

UNIVERSIDADE FEDERAL FLUMINENSE

RODRIGO DE ALMEIDA CARDOSO

**MYVOYAGE: SISTEMA WEB / TELEGRAM PARA ARMAZENAMENTO E
COMPARTILHAMENTO DE IMAGENS**

NITERÓI

2022

RODRIGO DE ALMEIDA CARDOSO

**MYVOYAGE: SISTEMA WEB / TELEGRAM PARA ARMAZENAMENTO E
COMPARTILHAMENTO DE IMAGENS**

Trabalho de Conclusão de Curso
submetido ao Curso de Tecnologia em
Sistemas de Computação da
Universidade Federal Fluminense como
requisito parcial para obtenção do título
de Tecnólogo em Sistemas de
Computação.

Orientador: Nilson Luís Damasceno

**NITERÓI
2022**

Ficha catalográfica automática - SDC/BEE
Gerada com informações fornecidas pelo autor

C268m Cardoso, Rodrigo de Almeida
MyVoyage: Sistema Web/Telegram para Armazenamento e
Compartilhamento de Imagens / Rodrigo de Almeida Cardoso. -
2022.
75 f.: il.

Orientador: Nilson Luís Damasceno.
Trabalho de Conclusão de Curso (graduação)-Universidade
Federal Fluminense, Instituto de Computação, Niterói, 2022.

1. Site da Web. 2. Imagem. 3. Mensagem instantânea. 4.
Compartilhamento. 5. Produção intelectual. I. Damasceno,
Nilson Luís, orientador. II. Universidade Federal Fluminense.
Instituto de Computação. III. Título.

CDD - XXX

Bibliotecário responsável: Debora do Nascimento - CRB7/6368

Dedico este trabalho aos meus queridos
filhos Henrique, Nicholas e Charlotte.

AGRADECIMENTOS

Agradeço à minha esposa Fabiola pelo suporte e dedicação enquanto tive que me isolar para poder realizar este trabalho.

Agradeço ao orientador Nilton pela paciência em corrigir meus textos e figuras, e em especial pela sua excepcional capacidade de corrigir bugs.

“Qualquer produto que precise de um manual está malfeito”.
Elon Musk

RESUMO

Este trabalho apresenta o sistema MyVoyage, que é um website onde usuários armazenam suas fotos na nuvem, as compartilham com outros usuários, criam álbum de fotos favoritas e escrevem comentários. O diferencial em relação aos websites de compartilhamento e exibição de imagens é a possibilidade de fazer *upload* de fotos através de um *bot* para Telegram, chamado de MyVoyagebot.

Palavras-chave: fotos, nuvem, banco de dados, bot, Telegram

ABSTRACT

This work presents the MyVoyage system, which is a website where users keep their photos in the cloud, share them with other users, create an album of favorite photos and write comments. The differential in relation to other image sharing and viewing websites is the possibility of uploading photos using a Telegram bot, called MyVoyagebot.

Keywords: photos, cloud, database, bot, Telegram

LISTA DE FIGURAS

Figura 1 - Exemplo de arquitetura de um bot.....	20
Figura 2 - Visualização de imagem no Flickr.....	24
Figura 3 – Visualização de foto vencedora de quest no 500px.....	25
Figura 4 - Imagens de terceiros no Deviant Art.....	26
Figura 5 - Portfolio próprio no SmugMug.....	27
Figura 6 - Página da África do Sul no TrekEarth.....	28
Figura 7 - Portfolio próprio no Google Photos.....	29
Figura 8 - Shutterfly.....	30
Figura 9 - Diagrama de Casos de Uso do sistema MyVoyage.....	37
Figura 10 - Diagrama Entidade-Relacionamento do sistema MyVoyage.....	41
Figura 11 - Máquina de estados finitos do bot.....	42
Figura 12 - Arquitetura do sistema MyVoyage.....	45
Figura 13 - Login.....	48
Figura 14 - Usuário não encontrado e/ou senha incorreta.....	48
Figura 15 - Cadastro de novo usuário.....	49
Figura 16 - Tela inicial do administrador.....	49
Figura 17 - Arquivo carregado com sucesso.....	50
Figura 18 - Arquivo não está no formato aceitável.....	51
Figura 19 - Portfolio.....	51
Figura 20 - Tela de visualização de imagem própria.....	52
Figura 21 - Confirmação para apagar imagem.....	53
Figura 22 - Comentários na imagem.....	54
Figura 23 - Tela de visualização de imagem de terceiros.....	55
Figura 24 - Busca de usuários.....	55
Figura 25 - Resultado da busca de usuários.....	56
Figura 26 - Lista de usuários da MyVoyage.....	57
Figura 27 - Confirmação para apagar usuário.....	57
Figura 28 - Imagem enviada sem cadastro do ID_Telegram.....	58
Figura 29 - Bot pergunta o nome do usuário.....	59
Figura 30 - Nome do usuário não encontrado.....	60
Figura 31 - Bot pede a senha.....	61

Figura 32 - Senha digitada é inválida.....	62
Figura 33 - Cadastro do ID_Telegram realizado.....	63
Figura 34 - Envio de texto após cadastro do ID_Telegram.....	64
Figura 35 - Imagem salva no portfolio pelo MyVoyagebot.....	65
Figura 36 - Formato inválido.....	66

LISTA DE TABELAS

Tabela 1 - Comparativo entre sistemas de armazenamento de imagens.....	31
Tabela 2 - Requisitos funcionais do sistema MyVoyage.....	34
Tabela 3 - Requisitos não-funcionais do sistema MyVoyage.....	35
Tabela 4 - Caso de uso 02.....	37
Tabela 5 - Caso de uso 03.....	37
Tabela 6 - Caso de uso 08.....	38
Tabela 7 - Caso de uso 12.....	38

LISTA DE ABREVIATURAS E SIGLAS

HTML – Hypertext Markup Language

PHP – Hypertext PreProcessor

SQL – Structured Query Language

CSS – Cascading Style Sheets

XAMPP – X Apache MariaDB PHP Perl

MD5 – Message-Digest algorithm 5

SUMÁRIO

RESUMO.....	8
ABSTRACT.....	9
LISTA DE FIGURAS.....	10
LISTA DE TABELAS.....	12
LISTA DE ABREVIATURAS E SIGLAS.....	13
1 INTRODUÇÃO.....	17
2 FUNDAMENTAÇÃO TEÓRICA.....	19
2.1 BOTS.....	19
2.2 CHAT-BOTS.....	20
2.3 ARQUITETURA DE BOTS.....	21
2.4 BOTS EM APLICATIVOS DE MENSAGENS INSTANTÂNEAS.....	22
3 TRABALHOS RELACIONADOS.....	24
3.1 SISTEMAS DE ARMAZENAMENTO E COMPARTILHAMENTO DE IMAGENS ATUAIS.....	24
3.1.1 Flickr.....	24
3.1.2 500px.....	25
3.1.3 Deviant Art.....	26
3.1.4 SmugMug.....	27
3.1.5 TrekEarth.....	28
3.1.6 Google Photos.....	29
3.1.7 Shutterfly.....	30
3.2 ANÁLISE COMPARATIVA.....	31
3.3 DISCUSSÃO SOBRE OS SISTEMAS SELECIONADOS.....	33
4 ANÁLISE.....	35
4.1 STAKEHOLDERS.....	35
4.2 REQUISITOS FUNCIONAIS.....	35

4.3 REQUISITOS NÃO-FUNCIONAIS.....	36
4.4 CASOS DE USO.....	37
5 MODELAGEM.....	41
5.1 DIAGRAMA ENTIDADE-RELACIONAMENTO.....	41
5.2 MÁQUINA DE ESTADOS DO BOT.....	43
6 IMPLEMENTAÇÃO.....	45
6.1 DESENVOLVIMENTO.....	45
GLOSSÁRIO.....	73
6.2 ARQUITETURA.....	46
6.2.1 FUNCIONAMENTO DO BOT.....	47
6.3 INTERFACE WEB.....	48
6.4 INTERFACE TELEGRAM.....	59
7 CONSIDERAÇÕES FINAIS.....	68
REFERÊNCIAS BIBLIOGRÁFICAS.....	70
GLOSSÁRIO.....	73

1 INTRODUÇÃO

Pessoas gostam de compartilhar experiências com familiares e amigos [1]. A partir desta premissa, este trabalho apresenta o desenvolvimento de um sistema para armazenar e compartilhar imagens, como fotos de família e viagens, que seja simples de usar, e com um diferencial em relação aos sistemas mais populares. Esse diferencial é a possibilidade de fazer *upload* de imagens através de envio de mensagens pelo sistema Telegram [2].

Sistemas de compartilhamento de imagens podem ser considerados redes sociais a depender do nível de interação permitido entre usuários. Dentre os diversos sistemas existentes, alguns foram desenvolvidos para uso abrangente e outros para propósitos específicos [3]. Existem sistemas e redes sociais voltados para compartilhamento exclusivo de textos, compartilhamento exclusivo de imagens e/ou videos, e redes híbridas, onde os usuários postam o que lhes interessa, sejam textos simples, textos com imagens ou apenas videos. Os temas das postagens podem ser os mais variados, como negócios, esportes, culinária, games ou filmes.

As três maiores redes sociais da atualidade, Facebook [4], Instagram [5] e WhatsApp [6] pertencem à mesma empresa, Meta [7]. Juntas, estas três redes concentram mais de 3,6 bilhões de usuários [8]. Por serem tão abrangentes, são usadas por pessoas de todas as classes sociais e níveis de instrução diferentes.

Este trabalho apresenta o sistema de armazenamento e compartilhamento de imagens online MyVoyage¹. Esse sistema tem como objetivo que seus usuários armazenem imagens, as compartilhem com outros usuários, façam comentários em imagens próprias ou de terceiros, e marquem suas imagens favoritas (próprias ou de terceiros), de forma que elas possam ser acessadas todas de uma vez. O sistema é acessado via web, porém para facilitar seu uso, é implementado um *bot* para Telegram [2] chamado de MyVoyagebot. Esse *bot* recebe as imagens dos usuários enviadas através do aplicativo Telegram e as coloca automaticamente em seu portfolio, dando assim ao usuário mais de uma forma de fazer o *upload* de suas imagens. Esta facilidade não foi encontrada em nenhum sistema pesquisado.

¹ Nas telas de interface gráfica, “myVoyage” é escrito com “m” minúsculo propositalmente.

O Capítulo 2 contém a fundamentação teórica e a terminologia usada ao longo deste trabalho. O Capítulo 3 apresenta os sistemas existentes que prestam um serviço similar e um comparativo entre eles e a MyVoyage. O Capítulo 4 faz a análise da MyVoyage, apresenta os *stakeholders*, requisitos e casos de uso. O Capítulo 5 apresenta a modelagem do sistema, com o Diagrama Entidade-Relacionamento do banco de dados e o diagrama de máquina de estados do *bot*. O Capítulo 6 mostra as telas do sistema MyVoyage. O Capítulo 7 apresenta as conclusões finais, possibilidades de melhorias e outras possíveis aplicações da tecnologia.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados alguns conceitos que são utilizados ao longo deste trabalho. Inicialmente é apresentado o conceito de *bot* e suas aplicações. Em seguida é explicado o conceito de *chat-bot*, um tipo específico de *bot*. A seguir é apresentada a arquitetura geral de sistemas que utilizam *bots*, e ao final é mostrado como se pode integrar sistemas de *bots* com aplicativos de mensagens instantâneas.

2.1 BOTS

Bots são softwares criados para se passar por humanos dentro de um ambiente virtual. A interação de humanos com *bots* pode se dar através de texto ou áudio, de forma que o humano não precisa ter nenhum conhecimento prévio do software com o qual ele irá interagir, pois usará opções em menus ou linguagem natural para a comunicação. Outras formas de interação também são possíveis, como a colaboração ou competição dentro de um jogo virtual ou mediação numa sala de bate-papo virtual. *Bots* também são criados para fazer varreduras em sistemas online como se fossem pessoas [9].

Dentro do universo de aplicação dos *bots*, existem os *bots* criados especificamente para conversar com humanos via chat ou telefone, como os *bots* para atendimento a clientes de grandes empresas. Esta categoria de *bots* é chamada de *chat-bots*, e são muito comuns em serviços de atendimento ao cliente de grandes empresas de diversos ramos, como varejo online, companhias aéreas e bancos [10].

Bots podem ser utilizados para monitorar o texto produzido por humanos em salas de bate-papo e atuar como mediadores [11], onde possuem privilégio para expulsar alguém que escreveu palavras-chave proibidas. Neste caso eles não precisam necessariamente conversar com os humanos. *Bots* que não conversam com humanos são também usados como jogadores com inteligência artificial de jogos multi-jogador, ocupando o lugar de jogadores humanos [12] para nivelar o nível de dificuldade do jogo.

Bots que realizam varreduras, conhecidos como *crawlers*, são usados para pesquisar websites e redes sociais para coletar dados. *Bots* são criados para pesquisar um grande volume de websites em busca de uma determinado tema, produto ou nome, ou por outro lado são criados para monitorar determinadas redes sociais, e respondem automaticamente a comentários com determinadas palavras-chave. *Crawlers* podem ser usados também para pesquisar o melhor preço de um produto dentre um determinado número de lojas online. Os buscadores de Internet possuem sua base de dados criada a partir de informações coletadas por *crawlers* [13].

2.2 CHAT-BOTS

Graças ao acesso rápido à Internet e avanços significativos em *Natural Language Processing* (NLP), um ramo da Inteligência Artificial dedicado a interpretar mensagens criadas por seres humanos em linguagem natural, são cada vez mais populares os assistentes por voz [14]. Dentre eles os mais populares são o Echo [15], fabricado pela Amazon [16], e o Siri [17], da Apple [18]. Esses assistentes interpretam frases ditas por seres humanos e executam o que foi pedido. Podem ser usados para tocar músicas pedidas pelo ser humano, criar anotações, falar a previsão do tempo, controlar dispositivos inteligentes da casa, como lâmpadas, ajustar alarme, contar piadas, etc. O repertório de ações disponíveis para *bots* está em constante expansão pelos seus fabricantes.

Chat-bots podem ser divididos em dois grupos: os que utilizam algoritmos baseados em regras, e os que utilizam algoritmos de inteligência artificial. Os *bots* baseados em regras são mais simples de programar, possuem propósitos bem definidos e suas respostas são muito previsíveis. Normalmente um ser humano se comunica com eles através de menus. Por outro lado, *bots* com inteligência artificial recebem mensagens escritas (ou faladas) por um ser humano como se estivesse se dirigindo a outro ser humano, e através de técnicas de NLP fazem sua interpretação. A partir da análise da mensagem apresentam o resultado mais provável. Por se tratar de um ramo extenso da Inteligência Artificial, este tipo de *bot* não será analisado neste trabalho.

