

Proposta de Trabalho (Relatório)

MedsReminder

Mestrado em Engenharia Informática

**Disciplina:** Desenvolvimento de Aplicações Moveis

**Ano letivo:** 2019/2020

**Docente:** José Benjamim Ribeiro da Fonseca

**Autores**

João Assunção nº70021

João Ribeiro nº62054

Tiago Teixeira nº70034

Vila Real, 2020

Índice

[1. Introdução 3](#_Toc36197117)

[2. Objetivos da Etapa do Trabalho Prático 4](#_Toc36197118)

[3. Análise de Requisitos 5](#_Toc36197119)

[4. Casos de Uso 6](#_Toc36197120)

[5. Diagrama de Classes 13](#_Toc36197121)

[6. Diagrama de Entidade-Relacionamento 15](#_Toc36197122)

[7. Diagrama de Componentes 16](#_Toc36197123)

[8. Mockups 17](#_Toc36197124)

# Introdução

Devido à grande existência de consumo de medicamentos em Portugal, a grande dificuldade em tomar a medicação devidamente correta e fazer o controlo de cada pessoa, exige uma grande responsabilidade e esforço desnecessário por parte de uma entidade, que é muito difícil de arranjar. Nesse âmbito foi pensado em desenvolver uma aplicação mobile que faça toda essa gestão de forma automática e com simplicidade de uso.

O uso da aplicação pode vir a melhorar os cuidados de saúde a nível global e um melhor controlo dos seus medicamentos diários, em relação à gestão de stock, se já foi tomado, se a medicação está a fazer efeito, entre outros. No caso de o utilizador não estar a seguir as regras devidamente, isso pode ser consultado mais tarde por uma terceira pessoa ao aceder ao histórico da app. Devido à existência de um grande número de população idosa em Portugal e falta de experiência com as “novas tecnologias” a App pode perfeitamente ser programada para tal pessoa, ou uma entidade gerir um grupo de pessoas no mesmo smartphone. A App disponibiliza a criação de vários utilizadores no mesmo dispositivo.

No desenvolvimento deste trabalho irá ser utilizada uma App já existente chamada MyTherapy, essa app tem um grande número de funcionalidades disponíveis, irá ser implementado alguns desses serviços e adicionar mais alguns da nossa parte.

# Objetivos da Etapa do Trabalho Prático

Esta etapa do trabalho prático tem como principal objetivo apresentar a representação do sistema através da recolha de todos os requisitos, funcionais e não funcionais, necessários para a elaboração da aplicação.

Após a recolha de todos os requisitos, os requisitos funcionais foram utilizados para elaborar os diagramas de caso-de-uso da UML.

Vamos ainda representar o sistema através de um Diagrama de Classes, Diagrama de Entidade-Relacionamento, Diagrama de Componentes e apresentação das Mockups.

# MedsReminder

MedsReminder é o nome atribuído à nossa aplicação mobile, é uma App de gestão de medicamentos em que cabe ao utilizador introduzir a sua medicação ou algum outro tipo de produto para tratamento, para que seja notificado à hora correta que deve tomar/aplicar esse medicamento, assim evitando falhas e tornando o tratamento mais eficaz. Para além disso a App regista se o medicamento foi tomado criando um histórico, assim sendo mais fácil de controlar o estado de uma pessoa. A App usará calendário e o fuso horário para do telemóvel para que o utilizador tenha uma experiência em tempo real. Em caso do paciente não ter recursos mínimos para o usso da App, pode pedir a uma terceira pessoa, por exemplo uma assistente social, que tem várias pessoas ao seu cuido pode usar a App para fazer a gestão dos seus paciente, a App permite a criação de vários utilizadores.

Esta App é essencialmente importante para as pessoas que têm muita dificuldade em seguir o regime da sua medicação, ela avisa qual o medicamente a tomar e a sua hora. Como a App é direcionada para qualquer tipo de pessoa, o seu sistema de navegação irá ser o mais simples possível e com várias funções automáticos. Apenas exige uma configuração mais elaborada ao início, ao introduzir cada medicamento para cada utilizador, o resto da gestão será feita pelo aplicativo, como por exemplo o controlo de stock de cada medicamento.

