

UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE COMPUTAÇÃO
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

GIULIA DE ANDRADE CARDIERI

**KEEP ME POSTED - UMA REDE SOCIAL PARA
COLECIONADORES DE CARTÕES POSTAIS**

Bauru, SP

2015

GIULIA DE ANDRADE CARDIERI

**KEEP ME POSTED - UMA REDE SOCIAL PARA
COLECIONADORES DE CARTÕES POSTAIS**

Trabalho de Conclusão de Curso de graduação apresentado à disciplina Projeto e Implementação de Sistemas do curso de Bacharelado em Ciência da Computação da Faculdade de Ciências da Universidade Estadual Paulista “Júlio de Mesquita Filho”, como requisito parcial para obtenção do título de bacharel.

Orientador: Prof.^a Dr.^a Simone das G Domingues Prado

BAURU, SP

2015

GIULIA DE ANDRADE CARDIERI

**KEEP ME POSTED - UMA REDE SOCIAL PARA
COLECIONADORES DE CARTÕES POSTAIS**

Trabalho de Conclusão de Curso de graduação apresentado à disciplina Projeto e Implementação de Sistemas do curso de Bacharelado em Ciência da Computação da Faculdade de Ciências da Universidade Estadual Paulista “Júlio de Mesquita Filho”, como requisito parcial para obtenção do título de bacharel.

Aprovado em ____/____/____

BANCA EXAMINADORA

Profa. Dra. Simone das Graças Domingues Prado

Prof. Dr. Aparecido Nilceu Marana

Prof. Dr. João Paulo Papa

AGRADECIMENTOS

Agradeço a meus pais, que sempre estiveram ao meu lado durante esses quatro anos, me apoiando, ajudando e incentivando ideias e planos. Não chegaria até aqui sem vocês.

Aos meus amigos que sempre me ajudaram com ideias e incentivos não só nesse projeto como durante todos esses anos. Especialmente a minha amiga Raquel, com suas ótimas ilustrações, e a todos os meus amigos que testaram e emprestaram seus dados para esse projeto.

Aos meus professores da Unesp e do *Rochester Institute of Technology*, por todo conhecimento compartilhado e oportunidades, especialmente a minha orientadora Simone.

RESUMO

O desenvolvimento da *World Wide Web* ocasionou no crescimento do volume de dados produzidos, com isso surge um novo desafio: como é possível encontrar informações relevantes em meio a uma grande quantidade de conteúdo? Sistemas de recomendação são utilizados em aplicações *web* com diferentes funções como redes sociais, utilizadas especialmente para interações e troca de informações entre usuários e serviços de *streaming* de música e vídeo, por exemplo. Esses sistemas obtém informações dos usuários, e baseados nisso selecionam itens que esse indivíduo provavelmente terá interesse. Esse projeto é uma aplicação *web* denominada Keep Me Posted, no formato de uma rede social, que permite aos colecionadores de cartões postais organizarem suas coleções e interagirem com outros indivíduos que tenham interesses similares. Um sistema de recomendação foi implementado com o objetivo de facilitar a busca de postais similares ao gosto do usuário. Para desenvolver essa aplicação foi realizada uma revisão bibliográfica e uma análise de materiais e métodos, seguidos pela implementação. Essa aplicação beneficia colecionadores de cartões postais, auxiliando a organizarem suas coleções digitalmente, interagirem com outros colecionadores e encontrarem postais de acordo com seus interesses.

PALAVRAS-CHAVE:

Redes Sociais, Sistemas de Recomendação, Aplicação *Web*, *Postcrossing*.

ABSTRACT

The World Wide Web development increased the growth in the volume of data generated, which brings a new challenge: how to find relevant information in the middle of a large amount of content? Recommendation systems are used in web applications with different purposes such as social networks, which are mainly used to exchange information and interact with other users, as well as music and video streaming services. These systems acquire information from users, and based on those select items that these individuals will probably be interested. This project is a web application called Keep Me Posted, working as a social network, focused on organising a postcard collection and allow interactions among postcard collectors with similar interests. A recommendation system was implemented to improve how the user will find a postcard similar to his/hers preferences. In order to develop this application a literature review and an analysis of materials and methods were made, followed by the implementation. This application benefits postcard collectors, supporting them on organising digitally their collections, interact with other collectors and find postcards that fit their interests.

KEYWORDS:

Social Networks, Recommendation Systems, Web Applications, Postcrossing.

LISTA DE ABREVIATURAS E SIGLAS

CSS	<i>Cascading Style Sheets</i>
HTML	<i>Hypertext Markup Language</i>
MVC	<i>Model View Controller</i>
PHP	<i>Hypertext Preprocessor</i>
SASS	<i>Syntactically Awesome Style Sheets</i>
SQL	<i>Structured Query Language</i>
UML	<i>Unified Modelling Language</i>

LISTA DE FIGURAS

Figura 1: Arquitetura MVC.	16
Figura 2: Estrutura de um Sistema de Recomendação.	17
Figura 3: Exemplo de lista de postais enviados e recebidos.	26
Figura 4: Exemplo de um documento HTML.	27
Figura 5: Exemplo de uma folha de estilo CSS.	29
Figura 6: Exemplo de uma página web sem folha de estilo CSS.	29
Figura 7: Exemplo de uma página web com folha de estilo CSS.	30
Figura 8: Exemplo de uma tag PHP embutida em HTML.	33
Figura 9: Distribuição de usuários por país no Postcrossing.	35
Figura 10: Tela de cadastro do Keep Me Posted.	36
Figura 11: Tela de login do Keep Me Posted.	37
Figura 12: Categorias favoritas na página de configurações do usuário.	38
Figura 13: Adicionar um postal recebido.	41
Figura 14: Adicionar um postal para troca.	41
Figura 15: Erros nos atributos ao adicionar um postal recebido.	42
Figura 16: Visualização de um postal.	43
Figura 17: Edição de um postal.	43
Figura 18: Coleção do usuário.	44
Figura 19: Coleção do usuário filtrada pela categoria Maps.	45
Figura 20: Postais marcados como favoritos pelo usuário.	46
Figura 21: Postais de troca do usuário.	47
Figura 22: Perfil de um usuário.	48
Figura 23: Edição de um perfil de usuário.	48
Figura 24: Busca por usuários.	50
Figura 25: Busca por postais.	50
Figura 26: Busca por tags.	51
Figura 27: Busca por postais na coleção do usuário.	51
Figura 28: Configurações da conta do usuário.	52
Figura 29: Estatísticas da coleção do usuário.	53

LISTA DE QUADROS

Quadro 1: Categorias de postais no Keep Me Posted.	38
Quadro 2: Atributos de um postal.	40

SUMÁRIO

1. INTRODUÇÃO	12
1.1. Objetivos	13
<i>1.1.2 Objetivo Geral</i>	<i>13</i>
<i>1.1.3 Objetivos Específicos</i>	<i>13</i>
1.2 Organização da Monografia	14
2. FUNDAMENTAÇÃO TEÓRICA	15
2.1 Aplicações Web	15
<i>2.1.1 Arquitetura MVC</i>	<i>15</i>
2.2 Redes Sociais	16
2.3 Sistemas de Recomendação	16
<i>2.3.1 Sistemas de Recomendação Baseados em Conteúdo</i>	<i>19</i>
<i>2.3.2 Sistemas de Recomendação Colaborativos</i>	<i>20</i>
<i>2.3.3 Sistemas de Recomendação Híbridos</i>	<i>21</i>
<i>2.3.4 Sistemas de Recomendação Não Personalizados</i>	<i>22</i>
2.4 Formação de Perfis de Usuário	22
<i>2.4.1 Perfil de Usuário</i>	<i>22</i>
<i>2.4.2 Identificação no servidor</i>	<i>22</i>
<i>2.4.3 Identificação no cliente.</i>	<i>23</i>
<i>2.4.4 Coleta Explícita</i>	<i>23</i>
<i>2.4.5 Coleta Implícita</i>	<i>23</i>
2.5 Estratégias de recomendação	23
2.6 Postcrossing	24
3. MATERIAIS E MÉTODOS	27
3.1 Material	27
<i>3.1.1 HTML</i>	<i>27</i>
<i>3.1.2 CSS</i>	<i>28</i>
<i>3.1.3 Sass</i>	<i>30</i>
<i>3.1.4 Bootstrap</i>	<i>30</i>
<i>3.1.5 JavaScript</i>	<i>31</i>
<i>3.1.6 jQuery</i>	<i>31</i>
<i>3.1.7 PHP</i>	<i>32</i>
<i>3.1.8 MySQL</i>	<i>33</i>
<i>3.1.9 CodeIgniter</i>	<i>33</i>
3.2 Metodologia	34

4. DESENVOLVIMENTO DO KEEP ME POSTED	35
4.1 Detalhamento da aplicação	36
4.1.1 <i>Signup e login</i>	36
4.1.2 <i>Sistema de Recomendação</i>	37
4.1.3 <i>Adicionar um postal</i>	39
4.1.4 <i>Visualização e edição de um postal</i>	42
4.1.5 <i>Coleção</i>	44
4.1.6 <i>Favoritos</i>	45
4.1.7 <i>Troca (swap)</i>	46
4.1.8 <i>Perfil</i>	47
4.1.9 <i>Pesquisa</i>	49
4.1.10 <i>Configurações</i>	52
4.1.11 <i>Estatísticas</i>	52
4.2 Implementação do Sistema de Recomendação	53
5. CONCLUSÃO	55
5.1 Trabalhos Futuros	55
APÊNDICE A - Modelo Entidade Relacionamento	60

1. INTRODUÇÃO

A *World Wide Web* é uma das maiores fontes de informações e conhecimento que existe atualmente. Seus dados são facilmente acessados e aumentam em uma velocidade cada vez maior através de diferentes plataformas e aplicações. As redes sociais, por exemplo, são um dos principais meios para disseminar informações de forma rápida e com grande alcance. O enorme volume de dados produzido traz um novo desafio: localizar informações relevantes em meio a tanto conteúdo.

Aplicações *web* são utilizadas para as mais variadas funções, desde entretenimento como música e filmes, até para objetos de aprendizagem, largamente empregados em universidades. Esses sistemas auxiliam empresas e indivíduos a organizarem seus dados, para fácil acesso e visualização. Um dos principais exemplos desse tipo de aplicação são redes sociais como Facebook, Twitter e Instagram.

Nos dias atuais redes sociais são amplamente utilizadas. Sua principal função é a troca de informações, tanto pessoais quanto profissionais. Capra¹ (2002, p.267 apud TOMAÉL, ALCARÁ, DI CHIARA 2005 p.93) constata que "a organização em rede tornou-se um fenômeno social importante e uma fonte crítica de poder". Ao compartilhar alguma informação o usuário espera interagir com outros de interesse similar, e ao navegar deseja encontrar informações de seu interesse, fornecidas por seus amigos na rede.

Sistemas de recomendação são utilizados em diferentes tipos de aplicações *web*. Em redes sociais, por exemplo, têm o objetivo de indicar pessoas conhecidas e páginas que o usuário potencialmente possa se interessar. Para serviços de *streaming* de música, esses sistemas são responsáveis por sugestões de artistas e álbuns. Esse tipo de recomendação é realizada baseando-se em informações previamente obtidas dos usuários e são uma forma eficiente de encontrar conteúdo que possivelmente irá interessar ao indivíduo em questão. Sistemas de recomendação também são largamente utilizados em sites de comércio eletrônico como forma de sugerir compra de produtos aos clientes.

¹ CAPRA, Fritjof. **As conexões ocultas: ciência para uma vida sustentável**. São Paulo: Cultrix, 2002.

O *Postcrossing* é um projeto que existe desde 2005 e consiste em enviar e receber cartões postais de desconhecidos por todo o mundo. Atualmente, esse projeto possui mais de 500 mil usuários, de 215 países, que já trocaram mais de 30 milhões de postais (MAGALHÃES 2015a). A maioria de seus usuários são colecionadores de cartões postais, porém, os postais mostrados no Postcrossing não são uma representação exata de suas coleções, pois essas também incluem cartões postais que não foram trocados através do site do projeto. Para resolver esse problema, são utilizadas diferentes aplicações *web* e redes sociais, sem um padrão definido, onde também são mostrados digitalmente os postais disponíveis para troca com outros colecionadores.

Este trabalho pretende desenvolver uma aplicação *web* que atue no formato de uma rede social, denominada Keep Me Posted, direcionada a colecionadores de cartões postais, promovendo a integração entre este grupo de pessoas através de opções colaborativas, tais como criar bases de dados de coleções e dos postais disponíveis para troca, e emitir possíveis recomendações para trocas de acordo com preferências do usuário.

1.1. Objetivos

1.1.2 Objetivo Geral

Desenvolver uma aplicação *web* em forma de uma rede social, com um sistema de recomendação para auxiliar colecionadores de cartões postais a organizarem sua coleção e interagirem com outros colecionadores de interesses similares.

1.1.3 Objetivos Específicos

- Definir uma aplicação *web* padrão para a exibição de coleções de cartões postais;
- Implementar um sistema de recomendação para auxiliar os usuários a encontrarem informações relevantes;
- Permitir interações entre usuários de interesses similares;
- Facilitar a organização e comunicação entre usuários de interesses similares;

1.2 Organização da Monografia

Esta monografia está organizada a partir deste em outros quatro capítulos da seguinte maneira:

Capítulo 2 - FUNDAMENTAÇÃO TEÓRICA: revisão de literatura sobre temas relevantes para o desenvolvimento do projeto como Aplicações *Web*, Redes Sociais, Sistemas de Recomendação, Formação de Perfis de Usuário e *Postcrossing*.

Capítulo 3 - MATERIAIS E MÉTODOS: definição e apresentação dos materiais utilizados no desenvolvimento do projeto.

Capítulo 4 - DESENVOLVIMENTO DO KEEP ME POSTED: definição da forma como o projeto foi desenvolvido e esclarecimento das decisões tomadas durante a implementação.

Capítulo 5 - CONCLUSÃO: apresenta uma descrição dos resultados e conceitos apresentados nessa monografia e os trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta a revisão da literatura e fundamentação teórica para o trabalho.

2.1 Aplicações Web

Nos dias atuais aplicações *web* fazem parte da rotina de todos que utilizam a rede. Alguns exemplos são *e-commerce*, redes sociais, objetos de aprendizagem e serviços de *streaming*. Esses sistemas trazem uma grande vantagem em relação às aplicações *desktop*, pois não é necessário instalar nenhum *software* e há facilidade no acesso em qualquer lugar do mundo conectado à internet.

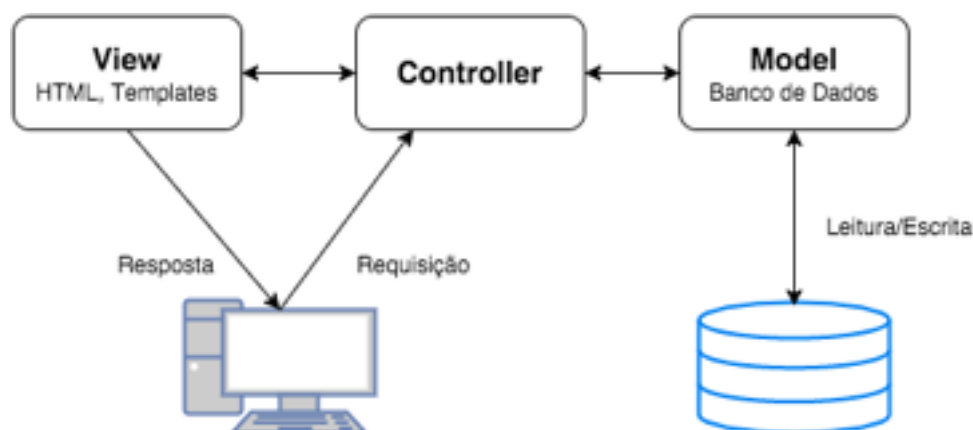
Aplicações *web* utilizam o modelo cliente-servidor. O cliente é o local onde a aplicação é utilizada, nesse caso é no navegador *web*. O servidor é responsável por manter a aplicação funcionando, pode ser local ou *web* (NATIONS, 2014).

2.1.1 Arquitetura MVC

A arquitetura *Model View Controller*, conhecida também como MVC, é um padrão de arquitetura de *software* que divide a aplicação em três camadas interconectadas, as quais são ilustradas na Figura 1. A camada *Model* trata da manipulação dos dados e da lógica de negócios da aplicação (RAMOS 2015). Além disso, de acordo com Museth (2011), a *Model* representa as estruturas de dados e tem como função interagir com o banco de dados. Essa camada não pode ser acessada diretamente por uma *View*.

A camada *View* é responsável pela interação do usuário (RAMOS 2015), e realiza a exibição dos dados, exibindo, por exemplo, um fragmento de uma página *web* ou uma página completa (MUSETH 2011). Já a camada *Controller* controla a ligação entre as outras duas camadas (RAMOS 2015), e recebe requisições do usuário e atua como o intermediário entre o *Model* e a *View* (MUSETH 2011). De acordo com Museth (2011) ao utilizar essa arquitetura é possível garantir que cada camada irá exercer sua função de forma eficiente.

Figura 1: Arquitetura MVC.



Fonte: Elaborada pelo autor.

2.2 Redes Sociais

Redes sociais são utilizadas por grande parte da população mundial. Bennett (2014) afirma que um estudo da GlobalWebIndex constatou que 67% da população acima de 12 anos dos Estados Unidos utiliza algum tipo de rede social com frequência. Esse público produz e acessa uma imensa quantidade de dados diariamente.

Atualmente, redes sociais não são utilizadas somente como forma de entretenimento. Um exemplo é o Flickr (FLICKR 2015), que é uma aplicação *web* com a finalidade de gerenciar e compartilhar imagens e vídeos, e também de organizar dados e classificá-los de acordo com parâmetros como data, local e *tags* específicas. Nessa aplicação, ainda é possível adicionar itens em grupos públicos, compartilhar álbuns com indivíduos que não utilizam a rede, adicionar amigos e enviar mensagens aos mesmos.

2.3 Sistemas de Recomendação

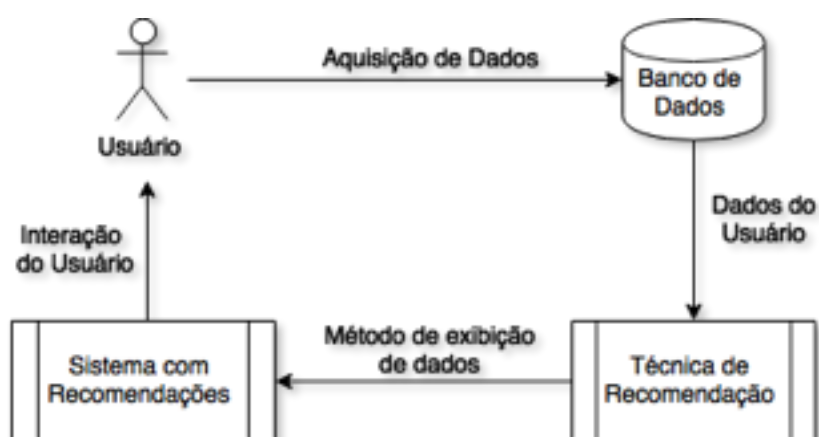
É comum pedir a opinião ou recomendação de outras pessoas em diversos assuntos, desde dicas de viagem até se a pessoa aceita ou não uma oferta de emprego. De acordo com Resnick e Varian (1997), sistemas de recomendação são responsáveis por auxiliar e ampliar a eficácia desse processo social natural.

A quantidade de informações produzida na *web* tem crescido de forma cada vez mais rápida. Um meio de reduzir a sobrecarga de dados é através de sistemas

de recomendação. Segundo Figueira Filho, Geus e Albuquerque (2008, p.1) “Esses sistemas têm por objetivo reduzir a sobrecarga de informação da *web* atual por meio da seleção de conteúdo baseada em preferências do usuário”. Um sistema de recomendação necessita de dados específicos de cada usuário para determinar quais itens são relevantes para este indivíduo, definindo um perfil de usuário. De acordo com Barcellos, Musa, Brandão e Warpechowski (2007), a aquisição desses dados não é facilmente realizada.

A classificação de um sistema de recomendação, segundo Schafer, Konstan e Riedl (1999), pode ser dividida em três principais características: a forma de aquisição de dados dos usuários, a técnica de recomendação e o método de exibição das recomendações. A Figura 2 ilustra a estrutura de um sistema de recomendação baseado na classificação proposta no início do parágrafo.

Figura 2: Estrutura de um Sistema de Recomendação.



Fonte: Elaborado pelo autor.

A aquisição de dados é o processo utilizado pelo sistema para coletar dados do usuário. A técnica de recomendação é o método implementado para selecionar quais itens são relevantes para o usuário. O método de exibição de dados define como as recomendações serão mostradas para cada indivíduo (SCHAFER, KONSTAN e RIEDL 1999).

Em 1992, Goldberg, Nichols, Oki e Terry (1992) criaram *Tapestry* (ou tapeçaria), um sistema experimental de *e-mails* que é considerado o primeiro sistema de recomendação (RESNICK, VARIAN 1997). Esse sistema foi criado com a

finalidade de melhorar a forma de gerenciar *e-mails*, pois com a expansão do uso dessa tecnologia os usuários estavam, segundo Goldberg, Nichols, Oki e Terry (1992, p.1, tradução nossa), “sendo inundados por uma grande corrente de documentos recebidos”. Com essa criação, a expressão “filtragem colaborativa” foi usada pela primeira vez e adotada por diversos autores.

Os sistemas de recomendação possuem diferentes abordagens, sendo classificados em três categorias segundo Figueira Filho, Geus e Albuquerque (2008 p.1):

“a) abordagem baseada em conteúdo, nos quais são recomendados itens similares com aqueles que o usuário mostrou preferência no passado; b) abordagem colaborativa, na qual são recomendados itens escolhidos por pessoas com preferências similares às do usuário e c) abordagens híbridas, que de alguma forma combinam técnicas na tentativa de solucionar alguns problemas potenciais das abordagens puras anteriores.”.

Além das três categorias propostas por Figueira Filho, Geus e Albuquerque (2008), Schafer, Konstan e Riedl (1999) escrevem sobre uma quarta categoria, os sistemas de recomendação não personalizados. Esse tipo de sistema recomenda os mesmos itens para todos os usuários, baseando-se na média entre a avaliação feita por todos os usuários.

Schafer, Konstan e Riedl (1999) afirmam que um sistema de recomendação pode ser classificado em automático ou manual segundo a forma de exibição das recomendações e do índice de esforço necessário pelo usuário para gerar uma recomendação. Por exemplo, um sistema automático exibe recomendações após interações simples do usuário, como navegar pelo *website*. Em comparação, um sistema manual exige esforço do usuário para buscar itens de seu interesse, que serão obtidos quando o sistema gerar recomendações.

A persistência de um sistema também pode ser um fator de classificação, sendo dividido em persistente ou efêmero. Um sistema persistente utiliza os dados da sessão atual e de sessões anteriores para gerar recomendações. Um sistema efêmero não necessita de cadastro ou *login* do usuário, gerando recomendações somente com os dados da sessão atual (SCHAFFER, KONSTAN e RIEDL 1999).

2.3.1 Sistemas de Recomendação Baseados em Conteúdo

A filtragem baseada em conteúdo (HERLOCKER, KONSTAN, RIEDL 2000) analisa informações fornecidas pelo usuário e realiza uma seleção baseada nessa análise. Essa técnica parte do princípio que os usuários, ao passar do tempo, tendem a se interessar por itens semelhantes. Sua origem vem dos avanços ocorridos na área de pesquisa sobre recuperação da informação e filtragem de informação, como a utilização do perfil do usuário, com seus gostos, preferências e necessidades (ADOMAVICIUS, TUZHILIN, 2005). Os dados relativos ao perfil do usuário podem ser obtidos de forma explícita, através de questionários ou avaliações, ou de forma implícita, seguindo o comportamento das transações anteriores do usuário.

Um sistema de recomendação baseado em conteúdo tem por objetivo, segundo Cazella, Nunes e Reategui (2010), definir se um item é relevante para determinado usuário combinando a descrição do conteúdo de cada item com os seus interesses. Consequentemente, essa técnica é recomendada para itens textuais ou com uma descrição em forma de texto, onde o conteúdo é descrito com palavras-chave (SALTON, MCGILL² 1983 apud CAZELLA, NUNES, REATEGUI, 2010). Pode-se, assim utilizar a técnica de filtragem baseada em conteúdo, solicitando ao usuário uma análise de itens, realizada na forma de avaliação. Após essa análise, o sistema busca itens com conteúdo semelhante ao que foi classificado como interessante pelo usuário. (CAZELLA, REATEGUI, 2005).

De acordo com a classificação proposta por Schafer, Konstan e Riedl (1999), os sistemas de recomendação baseados em conteúdo podem ser automáticos, quando construídos a partir da observação do comportamento constante do usuário, ou manuais, se é exigido que o usuário explicitamente indique itens de seu interesse. Com relação à persistência, essa técnica é, em sua maioria, efêmera, pois o histórico de um indivíduo não é necessário para definir itens similares aos selecionados.

Esse sistema possui limitações. Não é todo conteúdo que pode ser analisado de forma eficiente, pois dados multimídia como áudios, imagens e vídeos, por

² SALTON, Gerard; MCGILL, Michael J. **Introduction to Modern Information Retrieval**. McGraw-Hill' Inc. New York, 1986.

exemplo, exigem uma análise mais complexa. Além disso, é difícil distinguir um texto bem escrito de um mal escrito se os dois utilizarem os mesmos termos, e sinônimos podem atrapalhar a análise. Ainda pode ocorrer a super especialização, quando o sistema recomenda somente itens muito semelhantes àqueles previamente avaliados (ADOMAVICIUS, TUZHILIN, 2005).

2.3.2 Sistemas de Recomendação Colaborativos

A abordagem colaborativa em um sistema de recomendação, de acordo com Cazella, Nunes e Reategui (2010), foca na troca de experiências entre usuários com interesses similares, e não na análise de conteúdo por uma máquina (HERLOCKER, KONSTAN, RIEDL, 2000). Um item será recomendado para determinado indivíduo quando outros com gostos similares fornecerem avaliações positivas sobre esse item. Essa técnica necessita que os usuários realizem avaliações explícitas do conteúdo, o que em certos casos é um ponto negativo. Parte dos indivíduos que utilizam o sistema não possuem tempo e paciência necessários para avaliar itens (FIGUEIRA FILHO, GEUS, ALBUQUERQUE, 2008).

Os ambientes com filtragem colaborativa têm vantagens em comparação com os sistemas de recomendação baseados em conteúdo, pois, de acordo com Herlocker, Konstan e Riedl (2000, p.1, tradução nossa), a filtragem colaborativa “não depende da análise de conteúdo realizada por máquinas suscetíveis a erros”. Os mesmos autores ainda comentam sobre a habilidade com que os sistemas colaborativos têm para filtrar conteúdo de qualquer tipo, utilizar conceitos complexos como gosto e qualidade na filtragem e também fazer recomendações que surpreendam os usuários.

Sistemas de recomendação colaborativos podem ser automáticos, quando as recomendações são geradas pelo sistema, ou manuais, quando os itens selecionados dependem de avaliações explícitas dos usuários. Com relação à persistência, essa técnica normalmente é classificada como persistente, pois o sistema analisa sessões anteriores de cada indivíduo. Porém, se as sessões forem longas o suficiente, é possível classificar como efêmera (SCHAFFER, KONSTAN e RIEDL 1999). Um exemplo de um sistema de recomendação colaborativo é o MovieLens (GOOD, SCHAFFER, KONSTAN, BORCHERS, SARWAR, HERLOCKER e RIEDL 1999). Nessa aplicação, cada usuário deve avaliar filmes escolhendo uma

pontuação entre meia e cinco estrelas. A partir disso, o sistema procura outros usuários com opiniões semelhantes. O objetivo é encontrar filmes que os indivíduos com gostos parecidos assistiram mas o usuário não.

Reategui e Cazella (2005) afirmam que os sistemas de recomendação colaborativos possuem limitações. O problema do primeiro avaliador ocorre quando um novo item é adicionado no banco de dados; porém, esse não pode ser recomendado para nenhum usuário enquanto não for avaliado por mais pessoas. O problema de pontuações esparsas acontece quando o número de usuários do sistema é pequeno em relação à quantidade de informações no sistema, aumentando a probabilidade das pontuações serem muito esparsas. Outra limitação ocorre quando um usuário tem gostos diferentes do normal, tornando mais difícil encontrar outro indivíduo com interesses semelhantes e, conseqüentemente, tornando as recomendações mais fracas.

2.3.3 Sistemas de Recomendação Híbridos

Sistemas de recomendação híbridos são a integração dos sistemas colaborativos e baseados em conteúdo, evitando limitações presentes nas duas técnicas. De acordo com Adomavicius e Tuzhilin (2005), há quatro formas de combinar os dois métodos:

- A. Implementar as duas técnicas, filtragem colaborativa e filtragem baseada em conteúdo, e combinar seus resultados;
- B. Incorporar algumas características da filtragem colaborativa na baseada em conteúdo;
- C. Incorporar algumas características da filtragem baseada em conteúdo na colaborativa e;
- D. Implementar um sistema que una as duas técnicas.

Adomavicius e Tuzhilin (2005) ainda afirmam que métodos híbridos resultam em recomendações mais precisas em relação aos métodos puros.

2.3.4 Sistemas de Recomendação Não Personalizados

Os itens indicados por um sistema de recomendação não personalizado são os mesmos para todos os usuários. A recomendação é realizada através de uma análise da opinião e avaliação dos itens por outros indivíduos (SCHAFER, KONSTAN e RIEDL 1999).

De acordo com a classificação proposta por Schafer, Konstan e Riedl (1999), essa técnica de recomendação sempre gera sistemas automáticos e efêmeros. Esse fato ocorre pois as recomendações não são individualizadas, logo o usuário não precisa gerar recomendações explicitamente e não há necessidade do sistema acessar antigas sessões do usuário.

2.4 Formação de Perfis de Usuário

Antes de fazer uma recomendação, é necessário possuir algum conhecimento prévio sobre o indivíduo. Identificar quais informações serão relevantes para gerar uma recomendação personalizada é um importante passo antes de coletar e armazenar informações sobre os usuários (CAZELLA, REATEGUI, NUNES, 2010). Reategui e Cazella (2005) afirmam que, após a definição de comportamentos e dados pessoais, é possível identificar os gostos de cada usuário e customizar o sistema de acordo com as características de cada um.

2.4.1 Perfil de Usuário

Cazella, Reategui e Nunes (2010) afirmam que “Perfis de Usuários são conceitos aproximados, eles refletem o interesse do usuário com relação a vários assuntos em um momento particular”. Os mesmos autores ainda afirmam que o perfil do usuário pode ser observado como uma base de dados contendo informações, interesses e preferências sobre esse indivíduo.

Nas próximas seções serão descritos os dois principais métodos para identificar um usuário, identificação no servidor e identificação no cliente, e duas diferentes formas de coletar dados, a coleta explícita e a coleta implícita.

2.4.2 Identificação no servidor

Antes do usuário acessar o sistema é necessário realizar um cadastro com informações pessoais como nome, data de nascimento, *e-mail*, senha, entre outros.

Além disso, em cada acesso ao sistema será necessária a autenticação do usuário, através de um *login* com senha.

Todas as informações pessoais obtidas serão armazenadas em um banco de dados no servidor, que será acessado em cada tentativa de autenticação dos usuários (REATEGUI, CAZELLA 2005). De acordo com Reategui e Cazella (2005, p. 308) “Este mecanismo permite que o *website* identifique com mais precisão o usuário que nele se conecta”.

2.4.3 Identificação no cliente.

Esse método utiliza *cookies*, um mecanismo que permite ao *website* identificar qual computador está se conectando a ele. Com isso, não é possível definir acessos individuais da mesma máquina, e é necessário supor que o computador conectado é sempre utilizado pela mesma pessoa. A identificação no cliente é um mecanismo mais simples e menos confiável em relação a identificação do servidor (REATEGUI, CAZELLA 2005).

2.4.4 Coleta Explícita

Essa forma de coleta também é conhecida como customização ou personalização, pois os itens importantes são explicitamente indicados pelo usuário através de formulários ou questionários (BARCELLOS, MUSA, BRANDÃO e WARPECHOWSKI 2007). Após a escolha dos itens mais relevantes, o sistema os armazena no banco de dados em conjunto com a identificação do usuário responsável pela ação (REATEGUI, CAZELLA 2005).

2.4.5 Coleta Implícita

Segundo Cazella, Reategui e Nunes (2010), nesse método de coleta as preferências do usuário são obtidas através da análise de informações armazenadas no banco de dados. Esses dados são coletados pelo monitoramento das ações realizadas pelo indivíduo durante sua navegação pelo sistema (BARCELLOS, MUSA, BRANDÃO e WARPECHOWSKI, 2007).

2.5 Estratégias de recomendação

Existem diferentes formas de exibir recomendações aos usuários. Schafer, Konstan e Riedl (1999) identificaram quatro diferentes métodos de encontrar

recomendações, navegação orgânica, lista de recomendação, selecionar opções e palavra-chave, as quais serão especificadas a seguir.

- **Navegação Orgânica:** nesse método as recomendações são itens da navegação uma página, portanto não é exigida nenhuma tarefa adicional dos usuários para encontrar itens recomendados (SCHAFFER, KONSTAN e RIEDL 1999).
- **Lista de Recomendação:** esse método exige que os usuários façam uma requisição das recomendações para o sistema, as quais serão exibidas como uma lista (SCHAFFER, KONSTAN e RIEDL 1999). As listas de recomendação não analisam profundamente o perfil de cada usuário, ordenando itens de forma geral como “Itens mais vendidos” ou de interesse especial (REATEGUI, CAZELLA 2005). A vantagem desse método é sua facilidade de implementação e adequação às necessidades do sistema, como lucratividade e *marketing*. A desvantagem é a falta de personalização das listas (REATEGUI, CAZELLA 2005).
- **Selecionar Opções:** para obter recomendações, os usuários precisam interagir com o sistema. Schafer, Konstan e Riedl (1999) afirmam que: “Normalmente, consumidores escolhem a partir de um conjunto de opções/critérios pré-definidos para basear suas recomendações” (SCHAFFER, KONSTAN e RIEDL 1999, tradução nossa). Os critérios pré-definidos podem ser categorias ou um conjunto finito de opções específicas como título de uma obra num sistema de recomendação de filmes, por exemplo.
- **Palavra-chave:** esse método exige que o usuário providencie palavras-chave para o sistema. Essas palavras serão utilizadas para encontrar itens similares e indicá-los a um indivíduo. Para utilizar esse método, é necessário que o usuário saiba especificamente os termos e itens que está interessado (SCHAFFER, KONSTAN e RIEDL 1999).

2.6 Postcrossing

O *Postcrossing* é um projeto que existe desde 2005 e consiste em enviar e receber cartões postais de desconhecidos por todo o mundo. Segundo a página do

projeto³, seu objetivo é “conectar pessoas pelo mundo através de cartões postais, independente de sua localização, idade, gênero, raça ou religião” (MAGALHÃES 2015 B, tradução nossa). Atualmente o projeto possui mais de 500 mil usuários de 215 países que já trocaram mais de 30 milhões de postais (MAGALHÃES 2015 A).

Para participar do projeto, é necessário se cadastrar no site do mesmo, criando um nome de usuário e fornecendo informações como *email* e endereço, e aceitar as regras de uso. Após esse cadastro, deve-se selecionar a opção de enviar um postal. O sistema irá selecionar um usuário aleatoriamente para enviar um postal pelo serviço de correios tradicional.

A identificação de cada postal é feita por um código chamado *Postcrossing* ID, que consiste em duas letras identificadoras de cada país e uma série numérica contínua indicando a quantidade de postais com origem naquele país separados por um hífen. Por exemplo, o postal número 2693 enviado do Brasil tem o código BR-2693. Esse código deverá ser escrito ou impresso no postal e o destinatário é responsável por cadastrar o recebimento desse postal no sistema.

Cada usuário possui um perfil público, onde é possível ver uma lista de postais enviados e recebidos com seus respectivos *Postcrossing* ID, nomes de usuário dos destinatários, data recebida, e uma imagem, quando disponibilizada pelo remetente ou destinatário. Um exemplo dessa lista é exibido na Figura 3, com o histórico dos postais recebidos e enviados por um usuário em 2015. O conjunto de imagens referentes aos postais enviados fica disponível no perfil do usuário em um mural de postais, que pode ser filtrado de acordo com o país recebido.

No Postcrossing há quatro tipos de interação entre usuários:

- A. Recebimento de um postal: o usuário deve entrar no site do projeto e registrar o recebimento de um postal imediatamente, nesse caso o remetente será informado por *e-mail*.
- B. Envio de um postal: todo envio cria uma relação no sistema entre o remetente e o destinatário.

³ MAGALHÃES, Paulo. **About Postcrossing**. 2015. Disponível em <<https://www.postcrossing.com/about>>. Acesso em 19 de Maio de 2015.

- C. Envio de mensagem direta: é possível enviar uma mensagem para um usuário ao entrar em seu perfil, porém essa será recebida somente por *e-mail*.
- D. Marcar um postal em um mural como favorito: pode-se marcar como favorito qualquer postal presente no mural de um usuário. Após a marcação, essa imagem fica disponível no mural de postais do usuário que efetuou a ação na área de favoritos.

Figura 3: Exemplo de lista de postais enviados e recebidos.

Postcards exchanged in 2015

Sent 18 postcards				Received 20 postcards			
 BR-360225	Amelie2310 	8 Sep		 RU-3991233	Gortenziya	30 Oct	
 BR-360223	StarWarsHips	2 Sep		 PL-1094050	EwaiWiesiu	30 Oct	
 BR-360222	Topaze	2 Sep		 DE-4503498	EmKay89 	22 Sep	
 BR-360220	Saelita	31 Aug		 BY-1660462	olgapronkevich	4 Sep	
 BR-360224	Snug-as-a-bug	21 Aug		 IL-42829	Matzpen	4 Sep	
 BR-355062	nathaliesarah_	21 Jul		 HU-93234	kicsek89	4 Sep	
 BR-354457	Kekel 	16 Jul		 BY-1677389	RoxyOxygen	3 Sep	
 BR-354460	karryan	15 Jul		 ZA-77345	Willie28	3 Sep	
 BR-355061	Jochelbeere	11 Jul		 US-3501466	arianneroxa 	31 Jul	
 BR-354458	MinaM	10 Jul		 RU-3841693	Iarashipovsky	31 Jul	
 BR-352913	Feba8	9 Jul		 UA-1322422	PaniYurinska 	31 Jul	
 BR-354459	Roowan	4 Jul		 RU-3841689	Bosikom2	31 Jul	
 BR-351642	mamatory	3 Jul		 DE-4351284	qg_from_nu	23 Jul	
 BR-352914	rina_novi	2 Jul		 DE-4337896	meikejohanna	23 Jul	
 BR-351638	Sechina	30 Jun		 DE-4337891	Dorle	23 Jul	
 BR-351134	karkunny	29 Jun		 RO-105206	Bazalai 	21 Jul	
 BR-351133	Litvinenko	28 Jun		 NL-3057687	Indra1975	21 Jul	
 BR-351132	imaginary	25 Jun		 LV-193882	Maija_Meiere	17 Jul	
				 IT-322593	wimey	10 Jul	
				 FI-2434480	Skriivar	10 Jul	

Fonte: https://www.postcrossing.com/user/Giulia_cardieri.

3. MATERIAIS E MÉTODOS

Neste Capítulo são apresentados os materiais e métodos utilizados para o desenvolvimento do trabalho.

3.1 Material

3.1.1 HTML

HTML é a sigla para *Hyper Text Markup Language*, que significa Linguagem de Marcação de Hipertexto (W3 SCHOOLS 2015). Essa linguagem foi criada por Tim Berners-Lee, com a finalidade de auxiliar a comunicação e espalhamento de pesquisas entre seu grupo de colegas, mas foi somente com a criação da internet que tornou-se amplamente conhecida (EIS 2011).

O HTML é uma linguagem de marcação, portanto seus elementos devem ser marcados por *tags* para gerar o que será exibido na página. Cada *tag* descreve um diferente conteúdo. A Figura 4 mostra o exemplo de um documento HTML com os elementos que serão exibidos na página *web* marcados por *tags* HTML.

Figura 4: Exemplo de um documento HTML.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="UTF-8">
5      <title>Título da Página</title>
6      <meta name="Author" content=""/>
7      <link rel="stylesheet" type="text/css" href="style.css">
8  </head>
9  <body>
10     <nav>
11         <ul>
12             <li>Item 1</li>
13             <li>Item 2</li>
14             <li>Item 3</li>
15         </ul>
16     </nav>
17     <main>
18         <h1>Um Título</h1>
19         <p>Um parágrafo</p>
20     </main>
21 </body>
22 </html>
23
```

Fonte: Elaborado pelo autor.

No exemplo de documento HTML exibido na Figura 4, observam-se diferentes *tags*, como a responsável por exibir um parágrafo, representada por `<p>`. Um documento HTML é interpretado pelo navegador e exibido como uma página *web*.

3.1.2 CSS

CSS é a sigla para *Cascading Style Sheets*, que significa folhas de estilo em cascata. Essa linguagem é utilizada para aplicar estilo nas páginas criadas com HTML, customizando sua aparência (W3C 2015). Uma folha de estilo pode ser utilizada em diferentes páginas HTML, permitindo editar o estilo gráfico de um projeto a partir de um único documento, como um efeito cascata.

O CSS possui várias declarações, cada uma com um seletor relativo a uma *tag* HTML e uma regra para esse elemento. A Figura 5 é um exemplo de um documento CSS, com seletores e regras responsáveis por alterar o estilo da página HTML.

No exemplo de folha de estilo exibido na Figura 5, cada declaração é relativa a uma *tag* do documento HTML exibido na Figura 4, como *body*, *main* e *nav*. As regras são responsáveis por alterar o estilo padrão da página. Uma página *web* padrão, sem alterações em sua folha de estilo, pode ser observada na Figura 6. Um exemplo de uma página *web* alterada pela folha de estilo da Figura 5 pode ser observada na Figura 7.

Figura 5: Exemplo de uma folha de estilo CSS.

```

1  body {
2      background-color: #3d3d3d;
3      color: #f9f9f9;
4      font-family: helvetica, arial;
5      margin: 0;
6  }
7
8  p {
9      font-size: 14pt;
10     line-height: 16pt;
11 }
12
13 main {
14     margin-left: 10%;
15     padding: 0 2.5%;
16     width: 90%;
17 }
18
19 nav {
20     background-color: #7E5691;
21     height: 100%;
22     left: 0;
23     position: fixed;
24     top: 0;
25     width: 10%;
26 }
27
28 ul {
29     margin: 0;
30     padding: 0;
31 }
32
33 ul li {
34     background-color: #7E5691;
35     border-bottom: 1px solid #555;
36     list-style: none;
37     padding: 10px 20px;
38 }

```

Fonte: Elaborado pelo autor.

Figura 6: Exemplo de uma página web sem folha de estilo CSS.



Fonte: Elaborado pelo autor.

Figura 7: Exemplo de uma página web com folha de estilo CSS.



Fonte: Elaborado pelo autor.

3.1.3 Sass

Sass é a sigla para *Syntactically Awesome Stylesheets*, que significa folhas de estilo com incrível sintaxe. Sass é um pré-processador de CSS e permite o uso de atributos inexistentes no CSS como variáveis, indentação, herança e outros. Um arquivo criado em Sass é compilado e transformado em um arquivo CSS comum (Sass 2015).

Optou-se usar um pré-processador de CSS pela facilidade em manipular vários arquivos menores no lugar de um muito extenso, e pela facilidade em criar variáveis para cores e outros atributos que se repetem e criar funções (*mixins*) para gerar várias linhas de código similares, porém com diferentes parâmetros.

O Sass foi escolhido, pois sua estrutura é mais fácil de ser utilizada em relação a outros pré-processadores como o Less (THE CORE LESS TEAM 2016). Além disso, esse pré-processador possui extensa documentação que beneficia o aprendizado dessa tecnologia.

3.1.4 Bootstrap

"Bootstrap é o *framework* de HTML, CSS e JavaScript mais popular para desenvolver projetos responsivos e *mobile first* na web" (BOOTSTRAP 2015, tradução nossa). Esse *framework* foi desenvolvido por um designer e um desenvolvedor do Twitter em 2010 como um projeto interno chamado *Twitter*

Blueprint, e após perceber seu sucesso foi lançado oficialmente em 2011 (BOOTSTRAP 2015).

Esse *framework* fornece uma estrutura para o desenvolvimento de projetos *web* através de uma coleção personalizável de elementos e funções dentro de arquivos CSS e JavaScript. Para usar o Bootstrap, é necessário incluir no projeto a biblioteca JavaScript jQuery.

O Bootstrap foi escolhido para o Keep Me Posted pela facilidade em criar projetos com boa usabilidade em diferentes tipos de dispositivos e tamanhos de telas. De acordo com David (2013), Bootstrap agiliza o tempo de desenvolvimento pois possui vários elementos HTML já estilizados e em alguns casos com o *plugin* JavaScript necessário para seu funcionamento.

Esse *framework* possui uma grande comunidade ativa, o que auxilia o desenvolvedor a encontrar possíveis soluções em caso de erros. Além disso, esse *framework* é continuamente atualizado e mantido no GitHub (GIMMER 2014). O site oficial do Bootstrap fornece uma documentação completa sobre os elementos do *framework* e mostra exemplos práticos de seu uso.

3.1.5 JavaScript

JavaScript é uma linguagem que atua na parte do cliente, funcionando inteiramente dentro do navegador *web* (NIXON 2014, p.324, tradução nossa). Essa linguagem foi criada por Brendan Erich em 1995 na Netscape (TABLELESS 2013). De acordo com Nixon (2014), não há relação entre a linguagem Java e JavaScript além do nome em comum. A similaridade se originou em uma jogada de *marketing* visando beneficiar a nova linguagem a partir da popularidade de Java.

Por se tratar de uma linguagem *client-side*, JavaScript é utilizada para manipular elementos e comportamentos numa página através do HTML e CSS, adicionando funcionalidades dinâmicas a um *website* como novas imagens e cores. (NIXON 2014 p. 323, tradução nossa).

3.1.6 jQuery

jQuery é uma biblioteca JavaScript. É conhecida por ser rápida, pequena e com muitos atributos. Seu objetivo é facilitar a manipulação de eventos, animações e

outras funções realizadas com JavaScript (THE JQUERY FOUNDATION, 2015). Entre os atributos do jQuery estão incluídos manipulação do HTML e CSS, métodos de eventos HTML, efeitos e animações, AJAX e utilidades (W3 SCHOOLS 2016). É necessário ter conhecimento básico de HTML, CSS e JavaScript antes de utilizar a biblioteca jQuery, pois seu conteúdo manipula os elementos criados com essas linguagens (W3 SCHOOLS 2016).

A escolha de usar jQuery está associada ao *framework* Bootstrap, que necessita dessa biblioteca para funcionar corretamente. Além disso, o jQuery possui muitas funções prontas que facilitam a utilização de JavaScript, e todas essas funções estão na documentação da biblioteca, de fácil acesso *online*.

3.1.7 PHP

PHP, um acrônimo recursivo para PHP:*Hypertext Preprocessor*, é uma linguagem de *script open source* embutido no HTML. Seu objetivo é permitir escrever, de forma rápida, páginas geradas dinamicamente (THE PHP GROUP, 2015). Diferente do JavaScript, PHP é uma linguagem executada no servidor. O cliente, no caso o navegador, recebe somente os resultados desse *script*, sem acesso ao código fonte (THE PHP GROUP, 2015).

O código PHP é delimitado por *tags* de início e fim dentro de documentos HTML. Na Figura 8 pode-se observar que uma *tag* PHP é iniciada por “<?php” e finalizada por “?>”. Uma *tag* PHP pode ser incluída em qualquer parte de uma página HTML.

Figura 8: Exemplo de uma tag PHP embutida em HTML.

