Apartment complex communication application

Erman Yalçın

190706027

Content

1. Scenario
2. My app’s details
3. Activities
4. Some UML Diagrams
5. System Architecture
6. Database Tables
7. Output Screens

1) Scenario

In the Kardeşler Apartment complex, there are 8 apartments, Erman lives in one of these apartments and wants to know announcements daily and learn current events without looking at the papers hanging in the elevator, also he wants an easier way to communicate with his neighbors and ask the manager for an easier way to make a request or complaint, He also likes that it is an easy way to make an appointment to take advantage of the apartment services. So, after all these thoughts, he decides to make an application for the apartment complex and there are some features of this application.

Some Features:

User (resident), apartment manager, or security enters the application, selects their apartment from the application, and chooses the option that suits her or him (resident, manager, security).

Residents can see events, announcements, etc. They can request events and notify that they are joining events to the manager, and they also can send and take messages from the manager, security, and residents. Can make appointments for apartment services like pool, gym, etc.

Security can see events, announcements, etc. And can send and take messages from the manager, guards, and residents. They can arrange appointments for apartment services.

Managers can create and see events and announcements. Can write and take messages, can arrange appointments for apartment services.

2) My app’s details

* Server:
  + Save events, announcements, messages, apartment services
  + Save Manager, Residents, and Securities information
* Client:
  + Select your position (Manager, resident, guard)
  + Take information from the Server and send new information to Server.
  + Manager:
    - Can create events, announcements
    - Can write messages
    - Can arrange appointments for apartment services
  + Resident:
    - Can see events, announcements, messages
    - Can join events, write messages
    - Can make appointments for apartment services like pool, gym, etc.
  + Guard:
    - Can see events, announcements, messages
    - Can write messages
    - Can arrange appointments for apartment services

3) Activities:

Main Page: Choosing the user type

Registration Page: Create new user information from data and then redirects to login page

Login Page: Checks for user information from data and if there is a match it directs to the correct activation

Security Page: It is a page where security can use its privileges.

Manager Page: It is a page where the manager can use its privileges.

Resident Page: It is a page where residents can use its privileges.

Messages Page: It is a page that shows the messaging between people.

Events Page: It is a page showing events.

Diagram

Description automatically generated

Login or Registers

Select one of them

on-demand processing

Diagram

Description automatically generated

4) Some UML Diagrams:

* **Class Diagram**
* **Use Case Diagram**
* **Activity Diagram**
* **Sequence Diagram**
* **Communication Diagram**
* **Timing Diagram**

**Class Diagram**

**Diagram

Description automatically generated**

**Use Case Diagram**

Diagram, schematic

Description automatically generated

**Activity Diagram**

**Diagram

Description automatically generated**

**Sequence Diagram**

**Diagram

Description automatically generated**

**Communication Diagram**

**Diagram

Description automatically generated**

**Timing Diagram**

**Diagram

Description automatically generated**

1. System Architecture

Activities: Admin Entrance, IdAndPasswordControll, MainActivity, Manager, RegisterActivity, Resident, Security

Classes: SharedPrefManager, URLs, User, VolleySingleton

AdminEntrance.kt .

IdAndPasswordControll .

MainActivity .

Manager .

RegisterActivity .

Resident .

Security .

SharedPrefManager URLs User VolleySingleton

PHP files: login.php, register.php, pconnection.php

login:



register:



pconnection:



Android Manifest:



Build Gradle:



Data Base Photo:

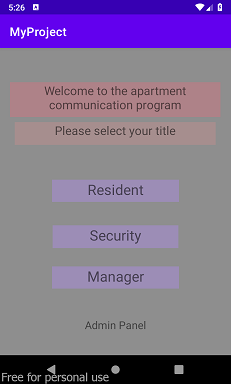
Graphical user interface, text, application

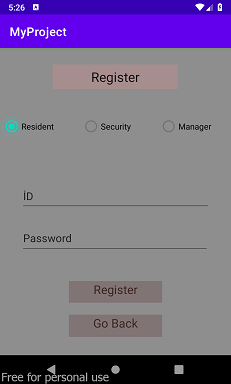
Description automatically generated

A screenshot of a computer

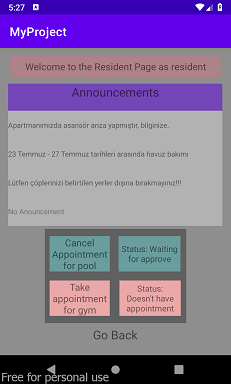
Description automatically generated

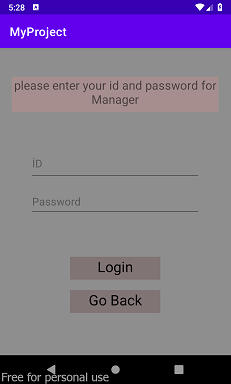
Emulator Photos:

Graphical user interface, application

Description automatically generatedTable

Description automatically generated with low confidenceGraphical user interface, application, Teams

Description automatically generatedTable

Description automatically generated with low confidence

Kotlin Project file:

