ANDROID DOMOBUS INTERFACE

Smart Home App

Ambient Intelligence

Isabel Costa | 76394 | METI

MOTIVATION

- Ambience Intelligence is applied in multiple contexts
- It's important to design and develop smooth interfaces to control houses
- There is a lot of devices from different manufactures that makes the creation of generic applications a difficult task

OBJECTIVES

- Create user-friendly Android application to control a smart home
  - Generic and flexible according to DomoBus
  - Show and control divisions and devices
- Create home server to simulate a central server which provides devices current values
  - Server must provide simple API

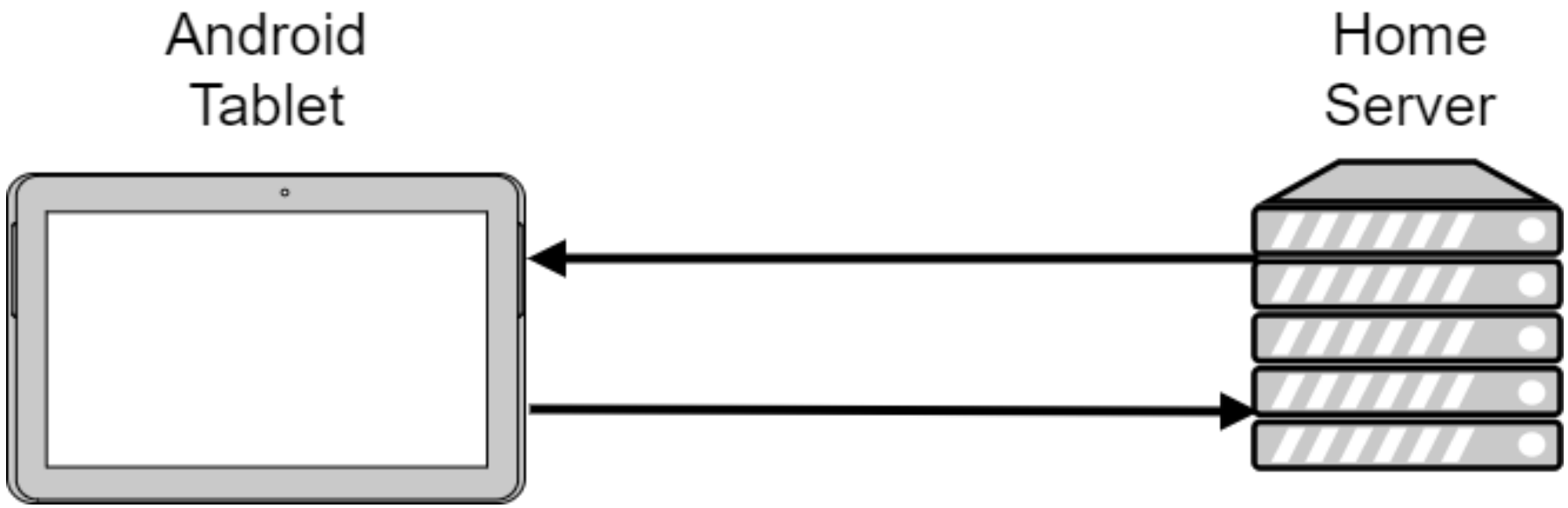
RELATED WORK

- DomoBus System
- RESTful APIs – (GET, POST, PUT, DELETE)
- JSON
- Android Architectures
  - Model-View-ViewModel (MVVM)
  - Clean Architecture
  - Model-View-Presenter (MVP)

TOOLS

- Android Studio
- Git & Github
- Open Source libraries:
  - Retrofit
  - ButterKnife
  - GSON
- Pixabay
- Canva

