2016/2017

Escola Superior de Tecnologia

Henrique Costa nº12251

Joel Carvalho nº12607

Daniel Eleutério nº12715

uControl

IPCA – Computação Móvel ll



Índice

[Introdução 4](#_Toc472004477)

[Enquadramento 4](#_Toc472004478)

[Apresentação do projeto 4](#_Toc472004479)

[Tecnologias utilizadas 4](#_Toc472004480)

[Organização do relatório 5](#_Toc472004481)

[Desenvolvimento 6](#_Toc472004482)

[Persistência de dados 6](#_Toc472004483)

[Dados Exemplo 6](#_Toc472004484)

[Tabela 1 - Divisão 6](#_Toc472004485)

[Tabela 2 – Ac-Schedule 7](#_Toc472004486)

[Tabela 3 - Alarme 7](#_Toc472004487)

[Tabela 4 – Ar Condicionado 7](#_Toc472004488)

[Tabela 5 - Audio 7](#_Toc472004489)

[Tabela 6 - Estore 7](#_Toc472004490)

[Tabela 7 - Gravacoes 7](#_Toc472004491)

[Tabela 8 – Iluminacao 7](#_Toc472004492)

[Tabela 9 - Porta 8](#_Toc472004493)

[Tabela 10 - Tv 8](#_Toc472004494)

[Fig. 1 DER da aplicação 8](#_Toc472004495)

[Interface 9](#_Toc472004496)

[Fig. 2 Splash Screen 9](#_Toc472004497)

[Fig. 3 Menu principal 10](#_Toc472004498)

[Fig. 4 Blinds Fig. 5 Audio 10](#_Toc472004499)

[Fig. 6 Adicionar Fig. 7 Remover 11](#_Toc472004500)

[Implementação e processos 11](#_Toc472004501)

[Conclusão 13](#_Toc472004502)

[Objetivos realizados 13](#_Toc472004503)

[Limitações e trabalho futuro 14](#_Toc472004504)

[Apreciação final 14](#_Toc472004505)

[Anexos 15](#_Toc472004506)

[Fig. 3 Recordings 15](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004507)

[15](#_Toc472004508)

[Fig. 2 Television 15](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004509)

[Fig. 1 Menu 15](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004510)

[Fig. 5 Air Conditioner 15](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004511)

[Fig. 6 Audio 15](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004512)

[Fig. 4 Lights 15](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004513)

[Fig. 8 Alarms 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004517)

[Fig. 7 Blinds 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004518)

[Fig. 12 Add Tv 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004519)

[Fig. 12 Add Tv 16](#_Toc472004520)

[Fig. 11 Add 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004521)

[Fig. 10 Manage 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004522)

[Fig. 15 Remove division 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004523)

[Fig. 14 Remove tv 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004524)

[Fig. 13 Add division 16](file:///C:\Sdk\uControl\Relatório%20final%20-%20uControl.docx#_Toc472004525)

# Introdução

## Enquadramento

No âmbito da unidade curricular Computação Móvel 2, foi-nos proposto a realização de um projeto com base na linguagem Java, que consiste numa aplicação *Android*, focalizada na área da domótica.

A aplicação é capaz de facilitar a tarefa do utilizador, para dominar as tarefas básicas de uma casa inteligente.

Para a realização deste projeto foram utilizadas várias ferramentas aprendidas noutras disciplinas.

## Apresentação do projeto

O projeto centralizar-se-á, no desenho e no desenvolvimento uma aplicação que permite ao utilizador controlar as televisões, a iluminação, os ar condicionados, os dispositivos, os estores, os alarmes e todas as portas da sua habitação.

Objetivos a cumprir nesta aplicação:

1. O utilizador deve conseguir introduzir divisões ou qualquer outro elemento;
2. O utilizador pode atualizar o estado de qualquer elemento;
3. O utilizador pode consultar ou inserir divisões;
4. O utilizador pode consultar ou inserir qualquer elemento;
5. A interface deve ser simples e atrativa;
6. A aplicação deve ter uma boa estrutura de dados.
7. A aplicação deve ter embutido sensores.

## Tecnologias utilizadas

Para a realização deste projeto prático foram utilizadas várias ferramentas, nomeadamente:

* *Android Studio* – Programação da aplicação;
* *Visio* – Criação do DER;
* *PhpMyAdmin* – Criação da base de dados;
* *NinjaMock* – Planear e criar *Wireframes*;
* *Adobe Color* – Definir cores;
* *Photoshop* – Tratamento e criação de imagens.

## Organização do relatório

Inicialmente, começamos por explicar da base de dados do projeto, todo este planeado e criado num serviço *online*, intitulado de [www.000webhost.com](http://www.000webhost.com). Neste serviço, criamos a base de dados, bem como, o Diagrama Entidade Relação, composto por relações e tabelas.

De seguida, são descritos alguns detalhes do *wireframe,* desenhado em *NinjaMock*.

Para concluir, a parte da implementação do código e passar de toda esta teoria para a prática.

# Desenvolvimento

Este projeto, foi concebido por três alunos, sendo mais fácil a divisão de tarefas e partilha de conhecimentos.

Inicialmente, o maior foco foi a idealização do projeto. Iniciamos pelo desenho da aplicação em papel, para discutirmos o posicionamento dos elementos a aparecer no ecrã. Sequentemente, passamos do papel para o *NinjaMock*, onde é possível colocar *links* entre as páginas e visualizar tudo com maior rigor. Posto isto, definimos cores e o tipo de letra da nossa aplicação, usando o *Photoshop*.

De seguida, cuidamos da base do projeto, a base de dados, estruturamos a base de dados e construímos o DER , com o auxílio do *Visio.*

Após a base de dados, passamos para o *AndroidStudio*, construindo as classes principais. Durante a construção da base de dados no *AndroidStudio*, fomos adequando, em paralelo, a base de dados no *PhpMyAdmin*, bem como, os *php´s* necessários para a listagem, remoção e atualização dos elementos que compõem a base de dados.

Para terminar, algumas das maiores dificuldades sentidas, foi na implementação do código, pois este é um processo demorado, que exige bastante esforço e dedicação.

## Persistência de dados

### Dados Exemplo

|  |  |
| --- | --- |
| **idDivisao** | **descricao** |
| 1 | Quarto |

Tabela - Divisão

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **idAcSchedule** | **horaInicio** | **horaFim** | **modo** | **intensidade** | **idAc** |
| 1 | 11:30:00 | 21:30:00 | Regular | 10 | 1 |

Tabela 2 – Ac-Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| **idAlarme** | **descricao** | **divisao** | **estado** |
| 1 | Alarme do quarto | 1 | 1 |

Tabela 3 - Alarme

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **idArCondicionado** | **descricao** | **divisao** | **estado** | **Temperatura** | **modo** |
| 1 | Principal | 1 | 1 | 27 | Cold |

Tabela 4 – Ar Condicionado

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **idAudio** | **descricao** | **divisao** | **estado** | **Volume** |
| 1 | Colunas Boss | 1 | 1 | 10 |

Tabela 5 - Audio

|  |  |  |  |
| --- | --- | --- | --- |
| **idEstore** | **descricao** | **divisao** | **Posição** |
| 1 | Estore principal | 1 | 5 |

Tabela 6 - Estore

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **idGravacao** | **canal** | **horaInicio** | **horaFim** | **idTv** |
| 1 | 1 | 13:00:00 | 21:00:00 | 1 |

Tabela 7 - Gravacoes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **idIluminacao** | **descricao** | **divisao** | **estado** | **intensidade** |
| 1 | Luz do quarto | 1 | 1 | 10 |

Tabela 8 – Iluminacao

|  |  |  |  |
| --- | --- | --- | --- |
| **idPorta** | **descricao** | **divisao** | **Estado** |
| 1 | Porta da rua | 1 | 0 |

Tabela 9 - Porta

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **idTv** | **descricao** | **divisao** | **estado** | **Canal** |
| 1 | Tv do quarto | 1 | 1 | 10 |

Tabela 10 - Tv

**DER**

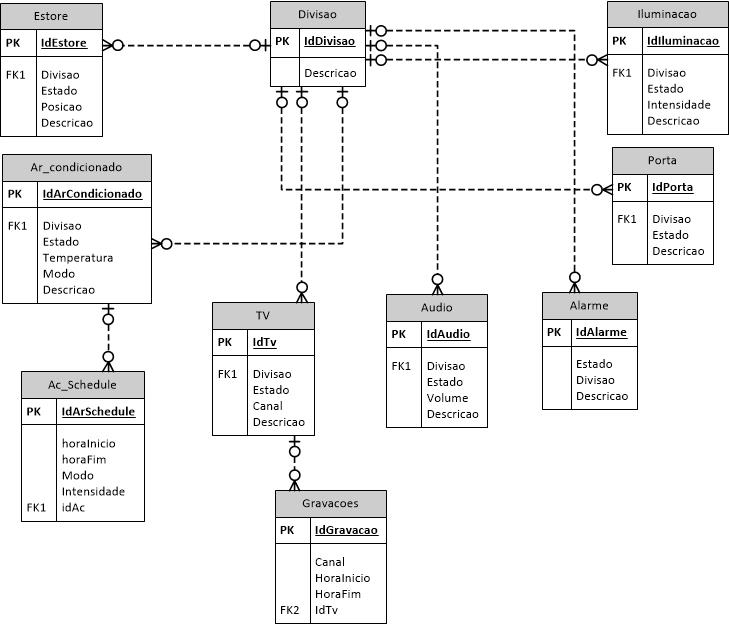


Fig. 1 DER da aplicação

Como é observável, na Fig.1, o Diagrama de Entidade Relação, é composto por 10 tabelas, todas relacionadas entre si.

