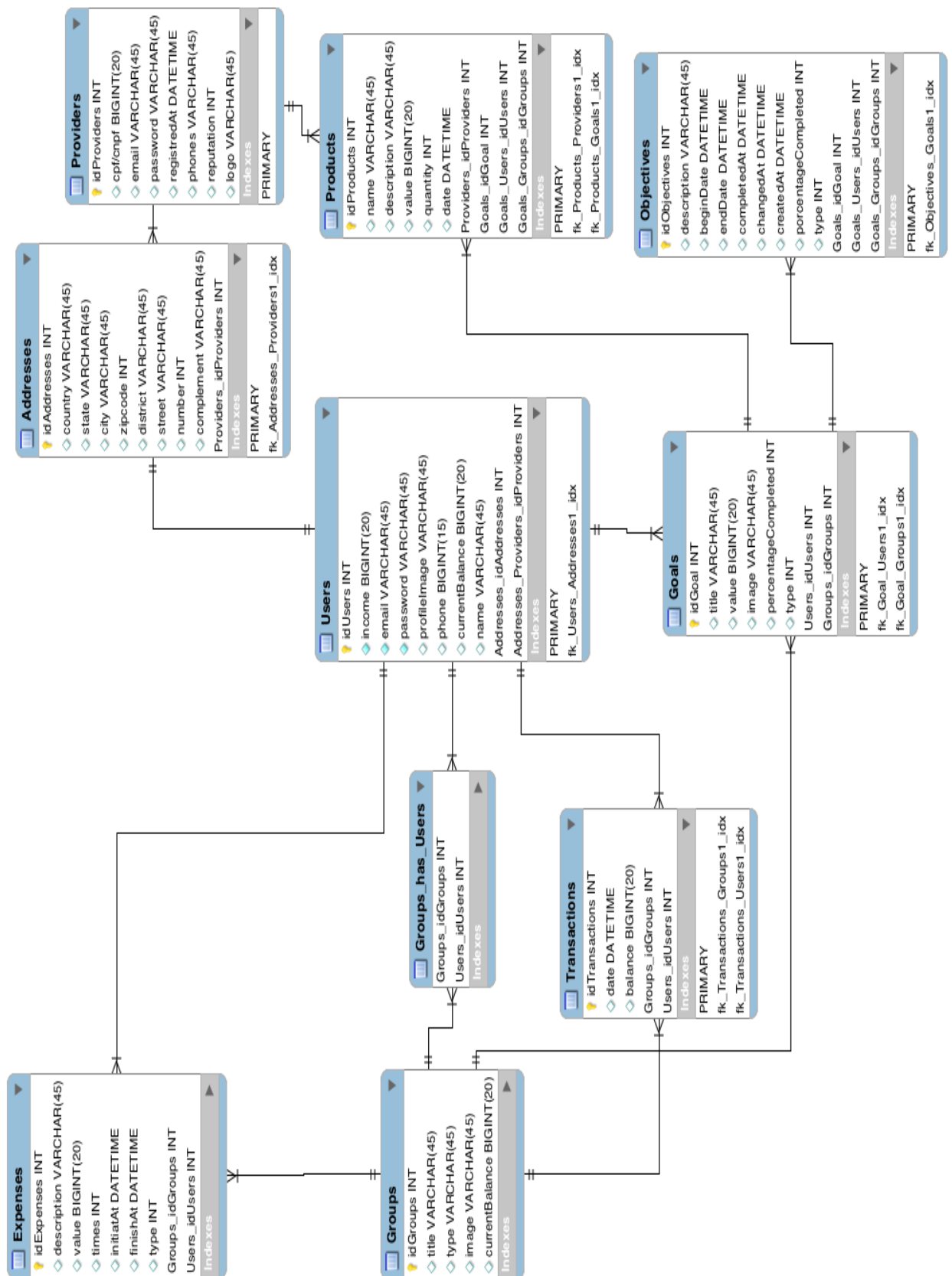


Figura 12 - Modelagem do banco de dados do sistema.

Fonte: Autor.

