

Lovebox Web - Sistema de gerência web de medicamentos

Angélica S. Mizutani¹, Eduarda S. B. Silva¹, Laiza G. Silva¹,
Carlos D. S. Nunes¹, Luis F. Brasil¹, Vitor F. Uemura¹,
Rafael H. D. Zottesso¹, Eduardo H. M. Cruz¹,
Daniela E. Flôr¹, Linnyer B. R. Aylon²

¹Instituto Federal do Paraná – campus Paranavaí – Brasil
Av. José Felipe Tequinha, 1400 - Jardim das Nações, Paranavaí - PR

²Universidade Estadual de Maringá
Av. Colombo, 5790 - Zona 7, Maringá - PR, 87020-900

{angelicamizutani, dudaschuroff.b, laisags06, carlosnunes9852}@gmail.com

luis4756@live.com, vitoruemura@hotmail.com

{rafael.zottesso, eduardo.cruz, daniela.flor}@ifpr.edu.br

lbruiz@uem.br

Abstract. *This work aims to present a possible solution to increase patient adherence to polypharmaceutical treatments. The web platform will work in conjunction with an automated drug dispenser for patients under polypharmacy. The web interface, developed in the Python programming language, has the main functionality to serve as the medication box management interface, allowing users to perform the necessary registrations regarding their treatments or those of their dependents, so that the dispenser can be used. In the usage simulations, the platform fulfilled its purpose, including the integration with the physical prototype. Such a solution can serve as a way to increase adherence to treatments, especially in the elderly population.*

Resumo. *Este trabalho tem por objetivo apresentar uma possível solução para aumentar a adesão de pacientes a tratamentos polifarmacêutico. A plataforma web funcionará em conjunto com um dispensador automatizado de medicamentos para pacientes em polifarmácia. O sistema web, desenvolvida na linguagem de programação Python, tem por principal funcionalidade servir como a interface de gerenciamento da caixa de medicamentos, permitindo que os usuários realizem os cadastros necessários referentes aos seus tratamentos ou de seus dependentes, para que o dispensador possa realizar seu funcionamento. Nas simulações de uso, a plataforma cumpriu seu propósito, incluindo a integração com o protótipo físico. Tal solução pode servir como uma maneira de aumentar a adesão aos tratamentos, principalmente na população idosa.*

1. Introdução

Nas últimas décadas, a proporção da população idosa não só do Brasil mas também Global, vem crescendo em ritmo acelerado. De acordo com a ONU (Organização das Nações Unidas) em 2019, estima-se que em todo o mundo haviam aproximadamente 703 milhões

de idosos com 65 anos ou mais. A atual projeção é que tal número mais do que dobre até 2050 atingindo a quantidade de 1,5 bilhões [Alves 2019].

A população idosa também é faixa etária que apresenta proporcionalmente uma maior chance de desenvolver múltiplas doenças crônicas, como diabetes, doenças cardiovasculares, derrames, câncer e infecções. De tal forma, a porção idosa da população é a principal em situação de tratamentos polifarmacêuticos (tratamentos que envolvam o uso rotineiro de quatro ou mais medicamentos)[PENTEADO et al. 2002].

Diante de tal realidade, esse trabalho tem por objetivo desenvolver uma plataforma web, que funcionará em conjunto com um dispensador automatizado de medicamentos para pacientes em polifarmácia, assim servindo como uma possível solução para a necessidade de formas de aumentar a adesão de pacientes à polifarmácia, em especial a população idosa.

O dispositivo, nomeado como Lovebox, receberá do sistema web as informações referente as doses do tratamento, este então deverá emitir um alerta visual e sonoro no horário estabelecido para cada dose, alertando o paciente que este deverá tomar seu medicamento. O display visual LCD informará ao paciente a dosagem e o nome do medicamento a ser ingerido. Ademais, no interior da caixa, cada compartimento conterá um LED que será acionado no horário das doses para auxiliar o usuário a encontrar o medicamento correto.

O sistema web da Lovebox, desenvolvido com o framework Django, na linguagem de programação Python, será responsável pelos cadastros necessários para o funcionamento do sistema, como paciente, cuidador, medicamento e tratamento. A plataforma web também será responsável pela sincronização do protótipo físico com as doses de tratamentos cadastrados no banco de dados.

Além da sua função na integração com o protótipo físico, o sistema web também tem o objetivo de promover a segurança e controle no acompanhamento médico dos tratamentos, uma vez que o cuidador/familiar/profissional de saúde responsável pelo paciente poderá ter acesso ao histórico de doses, permitindo que eles realizem o monitoramento da ingestão de medicamentos pelos pacientes.

Este trabalho é organizado da seguinte forma. A Seção 2 apresenta uma pesquisa das principais soluções similares à proposta encontradas, tanto no meio acadêmico quanto no ambiente comercial, comparando as principais funcionalidades de cada uma destas. Na Seção 3, estão listadas as principais tecnologias usadas para o desenvolvimento do trabalho e suas principais aplicações e também os principais diagramas e etapas desenvolvidos no decorrer do trabalho. Na Seção 4, são apresentados os resultados obtidos do trabalho até o atual momento. Na Seção 5, discorre-se sobre a conclusão do trabalho, realizando uma análise da proposta inicial com o resultado obtido e também perspectivas para o futuro.

2. Trabalhos correlatos

A polifarmácia e a não adesão ao tratamento medicamentoso são realidades frequentes entre a população idosa, sendo que a segunda se caracteriza como um perigo para esta parcela da população, visto que pode acarretar problemas de saúde e trazer impactos financeiros. Nesse sentido, é evidente a necessidade do desenvolvimento de ferramentas

que contribuam para a melhor aderência ao tratamento e, assim, melhorem a qualidade de vida da população idosa usuária de polifarmácia.

Por meio de uma pesquisa, foi possível identificar propostas de soluções acadêmicas e comerciais, não apenas para a população idosa polimedicada, mas também para a população em geral que faz uso de medicações constantemente.

Tabela 1. Soluções disponíveis atualmente.

Nome	Número de compartimentos aceitos	Medicamentos aceitos	Tecnologia	Solução	Preço
Supermed [Vieira et al. 2016]	31 sachês	Comprimidos	Arduino Uno	Acadêmica	Não cotado
Caixa inteligente de lembrete de medicamento [Bhati et al. 2017]	Não especificado pelos autores	Comprimidos	IoT	Acadêmica	Não cotado
Caixa de medicamentos inteligente baseada em IOT [Bhavya et al.]	Não especificado pelos autores	Comprimidos	IoT	Acadêmica	Não cotado
Sistema de lembrete e monitoramento de comprimidos baseado em IOT [Ahmad 2020]	Não especificado pelos autores	Comprimidos	IoT	Acadêmica	Não cotado
MEDIBOX [Ilakiya et al. 2019]	Não especificado pelos autores	Comprimidos	IoT	Acadêmica	Não cotado
HERO ¹	10 medicamentos diferentes	Comprimidos	IoT	Comercial	US\$ 29,99 a US\$ 44,99 por mês
Alertmed ²	28 compartimentos	Comprimidos	Não informado	Comercial	R\$ 280,00
CompuMed ³	Não especificado pela empresa	Comprimidos	Não informado	Comercial	US\$ 695,00
Pria ⁴	28 compartimentos	Comprimidos	IoT	Comercial	US\$ 245,00 com plano de assinatura de US\$ 9,99
Station ⁵	28 compartimentos	Comprimidos	Não informado	Comercial	US\$ 389,95

A Tabela 1 apresenta uma síntese das soluções, acadêmicas e comerciais, encontradas na pesquisa. Tais propostas, apesar de buscarem resolver a questão da não adesão ao tratamento medicamentoso, possuem certas limitações, como, por exemplo, em relação ao tipo de medicação que são capazes de armazenar. Nesse sentido, a Lovebox está sendo desenvolvida para proporcionar uma solução mais completa para o problema.

3. Lovebox Web – Sistema de gerencia web de medicamentos

Nesta seção, serão apresentadas as principais etapas realizadas no desenvolvimento da plataforma web para o dispensador automatizado de medicamentos para pacientes em polifarmácia. Serão apresentados diagramas, com o objetivo de facilitar a compreensão do escopo do trabalho e suas principais funcionalidades.

3.1. Tecnologias Utilizadas

Nesta seção, serão descritas as tecnologias usadas e uma breve explicação de quais foram suas aplicações no desenvolvimento da plataforma web.

3.1.1. Linguagem de programação Python

Desenvolvida por Guido Van Rossum, Python é uma linguagem de programação interpretada de alto nível, orientada a objetos, que possui uma semântica dinâmica, tendo como

prioridade uma fácil compreensão. Esta linguagem está sendo utilizada na escrita dos comandos de funcionamento do protótipo.

3.1.2. Framework Django

Django é um framework gratuito e de código aberto que proporciona desenvolvimento rápido e design limpo e pragmático. Este framework, implantado em um servidor web, ajuda desenvolvedores a produzirem rapidamente um frontend web rico em recursos, seguro e escalável, representando um meio muito eficiente para a construção de um aplicativo web. O framework Django está sendo utilizado no desenvolvimento do sistema web da Lovebox.

O framework Django também funciona muito bem com a modularização. No Django, estes módulos são chamados de *Apps*. Em um projeto, cada um destes *Apps* terá seu próprio diretório com os arquivos de configuração. O sistema web Lovebox foi dividido nos seguintes módulos: cadastros (responsável pelo gerenciamento das classes), páginas (responsável pelas telas que não pertencem a alguma classe em específico) e usuários (responsável pela parte de autenticação).

3.1.3. PostgreSQL

Devido à necessidade de um servidor para a integração do dispositivo físico com o sistema web, o sistema gerenciador de banco de dados (SGBD) PostgreSQL⁶ foi utilizado. Este é um sistema de base de dados objeto-relacional de código aberto, possuindo mais de 30 anos de desenvolvimento ativo, adquirindo uma forte reputação em relação a confiabilidade, robustez e performance.

3.1.4. Visual Studio Code

O Visual Studio Code⁷ ou VS Code é um editor de código gratuito que foi desenvolvido pela Microsoft. Este conta com suporte a depuração, versionamento através do git e diversas extensões que permitem um funcionamento similar a de uma interface de desenvolvimento. O VS Code foi utilizado no desenvolvimento do sistema web como uma maneira de a o acesso às diversas pastas e arquivos do projeto em Django.

3.2. Diagrama de Caso de Uso

O diagrama de casos de uso tem por objetivo representar em um alto nível de abstração os principais atores, casos de uso e seus relacionamentos dentro de um sistema. A Figura 1 conta com o diagrama de caso de uso elaborado para o sistema web da Lovebox. Para sua correta interpretação, é preciso ter em mente que onde aparece o termo gerenciar esta referindo-se ao cadastro, listagem, alteração e exclusão de registros.

Conforme é mostrado na Figura 1, o diagrama de caso de uso conta com quatro atores, sendo estes Paciente, Cuidador e Administrador, que herdaram de Usuário. Tal