A tabela principal “Divisao”, importantíssima, porque sem ela, a aplicação não poderia funcionar, qualquer elemento da casa é composto por uma divisão. Esta tabela é composta por um id único e uma descrição.

As tabelas “Estore”, “Iluminação”, “Porta”, “Ar condicionado”, “Tv”, “Alarme” e “Audio”, são compostas pelos seus atributos complementares, sendo que dois estão presentes nelas todas, um id único e uma descrição, fundamentais para a identificação de cada objeto manipulado pela *app*.

Finamente, as tabelas “Ac\_Schedule” e “Gravacoes”, são fundamentais para o bom uso de ar condicionado e da televisão, respetivamente. Isto é, em relação à “Ac\_Schedule” o utilizador é capaz de programar, qualquer ar condicionado, para a hora que ele desejar, escolhendo o modo e a temperatura a pretender para o momento. Relativamente, às gravações, o *user* pode gravar qualquer programa televisivo e aceder a ele, em qualquer situação.

## Interface

Iniciamos por criar vários *wireframes* lançados na plataforma *NinjaMock*, onde desenvolvemos a estrutura básica da aplicação para perceber como iria funcionar depois de programada.

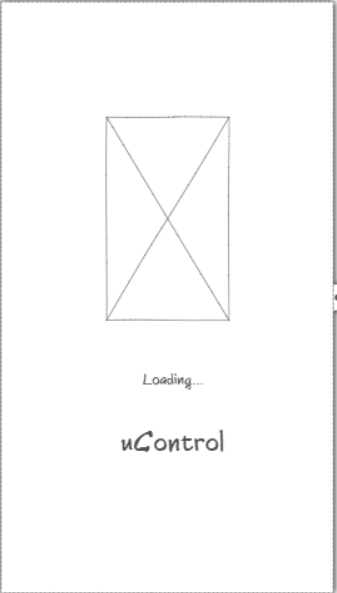


Fig. 2 Splash Screen

Como é observável, na Fig. 2, na elaboração do protótipo, começamos por criar um *splash screen,* para dar uma boa imagem à aplicação. E não surgir imediatamente, o menu da *app.*

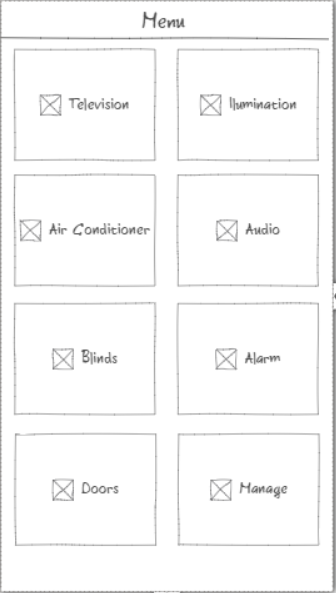


Fig. 3 Menu principal

Na Fig. 3, de seguida criamos um layout principal, com oito botões, sete deles para elementos de uma casa e os restantes para a gestão da aplicação.

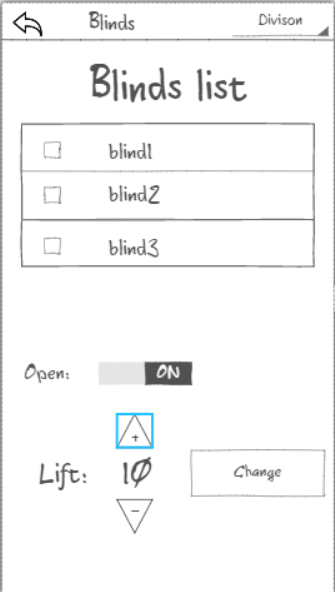


Fig. 4 Blinds Fig. 5 Audio

A Fig. 4 e a Fig. 5, dizem respeito, à atividade *Blinds* e à atividade *Audio,* respetivamente, após o utilizador clicar num botão do menu, obtém uma página idêntica a estas. Relativamente às outras páginas, contêm algumas diferenças, mas na generalidade são muito idênticos.



Fig. 6 Adicionar Fig. 7 Remover

Sobre as Fig. 6 e Fig. 7, correspondem ao adicionar e ao remover uma televisão. Para adicionar, o utilizador escolhe a divisão onde está colocada a nova televisão e para remover basta selecionar o nome a remover e confirmar.

Posto isto, ocorreram algumas melhorias nos *layouts*, especialmente nas cores, visto que, o *feedback* obtido por potenciais utilizadores não foi o melhor.

## Implementação e processos

Relativamente à programação, dividimos o projeto em dois *packages, add* e *remove* para uma melhor organização do projeto.

Tentamos resolver os objetivos da aplicação de forma sequencial, porque toda a aplicação precisava de ser montada de forma solida, pois vários aspetos dependiam uns dos outros. Sendo assim, começamos por definir *layouts,* começando mesmo pelo menu, para que depois pudéssemos ir resolvendo cada aspeto e testando no *smarphone*.

No menu principal, possui oito botões, que permitem ao utilizador aceder a qualquer parte da aplicação, cada um deles representa um elemento duma casa, onde é possível controlar, televisões, a iluminação, os ares condicionados, os áudios, os estores, o alarme, as portas e por último o *user* pode gerir estes elementos, ou seja, pode remover ou adicionar componentes à aplicação. Relativamente às cores, elegemos seis cores, sendo elas, laranja escuro, laranja claro, laranja avermelhado, roxo, amarelo e vermelho.

Na atividade *television,* o utilizador tem ao ser dispor todas as televisões, podendo assim, mudar de canal, bem como ligar/desligar. A partir desta página, podemos ter acesso às gravações, onde é possível selecionar uma televisão e gravar qualquer programa televisivo. Damos também a possibilidade de utilizador, poder ver as gravações.

Na atividade *illumination,* o utilizador tem a possibilidade de escolher a lâmpada a usar, podendo liga-la ou desliga-la, ou se preferir mudar apenas a intensidade de luz.

De seguida, temos a atividade *air conditioner*, a mais complexa do projeto, onde o utilizador tem duas opções: programar os ares condicionados, definindo uma hora inicial e fim, um modo (*Cold*, *Heat*, *Regular*) e uma temperatura, para que o aparelho se ligue à hora pretendida. A outra opção, o *user*, simplesmente, define uma temperatura e um modo para cada aparelho, no ato, ou apenas desliga ou liga os ares condicionados.

Na atividade *audio*, o utilizador pode aumentar ou diminuir o som e ligar/desligar os dispositivos.

Na atividade *blinds*, o utilizador tem livre vontade, de escolher a abertura dos estores numa escala de 0 a 10.

Na atividade *alarms*, o utilizador apenas seleciona o alarme e poderá liga-lo ou desliga-lo.

Na atividade *doors*, o utilizador opta por uma porta e pode abri-la ou fecha-la.

Por último, a atividade *manage*, é das mais relevantes para o utilizador usufruir da *app*, conseguindo adicionar ou remover qualquer componente, à sua casa.

# Conclusão

Em suma, este trabalho foi realizado com satisfação, onde todos os objetivos foram realizados, com sucesso.

Relativamente, à programação, foi onde sentimos mais dificuldades, nomeadamente na parte de incorporar os sensores na aplicação.

Outra das complexidades, foi na remoção de dados, isto é, as tabelas que compõem a aplicação, continham “id’s” automáticos e, na listagem de qualquer elemento da casa, a remoção era feita através do posicionamento dos “id’s” num *spinner*, por outras palavras, os “id’s” eram igualados a essa mesma posição. Então gerou, certos problemas, porém mais tarde percebemos que conseguíamos remover de forma correta a partir do campo “descrição”.

Este trabalho foi muito produtivo, porque nos fez refletir e trabalhar em conjunto, partilhando ideias entre nós. Deparamo-nos com uma situação bastante semelhante àquelas que existem no mundo de trabalho e deixou-nos mais contextualizados com o que iremos enfrentar no mundo de trabalho.

