

CENTRO UNIVERSITÁRIO LA SALLE DO RIO DE JANEIRO

Unilasalle-RJ

Sistemas de Informação

Felipe Lima Guimarães

Sis-Atleta:

Uma ferramenta para análise de desempenho dos atletas de futebol

Niterói 2017

CENTRO UNIVERSITÁRIO LA SALLE DO RIO DE JANEIRO

Unilasalle-RJ Sistemas de informação

Sis-Atleta:

Uma ferramenta para análise de desempenho dos atletas de futebol

FELIPE LIMA GUIMARÃES

Monografia apresentada ao curso de Sistemas de Informação para obtenção do certificado de graduação.

Orientador: Prof. MSc. Alex Vanderlei Salgado

Niterói 2017

FELIPE LIMA GUIMARÃES

Sis-Atleta

Monografia apresentada ao curso de Sistemas de Informação para obtenção do certificado de graduação.

Aprovada em julho de 2017.

BANCA EXAMINADORA

Prof.Orientador(a). Alex Vanderlei Salgado Centro Universitário La Salle do Rio de Janeiro

Prof. MSc. XXX

Centro Universitário La Salle do Rio de Janeiro

Prof. MSc. XXX  
Centro Universitário La Salle do Rio de Janeiro

Niterói 2017

AGRADECIMENTOS

À minha família, que sempre me apoiou.  
Aos professores por toda a dedicação, auxílio e amizade.

A Marley Coutinho Lima

Sis-Atleta:

Uma ferramenta para análise de desempenho dos atletas de futebol

Felipe Lima Guimarães

# RESUMO

Analista de desempenho é uma profissão que vem ganhando destaque no esporte, principalmente no futebol. Atualmente, nas bases dos clubes, estes profissionais não possuem a atenção necessária para que seu trabalho seja realizado conforme o esperado. O projeto visa o desenvolvimento de um sistema que auxilie o trabalho dos analistas de desempenho a possuir os dados dos atletas de forma que haja um apoio em suas tomadas de decisão, em plataforma Web, utilizando as mais novas tecnologias do mercado e com uma interface amigável para facilitação do uso. Neste trabalho, será apresentado todo o processo de modelagem e construção, com as documentações e diagramas utilizados, além das ferramentas de apoio, abordando sua forma de utilização. Ao final, são mostrados os resultados obtidos através das principais telas criadas e suas funcionalidades.

Palavras-chave: Futebol, Estatísticas, Táticas, Web.

# ABSTRACT

Performance analyst is a profession that has been gaining prominence in sports, especially in soccer. Currently, at the club bases, these professionals do not have the necessary attention so that their work is carried out as expected. The project aims to develop a system that helps the work of performance analysts to possess the data of the athletes so that there is support in their decision making, in a Web platform, using the latest technologies on the market and a friendly interface for ease of use. In this work, the whole modeling and construction process will be presented, with the documentation and diagrams used, as well as the support tools, addressing its use. At the end, the results obtained through the main screens and their functionalities are shown.

Keywords: Soccer, Statistics, Tatics, Web.

[Figura 1: Diagrama de caso de uso 21](#_bookmark0)

[Figura 2: Diagrama de classe conceitual 37](#_bookmark1)

[Figura 3: Diagrama de sequência – Cadastrar Usuário 38](#_bookmark2)

[Figura 4: Diagrama de sequência – Realizar login padrão 39](#_bookmark3)

[Figura 5: Diagrama de sequência – Realizar login Google 40](#_bookmark4)

[Figura 6: Diagrama de sequência – Selecionar bulário 41](#_bookmark5)

[Figura 7: Diagrama de sequência – Selecionar diagnósticos 42](#_bookmark6)

[Figura 8: Diagrama de sequência – Solicitar reset de senha 43](#_bookmark7)

[Figura 9: Diagrama de objetos 44](#_bookmark8)

[Figura 10: Diagrama Molic – Apresentar tela de login 45](#_bookmark9)

[Figura 11: Diagrama Molic – Apresentar menu principal 46](#_bookmark10)

[Figura 12: Diagrama Molic – Apresentar calculadora 47](#_bookmark11)

[Figura 13: Diagrama Molic – Apresentar funcionalidades de consulta 48](#_bookmark12)

[Figura 14: Firebase Crash Reporting – Registros de falhas encontrados 50](#_bookmark13)

[Figura 15: Firebase Crash Reporting – Resumo detalhado do erro 51](#_bookmark14)

[Figura 16: Firebase Crash Reporting – Rastreamento do Erro 51](#_bookmark15)

[Figura 17: Projeto armazenado no Github 53](#_bookmark16)

[Figura 18: Bug cadastrado utilizando a ferramenta Issue do Github 54](#_bookmark17)

[Figura 19: Tela de login 56](#_bookmark18)

[Figura 20: Tela de login Google 56](#_bookmark19)

[Figura 21: Tela de cadastro de usuário 57](#_bookmark20)

[Figura 22: Tela de cadastro de usuário - Dados inválidos 57](#_bookmark21)

[Figura 23: Tela redefinição de senha 58](#_bookmark22)

[Figura 24: Tela redefinição de senha preenchida 58](#_bookmark23)

[Figura 25: E-mail de redefinição de senha 58](#_bookmark24)

[Figura 26: Menu inicial 59](#_bookmark25)

[Figura 27: Menu lateral 59](#_bookmark26)

[Figura 28: Tela de diagnósticos 60](#_bookmark27)

[Figura 29: Filtragem de diagnósticos 60](#_bookmark28)

[Figura 30: Conteúdo do diagnóstico selecionado – aba Tratamento 61](#_bookmark29)

[Figura 31: Conteúdo do diagnóstico selecionado – aba principal 61](#_bookmark30)

[Figura 32: Conteúdo do diagnóstico selecionado – aba Diagnóstico 61](#_bookmark31)

[Figura 33: Calculadora de Corticóide 62](#_bookmark32)

[Figura 34: Menu de calculadoras 62](#_bookmark33)

[Figura 35: Calculadora de doses - selecionar peso 63](#_bookmark34)

[Figura 36: Calculadora de corticóide resultado 63](#_bookmark35)

[Figura 37: Calculadora de doses - Resultado 64](#_bookmark36)

[Figura 38: Calculadora de doses - lista de medicamentos 64](#_bookmark37)

[Figura 39: Informações medicamento 65](#_bookmark38)

[Figura 40: Bulário 65](#_bookmark39)

[Figura 41: Nota selecionada 66](#_bookmark40)

[Figura 42: Lista de notas do usuário 66](#_bookmark41)

[Figura 43: Tela de alteração de dados cadastrais 67](#_bookmark42)

[Figura 44: Mensagem após atualização dos dados cadastrais 67](#_bookmark43)

1. [INTRODUÇÃO 7](#_bookmark44)
2. [TRABALHOS RELACIONADOS 9](#_bookmark45)
3. [CONCEPÇÃO 11](#_bookmark46)
   1. [Visão 11](#_bookmark47)
      1. [Introdução 11](#_bookmark48)
         1. [Propósito 11](#_bookmark49)
         2. [Escopo 11](#_bookmark50)
      2. [Posicionamento 12](#_bookmark51)
         1. [Descrição do problema 12](#_bookmark52)
         2. [Definição do Produto 12](#_bookmark53)
      3. [Descrição dos usuários 13](#_bookmark54)
         1. [Usuários 13](#_bookmark55)
         2. [Ambientes dos usuários 13](#_bookmark56)
         3. [Necessidades dos Usuários 13](#_bookmark57)
         4. [Alternativas e Concorrência 15](#_bookmark58)
4. [ELABORAÇÃO 20](#_bookmark67)
   1. [Arquitetura Proposta 20](#_bookmark68)
      1. [Primeira Alternativa 20](#_bookmark69)
      2. [Segunda Alternativa 20](#_bookmark70)
   2. Arquitetura Escolhida…………………………. 21
      1. Desenho da Arquitetura 21
      2. Banco de Dados 21
   3. [Caso de Uso 21](#_bookmark71)
      1. [Diagrama de caso de uso 21](#_bookmark72)
      2. [Descrição de caso uso 22](#_bookmark73)
         1. [Cadastrar Usuário 22](#_bookmark74)
         2. [Realizar Login Convencional 23](#_bookmark75)
         3. [Realizar Login Via Plataforma Google 24](#_bookmark76)
         4. [Realizar Reset de Senha 26](#_bookmark77)
         5. [Gerenciar Usuário 27](#_bookmark78)
         6. [Consultar Diagnósticos 28](#_bookmark79)
         7. [Consultar Bulário 29](#_bookmark80)
         8. [Utilizar Calculadora de Antibióticos 30](#_bookmark81)
         9. [Utilizar Calculadora de Corticóides 31](#_bookmark82)
         10. [Enviar Mensagem 32](#_bookmark83)
         11. [Cadastrar Notas 34](#_bookmark84)
         12. [Gerenciar Notas 35](#_bookmark85)
   4. [Diagrama de classe conceitual 37](#_bookmark86)
   5. [Diagrama de sequência 38](#_bookmark87)
      1. [Cadastrar usuário 38](#_bookmark88)
      2. [Realizar Login Padrão 39](#_bookmark89)
      3. [Realizar Login Google 40](#_bookmark90)
      4. [Selecionar bulário 41](#_bookmark91)
      5. [Selecionar diagnósticos 42](#_bookmark92)
      6. [Solicitar reset de senha 43](#_bookmark93)
   6. [Diagrama de objetos 44](#_bookmark94)
   7. [Diagrama Molic 45](#_bookmark95)
      1. [Apresentar tela de login 45](#_bookmark96)
      2. [Apresentar menu principal 46](#_bookmark97)
      3. [Apresentar calculadora 47](#_bookmark98)
      4. [Apresentar funcionalidades de consulta 48](#_bookmark99)
5. [CONSTRUÇÃO 49](#_bookmark100)
   1. [Crash Reporting 49](#_bookmark101)
   2. [GitHub 52](#_bookmark102)
      1. [Gerenciamento de Configuração 52](#_bookmark103)
      2. [Documentação de Requisitos e Bugs 53](#_bookmark104)
      3. [Gerenciamento de conteúdo 54](#_bookmark105)
   3. [Apresentação das telas 55](#_bookmark106)
      1. [Tela de login padrão 55](#_bookmark107)
      2. [Tela de autenticação Google 55](#_bookmark108)
      3. [Tela de cadastro de usuário 56](#_bookmark109)
      4. [Tela de reset de senha 57](#_bookmark110)
      5. [Menu Inicial 59](#_bookmark111)
      6. [Tela de diagnósticos 60](#_bookmark112)
      7. [Calculadoras 62](#_bookmark113)
      8. [Bulário 64](#_bookmark114)
      9. [Notas 65](#_bookmark115)
      10. [Atualizar dados cadastrais 66](#_bookmark116)
6. [CONCLUSÃO 68](#_bookmark117)

[REFERÊNCIAS BIBLIOGRÁFICAS 70](#_bookmark118)

# INTRODUÇÃO

A utilização de aplicativos WEB se dá pelo fato da necessidade do profissional, e pela facilidade de poder acessar a ferramenta a qualquer hora, em qualquer momento e lugar, bastando a conexão com a internet. O fato de que esses aplicativos não necessitam de uma instalação local o que livra o usuário de possuir requisitos altos para acesso, manutenção e uso da ferramenta, seguindo a tendência de SaaS (Software as a Service).

A web possibilita termos acesso as informações triviais para o negócio, em tempo real com alta disponibilidade, pois os servidores ficam ligados 7×24. Assim a Web acaba auxiliando na tomada de decisão e permitindo que a mesma tenha maior embasamento, consistência e principalmente agilidade. Melhorando significativamente a performance no atendimento a clientes e possibilitando que os mesmos estejam espalhados pelo globo. Além disso, o custo em se ter um aplicativo Web é mais baixo comparado a um aplicativo Desktop, devido a sua manutenção ser centralizada.

A análise de desempenho é uma área que está consolidada há muitos anos no futebol europeu e aos poucos chega no futebol brasileiro. Por aqui ainda são pouquíssimos os clubes que possuem um departamento realmente estruturado para desenvolver esse trabalho.

No futebol de hoje é vital que existam profissionais realizando esses tipos de análises nos clubes. Deve-se conhecer muito bem seus adversários e também sua própria equipe. Como afirma Sun Tzu em sua obra A Arte da Guerra: “Conheça o seu inimigo e conheça a si mesmo e em 100 batalhas, você nunca correrá perigo”.

O Sis-Atleta tem como objetivo atender as necessidades do analista de desempenho, que é responsável principalmente pelas categorias de base dos clubes do Rio de Janeiro, possibilitando o armazenamento e unificação das informações dos seus jogadores e a sua tomada de decisão.

Os requisitos e necessidades do projeto foram levantados juntamente com um profissional da área, e para o mesmo utilizaremos as tecnologias mais modernas do mercado que possibilite uma interação amigável para o usuário, e permita também uma fácil manutenção.

# TRABALHOS RELACIONADOS

Dentro os softwares existentes no mercado para a realização da análise de desempenho tático nos clubes de futebol, serão abordados dois que possuem propostas semelhantes e com certo destaque no mercado.

O primeiro deles é o Wyscout, software contratado pela CBF em 2015, e que já atende grandes clubes como Barcelona (Espanha) e Bayern (Alemanha), que disponibiliza conteúdo de qualidade e atualizado sobre todos os principais campeonatos do mundo (apenas os televisionados no Brasil). (GloboEsporte, 2015)

Um dos grandes diferenciais do software é possuir filmagens de boa parte das equipes que integram competições do Brasil e de aproximadamente outros 80 países. A partir disso, o próprio software compila os dados de forma individual e coletiva (ESPORTES UOL, 2015).

O Wyscout fornece vários planos de serviços, tanto para agente de atleta, jogador, analista, árbitro, clube, jornalista e treinador. Além dos planos, ele dá a possibilidade do usuário experimentar o software por 15 dias, disponibilizando uma parte do seu conteúdo, contendo apenas as funcionalidades básicas. Para utilizar a versão trial do software basta realizar o cadastro no site e utilizar o sistema, que é todo em versão WEB. Nos planos pagos, existe a opção de incluir alguns serviços extras ao plano, como maior espaço de armazenamento para os vídeos e dados do clube. A forma de pagamento pode ser feita mensalmente ou anualmente, e ao assinar, o usuário pode cancelar a assinatura a qualquer momento e terá suporte técnico 24x7.

O Wyscout é conhecido internacionalmente, utilizado pelos principais clubes e seleções do mundo e pode ser utilizado em qualquer dispositivo com conexão a internet.

Apesar de todas as características positivas apontadas acima, o Wyscout possui algumas características negativas, principalmente no que se refere a agilidade de navegação. A grande quantidade de conteúdo e informação disponibilizados pode atrapalhar a execução de processos rápidos, além de causar certo desconforto ao usuário.

Outra questão relevante, é que, no Brasil, o Wyscout não possui dados de partidas e campeonatos que não são televisionados. Segundo comentários de analistas de clubes grandes do Rio de Janeiro, as categorias de base, que deveriam utilizar esse software para descobrir destaques e novos talentos, acabam caindo no esquecimento.

O segundo software analisado é o Footstats, desenvolvido pela empresa brasileira SkySoftware. O Footstats é um sistema de alta tecnologia, online real time, que cobre simultaneamente todas as rodadas dos campeonatos mais importantes do Brasil. Estaduais, Brasileiros A e B, Libertadores da América, Copa Sul Americana, Eliminatória, Copa do Mundo, Copa América e oito campeonatos europeus. (Footstats, 2016).

Um grande diferencial dessa ferramenta está na quantidade de dados coletados, com alto nível de detalhamento, com alta precisão e confiabilidade. (Skysoftware, 2008).

Assim como o Wyscout, ele fornece informação dos principais campeonatos e atletas a nível nacional. Os usuários poderão usufruir dos recursos do portal gratuitamente, em um primeiro momento, por meio de cadastro. Já as empresas podem comprar cotas de patrocínio para ter o direito de expor sua marca no portal ou nos aplicativos multiplataforma.

O Footstats também está disponível para Android e iOS, sendo o aplicativo mais bem avaliado dentre os semelhantes disponíveis e possui cerca de 10 mil downloads. (Google Play, 2017).

# CONCEPÇÃO

* 1. Visão
     1. Introdução

A finalidade desta seção é analisar e definir as necessidades e funcionalidades gerais do software Sis-Atleta. Ela se concentra nos recursos necessários aos envolvidos e usuários-alvo, e nas razões que levam a essas necessidades.

* + - 1. Propósito

A presente seção tem, por finalidade, descrever as principais características do software a ser desenvolvido. A busca de soluções dos problemas enfrentados e a criação de funcionalidades farão com que ele se torne uma ferramenta essencial para o analista.

* + - 1. Escopo

O software Sis-Atleta se destina, de maneira objetiva, ao dia a dia das categorias de base dos clubes de futebol, auxiliando analistas de desempenho. Sua principal área abrangerá as estatísticas do atleta por jogo, fornecendo, em relatórios, os detalhes para que o analista saiba do presente e do potencial do jogador. Dessa maneira, o software será uma ferramenta útil e rápida no cotidiano do analista de desempenho, facilitando seu trabalho.

O software foi desenvolvido em conjunto com um analista de desempenho da base do Botafogo de Futebol e Regatas, que ajudou na revisão do conteúdo.

* + 1. Posicionamento

Por meio desse documento de visão teremos uma noção geral do atual contexto de trabalho enfrentado e apresentaremos prováveis soluções.

* + - 1. Descrição do problema

|  |  |
| --- | --- |
| Problema | Atualmente, o analista de base cria uma nova planilha de Excel a cada jogo para que as estatísticas de cada atleta sejam armazenadas. Isso prejudica a análise futura, visto que o número de planilhas criadas será muito grande. |
| Quem é afetado? | O analista de desempenho, que pode acabar tomando uma decisão errada, e o atleta, que pode estar sendo mal aproveitado devido a dificuldade de acesso as informações. |
| A solução deve | Possuir uma interface amigável para facilitar a análise deste profissional, gerando relatórios completos contendo as estatísticas e todos os dados necessários para a análise. |

* + - 1. Definição do produto

|  |  |
| --- | --- |
| Para | Analistas de desempenho e interessados nos dados do atleta. |
| Que | Desejam ser auxiliados na análise das estatísticas para tomada de decisão. |

|  |  |
| --- | --- |
| O | Sis-Atleta é uma software WEB. |
| Que | Permite a interação de modo fácil, simples e completo. |
| Ao contrário | Atualmente, não existem ferramentas para as categorias de base dos clubes. |
| O produto | Possibilitará que o analista preencha |

|  |  |
| --- | --- |
|  | os dados de cada atleta por jogo, e ao fim, analisar através de relatórios os melhores atletas por posição, estatísticas, etc., ajudando na tomada de decisão. |

* + 1. Descrição dos usuários
       1. Usuários

|  |  |
| --- | --- |
| Nome | Descrição |
| Analistas de Desempenho | O software foi desenvolvido com o objetivo de ser utilizado pelos analistas de desempenho, para exercer sua função com excelência. |

* + - 1. Ambientes dos usuários

O acesso deve ser possível em qualquer dispositivo que possua conexão com a internet.

* + - 1. Necessidades dos usuários

|  |  |
| --- | --- |
| Necessidade | Cadastro de Atletas |
| Prioridade | Essencial |
| Preocupações | Desempenho do software devido ao contexto de uso (antes do jogo; durante o jogo e após o jogo) |

|  |  |
| --- | --- |
| Solução Atual | Atualmente não existe um cadastro de atleta, apenas o preenchimento da planilha, caso o atleta participe da atividade. |
| Soluções propostas | Reunir, em uma tela, de maneira simples e organizada os atletas que o analista deverá observar. |

|  |  |
| --- | --- |
| Necessidade | Cadastro das Estatísticas do Atleta |
| Prioridade | Crítica |
| Preocupações | Desempenho do software devido ao contexto de uso (antes do jogo; durante o jogo e após o jogo) |
| Solução Atual | Preenchimento de uma planilha de Excel, criada a cada jogo realizado. |
| Soluções propostas | Organizar os jogos em uma tela, contendo os jogadores que participaram com suas devidas estatísticas da partida. |

|  |  |
| --- | --- |
| Necessidade | Relatórios baseados nas estatísticas do Atleta |
| Prioridade | Crítica |
| Preocupações | Desempenho do software devido ao contexto de uso (antes do jogo; durante o jogo e após o jogo) |
| Solução Atual | Consultar, planilha por planilha, as estatísticas de cada atleta. |
| Soluções propostas | Reunir, em tela, diversos relatórios com os filtros mais utilizados pelo analista no dia a dia. |

* + - 1. Alternativas e Concorrência

Conforme comentado no capítulo dois, existem softwares que possuem propostas e funcionalidades semelhantes com as do Sis-Atleta, porém nenhum deles possui como foco o analista das categorias de base.

* 1. Requisitos funcionais
* Permitir cadastro de usuário
* Realizar login utilizando e-mail e senha cadastrados
* Para ter acesso as funcionalidades fornecidas pelo software, os usuários cadastrados deverão se autenticar com seu respectivo login e senha.
* Possuir reset de senha, caso o usuário esqueça a mesma
* Cadastro de atletas
* Cadastro de clubes
* Cadastro de jogos
* Preenchimento das estatísticas dos atletas no jogo
* Disponibilizar relatórios com os filtros desejados pelo analista
  1. Requisitos não funcionais:
* O software deverá ser desenvolvido para funcionar em qualquer dispositivo com conexão a internet.
* O sistema deverá possuir uma interface simples e de fácil utilização para o analista
* O sistema deverá ter alta disponibilidade, por exemplo, 99% do tempo.

# ELABORAÇÃO

* 1. Arquitetura Proposta
     1. Primeira alternativa

Desenvolvimento de um aplicativo Android nativo utilizando a plataforma móvel Firebase. Nesse cenário, a aplicação será desenvolvida na linguagem Java e utilizará APIs fornecidas pelo Firebase para auxiliar no gerenciamento dos dados armazenados. Será utilizado um banco de dados NoSql disponibilizado por essa plataforma que tem como características principais ser de fácil manutenção e possuir melhor desempenho em comparação ao modelo relacional.

* + 1. Segunda alternativa

Desenvolvimento de um aplicativo baseado em web utilizando o banco de dados Mysql. Nesse cenário, o aplicativo será desenvolvido com as tecnologias HTML5, Css, Javascript e utilizará um web service para fazer a comunicação com o banco de dados. Por questões de segurança, o aplicativo não pode conectar-se diretamente ao servidor, por isso é necessário o uso de web services.

Outra vantagem de sua utilização está em contribuir com interoperabilidade entre diversas plataformas, pois independe da tecnologia que o cliente e servidor utilizam. (Lecheta, 2016).

* + 1. Arquitetura escolhida

Nesse projeto foi utilizada a primeira alternativa apresentada. O principal fator para escolha foram as funcionalidades fornecidas pelo Firebase, que foram de grande auxílio no desenvolvimento e também o serão na manutenção do aplicativo. O Firebase é uma ferramenta desenvolvida pela Google que auxilia no desenvolvimento de aplicações de alta performance, fornecendo recursos que

podem ser integrados ao aplicativo de acordo com suas necessidades (TECMUNDO, 2016). Além do serviço de banco de dados comentado acima, serão utilizados também serviços de autenticação de usuário e Crash Report. Este último será descrito mais a frente.

Outro fator determinante para essa escolha é o uso de tecnologias nativas, que permitem ao aplicativo melhor compatibilidade com os recursos disponibilizados pelo dispositivo (DEVMEDIA, 2014). Nesta seção, serão abordados detalhadamente os componentes trabalhados.

* + - 1. Autenticação

Para obter-se controle de acesso aos usuários e garantir proteção dos dados armazenados no aplicativo, serão disponibilizadas duas formas de autenticação. A primeira delas utilizará um e-mail e senha cadastrados na aplicação. A segunda, uma conta Google ativa.

A aplicação irá se integrar ao serviço de autenticação do Firebase que, juntos, irão gerenciar essa atividade garantindo segurança no acesso aos dados. Além disso, para o modelo de autenticação via Google, utilizaremos também as APIs fornecidas pela empresa, que serão responsáveis por validar os dados inseridos, caso o usuário opte por essa forma de autenticação.

Outro serviço que será disponibilizado é a possibilidade de recuperação de senha, onde será possível alterar uma senha associada a um determinado e-mail.

* + - 1. Banco de Dados

Conforme já comentado durante o capítulo, foi utilizado na aplicação um banco de dados NoSQL hospedado em nuvem e disponibilizado pelo Firebase. Os dados foram armazenados como um JSON e poderão ser sincronizados com o dispositivo para que também possam estar disponíveis quando o aplicativo

estiver off-line. Podemos considerar esse modelo de banco de dados como uma árvore JSON hospedada na nuvem. Ele se difere dos modelos relacionais principalmente por não possuir tabelas e tuplas: os dados quando adicionados viram um nó da estrutura (DEVMEDIA, 2012).

Outra característica desse modelo está no melhor desempenho durante a execução das consultas, além de outros atributos importantes como escalabilidade.

* + - 1. Padrão de arquitetura

O projeto foi desenvolvido utilizando o padrão de arquitetura MVC (Model- View-Controller), que se caracteriza pela divisão da aplicação em 3 partes principais: Modelo (Model), Visão (View) e Controlador (Controller). A camada de Modelo é responsável por armazenar e gerenciar os dados que trafegam pela aplicação. A camada de Visão é responsável pela interface com o usuário, isto é, onde as informações serão apresentadas. E, por fim, o Controlador serve de interface com os demais componentes da arquitetura. Ele é responsável por mapear as ações do usuário enviado-as para as camadas de Modelo, e o mesmo acontece ao contrário: o controlador atualiza a camada de Visão utilizando dados recuperados do modelo (DEVMEDIA, 2013).

O MVC é um padrão conhecido e adotado por diversas empresas. Dentre as suas principais vantagens, podemos citar: organização, facilidade de manutenção e adição de recursos, reaproveitamento do código, contribuição para o desenvolvimento de um código limpo, entre outras (DOOLEY, 2011).

A arquitetura padrão de projetos Android auxilia no desenvolvimento desse modelo. Os arquivos XML de layout representam a tela do aplicativo. É nele que são incluídos componentes como botões, labels, listas, imagens, campos de texto e etc. A classe Activity é o controller da nossa View, sendo responsável por tratar todos os eventos de tela, além de referenciar o XML criado definindo a interface

gráfica do usuário. A partir disso, podemos desenvolver nossa camada de Modelo, respeitando as características comentadas acima.

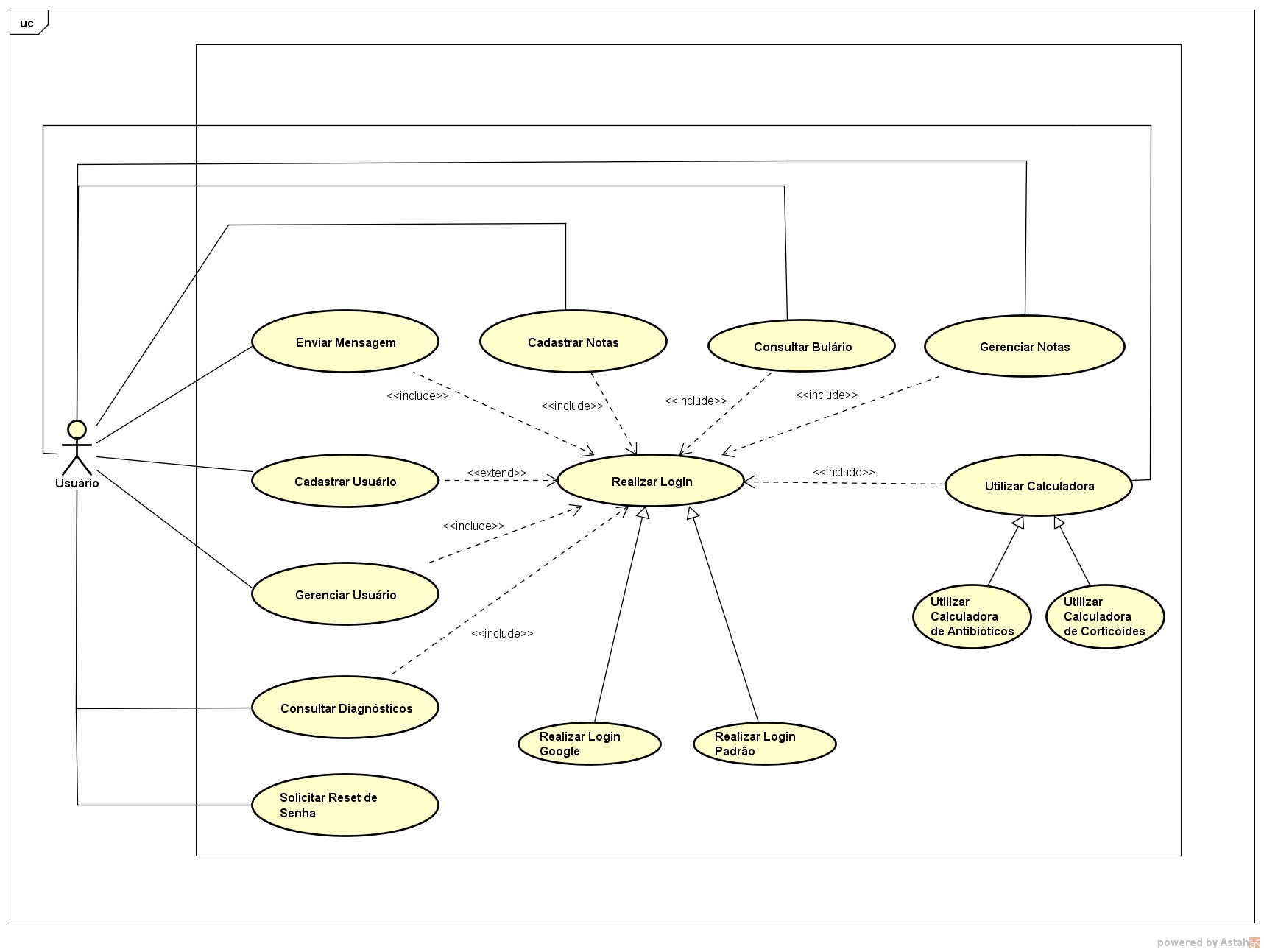
Vantagens:

* + - * + Aplicações nativas de um sistema operacional possuem melhor comunicação com o dispositivo e seus componentes, o que permite desenvolver aplicações mais completas e estáveis.
        + A utilização do Firebase pode significar redução de tempo e custo no desenvolvimento de uma infraestrutura para suportar a aplicação.
        + Melhor desempenho no gerenciamento dos dados devido a utilização de banco de dados NoSQl.
        + A utilização do padrão MVC contribui para facilidades de manutenção, adição de recursos e reaproveitamento de código.

Desvantagens:

* + - * + Acoplamento causado pela utilização de serviços Firebase como parte do backend da aplicação.
        + Aplicações nativas não podem ser utilizadas em outros sistemas.

* 1. Caso de Uso
     1. Diagrama de Caso de Uso



*Figura 1: Diagrama de caso de uso*

* + 1. Descrição de Caso de Uso
       1. Cadastrar Usuário

# NOME

Cadastrar Usuário

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para um usuário realizar seu cadastro no aplicativo E-ORL

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá possuir um e-mail

# FLUXO BÁSICO

1. O Usuário acessa o Aplicativo
2. O Aplicativo exibe a tela de login
3. O Usuário seleciona a opção “Cadastre-se”
4. O Aplicativo exibe a tela de cadastro
5. O Usuário preenche corretamente todos os campos exibidos
6. O Usuário seleciona a opção “Cadastrar”
7. O Aplicativo verifica os dados inseridos
8. O Aplicativo realiza o cadastro do usuário
9. O Aplicativo redireciona o usuário para a tela de login

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 8-Usuário preenche um dos campos incorretamente**

* 1. O Aplicativo informa qual campo foi preenchido incorretamente e qual seria o valor esperado
  2. O Aplicativo retorna ao passo 4

# (A2) Alternativa ao passo 8-Usuário submete os dados de um usuário já cadastrado no sistema

* 1. O Aplicativo informa que já existe um usuário associado aos dados inseridos
  2. O Aplicativo retorna ao passo 4

# ESTRUTURA DE DADOS

**(ED1) Formulário de Cadastro de Usuário**

* 1. Nome
  2. E-mail
  3. Data de Nascimento
  4. Senha

# REGRAS DE NEGÓCIO

**(RN1)** A senha deve conter no mínimo 6 caracteres

**(RN2)** O nome deve ter no mínimo 6 caracteres

* + - 1. Realizar Login Convencional

# NOME

Realizar Login Convencional

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para um usuário realizar o login de forma convencional no aplicativo

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá possuir um cadastro no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o Aplicativo
2. O Aplicativo exibe a tela de login
3. O Usuário preenche os campos “E-mail” e “Senha”
4. O Usuário seleciona a opção “Entrar”
5. O Aplicativo valida os dados inseridos
6. O Aplicativo exibe o menu principal (Área Logada)

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 5 - Usuário preenche um dos campos incorretamente**

* 1. O Aplicativo informa que os dados inseridos estão incorretos
  2. O Aplicativo retorna ao passo 2

# (A2) Alternativa ao passo 5 - Usuário submete os dados de login com um dos campos vazios.

* 1. O Aplicativo informa que todos os campos devem ser preenchidos
  2. O Aplicativo retorna ao passo 2
     + 1. Realizar Login Via Plataforma Google

# NOME

Realizar Login Via Plataforma Google

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para um usuário realizar o login no aplicativo utilizando a API do Google.

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá possuir um e-mail Google

# FLUXO BÁSICO

1. O Usuário acessa o Aplicativo
2. O Aplicativo exibe a tela de login
3. O Usuário seleciona a opção “Logar com o Google”
4. O Aplicativo exibe a tela de login via Google
5. O Usuário preenche os campos “E-mail” e “Senha”
6. A plataforma da Google valida os dados inseridos
7. O Aplicativo exibe o menu principal (Área Logada)

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 7 - Usuário preenche um dos campos incorretamente**

* 1. A plataforma da Google informa que os dados inseridos estão incorretos
  2. O Aplicativo retorna ao passo 2

# (A2) Alternativa ao passo 7 - Usuário submete os dados login com um dos campos vazios

* 1. A plataforma da Google informa que todos os campos devem ser preenchidos
  2. O Aplicativo retorna ao passo 2
     + 1. Realizar Reset de Senha

# NOME

Realizar Reset de Senha

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para um usuário solicitar o reset de sua senha

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá possuir um cadastro no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o Aplicativo
2. O Aplicativo exibe a tela de login
3. O Usuário seleciona a opção “Esqueci minha senha”
4. O Aplicativo exibe a tela de Reset de Senha
5. O Usuário preenche o campo “E-mail” informado o endereço de e-mail cadastrado
6. O Usuário seleciona a opção “Enviar Solicitação”
7. O Aplicativo enviará um e-mail contendo um link para efetuar o reset de senha
8. O Usuário acessa o e-mail informado e seleciona o link exibido
9. O Aplicativo exibirá via navegador o formulário para atualização da nova senha
10. O Usuário informa a nova senha
11. O Aplicativo valida os dados informados e informa que o procedimento foi efetuado com sucesso

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 12 - Usuário submete a nova senha com o campo vazio**

* 1. O Aplicativo informa que todos os campos devem ser preenchidos
  2. O Aplicativo retorna ao passo 10
     + 1. Gerenciar Usuário

# NOME

Gerenciar Usuário

# DESCRIÇÃO SUCINTA

Este caso de uso descreve as possíveis atividades de gerenciamento das informações do usuário. Sendo possível consultar e alterar os dados cadastrados no aplicativo.

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo(Área Logada)
2. O Usuário seleciona a opção “Alterar dados cadastrais”
3. O Aplicativo exibe a tela de alteração de dados
4. O Aplicativo exibe os dados cadastrados pelo Usuário
5. O Usuário altera os dados exibidos (Opcional)
6. O Usuário informa sua senha
7. O Usuário seleciona a opção “Confirmar Alteração”
8. O Aplicativo valida os dados informados
9. O Aplicativo redireciona o Usuário para menu inicial e informa sucesso na operação

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 8 - Usuário preenche um dos campos incorretamente**

* 1. O Aplicativo informa que os dados inseridos estão incorretos
  2. O Aplicativo retorna ao passo 3

# (A2) Alternativa ao passo 8 - Usuário preenche o campo senha incorretamente

* 1. O Aplicativo informa que a senha informada está incorreta
  2. O Aplicativo retorna ao passo 3
     + 1. Consultar Diagnósticos

# NOME

Consultar Diagnósticos

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para consultar os diagnósticos cadastrados no aplicativo

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo(Área Logada)
2. O Usuário seleciona a opção “Diagnósticos”
3. O Aplicativo exibe uma lista de diagnósticos
4. O Usuário seleciona um item da lista
5. O Aplicativo exibe uma tela contendo informações referentes ao diagnóstico selecionado
   * + 1. Consultar Bulário

# NOME

Consultar Bulário

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para consultar o bulário do aplicativo

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo(Área Logada)
2. O Usuário seleciona a opção “Bulário”
3. O Aplicativo exibe uma lista de remédios
4. O Usuário seleciona um item da lista
5. O Aplicativo exibe uma tela contendo informações referentes ao remédio selecionado
   * + 1. Utilizar Calculadora de Antibióticos

# NOME

Utilizar Calculadora de Antibióticos

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para utilizar a calculadora de antibióticos

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo(Área Logada)
2. O Usuário seleciona a opção “Calculadora”
3. O Aplicativo exibe duas opções de calculadoras disponíveis
4. O Usuário seleciona a opção “Antibióticos”
5. O Aplicativo exibe uma lista contendo os pesos em kg
6. O Usuário seleciona um item da lista
7. O Aplicativo exibe uma lista contendo medicamentos
8. O Usuário seleciona um item da lista
9. O Aplicativo informa o resultado do cálculo em ml para o peso e medicamento selecionados
   * + 1. Utilizar Calculadora de Corticóides

# NOME

Utilizar Calculadora de Corticóides

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para utilizar a calculadora de corticóides

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo(Área Logada)
2. O Usuário seleciona a opção “Calculadora”
3. O Aplicativo exibe duas opções de calculadoras disponíveis
4. O Usuário seleciona a opção “Corticóides”
5. O Aplicativo exibe a tela de conversão de corticoide
6. O Usuário informa a dose em mg
7. O Usuário informa qual corticóide a dose informada pertence
8. O Usuário informa para qual corticóide deseja converter o valor
9. O Aplicativo informa o resultado do cálculo em mg

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 8 - O Usuário submete os dados sem preecher o valor da dose**

* 1. O Aplicativo informa que deve ser inserido um valor referente a dose
  2. O Aplicativo retorna ao passo 5

# (A2) Alternativa ao passo 8 - Usuário seleciona os campos de corticóides com valores iguais

* 1. O Aplicativo informa que os corticóides selecionados não pode ser iguais
  2. O Aplicativo retorna ao passo 5
     + 1. Enviar Mensagem

# NOME

Enviar Mensagem

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para enviar uma mensagem para o administrador do aplicativo

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo (Área Logada)
2. O Usuário seleciona a opção “Mensagem”
3. O Aplicativo exibe a tela “Nova Mensagem”
4. O Usuário preencher todos os campos exibidos
5. O Usuário seleciona a opção “Enviar Mensagem”
6. O Aplicativo valida os dados inseridos
7. O Aplicativo realiza o envio dos dados e informa sucesso na operação
8. O Aplicativo redireciona o usuário para o menu inicial da aplicação

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 6 - O Usuário submete os dados sem preencher todos os campos**

* 1. O Aplicativo informa que todos os campos devem ser inseridos
  2. O Aplicativo retorna ao passo 3

# ESTRUTURA DE DADOS

**(ED1) Formulário de Mensagem**

* 1. Título da mensagem
  2. Assunto da mensagem
  3. Corpo da mensagem

4.2.2.11 Cadastrar Notas

# NOME

Cadastrar Notas

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para cadastrar notas do usuário

# ATORES

1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo

# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo (Área Logada)
2. O Usuário seleciona a opção “Notas”
3. O Aplicativo exibe a tela “Notas” contendo uma lista de notas cadastradas
4. O Usuário seleciona a opção de adicionar uma nova nota
5. O Aplicativo exibe a tela “Adicionar Nova Nota”
6. O Usuário preencher todos os campos exibidos
7. O Usuário seleciona a opção para adicionar a nota
8. O Aplicativo valida os dados inseridos
9. O Aplicativo envia os dados e informa sucesso na operação
10. O Aplicativo executa o passo 3 adicionando a nova nota cadastrada na lista

# FLUXOS ALTERNATIVOS

**(A1) Alternativa ao passo 6 - O Usuário submete os dados sem preencher todos os campos**

* 1. O Aplicativo informa que todos os campos devem ser inseridos
  2. O Aplicativo retorna ao passo 5

# ESTRUTURA DE DADOS

**(ED1) Formulário de Notas**

* 1. Título da nota
  2. Corpo da nota(Texto)

4.2.2.12 Gerenciar Notas

# NOME

Gerenciar Notas

# DESCRIÇÃO SUCINTA

Este caso de uso descreve os passos necessários para gerenciar as notas do usuário. Sendo possível consultar, alterar e remover as notas cadastradas.

# ATORES

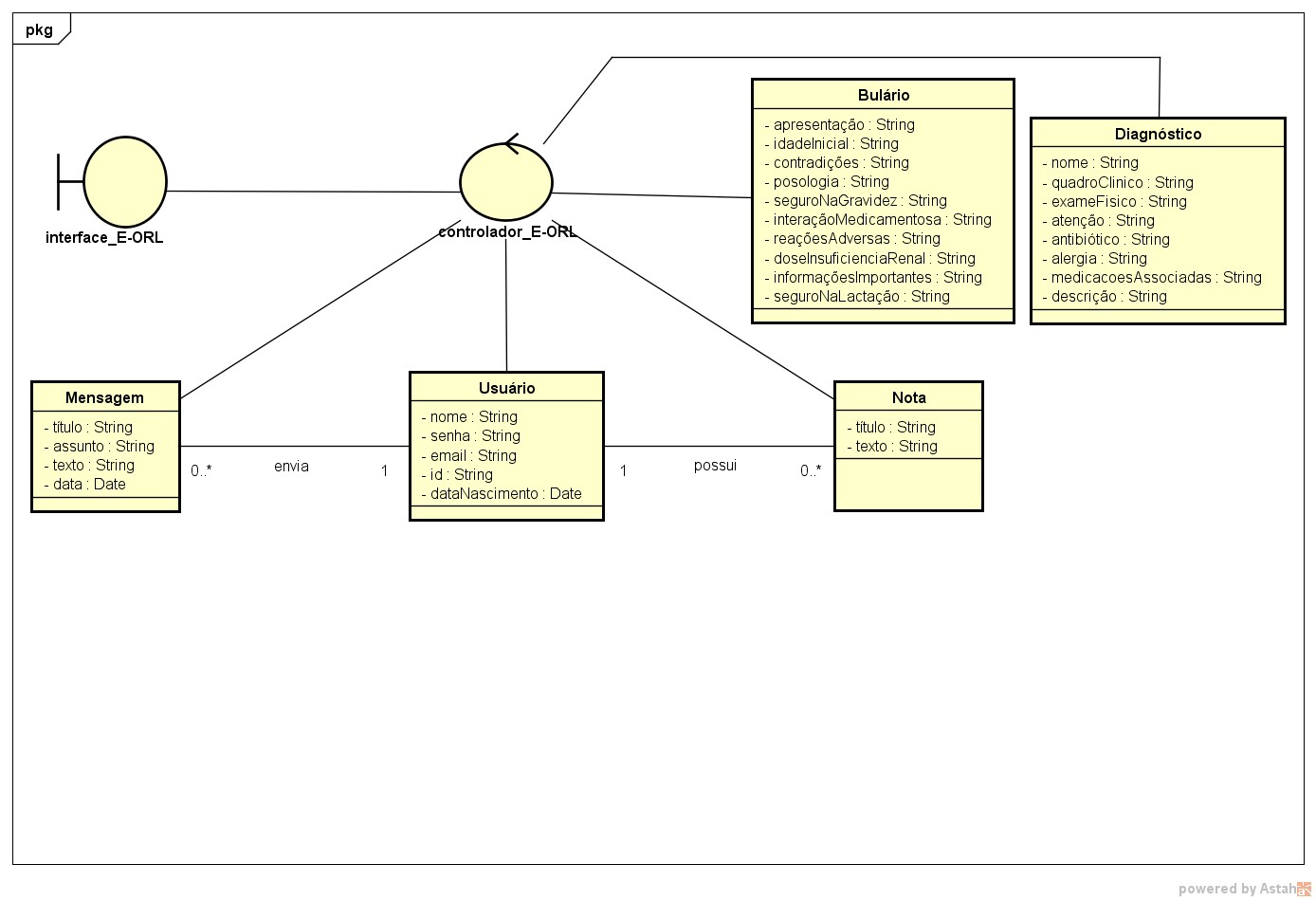
1. Usuário

# PRÉ-CONDIÇÕES

1. O Aplicativo deverá estar instalado no smartphone
2. O Usuário deverá estar logado no aplicativo
3. O Usuário deverá ter, pelo menos, uma nota cadastrada

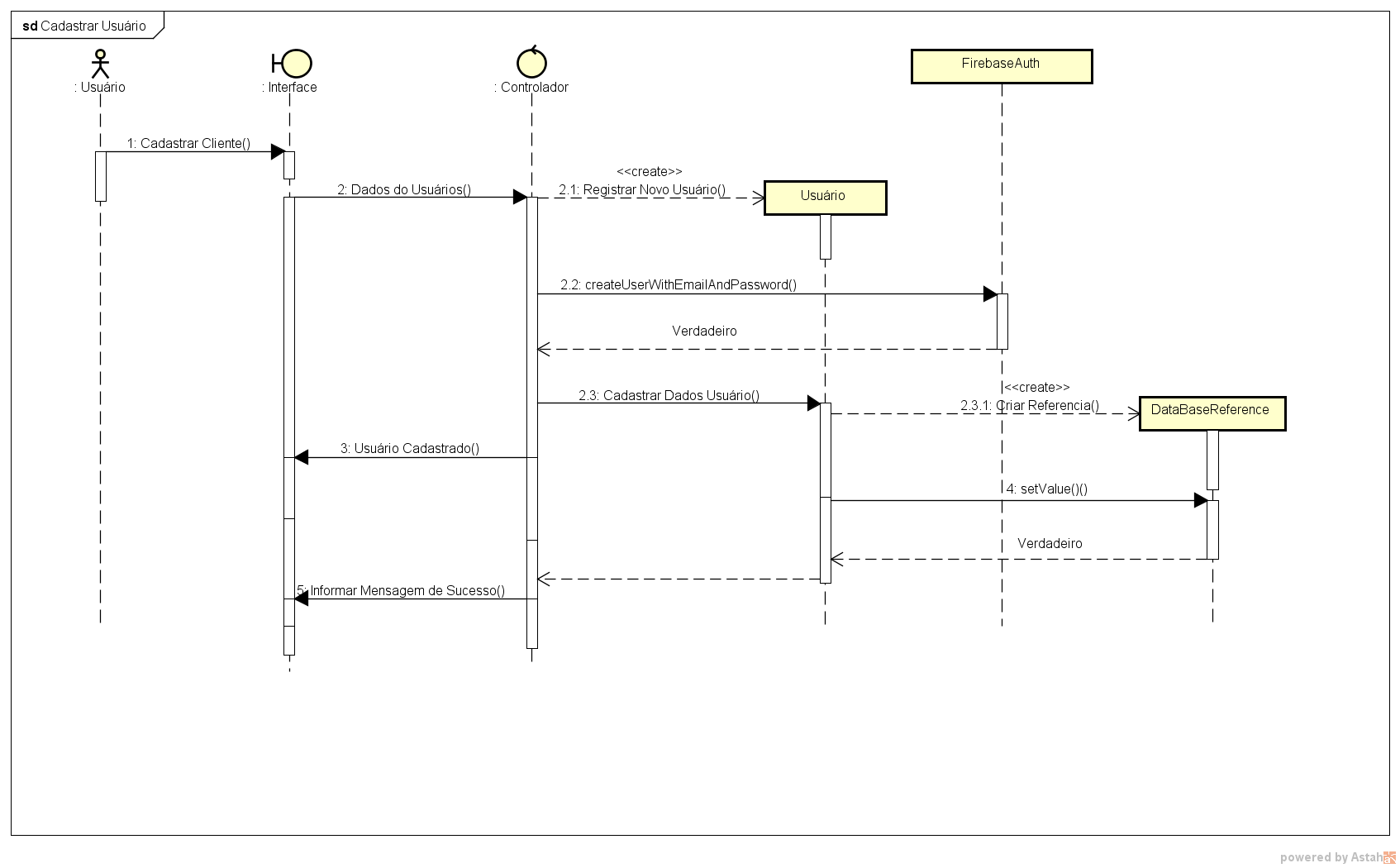
# FLUXO BÁSICO

1. O Usuário acessa o menu inicial do aplicativo (Área Logada)
2. O Usuário seleciona a opção “Notas”
3. O Aplicativo exibe a tela “Notas” contendo uma lista de notas cadastradas
4. O Usuário seleciona um item da lista
5. O Aplicativo exibe a nota selecionada
6. O Aplicativo exibe opções para alterar ou remover a nota selecionada
7. Caso o Usuário selecione as opções de alterar ou remover, o aplicativo efeturá as alterações e executará Passo 3 atualizando a lista de notas de acordo com a operação realizada.
   1. Diagrama de classe conceitual