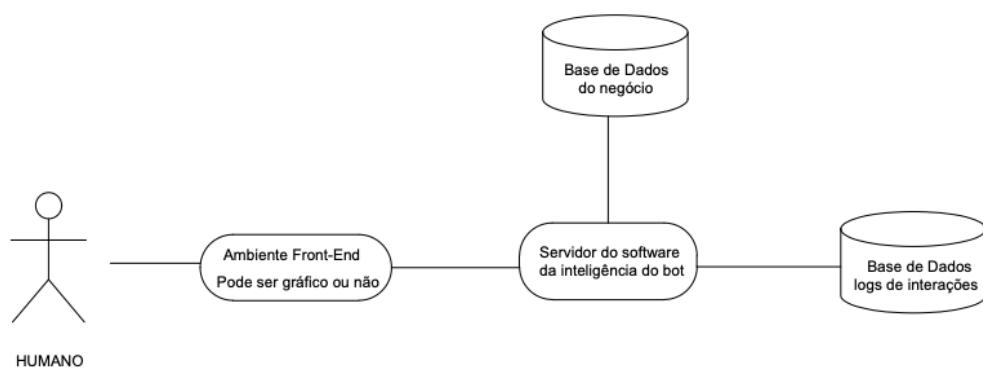
2.3 ARQUITETURA DE BOTS

Os *bots*, independentemente da sua aplicação ou dos algoritmos utilizados, possuem algumas características comuns em sua arquitetura. É necessário um componente *front-end* para a interface com o usuário. Esse componente pode ser um terminal, uma página web, um aplicativo para celular, ou até mesmo um dispositivo com microfone e caixa de som.

O software responsável pela inteligência do *bot* roda num servidor dedicado, que para aplicações comerciais costuma se manter ligado 24/7. Esse servidor é dimensionado de acordo com o tráfego esperado e com a complexidade do software do *bot*. De forma a não sobrecarregar o servidor dedicado, uma parte do software do *bot* pode rodar diretamente no *front-end*. O dispositivo Echo [15], por exemplo, só se comunica com o servidor após identificar que algum humano disse a palavra “Alexa”. Esta capacidade de filtrar o que está sendo dito no ambiente está implementada no software do dispositivo *front-end* [15].

Para que seja útil, o *bot* usualmente precisa ter acesso à base de dados do sistema ao qual ele dá suporte, para que possa realizar consultas e fazer alterações. A base de dados pode estar em outro servidor, acessado remotamente pelo servidor do *bot*, conforme mostra a Figura 1. Desta forma o *bot* pode, por exemplo, consultar o status de um pedido do cliente, alterar seu endereço ou registrar uma reclamação. Esta base de dados tipicamente contém as informações da empresa ou negócio.

Figura 1 - Exemplo de arquitetura de um bot



Fonte: Elaboração do autor

Uma base de dados secundária se faz necessária quando se deseja armazenar os *logs* das interações com os usuários. Isto é especialmente útil em sistemas com inteligência artificial que utilizam aprendizado (*machine learning*) para aprimoramento constante [19]. Mesmo com o uso de *bots* baseados em regras, os *logs* são importantes porque a partir deles pode-se fazer uma série de inferências, como quais as transações mais usadas pelos usuários, quais não estão obtendo respostas satisfatórias do *bot*, quantidade diária de acessos, etc. Todas estas informações são úteis para aprimorar o *bot* e consequentemente para que a empresa possa atender melhor seus clientes.

2.4 BOTS EM APLICATIVOS DE MENSAGENS INSTANTÂNEAS

Aplicativos de mensagens instantâneas permitem que seus usuários enviem mensagens entre si. A mensagem pode conter texto, áudio, imagem, documento e/ou vídeo. Cada usuário possui uma lista de contatos, que são outros usuários do mesmo software, e a partir daí pode escolher para quem enviar uma mensagem. Por outro lado, quem recebe uma mensagem é instantaneamente notificado, e pode respondê-la quando e como quiser [20].

Esses aplicativos permitem ainda a criação de grupos de usuários, que são salas virtuais onde todos podem ver as mensagens postadas pelos membros do grupo. O número máximo de membros permitidos por grupo depende do aplicativo, sendo que o WhatsApp, o aplicativo de mensagens mais popular do Brasil [21], permite criação de grupos com até 256 pessoas, enquanto que o Telegram [3], aplicativo similar, permite grupos com até 200.000 pessoas [22].

Os aplicativos de mensagens possibilitam também a criação de *bots* pelos próprios usuários, que são consideradas aplicações de terceiros (“*third-party*”). O usuário inicialmente registra o nome para o seu *bot*. No Telegram, isto é feito enviando-se uma mensagem para @BotFather. Ao ser registrado, é atribuído um código identificador a este bot, e este código é recebido pelo usuário que solicitou a criação do *bot* [23].

Após cadastrar o seu *bot*, o usuário pode criar um programa na linguagem de sua preferência utilizando as bibliotecas adequadas ao sistema escolhido

(Telegram, WhatsApp, etc.). Especificamente para o Telegram, existem duas maneiras de operacionalizar um *bot*. Uma delas é mantendo um servidor especializado e conectado à Internet. Através da Telegram-bot-API, criada pelo Telegram, o programa do *bot* configura um *webhook* com o servidor do Telegram. Desta forma sempre que ocorrer algum determinado evento no servidor do Telegram definido pelo programa, como a chegada de uma mensagem tipo texto, por exemplo, o servidor do Telegram envia para a URL do servidor especializado do *bot* a mensagem recebida. Com o uso de servidor especializado não há limite para o tamanho de mensagem recebido e não é necessário que o programa do *bot* realize *polling* com o servidor do Telegram.

A outra alternativa, que é utilizada neste trabalho, é manter um computador comum com conexão à Internet e com o software do *bot* executando um *polling* no servidor do Telegram continuamente. Durante o *polling* o servidor do Telegram é consultado para checar se alguma nova mensagem foi recebida.

De qualquer forma, toda mensagem recebida contém o código identificador de quem a enviou, bem como outras informações, como tipo de mensagem enviada (texto, imagem, audio, etc.), horário de envio, plataforma usada para acesso ao sistema (sistema operacional do dispositivo usado), etc. Sabendo o código identificador de quem enviou a mensagem, o *bot* está apto a continuar a conversa enviando outra mensagem de volta.

3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados alguns dos sistemas atuais, suas características principais e um comparativo entre eles. Apesar de existirem dezenas de sistemas online para armazenamento e compartilhamento de imagens, cada um possui características específicas e são voltados para públicos diferentes. Todos eles, entretanto, possuem uma característica em comum: permitem ao usuário fazer *upload* de imagens e armazená-las em um servidor com acesso à Internet.

3.1 SISTEMAS DE ARMAZENAMENTO E COMPARTILHAMENTO DE IMAGENS ATUAIS

Uma pesquisa inicial na Internet para este trabalho revelou que existem dezenas de sistemas diferentes, e portanto a análise de todos é inviável. Foram selecionados sete sistemas diferentes e realizada uma comparação entre eles. Todos os sistemas selecionados têm em comum uma elaborada interface com usuário, muitas opções de interatividade e grande base de usuários. Alguns destes sistemas são adequados a fotógrafos e/ou artistas profissionais, e contam com opção de venda e licenciamento de imagens.

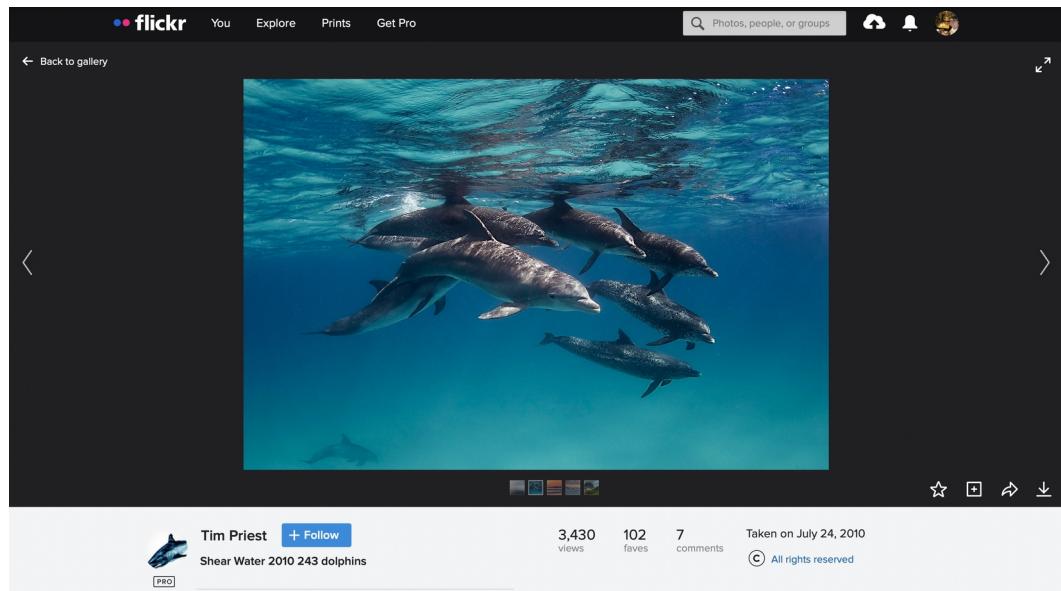
3.1.1 Flickr

Flickr [24] é um sistema lançado em 2004 por uma empresa canadense. Esse sistema atende muito bem aos profissionais, mas também tem um apelo ao público em geral, pois além de permitir armazenamento grátis de até mil imagens ou vídeos sem compressão (até 200 Mb cada foto e até 1 Gb cada vídeo), vende impressão de imagens.

Os usuários podem atribuir tags às imagens, de forma a facilitar sua busca. Galerias de imagens podem ser criadas e a elas atribuídos nomes e temática. Pode-se adicionar imagens próprias ou de outros usuários às galerias criadas e marcar usuários nas imagens. Para cada imagem armazenada o usuário pode inserir

informações do equipamento (máquina fotográfica, tipo de lente) e configuração usada. Nas fotos tiradas pelo celular, as informações sobre o modelo do aparelho já são automaticamente colocadas nas imagens. A Figura 2 apresenta a visualização de imagem de terceiros no Flickr.

Figura 2 – Visualização de imagem no Flickr



Fonte: FLICKR [24]

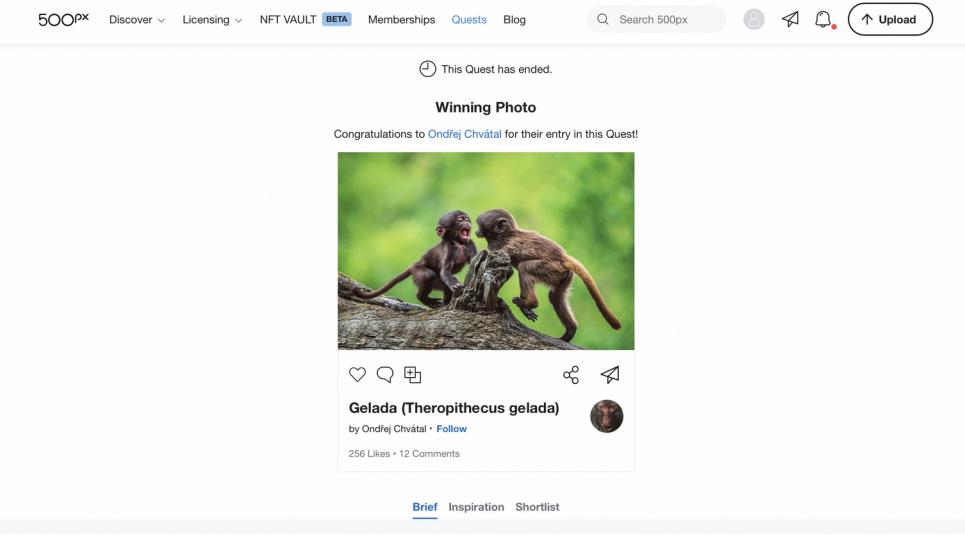
Nas contas pagas são disponibilizadas estatísticas de acesso às imagens, licenciamento de imagens, remoção de anúncios, armazenamento ilimitado e descontos em assinaturas de outros sites relacionados.

3.1.2 500px

O sistema 500px [25] (pronuncia-se “*five hundred pixels*”) possui muitas similaridades com o Flickr, porém não é vantajoso para o público em geral, pois a conta gratuita só permite *upload* de sete imagens por semana e não permite favoritar imagens.

O 500px possui o diferencial das “*quests*”, onde determinados temas são especificados pelos administradores do sistema e usuários podem enviar fotos que se enquadram no tema e nas regras. As melhores fotos são premiadas em dólares. A Figura 3 apresenta a visualização de uma foto vencedora de uma *quest* no 500px.

Figura 3 – Visualização de foto vencedora de *quest* no 500px



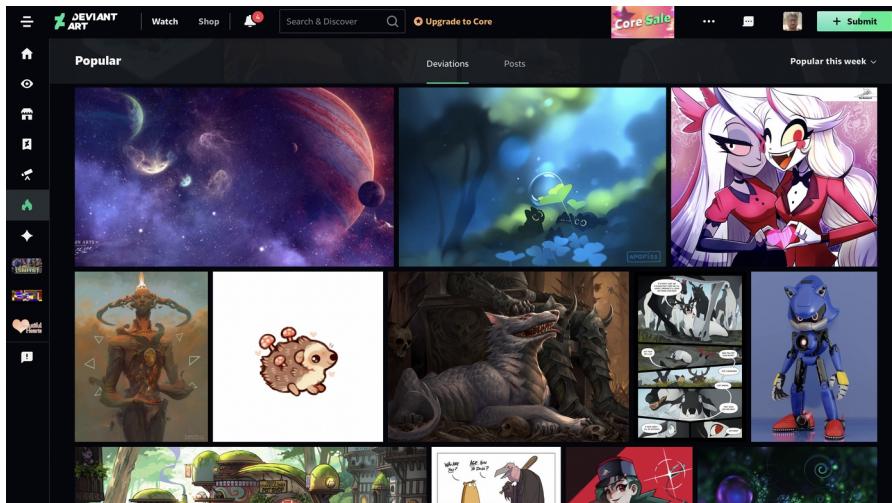
Fonte: 500PX [25]

O website do 500px conta ainda com uma seção chamada de “Blog”, que é uma revista virtual com uma série de artigos sobre fotografia. Os temas variam entre dicas de fotos, análises de câmeras, tutorias, guias para venda de imagens, etc.

3.1.3 Deviant Art

Dentro do acervo de imagens do website Deviant Art [26], fotografia é apenas um dentre dezenas de tópicos. A grande maioria das imagens são de personagens de animações e paisagens fantásticas criadas com computação gráfica, como pode ser visto na Figura 4. Esse sistema tem um grande acervo de arte conceitual para jogos e filmes.