4.3 CONCLUSÃO

Foram coletados os requisitos do sistema em formas de histórias. As histórias juntamente com seus critérios de aceitação e número de importância podem sofrer alterações durante o desenvolvimento do projeto, pela inclusão e exclusão de novas histórias. Como em um projeto que utiliza a metodologia ágil Scrum todos os desenvolvedores podem partilhar da mesma visão do produto, logo foi definida a visão do sistema por meio da técnica da caixa do produto. Como este projeto está sendo desenvolvido por uma única pessoa as técnicas da metodologia Scrum estão sendo utilizadas para auxiliar no desenvolvimento do mesmo, e espera-se que pessoas externas ao projeto possam partilhar do mesmo entendimento do projeto.

Na fase de projeto do sistema, foram elaborados os protótipos das principais telas do sistema. Os protótipos das telas trazem uma ideia visual do que pretende ser desenvolvido, mas as telas do sistema serão alteradas durante o desenvolvimento do projeto. Foi elaborado o diagrama de classes. Esse diagrama representa o banco de dados utilizado no servidor Google App Engine enquanto que o modelado do banco de dados representa o banco de dados que será utilizado localmente em cada dispositivo móvel. Também, foram definidas as tecnologias e *frameworks* que serão utilizados. Com todos os dados da fase de análise, projeto e definidas as tecnologias resta começar o desenvolvimento do sistema. O desenvolvimento irá complementar o sistema na versão *web* e móvel, também os testes unitários e de aceitação.

5 CONSIDERAÇÕES FINAIS

O objetivo desse trabalho é desenvolver um sistema *web* e móvel que permita a um usuário ou a um grupo de usuário gerenciar objetivos financeiros individualmente ou coletivamente. Nota-se que é difícil para algumas pessoas conquistar objetivos financeiros sem utilizar alguma técnica, como o controle dos gastos, apoio de amigos, desconhecido ou mesmo de organizações. Logo, esse projeto visa criar um ambiente digital onde as pessoas possam alcançar seus objetivos financeiros que consistem na aquisição de produtos ou serviços.

O motivo que levou ao desenvolvimento dessa ideia foi a falta de um sistema que atuasse no gerenciamento financeiro de um grupo de pessoas. Pelas experiências vivenciadas pelo autor desse trabalho, percebeu-se a necessidade de dividir despesas entre os integrantes de uma família ou qualquer grupo de pessoas que compartilhem o mesmo objetivo. Assim, surgiu a ideia de criar um sistema que possa não apenas ajudar o autor do trabalho, mas também as pessoas que vivem situação similar ou queiram adquirir um bem material ou serviço.

Com a pesquisa presente no Apêndice A, verificou-se que a maioria das pessoas não conhecem um sistema para controle de objetivos financeiros. Também, verificou-se que a maioria dos questionados utilizariam um sistema para controle financeiro, principalmente para controlar suas despesas a fim de atingir um objetivo financeiro. Certamente, este questionário reforça a importância e utilidade do trabalho a ser desenvolvido e aumenta a expectativa do autor em ver este aplicativo em funcionamento a fim de ajudar muitas pessoas a atingir o que almejam.

REFERÊNCIAS

ADOBE SYSTEMS. **PhoneGap**. Disponível em: <<http://phonegap.com/>>. Acesso em: 19 out. 2015.

AMERICAN HERITAGE. **Dictionary of the English Language**. Houghton Mifflin Harcourt, 2015. Disponível em: <<https://www.ahdictionary.com/>>.

APACHE CORDOVA. Página Inicial. Disponível em: <<http://cordova.apache.org/docs/en/5.1.1/guide/overview/>>. Acesso em: 18 out. 2015.

BARKER, Erick. **How to Motivate People: 4 Steps Backed by Science**. Disponível em: <<http://time.com/53748/how-to-motivate-people-4-steps-backed-by-science/>>. Acesso em: 31 ago. 2015.

CATARSE. **Página Inicial**. Disponível em: <<https://www.catarse.me>>. Acesso em: 03 set. 2015.

CROCKFORD, D. **JavaScript: The Good Parts: The Good Parts**. Sebastopol, CA: O'Reilly Media, Inc, 2008. 172 p. ISBN 0596554877, 9780596554873 Disponível em: <https://books.google.com.br/books?id=PXa2bby0oQ0C&dq=javascript&hl=pt-BR&source=gbs_navlinks_s>.

DOMMERT, Travis. **Written goals...fact and fiction**. Disponível em: <<http://www.irunurun.com/written-goals/>>. Acesso em: 07 set. 2015.

DRIFTY. Página Inicial. **Ionic Framework**, 19 out. 2015. Disponível em: <<http://ionicframework.com/>>. Acesso em: 19 out. 2015.

GOOGLE. App Engine. **Google Cloud Platform**. Disponível em: <<https://cloud.google.com/appengine/docs>>. Acesso em: 22 out. 2015.

IBGE. **Pesquisa Nacional Por Amostra De Domicílios Contínua**. Disponível em: <<http://saladeimprensa.ibge.gov.br/noticias?view=noticia&id=1&idnoticia=2968&busca=1&t=pnad-continua-mostra-desocupacao-8-3-2%C2%BA-tri-2015>>. Acesso em: 06 set. 2015.

MEEK, Will. **How to Set Goals**. Disponível em: <<https://www.psychologytoday.com/blog/notes-self/201308/how-set-goals>>. Acesso em: 01 set. 2015.

MENDES, D. R. **Programação Java com Ênfase em Orientação a Objetos**. São Paulo, SP: Novatec, 2009. 456 p. ISBN 9788575221761.

MICHAELIS: moderno dicionário da língua portuguesa. São Paulo: Companhia Melhoramentos, 2015. Disponível em: <<http://michaelis.uol.com.br/>>.

MOBILLION. **GoalTracker**. Disponível em: <<http://mobillionapp.com/>>. Acesso em: 23 ago. 2015.

MOBILLS. **Página Inicial**. Disponível em: <<https://www.mobills.com.br/>>. Acesso em: 23 ago. 2015.

ORACLE. Learn About Java Technology. **Java**, 2015. Disponível em: <<https://www.java.com/en/about/>>. Acesso em: 24 out. 2015.

PHAM, A.; PHAM, P.-V. **Scrum em Ação: Gerenciamento e Desenvolvimento Ágil de Projetos de Software**. São Paulo: Novatec Editora, 2012. ISBN 9788575222850.

POLITI, Cassio. **Qual a diferença entre metas e objetivos?** Disponível em: <<http://www.tracto.com.br/qual-a-diferenca-entre-meta-e-objetivo/>>. Acesso em: 23 ago. 2015.

PRESCOTT, P. **HTML 5**. Tradução de Paulo Alexandre F. M. Torres. [S.l]: Babelcube Inc, 2015. ISBN 1507120443, 9781507120446. 1000 p. Disponível em: <https://books.google.com.br/books?id=5OyNCgAAQBAJ&hl=ptBR&source=gbs_navlinks>

QUIERELLI, D. A. **Criando sites com HTML-CSS-PHP: Construindo um projeto - Iniciante**. Clube de Autores, 2012. 91 p. Disponível em:
<https://books.google.com.br/books?id=TDZJ2AFV108C&dq=html&hl=ptBR&source=gb_s_navlinks_s>

SANDERSON, D. **Programming Google App Engine: Build and Run Scalable Web Apps on Google's Infrastructure**. Sebastopol, CA: O'Reilly Media, 2009. 394 p. ISBN 1449383033, 9781449383039. Disponível em:
<https://books.google.com.br/books?id=6cL_kCZ4NJ4C&dq=google+app+engine&hl=pt-BR&source=gb_s_navlinks_s>.

SCRUM.ORG. What is scrum? **Scrum.org**, 2015. Disponível em:
<<http://www.scrum.org/resources/what-is-scrum>>. Acesso em: 22 out. 2015.

SESHADRI, S.; GREEN, B. **Desenvolvendo com AngularJS: AUMENTO DE PRODUTIVIDADE COM APLICAÇÕES WEB ESTRUTURADAS**. São Paulo, SP: Novatec Editora, 2014. 352 p. ISBN 8575224093, 9788575224090. Disponível em:
<https://books.google.com.br/books?id=ZEgcBQAAQBAJ&dq=angularjs&hl=pt-BR&source=gb_s_navlinks_s>

SILVA, M. S. **Criando Sites com HTML: Sites de Alta Qualidade com HTML e CSS**. São Paulo, SP: Novatec Editor, 2008. ISBN 8575221663, 9788575221662. 432 p. Disponível em:
<https://books.google.com.br/books?id=_HBVQ-w5ZcoC&hl=pt-BR&source=gb_s_navlinks_s>

SILVA, M. S. **CSS3: desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3**. Primeira edição. São Paulo: Novatec Editora, 2012. ISBN 9788575222898. Disponível em:
<<https://novatec.com.br/livros/css3/capitulo9788575222898.pdf>>

SILVA, M. S. **JavaScript: guia do programador**. Primeira edição. ed. São Paulo: Novatec Editora, 2010. ISBN 9788575222485 Disponível em:
<<https://www.novatec.com.br/livros/javascriptguia/capitulo9788575222485.pdf>>.

VANCOUVER IT SERVICES. **GoalsOnTrack**. Disponível em:
<<http://www.goalsontrack.com/>>. Acesso em: 25 ago. 2015

VANDERDRIFT, Laura E.; AGNEW, Christopher R.; Relational Consequences of Personal Goal Pursuits. **Journal of Personality and Social Psychology**, v.106, n.6, p. 927-940, 2014.

W3C. WHAT IS CSS? **Cascading Style Sheets**, 2015. Disponível em: <<http://www.w3.org/Style/CSS/>>. Acesso em: 18 out. 2015.

WEYL, E. **What's New in CSS3**. First Edition. Sebastopol, CA: O'Reilly Media, Inc. , 2012. ISBN 9781449344931. Disponível em: <<http://www.oreilly.com/programming/free/whats-new-in-css3.csp>>

WILLIAMSON, K. **Introdução ao AngularJS**. São Paulo, SP: Novatec Editora, 2015. 256 p. ISBN 8575224301, 9788575224304 Disponível em: <https://books.google.com.br/books?id=W4jDCAAAQBAJ&dq=angularjs&hl=pt-BR&source=gbs_navlinks_s>.

WARGO, J. M. **PhoneGap Essentials: Building Cross-platform Mobile Apps**. [S.l.]: Addison-Wesley Professional, 2012. 359 p. ISBN 0321814290, 9780321814296. Disponível em: <https://books.google.com.br/books?id=HllmW5fj6HwC&dq=phonegap&hl=ptPT&source=gbs_navlinks_s>.

APÊNDICE A - QUESTIONÁRIO SOBRE A ACEITABILIDADE DO SISTEMA

Uma pesquisa foi realizada utilizando um questionário *online* com o intuito de verificar a aceitabilidade do sistema proposto e coletar dados que pudessem auxiliar na concepção do projeto.

O questionário foi divulgado ao público por meio de redes sociais e ficou ativo por um curto intervalo de tempo, isto é, do dia 28 de outubro de 2015 até o dia 31 de outubro de 2015. Neste intervalo, houve um número considerável de 100 participantes que contribuíram para sanar algumas dúvidas e reafirmar a importância deste projeto.

Basicamente, o questionário objetivo foi elaborado com sete perguntas, havendo a possibilidade das seguintes opções de respostas: sim, não e talvez. O formulário é mostrado na Figura 13.