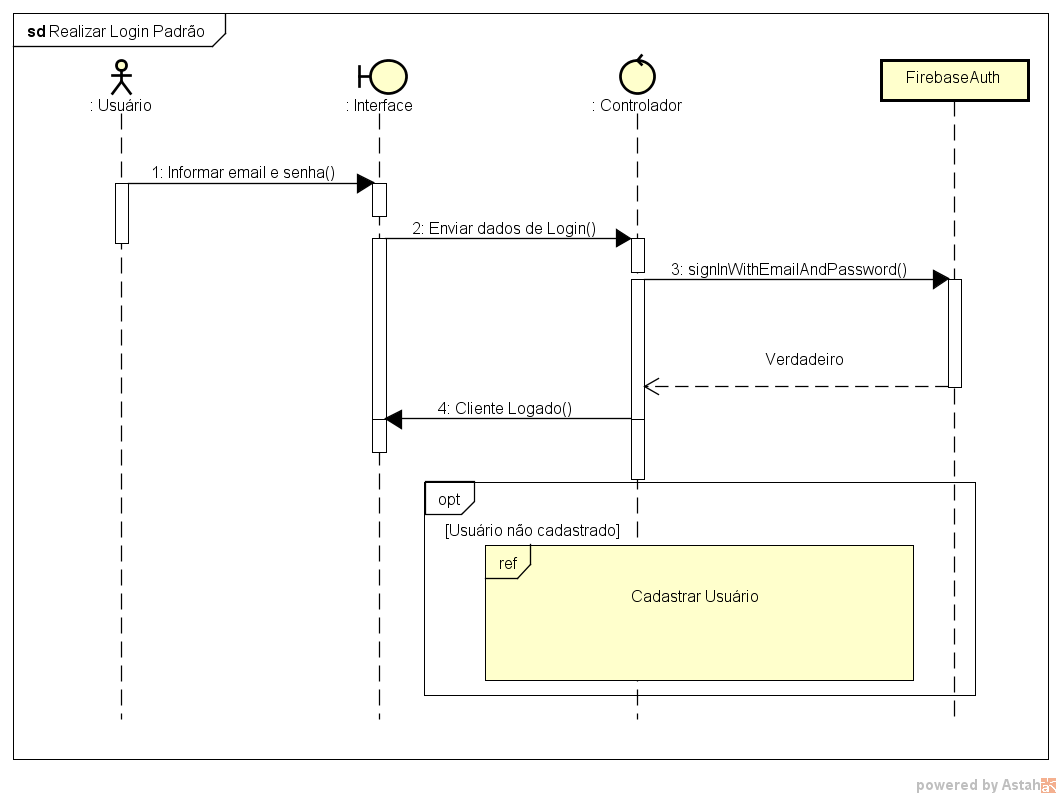
*Figura 2: Diagrama de classe conceitual*

* 1. Diagrama de sequência
     1. Cadastrar usuário



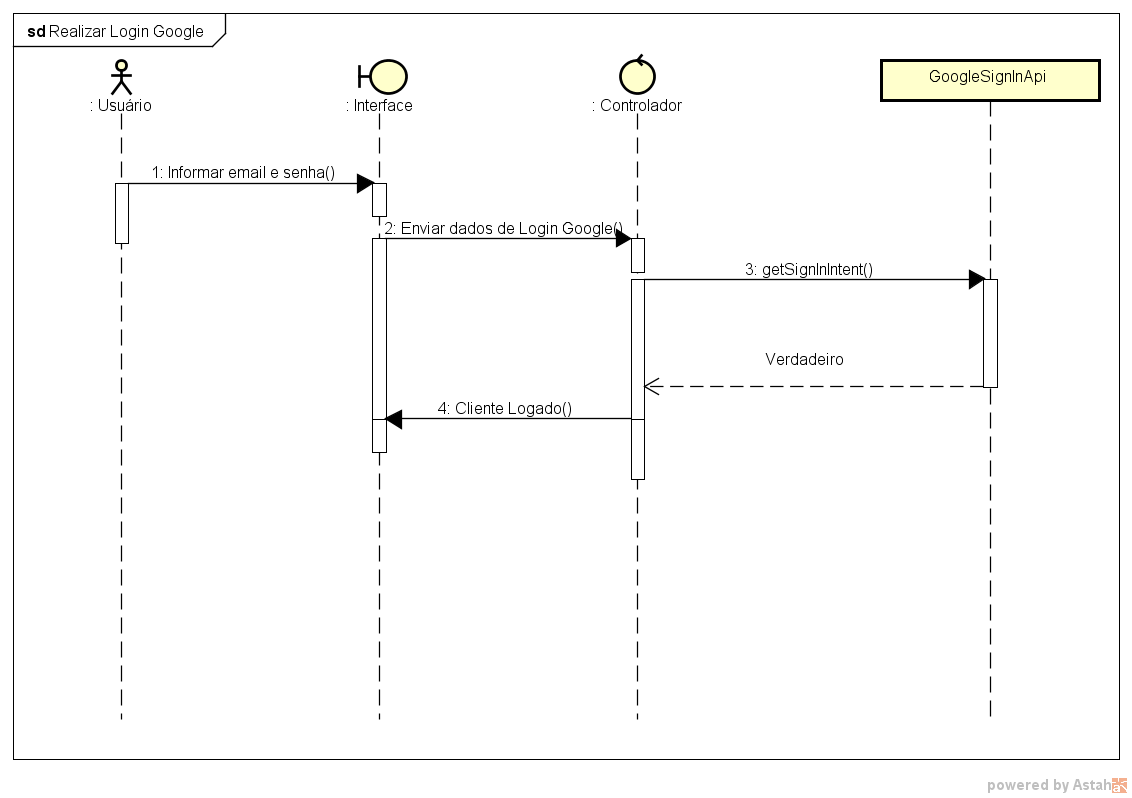
*Figura 3: Diagrama de sequência – Cadastrar Usuário*

* + 1. Realizar Login Padrão



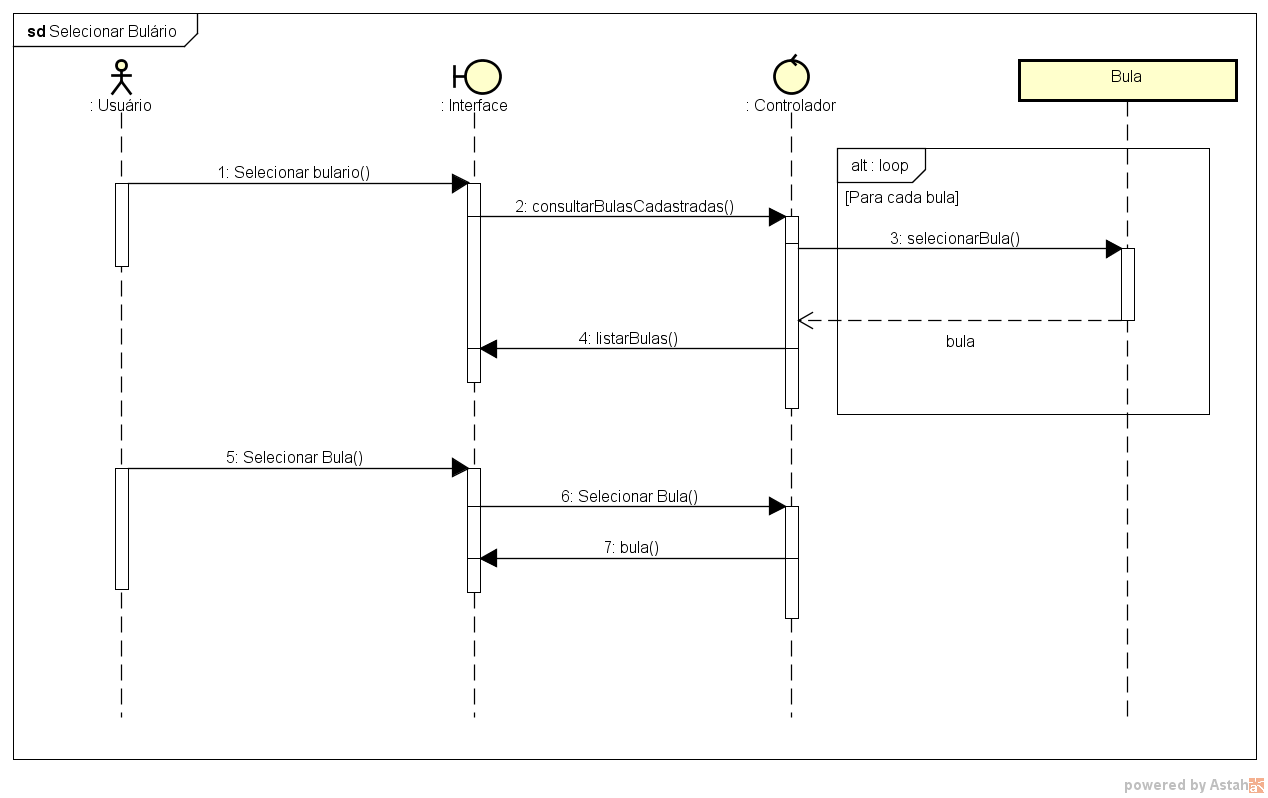
*Figura 4: Diagrama de sequência – Realizar login padrão*

* + 1. Realizar Login Google



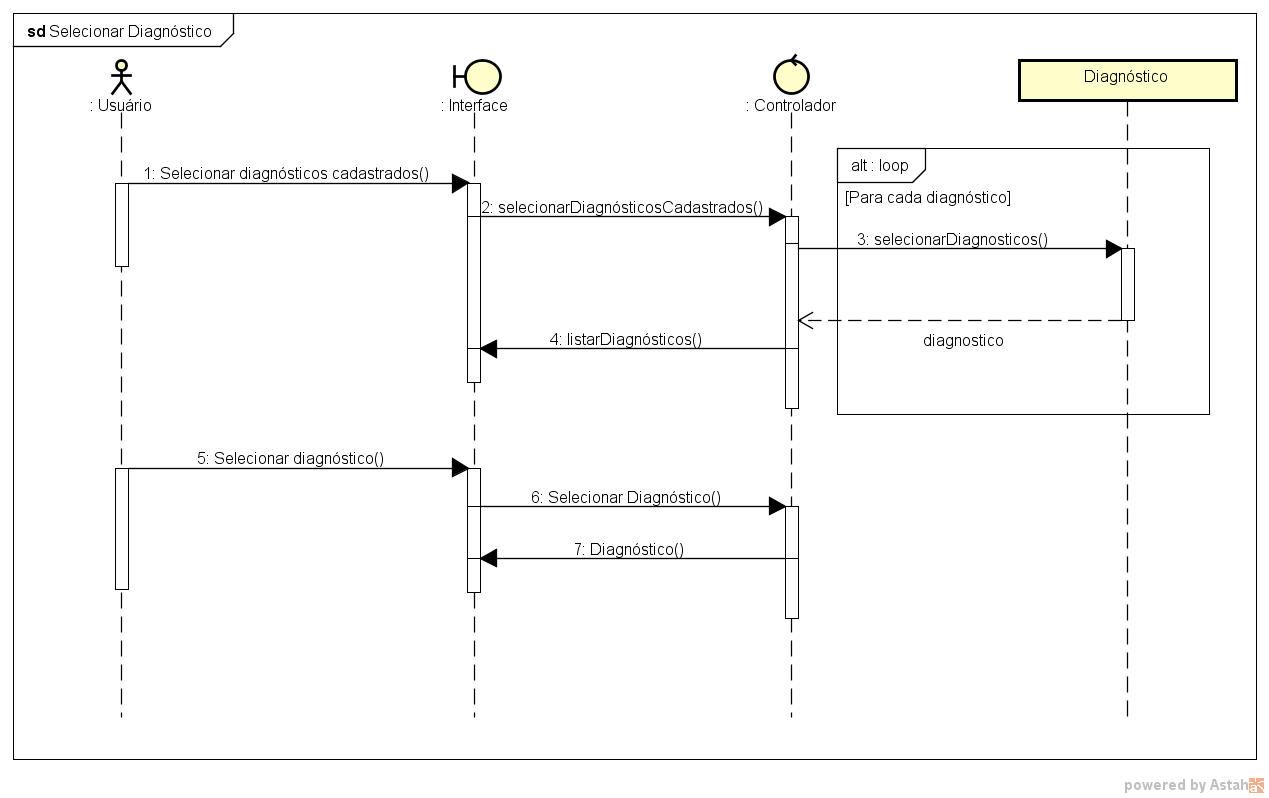
*Figura 5: Diagrama de sequência – Realizar login Google*

* + 1. Selecionar bulário



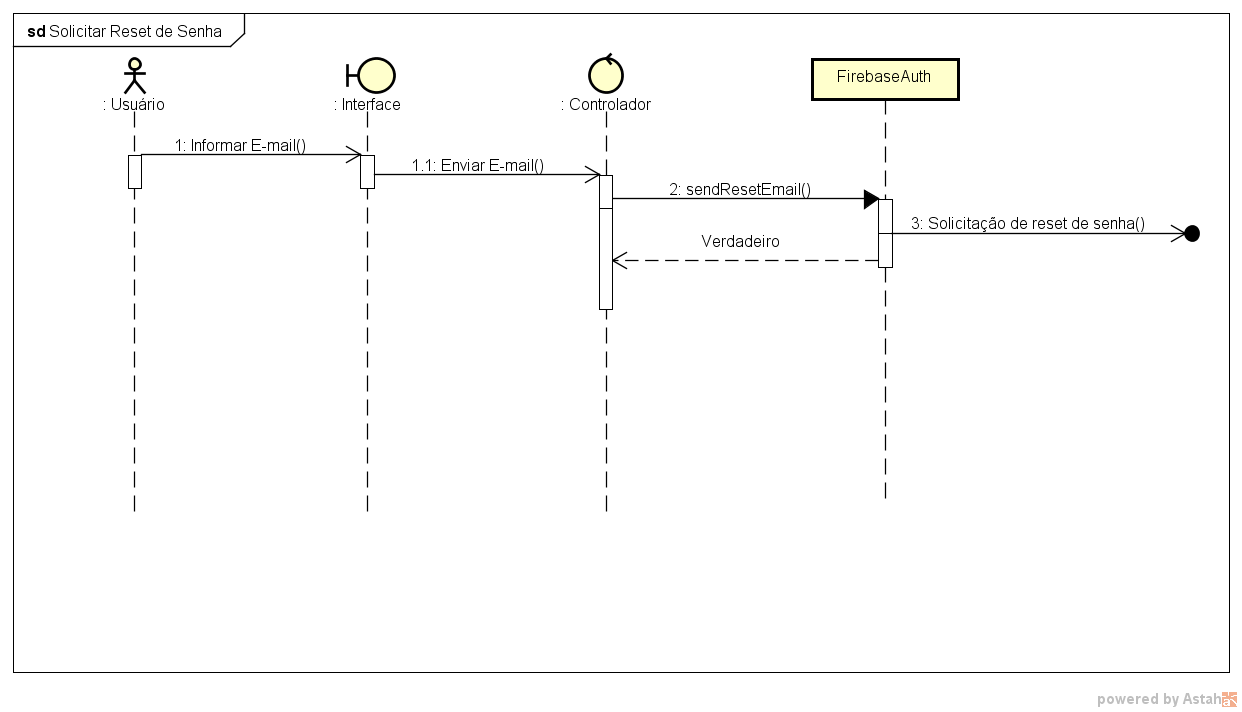
*Figura 6: Diagrama de sequência – Selecionar bulário*

* + 1. Selecionar diagnósticos



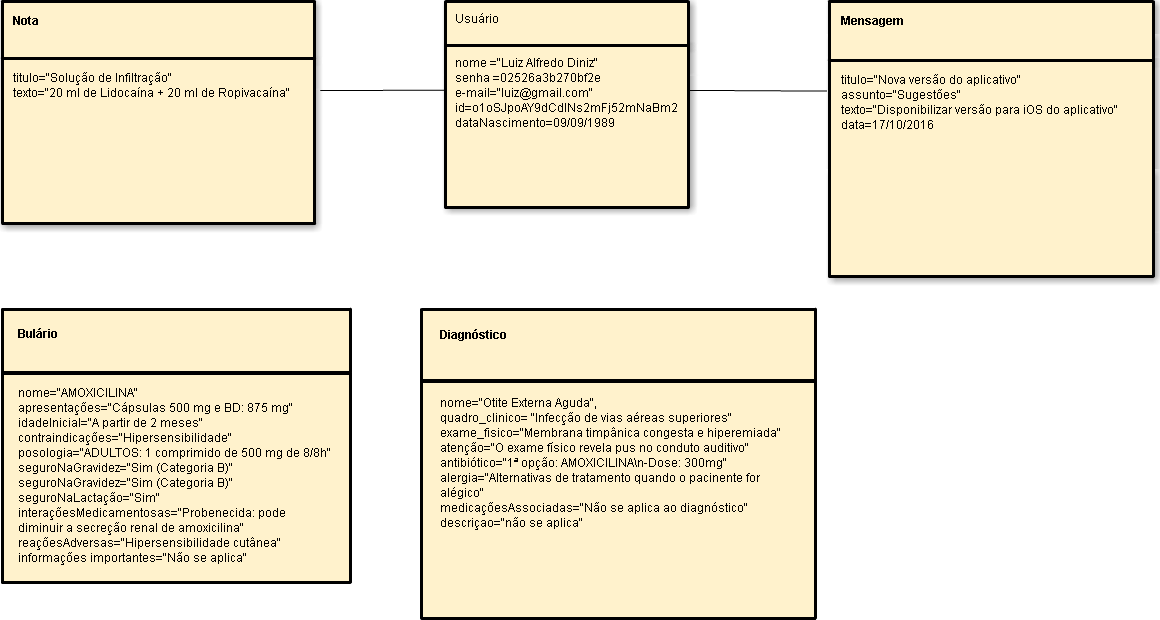
*Figura 7: Diagrama de sequência – Selecionar diagnósticos*

* + 1. Solicitar reset de senha



*Figura 8: Diagrama de sequência – Solicitar reset de senha*

* 1. Diagrama de objetos

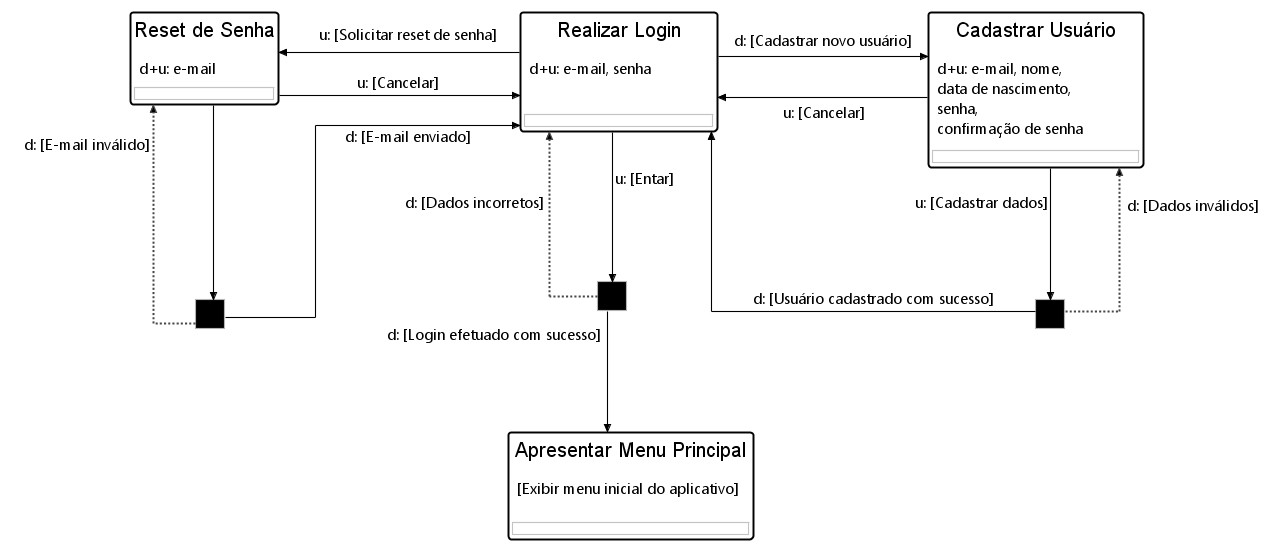


*Figura 9: Diagrama de objetos*

* 1. Diagrama Molic

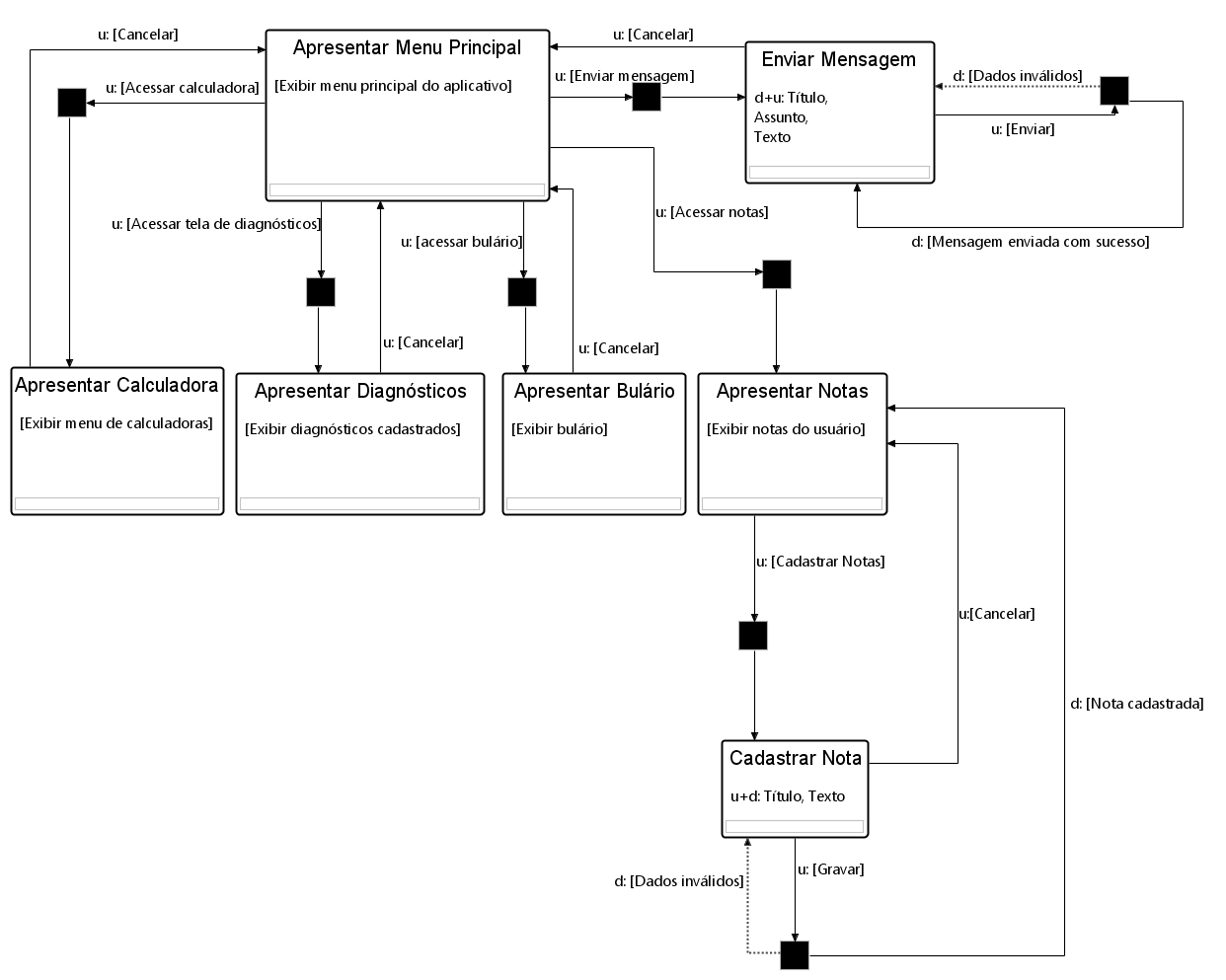
Molic é uma linguagem utilizada para modelar a interação do usuário com o sistema. Durante o processo de desenvolvimento, a Molic funciona com um elo entre as etapas de análise de requisitos e construção de interface (BARBOSA & SILVA, 2007).