Figura 4 – Imagens de terceiros no Deviant Art



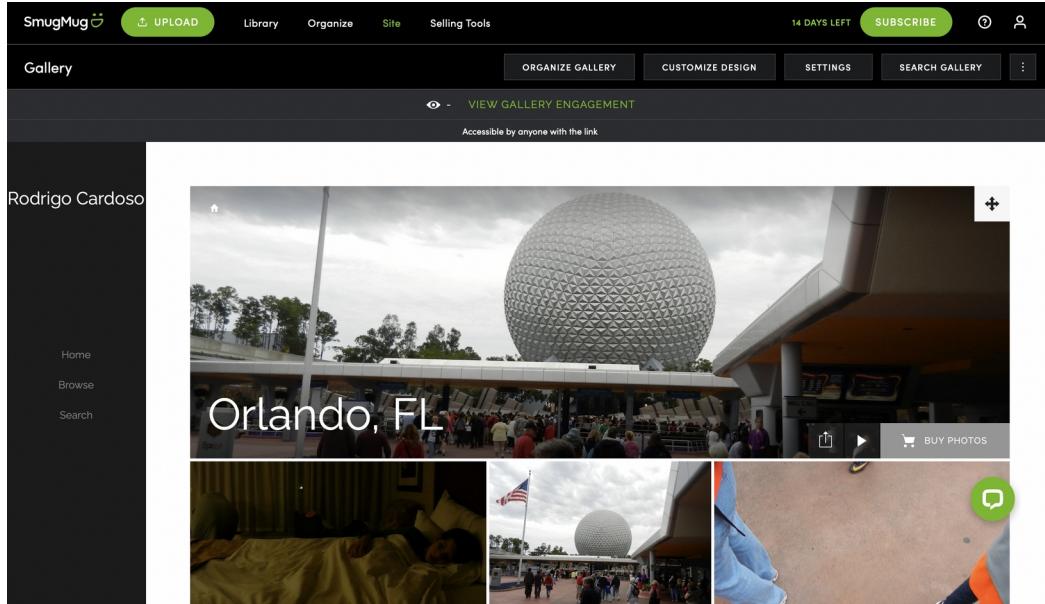
Fonte: DEVIANT ART [26]

O Deviant Art permite que usuários postem artigos sobre temas diversos, com uso de texto e imagens. O website contém um fórum onde os usuários discutem diversos assuntos, como jogos, filmes, tecnologia, emprego e até música. É possível comprar impressões das imagens escolhidas, na forma de fotografias, telas, e “presentes de arte” (“*art gifts*”), onde são vendidas canecas, *mouse pads*, ímãs e cartão-postais.

3.1.4 SmugMug

SmugMug [27] é uma plataforma que permite a criação de uma galeria virtual, onde o usuário pode expor seu trabalho. Cada usuário ganha um domínio com o endereço de sua galeria. O sistema permite venda de impressões, venda de downloads e colocação de marca d’água. O website do SmugMug permite *upload* de imagens no formato RAW, e também possui um sistema de inteligência artificial exclusivo para classificar imagens. A Figura 5 apresenta a visualização do portfolio próprio no website SmugMug.

Figura 5 – Portfolio próprio no SmugMug



Fonte: SMUGMUG [27]

Esse sistema é voltado para expor trabalhos próprios, e por não haver interação entre usuários, não pode ser considerado uma rede social. Ele não apresenta imagens de outros usuários na página inicial e não é possível enviar mensagens a eles. O sistema não possui contas gratuitas, porém a conta paga mais barata custa US\$ 3 / mês e permite armazenamento de até 512 Gb, o que torna o website atrativo para o público que deseja um lugar seguro para armazenar suas fotos, e além disto ainda vende impressão de fotos.

3.1.5 TrekEarth

Dentre os sistemas pesquisados para este trabalho, o mais próximo de um sistema voltado para turismo é o TrekEarth [28]. Além das características listadas ele permite marcar no mapa-mundi os lugares que o usuário já viajou e tem vontade de viajar, e permite busca de fotos a partir dos nomes de países, regiões e cidades. A Figura 6 apresenta a tela do TrekEarth relacionada à África do Sul.

O sistema TrekEarth tem a opção de descrever o equipamento usado para tirar as fotos e criar tags para facilitar busca de imagens. Os usuários podem ganhar títulos no website, como “*Gold Star Critiquer*” e “*Gold Note Writer*”, dependendo da

sua interação com o site, como postagem de imagens e comentários. Esta é uma forma de engajar os usuários sem gastar dinheiro.

Figura 6 – Página da África do Sul no TrekEarth



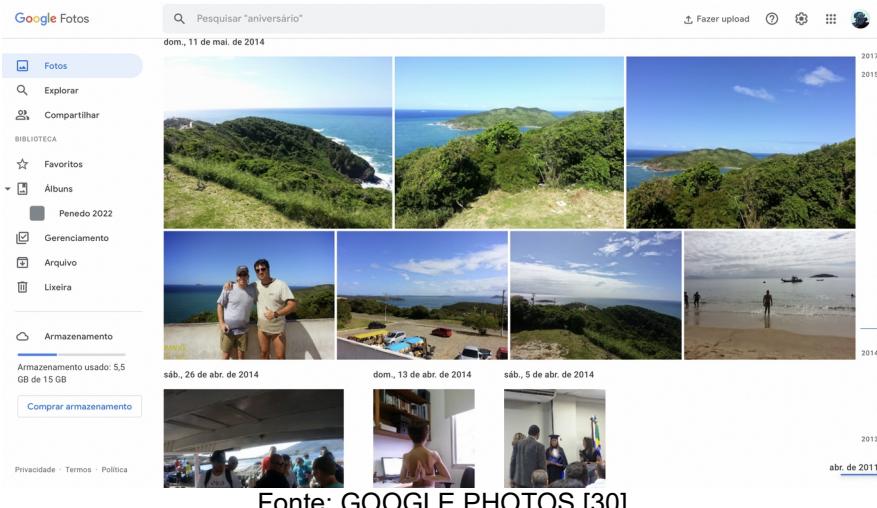
Fonte: TREKEARTH [28]

O sistema atualmente está com defeito e não permite a inclusão e novos usuários. Além disso, ao se tentar enviar mensagens para reclamar, devido a uma falha no sistema de *Captcha* (que sempre acusa que os caracteres digitados estão errados) não é possível contato com os administradores.

3.1.6 Google Photos

Dentre os vários serviços gratuitos oferecidos pelo Google [29] ao público em geral, o Google Photos [30] é o serviço de armazenamento de imagens e vídeos na nuvem. A interface é bem simples e intuitiva. Como o Google Photos tem acesso aos sistemas de inteligência artificial do Google, o Google Photos possui a vantagem de reconhecer rostos e várias figuras automaticamente, o que facilita na hora de buscar imagens. A Figura 7 apresenta a página com o portfolio pessoal de imagens do Google Photos.

Figura 7 – Portfolio próprio no Google Photos



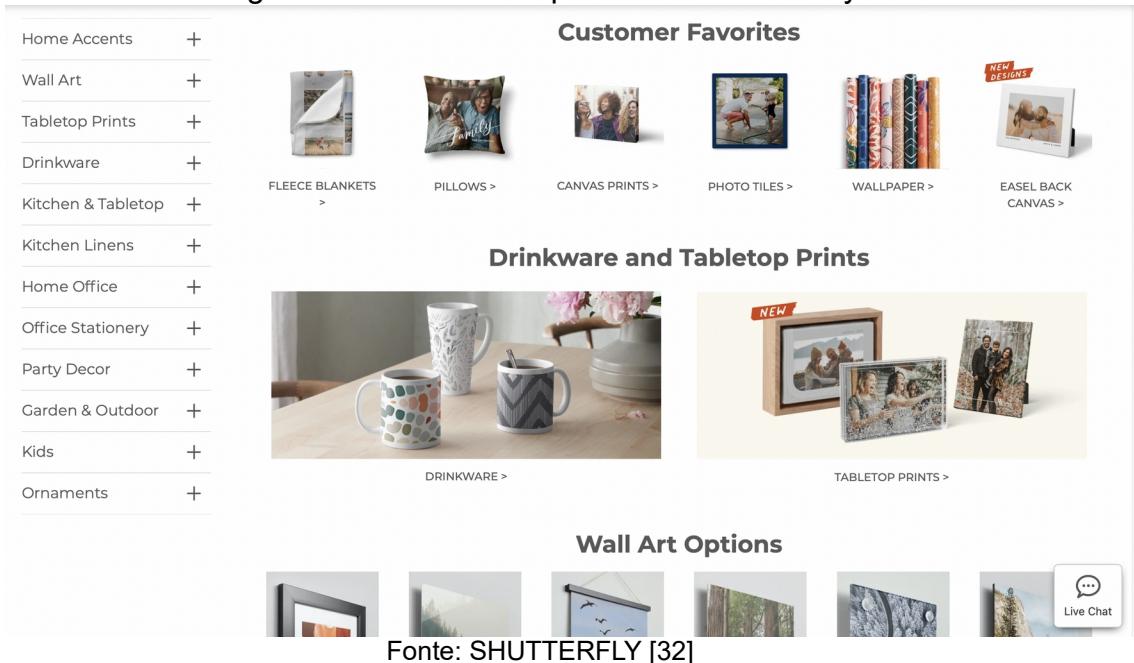
Fonte: GOOGLE PHOTOS [30]

O Google Photos é integrado ao sistema operacional Android [31], de forma que o usuário de aparelho Android pode optar por fazer *upload* automático das fotos tiradas pelo aparelho. Isto torna o sistema muito fácil de usar e com uma grande base de usuários. O armazenamento de até 15 Gb é gratuito. O Google oferece planos pagos para as capacidades de 100 Gb, 200 Gb e 2 Tb.

3.1.7 Shutterfly

O sistema Shutterfly [32] permite armazenamento ilimitado de imagens para usuários com conta grátis, e tem opção de contas pagas caso o usuário deseje armazenar vídeos. O sistema não permite interação alguma entre usuários. A Figura 8 mostra uma tela do sistema Shutterfly onde são apresentados diversos produtos que são vendidos e que podem ser personalizados com imagens do usuário.

Figura 8 – Produtos disponíveis no Shutterfly



Fonte: SHUTTERFLY [32]

O sistema vende diversos tipos de produtos com imagens impressas, como álbuns, quadros, canecas, calendários, camisetas, toalhas, etc. O design do sistema é muito similar ao de uma loja online, pois o faturamento é proveniente da venda de produtos. Pode-se entender este sistema como uma loja online que tem o diferencial de permitir que seus clientes armazenem imagens em sua nuvem.

3.2 ANÁLISE COMPARATIVA

Todos os sistemas online de armazenamento de imagens avaliados possuem uma ou mais das características apresentadas a seguir. A enumeração destas características permite que se faça uma comparação entre estes sistemas analisados de forma a permitir a escolha do mais adequado aos desejos do usuário.

- I. Inclusão de comentários nas imagens postadas: Tanto o usuário que postou a imagem quanto outros usuários podem colocar comentários sob as imagens postadas.
- II. Seguir usuários: Usuários podem marcar outros usuários que desejam “seguir”. Os usuários “seguidores” são notificados sempre que o usuário “seguido” posta uma nova imagem e/ou visualizam estas imagens assim que logam no

sistema. Além disso, eles têm facilidade de acesso à sua página através de um link onde são mostrados todos os usuários “seguidos”.

III. Contas pagas: O sistema cobra taxas para determinados tipos de conta, porém pode permitir contas gratuitas. Contas pagas possuem certos privilégios, como remoção de anúncios, maior capacidade de armazenamento, venda de direitos de imagens, inclusão de marca d’água, etc.

IV. Marcação de imagens favoritas: Os usuários podem marcar como “favorita” imagens suas e de outros usuários. Todas as imagens marcadas como “favoritas” são rapidamente visualizadas através de um clique na interface.

V. *Tags* nas Imagens: O sistema permite associar palavras-chave (*tags*) a cada imagem que o usuário envia, de forma a facilitar a busca por imagens que possuem a palavra-chave.

VI. Impressão de Imagens: O sistema permite que o usuário compre imagens impressas. As impressões podem ser em papel e vendidas como álbum de fotos, ou até mesmo em produtos diversos, como toalhas e canecas.

VII. Reconhecimento Facial: O sistema reconhece rostos nas imagens e informa quem é a pessoa mais provável.

VIII. Licenciamento de Imagens: O sistema permite que o usuário venda direitos de suas imagens e acesso a seu acervo privado. Sistemas que possuem esta característica são voltados para profissionais do ramo.

A Tabela 1 apresenta a comparação entre os sistemas pesquisados considerando estas oito características. Algumas conclusões podem ser tiradas a partir desta tabela.

Tabela 1 – Comparativo entre sistemas de armazenamento de imagens

	Comentários	Seguir Usuários	Contas Pagas	Imagens Favoritas	Tags nas imagens	Impressão	Reconhec. Facial	Licenciamento
Flickr	✓	✓	✓	✓	✓	✓		✓
500px	✓	✓	✓	✓	✓			✓
Deviant Art	✓	✓	✓	✓	✓			✓
TrekEarth	✓	✓		✓	✓			
Shutterfly			✓	✓	✓	✓		
Google Photos			✓	✓	✓		✓	
SmugMug			✓	✓	✓	✓		✓

Fonte: Elaboração do autor

É importante notar que o único sistema totalmente gratuito é o único que está com defeito e impede a entrada de novos usuários. Todos os sistemas permitem marcar imagens favoritas e criar *tags* para imagens, facilitando assim a busca pelas imagens favoritas ou sobre um determinado tema.

3.3 DISCUSSÃO SOBRE OS SISTEMAS SELECIONADOS

Os sistemas de armazenamento de imagens selecionados têm confiabilidade, design que facilita a navegação e um custo acessível. Para realizar *upload* de imagens, todos os sistemas requerem que o usuário faça login em sua conta através de um navegador web ou utilize um aplicativo dedicado no seu tablet ou celular. As exceções aparentes são o Google Photos, que envia automaticamente a imagem caso o sistema operacional do dispositivo seja Android, ou o iCloud [33], da Apple [18], caso o sistema operacional do dispositivo seja iOS [34] ou iPadOS [35]. Isso ocorre porque nestes casos o usuário faz o login no sistema operacional.

Usuários comuns utilizam seus próprios celulares para tirar fotos hoje em dia [36]. Acessar websites para realizar ações além de simples leitura através de celulares costuma ser uma experiência ruim para os usuários, pois é necessário clicar numa série de links, muitas vezes pequenos na tela, para se chegar na página desejada, além do desconforto de ter que digitar login e senha através de um

diminuto teclado virtual [37]. Quanto aos aplicativos, além de consumirem espaço no armazenamento do celular, precisam ser constantemente atualizados [38].

