**UNIVERSIDADE DA BEIRA INTERIOR**

Faculdade de Engenharia | Departamento de Informática

****

**Engenharia de Software**

**Badge7**

**Alunos:**

**Alexandre Matos, a44187**

**Bernardo Louro, a43271**

**João Maio, a43710**

**Índice**

[Introdução 2](#_Toc2805)

[Glossário 3](#_Toc27507)

[Requisitos de Utilizador 4](#_Toc27928)

[Requisitos Não Funcionais 4](#_Toc4992)

[Requisitos Funcionais 4](#_Toc8240)

[Requisitos de Sistema 5](#_Toc13463)

[Requisitos Não funcionais 5](#_Toc22897)

[Requisitos Funcionais 11](#_Toc20581)

[Arquitetura de Sistema 19](#_Toc22513)

[Wireframes 20](#_Toc25922)

[Mockups 25](#_Toc21980)

# Introdução

O processo de desenvolvimento que será usado neste projeto e o desenvolvimento incremental, pois trata-se de um software que a qualquer momento pode precisar de alterações. Optando por este modelo, temos a possibilidade de adaptação face às exigências que podem aparecer no futuro e a implementação de novas funcionalidades ser um processo mais fácil. Vamos também utilizar o método ágil, pois ele permite-nos, mais uma vez, evoluir o software e entregar atualizações do produto de forma mais rápida e eficaz e evitar certas limitações de outros processos. Sabemos que um dos grandes problemas do desenvolvimento incremental é o processo de reestruturação, logo vamos tomar mais atenção a esse processo para evitar que a estrutura da aplicação não seja corrompida pelas mudanças constantes.

# Glossário

# Requisitos de Utilizador

## Requisitos Não Funcionais

1-O sistema deve demorar menos de 1 segundo a responder aos inputs do utilizador.

2-O sistema deve manter os dados seguros.

3-Um sistema deve suportar todos os utilizadores ao mesmo tempo.

4-O sistema deve estar disponível 99.9% do tempo e informar quando está indisponível.

5-O sistema deve assegurar que os dados são enviados em segurança para o backoffice.

6-O sistema deve ter uma interface simples, intuitiva e eficiente.

## Requisitos Funcionais

1-O sistema deve ter uma opção de login para o funcionário.

2-O sistema deve conseguir realizar o registo de novos consumidores.

3-O sistema deve conseguir adicionar/alterar/remover contadores de consumidores.

4-O sistema deve apresentar um menu para os Avisos e Alertas.

5-O sistema deve possibilitar a criação de relatórios ao final de cada dia.

6-O sistema deve possibilitar a escrita de Notas/Observações sobre os consumidores/contadores.

7-O sistema deve permitir ao utilizador atualizar os dados automaticamente e manualmente.

8-O sistema deve permitir ao utilizador a criação de ocorrências.

# Requisitos de Sistema

## Requisitos Não funcionais

1.

1.1. O sistema deve ser capaz de responder de forma rápida e eficiente aos inputs do utilizador.

1.2. O sistema deve ser construído de forma a que tarefas maiores sejam divididas em tarefas menores, que demorem menos de 100ms para serem executadas.

|  |
| --- |
| Definição da função ou entidade:  O sistema deve demorar menos de 1 segundo a responder a inputs do utilizador. |
| Descrição dos inputs e de onde eles vêm:  Qualquer input, de qualquer parte do sistema. |
| Descrição dos outputs e para onde vão:  Apresentação da página e responsividade do sistema->ecrã. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Tempo de execução. |
| Descrição da ação a ser feita:  Todas as tarefas devem ser divididas em tarefas mais pequenas que levam menos de 100ms para serem executadas. |
| Condições Pré e Pós:  Tarefas grandes.  Tarefas pequenas. |
| Efeitos secundários: |

2.

2.1 O sistema deve guardar todos os dados numa base de dados gerida pelo SQL SERVER que será acedida apenas pelo administrador do sistema.

2.2 O sistema deve suportar a encriptação de dados.

|  |
| --- |
| Definição da função ou entidade:  O sistema deve manter os dados seguros. |
| Descrição dos inputs e de onde eles vêm:  Dados de login.  Dados de clientes. |
| Descrição dos outputs e para onde vão:  Dados encriptados.  Dados a serem guardados->base de dados. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Login. |
| Descrição da ação a ser feita:  Os dados vão ser guardados na base de dados gerenciada pelo SQL Server que apenas administradores podem aceder. |
| Condições Pré e Pós:  Informação não está encriptada.  Informação está encriptada. |
| Efeitos secundários:  Se uma tentativa de aceder aos dados é feita por alguém que não está autorizado, os administradores receberão um alerta. |

3.