```
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <meta charset="UTF-8">
6      <title>Título da Página</title>
7      <link rel="stylesheet" type="text/css" href="style.css">
8  </head>
9
10 <body>
11     <h1>Um Título</h1>
12     <p>Um parágrafo.</p>
13     <?php echo 'Uma tag PHP';| ?>
14 </body>
15
16 </html>
```

Fonte: Elaborado pelo autor.

3.1.8 MySQL

MySQL é um sistema de gerenciamento de banco de dados, é *open source*, distribuído pela Oracle Corporation (ORACLE CORPORATION 2015 A) e foi desenvolvido no meio da década de 1990 (NIXON 2014, tradução nossa). De acordo com Nixon (2014), MySQL é o sistema de gerenciamento de banco de dados mais popular para servidores *web*, e é extremamente poderoso e rápido, até mesmo ao ser utilizado no mais simples *hardware*. As tabelas criadas no MySQL são relacionais e utilizam a linguagem SQL, que é utilizada para inserir, acessar, alterar e deletar o conteúdo armazenado nos bancos de dados.

MySQL foi escolhido pois é um sistema gerenciador de banco de dados flexível, escalável e com rápida performance (ORACLE CORPORATION 2015 B). Essas características são importantes para esse projeto, pois há um número grande de dados armazenados que precisam ser exibidos com rapidez. Além disso, com o cadastro de novos usuários, os dados podem aumentar rapidamente pois um colecionador pode cadastrar centenas de postais.

3.1.9 CodeIgniter

CodeIgniter é um *framework* PHP criado por EllisLab, uma empresa que desenvolve *software*. De acordo com o site oficial, esse *framework* foi “construído

para desenvolvedores que precisam de um *toolkit* simples e elegante para criar aplicações *web* completas” (CODEIGNITER 2015, tradução nossa). Esse *framework* sugere o uso do padrão de arquitetura MVC e não necessita do uso de linguagens em forma de *templates* para exibir os dados (CODEIGNITER, 2015).

A escolha de usar um *framework* PHP surgiu da facilidade em utilizar o modelo MVC associado a um *framework*. O CodeIgniter possui várias bibliotecas e *helpers* para auxiliar no desenvolvimento de uma aplicação. Em comparação com outros *frameworks*, o CodeIgniter possui uma documentação completa, uma comunidade grande e é fácil de usar. Além disso, também é recomendado para iniciantes, o que foi um fator importante na escolha do CodeIgniter.

3.2 Metodologia

Para atingir os objetivos propostos nesse trabalho, inicialmente foi realizada uma revisão bibliográfica, onde foram pesquisados e analisados artigos científicos, teses, dissertações e livros sobre os assuntos relacionados. Em seguida, linguagens de programação e sistemas de gerenciamento de bancos de dados necessários para o desenvolvimento desse projeto foram estudados.

Após a fase inicial, o foco foi na análise de requisitos necessários para o desenvolvimento da aplicação proposta. Com as informações devidamente organizadas, as especificações do projeto foram definidas através de diagramas E-R para criação de tabelas de bancos de dados, e diagramas UML (*Unified Modeling Language*) para a modelagem do *software*.

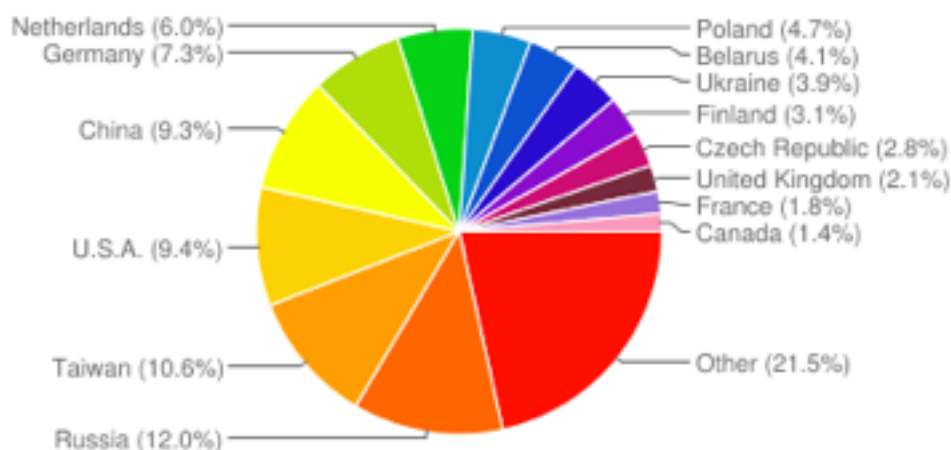
Com o projeto devidamente estruturado, iniciou-se a fase de implementação através do desenvolvimento da parte *front-end* de algumas páginas do projeto.

4. DESENVOLVIMENTO DO KEEP ME POSTED

Neste trabalho, foi desenvolvida uma aplicação *web* que atua na forma de uma rede social, denominada Keep Me Posted, voltada para colecionadores de cartões postais, possibilitando ao usuário organizar e administrar sua coleção, encontrar e interagir com outros colecionadores, encontrar postais de seu interesse e buscar postais de acordo com características específicas. A aplicação possui um sistema de recomendação com o objetivo de facilitar a busca de conteúdo para o usuário.

A aplicação foi desenvolvida em inglês, pois espera-se que os colecionadores de postais sejam de diferentes países. Essa suposição foi concluída a partir da variedade de países no *Postcrossing*. A Figura 9 mostra a distribuição de usuários por país do *Postcrossing* (MAGALHÃES 2016). As tabelas de banco de dados utilizadas nessa aplicação foram modeladas através de um modelo entidade relacionamento, que é exibido no Apêndice A.

Figura 9: Distribuição de usuários por país no Postcrossing.



Fonte: <https://www.postcrossing.com/stats/users>

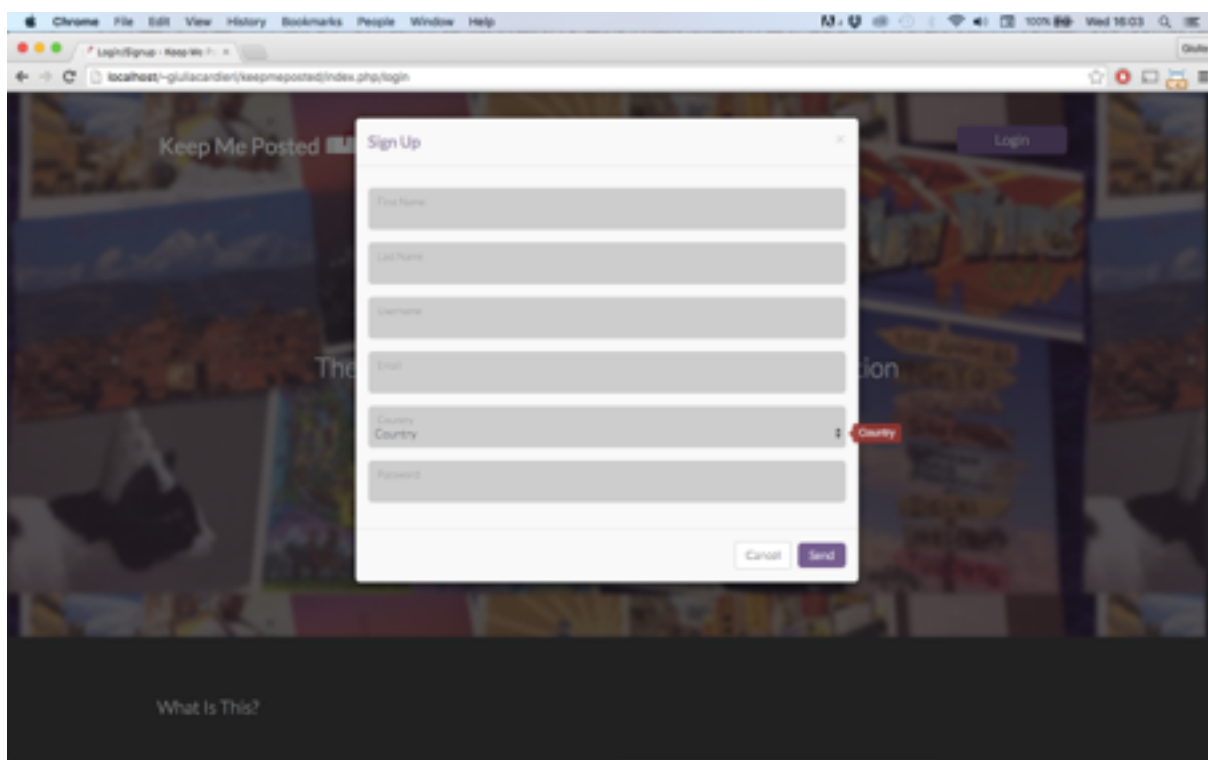
Essa aplicação foi desenvolvida utilizando o *framework* CodeIgniter com a linguagem PHP e o sistema gerenciador de banco de dados MySQL para a parte do servidor/*back-end*. Já o *framework* Bootstrap, o pré-processador de CSS Sass, a biblioteca de JavaScript jQuery, e a linguagem HTML para a parte do cliente/*front-end*.

4.1 Detalhamento da aplicação

4.1.1 Signup e login

Antes de utilizar a aplicação, é necessário se cadastrar/*signup* ou realizar o *login*. Após esse passo, o usuário é identificado e tem acesso a todos os componentes da aplicação. A Figura 10 ilustra a tela para usuários não cadastrados. A Figura 11 exibe o formulário de *login*.

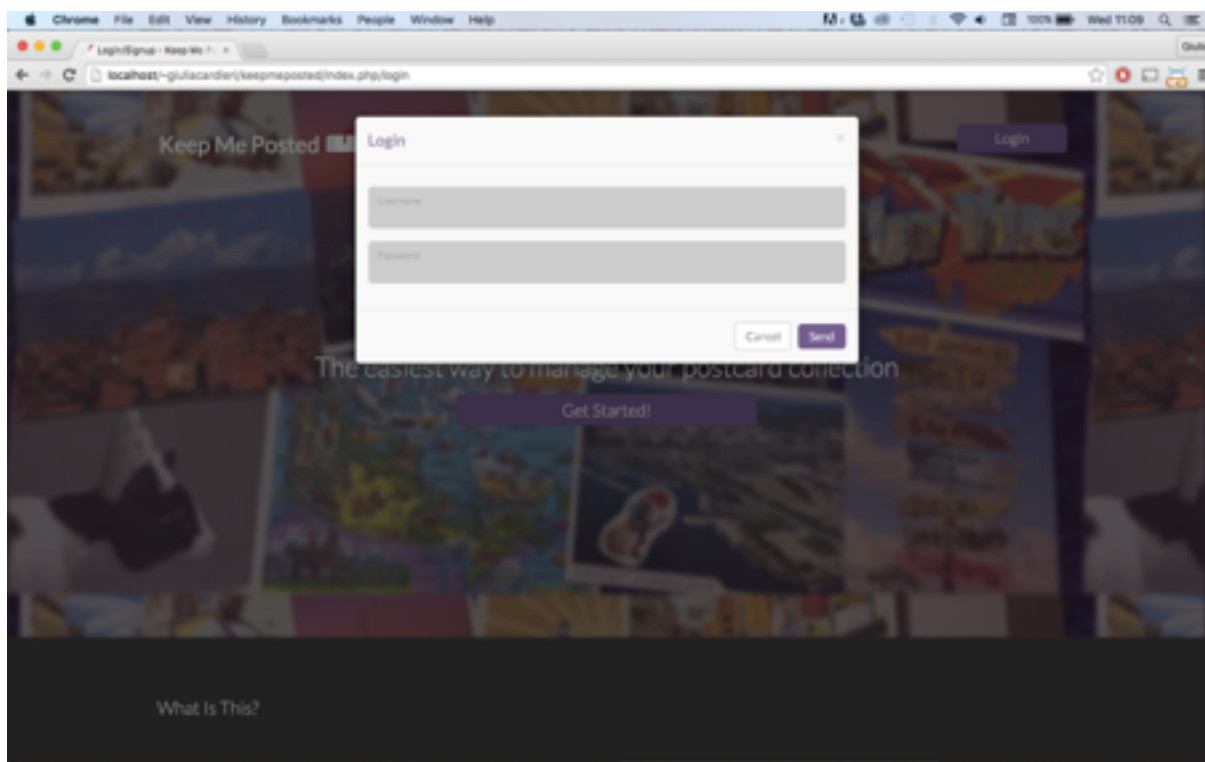
Figura 10: Tela de cadastro do Keep Me Posted.



The image shows a web browser window displaying the 'Sign Up' form for 'Keep Me Posted'. The browser's address bar shows the URL 'localhost/~glufacander/keepmeposted/index.php/login'. The form is a white modal box with the following fields: 'First Name', 'Last Name', 'Username', 'Email', 'Country' (with a dropdown arrow), and 'Password'. At the bottom of the form are 'Cancel' and 'Send' buttons. The background of the page is a collage of various images, and a 'Login' button is visible in the top right corner of the page area.

Fonte: Elaborado pelo autor.

Figura 11: Tela de login do Keep Me Posted.



Fonte: Elaborado pelo autor.

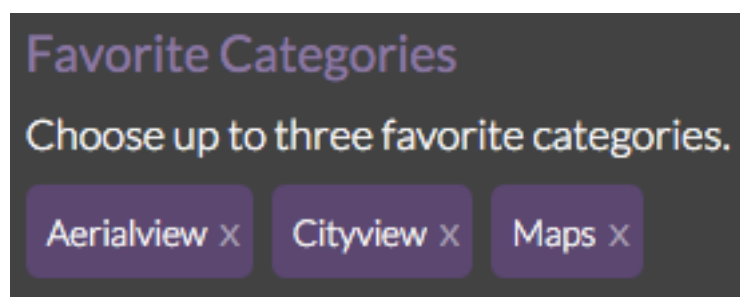
4.1.2 Sistema de Recomendação

A aplicação desenvolvida possui um sistema de recomendação que gera dois tipos diferentes de recomendação: populares e individuais para o usuário. Os postais recomendados são exibidos para o usuário na página *Home*, logo após realizar o *login*. As recomendações geradas podem ser classificadas como baseadas em conteúdo (recomendações individuais) e não personalizadas (recomendações populares), conforme explicado nas Seções 2.3.1 e 2.3.4, respectivamente.

As recomendações populares são geradas com base nas informações obtidas através da interação de todos os usuários que utilizam o sistema. Informações são obtidas, de forma implícita, analisando dados cadastrados no banco de dados. Os postais indicados são aqueles que mais pessoas marcaram como favorito e estão nas categorias e países mais populares entre todos os usuários. O objetivo desse tipo de recomendação é permitir explorar postais que a maioria dos colecionadores acha interessante, mas não necessariamente se encaixam no padrão atual de suas coleções.