Você conhece algum aplicativo ou sistema para controle de objetivos financeiros? *

☐ Sim
☐ Não

Usaria um aplicativo que te ajudaria a atingir metas e objetivos financeiros? *

☐ Sim
☐ Não
☐ Talvez

Compartilharia suas metas e objetivos para receber ajuda de amigos? *

☐ Sim
☐ Não
☐ Talvez

Gostaria de receber ofertas e descontos nos produtos de procura como objetivo financeiro? *

☐ Sim
☐ Não

Gostaria de dividir objetivos e metas financeiros com grupos de amigos, familiares, entre outros grupos de pessoas? *

☐ Sim
☐ Não
☐ Talvez

Gostaria de dividir e controlar gastos a fim de ajudá-lo a atingir o objetivo financeiro? *

☐ Sim
☐ Não

Gostaria de acompanhar a partir de gráficos o andamento de suas metas e objetivos financeiros? *

☐ Sim
☐ Não

Enviar

Figura 13 - Formulário da Pesquisa.
Fonte: Autor.

Os resultados da pesquisa podem ser conferidos graficamente nas figuras a seguir.



Figura 14 - Gráfico da Questão "Você conhece algum aplicativo ou sistema para controle de objetivos financeiros?"

Fonte: Autor.



Figura 15 - Gráfico da Questão "Usaria um aplicativo que te ajudaria a atingir metas e objetivos financeiros?"

Fonte: Autor.



Figura 16 - Gráfico da Questão "Compartilharia suas metas e objetivos para receber ajuda de amigos?"

Fonte: Autor.



Figura 17 - Gráfico da Questão "Gostaria de receber ofertas e descontos nos produtos que procura como objetivo financeiro?"

Fonte: Autor.



Figura 18 - Gráfico da Questão "Gostaria de dividir objetivos e metas financeiros com grupos de amigos, familiares, entre outros grupos de pessoas?"

Fonte: Autor.

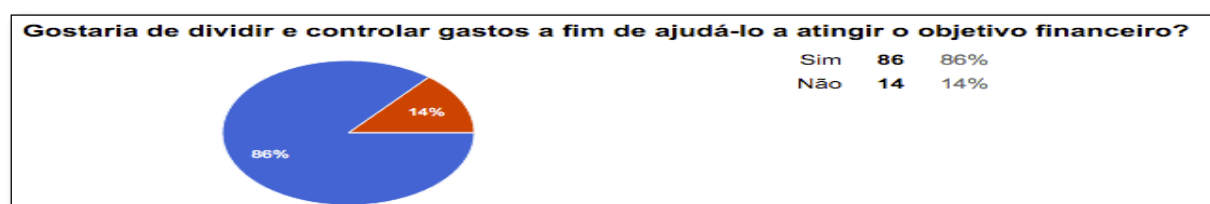


Figura 19 - Gráfico da Questão "Gostaria de dividir e controlar gastos a fim de ajudá-lo a atingir o objetivo financeiro?"

Fonte: Autor.

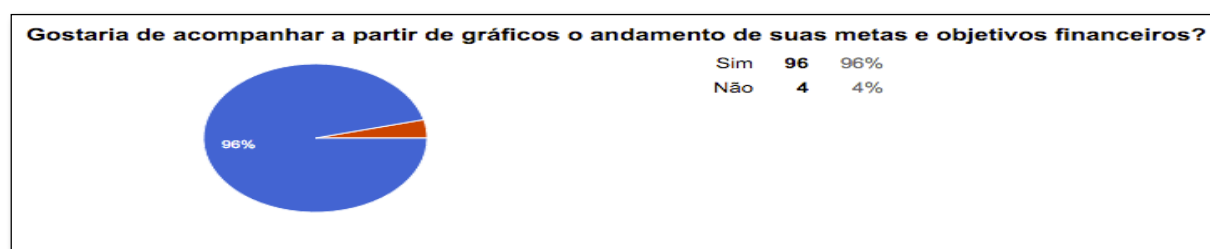


Figura 20 - Gráfico da Questão "Gostaria de acompanhar a partir de gráficos o andamento de suas metas e objetivos financeiros?"

Fonte: Autor.

APÊNDICE B - PROTÓTIPO DAS TELAS DA APLICAÇÃO

Os protótipos de algumas telas do sistema foram criados utilizando a ferramenta *online* gratuita draw.io. Os protótipos retratam algumas funcionalidades do sistema, tais como, *login*, cadastro, cadastro de grupos, entre outros. Os protótipos foram criados seguindo o *framework* Material Design da empresa Google.