Relativamente, à relação entre os elementos do grupo foi a melhor, porque nos entendemos bastante bem e já realizamos outros projetos em conjunto. Para concluir, reiteramos que foi com grande satisfação realizar este trabalho.

## Objetivos realizados

Os objetivos que foram realizados são:

1. O utilizador consegue introduzir divisões ou qualquer outro elemento;
2. O utilizador consegue atualizar o estado de qualquer elemento;
3. O utilizador consegue consultar ou inserir divisões;
4. O utilizador consegue consultar ou inserir qualquer elemento;
5. A interface consegue ser simples e atrativa;
6. A aplicação tem uma boa estrutura de dados.
7. A aplicação possui o sensor acelerómetro, para a inserção de elementos da casa.

## Limitações e trabalho futuro

A aplicação tem algumas limitações, a principal é, não mostrar a descrição da divisão nas listas de cada elemento, isto é, por exemplo, na visualização de um ar condicionado, o utilizador tem que ser descritivo na sua descrição, senão não saberá a que divisão pertence.

Esta limitação, pode ser resolvida no futuro, para isso, a lista tinha que ser composta por dois elementos, o nome do aparelho e a descrição da divisão, onde ele está colocado.

Outro aspeto a melhorar, é o facto de na programação automática de um ar condicionado, aparecer os segundos, estes segundos são mostrados porque o tipo “*Date”* na base de dados, necessita dos segundos. Este aspeto pode ser resolvido, colocando assim os segundos, só no *url* na inserção.

Com tudo, estas lacunas são para tentar resolver, porque seria uma melhoria da aplicação. Com a implementação de sensores nos aparelhos da casa, é possível tornar a aplicação utilizável de forma segura sem qualquer tipo de erro ou falha.

## Apreciação final

Este trabalho, foi benéfico, e nos deu um enorme prazer, porque na verdade conseguimos fazer algo consistente.

Embora tenhamos consciência que é difícil conseguir alcançar sucesso com a nossa aplicação, devido à incorporação de sensores nos aparelhos, envolvendo parceiros e alguns gastos financeiros.