As recomendações individuais para o usuário baseiam-se somente no perfil e no conteúdo já cadastrado. As informações necessárias para essa recomendação são obtidas de forma explícita e implícita. O usuário pode explicitamente escolher até três categorias favoritas em suas configurações, como exibido na Figura 12, as quais são utilizadas para gerar as recomendações, e no caso do usuário não cadastrar nenhum postal em sua coleção ou em seus favoritos essa será a única fonte de informação para o sistema de recomendação. As recomendações geradas utilizam como fonte implícita de informação os dados do usuário no banco de dados. Por exemplo, se o usuário tem muitos postais favoritos de uma categoria que não é uma de suas três categorias favoritas escolhidas, essa categoria será implicitamente considerada como a quarta categoria favorita do usuário e fará parte das recomendações. Além das categorias, os países com maior número de postais na coleção e nos favoritos do usuário também são obtidos implicitamente. O conjunto desses dados é utilizado para selecionar os itens que serão exibidos nas recomendações específicas.

Figura 12: Categorias favoritas na página de configurações do usuário.



Fonte: Elaborado pelo autor.

As categorias criadas para essa aplicação, e a explicação de quais tipos de postal se encaixam em cada uma delas são exibidas no Quadro 1.

Quadro 1: Categorias de postais no Keep Me Posted.

Nome da Categoria	Tipo de postal
Advertising/Propaganda	Postais de propaganda, normalmente são gratuitos.
Aerialview/Vista Aérea	Postais com fotos predominantemente de vistas aérea.
Animals/Animais	Postais com fotos animais.
Beaches/Praias	Postais com fotos de praias.

Nome da Categoria	Tipo de postal
<i>Black and White/Preto e Branco</i>	Postais em preto e branco.
<i>Christmas/Natal</i>	Postais temáticos de Natal.
<i>Churches/Igrejas</i>	Postais com fotos internas ou externas de igrejas.
<i>Cityview/Vista de cidade</i>	Postais com fotos de cidades.
<i>Flag/Bandeira</i>	Postais com ilustrações ou fotos de bandeiras.
<i>Food/Comida</i>	Postais com ilustrações ou fotos de comida.
<i>Greetings From/ Saudações De</i>	Postais temáticos com fotos e/ou ilustrações e a frase “ <i>Greetings From</i> ”.
<i>Handmade/Feito a mão</i>	Postais feitos a mão ou não impressos por uma gráfica.
<i>Illustration/Ilustrações</i>	Postais com ilustrações e/ou desenhos.
<i>Landscapes/Paisagens</i>	Postais com paisagens variadas que não se encaixam em outras categorias.
<i>Maps/Mapas</i>	Postais de qualquer tipo de mapas.
<i>Movies/Filmes</i>	Postais com fotos ou temáticos de filmes.
<i>Multiview/Vista Múltiplas</i>	Postais com duas ou mais fotos ou ilustrações.
<i>Music/Música</i>	Postais de artistas de música ou temático de alguma música.
<i>Nature/Natureza</i>	Postais com paisagens ou ilustrações sobre a natureza.
<i>Nightview/Vista Noturna</i>	Postais com fotos noturnas de qualquer paisagem ou cidade.
<i>Other/Outros</i>	Postais que não se encaixam em nenhuma outra categoria.
<i>TV Series/Séries de TV</i>	Postais temáticos de alguma série de tv. Fotos da série, de atores e atrizes ou ilustrações.
<i>Vintage</i>	Postais antigos, com 25 anos ou mais, ou com fotos antigas.
<i>People/Pessoas</i>	Postais que mostram somente pessoas.
Unesco	Postais com fotos de patrimônios mundiais da UNESCO.

Fonte: Elaborado pelo autor.

4.1.3 Adicionar um postal

Para adicionar um postal, o usuário precisa fornecer atributos sobre ele, como ilustrado no Quadro 2, e definir se foi recebido, ou seja, se fará parte de sua coleção, ou se é um postal para troca (*swap*). Até cinco *tags* podem ser adicionadas em cada postal. Uma *tag* tem como objetivo identificar o postal e facilitar as pesquisas por termos específicos no motor de busca.

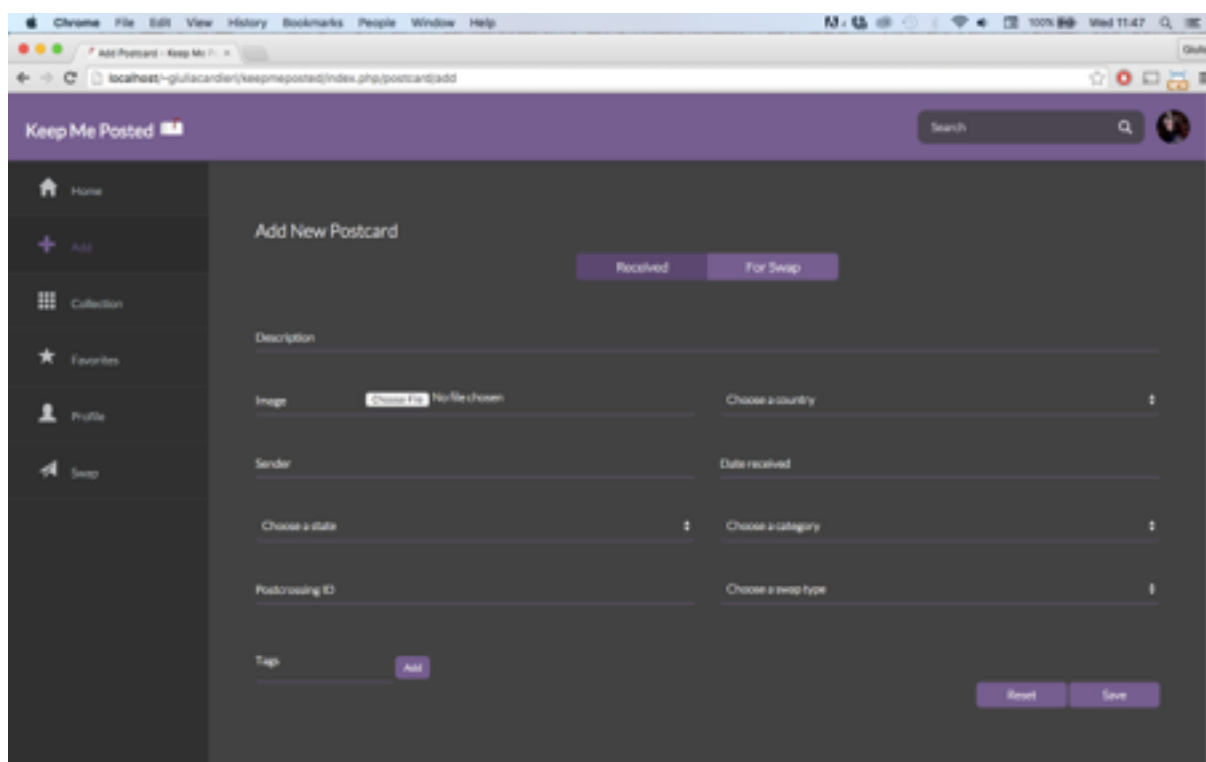
As Figuras 13 e 14 ilustram, respectivamente, as telas criadas para adicionar um postal recebido e de troca. Nessas telas o usuário deve preencher os atributos citados no Quadro 2. Após salvar os dados, o usuário é encaminhado para uma página de visualização de postal, que será explicada na Seção 4.1.4. No caso de erros ao preencher os atributos, o sistema indicará os erros para o usuário, que deve corrigi-los com a finalidade de adicionar um postal para troca ou em sua coleção. A Figura 15 ilustra o formulário de adicionar postais com erros nos atributos. Já os atributos de cada postal adicionado podem ser observados no Quadro 2.

Quadro 2: Atributos de um postal.

Atributo	Função	Obrigatoriedade
<i>Description/</i> Descrição	Descrever o postal	Obrigatório
<i>Image/Imagem</i>	Imagem do postal escaneado	Opcional
<i>Sender/Remetente</i>	Nome da pessoa que enviou o postal	Obrigatório para postais recebidos
<i>State/Estado</i>	A forma como o postal foi recebido: em branco, escrito, escrito e selado ou danificado	Obrigatório para postais recebidos
<i>Postcrossing ID</i>	O número de identificação do postal no site do <i>Postcrossing</i>	Opcional
<i>Country/País</i>	O país de origem do postal, não necessariamente de onde ele foi enviado	Obrigatório
<i>Date Received/</i> Data Recebida	A data que o postal foi recebido	Obrigatório para postais recebidos
<i>Category/Categoria</i>	A categoria que esse postal se encaixa	Obrigatório
<i>Swap/Troca</i>	O tipo de troca realizado para receber esse postal: forum do <i>postcrossing</i> , presente, pessoal (o próprio usuário comprou), pelo site oficial do <i>Postcrossing</i> ou através de uma troca com outra pessoa.	Obrigatório para postais recebidos
<i>Tags</i>	Identificam o postal com palavras-chave	Opcional
<i>Available/</i> Disponibilidade	Define se o postal está disponível para troca	Obrigatório para postais de troca

Fonte: Elaborado pelo autor.

Figura 13: Adicionar um postal recebido.



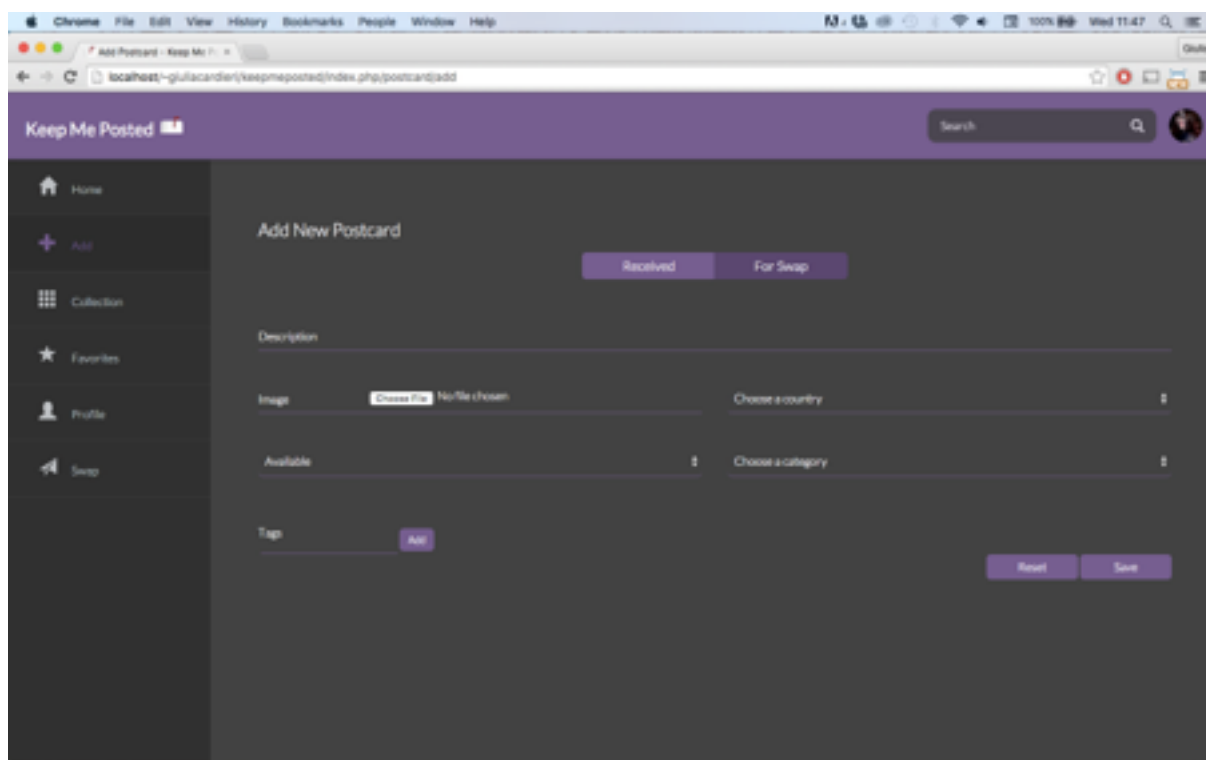
The screenshot shows a web browser window with the URL `localhost~guliacander/keepmeposted/index.php/postcard/add`. The page title is "Keep Me Posted". On the left is a sidebar with navigation links: Home, Add, Collection, Favorites, Profile, and Swap. The main content area is titled "Add New Postcard" and features two tabs: "Received" (selected) and "For Swap". The form includes the following fields:

- Description (text input)
- Image (file upload button labeled "Choose file", status "No file chosen")
- Choose a country (dropdown menu)
- Sender (text input)
- Date received (text input)
- Choose a state (dropdown menu)
- Choose a category (dropdown menu)
- Postcarding ID (text input)
- Choose a swap type (dropdown menu)
- Tags (text input with an "Add" button)

At the bottom right are "Reset" and "Save" buttons.

Fonte: Elaborado pelo autor.

Figura 14: Adicionar um postal para troca.



This screenshot is identical to the previous one, showing the same "Add New Postcard" form. However, the "For Swap" tab is now selected, and the "Image" field is no longer visible. The visible fields are:

- Description (text input)
- Choose a country (dropdown menu)
- Available (text input)
- Choose a category (dropdown menu)
- Tags (text input with an "Add" button)

The "Reset" and "Save" buttons remain at the bottom right.

Fonte: Elaborado pelo autor.

Figura 15: Erros nos atributos ao adicionar um postal recebido.

Keep Me Posted

Search

Home Add Collection Favorites Profile Stats Swap

Add New Postcard

Received For Swap

Description
This value length is invalid. It should be between 10 and 250 characters long.

Image
The value is required.

United States

Sender
Date received
15/11/2015

Written and Stamped
The value is required.

Postcrossing ID
Gift
The value is required.

Tags
usa x uk x map x united states x + Add Tag

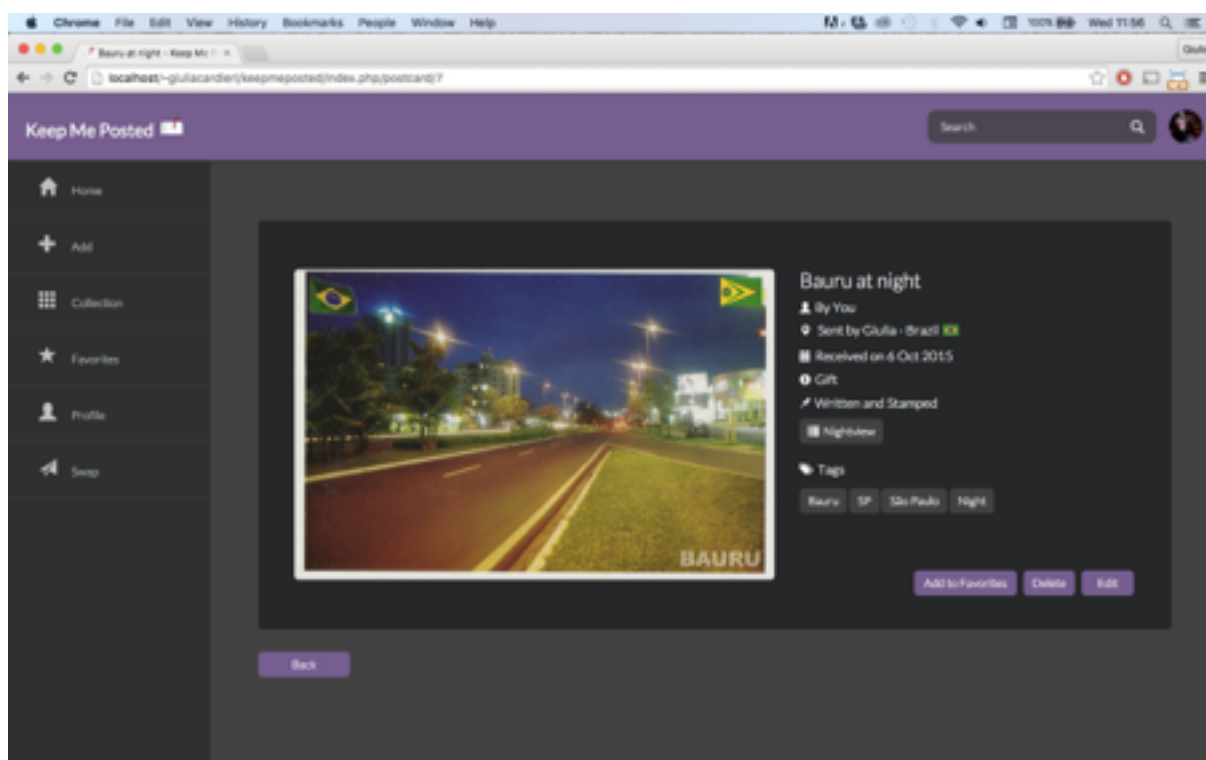
Reset Save

Fonte: Elaborado pelo autor.

4.1.4 Visualização e edição de um postal

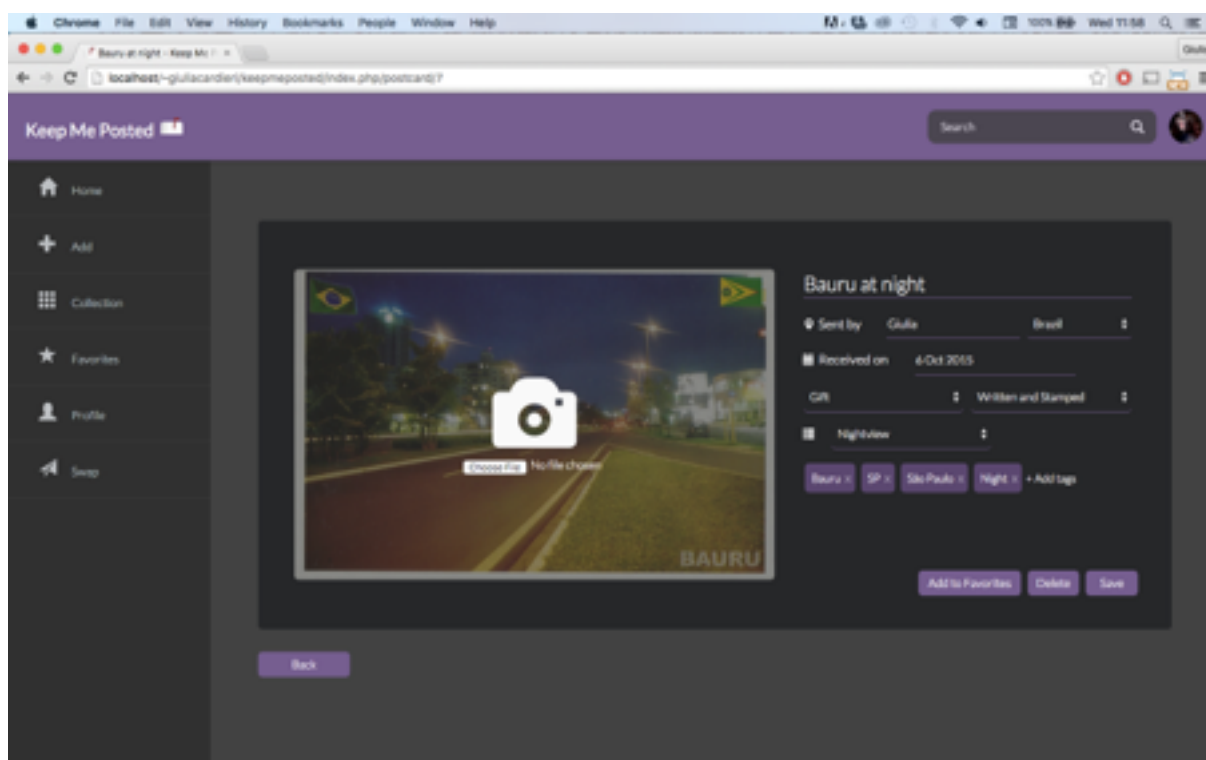