⁶<https://www.postgresql.org/>

⁷<https://code.visualstudio.com/>

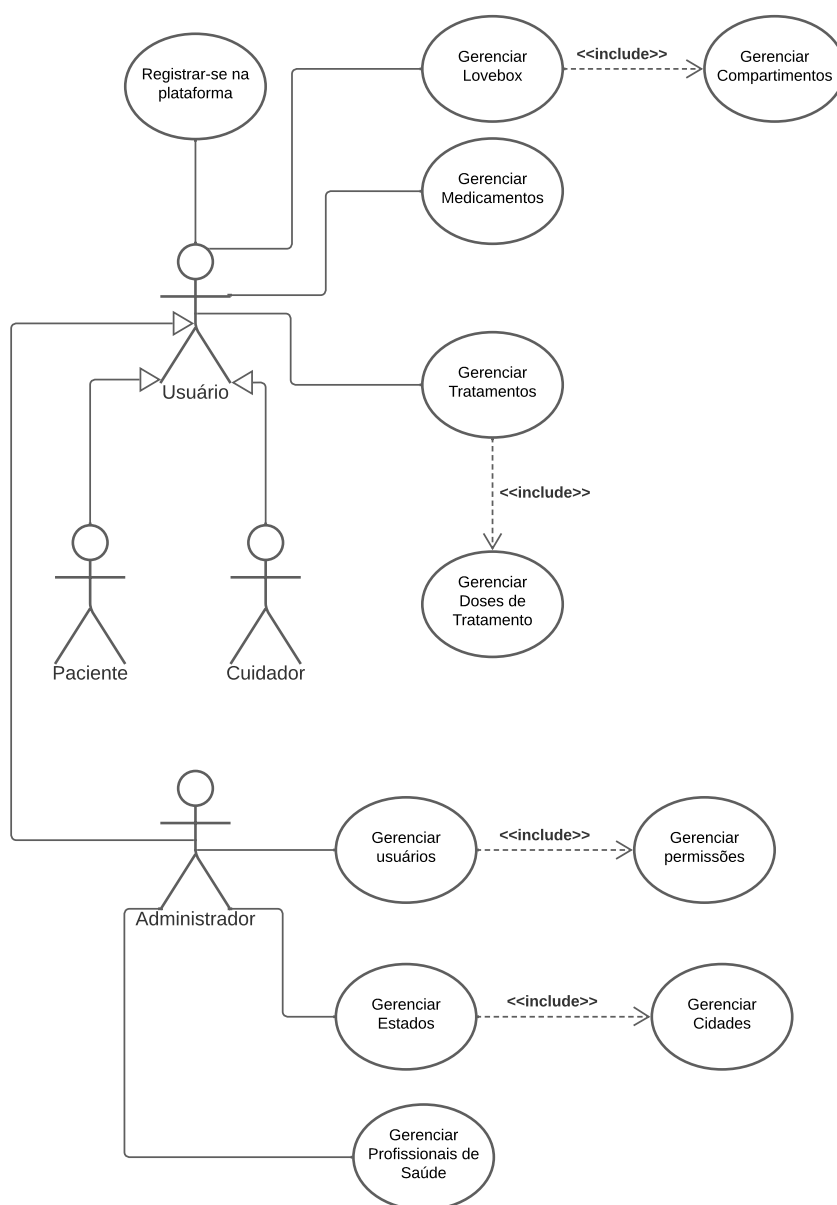


Figura 1. Diagrama de Caso de Uso.

herança se da ao fato de um paciente poder ser responsável pelo gerenciamento de seus próprios tratamentos ou essa responsabilidade ser delegada a um cuidador. Por fim, o Administrador herda os casos de uso de Usuário e também conta com outros referentes à parte de classes não acessíveis a usuários comuns e também o gerenciamento dos usuários.

3.3. Diagrama de Classes

O diagrama de classes tem por objetivo definir as estruturas de cada uma das classes pertencentes ao sistema, tais como seus atributos, procedimentos e a relação entre estas. Para o sistema web da Lovebox, o diagrama de classes desenvolvido conta com 12 classes necessárias para cumprir o objetivo do projeto. O diagrama de classes pode ser observado na Figura 2.