* + 1. Apresentar tela de login



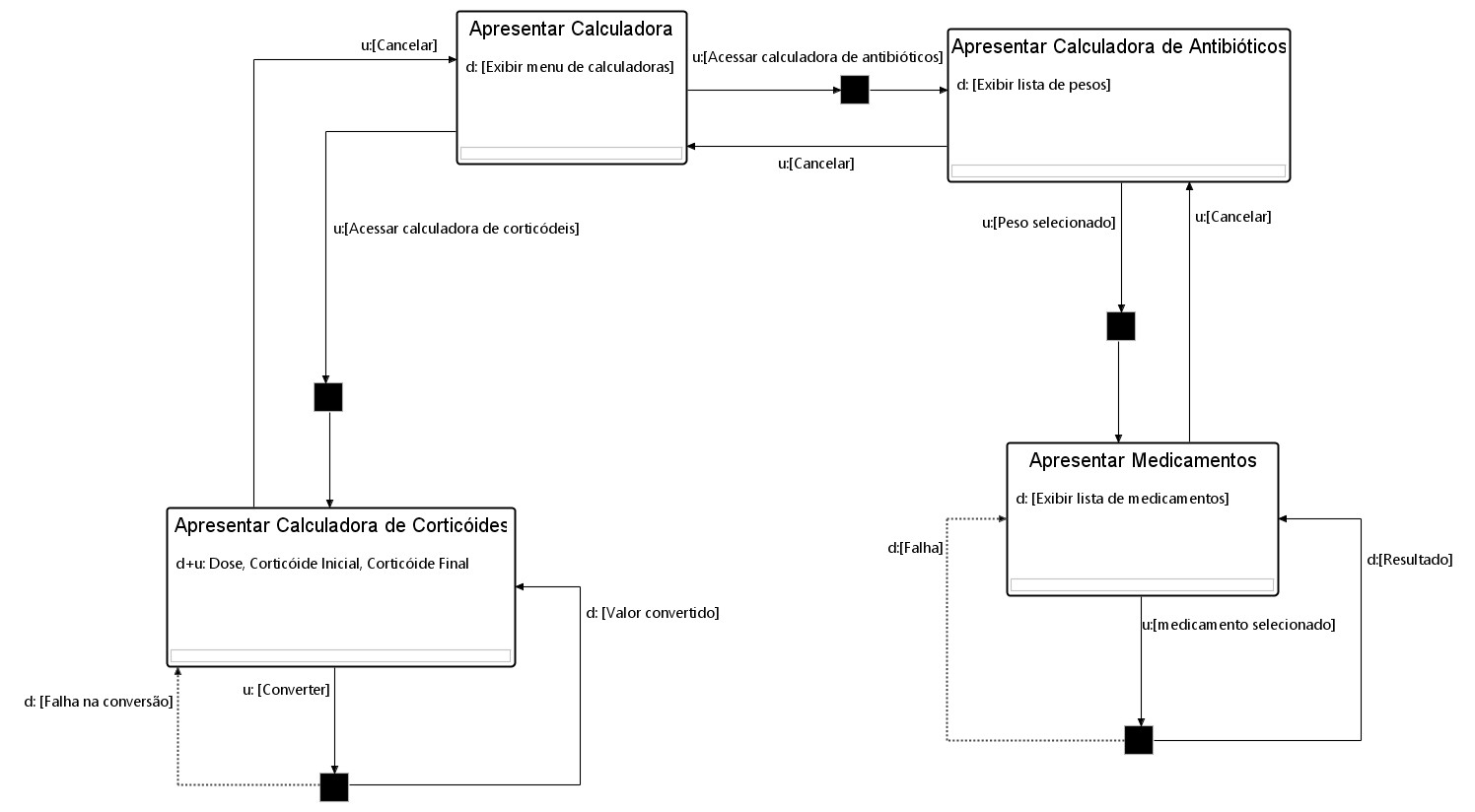
*Figura 10: Diagrama Molic – Apresentar tela de login*

* + 1. Apresentar menu principal



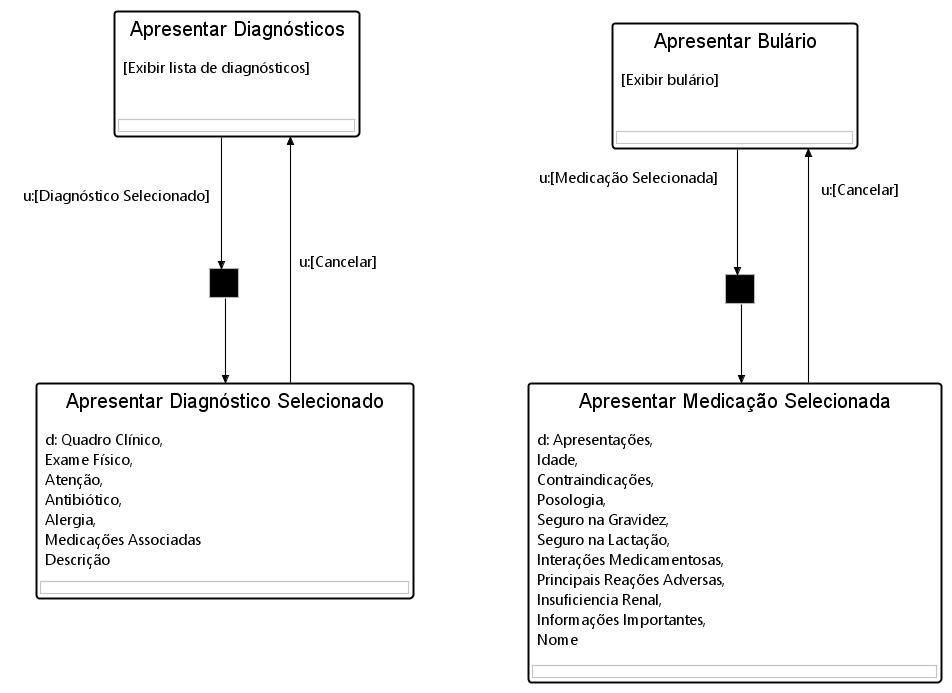
*Figura 11: Diagrama Molic – Apresentar menu principal*

* + 1. Apresentar calculadora



*Figura 12: Diagrama Molic – Apresentar calculadora*

* + 1. Apresentar funcionalidades de consulta



*Figura 13: Diagrama Molic – Apresentar funcionalidades de consulta*

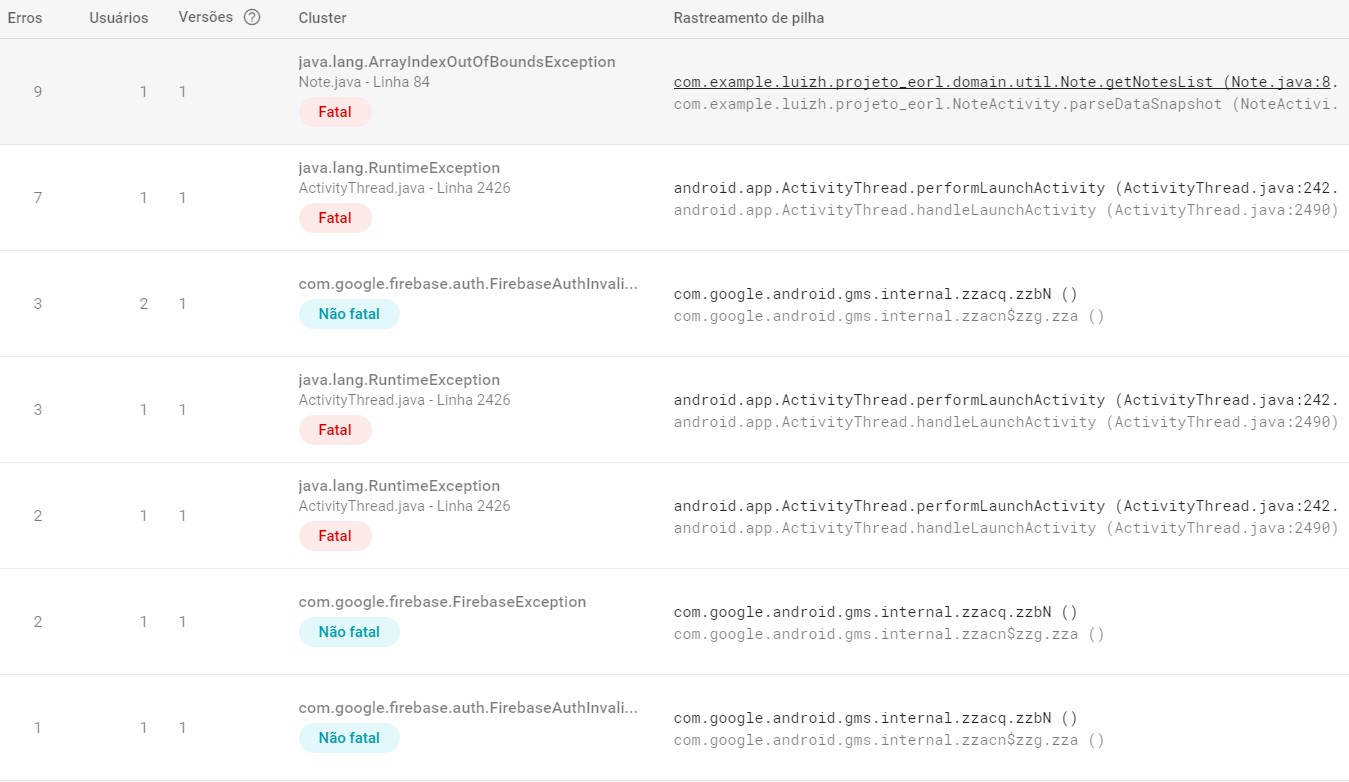
# CONSTRUÇÃO

Neste capítulo, serão abordadas as principais ferramentas utilizadas para apoiar o projeto durante construção, além de apresentar os resultados obtidos.

* 1. Crash Reporting

O Crash Reporting é um recurso disponibilizado pelo Firebase para criar relatórios detalhados de falhas encontradas na aplicação. Através desses relatórios, é possível realizar diagnósticos mais precisos e estabelecer padrões para a ocorrência de um determinado problema, o que contribuiu para a constante evolução da aplicação. Nesta seção, será abordada a utilização dessa ferramenta durante o desenvolvimento do projeto e como ela pode auxiliar na visualização de problemas que ocorrem na aplicação. Na verdade, esse recurso foi útil não apenas durante o desenvolvimento da primeira versão do aplicativo, mas também quando já estiver em produção ou até mesmo na criação de novas funcionalidades.

No início da implementação do aplicativo, as falhas encontradas eram brevemente anotadas em uma planilha, onde era feita uma descrição do problema encontrado para que futuramente elas fossem analisadas via código. A grande vantagem do Crash Reporting está justamente na automatização do processo de registro de falha, uma vez que cada uma delas é automaticamente registrada e incluída em um relatório que poderá ser acessado via Web utilizando o console do Firebase. Isso contribui no desenvolvimento, já que se tem acesso às informações de maneira detalhada, organizada e centralizada, facilitando sua análise e controle.



*Figura 14: Firebase Crash Reporting – Registros de falhas encontrados*

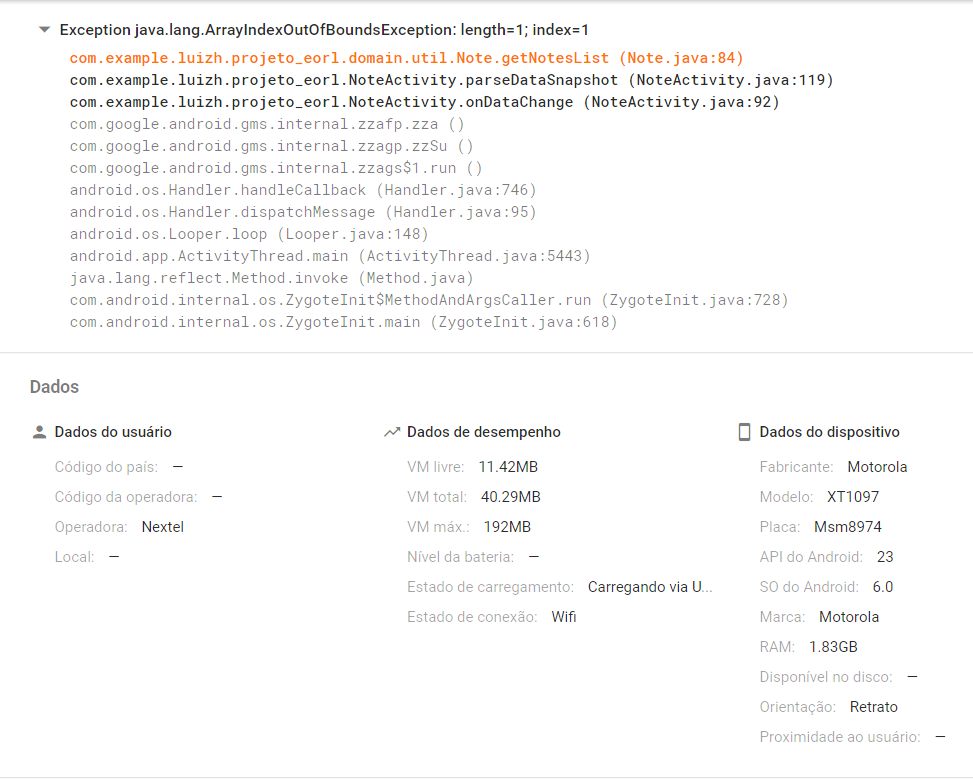
Conforme pode-se visualizar na figura 14, os erros encontrados no aplicativo são agrupados por Stack Trace. Essas informações são geradas na IDE durante o desenvolvimento e incluídas no relatório automaticamente. Apenas a partir delas já é possível verificar os tipos de erro ocorridos, sua severidade, além da quantidade de erros, usuários e versões afetadas, o que já fornece uma visão geral do problema.

Para uma melhor análise deste, pode-se selecionar um determinado erro e visualizar os detalhes de cada ocorrência em forma de um relatório resumido. Nele, tem-se as mensagens de erro completas, exibindo dentre outras informações a linha e a classe onde ocorreu o erro. É possível saber até o modelo do dispositivo onde ocorreu o problema, conforme mostrado na figura 15.

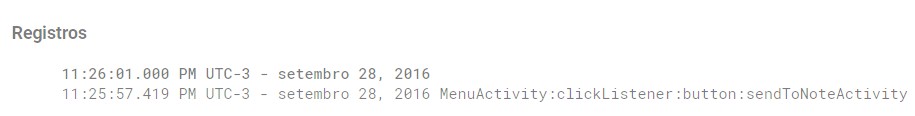
Outro recurso utilizado é o rastreamento dos eventos ocorridos na aplicação até o momento da falha. Cada passo executado pelo usuário, como por exemplo, clicks em botões e acessos a telas, é devidamente registrado e, em caso de erro, essas etapas também serão enviadas e anexadas ao relatório. Isso permite ter um controle ainda melhor sobre cada falha. Conforme pode-se ver na figura 16, antes

da falha ocorrer o usuário executou o método sendToNoteActivity, que é chamado sempre que se clica no botão Notas do menu principal.

Além dos benefícios já comentados, esse recurso de rastreamento também facilita na reprodução e documentação do erro.



*Figura 15: Firebase Crash Reporting – Resumo detalhado do erro*



*Figura 16: Firebase Crash Reporting – Rastreamento do Erro*

* 1. GitHub

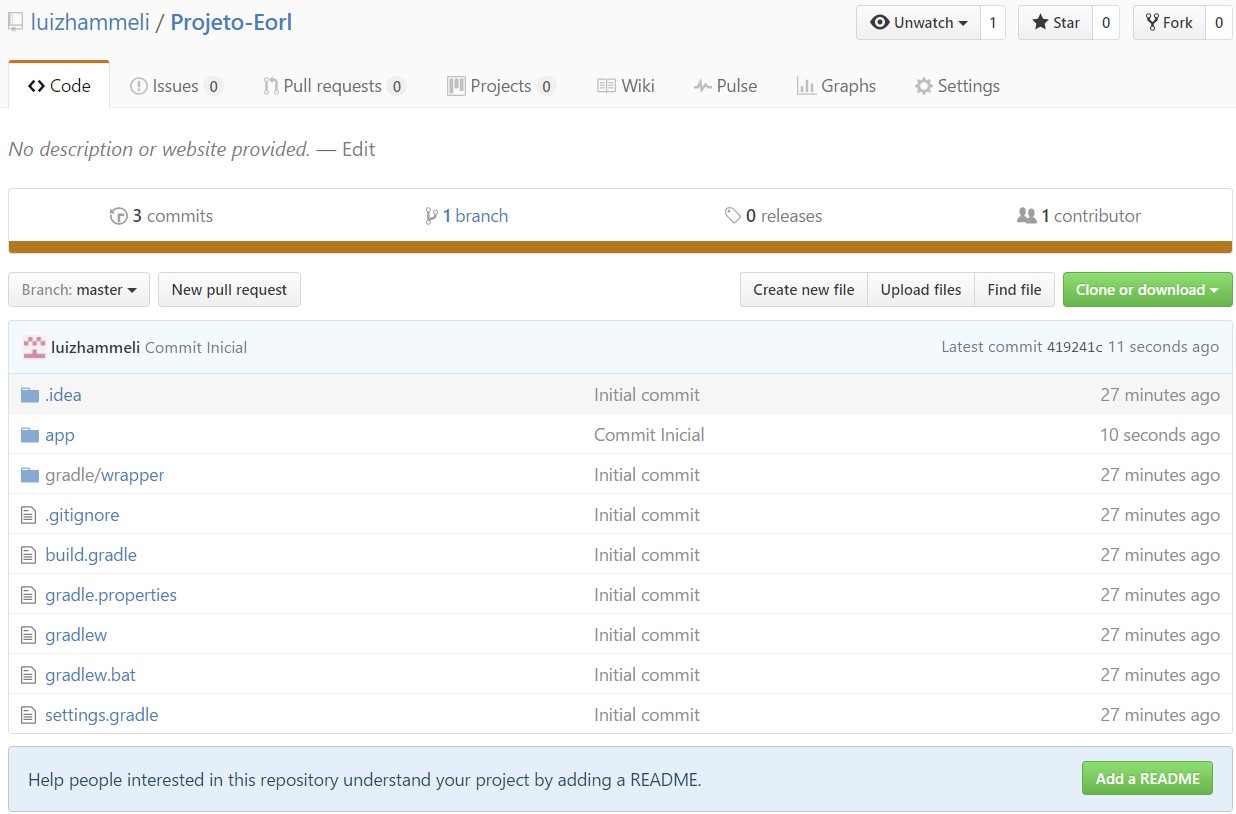
O Github é um site onde é possível carregar uma cópia de um repositório Git, que é um sistema de controle de versão desenvolvido para gerenciar as alterações feitas nos arquivos de um projeto ao longo do tempo (BELL & BEER, 2014).

Uma das grandes vantagens do Github é a disponibilização de um local centralizado para compartilhar o projeto utilizando uma interface Web. O Github possui diversos outros recursos que podem somar muito durante toda a etapa de desenvolvimento. Dentre eles, os utilizados nesse projeto atendem às seguintes necessidades:

* + 1. Gerenciamento de Configuração

O Github é usado como repositório central, responsável por armazenar todos os arquivos do projeto Android, controlando os arquivos do projeto a cada mudança realizada. Neste cenário, eles serão baixados do Github para a máquina de desenvolvimento e, a cada modificação realizada, os arquivos seriam novamente enviados para o Github, atualizando o repositório central.

Após finalizar e publicar a primeira versão do aplicativo, qualquer nova modificação, seja ela para correção de bugs ou inclusão de novas funcionalidades, seria feita através da criação de um branch, ou seja, uma cópia do repositório associada a uma determinada mudança ou melhoria. Após terminar uma demanda referente a um determinado branch criado, realizando todos os testes necessários, faz-se um merge com a versão master do projeto.



*Figura 17: Projeto armazenado no Github*

* + 1. Documentação de Requisitos e Bugs

O Issues é uma ferramenta disponibilizada para cadastro de bugs ou especificação de novos requisitos no projeto (BELL & BEER, 2014). Conforme comentado na seção 5.1, os erros encontrados no aplicativo são automaticamente capturados e disponibilizados em um relatório para análise. Após esse procedimento, caso necessário, pode-se criar um bug associado a essa falha e documentá-lo utilizando essa ferramenta para que futuramente ele possa ser corrigido. Isso fornece um melhor controle de todas as falhas encontradas e atividades pendentes. Por exemplo, na figura 14, observa-se que ocorreram nove casos de um erro fatal onde foi disparada a exceção do tipo ArrayBoundException. Analisando esse problema (Figura 15), pode-se julgar necessário documentá-lo para futuramente trabalhar em sua correção. O Issue permite até mesmo que, ao cadastrar um bug, ele seja marcado com o status de aberto (Figura 18), ou seja,

que ainda não foi corrigido. Após sua correção, ao subir com a nova versão, marca- se o bug como fechado, ou seja, resolvido.



*Figura 18: Bug cadastrado utilizando a ferramenta Issue do Github*

Os requisitos mapeados na seção 4.1.1, também foram adicionados na ferramenta Issue, desta vez não como bug, mas como um quadro de atividades pendentes onde também utilizou-se uma lógica semelhante àquela aplicada para os bugs. Os requisitos foram adicionados a ferramenta e marcados como abertos, ou seja, disponíveis para serem desenvolvidos. Após o desenvolvimento, eles foram fechados e testados. Caso um erro seja encontrado, seguirá o fluxo associado aos bugs.