3.1. O sistema deve ser implementado de forma a ter uma “Escalabilidade Vertical”, de modo a permitir, se necessário, suportar mais utilizadores.

3.2. Deve existir uma boa documentação para servir de suporte para futuras atualizações.

3.3. Baseado nos relatórios de manutenção será analisado se mais atualizações são necessárias.

|  |
| --- |
| Definição da função ou entidade:  O sistema deve aguentar com todos os usuários ao mesmo tempo. |
| Descrição dos inputs e de onde eles vêm:  Documentação.  Relatórios de manutenção. |
| Descrição dos outputs e para onde vão:  Atualizações.  Documentação. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Nenhuma |
| Descrição da ação a ser feita:  Se necessário, deve atualizar a sua infraestrutura para dar rápido acesso e eficiência ao sistema. |
| Condições Pré e Pós:  Nenhum |
| Efeitos secundários:  Nenhum |

4.

4.1. No caso do sistema está indisponível, para manutenção ou por outras razões, o sistema teve avisar o utilizador na página inicial no campo de Avisos/Alertas.

4.2. A manutenção é feita quinzenalmente fora dos horários de trabalho dos funcionários, ficando o sistema indisponível para receber atualizações. O tempo que o sistema fica indisponível não deverá ser superior a 1h e 10 minutos.

4.3. Se for necessária uma manutenção urgente o sistema não deverá ficar indisponível por mais do que 45 minutos.

4.4 Todas as manutenções urgentes devem ser guardadas num relatório que será analisado e que servirá de suporte para futuras atualizações e manutenções.

|  |
| --- |
| Definição da função ou entidade:  O sistema deve informar os usuários caso o sistema esteja indisponível. |
| Descrição dos inputs e de onde eles vêm: |
| Descrição dos outputs e para onde vão:  Mensagem de alerta. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Identificar o que deve ser arranjado. |
| Descrição da ação a ser feita:  O sistema deve informar os usuários que o sistema está indisponível mostrando uma pequena caixa de texto na homepage. |
| Condições Pré e Pós:  Sistema com bugs ou falhas ou em manutenção.  Sistema atualizado e a funcionar propriamente. |
| Efeitos secundários:  A ocorrência de manutenção urgente deve ser registada para ser analisada no futuro. |

5.

5.1. O sistema deve ter acesso à rede e ter um protocolo de confirmação que garantirá que os dados irão para o servidor em segurança.

5.2. Caso não exista rede o sistema deverá notificar o utilizador de que os dados não puderam ser enviados para o servidor e serão só guardados internamente até que seja possível enviar.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai manter os dados em segurança. |
| Descrição dos inputs e de onde eles vêm:  Dados que vão ser enviados. |
| Descrição dos outputs e para onde vão:  Envio dos dados para o backoffice. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Dados dos consumidores da base de dados. |
| Descrição da ação a ser feita:  Manter os dados seguros. |
| Condições Pré e Pós:  Dados para serem enviados/atualizados. |
| Efeitos secundários:  Senão existir rede o sistema deverá notificar o utilizador de que os dados são guardados internamente. |

6.

6.1. A interface do sistema deve ser intuitiva o bastante, para que os funcionários consigam a fim de 1 hora de workshop, utilizar a aplicação.

6.2. A interface deve ter todos os seus campos com pequenas e explicativas explicações do que quer ou faz.

6.3. A interface deve ter um aspeto visual minimalista e com uma palete de cores que combine.

## Requisitos Funcionais

1.

1.1. O sistema deve apresentar duas caixas de inputs, uma para o Nome de Utilizador e outra para a Password

1.2. Se o login for verificado, o sistema dará acesso ao staff a todas as funcionalidades do sistema.

1.3. Se o login falhar 3 vezes, o sistema deverá avisar os administradores e a conta irá bloquear.

1.4. A conta só irá ser desbloqueada por um administrador.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário fazer login. |
| Descrição dos inputs e de onde eles vêm:  Username e password->base de dados. |
| Descrição dos outputs e para onde vão:  Mensagem a dizer que o login foi bem sucedido. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Dados de login do funcionário da base de dados. |
| Descrição da ação a ser feita:  Login do funcionário no sistema. |
| Condições Pré e Pós:  Staff registado. |
| Efeitos secundários:  Se o login falhar 3 vezes a conta será bloqueada. |

2.

2.1. O utilizador irá escolher a opção “Registar Cliente”

2.2. O sistema deverá mostrar uma página com os campos: “Nome cliente”, “Idade”, “Morada”, “Nº do contador” e dois botões “Submeter” e “Voltar”.

2.3. Caso o utilizador tenha preenchido os campos “Nome cliente”, “Morada” e “Nº de contador” irá ser possível clicar no botão “Submeter”.

2.4. Caso o utilizador queira voltar para o menu anterior e cancelar o registo do novo cliente irá clicar no botão “Voltar”.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário registar um cliente. |
| Descrição dos inputs e de onde eles vêm:  nome cliente, idade, morada e nº do contador->Input do funcionário. |
| Descrição dos outputs e para onde vão:  Dados do cliente->base de dados.  Mensagem a dizer que o registo foi bem sucedido. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Dados de registo dos clientes de input do funcionário. |
| Descrição da ação a ser feita:  Registar um cliente no sistema. |
| Condições Pré e Pós:  Cliente não registado. |
| Efeitos secundários:  Nenhum. |

3.

3.1. O utilizador irá escolher a opção “Gerir Consumidores”

3.2. O sistema deverá mostrar uma página com um campo de pesquisa, onde será escrito um número de um contador e um botão “Voltar”, que caso seja clicado irá voltar para o menu anterior.

3.3 Caso não exista um contador com esse número irá aparecer uma mensagem de erro: “Contador não encontrado”.

3.4 Caso exista um consumidor com esse número de contador associado irá aparecer uma opção para selecionar esse consumidor.

3.5 Se o utilizador selecionar essa opção, o sistema apresentará uma página com todos os contadores associados a esse consumidor e com as opções de “Alterar”, “Remover” e “Adicionar novo contador”.

3.6 Caso o utilizador escolha “Alterar”, irá aparecer toda a informação associada a esse contador e o utilizador irá puder mudar qualquer uma dessas informações. No final será necessário clicar num botão “Guardar” para as alterações ficarem concluídas.

3.7 Caso o utilizador escolha “Remover”, o contador irá ser removido do consumidor.

3.8 Caso o utilizador escolha “Adicionar novo contador”, irá aparecer campos para adicionar a informação sobre contador e um botão “Guardar” para a operação ficar concluída.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário adicionar contadores de consumidores. |
| Descrição dos inputs e de onde eles vêm:  Nº contador->input do funcionário. |
| Descrição dos outputs e para onde vão:  Alterações nos contadores no cliente associado->base de dados.  Mensagem a dizer que as alterações foram bem sucedidas. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Dados dos clientes e dos contadores associados. |
| Descrição da ação a ser feita:  Adicionar contadores de consumidores. |
| Condições Pré e Pós:  Necessárias alterações para algum consumidor. |
| Efeitos secundários:  Nenhum. |

|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário alterar/remover contadores de consumidores. |
| Descrição dos inputs e de onde eles vêm:  Nº contador->input do funcionário. |
| Descrição dos outputs e para onde vão:  Alterações nos contadores no cliente associado->base de dados.  Mensagem a dizer que as alterações foram bem sucedidas. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Dados dos clientes e dos contadores associados. |
| Descrição da ação a ser feita:  Alterar/remover contadores de consumidores. |
| Condições Pré e Pós:  Necessárias alterações para algum consumidor. |
| Efeitos secundários:  Nenhum. |

4.

4.1. O sistema deve apresentar, na homepage, 1 caixa com todos os avisos/alertas que existem naquele momento.

4.2. O sistema deve poder diferenciar a prioridade desses avisos/alertas por cores das notificações deles, sendo o vermelho para Alertas/avisos que precisam de ser corrigidos o mais rápido possível (máxima prioridade) e amarelo para os de menor prioridade.

4.3. O sistema deve não só permitir os administradores enviarem alertas/avisos para os funcionários, mas também o contrário.

4.4. O sistema para além de representar visualmente esses avisos na homepage, também deverá enviar para o email de cada funcionário o aviso/alerta correspondente, podendo assim cada funcionário verificá-los não só na aplicação, mas também no email.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai mostrar avisos na homepage. |
| Descrição dos inputs e de onde eles vêm:  Avisos->input de um administrador ou funcionário. |
| Descrição dos outputs e para onde vão:  Aviso->homepage |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Aviso escrito pelo administrador ou funcionário. |
| Descrição da ação a ser feita:  Mostrar avisos na homepage. |
| Condições Pré e Pós:  Ocorrer algum problema. |
| Efeitos secundários:  Nenhum. |

5.

5.1. O utilizador irá escolher a opção “Relatório de atividades”.

5.2. O sistema deve apresentar uma página com os campos: “Número de contadores/clientes validados”, “Número de ocorrências criadas”, “Número de observações/notas”, que irão ser preenchidos automaticamente e um campo para digitar texto, onde o utilizador irá escrever o seu relatório. Terá também dois botões “Submeter” e “Voltar”.

5.3. Caso o utilizador escreva pelo menos 1 caracter será possível clicar no botão “Submeter” e caso clique em “Submeter” o relatório irá ser enviado para o backoffice.

5.4. Caso o utilizador deseje voltar para o menu anterior poderá clicar em “Voltar”.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário submeter relatórios de atividade. |
| Descrição dos inputs e de onde eles vêm:  Relatório->input do funcionário. |
| Descrição dos outputs e para onde vão:  Relatório->base de dados. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Relatório da atividade feita pelo membro do funcionário nesse dia. |
| Descrição da ação a ser feita:  Escrever um relatório. |
| Condições Pré e Pós:  Nenhum |
| Efeitos secundários:  Nenhum |

6.

6.1. O utilizador irá escolher a opção "Gerir Consumidores".

6.2. O utilizador irá escolher um consumidor.

6.3. Dentro da página do consumidor está lá um espaço no qual se pode escrever notas sobre o consumidor e os seus contadores.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário escrever notas sobre os consumidores e os seus contadores. |
| Descrição dos inputs e de onde eles vêm:  Nota/Observação->input do funcionário. |
| Descrição dos outputs e para onde vão:  Nota/Observação->base de dados. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Nota do funcionário sobre um consumidor ou contador. |
| Descrição da ação a ser feita:  Escrever uma nota. |
| Condições Pré e Pós:  Cliente registado. |
| Efeitos secundários:  Nenhum |

7.

7.1 O utilizador irá escolher a opção "Atualização".

7.2 O sistema deverá apresentar 3 botões. Um que permite ao utilizador enviar manualmente ("Envio Manual") os dados para plataforma web e o outro botão que permite enviar automaticamente ("Envio Automático") os dados para a mesma plataforma e outro botão que permite voltar ao menu anterior("Voltar").

7.3 Se o utilizador escolher o botão "Envio Manual", o sistema deverá enviar todos os dados do dispositivo móvel do utilizador para a plataforma web.

7.4 Se o utilizador escolher o botão "Envio Automático", o sistema deverá apresentar um campo que permitirá ao utilizador inserir um valor para o intervalo de tempo que o sistema terá de enviar os dados para a plataforma.

7.5 Se o utilizador escolher o botão "Voltar" o sistema deverá trazer o utilizador à home page.

|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário enviar manual ou automaticamente os dados para a plataforma web. |
| Descrição dos inputs e de onde eles vêm:  Tempo entre cada atualização->input do funcionário. |
| Descrição dos outputs e para onde vão:  Dados->plataforma web. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Atualização de dados manual ou automática dos dados da plataforma web. |
| Descrição da ação a ser feita:  Atualizar os dados da plataforma web. |
| Condições Pré e Pós:  Nenhum |
| Efeitos secundários:  Nenhum |

8.

8.1. O utilizador irá escolher a opção “Gestão de Ocorrências”.

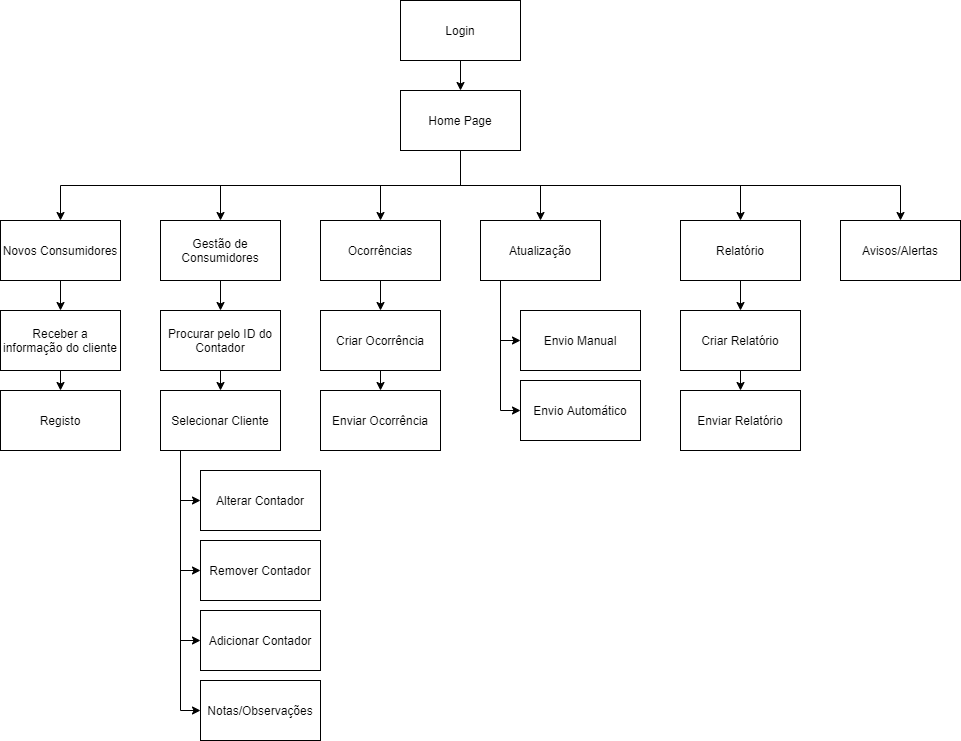
8.2. O sistema deve apresentar uma página com os campos : “Número da ocorrência”, que servirá como um identificador da ocorrência, sendo este um campo gerado automaticamente pelo sistema , “Descrição de ocorrência”, que é um campo onde o utilizador irá descrever a ocorrência, “Anexos”, que é um campo que permite a submissão de fotos e os botões “Submeter” e “Voltar”.

8.3. Caso o utilizador escreva pelo menos 1 caracter será possível clicar no botão “Submeter” e caso clique em “Submeter” o sistema irá pedir ao utilizador para confirmar se quer realmente submeter. Caso o utilizador confirme, será enviado para o backoffice.

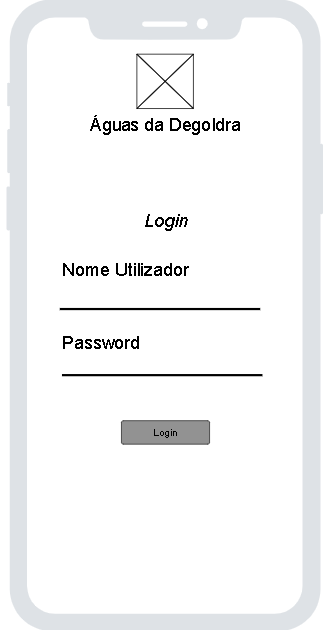
8.4. Caso o utilizador deseje voltar para o menu anterior poderá clicar no botão “Voltar”.

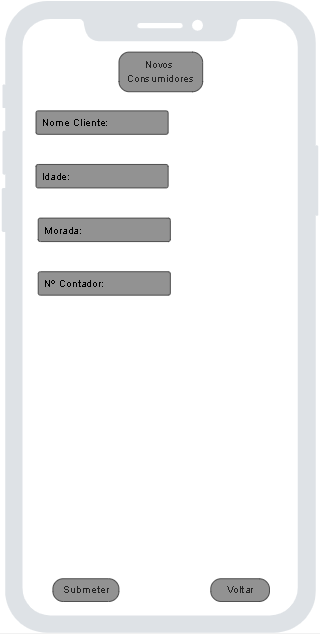
|  |
| --- |
| Definição da função ou entidade:  O sistema vai deixar o funcionário gerir ocorrências. |
| Descrição dos inputs e de onde eles vêm:  Ocorrência->input do funcionário. |
| Descrição dos outputs e para onde vão:  Ocorrência->base de dados. |
| Informação sobre a informação necessária para a computação e outras entidades usadas:  Gerir ocorrências. |
| Descrição da ação a ser feita:  Inserir ocorrência. |
| Condições Pré e Pós:  Nenhum |
| Efeitos secundários:  Nenhum |