Anexos

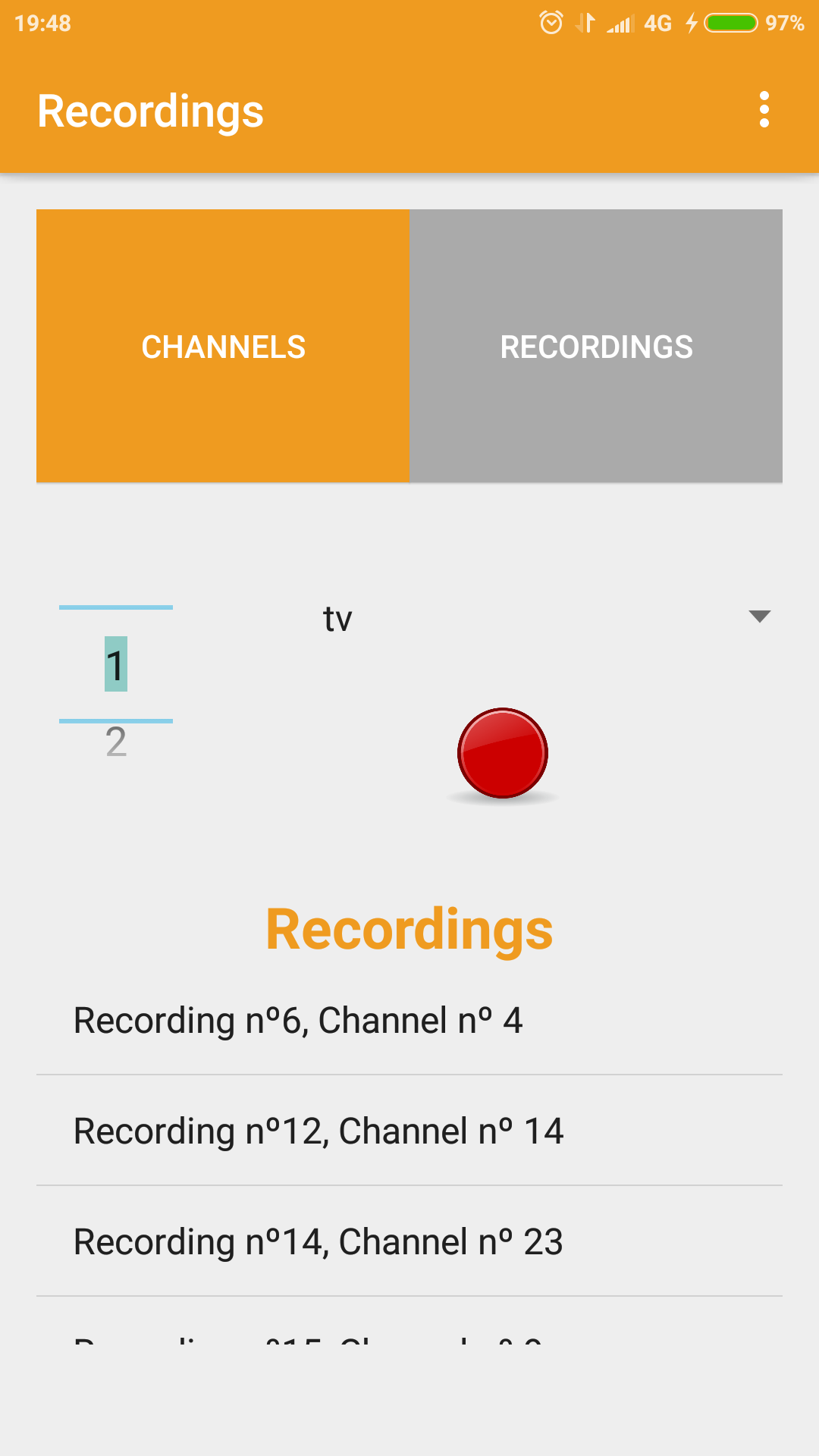
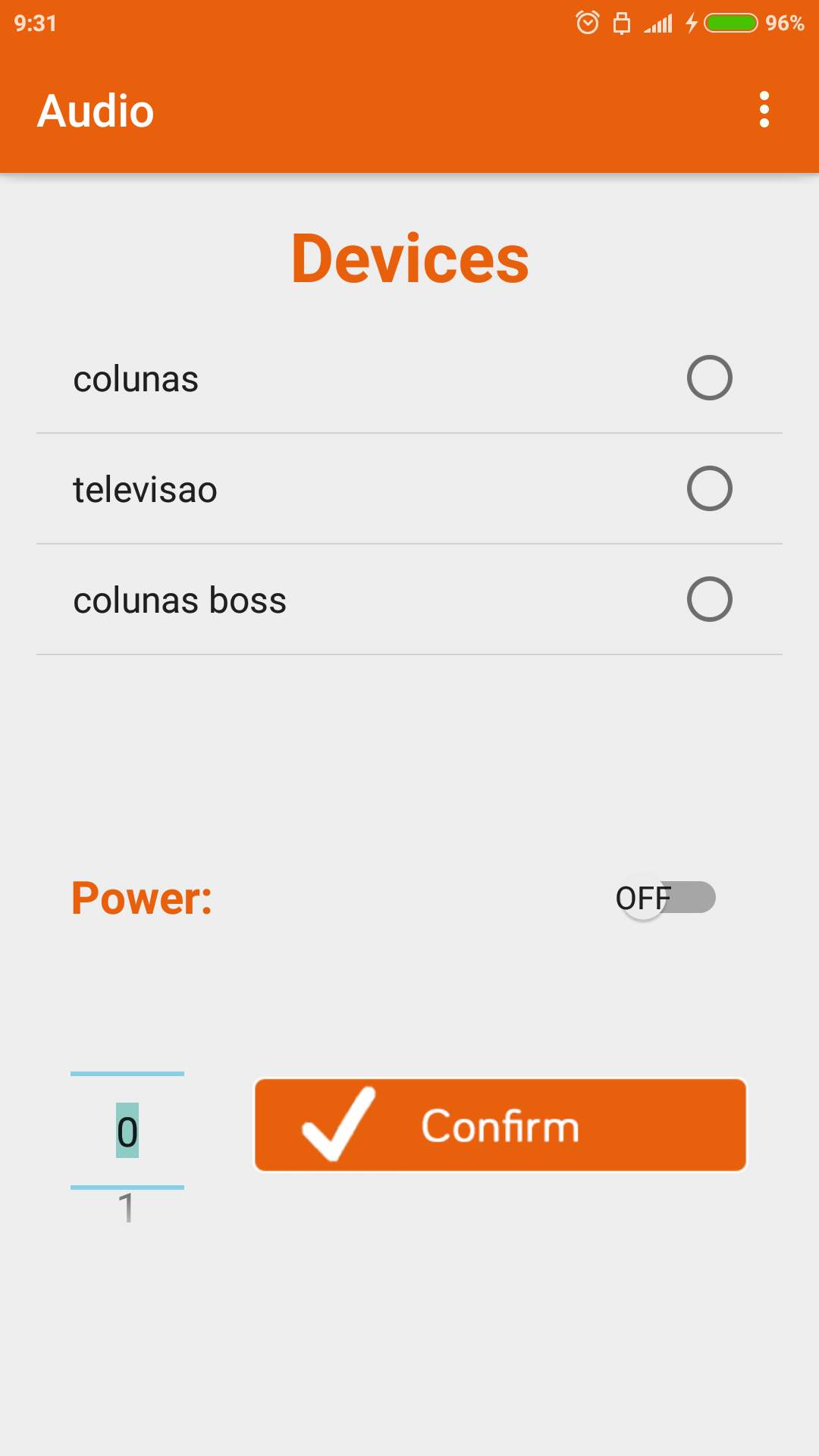
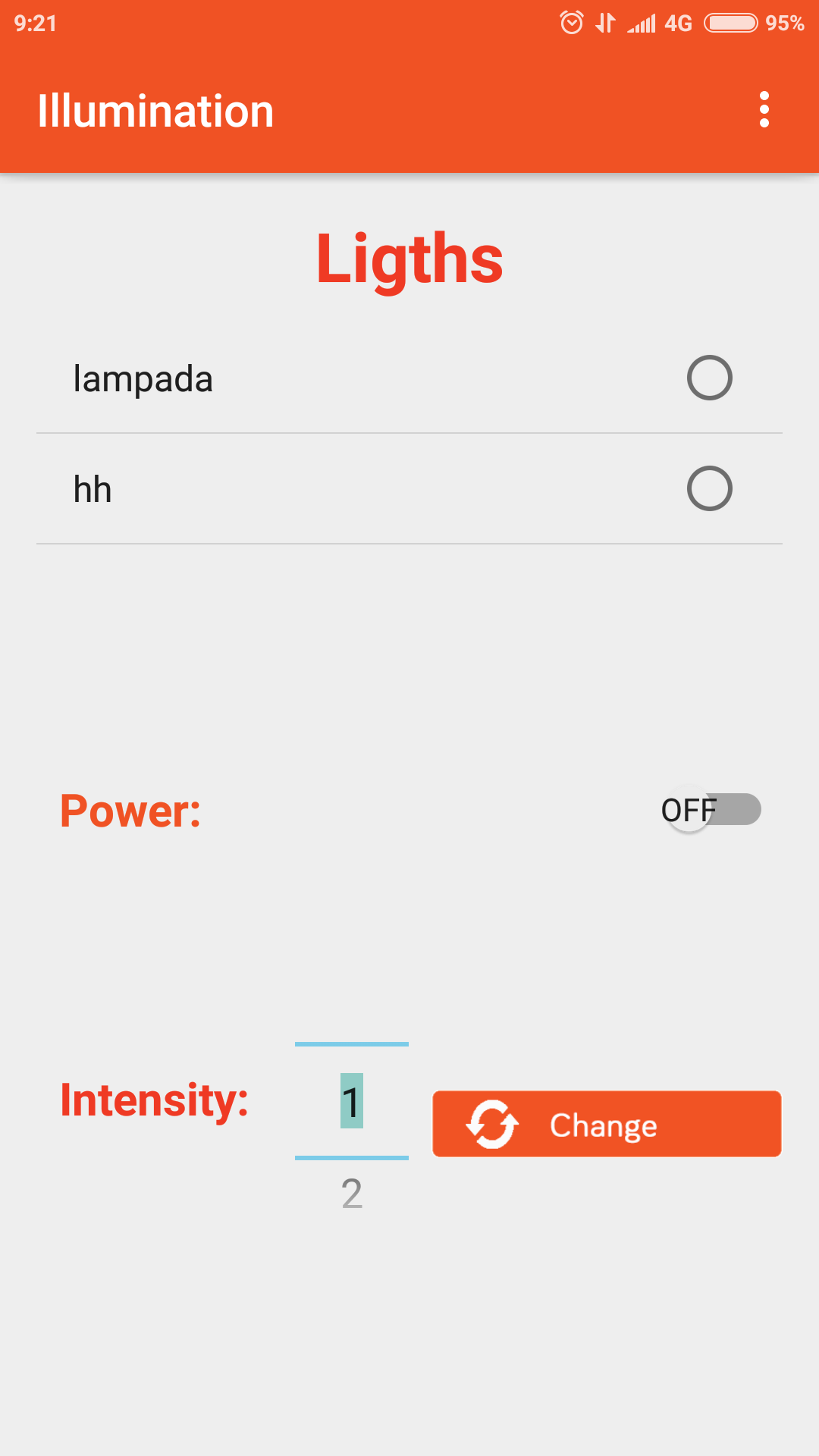
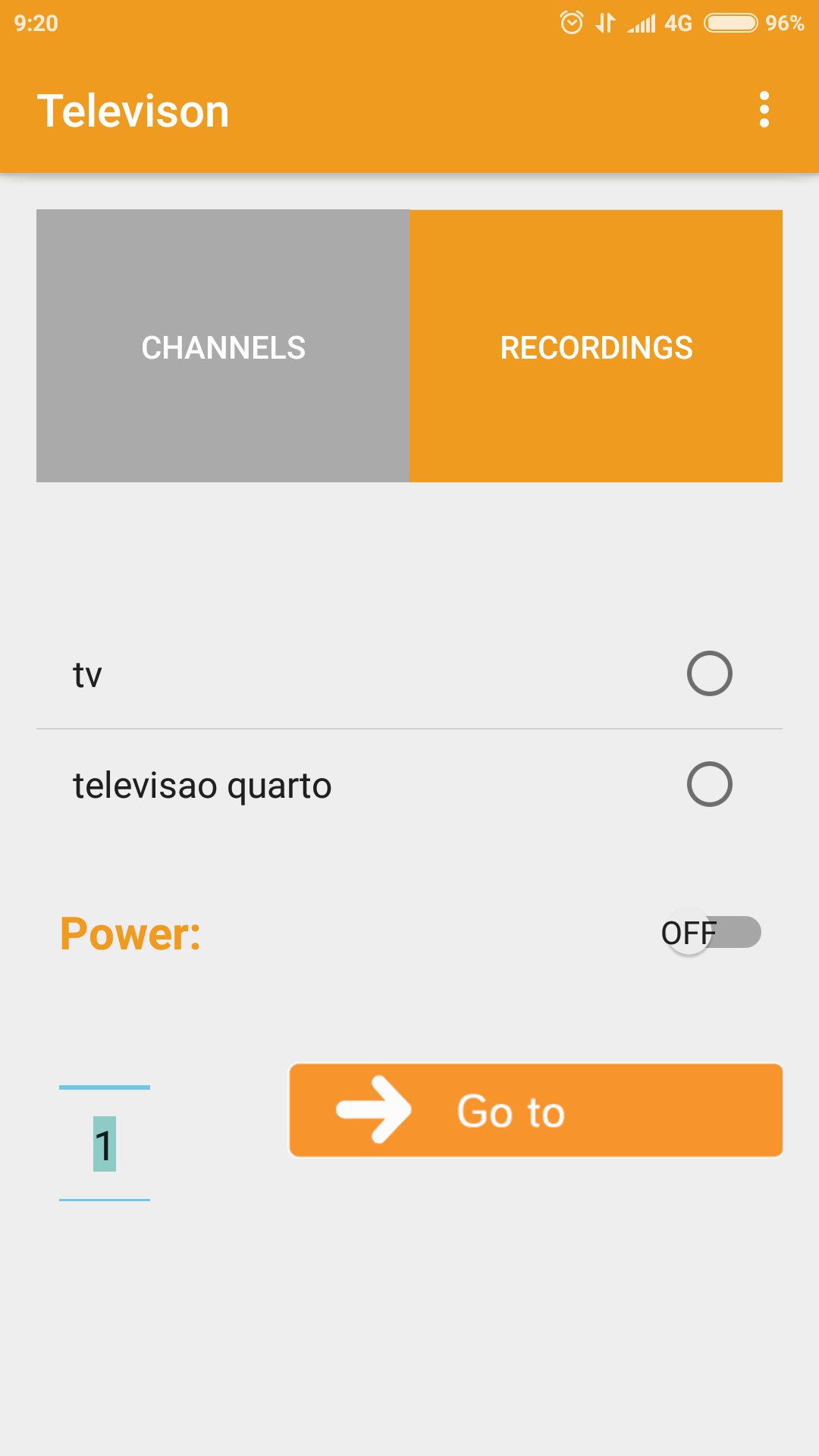