Ela irá ter uma versão grátis e outra paga, com uma grande diferença em relação ao número de funcionalidades existentes na App.

# Análise de Requisitos

Requisitos Funcionais

RF1: O sistema deverá permitir introduzir um medicamento.

RF2: O sistema deverá permitir ativar as notificações/alarme do telemóvel.

RF3: O sistema deverá permitir ver a quantidade de eventos diários.

RF4: O sistema deverá permitir a criação de utilizadores/ vários utilizadores.

RF5: O sistema deverá permitir visualizar o histórico da ingestão dos medicamentos.

RF6: O sistema deverá permitir exportar os dados relativos as ingestões por medicamento.

RF7: O sistema deverá permitir editar medicamentos.

RF8: O sistema deverá permitir eliminar medicamentos introduzidos.

RF9: O sistema deverá permitir visualizar um menu de ajuda.

RF10: O sistema deverá permitir visualizar os medicamentos adicionados.

RF11: O sistema deverá permitir visualizar os utilizadores registados.

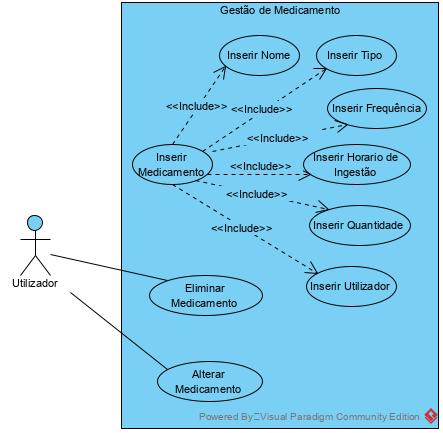
RF12: O sistema deverá permitir editar os utilizadores registados.

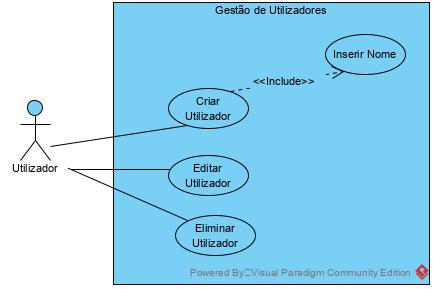
RF13: O sistema deverá permitir eliminar os utilizadores registados.

Requisitos Não-Funcionais

1. O sistema deverá comunicar com uma base de dados.
2. O sistema deverá ser implementado em linguagem java.

# Casos de Uso





|  |  |
| --- | --- |
| **Caso de Uso** | **Criar Medicamento** |
| **ID** | RF1 |
| **Atore(s)** | Utilizador |
| **Descrição** | Efetuar a criação do medicamento |
| **Cenário** | **Pré-condições:**   1. O sistema deve ter uma parte para inserir o nome do medicamento, o tipo, a frequência, a hora de ingestão, o utilizador que toma o medicamento e a quantidade 2. O utilizador deve encontrar-se na tab da Gestão/Secção do Medicamento |
| **Fluxo de Eventos:**   1. O utilizador preenche os campos necessários para criar um medicamento 2. O utilizador guarda os dados relativos ao medicamento. 3. As informações pertencentes a esses dados são enviadas para uma base de dados. |
| **Pós-condições:**   1. Após criar o medicamento, o utilizador recebe uma mensagem no ecrã a informar o medicamento foi inserido com sucesso. |