* + 1. Gerenciamento de Conteúdo

O Github possui uma Wiki, ou seja, uma ferramenta para gerenciamento de conteúdo. Utilizou-se essa ferramenta para armazenar toda a documentação do projeto, criando um repositório único para todas as informações.

* 1. Apresentação das telas

A partir do processo de análise desenvolvido nos capítulos anteriores, são apresentados nesta seção os resultados obtidos, exibindo as principais telas do aplicativo e suas funcionalidades. As próximas seções serão organizadas de acordo com a funcionalidade de cada tela.

* + 1. Tela de Login Padrão

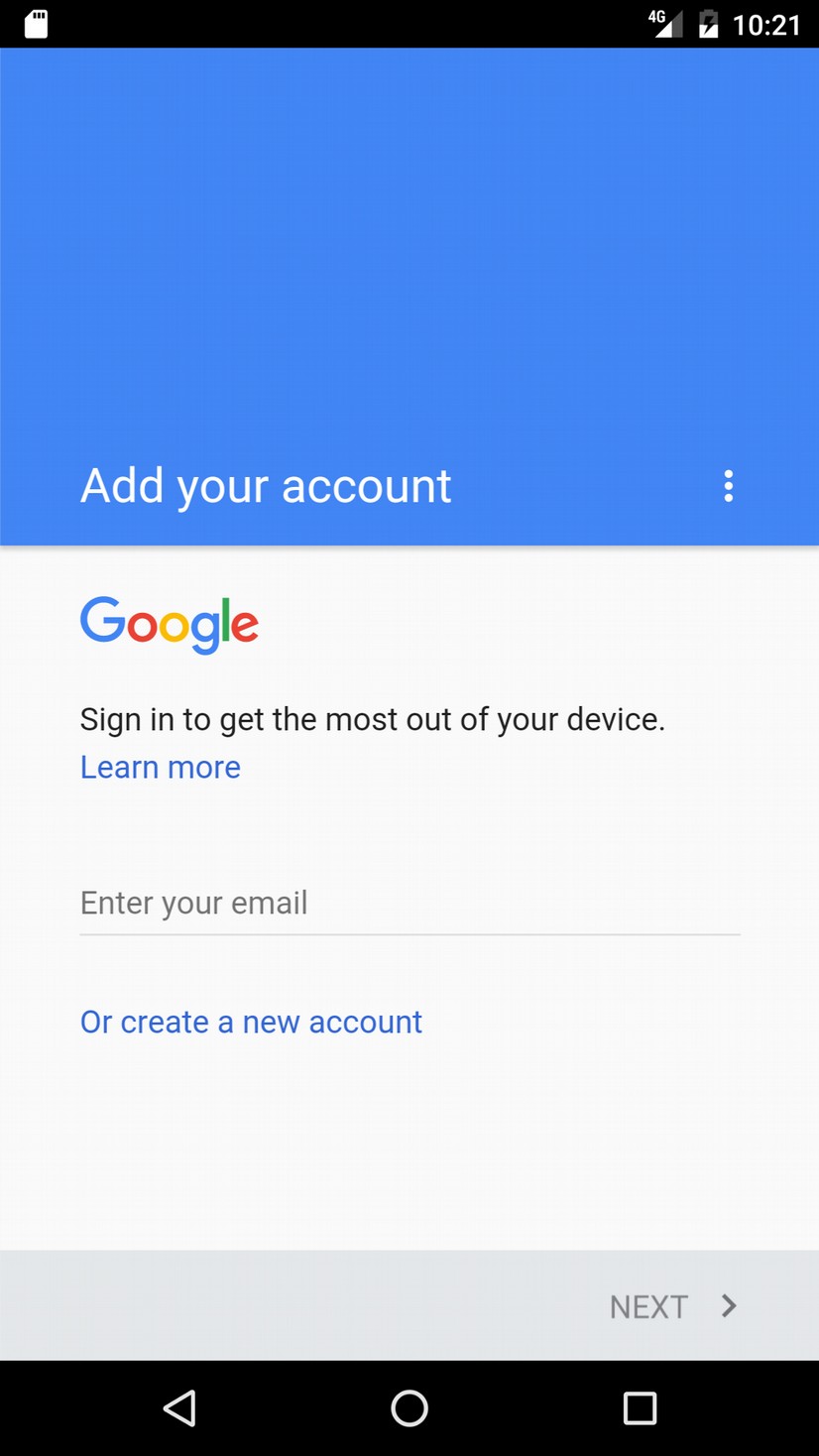
Nesta tela, o usuário insere seus dados de e-mail e senha para efetuar seu acesso ao aplicativo ou utiliza seu login Google disponibilizado pelo botão “Logar com o Google”. Caso o usuário não tenha realizado seu cadastro, ele poderá selecionar a opção “Cadastre-se” e será direcionado para a tela correspondente.

Outra situação tratada pela tela de login é o direcionamento para tela de reset de senha através da opção “Esqueci Minha Senha”, na qual o usuário que a esqueceu terá a possibilidade de criar uma nova. (Figura 19)

* + 1. Tela de Autenticação Google

Ao selecionar a opção “Logar com o Google” da tela de login (Figura 19), o usuário é direcionado para a tela de autenticação Google fornecida por sua API (Figura 20). Nela, o usuário informará os dados cadastrais de sua conta Google que serão devidamente validados fornecendo acesso ao aplicativo em caso de sucesso na operação. Caso o usuário não possua uma conta Google, a tela fornece um recurso para a criação de uma.

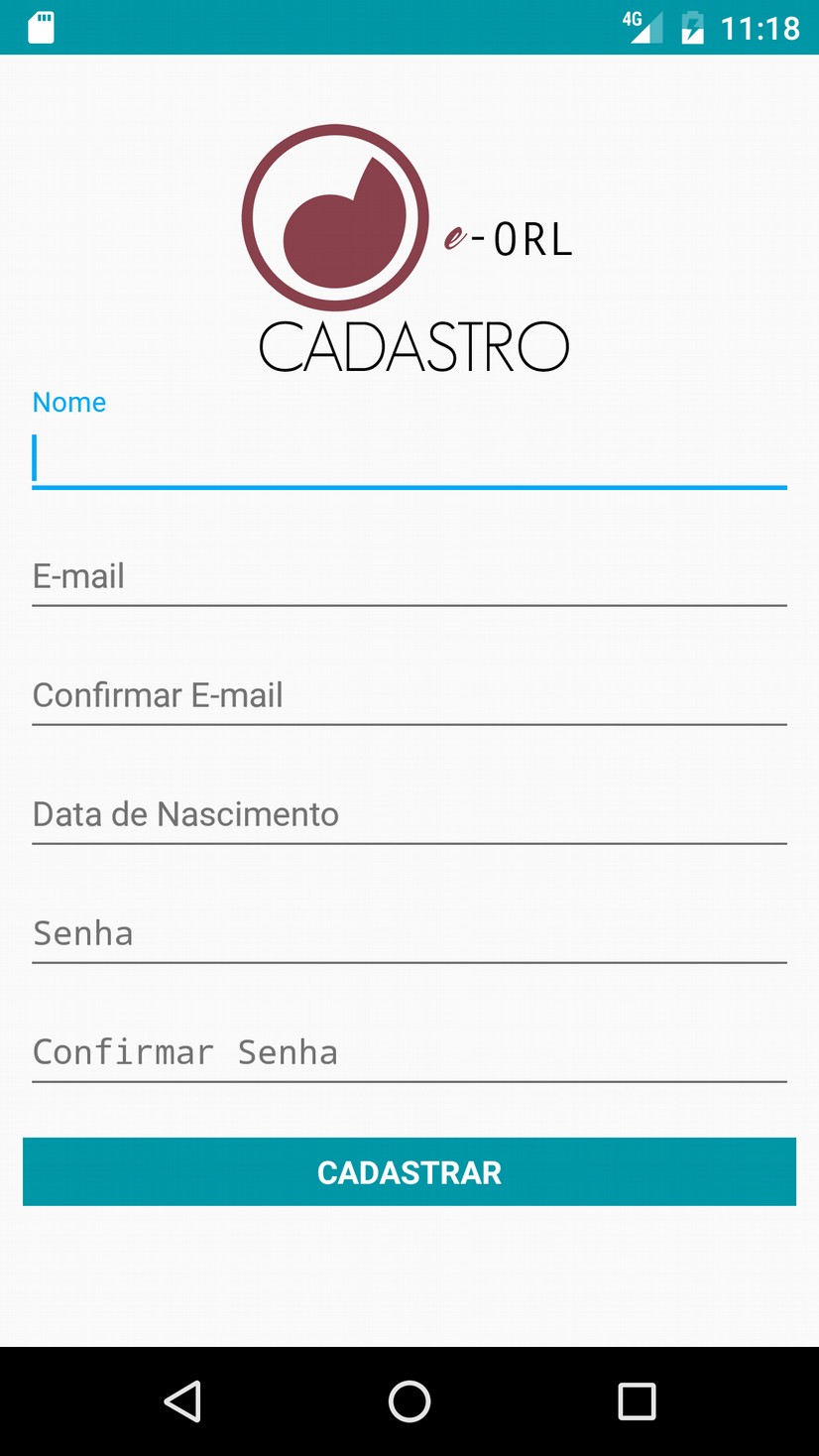
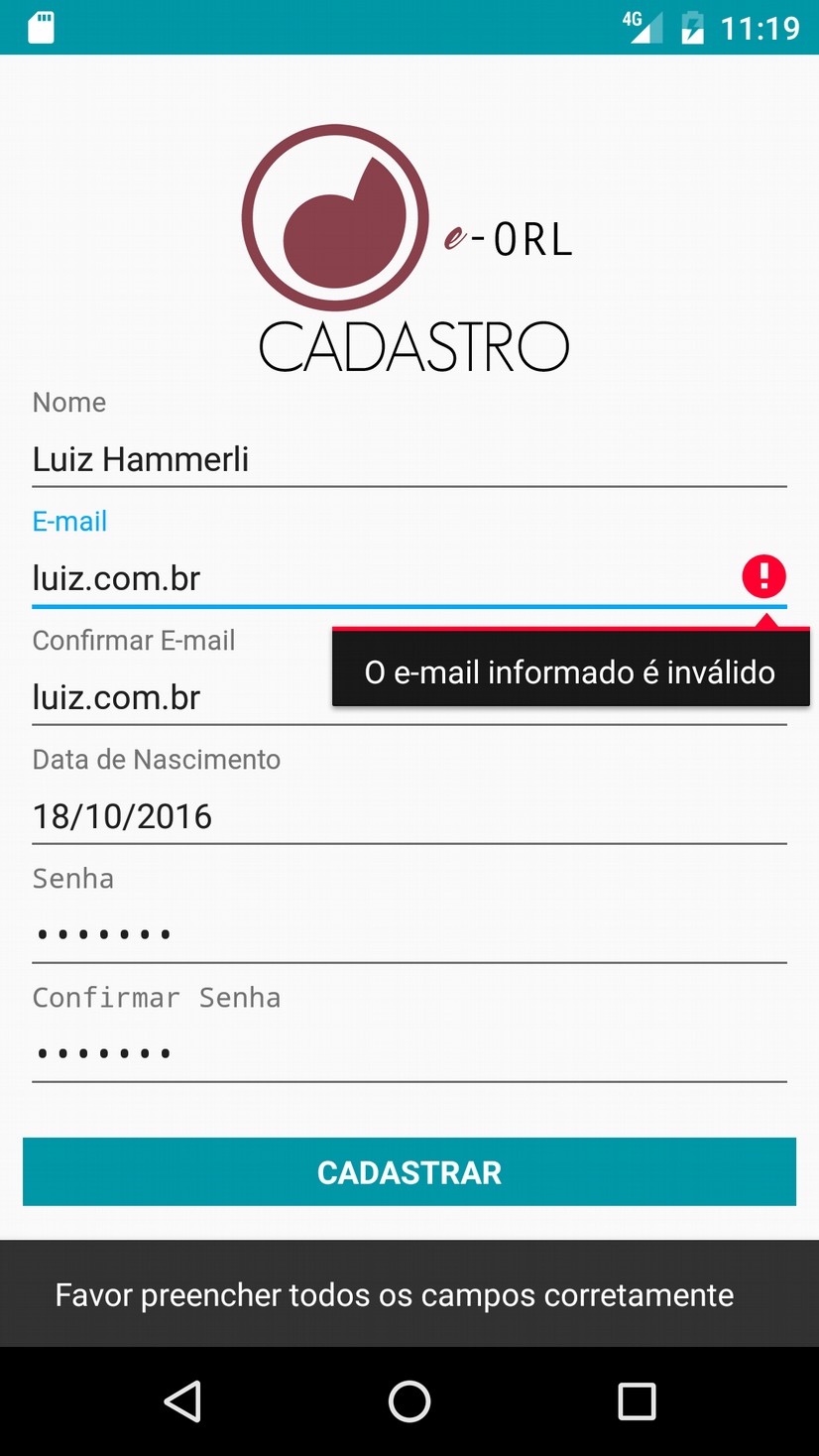
Essa funcionalidade é útil para agilizar o acesso ao aplicativo. Uma vez que o usuário já possua uma conta Google ativa ele não precisará efetuar o cadastro padrão do aplicativo.

*Figura 19: Tela de login Figura 20: Tela de login Google*

* + 1. Tela de Cadastro de Usuário

Nesta tela, o usuário insere suas informações e submete os dados (Figura 21). O aplicativo valida os dados enviados informando sucesso ou não na operação. Em caso de sucesso, o usuário estará autorizado para acessar o aplicativo, utilizando seu e-mail e senha cadastrados. Caso contrário, o aplicativo informará o problema, permitindo ao usuário corrigi-lo. (Figura 22)

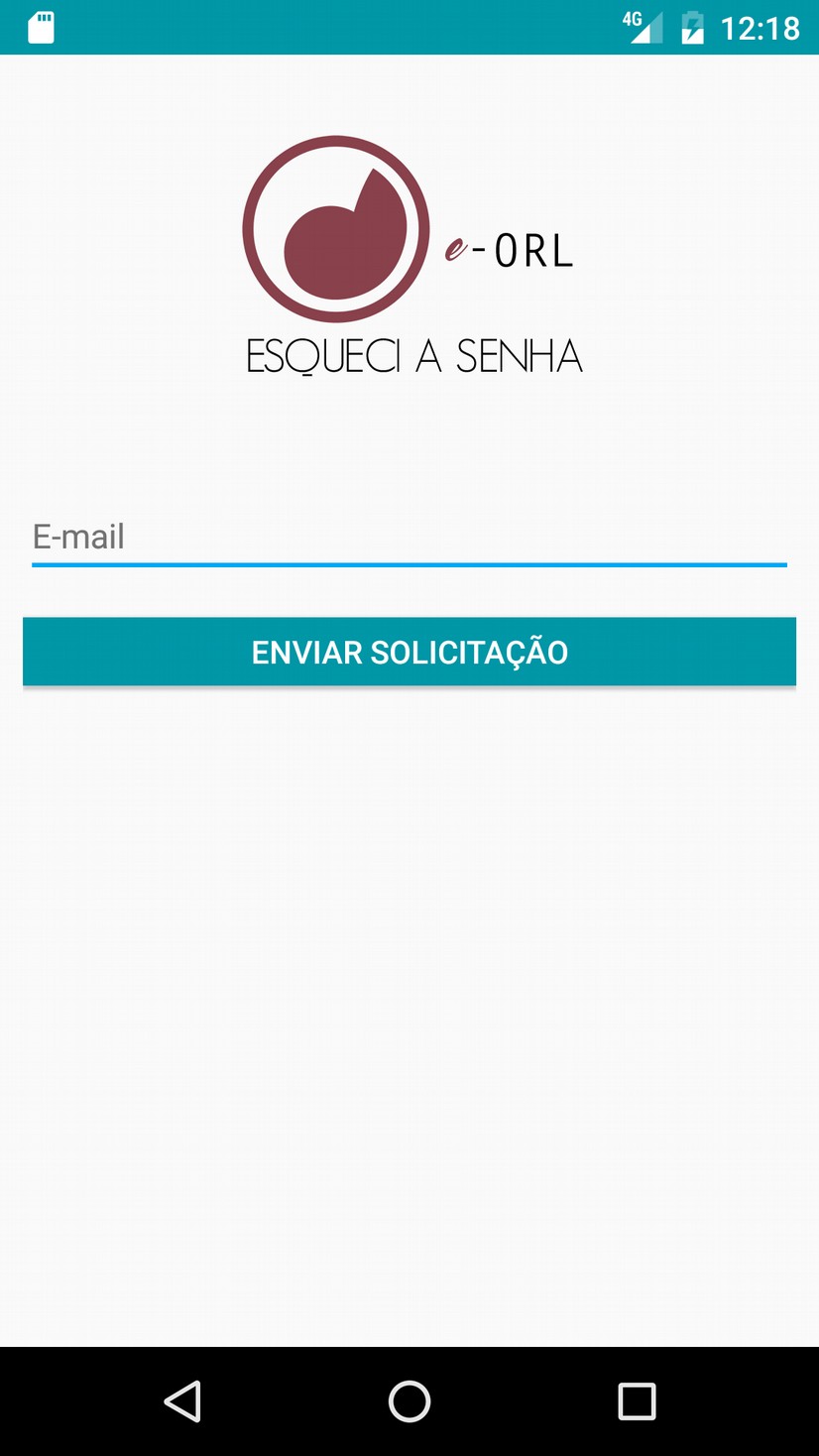
 

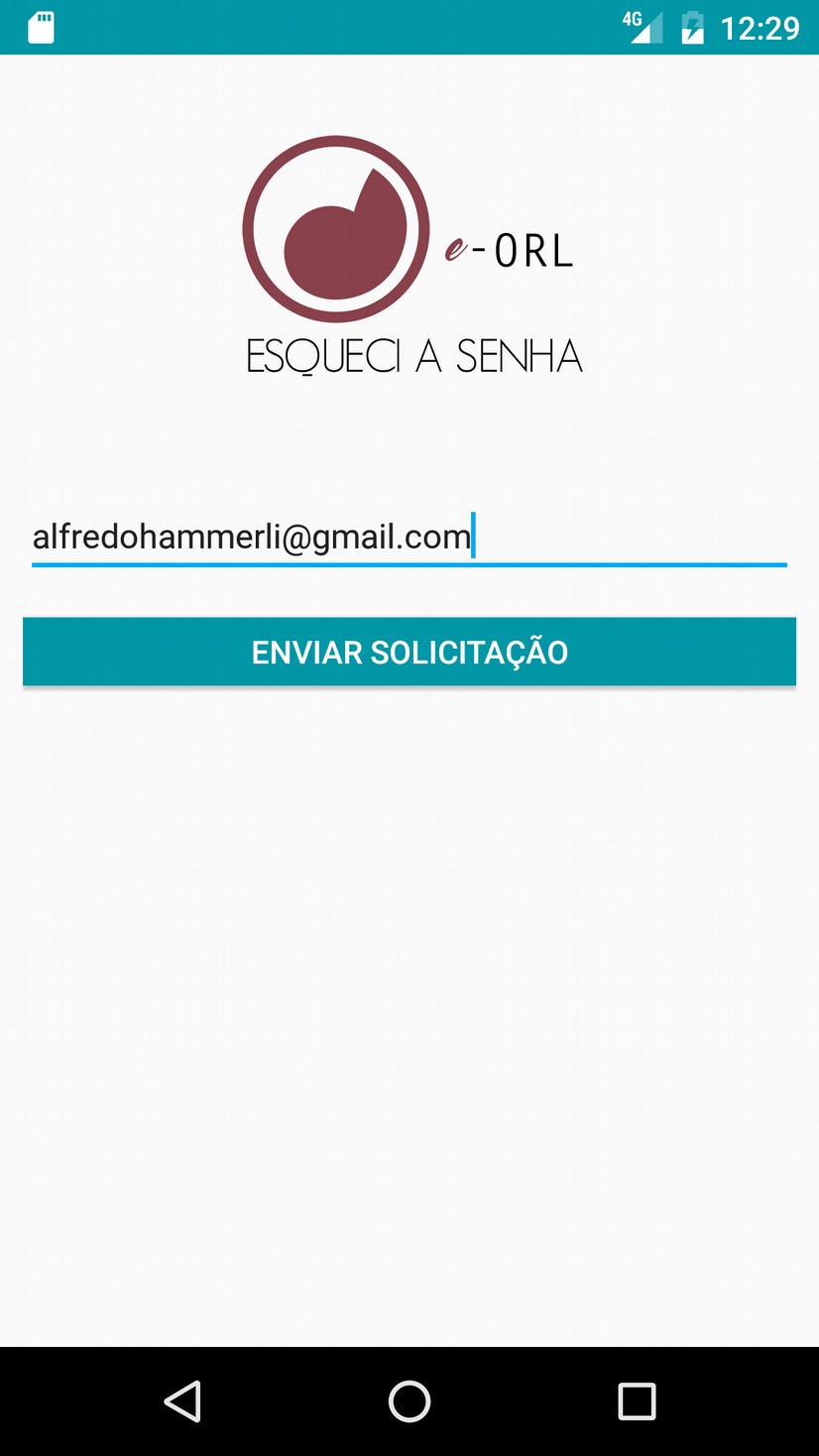
*Figura 21: Tela de cadastro de usuário Figura 22: Tela de cadastro de usuário -*

*Dados inválidos*

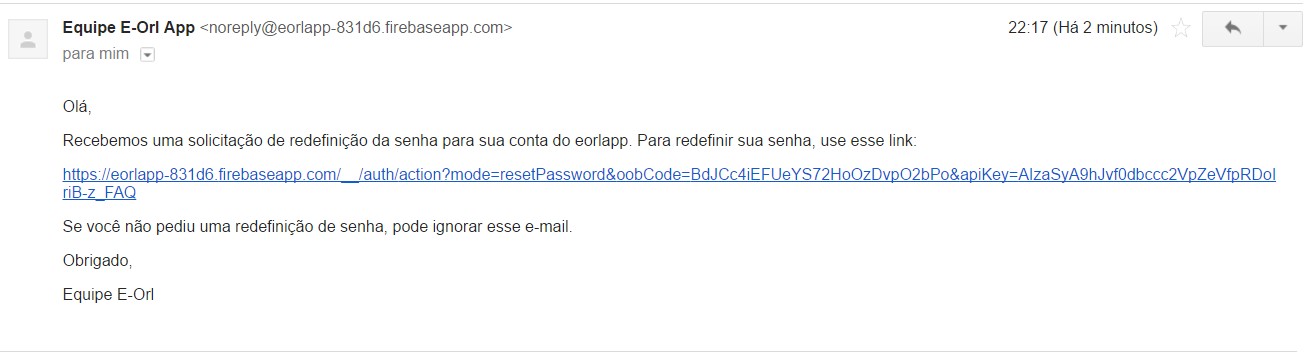
* + 1. Tela de Alteração de Senha

Nesta tela, o usuário insere o endereço de e-mail cadastrado no aplicativo (Figura 23). Um e-mail para redefinição de senha será enviado, onde constará um link para o cadastro da nova senha. Após alguns minutos, caso o usuário não efetue a redefinição da senha, o link expirará por questões de segurança.