Após adicionar um postal o usuário será redirecionado para a página desse novo postal, como ilustrado na Figura 16. Essa página mostra todas as informações desse postal com *links* para o perfil do usuário que possui esse item, todos os postais de uma determinada categoria e marcados com uma *tag*. Além de exibir o postal, também é possível marcá-lo ou removê-lo da lista de favoritos e editá-lo, caso o usuário da sessão atual seja o usuário que cadastrou o postal. A Figura 17 apresenta a interface criada para a edição de um postal.

Figura 16: Visualização de um postal.



Fonte: Elaborado pelo autor.

Figura 17: Edição de um postal.

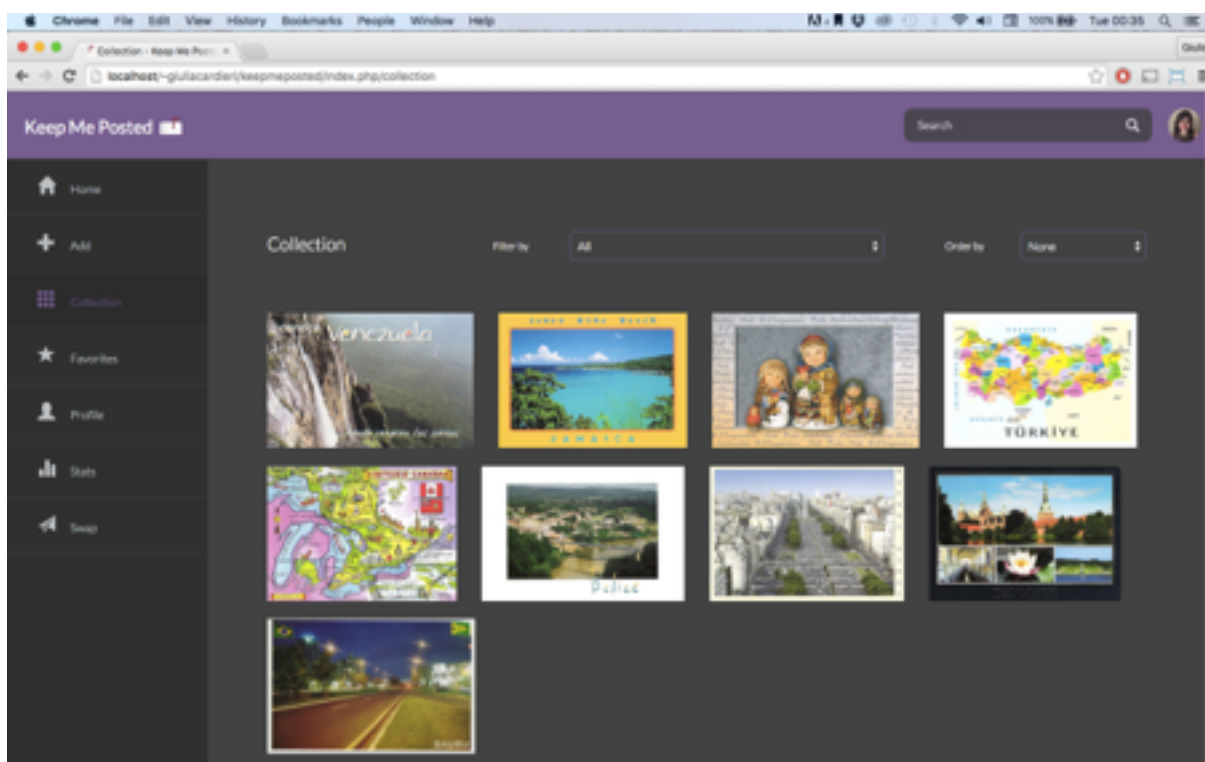


Fonte: Elaborado pelo autor.

4.1.5 Coleção

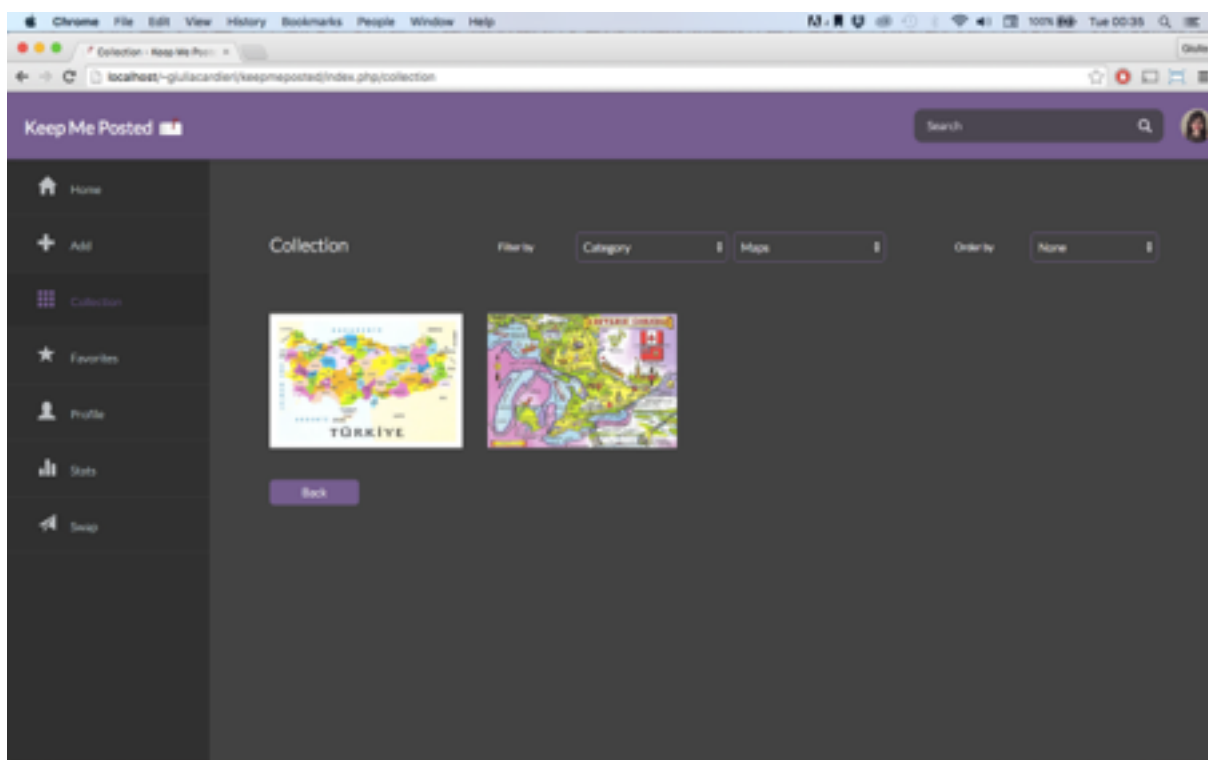
A página *Collection* mostra todos os postais da coleção do usuário da sessão atual, como pode ser observado na Figura 18. É possível filtrar os itens exibidos por uma categoria ou país específico, e ordenar de acordo com a popularidade ou data recebida. A Figura 19 exhibe a página *Collection* filtrada pela categoria *Maps*. Para acessar a página de visualização de um postal, deve-se clicar sobre ele. Ao passar o mouse sobre sua imagem, algumas informações, como a descrição e o *username* do usuário que cadastrou o postal, são exibidas e há um ícone de estrela que é um atalho para marcá-lo como favorito.

Figura 18: Coleção do usuário.



Fonte: Elaborado pelo autor.

Figura 19: Coleção do usuário filtrada pela categoria *Maps*.

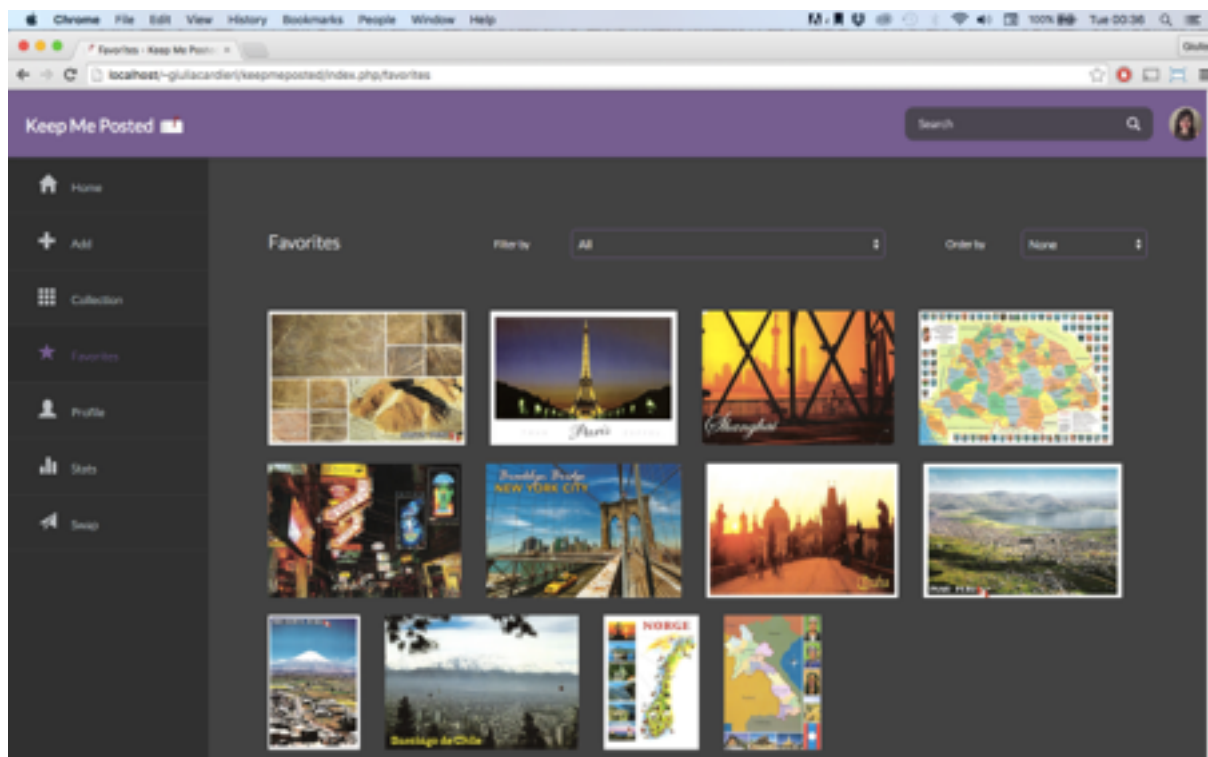


Fonte: Elaborado pelo autor.

4.1.6 Favoritos

A página *Favorites* segue o mesmo padrão da página *Collection*, porém nela serão exibidos somente os postais marcados como favorito pelo usuário. A Figura 20 exhibe a página *Favorites* do usuário da sessão atual.

Figura 20: Postais marcados como favoritos pelo usuário.

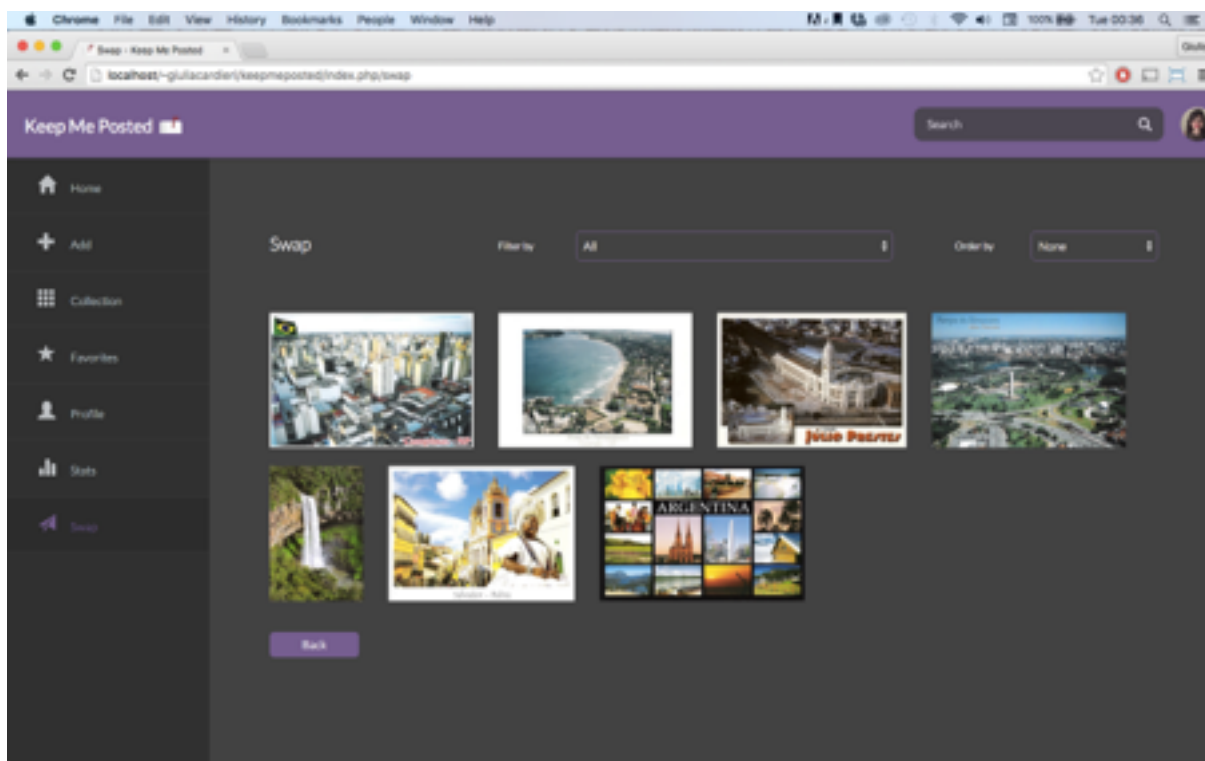


Fonte: Elaborado pelo autor.

4.1.7 Troca (swap)

A página Swap segue o mesmo padrão das páginas *Collection* e *Favorites*, porém os postais exibidos são somente os adicionados como um postal de troca. A Figura 21 apresenta a interface criada para exibir os postais de troca.

Figura 21: Postais de troca do usuário.



Fonte: Elaborado pelo autor.

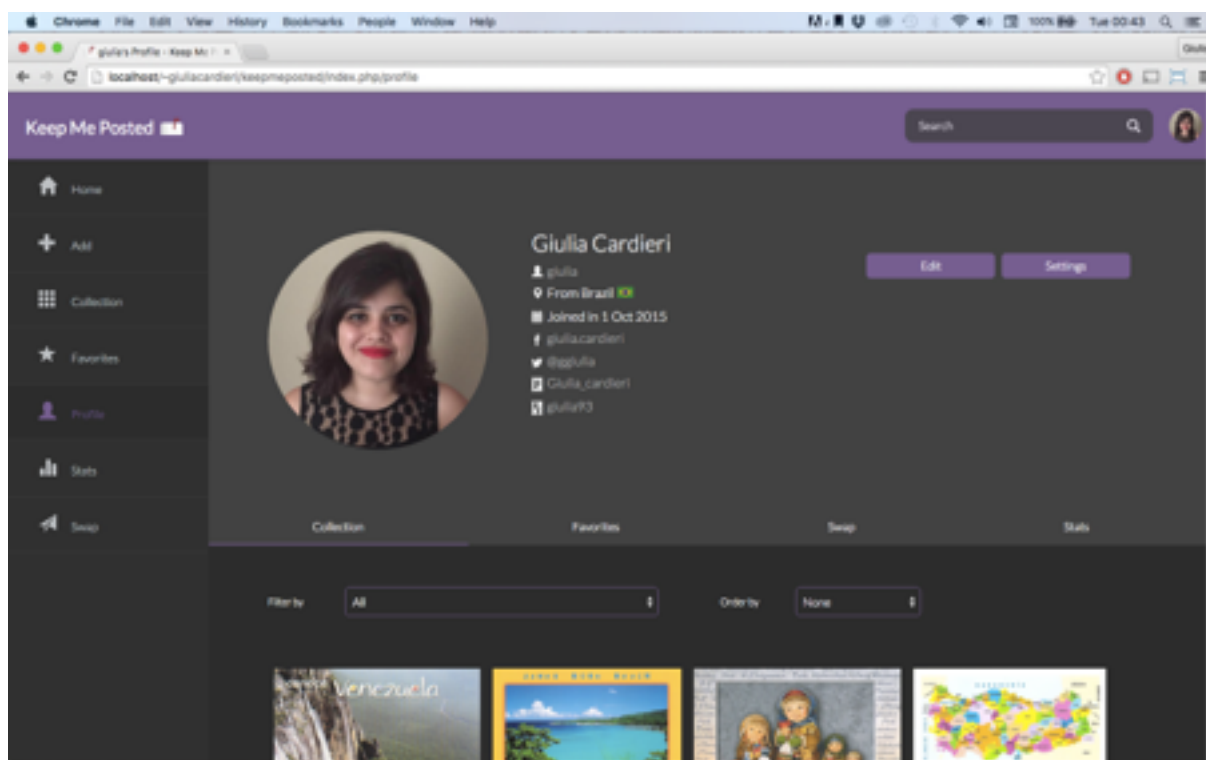
4.1.8 Perfil

O perfil de um usuário é público e pode ser visualizado por qualquer usuário cadastrado. Essa página exibe informações pessoais sobre o colecionador, como nome, *username*, data de cadastro, país e informações de contato, e um resumo das páginas de coleção, favoritos, troca e estatísticas. A Figura 22 exemplifica essa página.

As informações pessoais do usuário podem ser editadas no próprio perfil, incluindo a privacidade dos *links* para outras redes sociais. Essas informações são essenciais para permitir a interação entre colecionadores que não estão cadastrados no Keep Me Posted. A Figura 23 exibe a interface criada para edição de informações do perfil do usuário.

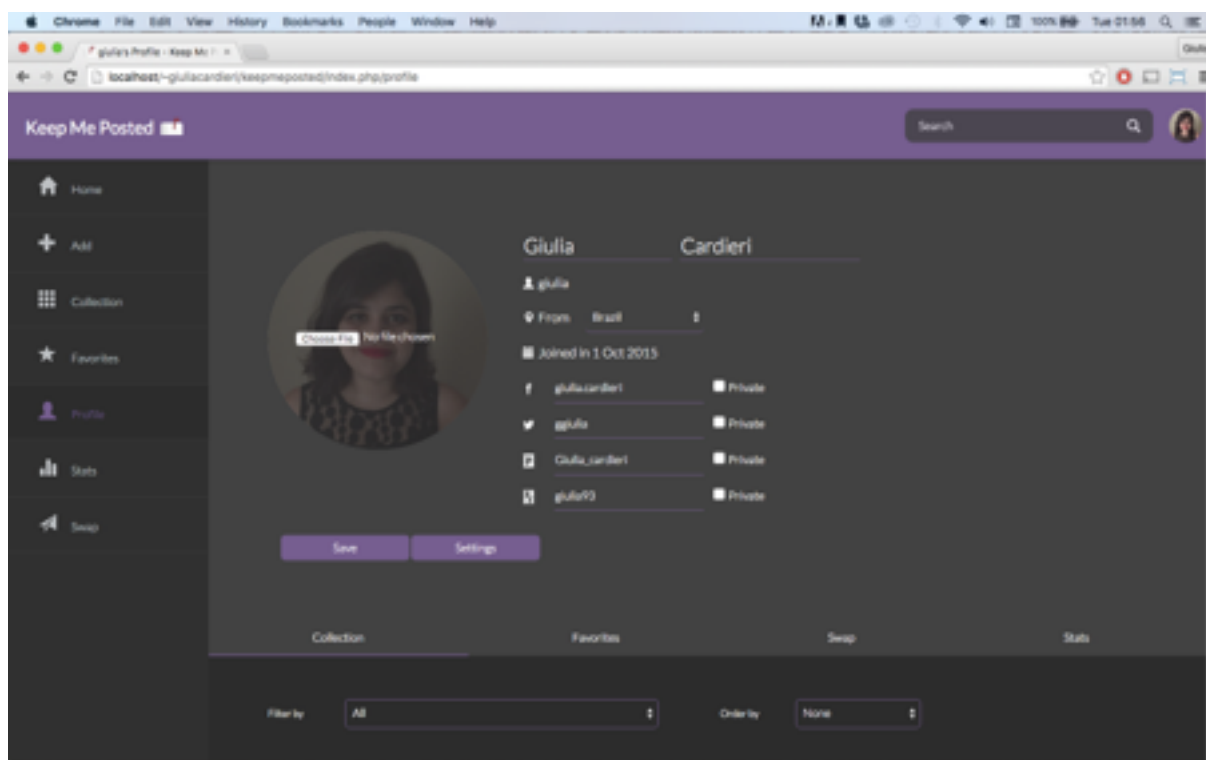
O resumo das páginas de postais e estatísticas é exibido logo abaixo das informações pessoais. Através dessa funcionalidade, os usuários tem acesso a: coleção, favoritos, postais de troca e estatísticas de outros usuários.

Figura 22: Perfil de um usuário.



Fonte: Elaborado pelo autor.

Figura 23: Edição de um perfil de usuário.



Fonte: Elaborado pelo autor.

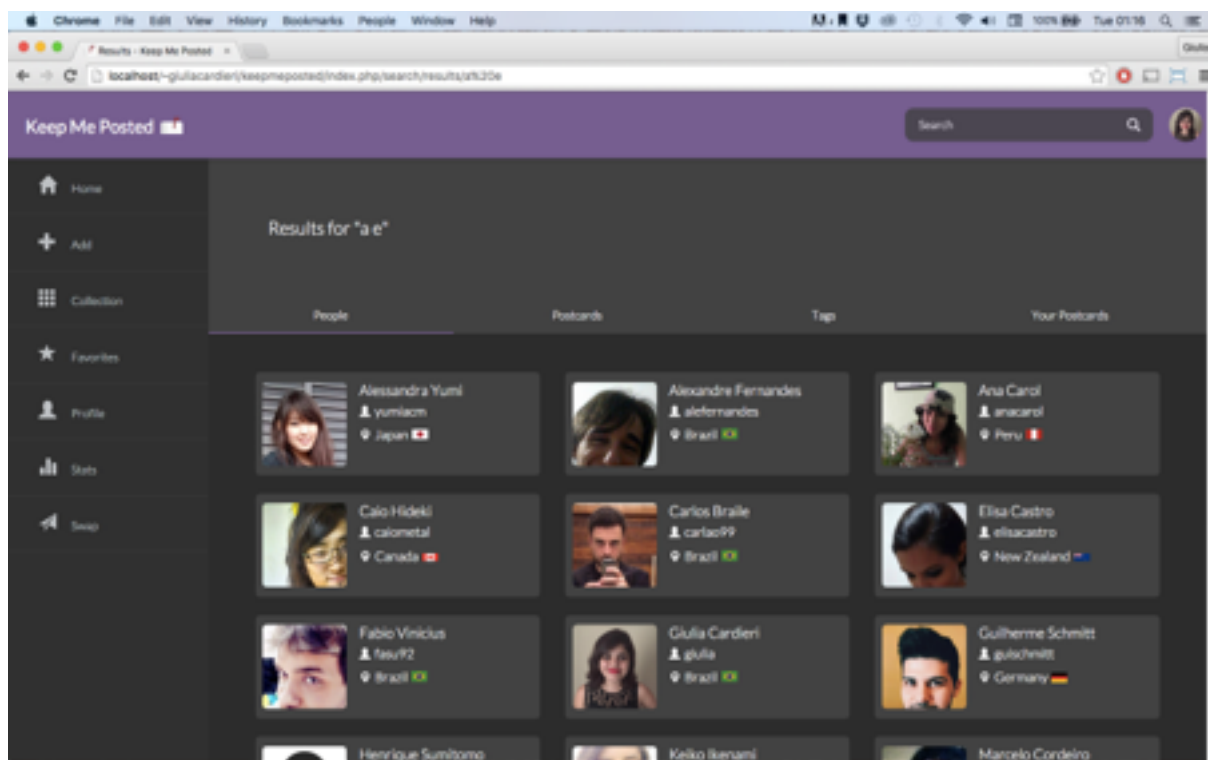
4.1.9 Pesquisa

O sistema desenvolvido possui um motor de busca no cabeçalho de todas as páginas. Essa funcionalidade auxilia os usuários a encontrar postais de tópicos específicos e também a encontrar um usuário por seu *username* ou nome.

Ao realizar uma busca, o usuário é redirecionado para a página de resultados, que são divididos em:

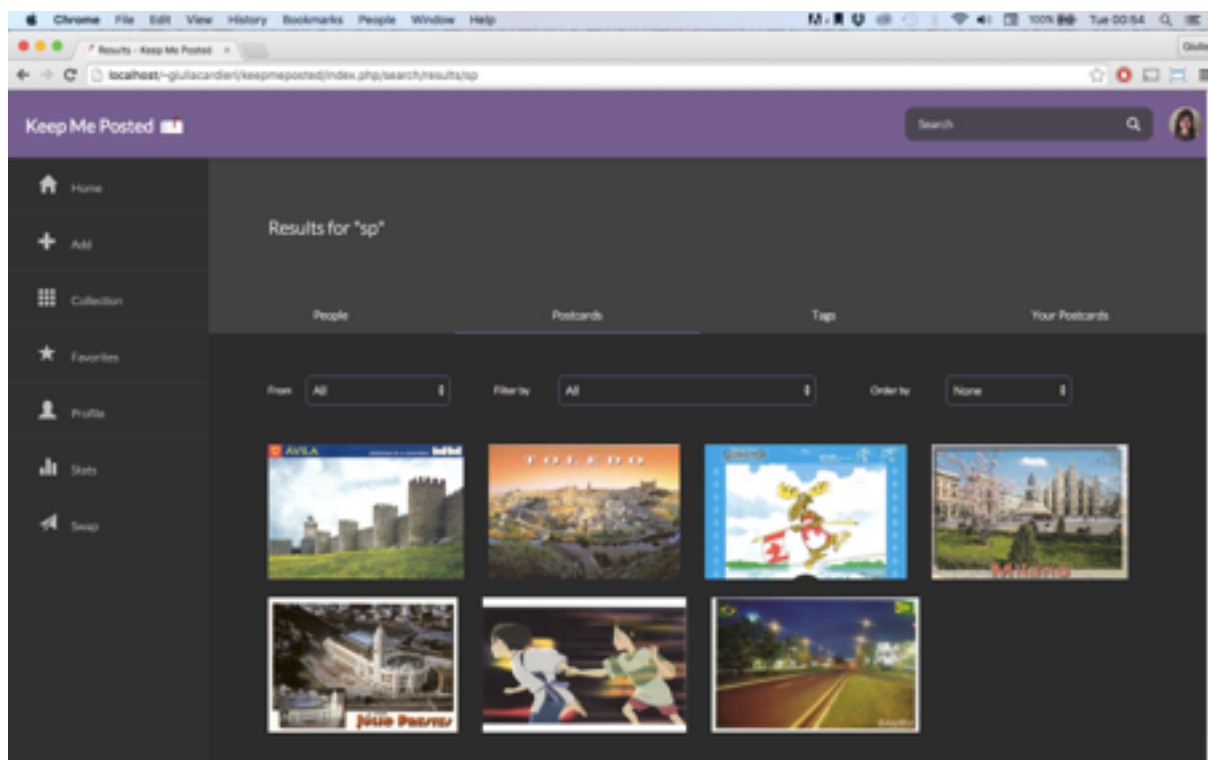
- *People/Pessoas*: retorna resultados de perfis de usuários com o nome ou *username* parecidos com os termos pesquisados. A Figura 24 exemplifica esse tipo de busca. Nessa Figura, é possível observar que, ao pesquisar o termo “a e”, são retornados todos os usuários que possuem, em seu nome completo ou *username*, uma dessas letras.
- *Postcards/Postais*: retorna postais com algum atributo similar aos termos pesquisados. A Figura 25 exibe a interface desenvolvida para esse tipo de busca. Nessa Figura, pode-se observar que, ao pesquisar o termo “sp”, retornam-se postais que possuem essas letras em algum de seus atributos, como tags, descrição, remetente, Postcrossing ID e país.
- *Tags*: retorna *tags* similares aos termos pesquisados. A Figura 26 exibe um exemplo com essa busca, e ao pesquisar o termo “sp”, o resultado são todas as tags que possuem essas letras em seu nome.
- *Your Postcards/Seus Postais*: retorna postais somente da coleção do usuário que realizou a busca. O objetivo dessa busca é permitir ao usuário realizar pesquisas em sua própria coleção e seus postais para troca. A Figura 27 apresenta um exemplo dessa busca. Diferente do exemplo da Figura 25, ao pesquisar o termo “sp”, retornam-se somente os postais da coleção do usuário que realizou a busca.

Figura 24: Busca por usuários.



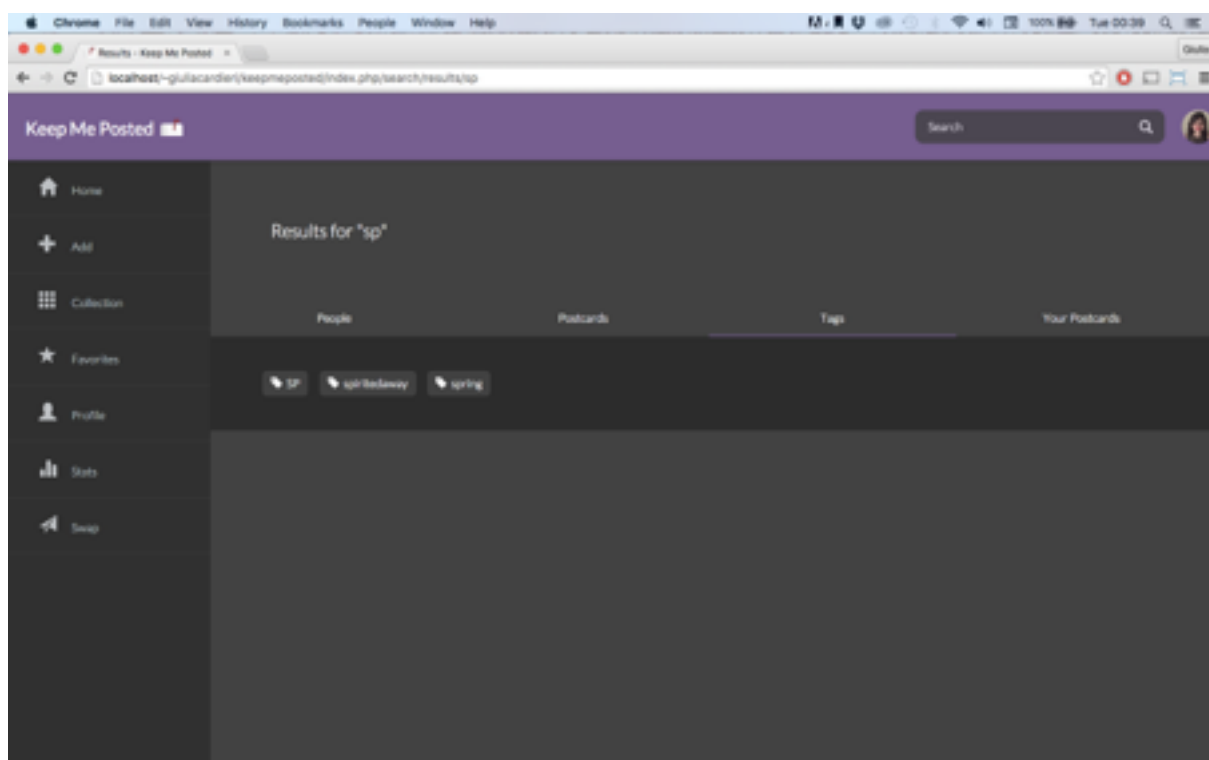
Fonte: Elaborado pelo autor.

Figura 25: Busca por postais.



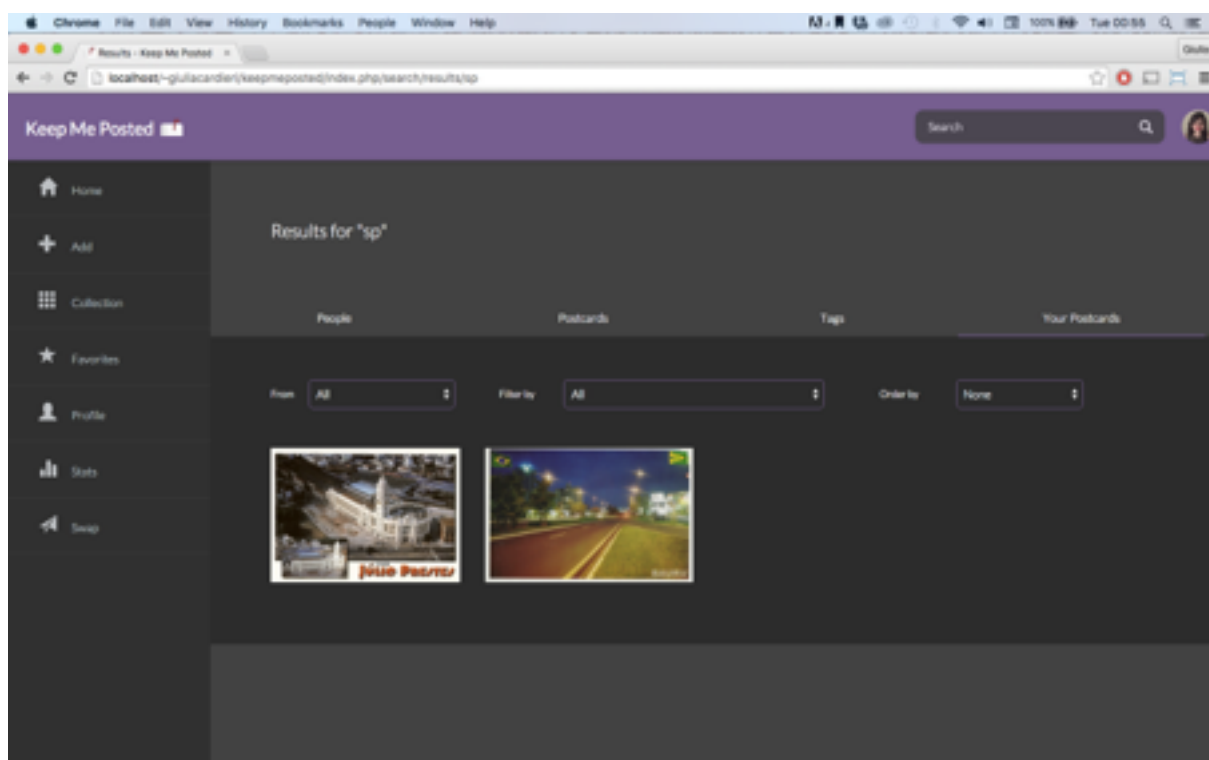
Fonte: Elaborado pelo autor.

Figura 26: Busca por tags.



Fonte: Elaborado pelo autor.

Figura 27: Busca por postais na coleção do usuário.

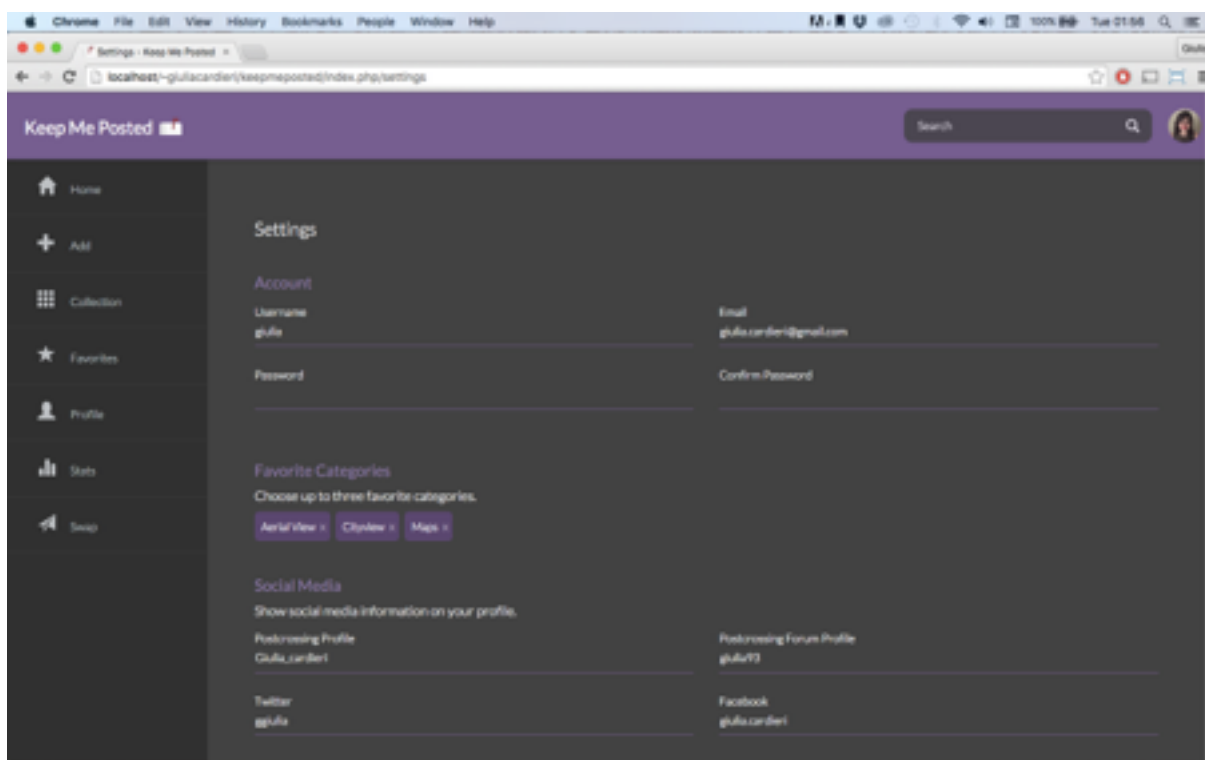


Fonte: Elaborado pelo autor.

4.1.10 Configurações

O usuário pode alterar configurações de seu cadastro, informações de contato, privacidade, notificações e deletar sua conta na página de configurações (*settings*). Essa página pode ser acessada utilizando o menu localizado ao clicar na foto do usuário no cabeçalho de qualquer página ou pelo botão *settings* no perfil. A Figura 28 exemplifica essa página.

Figura 28: Configurações da conta do usuário.



Fonte: Elaborado pelo autor.

4.1.11 Estatísticas

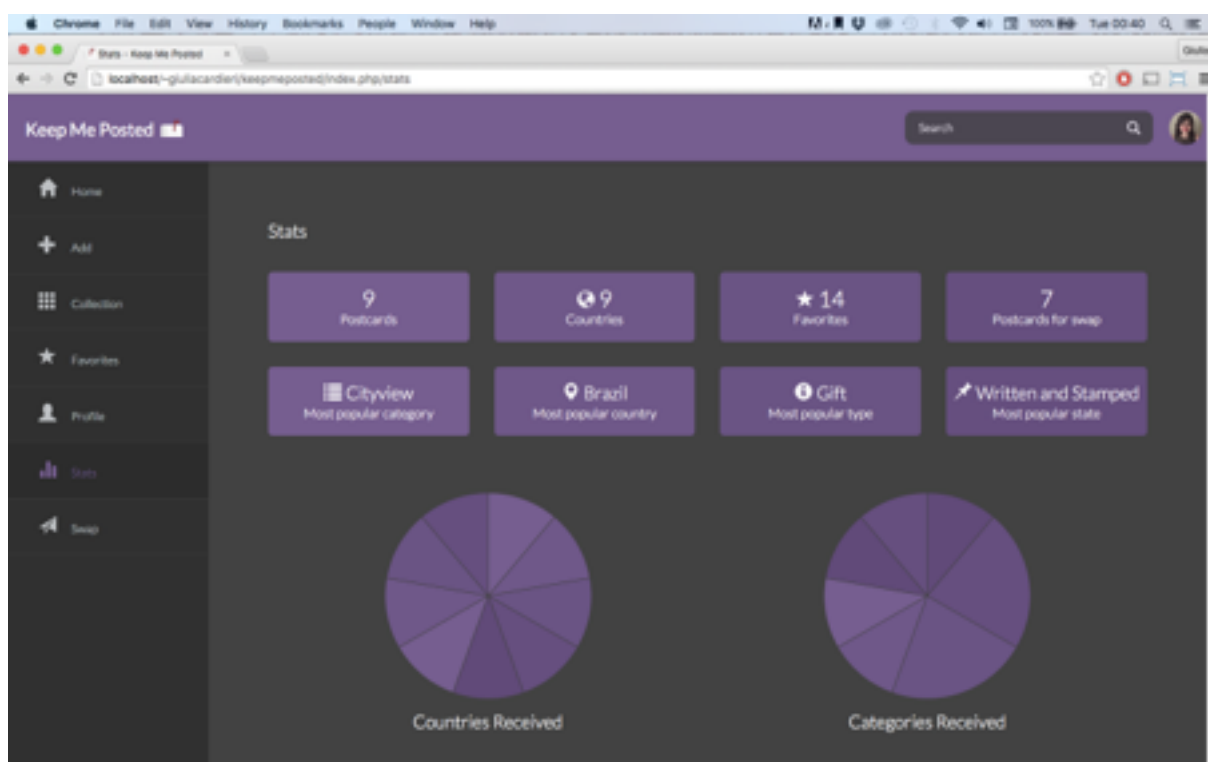
O sistema desenvolvido possui uma página com estatísticas sobre a coleção dos usuários. Esse conteúdo está disponível na página *Stats*, acessível pelo menu principal ou pelo menu secundário nas páginas de perfil dos usuários. A Figura 29 exhibe a página de estatísticas criada nesse projeto.

O conteúdo exibido nessa página gera um relatório sobre a condição atual da coleção de determinado usuário, exibindo a quantidade de postais na coleção, a quantidade de postais disponíveis para troca, favoritos, o número de países

diferentes na coleção, e o país, categoria, tipo e estado mais populares na coleção. Esses dados são exibidos em forma de texto e gráficos.

O objetivo dessa página é auxiliar o usuário a administrar sua coleção, exibindo dados precisos gerados automaticamente. Com esses dados, outros colecionadores podem entender os gostos de determinado usuário e o colecionador pode definir quais os próximos passos para ampliar sua coleção.

Figura 29: Estatísticas da coleção do usuário.



Fonte: Elaborado pelo autor.

4.2 Implementação do Sistema de Recomendação

A implementação do sistema de recomendação foi realizada utilizando diferentes filtros nas *queries* do banco de dados. Esses filtros foram definidos de acordo com uma análise de como um colecionador escolhe os postais que gostaria de ter. As recomendações foram divididas entre populares e individuais.