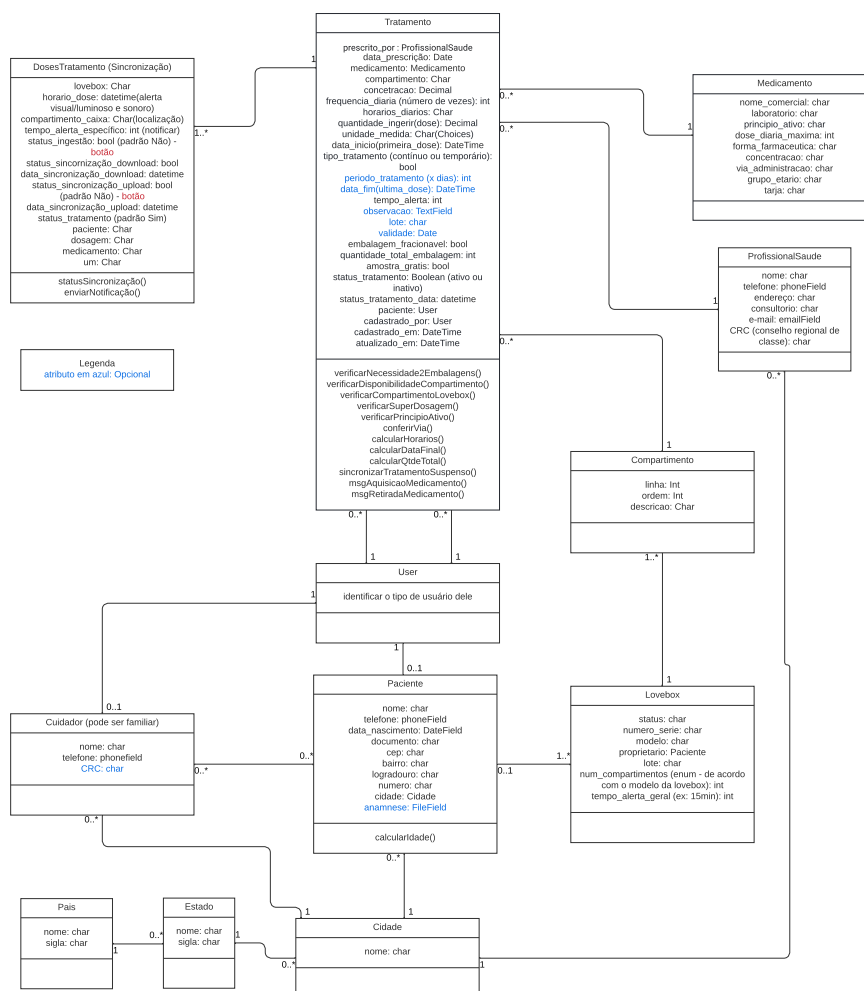


Figura 2. Diagrama de Classes da Lovebox.

As classes País, Estado e Cidade estão relacionadas entre si de modo a relacionar várias cidades com um estado, e vários estados com um país. Estas classes servem como parte do endereço para o cadastro dos usuários, desta maneira a classe cidade também está relacionada com as classes Cuidador, Paciente e ProfissionalSaude, de modo que cada um desses indivíduos pertençam a uma cidade.

As classes Cuidador e Paciente estão em um relacionamento muitos-para-muitos. De modo que um cuidador possa ser responsável por pelo menos um ou vários pacientes e um paciente esteja sobre o cuidado de nenhum ou vários cuidadores. Como nem sempre os pacientes terão a necessidade de auxílio na gestão de sua Lovebox, não é obrigatório a relação com um cuidador. A classe User tem por principal objetivo estabelecer os atributos e procedimentos compartilhados entre ambos Cuidadores e Pacientes, com foco na questão da autenticação com o sistema, sendo assim um usuário que acessa o sistema pode ser um paciente ou um cuidador.