|  |  |
| --- | --- |
| **Caso de Uso** | **Ativar as notificações** |
| **ID** | RF2 |
| **Atore(s)** | Utilizador |
| **Descrição** | O Utilizador precisa de ativar as notificações para que possa receber notificações. |
| **Cenário** | **Pré-condições:**   1. O sistema deve pedir ao utilizador que ative as notificações. 2. O utilizador deve iniciar a aplicação a primeira vez ou então ir as definições da aplicação. |
| **Fluxo de Eventos:**   1. O utilizador aceita que a aplicação lhe envie notificações. |
| **Pós-condições:**   1. Após aceitar, o utilizador está pronto para receber notificações. |
| **Caso de Uso** | **Ver a quantidade de eventos diários** |
| **ID** | RF3 |
| **Atore(s)** | Utilizador |
| **Descrição** | O Utlizado deve poder visualizar a quantidade de eventos diários. |
| **Cenário** | **Pré-condições:**   1. O Utilizar deve-se encontrar-se na tab Eventos. |
| **Fluxo de Eventos:**   1. O Utilizador seleciona a tab “Eventos”. 2. As bases de dados que contem a informação sobre os Medicamentos e a data é consultada 3. A informação é devolvida ao Utilizador. |
|  |

|  |  |
| --- | --- |
| **Caso de Uso** | **Criar Utilizador** |
| **ID** | RF4 |
| **Atore(s)** | Utilizador |
| **Descrição** | Efetuar a inserção de um Utilizador |
| **Cenário** | **Pré-condições:**   1. O sistema deve ter uma parte para inserir o nome do Utilizador. 2. O utilizador deve encontrar-se na tab da Gestão/ Secção do Utilizador. |
| **Fluxo de Eventos:**   1. O utilizador preenche o campo do nome de utilizador 2. O utilizador guarda os dados relativos ao Utilizador. 3. As informações pertencentes a esses dados são enviadas para uma base de dados. |
| **Pós-condições:**   1. Após criar o utilizador, o utilizador recebe uma mensagem no ecrã a informar o u foi inserido com sucesso. |

|  |  |
| --- | --- |
| **Caso de Uso** | **Ver a quantidade de eventos diários** |
| **ID** | RF5 |
| **Atore(s)** | Utilizador |
| **Descrição** | O Utlizado pode visualizar o Histórico da ingestão dos Medicamentos. |
| **Cenário** | **Pré-condições:**   1. O Utilizar deve-se encontrar-se na tab “Histórico”. |
| **Fluxo de Eventos:**   1. O Utilizador seleciona a tab “Histórico”. 2. As bases de dados que contem a informação sobre os Medicamentos e a data é consultada 3. A informação sobre o Histórico é devolvida ao Utilizador. |
|  |

|  |  |
| --- | --- |
| **Caso de Uso** | **Exportar os dados relativos as ingestões por medicamento** |
| **ID** | RF6 |
| **Atore(s)** | Utilizador |
| **Descrição** | O Utilizador pode exportar os dados relativos a ingestão dos medicamentos, os dados podem ser exportados para PDF |
| **Cenário** | **Pré-condições:**  1. O sistema deve permitir compilar a informação da base de dados para PDF   1. O utilizador deve encontrar-se na tab “Histórico” |
| **Fluxo de Eventos:**   1. O utilizador clica no botão “PDF” |
| **Pós-condições:**   1. O sistema permite ao utilizador descarregar o ficheiro |

|  |  |
| --- | --- |
| **Caso de Uso** | **Editar Medicamento** |
| **ID** | RF7 |
| **Atore(s)** | Utilizador |
| **Descrição** | Efetuar a edição do medicamento |
| **Cenário** | **Pré-condições:**  1. O sistema deve ter uma parte para editar o nome do medicamento, o tipo, a frequência, a hora de ingestão, o utilizador que toma o medicamento e a quantidade   1. O utilizador deve encontrar-se na tab da Gestão-> Clicar no medicamento que deseja editar |
| **Fluxo de Eventos:**   1. O utilizador preenche os campos que pretende alterar 2. O utilizador guarda os dados. 3. As informações pertencentes a esses dados alterados são enviadas para uma base de dados. |
| **Pós-condições:**   1. Após editar o medicamento, o utilizador recebe uma mensagem no ecrã a informar o medicamento foi editado com sucesso. |

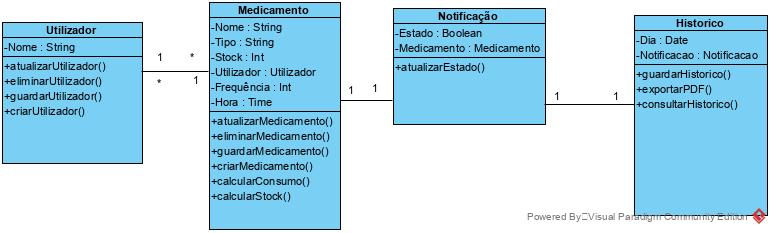
|  |  |
| --- | --- |
| **Caso de Uso** | **Eliminar Medicamento** |
| **ID** | RF8 |
| **Atore(s)** | Utilizador |
| **Descrição** | Apagar o medicamento |
| **Cenário** | **Pré-condições:**   1. O utilizador deve encontrar-se na tab da Gestão-> Clicar no medicamento pretendido. |
| **Fluxo de Eventos:**   1. O utilizador seleciona o medicamento pretendido da lista. 2. O utilizador clica no símbolo para eliminar. 3. O utilizador confirma essa ação. |
| **Pós-condições:**   1. Após eliminar o medicamento, recebe uma mensagem a dizer que foi apagado com sucesso. |
| **Caso de Uso** | **Visualizar os Medicamentos Adicionados** |
| **ID** | RF10 |
| **Atore(s)** | Utilizador |
| **Descrição** | O Utlizado pode visualizar os Medicamentos Adicionados |
| **Cenário** | **Pré-condições:**   1. O Utilizar deve-se encontrar-se na tab “Gestão” e encontrar-se na secção Medicamento. |
| **Fluxo de Eventos:**   1. O Utilizador seleciona a tab “Gestão”. 2. As bases de dados que contem a informação sobre os Medicamentos e a data é consultada |
| **Pós-condições:**  A informação é compilada sobre a forma de lista |

|  |  |
| --- | --- |
| **Caso de Uso** | **Visualizar os Utilizadores Adicionados** |
| **ID** | RF11 |
| **Atore(s)** | Utilizador |
| **Descrição** | O Utlizado pode visualizar os Utilizadores Adicionados |
| **Cenário** | **Pré-condições:**   1. O Utilizar deve-se encontrar-se na tab “Gestão” e encontrar-se na secção Utilizador. |
| **Fluxo de Eventos:**   1. O Utilizador seleciona a tab “Gestão”. 2. As bases de dados que contem a informação sobre os Medicamentos e a data é consultada |
| **Pós-condições:**  A informação é compilada sobre a forma de lista |

|  |  |
| --- | --- |
| **Caso de Uso** | **Editar Utilizador** |
| **ID** | RF12 |
| **Atore(s)** | Utilizador |
| **Descrição** | Efetuar a edição do utilizador |
| **Cenário** | **Pré-condições:**  1. O sistema deve ter uma parte para editar o nome do utilizador   1. O utilizador deve encontrar-se na tab da Gestão-> Clicar no utilizador que deseja editar |
| **Fluxo de Eventos:**   1. O utilizador preenche os campos que pretende alterar 2. O utilizador guarda os dados. 3. As informações pertencentes a esses dados alterados são enviadas para uma base de dados. |
| **Pós-condições:**   1. Após editar o utilizador, o utilizador recebe uma mensagem no ecrã a informar o utilizador foi editado com sucesso. |

|  |  |
| --- | --- |
| **Caso de Uso** | **Eliminar Utilizador** |
| **ID** | RF13 |
| **Atore(s)** | Utilizador |
| **Descrição** | Apagar o Utilizador |
| **Cenário** | **Pré-condições:**   1. O utilizador deve encontrar-se na tab da Gestão-> Clicar no utilizador pretendido. |
| **Fluxo de Eventos:**   1. O utilizador seleciona o utilizador pretendido da lista. 2. O utilizador clica no símbolo para eliminar. 3. O utilizador confirma essa ação. |
| **Pós-condições:**   1. Após eliminar o utilizador, recebe uma mensagem a dizer que foi apagado com sucesso. |