As recomendações populares utilizam um filtro não personalizado, pois analisam os dados de todos os usuários para gerar um padrão dos postais que agradam a maioria dos usuários, e os mesmos itens são exibidos para todos os usuários. As informações necessárias para esse tipo de recomendação foram

adquiridas de forma implícita através da análise de todo o conteúdo do banco de dados. Através deste tipo de aquisição é possível determinar os postais pelos quais a maioria dos usuários se interessou, bem como indicá-los para os colecionadores que ainda não encontraram esses itens. Essa decisão foi tomada para que os colecionadores possam encontrar postais que poderiam ser diferentes do padrão de suas coleções.

O filtro utilizado nas recomendações populares funciona da seguinte maneira: os postais são ordenados em ordem decrescente pelo atributo *favorite_count*, e a seguir são selecionados os postais com as categorias e países mais populares, ou seja, que possuem mais postais. A partir dessa seleção, os primeiros 40 postais serão exibidos para todos os usuários. Esse filtro foi definido a partir da forma como os colecionadores de postais buscam um postal no fórum do *website Postcrossing*.

As recomendações individuais utilizam a técnica da filtragem baseada em conteúdo, determinando os postais que serão recomendados após analisar os dados das ações previamente realizadas por determinado usuário. Essa decisão foi tomada devido à natureza dos itens recomendados. Não é comum encontrar colecionadores com gostos similares, portanto, recomendar postais baseando-se no conteúdo que o próprio colecionador já forneceu ou dos postais que esse indivíduo marcou como favorito é uma abordagem com mais chances de sucesso.

5. CONCLUSÃO

Conforme definido na introdução, este projeto teve como objetivo desenvolver uma aplicação *web* voltada para colecionadores de cartões postais, na forma de uma rede social com um sistema de recomendação, sendo por seu propósito auxiliar os usuários a organizar suas coleções e interagir com outros colecionadores de interesses similares.

Foram estudados diferentes conceitos e tecnologias, com o foco principal em sistemas de recomendação. O conhecimento obtido foi essencial para a definição dos materiais e métodos que melhor se encaixam no contexto dessa aplicação. As escolhas dos materiais foram justificadas no Capítulo 4.

Um fator importante no desenvolvimento dessa aplicação foi o estudo do *website Postcrossing*. Entender como os colecionadores organizam suas coleções digitalmente e planejar a forma como esses indivíduos gostariam de receber recomendações, colaborou para um maior entendimento da implementação de um sistema de recomendação.

O estudo de sistemas de recomendação foi um grande aprendizado, e foi o conceito mais estudado para esse projeto, o que foi fundamental para desenvolver um sistema eficiente. A recomendação de cartões postais tem diversos fatores diferentes de um sistema de recomendação de filmes ou livros, os quais são amplamente citados nas referências utilizadas.

Concluiu-se, portanto, que o projeto atingiu todos os objetivos propostos e espera-se que ele seja capaz de auxiliar colecionadores de cartões postais a organizar suas coleções, interagir com outros usuários e encontrar postais que atendam seus gostos com mais facilidade.

5.1 Trabalhos Futuros

Como trabalhos futuros propõem-se melhorias e novas funcionalidades, que embora não necessárias para o funcionamento correto dessa aplicação poderiam trazer benefícios aos usuários tais como

1. Aumentar a interação entre usuários;

2. Melhorias na usabilidade das páginas;
3. Melhorias no motor de busca, trazendo resultados mais precisos utilizando uma ferramenta como a Elastic Search;
4. Implementar *login* através de redes sociais como Twitter e Facebook;
5. Criar notificações quando algum postal do usuário da sessão atual é marcado como favorito por outro usuário;
6. Criação de um aplicativo para aparelhos móveis e;
7. Criação de versões da aplicação em diferentes idiomas.

REFERÊNCIAS

ADOMAVICIUS, Gediminas; TUZHILIN, Alexander. **Toward the Next Generation Of Recommender Systems: A Survey of State-of-the-Art and Possible Extensions**. IEEE Transactions on Knowledge and Data Engineering, v.17, n.6, 734-749, Junho 2005. Disponível em <<http://web.stanford.edu/class/ee378b/papers/adomavicius-recsys.pdf>>. Acesso em: 19 de maio de 2015

BARCELLOS, Carla D.; MUSA, Daniela L.; BRANDÃO, André L.; WARPECHOWSKI, Mariusa. **Sistema de Recomendação Acadêmico para Apoio a Aprendizagem**. RENOTE - Revista Novas Tecnologias na Educação, Porto Alegre, v.5, n.2, Dezembro 2007. Disponível em: <<http://www.cinted.ufrgs.br/ciclo10/artigos/3fDaniela.pdf>>. Acesso em: 21 de Julho de 2015.

BENNETT, Shea. **67% Of Americans Use Social Media (With One In Six Active On Twitter) [STUDY]**. Abril 2014. Disponível em: <<http://www.adweek.com/socialtimes/social-media-america/497615>>. Acesso em 19 de maio de 2015.

BOOTSTRAP. **Bootstrap**. 2015. Disponível em: <<http://getbootstrap.com/>>. Acesso em 5 de Julho de 2015.

CAZELLA, Sílvio César; NUNES, Maria Augusta S. N.; REATEGUI, Eliseo Berni. **A Ciência da Opinião: Estado da arte em Sistemas de Recomendação**. In: XXX Congresso da SBC Jornada de Atualização da Informática, 2010, Belo Horizonte. Disponível em: <<http://200.17.141.213/~gutanunes/hp/publications/JAI4.pdf>>. Acesso em: 17 de maio de 2015.

CODEIGNITER. **CodeIgniter Web Framework**. Disponível em: <<https://codeigniter.com/>>. Acesso em: 17 de novembro de 2015.

DAVID, Natalia. **11 Reasons to Use Twitter Bootstrap**. Janeiro 2013. Disponível em: <<http://www.sitepoint.com/11-reasons-to-use-twitter-bootstrap/>>. Acesso em 25 de novembro de 2015.

EIS, Diego. **O básico: O que é HTML?**. Janeiro 2011. Disponível em: <<http://tableless.com.br/o-que-html-basico/>> . Acesso em 28 de Junho de 2015.

FIGUEIRA FILHO, Fernando M. De; GEUS, Paulo Lício De; ALBUQUERQUE, João Porto. **Sistemas de Recomendação e Interação na Web Social**. In: I Workshop de Aspectos da Interação Humano-Computador na Web Social, 2008, Porto Alegre. Porto Alegre: SBC, 2008, p. 24-27. Disponível em: <http://www.ic.unicamp.br/~fmarques/papers/websocial_ihc08.pdf>. Acesso em: 17 de maio de 2015.

FLICKR. **Sobre o Flickr**. Disponível em: <<https://www.flickr.com/about>>. Acesso em 19 de maio de 2015.

GIMMER, Christopher. **Top 5 Reasons to use Bootstrap**. Abril 2014. Disponível em <<https://bootstrapbay.com/blog/reasons-to-use-bootstrap/>>. Acesso em 25 de novembro de 2015.

GOLDBERG, David; NICHOLS, David; OKI, Brian M.; TERRY, Douglas. **Using collaborative filtering to weave an information Tapestry**. Communications of the ACM, ACM, v.35, n.13, p. 161, 1992. Disponível em <https://www.ischool.utexas.edu/~i385d/readings/Goldberg_UsingCollaborative_92.pdf>. Acesso em 18 de maio de 2015.

GOOD, Nathaniel; SCHAFER, J. Ben; KONSTAN, Joseph A.; BORCHERS, Al; SARWAR, Badrul; HERLOCKER, Jon; RIEDL, John. **Combining Collaborative Filtering with Personal Agents for Better Recommendations**. Proceedings of AAAI, Proceedings, v. 35, p. 439-446, 1999. Disponível em: <<http://files.grouplens.org/papers/aaai-99.pdf>>. Acesso em 19 de maio de 2015.

HERLOCKER, Jonathan L.; KONSTAN, Joseph A.; RIEDL, John. **Explaining Collaborative Filtering Recommendations**. In: CSCW '00 Proceedings of the 2000 ACM conference on Computer supported cooperative work, 2000, Minneapolis. New York: ACM, 2000, p. 241-250. Disponível em <<http://recsys.cs.umn.edu/docs/explain-CSCW-2000.pdf>>. Acesso em 18 de maio de 2015.