SOLUTION - SCENARIO



SOLUTION – COMPONENTS/MODULES

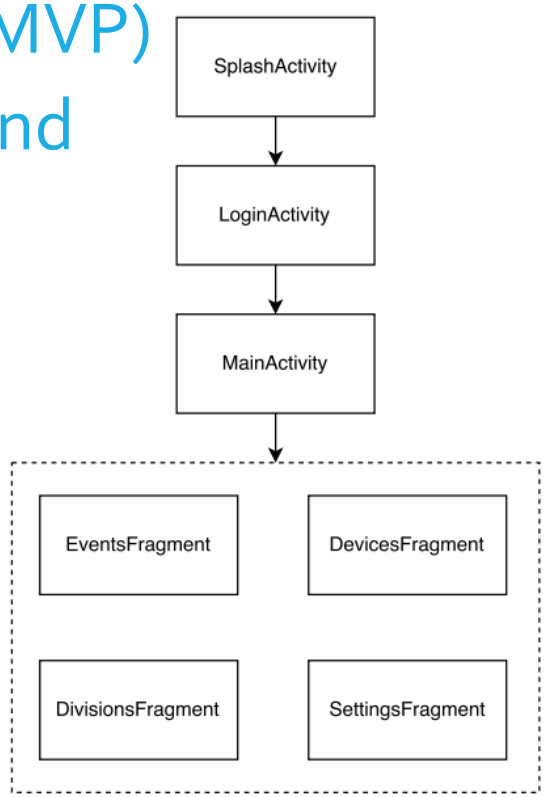
Android Application

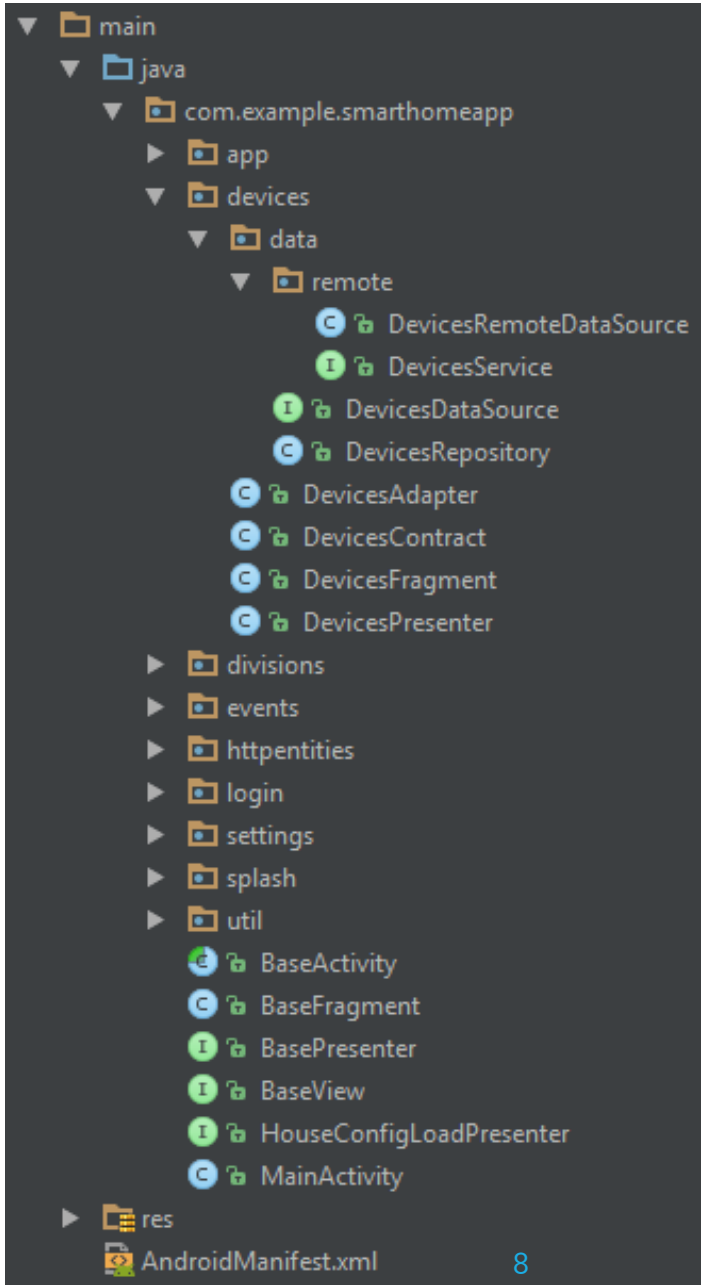
Java Server

Common Library
(XML Parser & Entities)

ANDROID

- Internal code architecture:
  - Model-View-Presenter (MVP)
  - Flexible code structure and easy to mantain
- Material Design
- Minimum version:
  - 4.4 - KitKat – API 19





SERVER

- HTTP Server implemented in Java
- URL available at: http://<ip address>:9000/

```
=> Creating Server
=> Parsing configuration file ...
Root element :DomoBusSystem
=> Starting HTTP Server ...
Server started at 9000
=> Starting Command Receiver ...

Execute one of these commands:
{1} INIT -D <deviceId> -v <valueX>   -> set a device's initial value
{2} SET  -D <deviceId> -v <valueX>   -> set a device's value
{3} GET  -D <deviceId>               -> get info from a device
{4} GET  -R <deviceId>               -> get info from a division/room
{5} GET  -D <deviceId>               -> get all devices {id :: name}
```