# Diagrama de Classes



**Utilizador**

Diagrama de classes é uma representação da estrutura e relações das classes que servem de modelo para objetos. Neste caso iremos ter quatro classes diferentes todos têm uma ligação entre si. No caso da classe utilizador, poderá ser possível criar um utilizador através a inserção de um nome que poderá ser alterado a qualquer altura. O utilizador criado vai estar relacionado/associado a vários medicamentos, sendo assim uma relação ,de um para muitos, com a classe medicamentos. Assim também como os medicamentos têm uma relação de um para muitos, quer dizer que 1 medicamente pode ser atribuído a vários utilizadores.

**Medicamento**

Na classe medicamento teremos vários atributos, como está representado no diagrama. Especificação do nome, o nome pode ser escolhido através de uma lista de medicamentos implementada ou então se não existir pode ser adicionado um nome pelo utilizador. O tipo é a indicar se o medicamento é de pilula, creme, supositório, etc. Stock é a quantidade de medicamentos que o utilizador possui, de seguida a frequência e hora que vai determinar o horário em que tem que tomar/aplicar o medicamento. Esta classe tem as operações de edição dos campos do medicamento, cálculo do stock e o calculo que determina a hora aplicação de determinado medicamento. Cada medicamento será associado a um ou vários utilizadores. Esta classe irá ser relacionada com as Notificações que irá ser explicado de seguida.

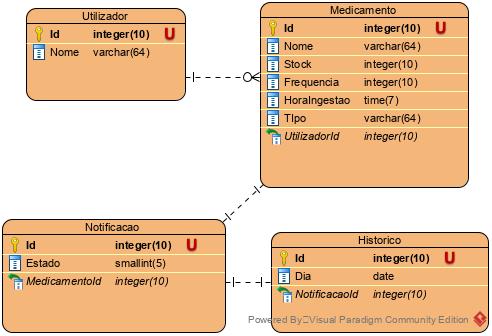
**Notificação**

Esta classe tem uma única função que será verificar se o utilizador tomou o comprimido às horas definidas pela App através de um atributo booleano, ou seja, verdadeiro ou falso. Como indica a relação, cada medicamento irá ter esta função. Ao utilizador responder à notificação enviada e clicar confirmar, esta classe vai guardar que o medicamento foi tomado, ou caso seja ignorada a notificação a App irá assumir que o utilizador não tomou o comprimido. Esta classe está relacionada com o histórico.

**Histórico**

A classe histórico irá ter como atributos o dia em que existem notificações e a validação feita anteriormente. Ela irá guardar esses dados na app e apresenta-lo ao utilizador de forma a ver o calendário com os dias em que tomou a medicação ou então os dias que falhou em tomá-los. Esta classe também tem uma função em que será possível imprimir esse calendário para um ficheiro PDF. A relação está definida de 1 para 1 porque cada notificação gere apenas um dado a ser guardado no histórico.

# Diagrama de Entidade-Relacionamento

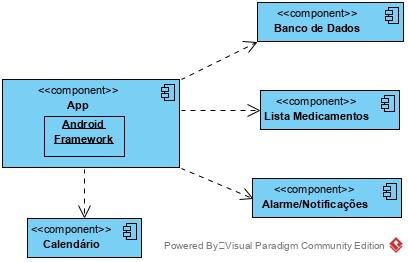


O Diagrama de Entidade-Relacionamento deve permitir armazenar todos os dados necessários ao funcionamento da aplicação. Para isso vão ser criadas uma série de tabelas e estabelecidas relações entre elas de modo a descrever todos os dados.