A classe Lovebox se relaciona com a classe Paciente, de modo com que um paciente seja proprietário de uma ou várias Lovebox e uma Lovebox pertença a um único Paciente. Além disso esta classe se relaciona com a classe Compartimento que tem por

objetivo endereçar e ordenar os compartimentos pertencentes a uma Lovebox.

A classe Tratamento é a principal classe no funcionamento do sistema, ela se relaciona com a classe User de modo a relacionar quem é o Paciente e o Cuidador de tal tratamento. Sua relação com o compartimento tem por objetivo definir qual será o compartimento e consequentemente a Lovebox utilizado naquele tratamento. Esta classe também se relaciona com a classe ProfissionalSaude, que tem por propósito o cadastro de profissionais de saúde que prescreverão os tratamentos. E por fim ela também se relaciona com a classe Medicamento que é a classe que armazena as informações dos medicamentos que serão atribuídos à tratamentos.

Já a classe DosesTratamento é a principal classe responsável pela integração do sistema web com o protótipo físico. Esta tem um relacionamento de um para muitos com a classe Tratamento de modo que ao criar-se um tratamento seja gerado as doses do mesmo. Para que a integração ocorra, o protótipo físico irá acessar o banco de dados compartilhado com o sistema web e, na classe DosesTratamento, terá acesso a todas as informações necessárias para o seu funcionamento em cada uma das doses de um tratamento.

3.4. Criação das Entidades

Devido à utilização do framework Django, a modelagem de dados a partir de um Diagrama Entidade Relacionamento (DER) foi dispensada, uma vez que o próprio framework tem a funcionalidade de gerar o projeto do banco de dados através das declarações das classes no arquivo models.py. Segue como exemplo o código da declaração da classe Medicamento.

Listing 1. Declaração da Classe Medicamento

```
from django.db import models
from django.contrib.auth.models import User
from phonenumber_field.modelfields import PhoneNumberField

class Paciente(models.Model):
    user = models.ForeignKey(
        User, on_delete=models.PROTECT)
    nome = models.CharField(max_length=100)
    telefone = PhoneNumberField(region='BR')
    data_nascimento = models.DateField(
        verbose_name='Data de Nascimento')
    documento = models.CharField(max_length=14)
    cep = models.CharField(max_length=9)
    cidade = models.ForeignKey(
        Cidade, on_delete=models.PROTECT)
    bairro = models.CharField(max_length=255)
    logradouro = models.CharField(max_length=255)
    numero = models.IntegerField()
    anamnese = models.FileField(blank=True, null=True)
    redes_sociais = models.CharField(
        max_length=255, blank=True, null=True,
        verbose_name='Redes Sociais')
```



```
cuidadores = models.ManyToManyField(Cuidador)
```

```
def __str__(self):
    return "{}".format(self.nome)
```

Para realizar a declaração, é necessário importar o *models* da biblioteca *django.db* que é a responsável pelo gerenciamento do banco de dados no Django. Então declara-se o nome da classe que recebe como parâmetro o *models.Model* e dentro do código da classe seus atributos. Estes podem ser de diferentes tipos, por exemplo no atributo *user* utiliza-se o *models.ForeignKey*, ou seja, uma chave estrangeira, que recebe como parâmetro a classe *User*, nativa do Django.

Quando definimos uma chave estrangeira, é necessário definir qual procedimento será adotado ao tentar deletar um objeto que está sendo utilizado como uma chave estrangeira. No caso de *Paciente* e *User*, foi utilizado o *on_delete = models.PROTECT*, que significa que quando houver a tentativa de deletar um usuário que esteja vinculado a um paciente não será possível, sendo necessário remover a relação ou o registro do paciente.

A classe *Paciente* também contém um campo *cuidadores*, do tipo *ManyToMany* (Muitos-para-muitos), sinalizando a relação entre paciente e cuidadores. Para implementar tal relação, o Django cria uma tabela adicional no banco de dados, com o nome paciente-cuidador, com duas chaves estrangeiras, uma de paciente e outra de cuidador. Por fim, o Django interpreta toda a classe e gera uma tabela em SQL.