Fig. 3 Recordings

Fig. 2 Television

Fig. 1 Menu

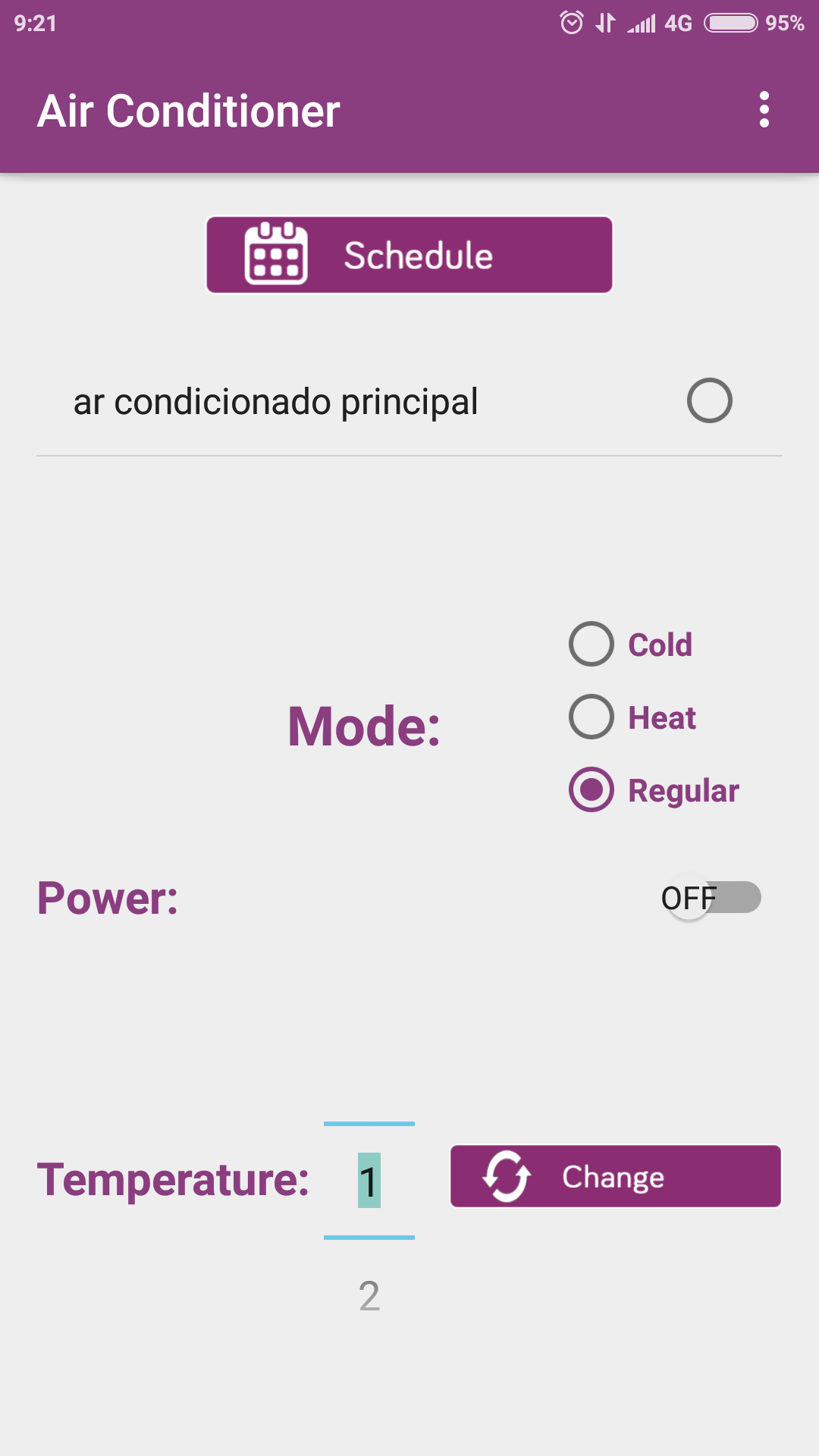


Fig. 5 Air Conditioner

Fig. 6 Audio