SERVER API

- GET /devices
- GET /divisions/{id}/devices
- GET /devices/{id}

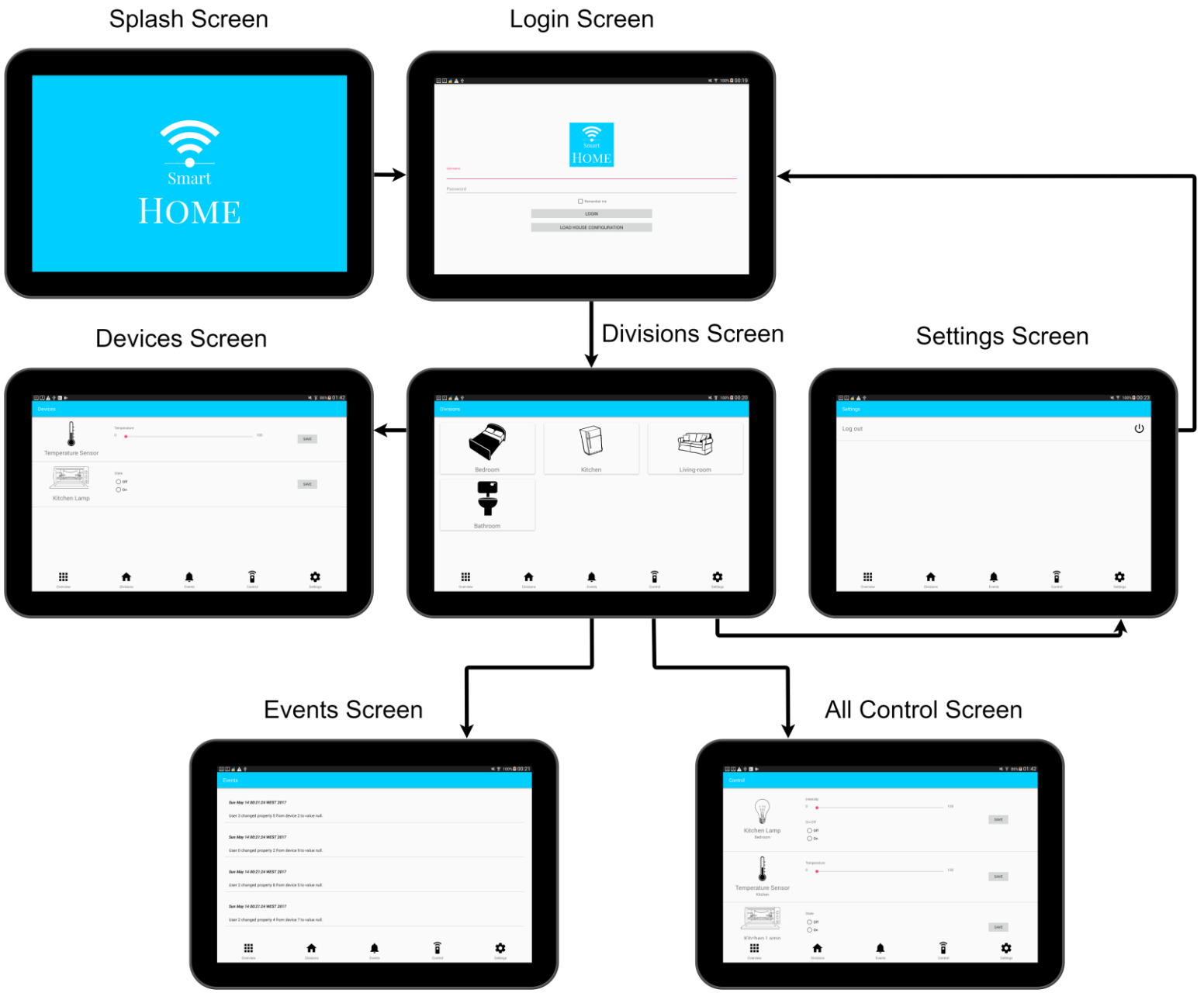
```
{
  "deviceId": "2",
  "values": [
    {
      "propertyId": "1",
      "propertyValue": "23"
    }
  ]
}
```

```
[
  {
    "deviceId": "2",
    "values": [
      {
        "propertyId": "1",
        "propertyValue": "23"
      }
    ]
  },
  {
    "deviceId": "3",
    "values": [
      {
        "propertyId": "1",
        "propertyValue": "OFF"
      }
    ]
  }
]
```

SOLUTION - FUNCIONALITIES

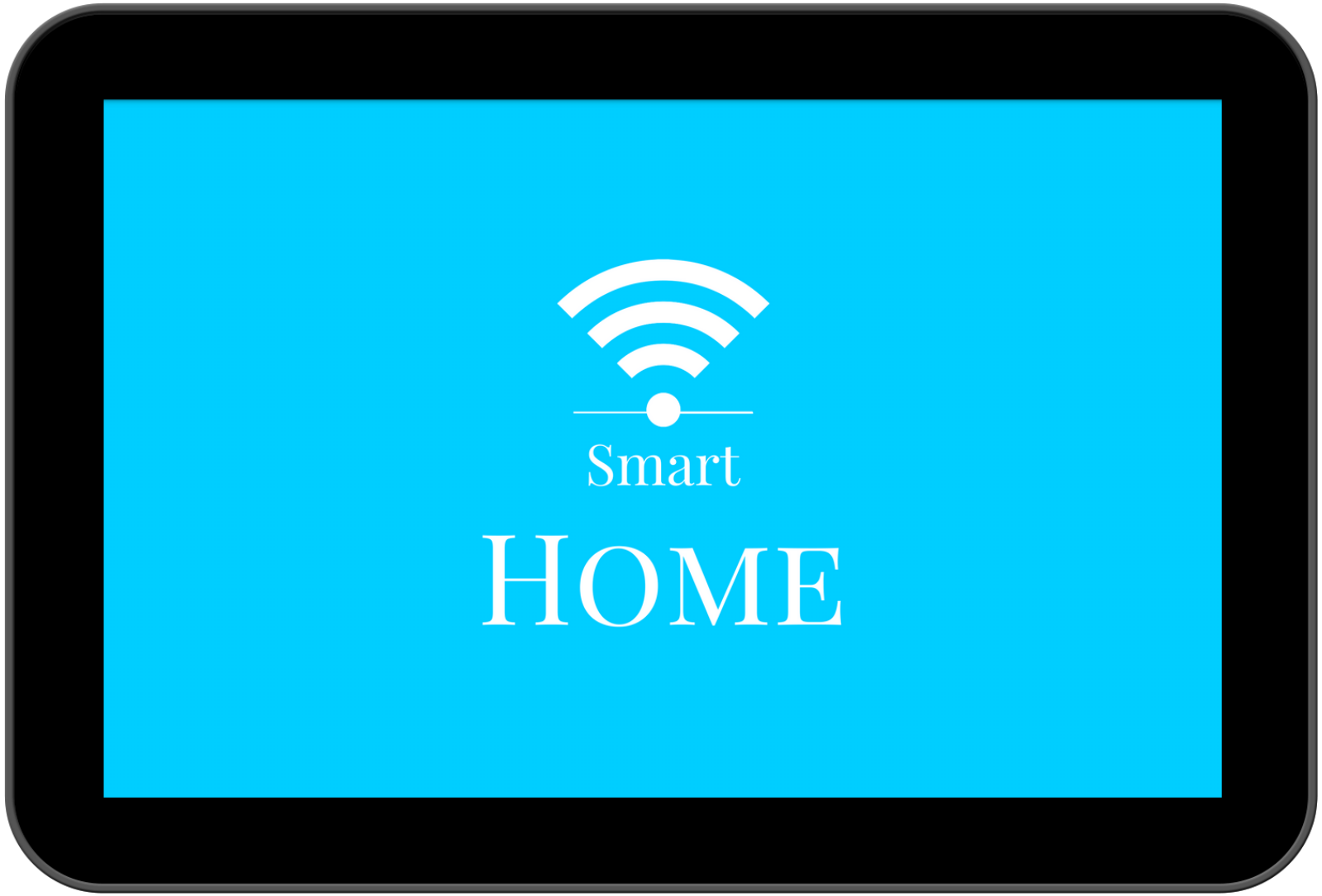
- SERVER
  - Return devices current states through RESTful APIs

- ANDROID APP
  - Login into the application after loading the XML.
  - See latest events
  - See all divisions in the house
  - Logout the application
  - See all devices
  - See devices per division
  - Login with remember me option

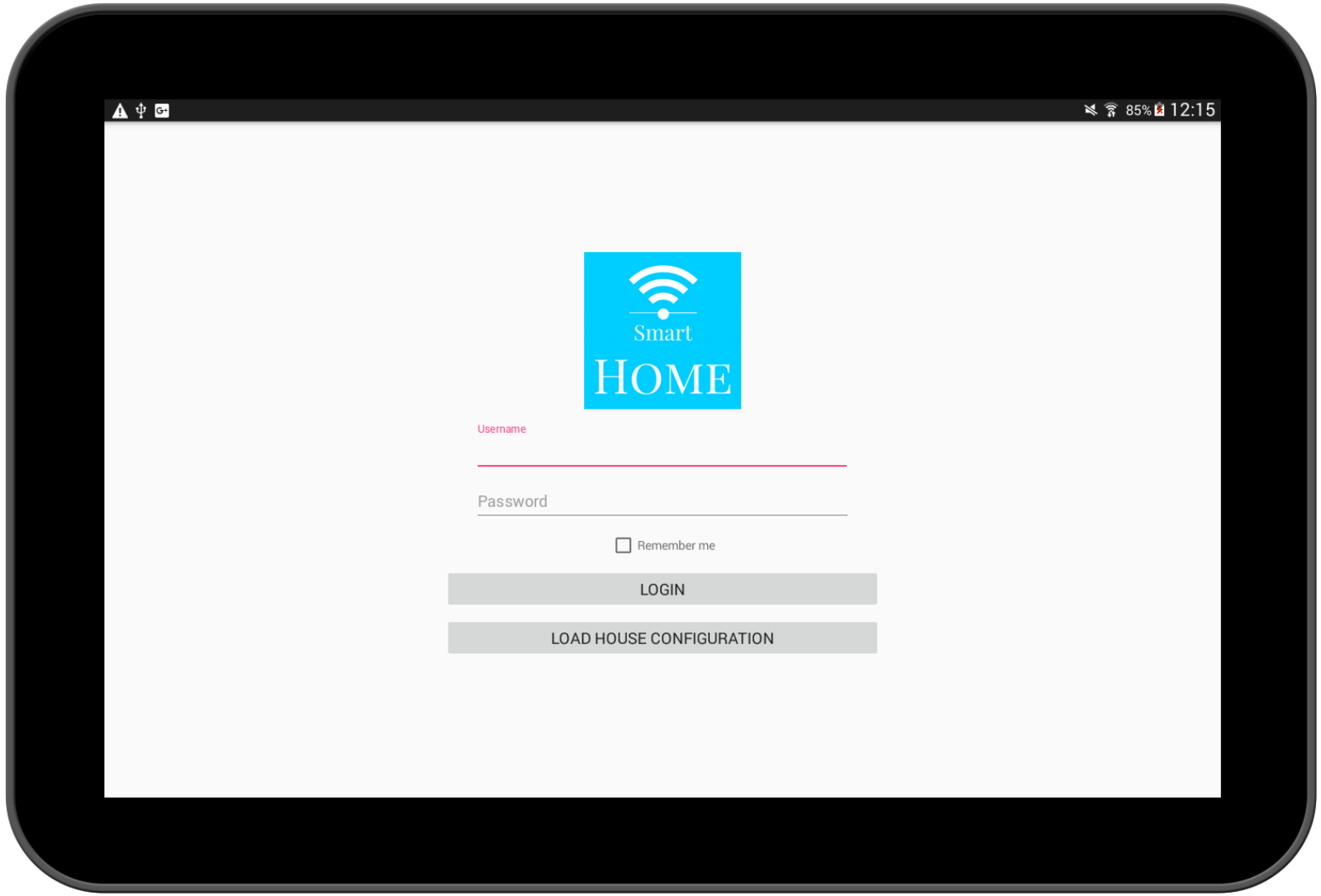


APPLICATION WIREFRAME

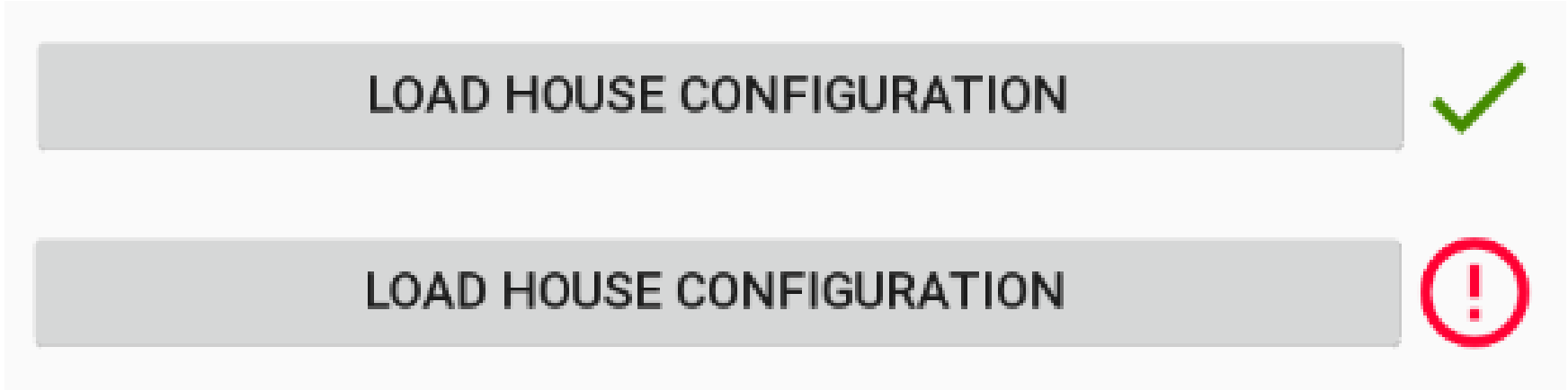
Splash



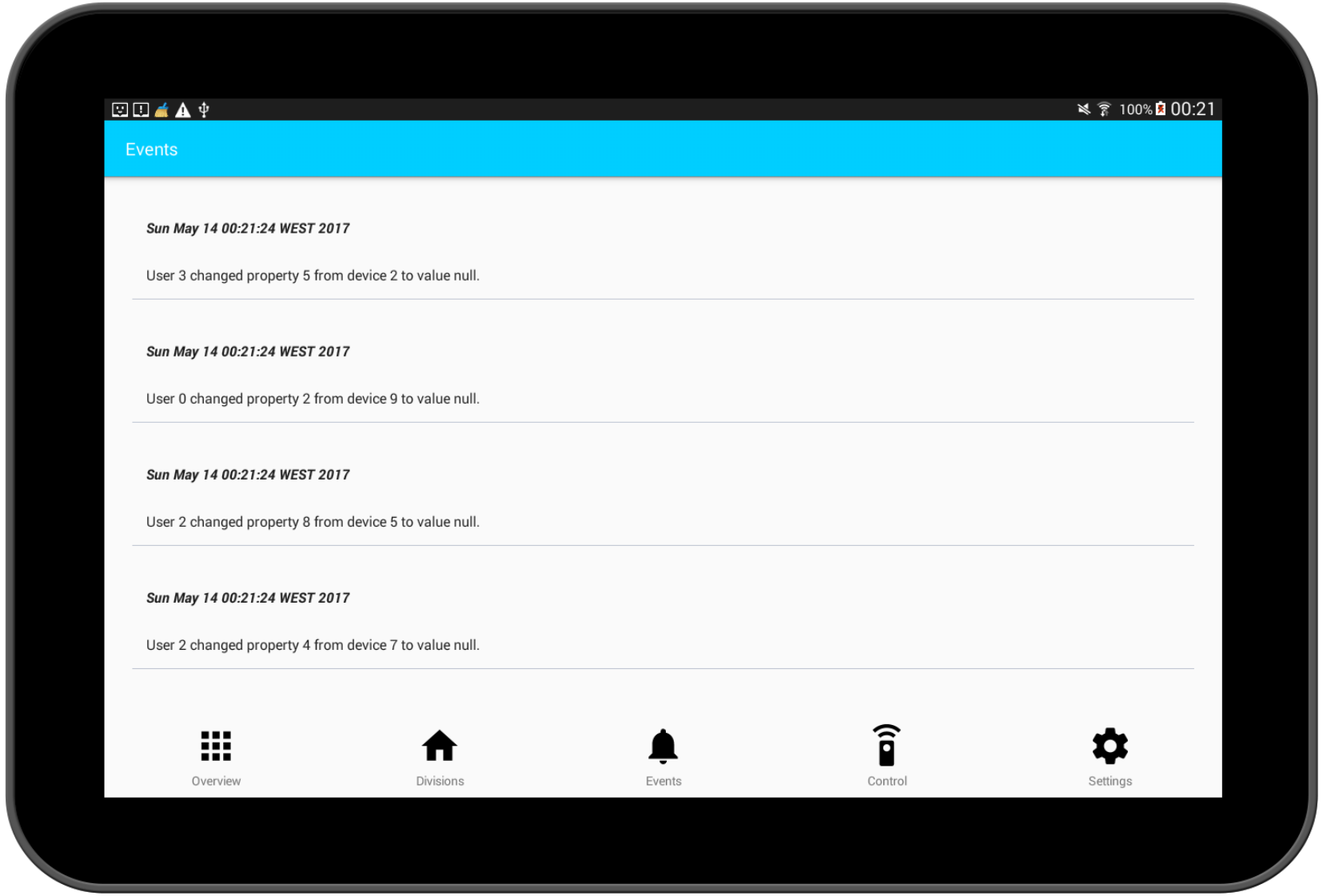
Login


85% 12:15
Smart
HOME
Username
Password
Remember me
LOGIN
LOAD HOUSE CONFIGURATION