*Figura 23: Tela redefinição de senha Figura 24: Tela redefinição de senha*

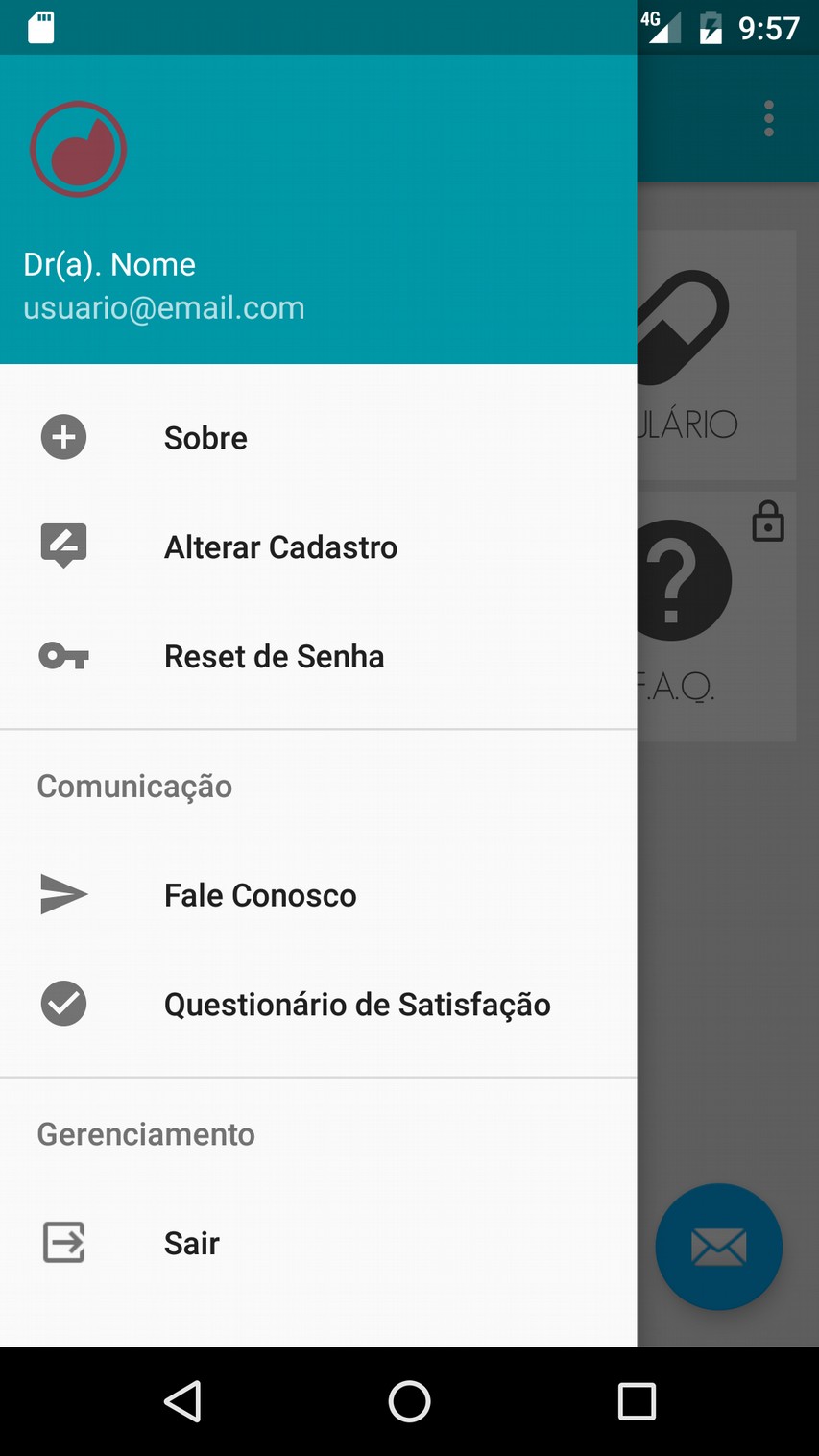
*preenchida*



*Figura 25: E-mail de redefinição de senha*

* + 1. Menu Inicial

Após ter seu acesso validado, o usuário será direcionado para o menu inicial da aplicação, onde terá acesso a todas as suas funcionalidades. Além das opções exibidas na figura 26, ainda há um menu lateral deslizante que é chamado ao clicar no botão do canto superior esquerdo da tela, ou deslizando o dedo da esquerda para direita na tela (Figura 27). As funcionalidades “Anatomia Radiológica” e “F.A.Q” exibidas no menu inicial ainda não estão disponíveis nessa versão e serão abordadas com mais detalhes no capítulo 6.

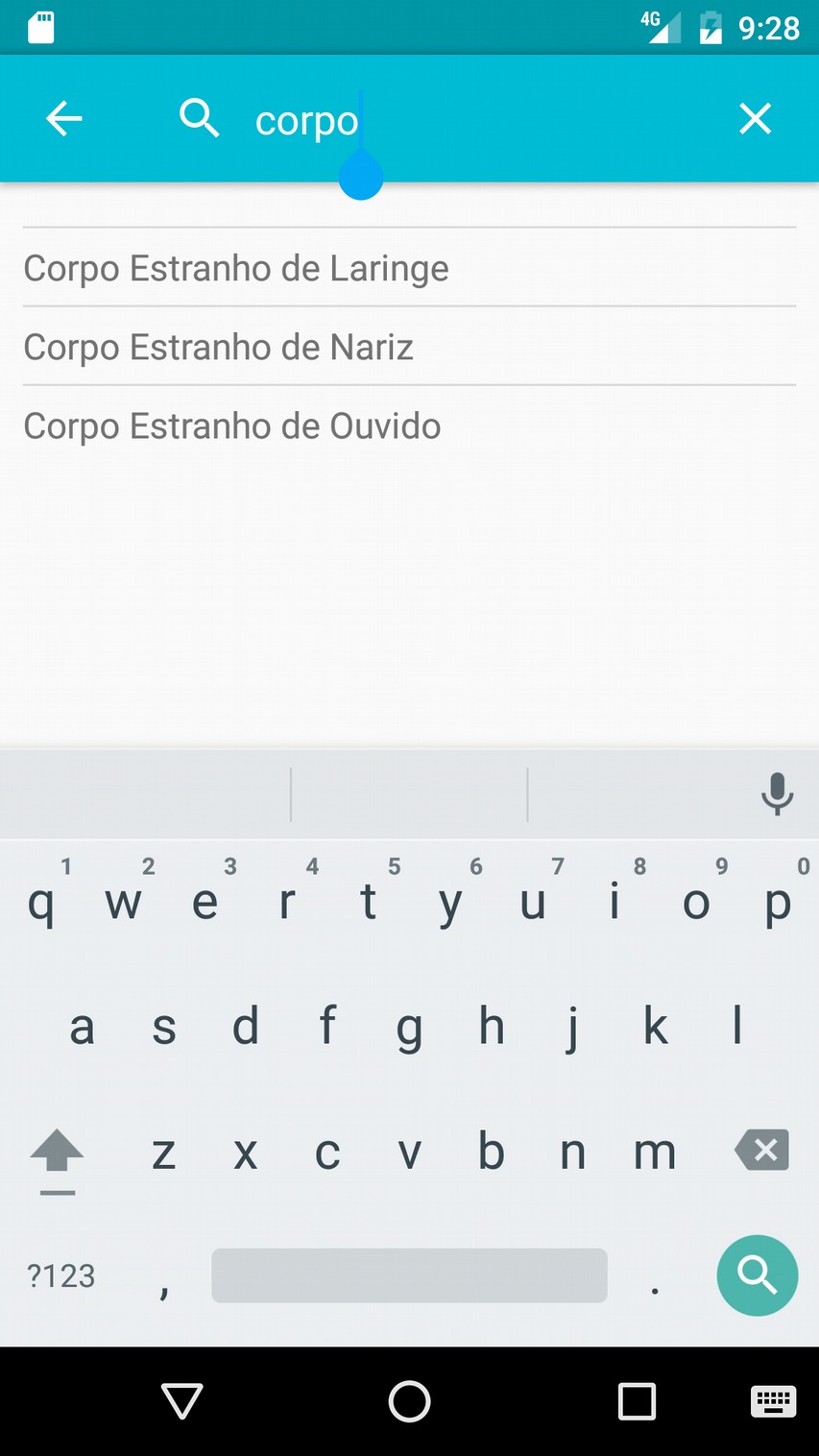


*Figura 26: Menu inicial Figura 27: Menu lateral*

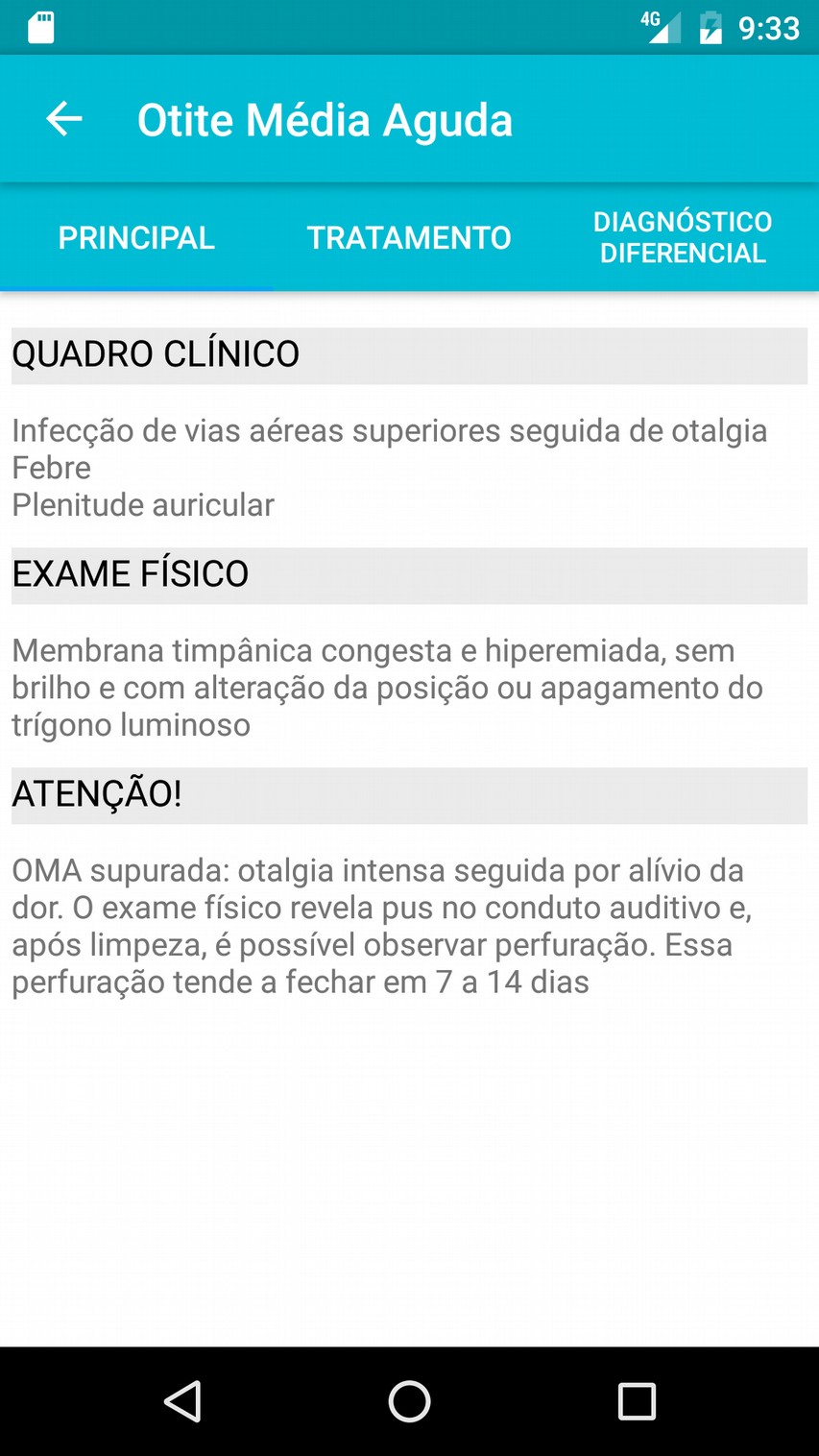
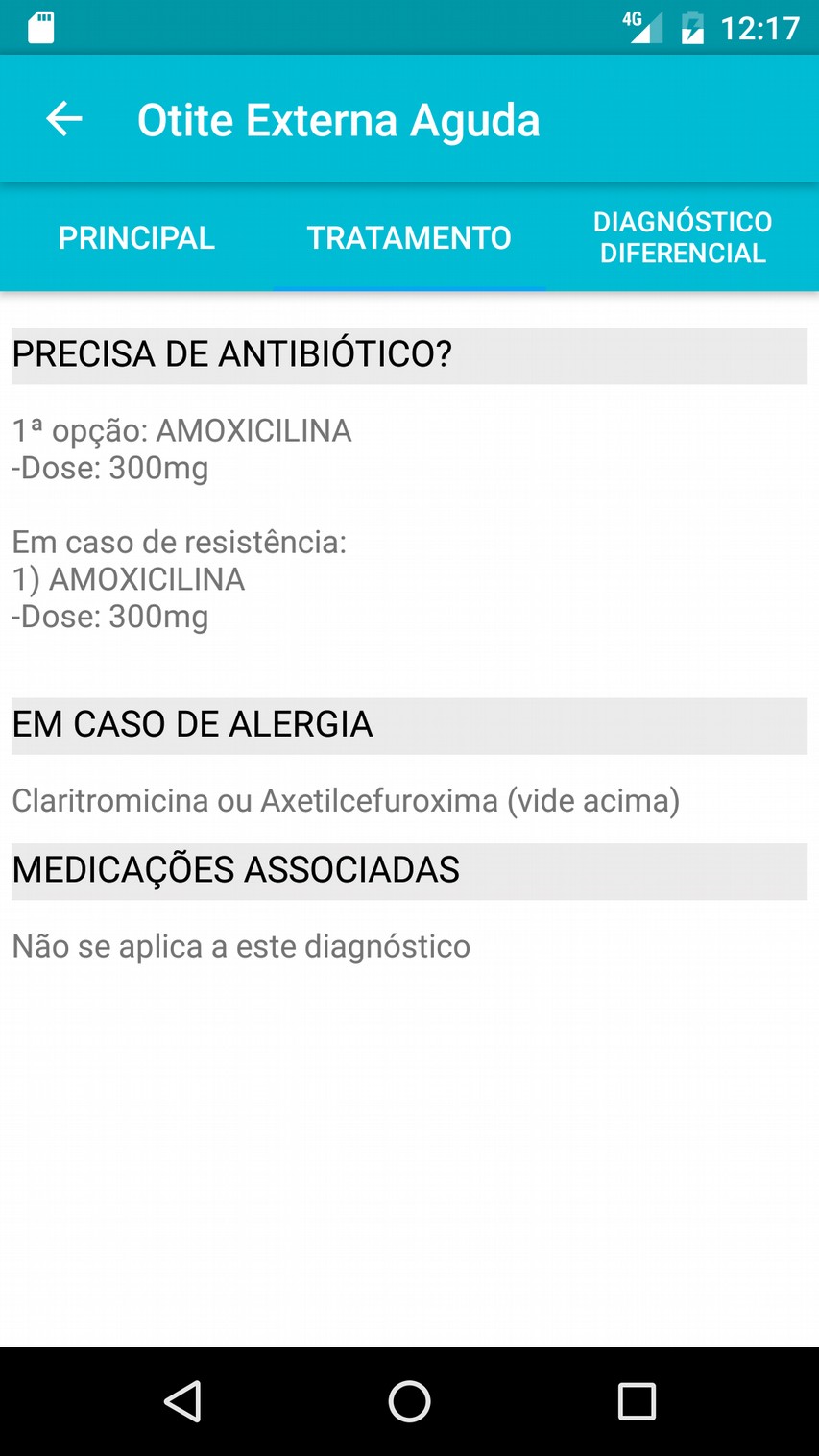
* + 1. Tela de Diagnósticos

Selecionando a opção “Diagnósticos” no menu inicial, o aplicativo direciona o usuário para uma tela contendo uma lista de diagnósticos cadastrados (Figura 28). Caso o usuário queira buscar um nome específico, pode selecionar o botão de lupa no canto superior direito da tela e digitar o nome que deseja procurar. À medida que vai escrevendo, o aplicativo filtra os resultados. (Figura 29)

Selecionando um determinado item da lista, o aplicativo direciona para a tela onde é apresentado o conteúdo para consulta. Esta tela possui três abas, nas quais o conteúdo é divido. (Figura 30)



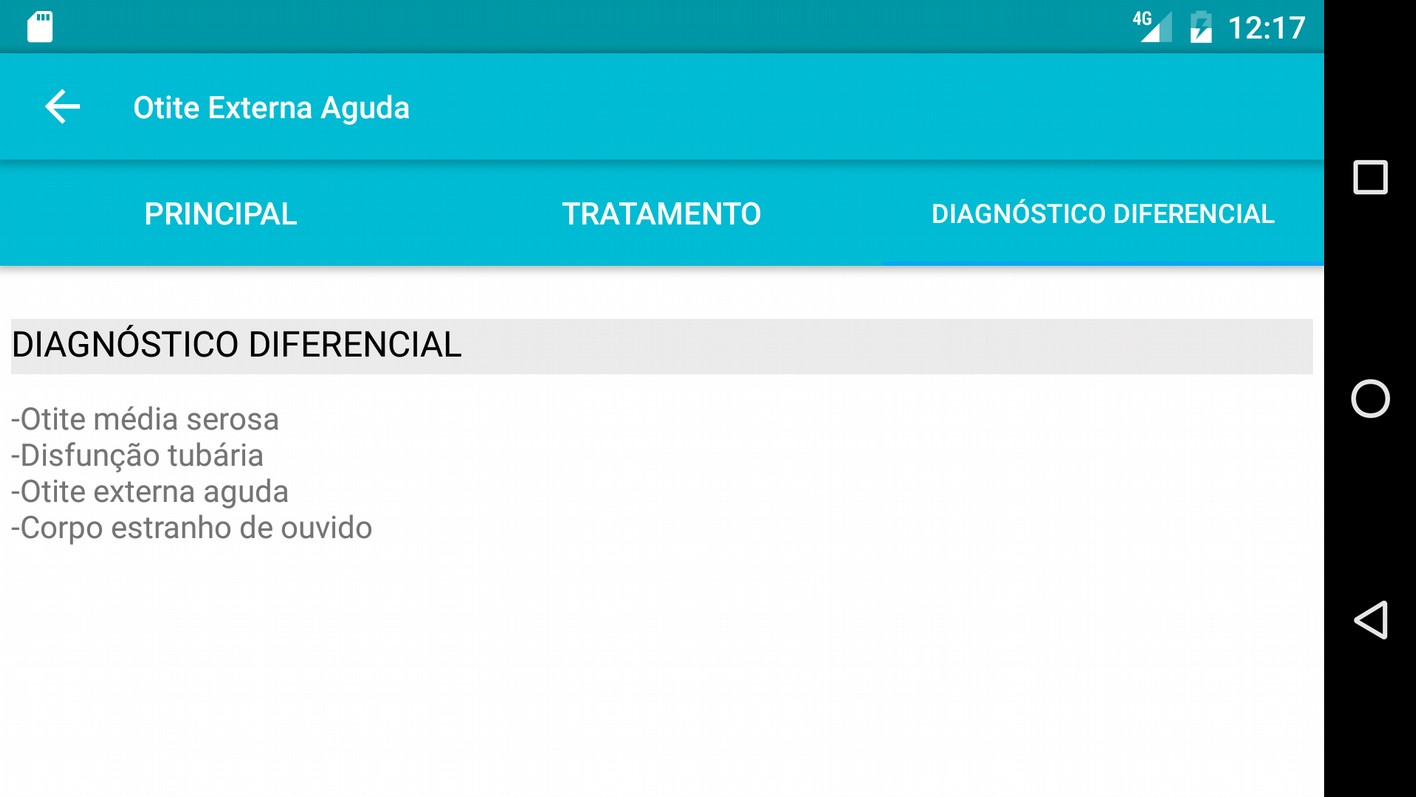
*Figura 28: Tela de diagnósticos Figura 29: Filtragem de diagnósticos*

*Figura 31: Conteúdo do diagnóstico selecionado – aba principal*

*Figura 30: Conteúdo do diagnóstico*

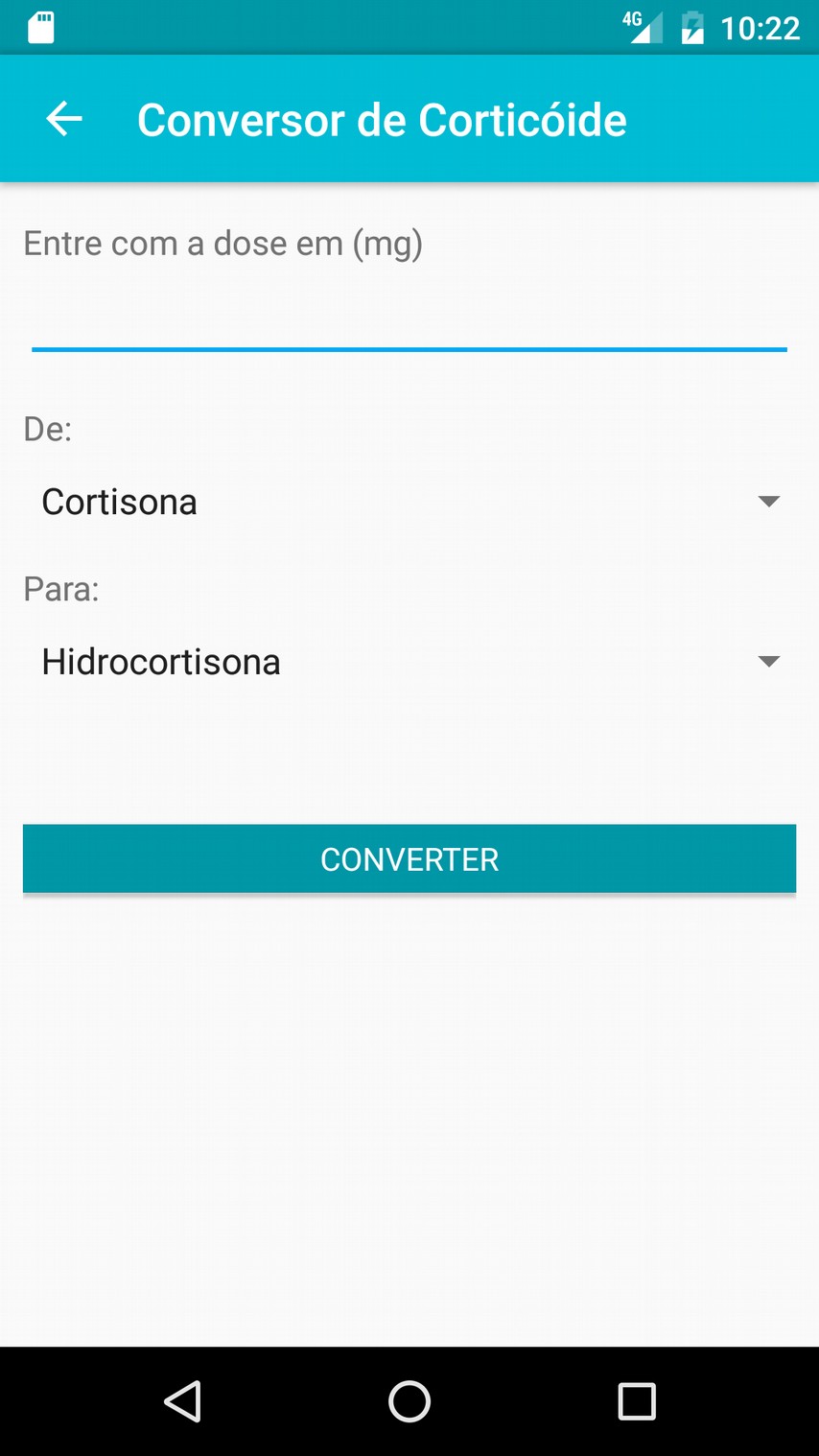
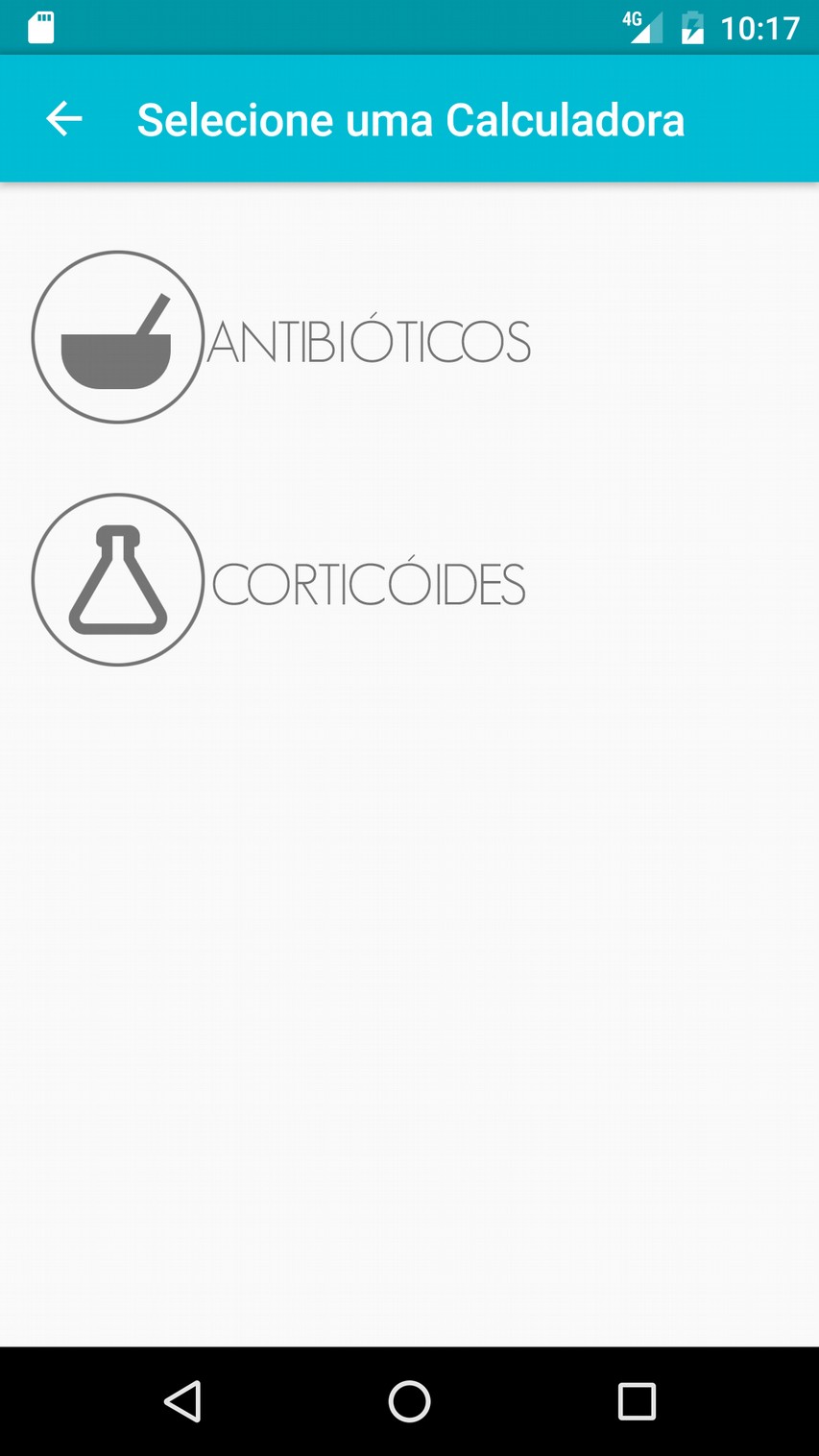
*selecionado – aba Tratamento*