Fig. 4 Lights

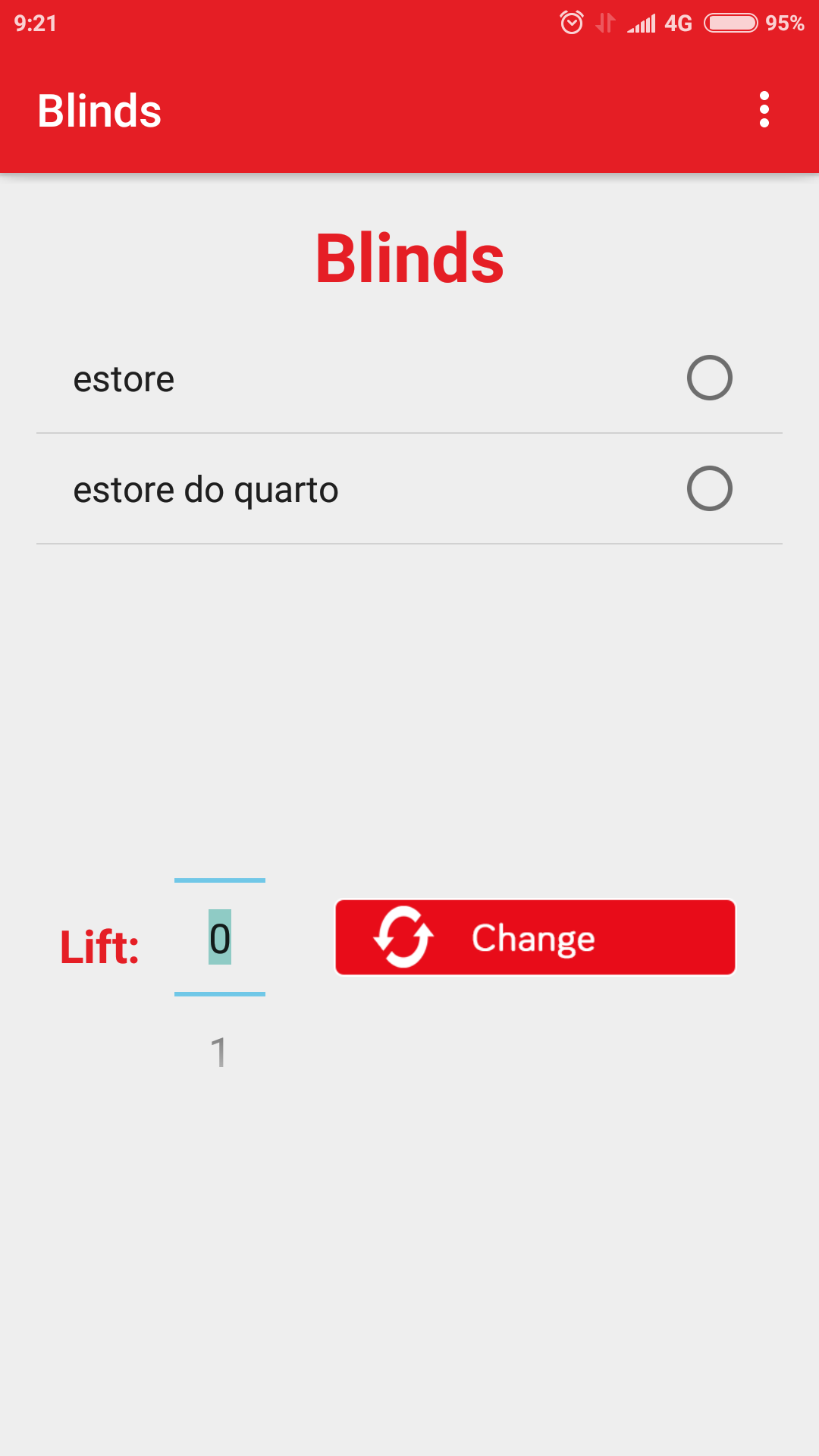
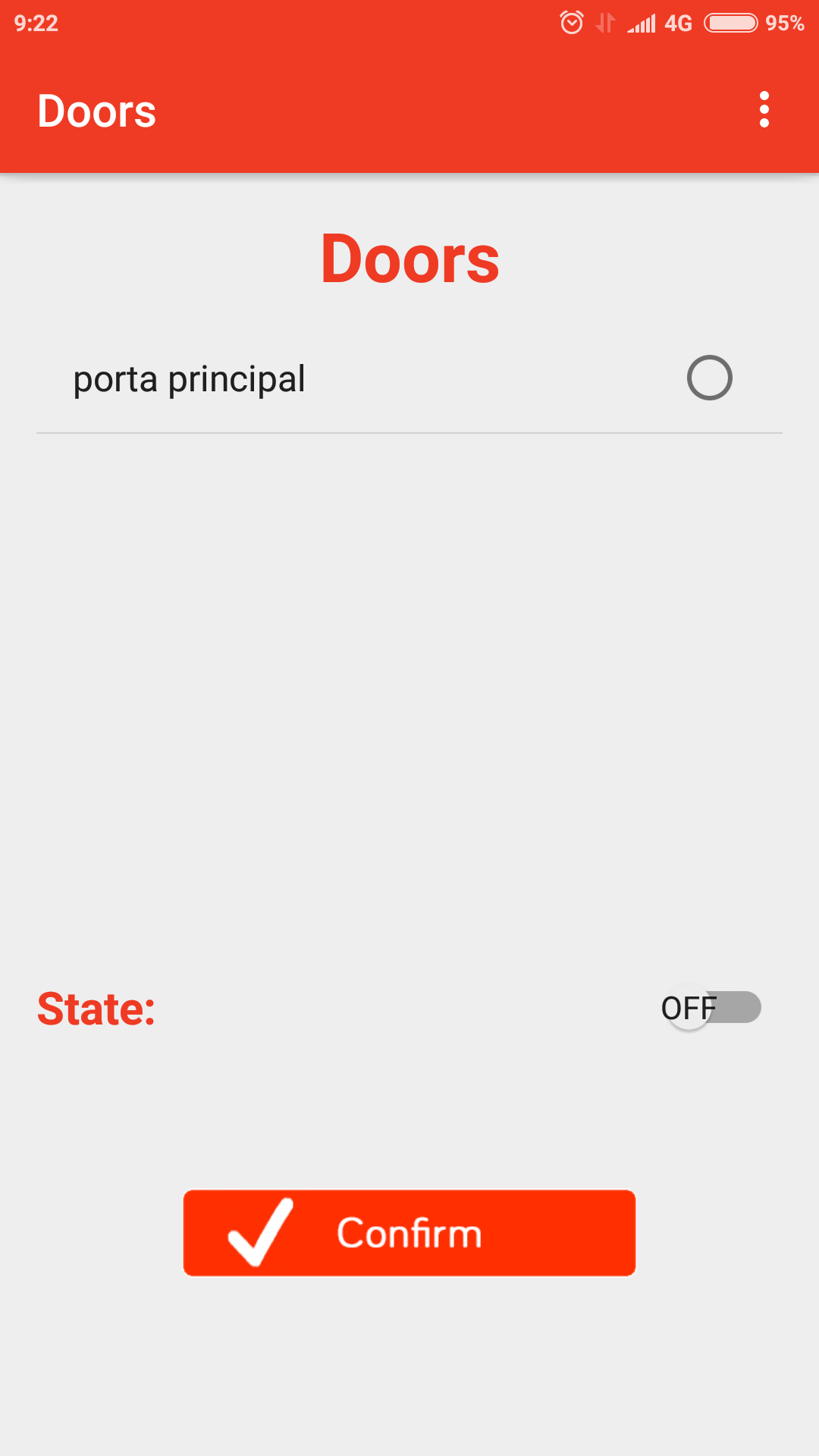
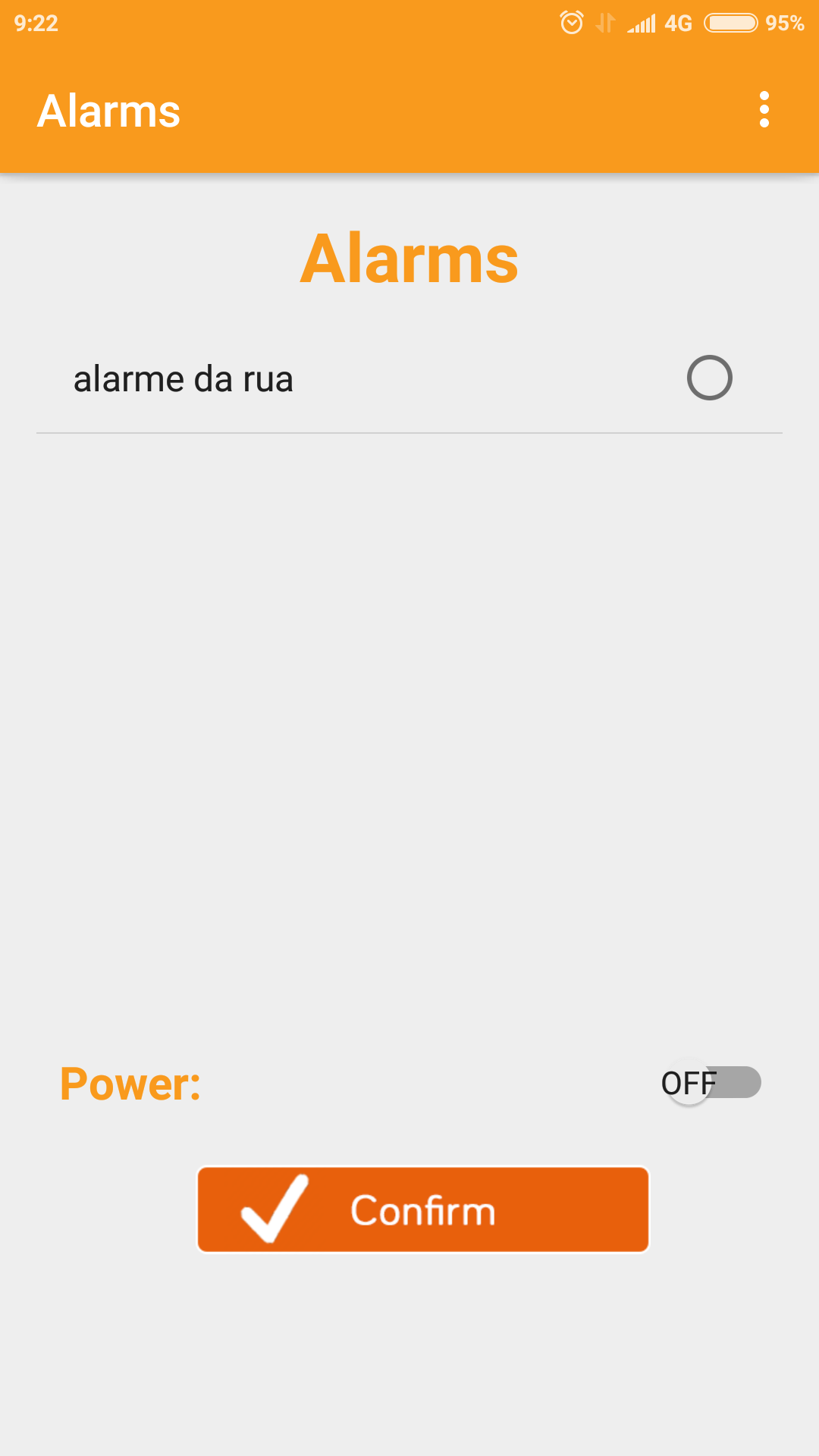