4 ANÁLISE

Este capítulo apresenta o resultado da fase de análise do sistema MyVoyage. No capítulo são identificados os *stakeholders*, os requisitos funcionais e não-funcionais, o diagrama de casos de uso e são detalhados casos de uso selecionados.

4.1 STAKEHOLDERS

Todo projeto de software, além de atender diretamente pessoas e organizações, pode afetar indiretamente outras pessoas e organizações [39]. Os indivíduos e/ou organizações que são afetados pelo projeto são chamados de *stakeholders*. Eles representam as partes interessadas afetadas pelo sistema [40]. Eles não precisam necessariamente ser usuários do sistema, mas são de alguma forma afetados por ele. Cada *stakeholder* tem objetivos distintos em relação ao sistema. A relação abaixo apresenta os *stakeholders* identificados para o sistema MyVoyage.

Usuário

objetivo: Armazenar suas imagens num local seguro e de fácil acesso, gerenciar suas imagens, ter acesso rápido e fácil às suas imagens preferidas, ver imagens de terceiros, comentar nas imagens, ler comentários de terceiros nas imagens

Obs: Todo Usuário já foi Visitante, porém nem todo Visitante vira Usuário

Administrador

objetivo: mediar o sistema e coletar informações sobre os usuários

Visitante

objetivo: conhecer o sistema e virar Usuário após cadastro

4.2 REQUISITOS FUNCIONAIS

Os requisitos funcionais são as condições de funcionamento do sistema [41], ou seja, eles definem como o sistema irá operar. A definição dos requisitos funcionais é uma etapa fundamental na criação de qualquer sistema, pois antes de

começar a programar o sistema precisamos definir exatamente como ele deverá se comportar. Para o sistema MyVoyage foram estabelecidos os requisitos funcionais descritos na Tabela 2:

Tabela 2 - Requisitos funcionais do sistema MyVoyage

RF01	O sistema deverá permitir que novos usuários se cadastrem sem intervenção do administrador
RF02	O sistema deverá armazenar imagens enviadas pelos usuários via interface web
RF03	O sistema deverá permitir que usuários marquem e desmarquem imagens de qualquer usuário como favoritas
RF04	O sistema deverá permitir que o usuário visualize seu portfolio
RF05	O sistema deverá permitir que o usuário apague imagens do seu portfolio
RF06	O sistema deverá permitir que o usuário visualize imagens aleatórias de terceiros (foto surpresa)
RF07	O sistema deverá permitir que o usuário visualize suas imagens favoritas
RF08	O sistema deverá permitir que o usuário faça buscas de usuários
RF09	O sistema deverá permitir que o usuário visualize imagens de terceiros
RF10	O sistema deverá permitir que usuários escrevam comentários em quaisquer imagens
RF11	O sistema deverá permitir ao administrador eliminar o cadastro de qualquer usuário
RF12	O sistema deverá realizar o cadastro do ID Telegram através do MyVoyagebot

Fonte: Elaboração do autor

4.3 REQUISITOS NÃO-FUNCIONAIS

Os requisitos não-funcionais descrevem não o que o sistema fará, mas como ele fará o que é esperado dele. Eles são requisitos do interesse do usuário que estão ligados a alguma característica computacional, como por exemplo o tempo máximo aceitável de resposta a uma busca ou o sistema operacional usado pelo usuário. Eles reúnem as condições tecnológicas que o sistema deverá ter. Estes requisitos não dizem respeito à como o sistema deve se comportar (que são os requisitos funcionais), mas às suas condições de desempenho e sobre que plataformas ele será construído [41]. A partir dos requisitos não-funcionais é que deverão ser escolhidos o hardware, sistema operacional, linguagens de programação, softwares, roteadores, e todos os demais dispositivos necessários para que o sistema possa operar. A lista de requisitos não-funcionais identificados para o sistema MyVoyage está mostrada na Tabela 3.

Tabela 3 - Requisitos não-funcionais do sistema MyVoyage

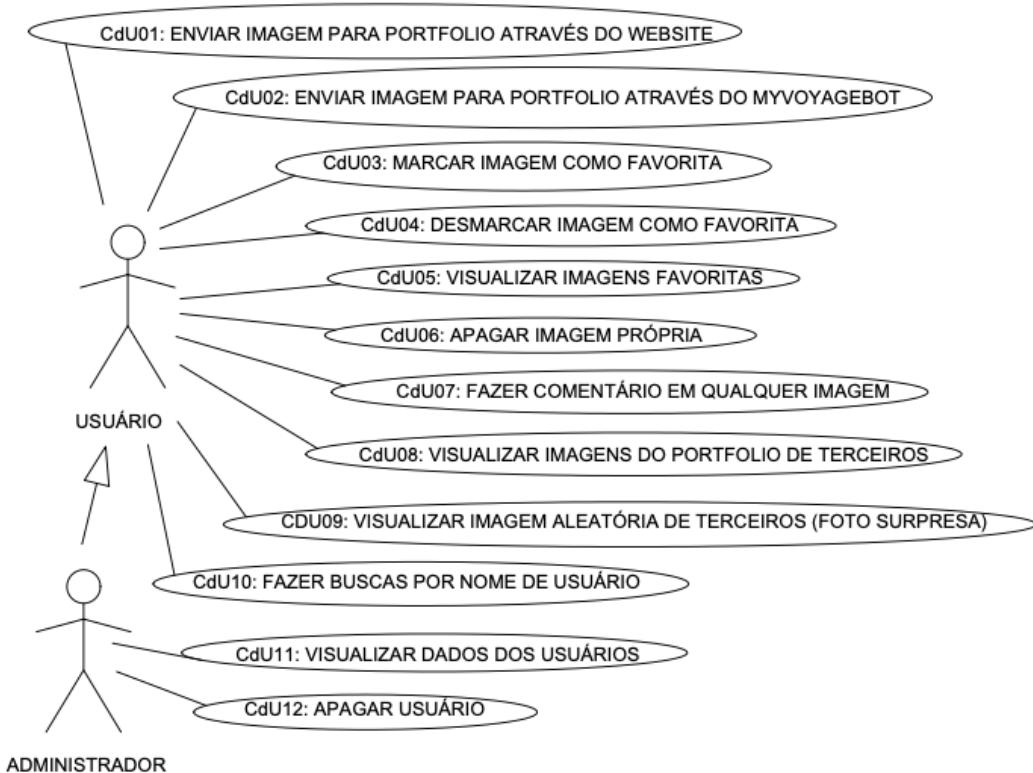
RNF01	O sistema deverá ser acessado via web
RNF02	O sistema possuirá um bot para envio de imagens que opera no Telegram
RNF03	O sistema utilizará o MySQL como banco de dados
RNF04	O sistema usará scripts em PHP para gerar as páginas web
RNF05	O sistema utilizará a biblioteca Telepot em Python no programa do bot
RNF06	O sistema deverá ter telas web projetadas apenas para desktop

Fonte: Elaboração do autor

4.4 CASOS DE USO

Segundo Craig Larman, “(...) caso de uso é uma coleção de cenários relacionados de sucesso e fracasso, que descrevem um ator usando um sistema como meio para atingir um objetivo.” [41]. Nesta seção são analisados os diversos casos de uso do sistema MyVoyage, criados a partir dos requisitos funcionais da seção 4.2. Todos os casos de uso identificados estão mostrados na Figura 9.

Figura 9 - Diagrama de Casos de Uso do sistema MyVoyage



Fonte: Elaboração do autor

Alguns casos de uso selecionados estão detalhados a seguir. No detalhamento foi considerado o fluxo principal, o fluxo alternativo e as pós-condições. As pós-condições refletem a situação final do sistema após aquele caso de uso ter sido utilizado. Caso a execução do caso de uso não resulte em nenhuma alteração no banco de dados do sistema, a pós-condição será “Não aplicável”.

O caso de uso 02, “Enviar imagem para o portfolio do sistema através do MyVoyagebot” é o diferencial em relação aos serviços de armazenamento de imagens pesquisados, e pode ser visto na Tabela 4. O caso de uso 03, “Marcar imagem como favorita”, em contrapartida, é comum a todos os serviços pesquisados, e está mostrado na Tabela 5.

Tabela 4 - Caso de uso 02

CdU02	
Nome	Enviar imagem para o portfolio do sistema através do MyVoyagebot
Atores	Usuário, MyVoyagebot
Pré-Condições	Usuário tem seu ID Telegram cadastrado no banco de dados (RF12) Usuário precisa estar logado no Telegram MyVoyagebot deve estar na lista de contatos do Telegram do Usuário
Fluxo Principal	1 - Usuário está logado no Telegram 2 - Usuário seleciona MyVoyagebot 3 - Usuário clica no ícone do clipe 4 - Usuário seleciona a imagem no seu dispositivo 5 - Usuário clica na seta para enviar imagem
Fluxo Alternativo	1 - Usuário está logado no Telegram 2 - Usuário entra na pasta de fotos do seu dispositivo 3 - Usuário seleciona a imagem e clica no botão Compartilhar do S.O. 4 - Usuário seleciona “Telegram” 5 - Usuário seleciona “MyVoyagebot”
Pós-Condições	A imagem está no banco de dados do MyVoyage A imagem está visível no portfolio do Usuário
Regras do Negócio	RF12, RF13

Fonte: Elaboração do autor

Tabela 5 - Caso de uso 03

CdU03	
Nome	Marcar imagem como favorita
Atores	Usuário
Pré-Condições	Usuário está visualizando a imagem que deseja marcar como favorita A imagem não está marcada como favorita
Fluxo Principal	1 - Usuário visualiza a imagem que deseja marcar como favorita 2 - Usuário clica no ícone “Favoritar”
Fluxo Alternativo	N/D
Pós-Condições	A imagem está marcada como favorita do Usuário A imagem está visível na pasta de fotos favoritas
Regras do Negócio	RF02, RF03, RF04, RF06, RF07, RF09

Fonte: Elaboração do autor

O caso de uso 08, “Visualizar imagens do portfolio de terceiros”, é importante porque é uma das principais funções do MyVoyage e há mais de um caminho para que isto seja feito. Este caso de uso está apresentado na Tabela 6.

Tabela 6 - Caso de uso 08

CdU08	
Nome	Visualizar imagens do portfolio de terceiros
Atores	Usuário
Pré-Condições	Devem haver no mínimo dois usuários cadastrados
Fluxo Principal	1 - Usuário clica em “Buscar Usuário” na tela inicial 2 - Usuário digita uma string para busca por nome de usuário 3 - Usuário clica em “Buscar” 4 - Na lista de resultados da busca, usuário clica no botão “Ver” da linha referente ao usuário buscado
Fluxo Alternativo	1 - Usuário clica em “Foto Surpresa” na tela inicial 2 - Na tela de visualização da foto surpresa, usuário clica no botão com o nome do dono da imagem
Pós-Condições	Não aplicável
Regras do Negócio	RF06, RF08, RF09

Fonte: Elaboração do autor

O caso de uso 7, “Apagar Usuário”, exclusivo do administrador, é importante porque sua execução gera uma série de pós-condições no banco de dados. Apesar disso ele é bem simples de ser executado. Este caso de uso é detalhado na Tabela 7.

Tabela 7 - Caso de uso 12

CdU12	
Nome	Apagar usuário
Atores	Administrador, usuário
Pré-Condições	Usuário estar cadastrado
Fluxo Principal	1 - Administrador clica em “Ver Usuários” na tela inicial 2 - Administrador clica no botão “Apagar” na linha referente ao usuário que ele deseja apagar 3 - Administrador clica em “Sim” na tela onde é perguntado se tem certeza
Fluxo Alternativo	Não aplicável
Pós-Condições	O usuário é apagado do banco de dados, assim como todas as suas imagens e seus comentários

Fonte: Elaboração do autor

5 MODELAGEM

Este capítulo apresenta modelos do funcionamento do sistema MyVoyage através de diagramas da linguagem UML [42]. O capítulo especifica o Diagrama Entidade-Relacionamento (DER), que modela a estrutura do banco de dados utilizado pelo sistema. A seguir é apresentado um Diagrama de Máquina de Estados [43] para descrever o funcionamento básico do MyVoyagebot.

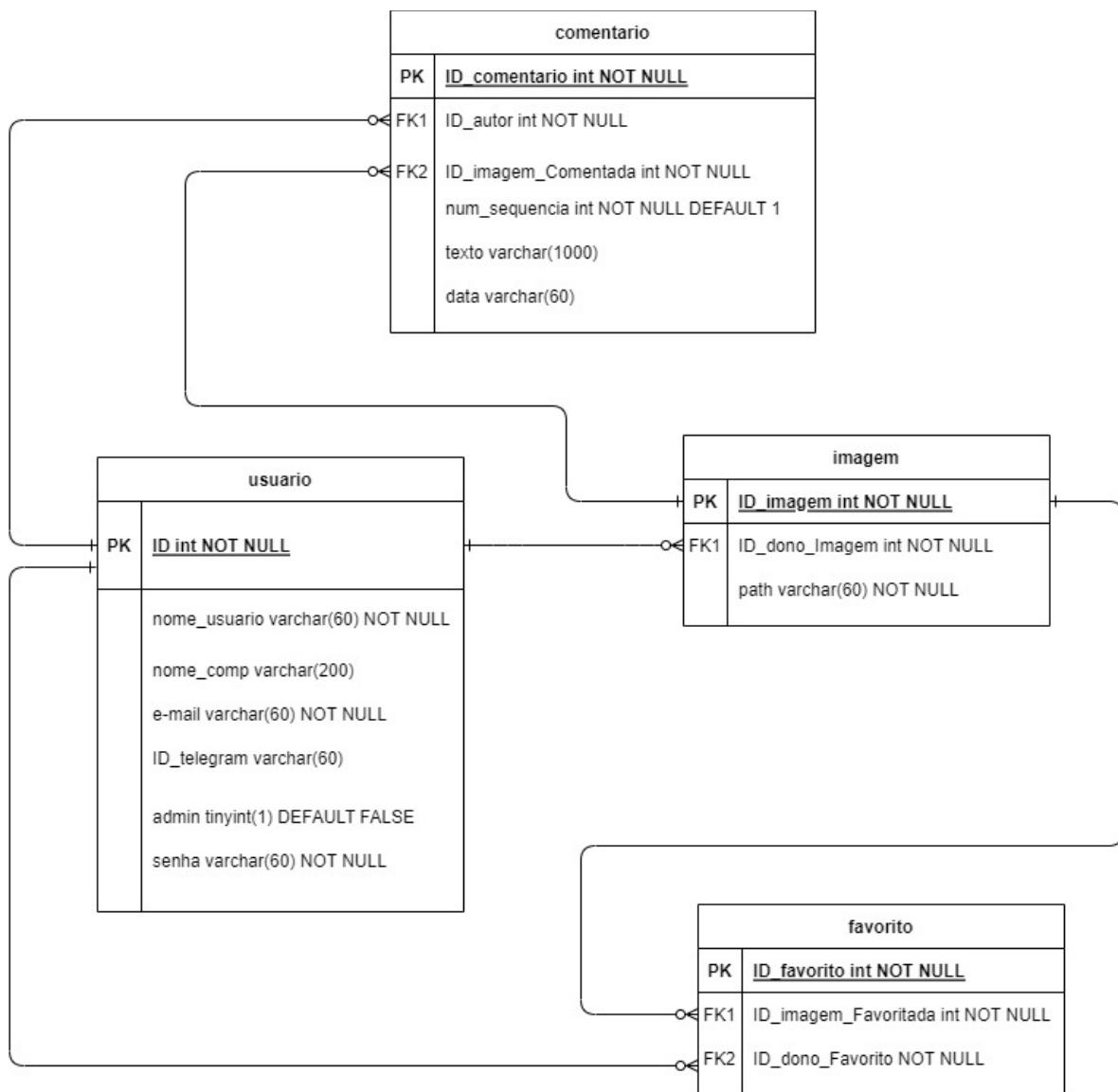
5.1 DIAGRAMA ENTIDADE-RELACIONAMENTO

O DER é um modelo gráfico em rede que descreve a diagramação dos dados armazenados de um sistema em alto nível de abstração [44]. Ele não representa as funções executadas pelo sistema ou as transições de estado. A partir da construção deste diagrama são definidos os dados essenciais que o banco de dados precisará armazenar, e como estes dados estão relacionados entre si.