*Figura 32: Conteúdo do diagnóstico selecionado – aba Diagnóstico*

* + 1. Calculadoras

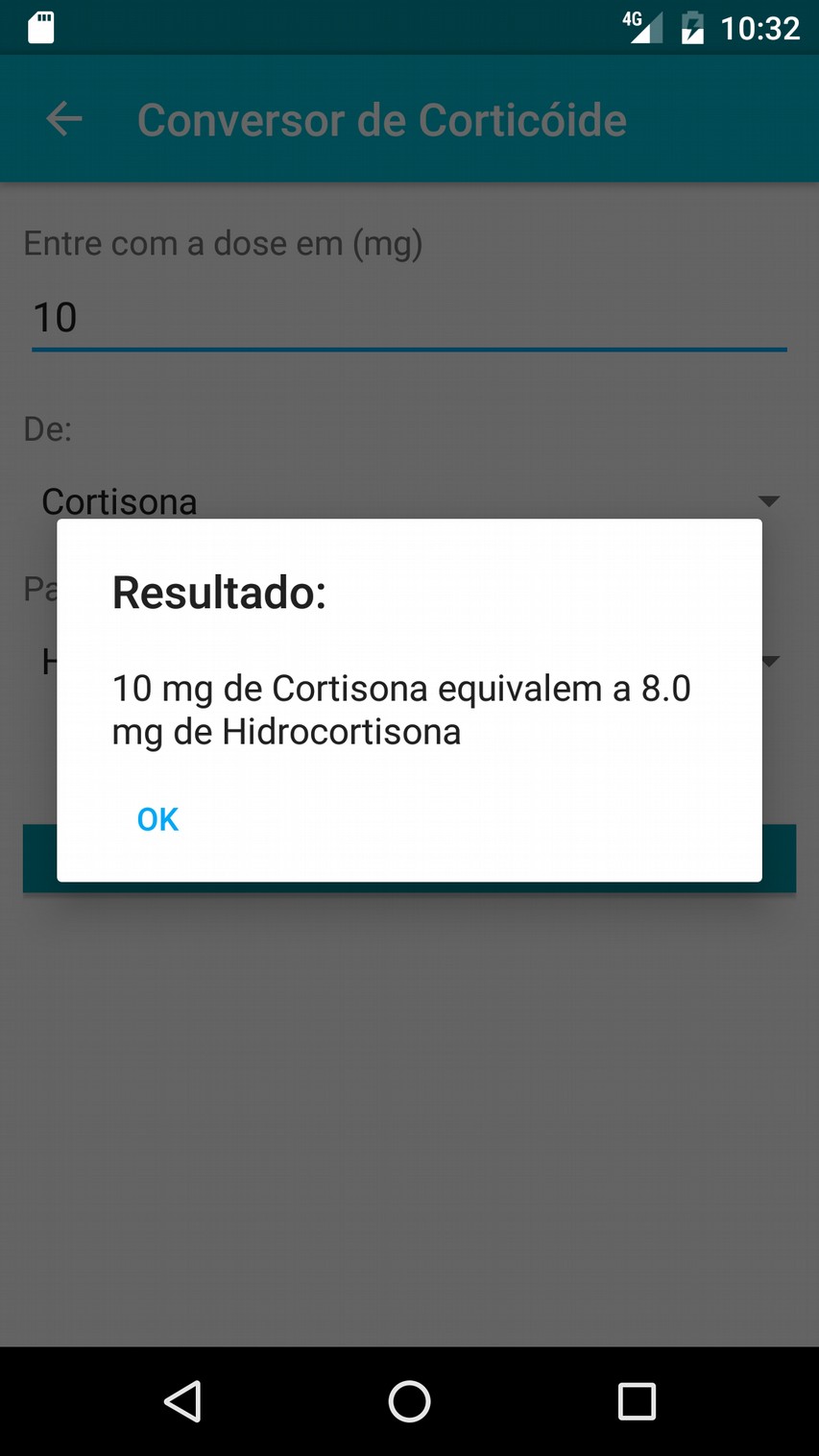
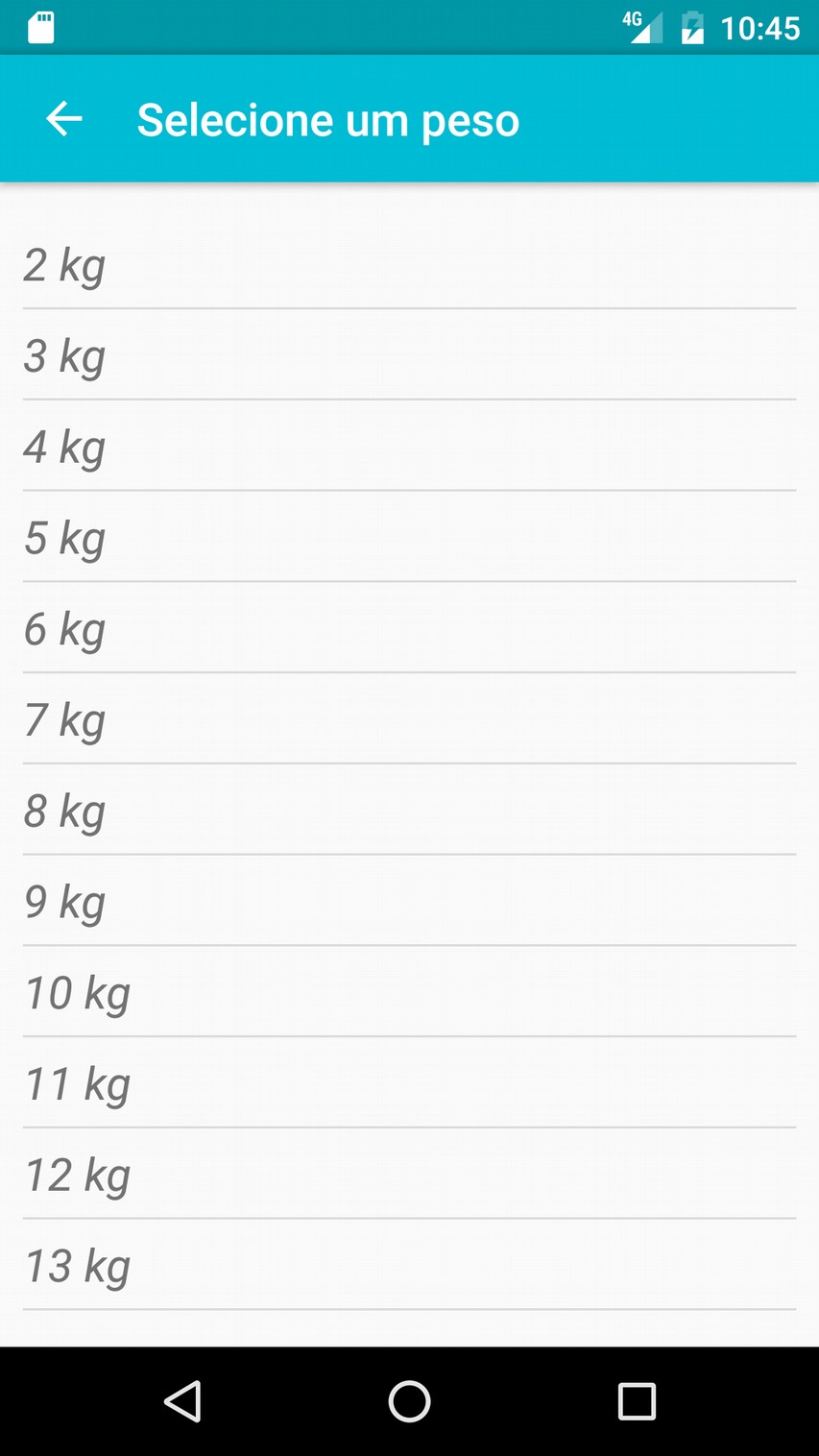
Ao selecionar a opção “Calculadoras” no menu inicial, o aplicativo exibe um menu onde é possível escolher dois tipos de calculadoras, conforme mostrado na figura 34. Escolhendo a opção “Corticóides” o usuário é direcionado para a tela de conversão de corticóide. (Figura 33)



*Figura 34: Menu de calculadoras Figura 33: Calculadora de Corticóide*

Conforme mostrado na figura 36, ao preencher todos os campos da tela de conversão de corticóides e selecionar a opção “Converter”, o aplicativo exibe o valor em uma caixa de diálogo.

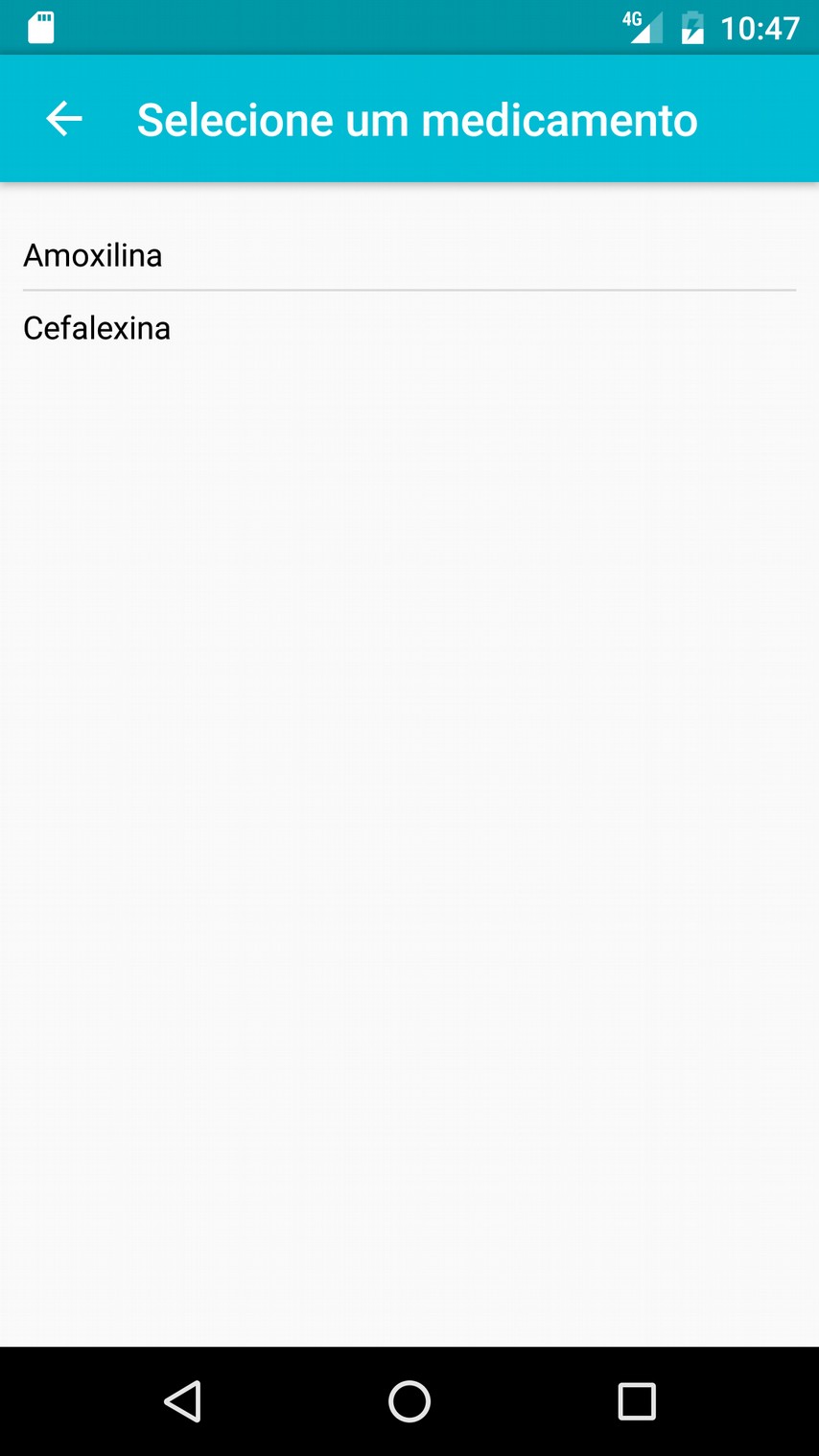
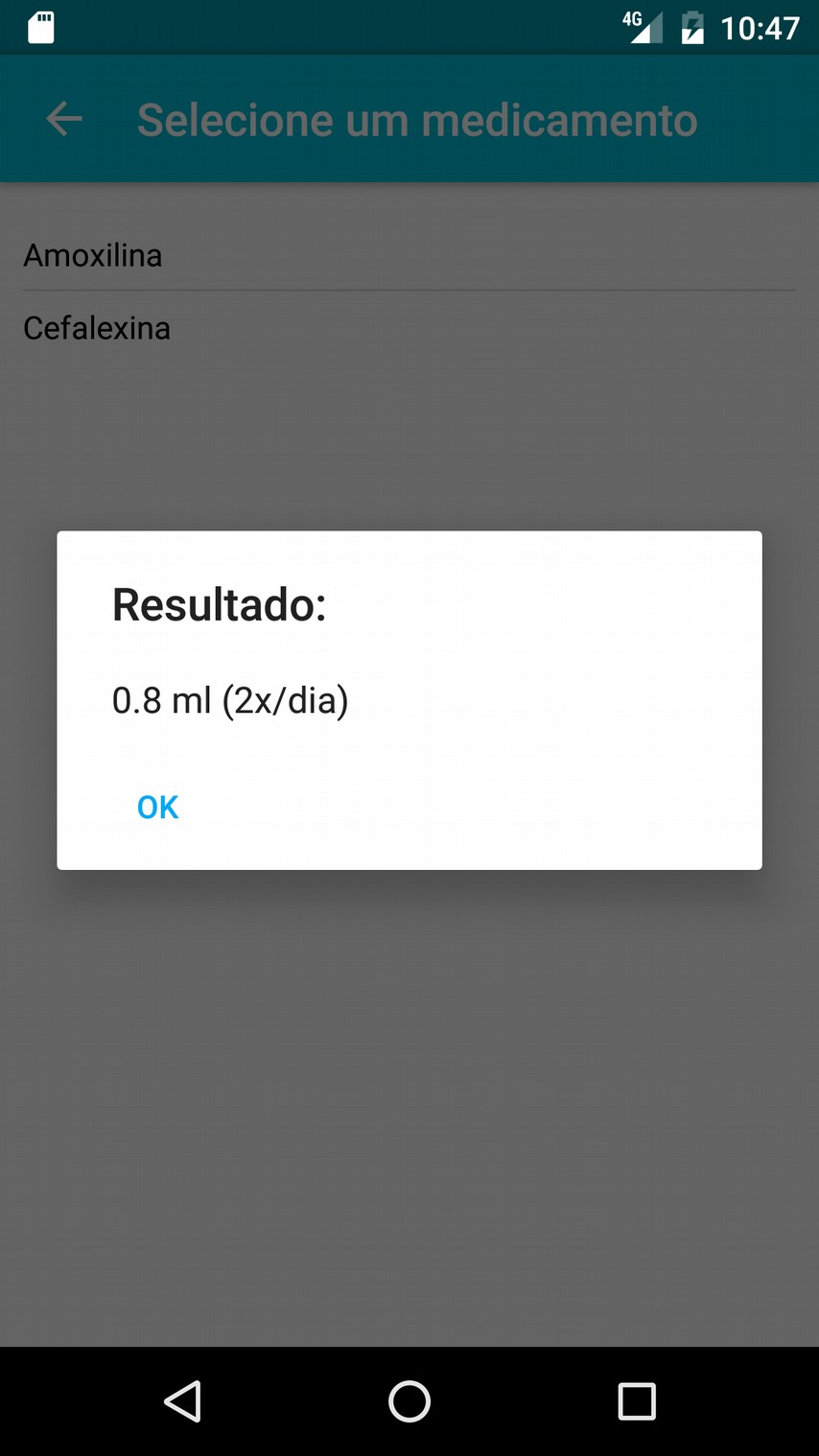
O Segundo tipo de calculadora disponível é a de antibióticos. Selecionando a opção “Antibióticos”, no menu de calculadoras, o usuário será direcionado, primeiramente, a uma tela onde selecionará um peso (Figura 35). Após escolhido o peso, será exibida uma lista de medicamentos (Figura 38). Selecionando-se um deles, é exibida uma caixa de diálogo com o resultado da operação, conforme exibido na figura 37.

*Figura 36: Calculadora de corticóide resultado*

*Figura 35: Calculadora de doses -*

*selecionar peso*

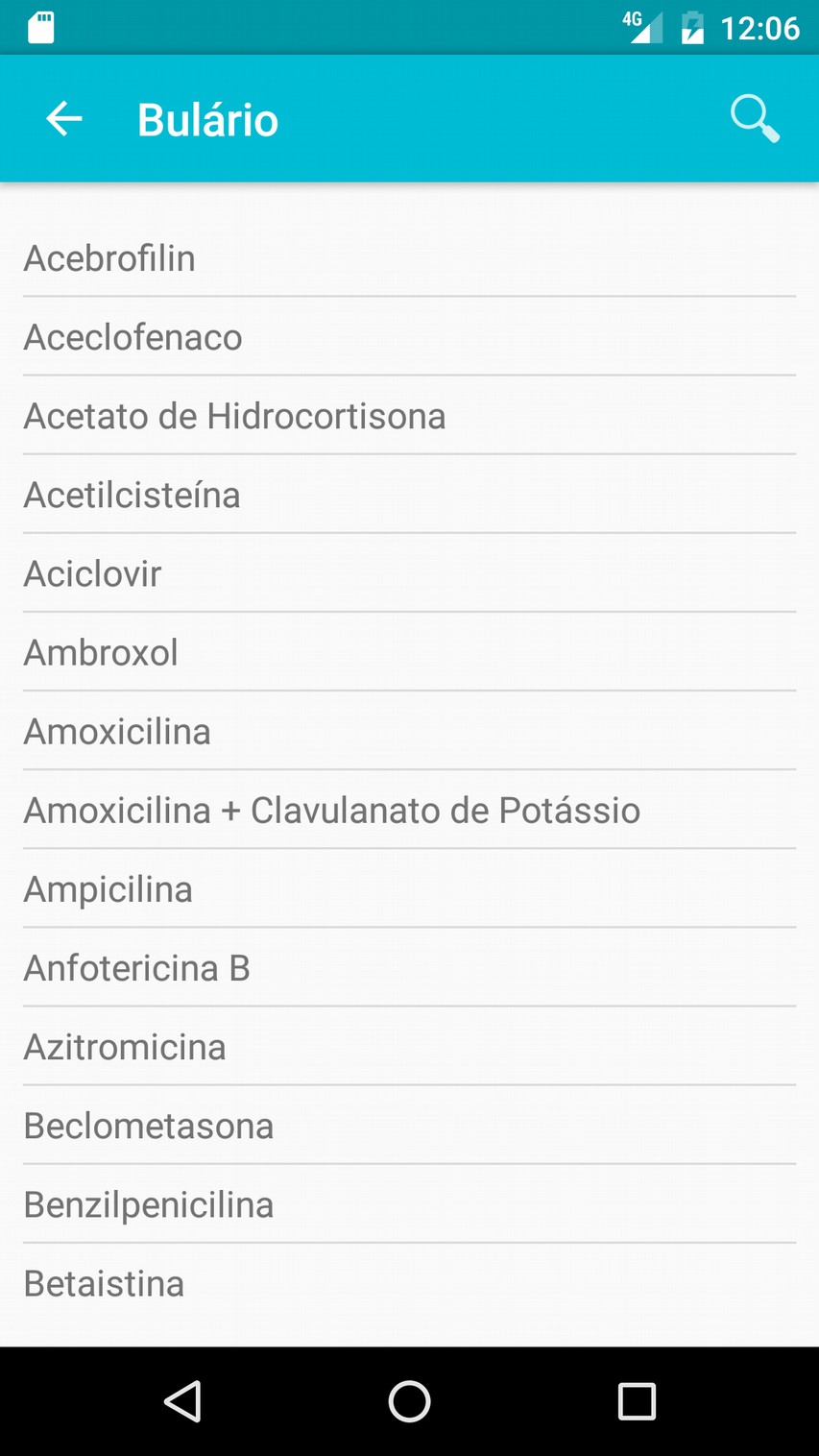
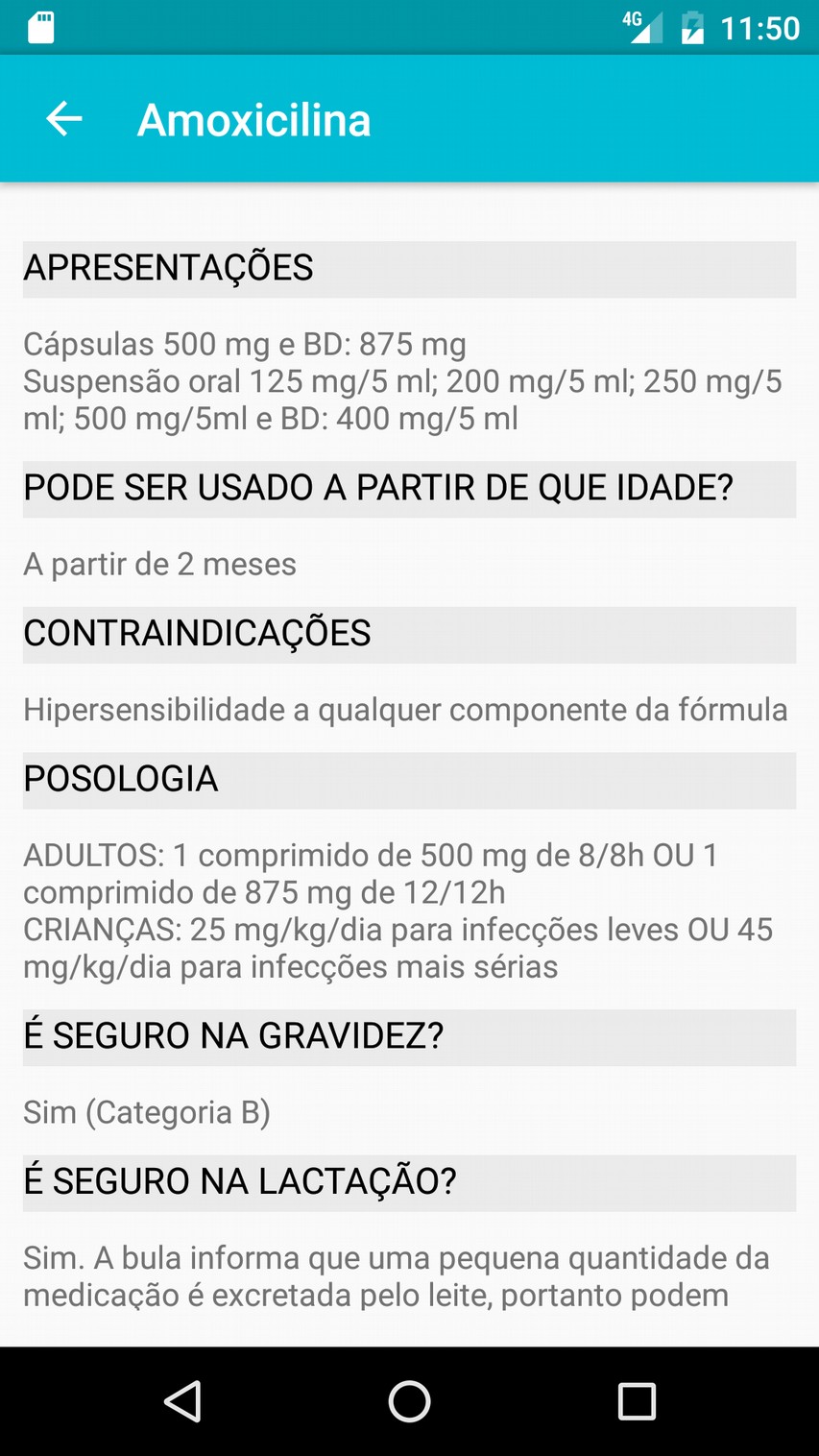
 