Fig. 9 Doors

Fig. 8 Alarms

Fig. 7 Blinds





Fig. 12 Add Tv

Fig. 12 Add Tv

Fig. 11 Add

Fig. 10 Manage

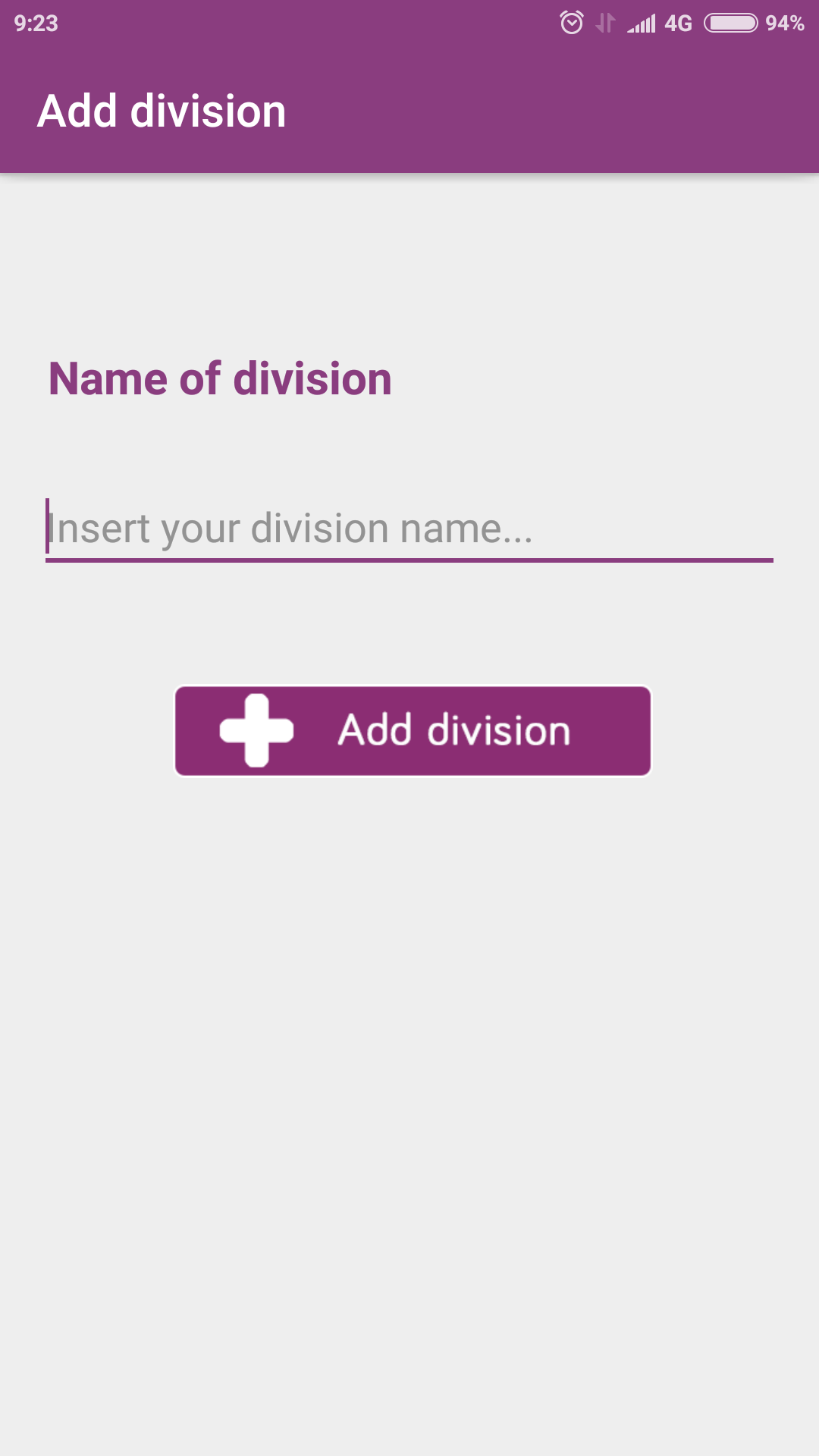
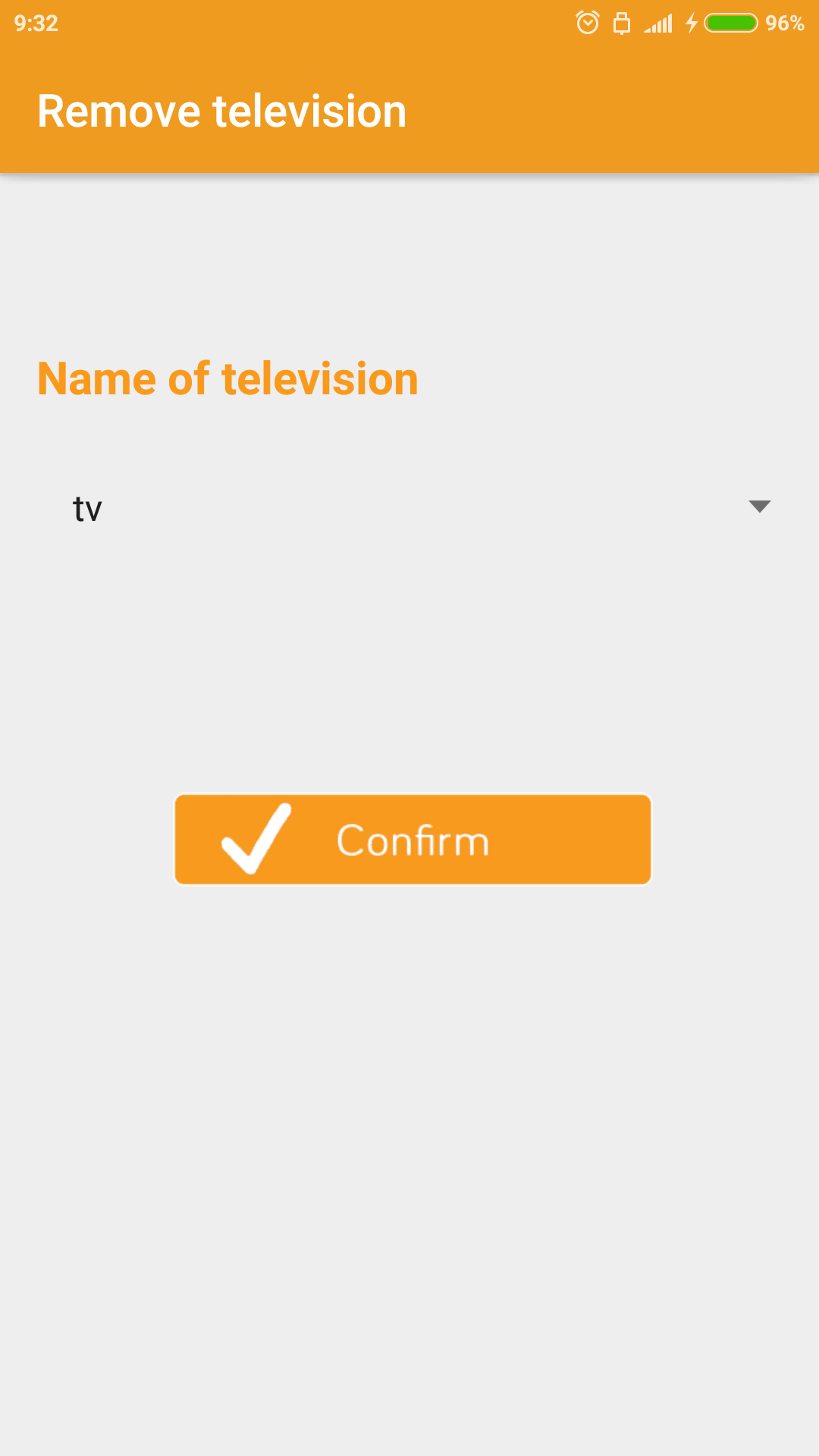
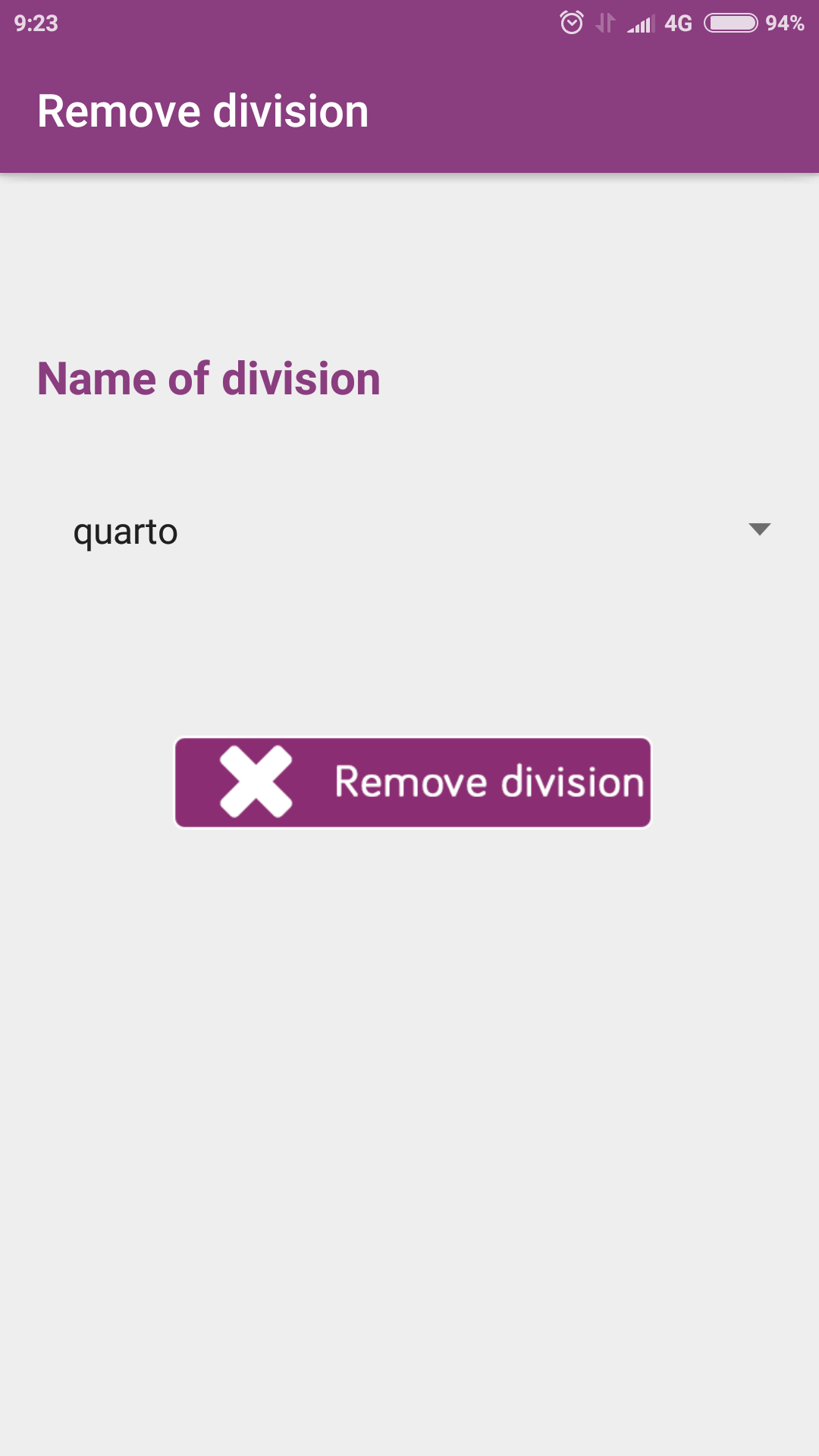