Para o sistema MyVoyage foram definidas quatro tabelas no banco de dados: “usuario”, “imagem”, “favorito” e “comentario”. A partir destas tabelas é possível armazenar os dados de todos os usuários, todas as imagens, quais imagens são favoritas e para quem, e todos os comentários em ordem sequencial de postagem. A cada cadastro de usuário novo é criada uma instância de ‘usuario’, a cada *upload* de imagem bem sucedido é criada uma instância de ‘imagem’, a cada clique no ícone ‘Favoritar’ é criada ou apagada uma instância de ‘favorito’, e a cada comentário postado é criada uma instância de ‘comentario’. O DER da MyVoyage pode ser visualizado na Figura 10.

A tabela ‘imagem’ apresenta um atributo chamado ‘path’, que armazena uma *string* que é o caminho da imagem no disco rígido. O banco de dados não armazena a imagem em si, mas o seu endereço e seu dono. Quando uma imagem é deletada pelo usuário, sua instância de ‘imagem’ é apagada do banco de dados, assim como todas as instâncias de ‘favorito’ e ‘comentario’ que fazem referência a ela. Em seguida o sistema deleta o arquivo da imagem através do sistema operacional.

Figura 10 - Diagrama Entidade-Relacionamento do sistema MyVoyage



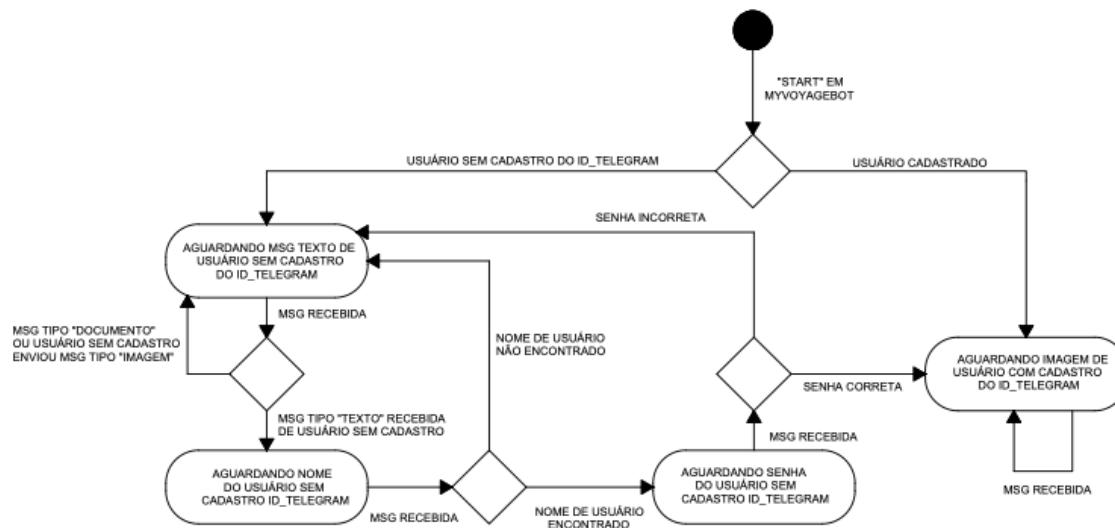
Fonte: Elaboração do autor

5.2 MÁQUINA DE ESTADOS DO BOT

O Telegram classifica as mensagens recebidas em três categorias: “texto”, “imagem” e “documento”. Toda mensagem recebida carrega o ID_Telegram do remetente e a data/hora do envio. O programa do *bot* é implementado como um *loop* infinito, onde uma função para checar se chegou alguma mensagem é constantemente chamada. Caso tenha chegado alguma mensagem, uma série de checagens e ações serão realizadas.

A troca de mensagens com o *bot* pode ser modelada como uma máquina de estados finitos, onde sempre que uma nova mensagem chega, o estado é alterado. O novo estado é definido a partir do estado anterior e do conteúdo/tipo da nova mensagem recebida. Todos os estados possíveis estão mostrados na Figura 11.

Figura 11 - Máquina de estados finitos do bot



Fonte: Elaboração do autor

Antes do usuário enviar sua primeira imagem para o portfolio através do MyVoyagebot, ele deverá enviar uma mensagem de texto para o *bot* para se identificar. O *bot* então iniciará uma conversa para realizar seu cadastro do ID_Telegram. Com o cadastro feito, as mensagens de texto passam a ser

desnecessárias, e o *bot* estará apto a enviar automaticamente para o portfolio todas as imagens que chegarem daquele usuário. Caso o usuário envie uma imagem ao *bot* antes de realizar seu cadastro do ID_Telegram, ele receberá uma mensagem dizendo para enviar uma mensagem de texto primeiro.

6 IMPLEMENTAÇÃO

Neste capítulo são apresentadas as linguagens de programação, banco de dados e ambiente de desenvolvimento utilizados para a programação do sistema MyVoyage. Em seguida são apresentadas todas as telas HTML que podem ser visualizadas pelos usuários. As telas apresentadas abrangem todos os casos de uso previstos na Figura 8.

6.1 DESENVOLVIMENTO

O IDE (“Integrated Development Environment”) utilizado foi o Visual Studio Code (VS Code) [45]. O VS Code é gratuito, possui suporte para várias linguagens de programação e é multiplataforma.

Todas as páginas HTML do MyVoyage são geradas dinâmicamente através dos *scripts* PHP que rodam no servidor. Sempre que há uma consulta ou alteração a ser feita no banco de dados, o programa abre um link com o banco de dados e envia o devido comando na linguagem SQL. Após receber a resposta do banco de dados o link é fechado. Para o gerenciamento do banco de dados foi usado o phpMyAdmin.

De forma a deixar o website visualmente mais agradável, foram utilizadas algumas classes de estilo CSS importadas da biblioteca Bootstrap [46]. Estas classes foram usadas para desenhar os botões e caixas de texto do MyVoyage.

Durante o desenvolvimento do sistema foi utilizado o pacote de programas XAMPP [47] para criar um servidor Apache local com suporte a PHP e MySQL. Este pacote é gratuito e multiplataforma.

Para aumentar a segurança, a senha armazenada no banco de dados é o resultado de uma função *Hash MD5* aplicada sobre uma *string* composta pelo nome do usuário e a senha digitada por ele. Desta forma não há como os administradores saberem qual é a senha digitada pelo usuário.

Os arquivos ficam armazenados numa pasta do disco rígido do servidor. A pasta é “*fotos/id_do_usuario/ano_da_postagem*”. O nome do arquivo salvo no

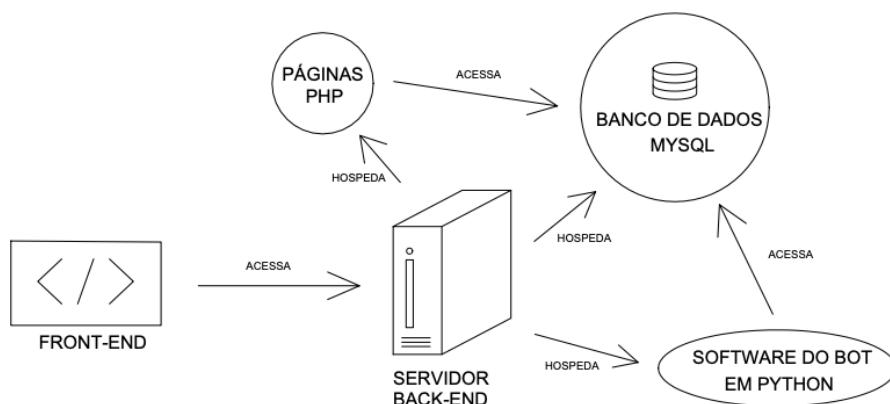
servidor é o resultado de uma função *Hash MD5* aplicada sobre a data/hora do *upload*. Como cada usuário tem suas imagens guardadas em sua própria pasta, mesmo se dois ou mais usuários enviarem imagens simultaneamente não há risco de nenhum arquivo ser sobrescrito. No banco de dados é armazenado o *path* completo da imagem, e não o arquivo.

6.2 ARQUITETURA

O sistema MyVoyage foi projetado para ser acessado a partir de um navegador web. O usuário faz o papel do cliente na relação cliente – servidor. O cliente faz uma requisição de página ao servidor. O servidor então executa o *script* em PHP referente à página solicitada. Ao executar o *script* o servidor gera dinamicamente a página HTML. Esta página é transmitida ao cliente, que a visualiza em seu navegador web.

Os *scripts* PHP que geram as páginas do website do sistema MyVoyage ficam armazenadas num servidor “*back-end*”. As páginas visualizadas no cliente formam o “*front-end*”. No servidor também é mantido um banco de dados no formato MySQL gerenciado através do phpMyAdmin v. 5.1.1. A arquitetura é apresentada na Figura 12.

Figura 12 - Arquitetura do sistema MyVoyage



Fonte: Elaboração do autor

O software do MyVoyagebot foi escrito em Python 3.10. Para a comunicação com o Telegram foi usada a biblioteca “Telepot” v.12.7. O programa do *bot* é executado no mesmo servidor dos *scripts* PHP e do banco de dados. Para que o *bot* funcione seu programa precisa estar sendo executado no servidor. O programa do *bot* fica o tempo todo à espera de nova mensagem recebida pelo servidor do Telegram. Quando uma nova mensagem chega ele toma as ações necessárias e retorna ao modo de espera por nova mensagem. Foi utilizado o VS Code v.1.72.2 para o desenvolvimento dos *scripts* PHP e do código em Python do MyVoyagebot.

6.2.1 FUNCIONAMENTO DO BOT

O MyVoyagebot, ou simplesmente *bot* para simplificar, possui duas funções:

1 - Realizar o cadastro do ID_Telegram do usuário (esta função é exclusiva do *bot*, não podendo ser realizada pelo website); e

2 - Armazenar as imagens enviadas pelo usuário em seu portfolio.

Todos os usuários do Telegram possuem um código numérico único de identificação, porém não é simples conhecer seu próprio código. O aplicativo Telegram para iOS e para Windows, por exemplo, não dá ao usuário a opção de visualizar seu próprio código. A única forma pela qual foi possível saber o identificador do usuário foi extraíndo esta informação da mensagem que o *bot* recebe. Toda mensagem que chega ao *bot* (isto se aplica a mensagem de texto, imagem ou documento) possui uma série de informações que podem ser extraídas facilmente, como o identificador (ID_Telegram) do usuário e a data de envio. Como nem os usuários têm como saber seu próprio identificador, o sistema MyVoyage não permite atualizar o cadastro do ID_Telegram via web, e sim exclusivamente pelo *bot*.

Antes de estar habilitado a enviar imagens para o *bot*, o usuário deverá então fazer o cadastro de seu ID_Telegram no sistema MyVoyage. Para isto ele apenas manda uma mensagem de texto qualquer, de forma que o *bot* possa saber seu ID_Telegram e consultar o banco de dados para saber se ele já está cadastrado. Caso não esteja, o usuário será solicitado a enviar seu nome de usuário e senha, para que o *bot* possa validar o usuário e assim fazer o cadastro do seu

ID_Telegram. Uma vez que o cadastro do ID_Telegram tenha sido feito com sucesso, não é mais necessário enviar mensagens de texto para o *bot*, basta enviar uma imagem qualquer para que ela seja armazenada no portfolio.

O *bot* foi programado na linguagem Python e para a comunicação com o sistema Telegram foi utilizada a biblioteca “Telepot”. O programa do *bot* é basicamente um *loop* infinito de escutar mensagem. Quando uma nova mensagem chega, o *bot* imediatamente identifica quem a enviou através da extração do ID_Telegram da mensagem e o tipo de mensagem que chegou. Existem três tipos: “texto”, “imagem” e “documento”.

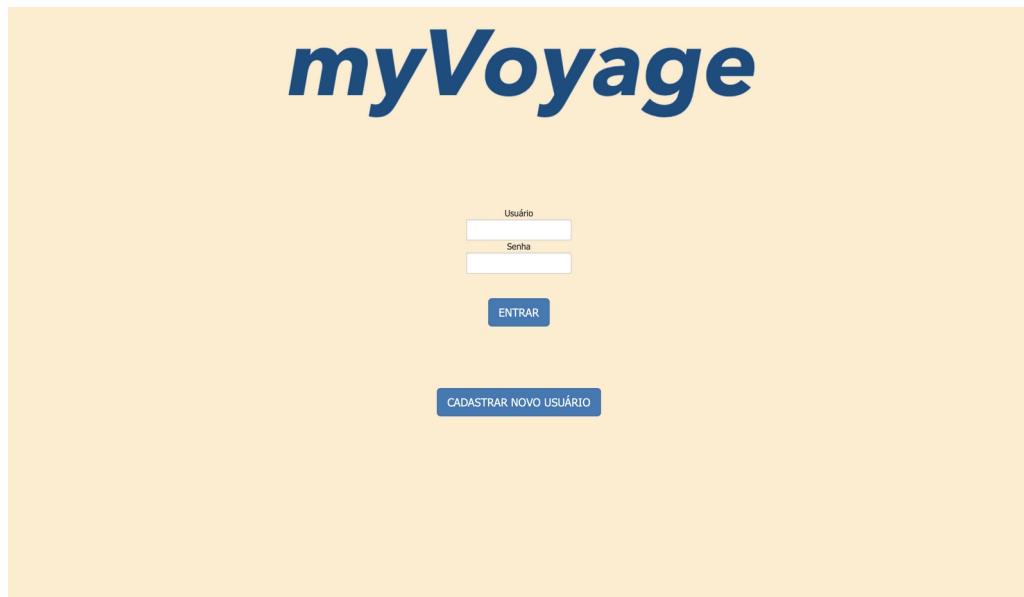
Caso a mensagem seja do tipo “imagem”, o *bot* confere se o ID_Telegram contido na mensagem já consta no banco de dados. Em caso negativo o *bot* envia uma mensagem dizendo ao usuário para que ele faça o cadastro do seu ID_Telegram enviando uma mensagem tipo “texto”. Em caso positivo a imagem é salva no disco do servidor e uma nova instância da tabela ‘Imagen’ é criada, e assim o portfolio do usuário é atualizado. Caso a mensagem seja do tipo “documento”, o *bot* informa que este formato não é válido.