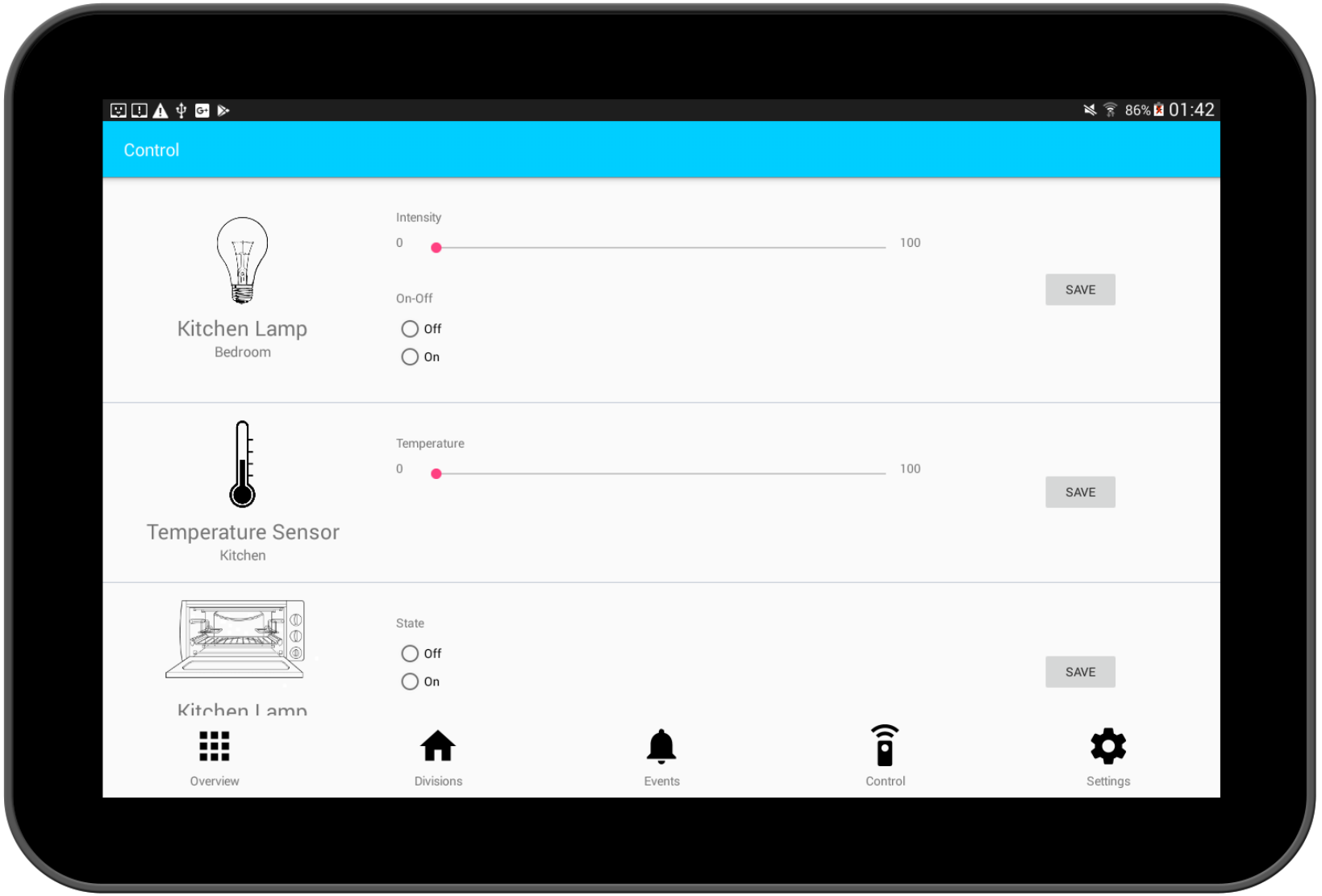
Login



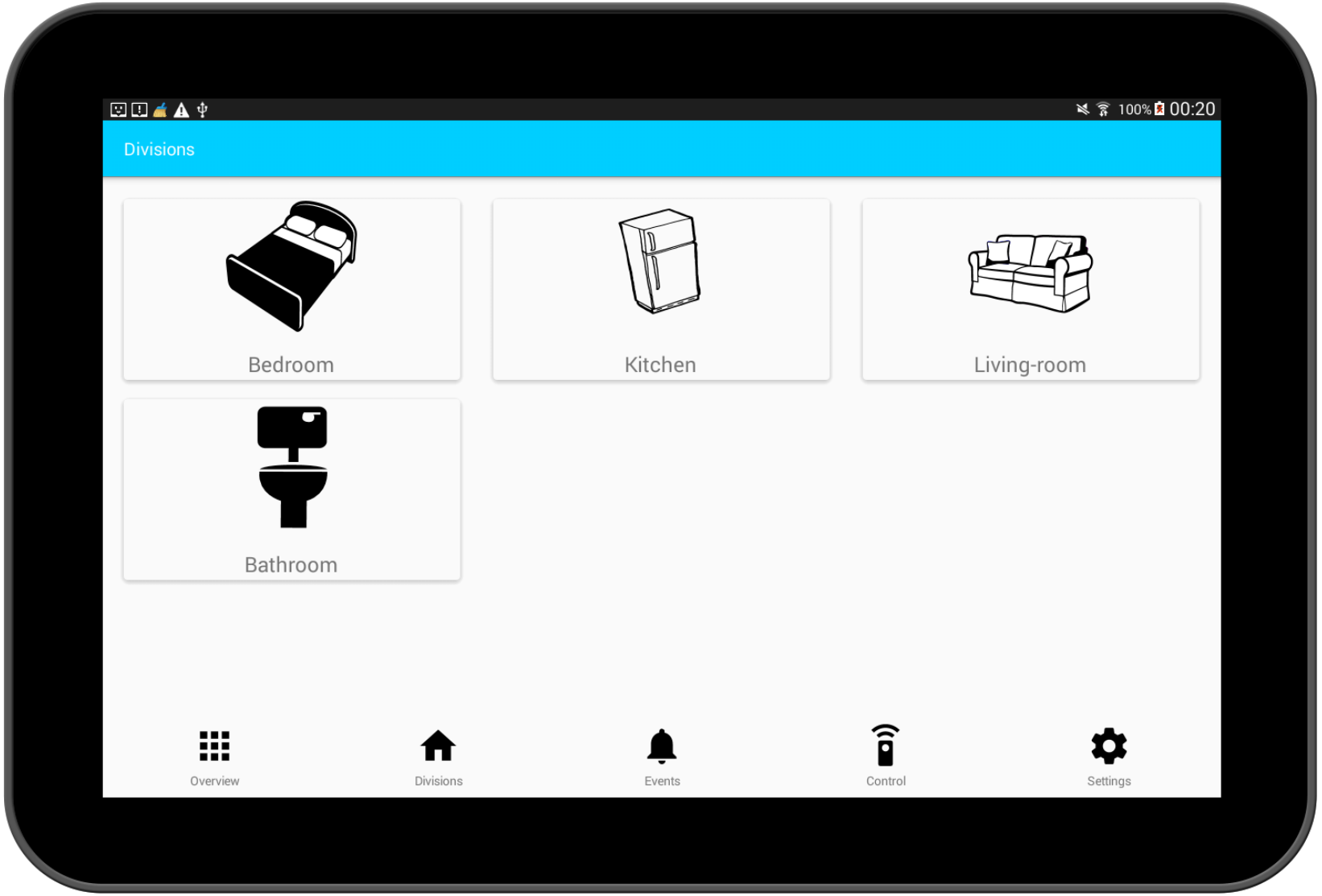
Events


100% 00:21
Events
Sun May 14 00:21:24 WEST 2017
User 3 changed property 5 from device 2 to value null.
Sun May 14 00:21:24 WEST 2017
User 0 changed property 2 from device 9 to value null.
Sun May 14 00:21:24 WEST 2017
User 2 changed property 8 from device 5 to value null.