Neste mesmo diagrama de Entidade-Relacionamento a tabela “Utilizador” tem o parâmetro nome só, o cliente da aplicação precisa de inserir os nomes de utilizador antes de inserir um medicamento, visto que muitas vezes o cliente faz a gestão de mais do que um utilizador. A tabela “Medicamento” é composta pelos seguintes componentes: Nome, Stock (Quantidade Estimada), Frequência (Frequência de Ingestão), HoraIngestão, o Tipo (Tipo de Medicamento) e uma lista de utilizadores, que o cliente inseriu anteriormente. A tabela “Notificacao” tem uma conexão com a tabela “Medicamento”, visto que esta precisa de fazer a gestão das notificações para isso guardara o estado associado ao medicamento, e faz uma conexão com a tabela “Histórico” para fazer associação ao dia.

A tabela “Histórico” tem o registo dos dias da semana assim como o estado para cada dia.

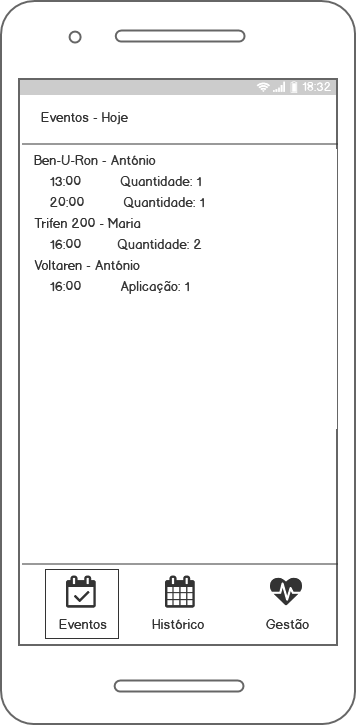
# Diagrama de Componentes



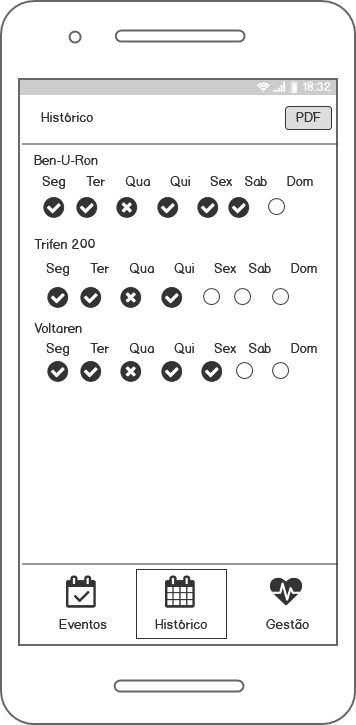
Em relação ao diagrama de componentes que representa o a forma como cada componente trabalha e as suas relações. Temos a Android Framework, onde está presente todo o código usado para o desenvolvimento da App. A esse código será adicionado uma base de dados, que neste caso é local, em que vai guardar os utilizadores introduzidos na App e novos medicamentos, não presente na lista. A lista com os medicamentos será retirada de uma base de dados disponibilizada, é uma lista oficial de praticamente todos os medicamentos existentes em Portugal. Essa lista irá ser implementada na App para que o utilizador posso escolher o seu medicamento de forma rápida. Da parte do smartphone irá ser usado o seu alarme para que possa avisar de novas notificações. E por fim um calendário para que a app tenha um sentido de orientação com base do tempo real em que o utilizador está presente.