Fig. 15 Remove division

Fig. 14 Remove tv

Fig. 13 Add division

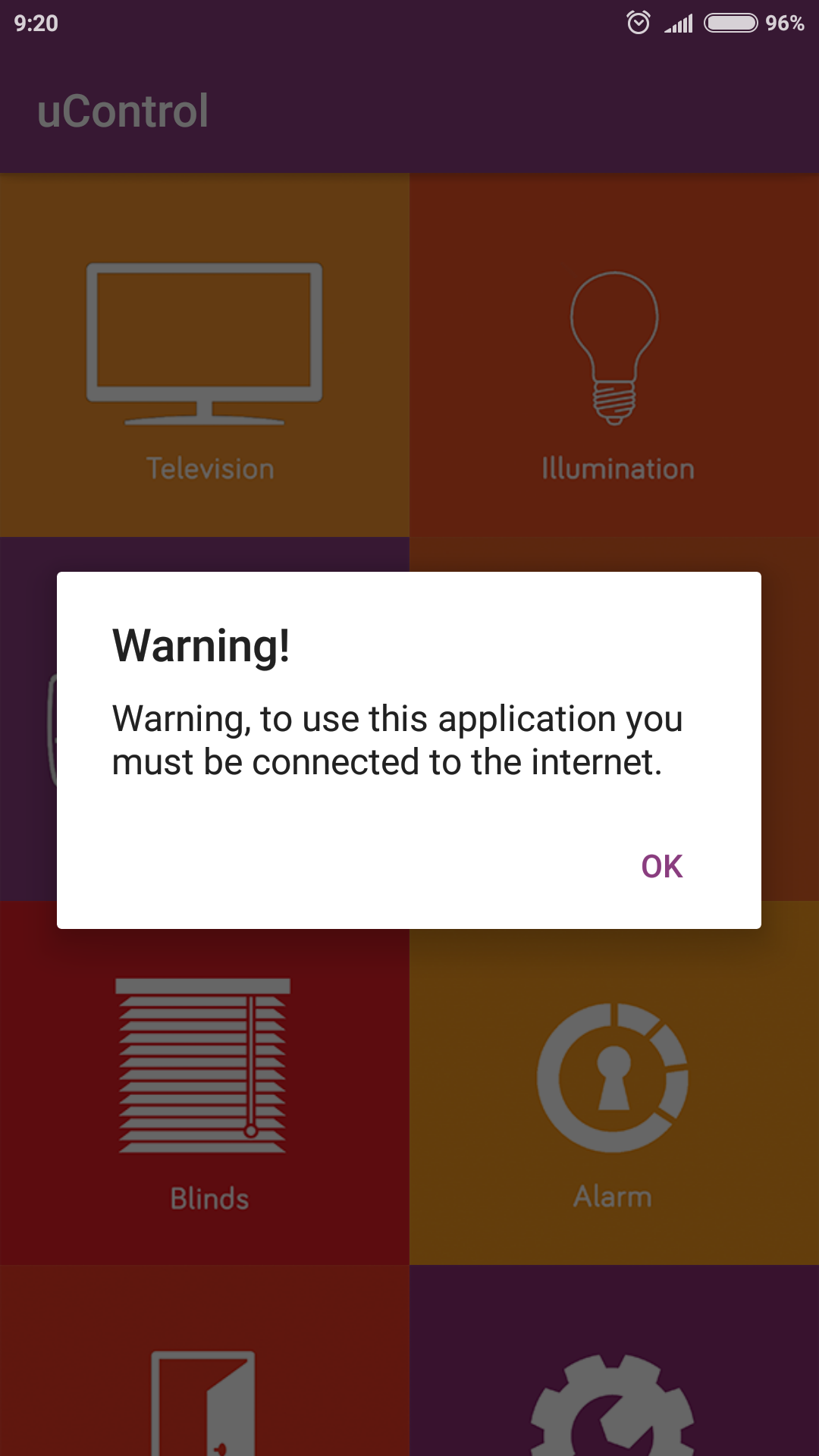
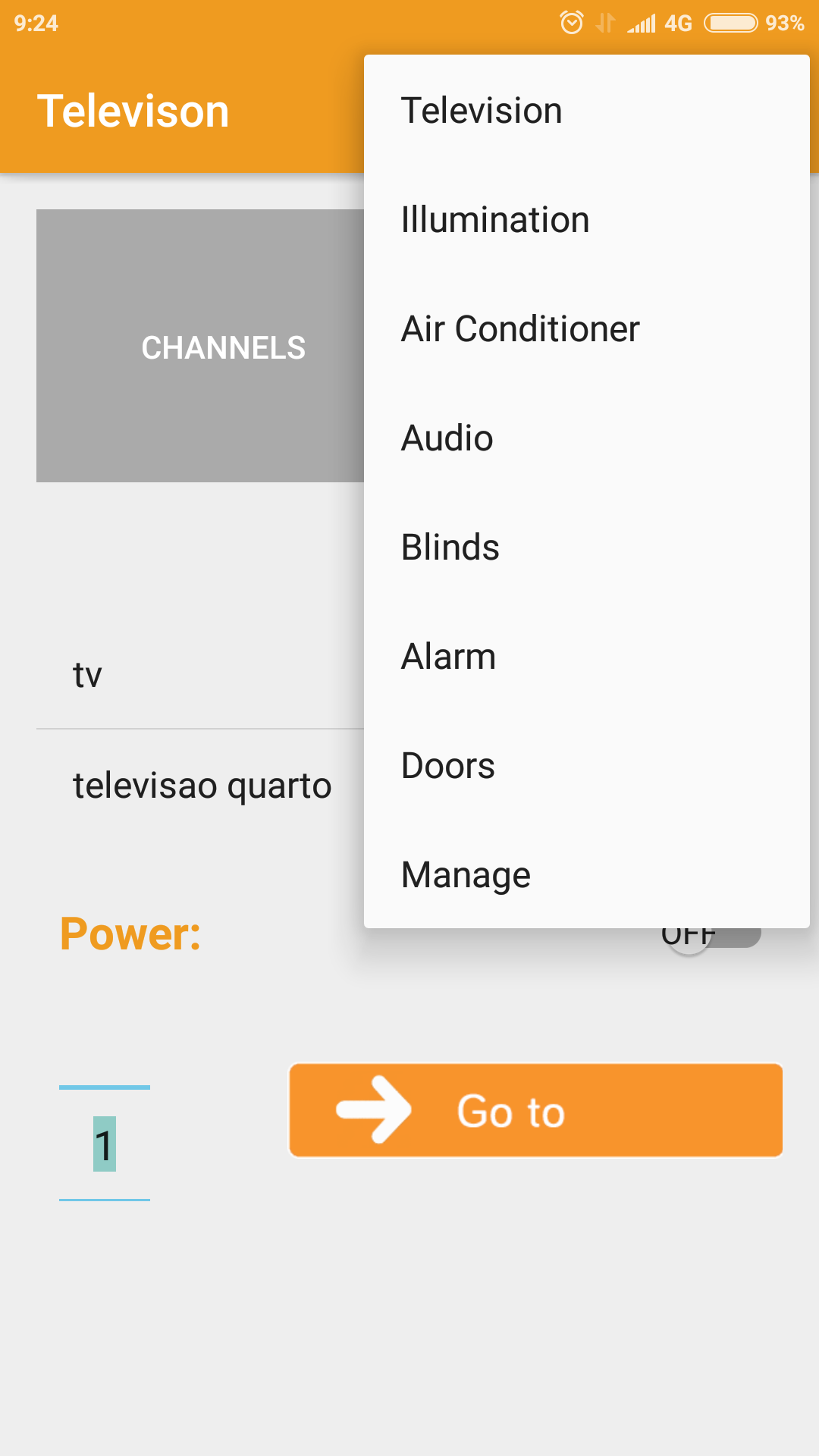


Fig. 17 DropMenu

Fig. 16 Internet warning