4. Resultados parciais e Discussão

Uma versão inicial e funcional do sistema web já se encontra pronta, sendo possível realizar todos os cadastros necessários para a criação de um tratamento, incluindo também o gerenciamento desses registros como a alteração, exclusão e listagem dos mesmos. A Figura 3 mostra todos os cadastros que podem ser acessados no sistema. O formulário de cadastros de medicamentos é exibido na Figura 4.

Além disso, uma versão inicial do sistema de autenticação já está implementada, permitindo o registro de usuários, *login* e *logout*. Em conjunto com o módulo de autenticação, o sistema também já possui parte do controle de permissões, de modo que somente os administradores tenham acesso a certas classes. A tela de autenticação de usuários é exibida na Figura 5.

A Figura 6 mostra um cuidador fictício acessando a lista de tratamentos. Como a associação de objetos com base no usuário autenticado já foi implementada no gerenciamento de tratamentos, o usuário só consegue ver os tratamentos onde este é o cuidador. No exemplo, o cuidador tem acesso a um tratamento, onde está especificado na lista o paciente e o medicamento do tratamento em questão.

O sistema web também já foi integrado com o protótipo físico de modo em que ambos compartilham o mesmo banco de dados. Sendo assim, já é possível cadastrar todo o tratamento de paciente e suas dependências pela interface web, assim gerando as doses que serão enviadas para o protótipo físico.



Figura 3. Lista de cadastros do sistema.

Figura 4. Formulário para cadastro de medicamentos.

5. Conclusão e Perspectivas Futuras

Em um contexto onde a proporção da população idosa mundial segue em crescimento acelerado, é perceptível a importância e necessidade de uma maneira que ajudasse pessoas, principalmente idosos, a tomarem seus medicamentos na rotina correta. Esse projeto, em conjunto com o protótipo físico da caixa de medicamento, se propôs a solucionar tal demanda.

Em conformidade com tal proposta, o trabalho desenvolvido conseguiu cumprir seu propósito. É possível aos cuidadores e pacientes cadastrarem seus tratamentos na interface web, gerando as doses que serão armazenadas no banco de dados Postgre com-

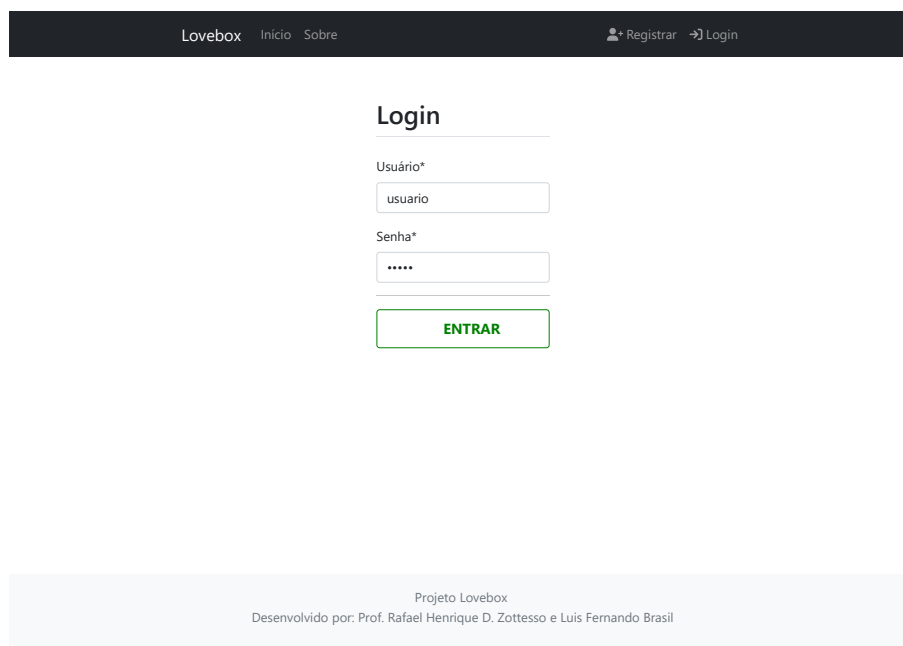
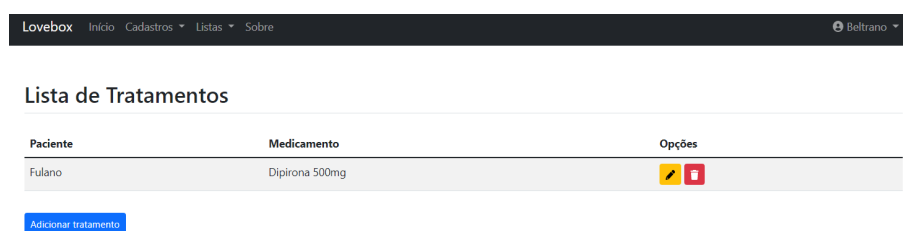