# Mockups

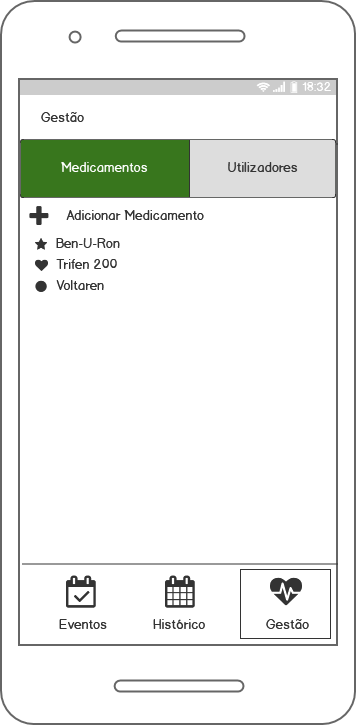
View Eventos



View Histórico

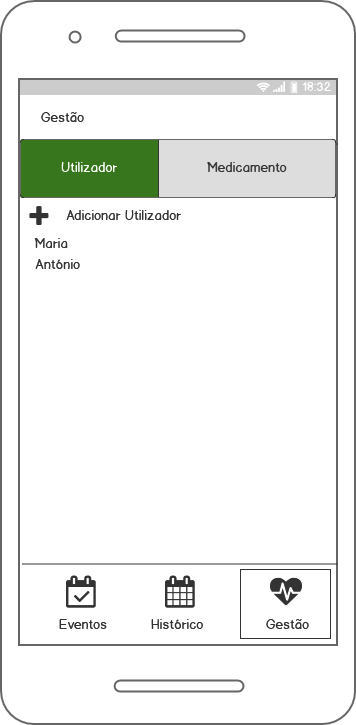


View Lista Medicamentos



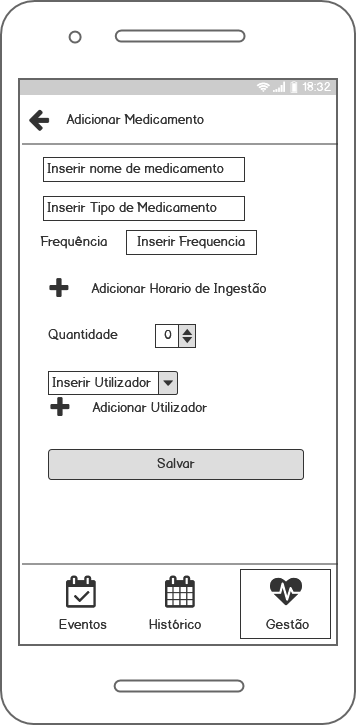
Ajuda

View Lista Utilizadores

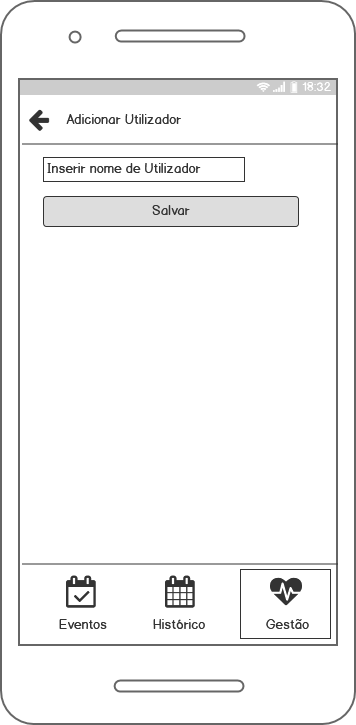


Ajuda

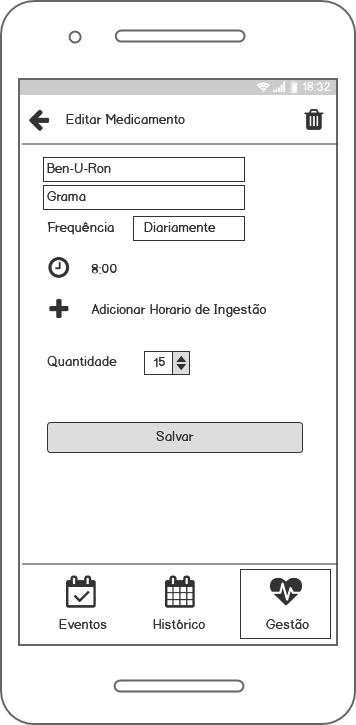
View Adicionar Medicamento



View Adicionar Utilizador



View Editar Medicamento



View Editar Utilizador