*Figura 38: Calculadora de doses - lista de medicamentos*

*Figura 37: Calculadora de doses - Resultado*

* + 1. Bulário

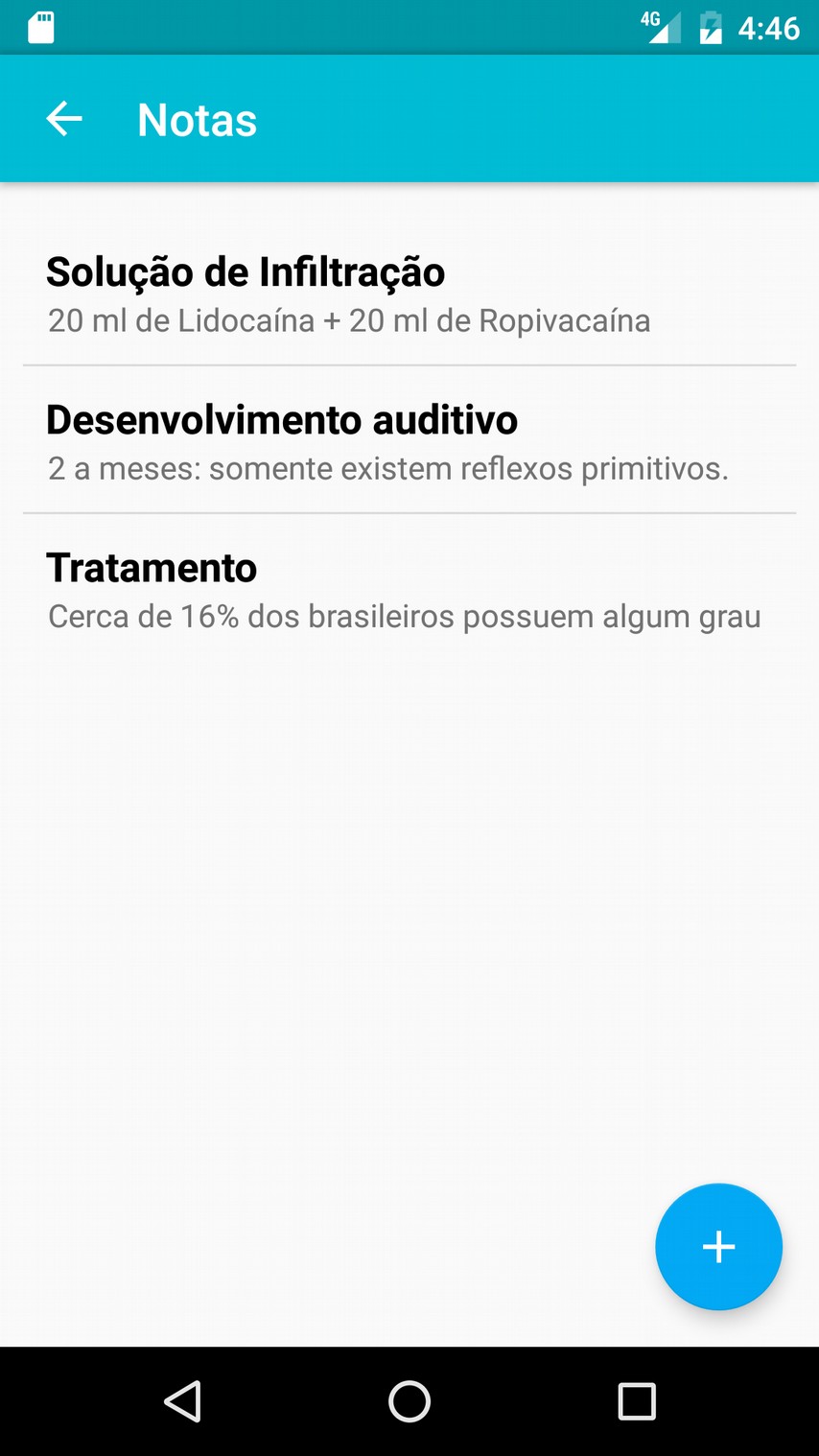
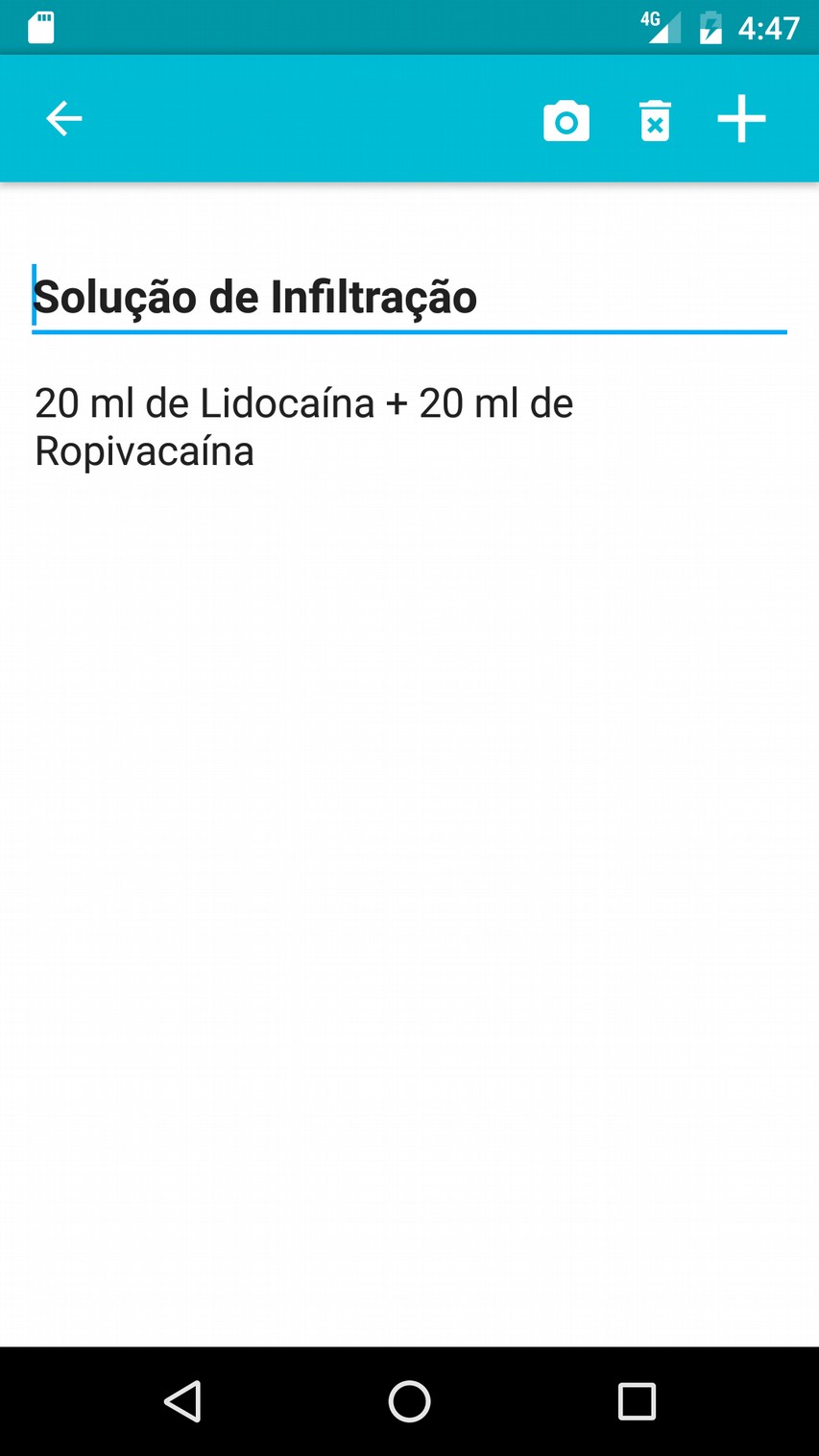
O fluxo do bulário se assemelha ao de diagnósticos. Ao selecionar a opção de menu “Bulário”, o usuário é direcionado a uma tela contendo uma lista de remédios (Figura 40). Ao realizar a seleção de um deles, este será apresentado na tela exibindo todas as informações mapeadas nos capítulos anteriores (Figura 39). Assim como na tela de diagnósticos, também é possível realizar um filtro para facilitar a busca de uma determinada medicação.

*Figura 40: Bulário Figura 39: Informações medicamento*

* + 1. Notas

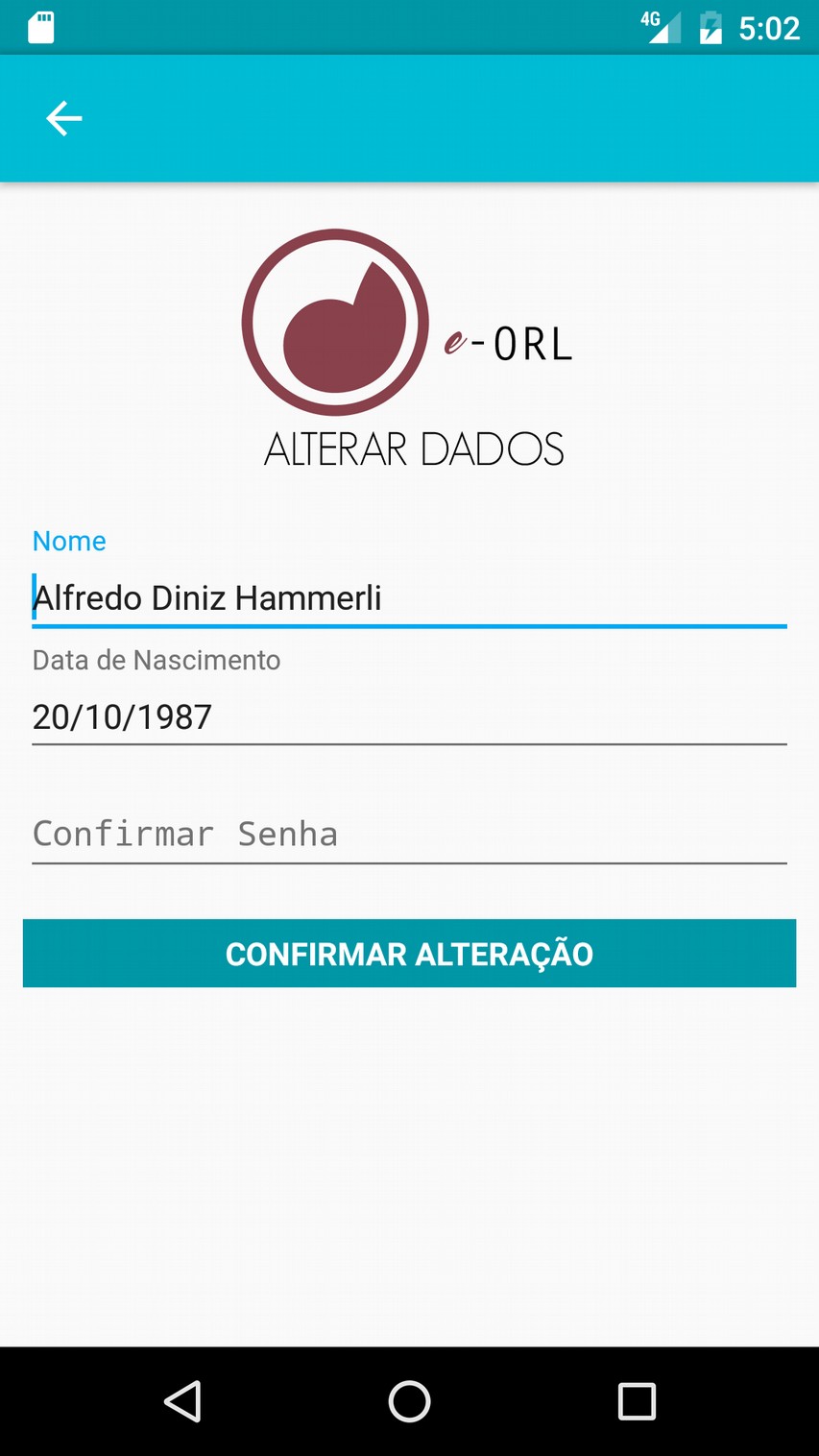
Selecionando a opção “Notas”, o usuário é direcionado para a tela de notas, que exibe em uma lista todas as anotações realizadas por ele (Figura 42). Ao selecionar uma delas, o aplicativo exibe o conteúdo detalhado de cada nota em uma outra tela, onde é possível também atualizar os dados ou remover a nota. (Figura 41). Selecionando o botão no canto inferior direito, é possível adicionar uma nova anotação que, depois de adicionada, será incluída na lista apresentada na tela anterior.

*Figura 42: Lista de notas do usuário Figura 41: Nota selecionada*

* + 1. Atualizar Dados Cadastrais

Na opção “Alterar Cadastro” do menu lateral (Figura 27), é possível alterar alguns dos dados cadastrais do usuário. Para que a alteração seja efetuada, é necessário informar a senha utilizada no aplicativo. Essa opção está disponível apenas para os usuários que efetuaram cadastro no aplicativo, realizando seu acesso de forma padrão, ou seja, utilizando login e senha cadastrados. Usuários que realizam seu acesso utilizando sua conta Google não possuem essa funcionalidade. (Figura 43)

*Figura 43: Tela de alteração de dados*

*cadastrais*

*Figura 44: Mensagem após atualização*

*dos dados cadastrais*

# CONCLUSÃO

Este trabalho apresentou a implementação de um aplicativo médico para a plataforma Android focando no desenvolvimento da solução. Além de todas as etapas de levantamento de requisitos e análise, apresentou-se algumas das ferramentas utilizadas para o desenvolvimento de um pequeno processo, no qual mapearam-se os requisitos baseados nas necessidades do usuário, modelaram-se os dados e estes foram implementados. A partir da implementação, em caso de falhas durantes os testes, foi apresentado um fluxo para cadastro e tratamento de bugs e demais falhas.

Durante todo o desenvolvimento desse projeto, pode-se destacar o documento de visão, abordado na seção dois, como um dos recursos mais importantes, servindo de base em todas as etapas. Devido principalmente à sua simplicidade e clareza, o entendimento do problema e das necessidades foi rápido. Na etapa de análise, o diagrama Molic foi de grande importância para o desenho das telas, além de servir de base para criação do fluxo associado a cada uma delas. Pode-se dizer que ele complementou a descrição de Caso de Uso,

agregando mais informações e ilustrando melhor os casos criados.

Como trabalhos futuros, pode-se citar melhorias na interface gráfica e na usabilidade, além da criação de duas novas funcionalidades, que foram comentadas e exibidas no capítulo quatro: F.A.Q e Anatomia Radiológica. O “F.A.Q.” aborda as principais dúvidas dos usuários com relação ao aplicativo, com respostas já prontas. Isso eliminaria a necessidade de entrar em contato com o suporte do programa para sanar dúvidas que são pertinentes a uma grande parte das pessoas que utilizam o programa. “Anatomia Radiológica” é uma sessão que visa a facilitar a relação do médico que utiliza o aplicativo com os exames de imagem presentes no seu dia a dia. Essa parte do aplicativo tem como proposta trazer imagens de raio-X, tomografia computadorizada, ressonância magnética e outros tipos de exames radiológicos presentes na prática otorrinolaringológica e fazer sua associação com as doenças descritas na sessão “Diagnósticos”.

Além de futuras seções, há também a proposta de desenvolvimento de uma versão do aplicativo para a plataforma iOS da Apple, utilizando a linguagem de programação Swift, que será de grande valor para o crescimento do aplicativo.

Ainda pensando no futuro da aplicação, mas tirando o foco do desenvolvimento, é importante a publicação do aplicativo na Play Store e a criação de políticas para divulgação da ferramenta, utilizando meios como redes sociais, sites médicos entre outros. O aplicativo não está disponível para download pelo público pois ainda está em construção de conteúdo por parte dos colaboradores.

Este aplicativo foi totalmente desenvolvido, tanto nas etapas de análise quanto nas de desenvolvimento, utilizando ferramentas gratuitas que possuem versões para os sistemas operacionais Windows, Mac OS e Linux.

# REFERÊNCIAS BIBLIOGRÁFICAS

ACADEMIA MÉDICA: 5 aplicativos aprovados pela FDA para o uso dos médicos. Disponível em: <[https://academiamedica.com.br/5-aplicativos-aprovados-pela-fda-](https://academiamedica.com.br/5-aplicativos-aprovados-pela-fda-para-o-uso-dos-medicos/) [para-o-uso-dos-medicos/](https://academiamedica.com.br/5-aplicativos-aprovados-pela-fda-para-o-uso-dos-medicos/)> Acesso em: 12/10/2016.

BARBOSA, Simone; SILVA, Bruno. **Interação Humano-Computador.** Rio de Janeiro: Elsevier, 2010.

BELL, Peter; BEER, Brent. **Introdução ao GitHub**. São Paulo: Novatec, 2014.

DEV MEDIA. Introdução ao Padrão MVC.

Disponível em <<http://www.devmedia.com.br/introducao-ao-padrao-mvc/29308>> Acesso em: 07/10/2016.

DEV MEDIA. Introdução aos bancos de dados NoSQL.

Disponível em <[http://www.devmedia.com.br/introducao-aos-bancos-de-dados-](http://www.devmedia.com.br/introducao-aos-bancos-de-dados-nosql/26044) [nosql/26044](http://www.devmedia.com.br/introducao-aos-bancos-de-dados-nosql/26044)> Acesso em: 07/10/2016.

DOOLEY, John F. **Software Development and Professional Practice**. 2 ed. Apress.

GUEDES, Gilleanes Thorwald Araujo. **UML 2: Uma Abordagem Prática**. 2 ed. São Paulo: Novatec, 2011.

LECHETA, Fernando. **Google Android**: **Aprenda a criar aplicações para dispositibos móveis com o Androis SDK**. 5 ed. São Paulo: Novatec, 2015.

LECHETA, Fernando. **Android Essencial**. São Paulo: Novatec, 2016.

PFIZER. Aplicativos aliados da saúde para dispositivos móveis.

Disponível em: <[http://www.pfizer.com.br/noticias/Aplicativos-aliados-da-sa](http://www.pfizer.com.br/noticias/Aplicativos-aliados-da-sa%C3%BAde-para-dispositivos-m%C3%B3veis)

[%C3%BAde-para-dispositivos-m%C3%B3veis](http://www.pfizer.com.br/noticias/Aplicativos-aliados-da-sa%C3%BAde-para-dispositivos-m%C3%B3veis)> Acesso em: 05/11/2016.

SOMMERVILLE, Ian. **Engenharia de Software**. 9 ed. São Paulo: Pearson Education, 2011.

TECMUNDO. 12 apps incríveis para quem trabalha com medicina. Disponível em:

<[http://www.tecmundo.com.br/medicina/28248-12-apps-incriveis-para-quem-](http://www.tecmundo.com.br/medicina/28248-12-apps-incriveis-para-quem-trabalha-com-medicina.htm) [trabalha-com-medicina.htm](http://www.tecmundo.com.br/medicina/28248-12-apps-incriveis-para-quem-trabalha-com-medicina.htm)> Acesso em: 12/10/2016.

TUDO CELULAR. Confira 5 vantagens e 5 desvantagens do sistema Android. Disponível em: <[http://www.tudocelular.com/google/noticias/n36150/confira-5-](http://www.tudocelular.com/google/noticias/n36150/confira-5-vantagens-e-desvantagens-do-android.html) [vantagens-e-desvantagens-do-android.html](http://www.tudocelular.com/google/noticias/n36150/confira-5-vantagens-e-desvantagens-do-android.html)> Acesso em: 12/10/2016.

UOL NOTÍCIAS. Apps de saúde vieram para ficar, e médicos precisam lidar com isso. Disponível em:<[http://noticias.uol.com.br/saude/ultimas-](http://noticias.uol.com.br/saude/ultimas-noticias/redacao/2016/01/22/apps-de-saude-vieram-para-ficar-medicos-precisam-lidar-com-isso.htm) [noticias/redacao/2016/01/22/apps-de-saude-vieram-para-ficar-medicos-precisam-](http://noticias.uol.com.br/saude/ultimas-noticias/redacao/2016/01/22/apps-de-saude-vieram-para-ficar-medicos-precisam-lidar-com-isso.htm) [lidar-com-isso.htm](http://noticias.uol.com.br/saude/ultimas-noticias/redacao/2016/01/22/apps-de-saude-vieram-para-ficar-medicos-precisam-lidar-com-isso.htm)> Acesso em: 12/10/2016.