O programa do *bot* utiliza duas classes criadas especialmente para ele: uma classe chamada ‘CConversa’, cujos atributos são ID_Telegram, estágio da conversa, nome do usuário e data de envio da última mensagem. A outra classe é ‘CListaConversas’, cuja instância é um vetor cujos elementos são instâncias da classe Conversa. Este vetor é usado para consultar se já existe uma conversa em andamento. Em caso negativo, uma nova instância de CConversa é criada e adicionada a este vetor. Quando uma conversa é finalizada, aquela instância relativa a ela é removida deste vetor.

6.3 INTERFACE WEB

Inicialmente o usuário faz o login com o nome de usuário e senha, ou se for um visitante, clica no botão para criar cadastro. A tela de login só apresenta estas duas funcionalidades para ser o mais simples possível, como pode ser vista na Figura 13.

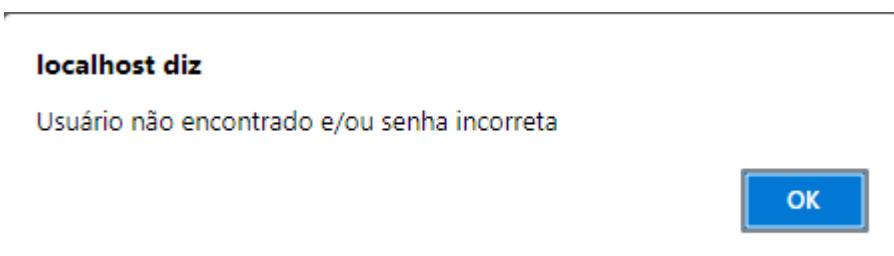
Figura 13 - Login



Fonte: Elaboração do autor

Caso o usuário não exista ou a senha estiver incorreta, um alerta de erro é apresentado. Após o clique em “Ok” o usuário volta para a tela de Login, como é mostrado na Figura 14.

Figura 14 - Usuário não encontrado e/ou senha incorreta



Fonte: Elaboração do autor

A tela de cadastro não solicita o ID_Telegram. Este dado só pode ser fornecido pelo MyVoyagebot. Como o cadastro é feito apenas pelo usuário, o atributo “admin”, que dá privilégio de administrador, é falso por *default*. Caso algum novo usuário deva ser promovido a administrador, então é necessário que seja feita uma intervenção na instância da pessoa na tabela ‘usuario’ dentro do PHPMyAdmin. A tela de cadastro com os atributos iniciais pode ser vista na Figura 15.

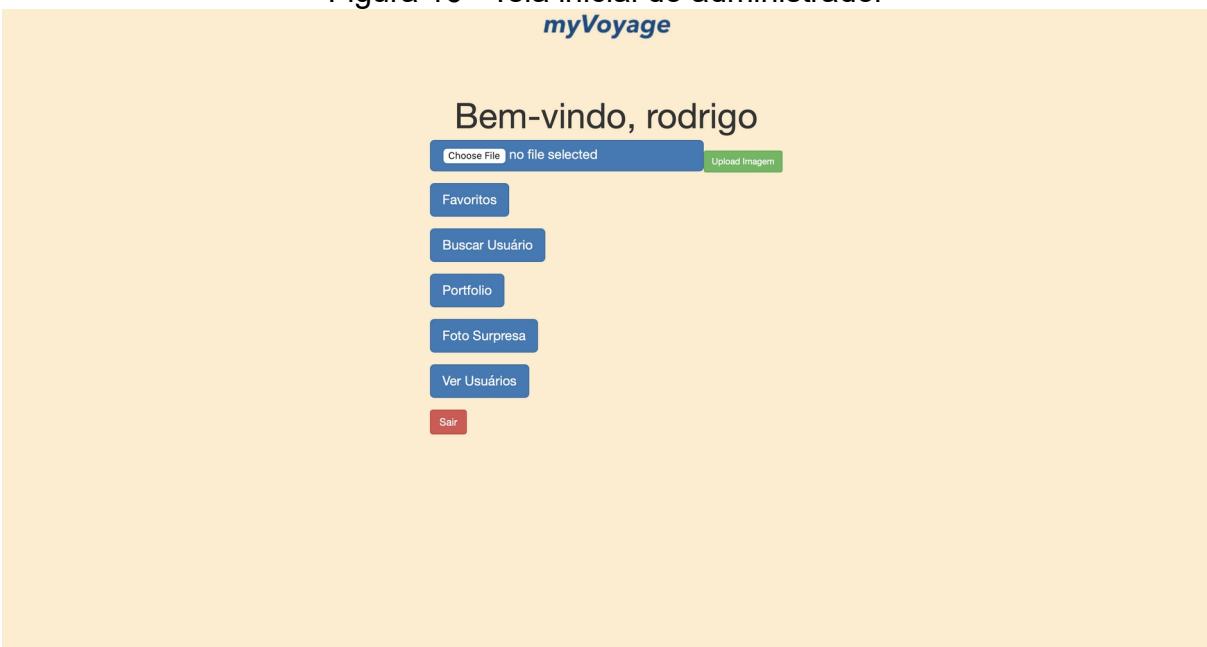
Figura 15 – Cadastro de novo usuário

The screenshot shows the 'myVoyage' user registration interface. At the top, the 'myVoyage' logo is displayed. Below it, the title 'Cadastro de novo usuário' is centered. The form contains four input fields: 'Usuário' (User) with the value 'roberto', 'Nome completo' (Full Name) with the value 'Roberto Carlos da Silva', 'E-mail' (Email) with the value 'rob.carlos@mv.com', and two password fields: 'Senha' (Password) and 'Confirma senha' (Confirm Password). Below the form are two buttons: 'CADASTRAR' (Register) in blue and 'SAIR' (Logout) in grey.

Fonte: Elaboração do autor

Após ser feito o login, o usuário é levado à tela inicial do sistema. A tela inicial é igual para usuários comuns e administradores exceto por uma única diferença: no caso do administrador haverá um botão a mais, chamado “Ver Usuários”, para fazer o gerenciamento dos usuários do website. A tela do administrador pode ser vista na Figura 16.

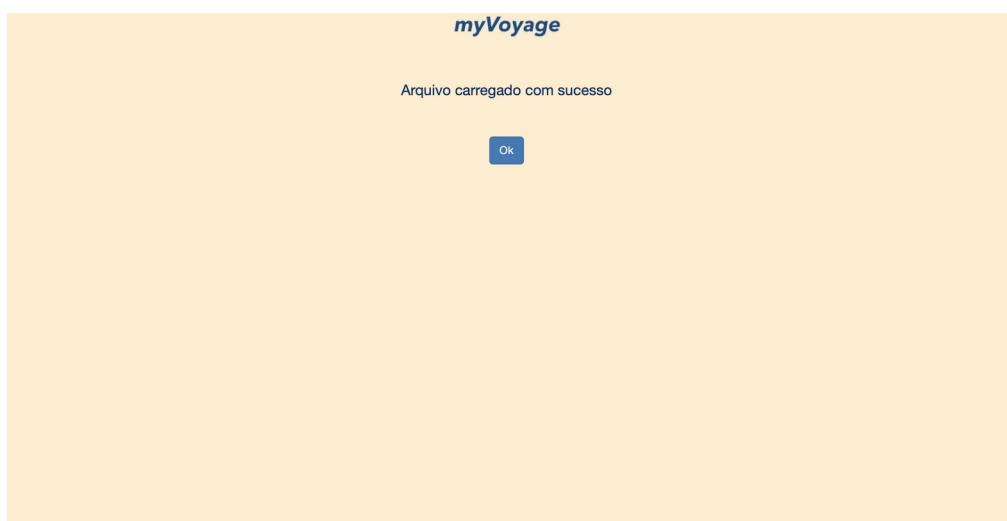
Figura 16 - Tela inicial do administrador
myVoyage



Fonte: Elaboração do autor

Ao clicar no botão “Escolher Arquivo”, aparecerá uma janela do sistema operacional para que o usuário selecione o arquivo desejado. Ao terminar a seleção, o nome do arquivo aparecerá ao lado do botão “Escolher Arquivo”. Basta então clicar no botão verde “Upload Imagem” para enviar a imagem para o seu portfolio. Caso o envio seja bem sucedido, uma mensagem de sucesso aparecerá na tela, caso contrário aparecerá uma mensagem com o motivo do arquivo não ser enviado para o portfolio. A Figura 17 mostra a tela mostrada ao usuário após o arquivo ser enviado com sucesso.

Figura 17 - Arquivo carregado com sucesso



Fonte: Elaboração do autor

Caso o arquivo escolhido não seja de um tipo aceitável pelo sistema, uma mensagem de erro será apresentada. O sistema está programado para aceitar apenas arquivos com terminação *jpg*, *jpeg*, *png* e *gif*. Ainda que o arquivo enviado tenha uma destas terminações, o arquivo é checado para confirmar que se trata de uma imagem, e não um arquivo qualquer que tenha sido simplesmente renomeado. Caso o sistema identifique que não se trata de uma imagem e/ou não tenha uma terminação aceitável, a mensagem da Figura 18 será exibida após o clique em “Upload Imagem” da tela inicial.

Figura 18 - Arquivo não está no formato aceitável



Fonte: Elaboração do autor

Ao clicar no botão “Portfolio”, na tela inicial, será apresentada a tela de Portfolio, aonde todas as imagens do usuário serão mostradas na tela, em formato de tabela, porém redimensionadas para que não ocupem mais do que 300 pixels de largura. Este valor foi escolhido para que não apareçam mais do que cinco imagens por linha. Um exemplo de portfolio está mostrado na Figura 19.

Figura 19 - Portfolio

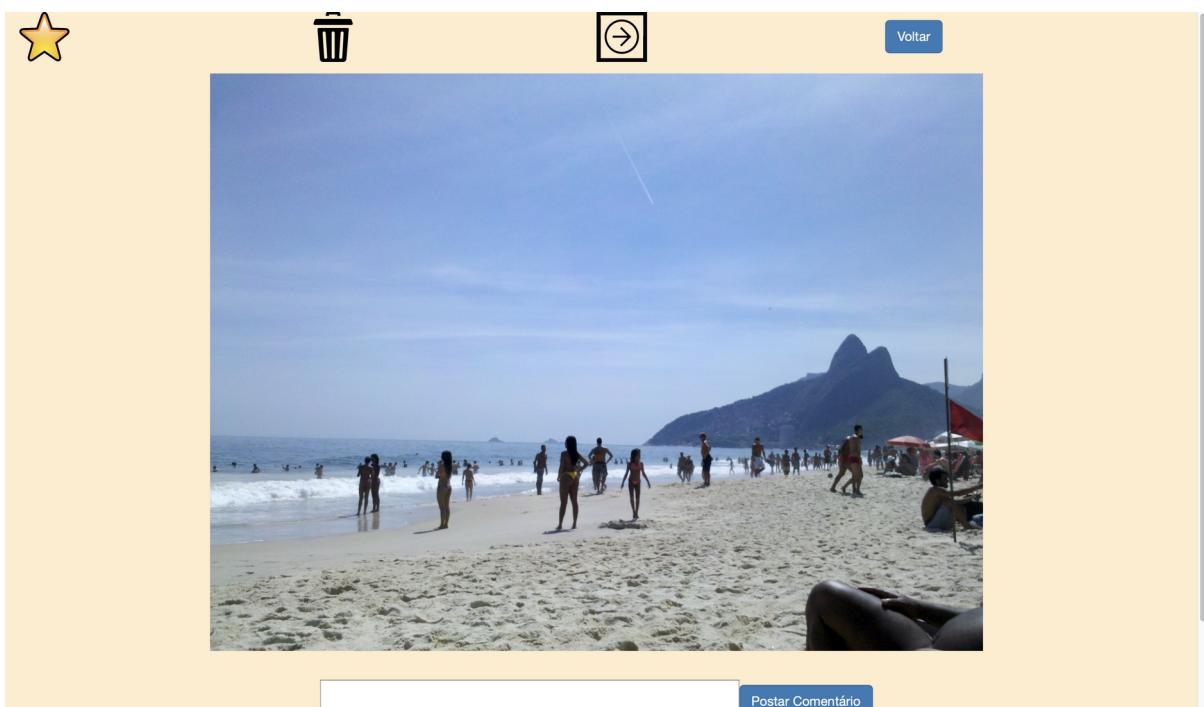


Fonte: Elaboração do autor

Na tela de portfolio as imagens funcionam como botões, que ao serem clicados levam a uma tela onde se pode visualizar a imagem sozinha. Esta tela é chamada de tela de visualização e apresenta uma série de funcionalidades e a tabela de comentários. As telas de portfolio próprio e de terceiros são similares, sendo a única diferença o nome do dono do portfolio em “Portfolio de (...)”.

Caso a largura da imagem seja maior que 1700 *pixels* ou sua altura maior que 800 pixels, ela será redimensionada mantendo sua proporção para que o usuário possa visualizá-la por inteiro sem precisar utilizar as barras de rolagem inferior e lateral do navegador. Se ele no entanto fizer um *download* da imagem, o arquivo baixado será a imagem no tamanho original. A tela de visualização de uma imagem própria pode ser vista na Figura 20.

Figura 20 - Tela de visualização de imagem própria



Fonte: Elaboração do autor

Os botões de comando foram projetados para serem facilmente acessíveis mesmo em telas de celular e para se manterem sempre na mesma posição, independente do tamanho da imagem exibida. Acima da imagem, no lado esquerdo, há o ícone no formato de estrela, que poderá ser branca ou dourada, que representa a ação de “favoritar” a imagem. Caso a estrela apareça como branca, isto significa que a imagem não é favorita do usuário. Ao clicar na estrela ela se tornará dourada, e a imagem passará a ser uma das favoritas. Ao clicar na estrela

dourada, aquela imagem deixa de ser favorita do usuário e a estrela volta a ser branca.