Sun May 14 00:21:24 WEST 2017
User 2 changed property 4 from device 7 to value null.
Overview
Divisions
Events
Control
Settings

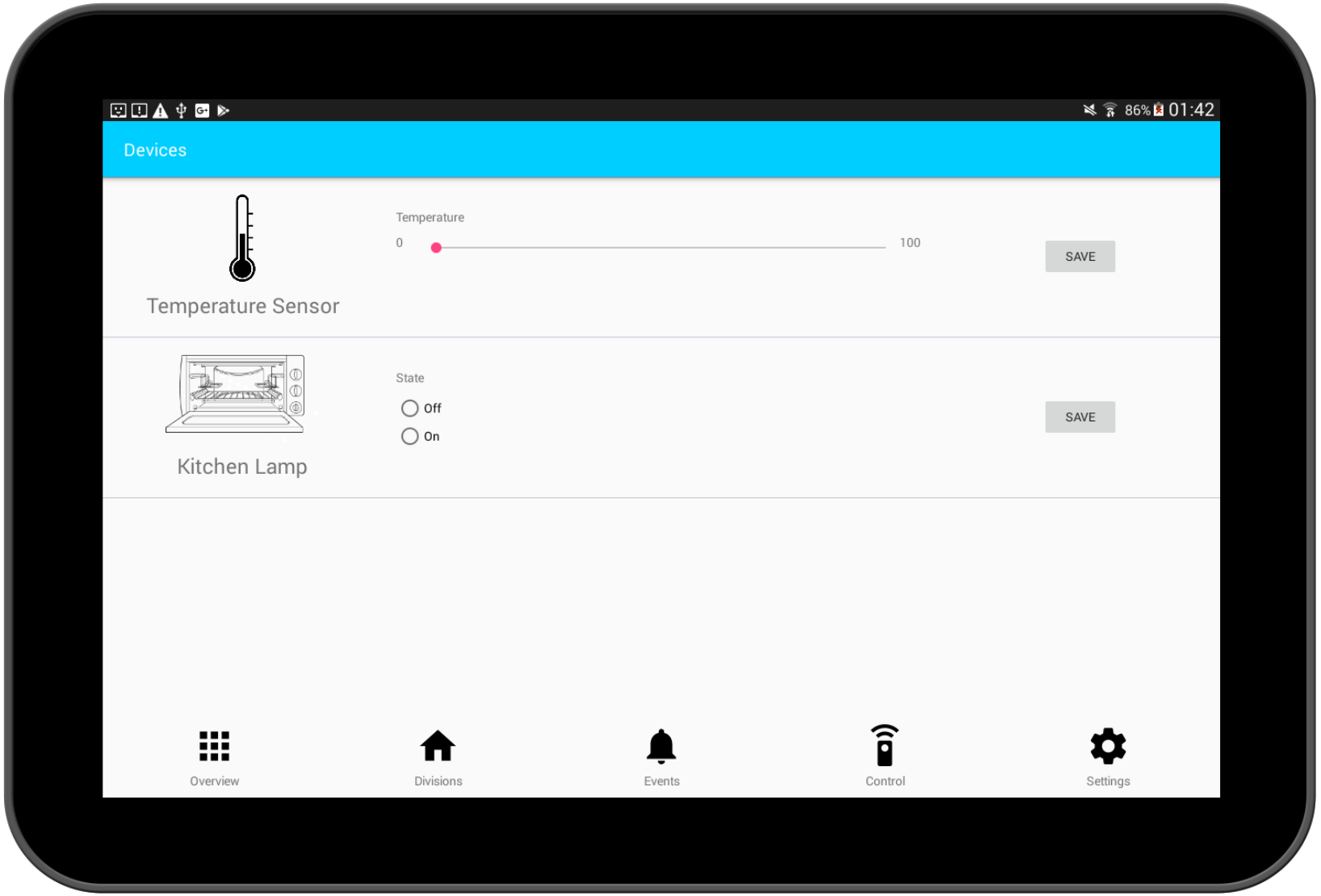
All Control


86% 01:42
Control
Intensity
0
100
On-Off
Off
On
SAVE
Kitchen Lamp
Bedroom
Temperature
0
100
SAVE
Temperature Sensor
Kitchen
State
Off
On
SAVE
Kitchen Lamp
Overview
Divisions
Events
Control
Settings