MAGALHÃES, Paulo. **Postcrossing History**. Disponível em: <<http://www.postcrossing.com/about/history>>. Acesso em: 8 de maio de 2015. A.

_____. **About Postcrossing**. Disponível em: <<http://www.postcrossing.com/about>>. Acesso em: 19 de maio de 2015. B.

_____. **User Stats**. Disponível em: <<https://www.postcrossing.com/stats/users>>. Acesso em: 8 de janeiro de 2016.

MUSETH, Tom. **An Introduction to CodeIgniter**. Abril 2011. Disponível em: <<http://www.sitepoint.com/an-introduction-to-codeigniter/>>. Acesso em 18 de novembro de 2015.

NATIONS, Daniel. **Web Applications: What is a Web Application?**. Dezembro 2014. Disponível em: <http://webtrends.about.com/od/webapplications/a/web_application.htm>. Acesso em 19 de maio de 2015.

NIXON, Robert. **Learning PHP, MySQL, JavaScript, CSS & HTML5**. 3. ed. Sebastopol: O'Reilly Media, Inc., Junho 2014. 700 p.

ORACLE CORPORATION. **MySQL 5.1 Reference Manual :: 1 General Information :: 1.3 Overview of the MySQL Database Management System :: 1.3.1 What is MySQL?**. 2015. Disponível em <<https://dev.mysql.com/doc/refman/5.1/en/what-is-mysql.html>>. Acesso em 21 de Julho de 2015. A.

_____. **Top Reasons to Use MySQL**. 2015. Disponível em <<https://www.mysql.com/why-mysql/topreasons.html>>. Acesso em 29 de Novembro de 2015. B.

RAMOS, Alan; **MVC - Afinal, o que é?**. Tableless. Fevereiro 2015. Disponível em: <<http://tableless.com.br/mvc-afinal-e-o-que/>>. Acesso em 17 de Novembro de 2015.

REATEGUI, Eliseo B.; CAZELLA, Sílvia C.; **Sistemas de Recomendação**. In: XVV Congresso da Sociedade Brasileira de Computação, 2005, São Leopoldo. Porto Alegre: SBC, 2005, p. 306-348. Disponível em <<http://www.cin.ufpe.br/~fab/aulas-RI/Sistemas-de-Recomendacao.pdf>>. Acesso em 16 de maio de 2015.

RESNICK, Paul; VARIAN, Hal R. **Recommender Systems**. Communications of the ACM, New York, v.40, n.3, p. 55-58, 1997. Disponível em: <https://www.ischool.utexas.edu/~i385d/readings/Resnick_Recommender_97.pdf>. Acesso em: 18 de maio de 2015.

SASS. **Sass: Sass Basics**. 2015. Disponível em: <<http://sass-lang.com/guide>>. Acesso em 5 de Julho de 2015.

SCHAFER, J. B.; KONSTAN, Joseph; RIEDL, John. **Recommender Systems in E-Commerce**. In: EC '99 Proceedings of the 1st ACM conference on Electronic commerce, 1999, Minneapolis. Nova Iorque: ACM, 1999, p. 158-166. Disponível em: <<http://files.grouplens.org/papers/ec-99.pdf>>. Acesso em 19 de maio de 2015.

TABLELESS. **Para Iniciantes**. 2013. Disponível em: <<http://tableless.github.io/iniciantes/manual/js/index.html>>. Acesso em 21 de Julho de 2015.

THE CORE LESS TEAM. **Getting started | Less.js**. 2016. Disponível em: <<http://lesscss.org/>>. Acesso em 8 de Janeiro de 2016.

THE JQUERY FOUNDATION. **jQuery**. 2015. Disponível em <<https://jquery.com/>>. Acesso em 21 de Julho de 2015.

THE PHP GROUP. **PHP: O que é?**. 2015. Disponível em <http://php.net/manual/pt_BR/intro-what-is.php>. Acesso em 21 de Julho de 2015.

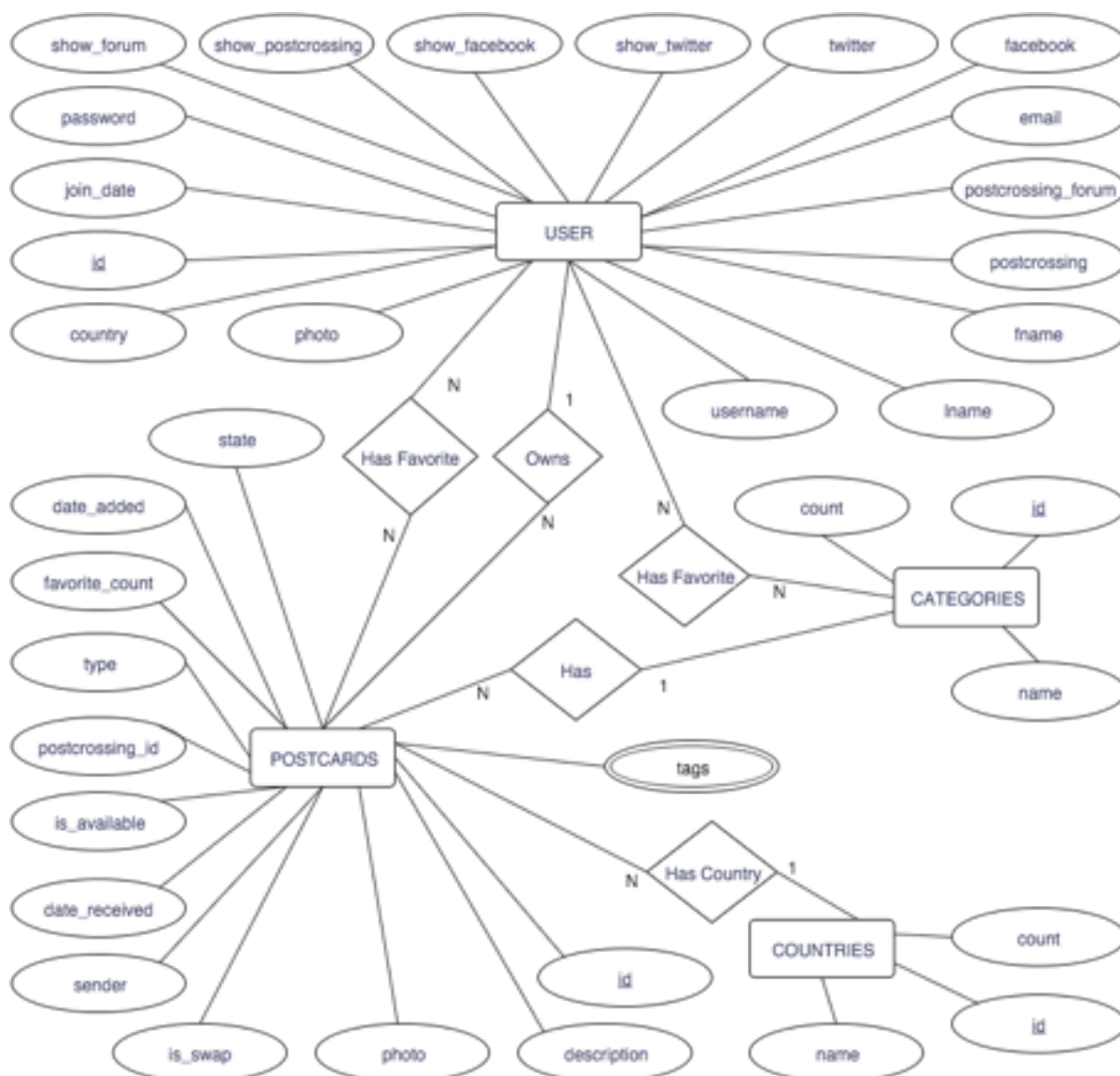
TOMAÉL, Maria Inês; ALCARÁ, Adriana Rosecler; DI CHIARA, Ivone Guerreiro. **Das redes sociais à inovação**. Brasília, v.34, n.2, 93-104, Maio/Agosto 2005. Disponível em: <<http://www.scielo.br/pdf/ci/v34n2/28559>>. Acesso em: 11 de maio de 2015.

W3 SCHOOLS. **HTML Introduction**. Disponível em: <http://www.w3schools.com/html/html_intro.asp>. Acesso em 28 de Junho de 2015.

_____. **jQuery Introduction**. Disponível em: <http://www.w3schools.com/jquery/jquery_intro.asp>. Acesso em 8 de Janeiro de 2016.

W3C. **Cascading Style Sheets**. Disponível em: <<http://www.w3.org/Style/CSS/>>. Acesso em 8 de Janeiro de 2016.

APÊNDICE A - Modelo Entidade Relacionamento



Nesse modelo entidade relacionamento são apresentadas as tabelas definidas para essa aplicação no banco de dados. Elas são *User*, *Postcards*, *Countries*, *Categories*, *User_favorite_category* e *User_favorite_postcard*.

A tabela *User* armazena todos os dados referentes ao usuário. Informações como o primeiro nome (*fname*), último nome (*lname*), *username*, foto (*photo*), *email*, país (*country*), senha (*password*), contato em redes sociais (*twitter*, *facebook*, *postcrossing* e *postcrossing_forum*) e a privacidade dos *links* para contato em redes sociais (*show_facebook*, *show_twitter*, *show_forum* e *show_postcrossing*) são

obtidos explicitamente por ações do usuário, como explicado na Seção 4.1.10. Alguns atributos, como a chave primária *id* e a data de cadastro (*join_date*) são obtidos automaticamente no cadastro do usuário.

A tabela *Postcards* armazena as informações de um cartão postal. Há dois tipos diferentes de postais que são armazenados: os postais de coleção, recebidos pelo usuário, e os postais para troca, que o usuário pretende trocar com outros colecionadores. Se o postal foi recebido pelo usuário, ele é um postal da coleção, e portanto o atributo *is_swap* é automaticamente configurado para zero. Caso o postal seja de troca, esse atributo recebe o valor um.

Um postal de troca recebe o valor nulo em alguns atributos como: remetente (*sender*), estado (*state*), *postcrossing_id*, tipo (*type*) e data recebida (*date_received*). Isso acontece pois esses atributos não são aplicáveis a um postal de troca. Um postal de coleção também recebe o valor nulo no atributo *is_available* que, conforme indicado no Quadro 2 na Seção 4.1.3, representa a disponibilidade para troca de um postal, o que não se aplica para postais de coleção.

Os atributos descrição (*description*), foto (*photo*) e *tags* são necessários para os dois tipos de postais (troca e coleção) e são obtidos através de ações do usuário na página de adicionar um postal, conforme explicado na Seção 4.1.3. Alguns atributos são obtidos automaticamente ao adicionar um postal, sendo eles a chave primária *id*, que é auto-incrementada, a data de quando o postal foi adicionado (*date_added*) e a contagem de quantas pessoas marcaram como favorito o postal (*favorite_count*), que inicialmente é zero.

A tabela *Postcards* possui três chaves estrangeiras. O atributo *user_id* é relativo ao usuário que cadastrou o postal, o atributo *category_id* indica a categoria à qual o postal pertence, e o atributo *country* representa o país desse postal. Já a tabela *Categories* possui todas as categorias existentes, indicadas no Quadro 1 na Seção 4.1.3. Além dos atributos como o nome da categoria (*name*) e a chave primária *id*, essa tabela também possui o *count*, com o número de postais cadastrados em todo sistema de determinada categoria.

A tabela *Countries* foi criada com o propósito de armazenar a lista de países disponíveis nessa aplicação. Assim como a tabela *Categories*, ela possui os

atributos nome (*name*), a chave primária *id* e o *count*, com o número de postais cadastrados em todo sistema de um determinado país.

A tabela *user_favorite_category* é criada a partir do relacionamento muitos-para-muitos entre as tabelas *User* e *Categories*. Essa tabela armazena as categorias favoritas dos usuários. Os atributos dessa tabela são as chaves estrangeiras *user_id* e *category_id*.

A tabela *user_favorite_postcard* é criada a partir do relacionamento muitos-para-muitos entre as tabelas *User* e *Postcards*. Essa tabela armazena os postais marcados como favorito pelos usuários. Os atributos dessa tabela são as chaves estrangeiras *user_id* e *postcard_id*.