Ao lado do ícone da estrela está o ícone da lixeira. Este ícone é usado para apagar a imagem. Este ícone só aparece quando a imagem visualizada é própria. Ao lado dele há o ícone de uma seta apontando para a direita, que é usado para visualizar a próxima imagem do portfolio (caso a imagem mostrada já seja a última, a próxima imagem será a primeira). Por fim há o botão de voltar para a tela anterior.

Ao se clicar no botão de apagar, uma confirmação é requerida antes de proceder com a eliminação da imagem, como mostrado na Figura 21. Quando a confirmação é dada, todas instâncias de ‘favorito’ onde o ID_imagem_Favoritada corresponde ao ID da imagem são apagadas. O mesmo é feito com as instâncias de ‘comentário’, onde a correspondência se dá pelo ID_imagem_Comentada. A seguir a instância é apagada da tabela ‘imagem’ e por fim o arquivo é fisicamente apagado no disco rígido.

Figura 21 - Confirmação para apagar imagem



Fonte: Elaboração do autor

Por fim temos abaixo da imagem a tabela de comentários, onde qualquer usuário pode postar um comentário, que ficará registrado com o nome do autor e

data/hora da postagem. Não foi implementada a opção de apagar comentários ou bloquear comentários de determinada imagem. Os comentários vão se acumulando de cima para baixo, portanto os de baixo são os mais recentes. Isso pode ser visto na Figura 22.

Figura 22 - Comentários na imagem



Postar Comentário

marcos escreveu em 15-10-22 05:04:05:
Esta praia é linda! Lembra deste dia ?

maria escreveu em 15-10-22 05:04:59:
Claro! Adorei tomar um banho de mar!

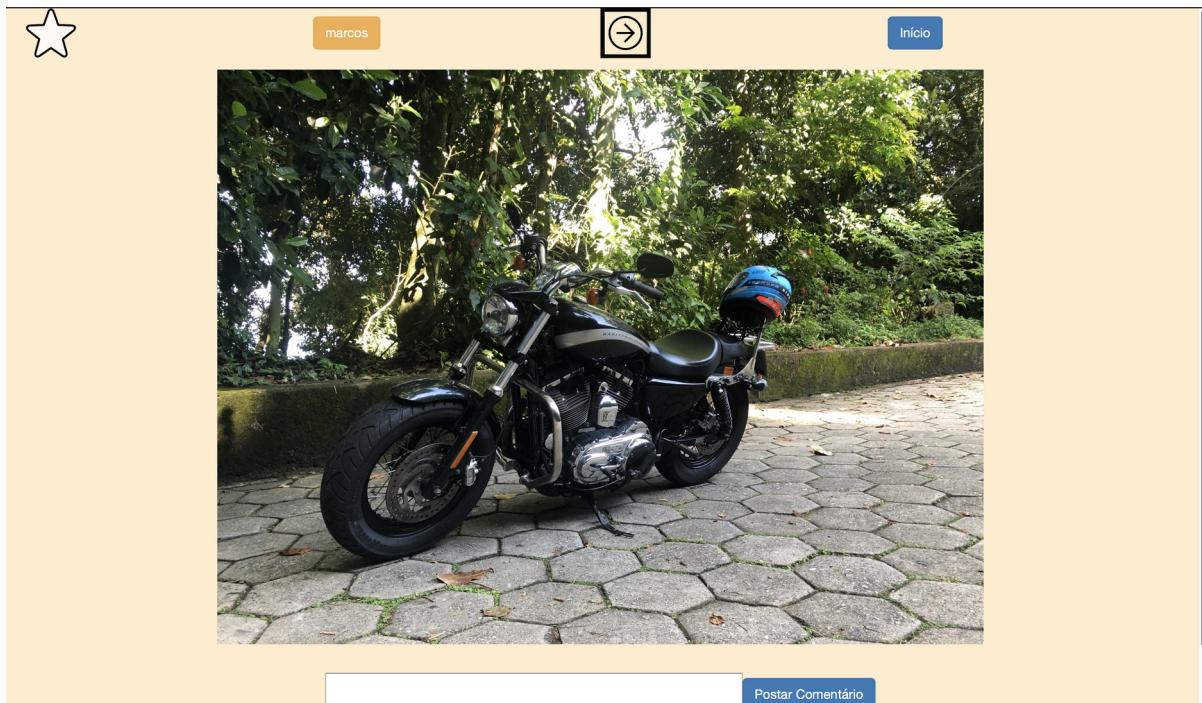
nilson escreveu em 28-10-22 03:50:54:
Esta foto ficou muito boa! Parabéns!

Fonte: Elaboração do autor

A tela inicial possui o botão “Favoritas”, que leva à tela onde todas imagens que foram “favoritadas” pelo usuário são apresentadas no mesmo formato do portfolio. A diferença em relação ao portfolio é que as imagens apresentadas como favoritas podem ser de qualquer usuário do MyVoyage.

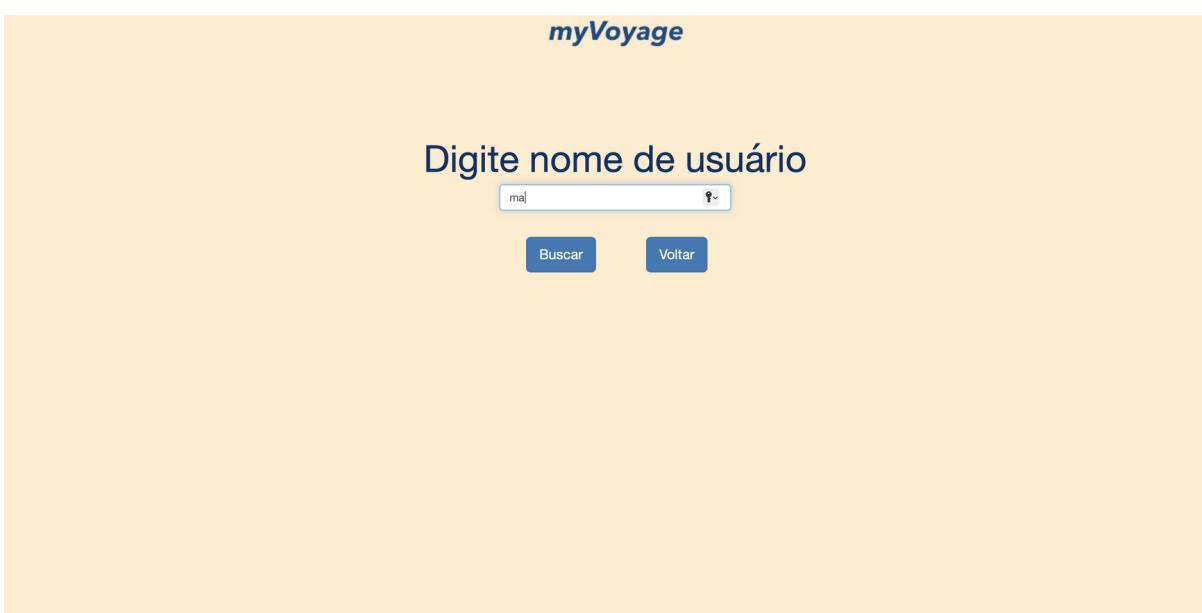
Ao visualizar uma imagem de terceiros, não aparecerá o ícone da lixeira, pois apenas o dono da imagem pode apagá-la. Ao invés do ícone da lixeira aparecerá um botão com o nome do dono da imagem. Ao clicar neste botão o usuário verá o portfolio do dono daquela imagem. A tela de visualização de imagens de terceiros é mostrada na Figura 23.

Figura 23 - Tela de visualização de imagem de terceiros



Na tela inicial o usuário pode fazer uma busca de nome de outros usuários. A *string* que o usuário digitar no campo da busca será usada para fazer uma busca no banco de dados no campo de nome de usuário, e a lista de resultados é apresentada em forma de tabela. A tela de busca é mostrada na Figura 24.

Figura 24 - Busca de usuários



Fonte: Elaboração do autor

Todos os nomes de usuários que contiverem aquela *string* digitada aparecerão na forma de tabela. Ao clicar no botão referente a um determinado usuário da tabela, será mostrado o seu portfolio. A tela do resultado da busca é mostrada na Figura 25.

Figura 25 - Resultado da busca de usuários

The screenshot shows a search results page for users. At the top, the text "myVoyage" is displayed. Below it, a message says "Total de Usuários = 2". A table lists two users: "marcos" and "maria". Each user entry has a "Ver" button to the right. At the bottom of the page is a "Voltar" button.

Total de Usuários = 2	
nome_usuario	
marcos	<button>Ver</button>
maria	<button>Ver</button>

Fonte: Elaboração do autor

O botão “Foto Surpresa” da tela inicial é usado para visualizar uma imagem aleatória de terceiros. Quando está sendo visualizada uma imagem aleatória, o ícone da seta para direita fará com que outra imagem de terceiros seja sorteada e visualizada.

Por fim, para gerenciamento do sistema foi criado o botão “Ver Usuários”. Ele é exclusivo para usuários com perfil de administrador. Ele é usado para ver no formato de tabela a lista de todos os usuários da MyVoyage, o total de imagens postadas de cada um deles e a opção de apagar cada usuário, como é mostrado na Figura 26.

Figura 26 - Lista de usuários da MyVoyage



The screenshot shows a table titled "Total de Usuários = 5". The columns are: ID, nome_usuario, nome_comp, e-mail, admin, Total Fotos, and a "Apagar" (Delete) button. The data rows are:

ID	nome_usuario	nome_comp	e-mail	admin	Total Fotos	
16	nilson	nilson damasceno	nilson@t.com	0	4	Apagar
23	rodrigo	Rodrigo Cardoso	rdac.quake@gmail.com	1	8	Apagar
24	marcos	Marcos Silva Gomes	marcx34@live.com	0	5	Apagar
25	maria	Maria Soares Filho	masofi561@gmail.com	0	4	Apagar
26	carlos26	Carlos Aquino	caqui26@outlook.com	0	20	Apagar

A blue "Voltar" (Back) button is located at the bottom left of the table.

Fonte: Elaboração do autor

Ao clicar em “Apagar”, é exigida uma confirmação, como mostrado na Figura 27. Ao ser dada, inicialmente todas as suas imagens são apagadas, conforme o procedimento mostrado anteriormente. Em seguida todas as instâncias de “favorito” onde o ID_dono_Favorito corresponde ao ID do usuário a ser apagado são apagadas, bem como todas as instâncias de ‘comentario’ utilizando o ID_autor para fazer a correspondência. Só então a instância do usuário é apagada da tabela “usuario”. Devido à irreversibilidade deste processo, é exigida esta confirmação antes do usuário ser apagado.

Figura 27 - Confirmação para apagar usuário



Fonte: Elaboração do autor

6.4 INTERFACE TELEGRAM

Nesta seção são apresentadas as telas com as interações entre o usuário e o MyVoyagebot para Telegram. Todos os estados apresentados na seção 5.2 são visualizados através de mensagens recebidas pelo usuário no Telegram. As telas a seguir foram capturadas a partir do aplicativo Telegram para iOS.

Caso o usuário envie uma mensagem sem ter feito o cadastro do seu ID_Telegram, ele receberá um aviso de que deve inicialmente enviar um texto, como é visto na Figura 28.

Figura 28 - Imagem enviada sem cadastro do ID_Telegram



Fonte: Elaboração do autor

A conversa de texto com o MyVoyagebot pode ser iniciada enviando um texto qualquer. Na tela da Figura 29 a conversa foi iniciada com o texto “Abc”, mas

qualquer *string* de texto produziria o mesmo resultado. O importante é que o tipo da mensagem recebida pelo *bot* seja “texto”.

Figura 29 - Bot pergunta o nome do usuário



Fonte: Elaboração do autor

A *string* digitada é automaticamente transformada em *string* de letras minúsculas pelo software do *bot*, uma vez que este é o formato armazenado no banco de dados. Caso o nome do usuário digitado não exista no banco de dados, uma mensagem de erro é enviada pelo *bot*, a conversa é encerrada e ele retorna ao estado de esperar uma nova mensagem, como é mostrado na Figura 30.

Figura 30 - Nome do usuário não encontrado



Fonte: Elaboração do autor

Caso o nome do usuário seja encontrado no banco de dados, o *bot* pedirá que o usuário digite a senha, conforme a Figura 31. Este é o segundo passo do cadastro do ID_Telegram. Não foi possível esconder na tela os caracteres digitados pelo usuário, portanto este é um ponto sensível de segurança.

Figura 31 - Bot pede a senha



Fonte: Elaboração do autor

O bot combina a string do nome do usuário com a senha digitada e aplica a função *Hash MD5*. Este resultado é comparado com a senha do banco de dados. Caso a senha digitada esteja errada, uma mensagem de erro é enviada e a conversa é encerrada, como é visto na Figura 32.

Figura 32 - Senha digitada é inválida



Fonte: Elaboração do autor

Caso a senha digitada esteja correta, o *bot* atualiza no banco de dados o campo ID_Telegram do usuário e encerra a conversa, conforme está mostrado na Figura 33. A vantagem de se realizar o cadastro do ID_Telegram é que para enviar imagens pelo *bot* não é necessário fazer login no sistema. Sempre que uma conversa é encerrada o *bot* volta para o estado de escutar mensagens de novo usuário.

Figura 33 - Cadastro do ID_Telegram realizado



Fonte: Elaboração do autor

Após realizar o cadastro do ID_Telegram, caso o usuário envie uma mensagem de texto, ele receberá uma mensagem dizendo que ele deverá enviar apenas imagens, como mostra a Figura 34. As interações de texto entre o *bot* e o usuário são apenas para realizar o cadastro do ID_Telegram. E o cadastro no sistema é somente permitido pelo website.