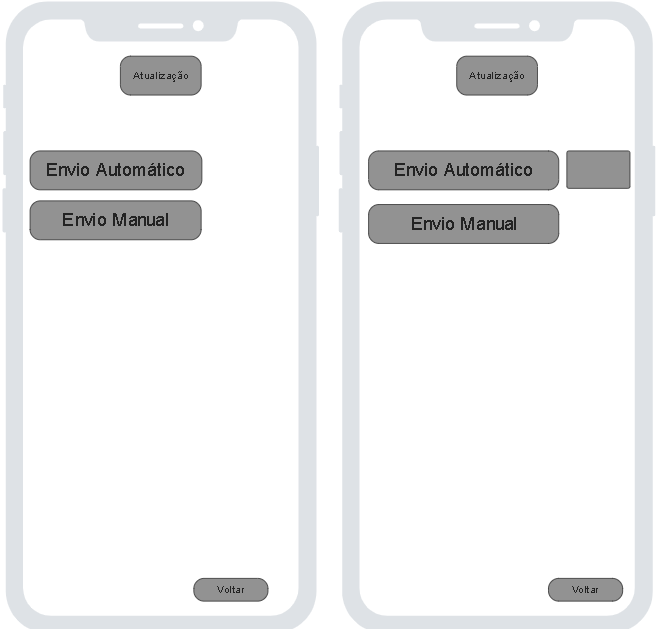
# Arquitetura de Sistema

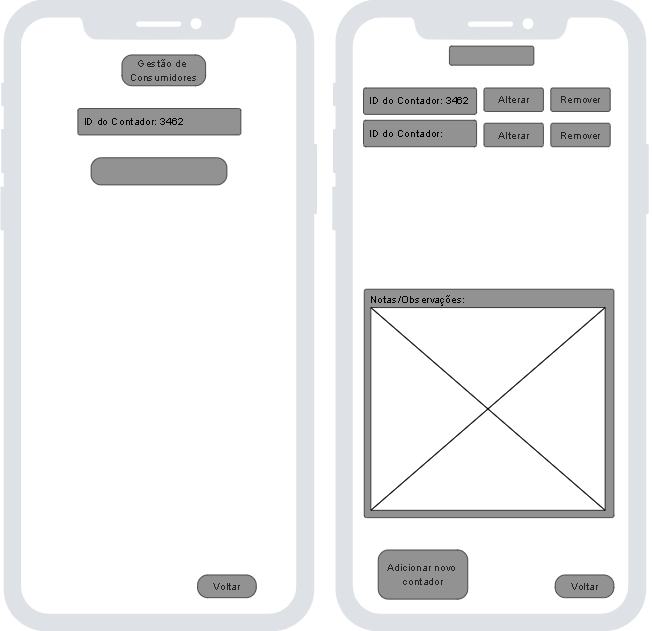


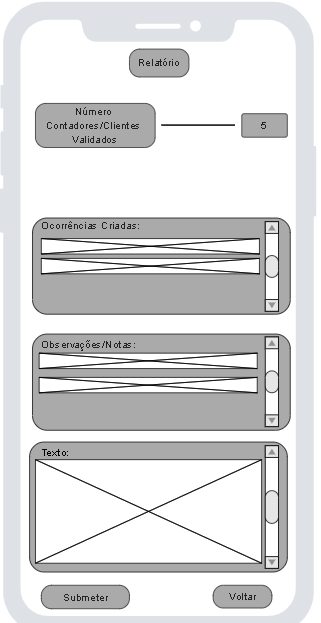
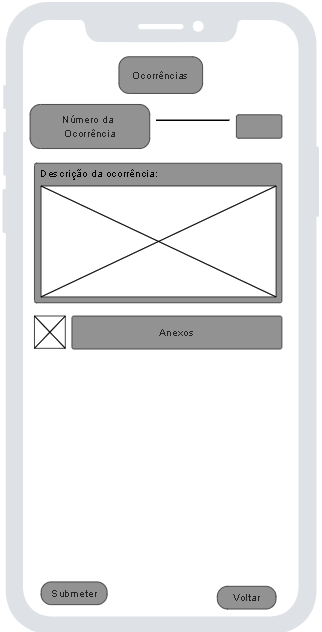
# Wireframes











# Mockups