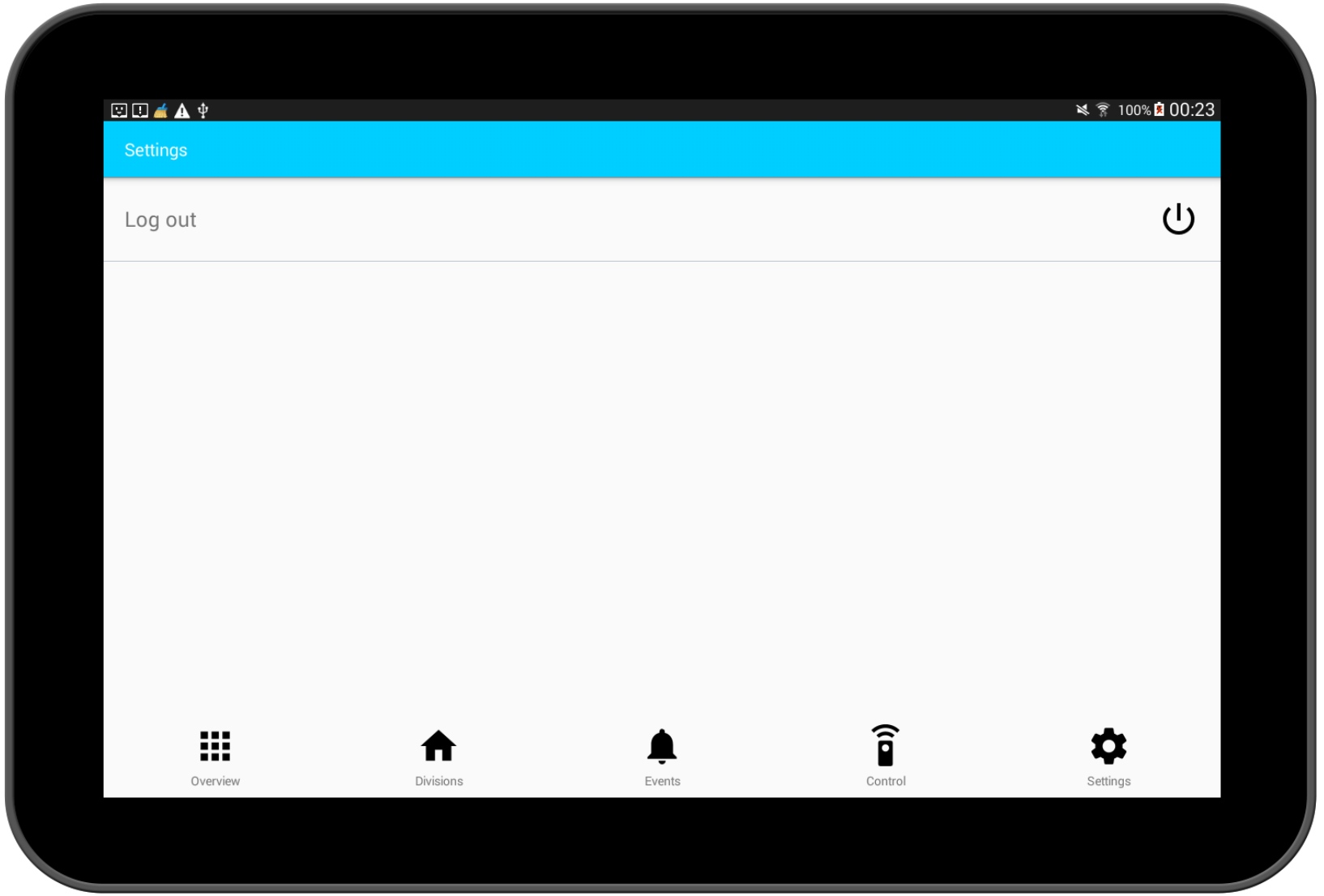
Divisions


100% 00:20
Divisions
Bedroom
Kitchen
Living-room
Bathroom
Overview
Divisions
Events
Control
Settings

Devices


86% 01:42
Devices
Temperature
0
100
SAVE
Temperature Sensor
State
Off
On
SAVE
Kitchen Lamp
Overview
Divisions
Events
Control
Settings

Settings


100% 00:23
Settings
Log out
Overview
Divisions
Events
Control
Settings

DEMONSTRATION – STEPS

- **What we need physically:**
  - Computer running Java
  - Android Tablet

- **What we need to test system:**
  - Change static IP Address in the application.
  - Valid DomoBus Configuration: *basic_config_1.xml*
  - Configure initial device values: *intial_values.json*

DEMO – DOMOBUS CONFIGURATION

- Added *DivisionType* to map types of divisions to icons
- *DeviceType* ID is mapped to a device icon
- Logistics:
  - 2 Users
  - 3 Devices
  - 4 Divisions
  - 2 Properties Type used

FUTURE WORK

- Real time statistics of the energy and water consumption (from server)
- Implement other services in server
- Initial Application Tutorial
- Allow the user to define general commands (E.g.: turn all light off)
- See an overview of the house state
- Define favorite commands
- Improve UX & UI Design

CONCLUSIONS

- The final app prpototype managed to:
  - Load house information from the XML configuration file.
  - Show divisions
  - Communicate with the Home Server
    - Show devices
- There is a lot room for improvement

QUESTIONS ?

Isabel Costa | 76394 | METI