Figura 21 - Protótipo da tela de *login*.
Fonte: Autor.

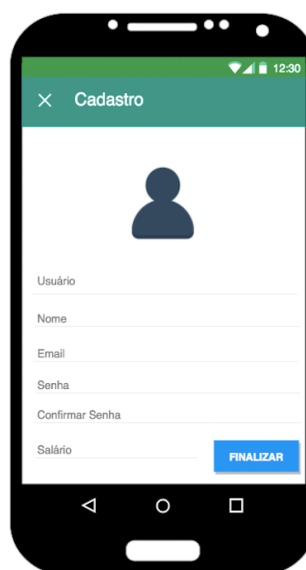


Figura 22 - Protótipo da tela de cadastro.
Fonte: Autor.



Figura 23 - Protótipo da tela de cadastro de grupo.
Fonte: Autor.

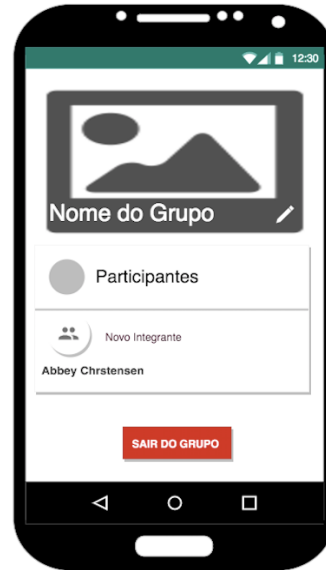


Figura 24 - Protótipo da tela das opções do grupo.
Fonte: Autor.

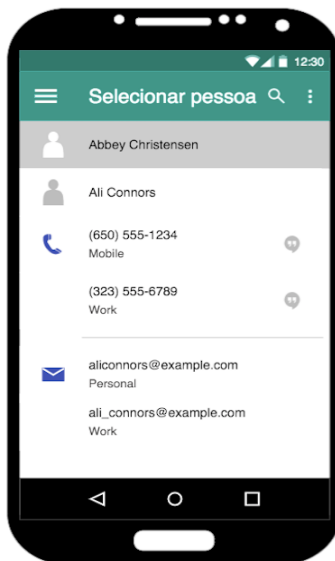


Figura 25 - Protótipo da tela de seleção de pessoas.
Fonte: Autor.



Figura 26 - Protótipo da tela das opções do objetivo.
Fonte: Autor.



Figura 27 - Protótipo da tela de criação de uma meta.
Fonte: Autor.



Figura 28 - Protótipo da tela de seleção da data da meta.
Fonte: Autor.

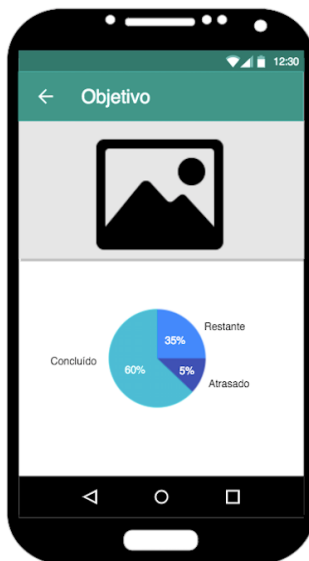


Figura 29 - Protótipo da tela do acompanhamento por meio de gráficos do objetivo.
Fonte: Autor.

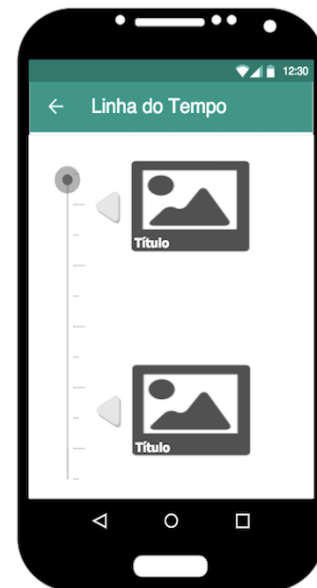


Figura 30 - Protótipo da tela da linha do tempo dos objetivos.
Fonte: Autor.