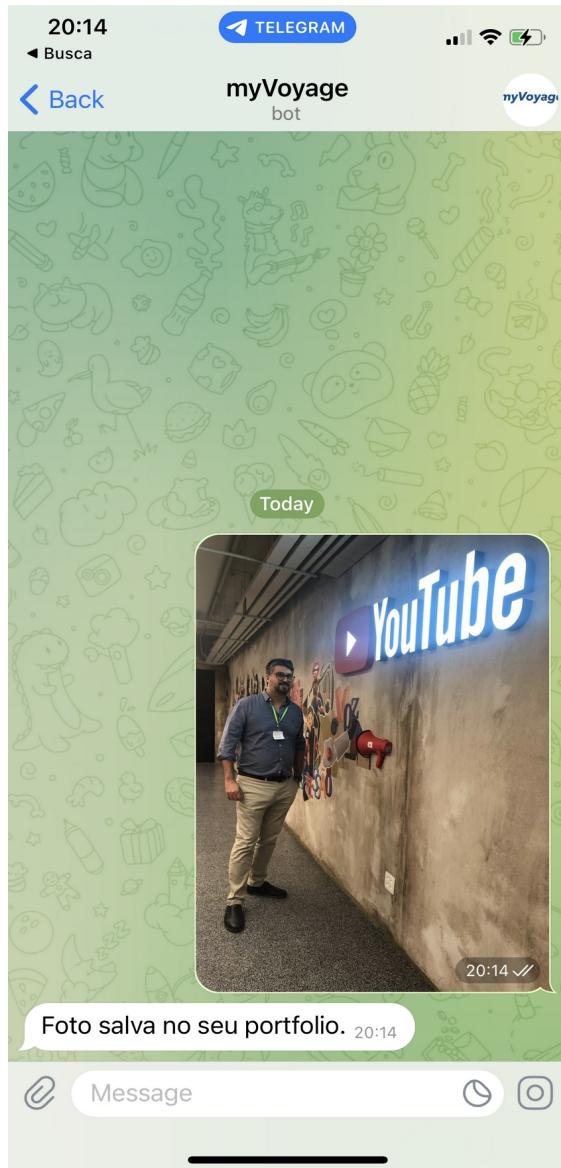
Figura 34 - Envio de texto após cadastro do ID_Telegram



Fonte: Elaboração do autor

Após realizar o cadastro do ID_Telegram basta que o usuário envie diretamente uma imagem ao *bot* para que a imagem seja colocada em seu portfolio. Como todas as mensagens recebidas pelo *bot* possuem o ID_Telegram do remetente, basta uma consulta ao banco de dados para saber se o remetente já está cadastrado. Em caso positivo a imagem é automaticamente salva em seu portfolio sem necessidade de fazer login, como é mostrado na Figura 35.

Figura 35 - Imagem salva no portfolio pelo MyVoyagebot



Fonte: Elaboração do autor

Caso a mensagem enviada seja do tipo “documento”, o *bot* envia uma mensagem padrão de erro para quem a enviou, e não chega a consultar o banco de dados, como é visto na Figura 36. Mensagens do tipo “documento” são envios de arquivos que não sejam imagens ou imagens enviadas pelo Telegram *desktop* em formato não comprimido.

Figura 36 - Formato inválido



Fonte: Elaboração do autor

7 CONSIDERAÇÕES FINAIS

O sistema MyVoyage permite armazenamento de imagens na nuvem, compartilhamento de fotos entre usuários, seleção de fotos favoritas para rápido acesso e opção de escrever comentários públicos em imagens próprias e de terceiros, portanto ele cumpre todos os objetivos propostos inicialmente. Com estas funções ele funciona como uma rede social básica. O uso do *bot* para Telegram facilita o envio de imagens para o portfolio de uma forma que não foi encontrada em nenhum sistema de armazenamento e/ou compartilhamento pesquisado. A simplicidade de uso do sistema permite que ele seja usado por pessoas de qualquer idade e nível de instrução.

As melhorias e possibilidades de expansão deste trabalho podem ser classificadas em duas vertentes. Uma vertente seria para melhorias nas páginas web que aumentem a funcionalidade do sistema, como permitir a criação de álbuns de imagens, permitir seguir usuários, sistema de notificações e troca de mensagens privadas entre usuários. Essas ideias podem servir como oportunidades para trabalhos futuros que podem ter o MyVoyage como base de desenvolvimento.

A outra vertente seria a expansão das possibilidades de uso do *bot* para o MyVoyage e como base para sistemas com outras finalidades. O MyVoyagebot poderia ser usado para enviar notificações, realizar cadastro completo ou buscar outros usuários, por exemplo. O uso de um *bot* para aplicativos de mensagens instantâneas integrado a um banco de dados permite inúmeras aplicações que podem ser implementadas a baixo custo e com grande potencial, principalmente na área de utilidades públicas. Uma prefeitura pode implementar um *bot* que recebe imagens de buracos de rua e calçadas, por exemplo, e assim pode mapeá-los e planejar seu reparo. Pode ser criado um boletim de ondas de surfe, onde os surfistas enviam ao *bot* fotos das ondas e o nome da praia naquele determinado dia, de forma que todos podem saber a condição de mar antes de ir até a praia. Outra possibilidade seria para ajudar pessoas sem teto, que estão morando nas ruas. Fotos de onde estas pessoas estão dormindo e sua localização seriam enviadas ao banco de dados da prefeitura para que o serviço social seja acionado. O *bot* pode fazer perguntas e assim coletar mais informações além da simples imagem.

REFERÊNCIAS BIBLIOGRÁFICAS

- 1 THE PSYCHOLOGY Behind Why We Share on Social Media". Disponível em <<https://www.shutterstock.com/blog/the-psychology-behind-why-we-share-on-social-media>>. Acesso em 19 de setembro de 2022.
- 2 TELEGRAM. Disponível em <<https://telegram.org>>. Acesso em 20 de outubro de 2022.
- 3 THE 7 different types of social media. Disponível em <<https://biteable.com/blog/the-7-different-types-of-social-media/>>. Acesso em 11 de outubro de 2022.
- 4 FACEBOOK. Disponível em <<https://www.facebook.com>>. Acesso em 20 de outubro de 2022.
- 5 INSTAGRAM. Disponível em <<https://www.instagram.com>>. Acesso em 17 de outubro de 2022.
- 6 WHATSAPP. Disponível em <<https://www.whatsapp.com>>. Acesso em 20 de outubro de 2022.
- 7 META. Disponível em <<https://www.meta.com>>. Acesso em 27 de setembro de 2022.
- 8 3 REASONS Why Meta's Social Media Dominance Should Scare You. Disponível em <<https://www.makeuseof.com/why-you-should-be-concerned-about-meta>>. Acesso em 13 de setembro de 2022.
- 9 DEFINITION Bot. Disponível em <<https://www.techtarget.com/whatis/definition/bot-robot>>. Acesso em 02 de outubro de 2022.
- 10 HOW CUSTOMER Service Chatbots are Transforming Support with AI. Disponível em <<https://www.revechat.com/blog/customer-service-chatbots/>>. Acesso em 13 de outubro de 2022.
- 11 TWITCH Chat & Chatbots. Disponível em <<https://dev.twitch.tv/docs/irc>>. Acesso em 20 de outubro de 2022.
- 12 BOTS are making Fortnite players question what's real. Disponível em <<https://www.polygon.com/2019/10/22/20926888/fortnite-bots-skill-based-matchmaking-epic-games>>. Acesso em 15 de outubro de 2022.
- 13 VISÃO Geral dos Rastreadores do Google. Disponível em <<https://developers.google.com/search/docs/advanced/crawling/overview-google-crawlers>>. Acesso em 20 de outubro de 2022.
- 14 ALEXA Beats Google Assistant and Siri in Voice Assistant Popularity as Voice AI Market Expands. Disponível em <<https://voicebot.ai/2022/06/24/alex Beats google>>. Acesso em 20 de outubro de 2022.

assistant-and-siri-in-voice-assistant-popularity-as-voice-ai-market-expands/>. Acesso em 10 de outubro de 2022.

- 15 ECHO Dot. Disponível em <https://www.amazon.com.br/dp/B084DWCZY6/ref=pd_sl_7ba9axz4d2_e&tag=hydrbrgk-20>. Acesso em 29 de setembro de 2022.
- 16 AMAZON. Disponível em <<https://www.amazon.com>>. Acesso em 20 de outubro de 2022.
- 17 APPLE Siri. Disponível em <<https://www.apple.com/siri/>>. Acesso em 18 de setembro de 2022.
- 18 APPLE. Disponível em <<https://www.apple.com>>. Acesso em 20 de outubro de 2022.
- 19 HOW does Machine Learning Work in AI Chatbots ?. Disponível em <<https://www.visor.ai/blog/machine-learning-ml-ai-chatbots/>>. Acesso em 10 de outubro de 2022.
- 20 THE BENEFITS of instant messaging in business communication. <<https://pumble.com/blog/benefits-of-instant-messaging-in-business-communication/>>. Acesso em 13 de novembro de 2022.
- 21 WHATSAPP é o aplicativo mais usado pelos brasileiros; confira a lista. Disponível em <<https://g1.globo.com/tecnologia/noticia/2022/01/11/whatsapp-e-o-aplicativo-mais-utilizado-por-metade-dos-brasileiros-confira-a-lista.ghtml>>. Acesso em 19 de outubro de 2022.
- 22 TELEGRAM expande número máximo de pessoas em grupo para 200 mil. Disponível em <<https://g1.globo.com/economia/tecnologia/noticia/2019/01/22/telegram-expande-numero-maximo-de-pessoas-em-grupo-para-200-mil.ghtml>>. Acesso em 18 de outubro de 2022.
- 23 BOTS: An Introduction for Developers. Disponível em <<https://core.telegram.org/bots>>. Acesso em 10 de outubro de 2022.
- 24 FLICKR. Disponível em <<https://www.flickr.com>>. Acesso em 15 de setembro de 2022.
- 25 500PX. Disponível em <<https://500px.com>>. Acesso em 15 de setembro de 2022.
- 26 DEVIANT Art. Disponível em <<https://www.deviantart.com>>. Acesso em 16 de setembro de 2022.
- 27 SMUGMUG. Disponível em <<https://www.smugmug.com>>. Acesso em 17 de setembro de 2022.

- 28 TREKEARTH. Disponível em <<https://www.trekearth.com>>. Acesso em 18 de setembro de 2022.
- 29 GOOGLE. Disponível em <<https://www.google.com>>. Acesso em 20 de outubro de 2022.
- 30 GOOGLE Photos. Disponível em <<https://www.google.com/photos/about/>>. Acesso em 18 de setembro de 2022.
- 31 ANDROID. Disponível em <<https://www.android.com>>. Acesso em 02 de outubro de 2022.
- 32 SHUTTERFLY. Disponível em <<https://www.shutterfly.com>>. Acesso em 18 de setembro de 2022.
- 33 ICLOUD. Disponível em <<https://www.icloud.com>>. Acesso em 20 de outubro de 2022.
- 34 IOS 16. Disponível em <<https://www.apple.com/ios/ios-16/>>. Acesso em 20 de outubro de 2022.
- 35 IPADOS 16. Disponível em <<https://www.apple.com/br/ipados/ipados-16-preview/>>. Acesso em 20 de outubro de 2022.
- 36 WHY smartphones are not a threat to professional photography. Disponível em <<https://www.dw.com/en/why-smartphones-are-not-a-threat-to-professional-photography/a-37063871>>. Acesso em 20 de setembro de 2022.
- 37 9 REASONS your website doesn't work on mobile. Disponível em <<https://www.vertical-leap.uk/blog/9-reasons-website-doesnt-work-mobile/>>. Acesso em 20 de setembro de 2022.
- 38 APP Maintenance: Why and How Often You Should Update Your App. Disponível em <<https://agilie.com/blog/app-maintenance-why-and-how-often-you-should-update-your-app>>. Acesso em 10 de outubro de 2022.
- 39 XEXÉO, G. "Modelagem de Sistemas de Informação. Da análise de requisitos ao modelo de interface". [S.I.]: Creative Commons, 2007
- 40 PMI (2017) "Um Guia do Conhecimento em Gerenciamento de Projetos (Guia PMBOK)". 6^a Edição, Newton Square, PA
- 41 LARMAN, C. "Utilizando UML e Padrões" 3^a Edição, Porto Alegre: Bookman Editora, 2007

- 42 WHAT IS Unified Modeling Language (UML) ?. Disponível em <<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/>>. Acesso em 17 de novembro de 2022.
- 43 STATE Machine Diagrams. Disponível em <<https://www.uml-diagrams.org/state-machine-diagrams.html>>. Acesso em 17 de novembro de 2022.
- 44 YOURDON, E. “Análise Estruturada Moderna” 3^a. Edição, Rio de Janeiro: Elsevier Editora, 1990
- 45 VISUAL Studio Code. Disponível em <<https://code.visualstudio.com/>>. Acesso em 10 de outubro de 2022.
- 46 BOOTSTRAP. Disponível em <<https://getbootstrap.com/>>. Acesso em 16 de outubro de 2022.
- 47 XAMPP. Disponível em <https://www.apachefriends.org/pt_br/index.html>. Acesso em 16 de outubro de 2022.

GLOSSÁRIO

Back-end:

Parte de um sistema computacional ou aplicativo que não é diretamente acessada pelo usuário, sendo normalmente responsável pelo armazenamento e processamento de dados.

Bootstrap:

Pacote para customização de estilos CSS para uso em páginas HTML.

Bot:

Software criado para interagir com seres humanos como se fosse outro ser humano.

Captcha:

Teste usado em computação para determinar se quem o está realizando é um ser humano.

Chat:

Troca de mensagens entre seres humanos ou entre seres humanos e bots.

Chat-bot:

Software criado especificamente para interagir com seres humanos através de troca de mensagens.

Crawler:

Software criado para visitar websites automaticamente para coletar dados.

Download:

Recebimento de dados de outro computador ou servidor.

Front-end:

Parte de um sistema computacional ou aplicativo que é diretamente acessada pelo usuário.

Gold Note Writer:

Escritor Nota Dourada – título concedido a certos usuários pelo website TrekEarth.

Gold Star Critiquer:

Crítico Estrela Dourada – título concedido a certos usuários pelo website TrekEarth.

Hash:

Processo de transformar uma chave ou cadeia de caracteres em outro valor que não é facilmente reconvertido para o valor original.

Log:

Registro de anotações.

Loop:

Termo utilizado em programação para representar uma sequência de operações que se repete.

Mouse pad:

Tapete para uso com o mouse do computador que permite deslize suave.

Path:

Cadeia de caracteres usada para identificar unicamente uma localidade dentro de uma estrutura de diretórios.

PhpMyAdmin:

Software livre escrito em PHP para gerenciar banco de dados tipo MySQL via web.

Pixel:

Unidade básica de cor programável numa imagem computacional.

Polling:

Processo de monitoramente repetitivo de um valor para detecção de alterações.

Python:

Linguagem de programação de alto nível que inclui tipagem dinâmica e orientação a objetos.

Quest:

Palavra em inglês que significa busca por alguma coisa.

Script:

Sequencia de instruções que é interpretada ou executada por outro programa ao invés do processador (como um programa compilado).

Stakeholder:

Indivíduo, grupo ou organização que é interessado na implementação de um determinado projeto.

String:

Sequência de caracteres.

Tag:

Palavra-chave usada para classificar imagens digitais, videos ou arquivos.

Third-party:

Termo em inglês que significa terceiros.

Upload:

Envio de dados para outro computador ou servidor.

Web:

Sistema interconectado de páginas públicas acessíveis através da Internet, também conhecido como World Wide Web (WWW).

Webhook:

Função baseada em HTTP que permite comunicação a partir de determinados eventos entre duas aplicações rodando em servidores diferentes.