Figura 5. Tela de autenticação de usuários.



Paciente	Medicamento	Opções
Fulano	Dipirona 500mg	 

[Adicionar tratamento](#)

Figura 6. Cuidador acessando a lista de tratamentos.

partilhado com o dispensador automático, que irá receber as informações referentes a essas doses, podendo assim realizar seu funcionamento adequado.

No futuro, deverá ser incrementada a acessibilidade do sistema, de modo que permita aos usuários gerirem seus tratamentos de maneira mais rápida e intuitiva com foco na população idosa. No intuito de que seja mais comum usuários que consigam gerenciar seus próprios tratamentos. Também é de extrema importância a elaboração de relatórios das doses de cada paciente, de modo que cuidadores e profissionais médicos tenham maior precisão na tomada de decisões sobre os tratamentos.

Agradecimentos

O presente trabalho foi realizado com apoio do programa PIBIC (DIPE/PROEPPI/IFPR e CNPq), MannaTeam, PRADI (AGIF/PROEPPI/IFPR), PIBITI (AGIF/PROEPPI/IFPR e Fundação Araucária), CNPq Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil (311685/2017-0) e da Fundação Araucária.

Referências

[Ahmad 2020] Ahmad, S. (2020). Iot based pill reminder and monitoring system.

- [Alves 2019] Alves, J. E. D. (2019). Envelhecimento populacional no brasil e no mundo. *Revista Longeviver*.
- [Bhati et al. 2017] Bhati, S., Soni, H., Zala, V., Vyas, P., and Sharma, Y. (2017). Smart medicine reminder box. *IJSTE-International Journal of Science Technology & Engineering*, 3(10):172–177.
- [Bhavya et al.] Bhavya, K., Pradeepa, B., Anandhapadmanaban, S., Ashifa, A., Kumar, S. S., and Suryalakshmi, R. An iot based smart medicine box for medication.
- [Ilakiya et al. 2019] Ilakiya, D., Niroopa, K., Priscilla, A., Vigneswari, B., and Mrs.sneha, k. (2019). A health-iot platform based on the integration of intelligent packaging, unobtrusive bio-sensor and intelligent medicine box. *SSRG International Journal of Computer Science and Engineering (SSRG-IJCSE)*.
- [PENTEADO et al. 2002] PENTEADO, P. d. S., Cunico, C., Oliveira, K., and Polichuk, M. (2002). O uso de medicamentos por idosos. *Visão acadêmica*, 3(1).
- [Vieira et al. 2016] Vieira, L. B., Ramos, C. d. Á., Castello, M. d. B., and Nascimento, L. C. d. (2016). Desenvolvimento de um dispositivo eletrônico para organizar medicamentos e promover a adesão medicamentosa. *Revista Panamericana de Salud Pública*, 39:208–212.