

Author: Frank Ekwomadu. C

Tech Stack: Kotlin, Firebase Realtime Database, Storage & Authentication

---

## EMPLOYEES WEEKLY SCHEDULE DOCUMENTATION

Completed this project by following the process:

1. Created a new application using “Basic Views Activity” on Android Studio.
2. I added and synced the following dependencies:
  - a. *Project Dependencies*:
    - id("com.google.gms.google-services") version "4.4.2" apply false
    - id("androidx.navigation.safeargs") version "2.8.3" apply false
  - b. *App Dependencies*:
    - id("com.google.gms.google-services")
    - id("androidx.navigation.safeargs")
    - implementation(libs.firebase.storage.ktx)
    - implementation(libs.firebase.auth.ktx)
    - implementation(platform("com.google.firebase:firebase-bom:33.5.0"))
    - implementation("com.google.firebase:firebase-database-ktx")
    - implementation("com.google.firebase:firebase-analytics")
    - implementation ("com.squareup.picasso:picasso:2.8")
3. Implemented my Project on [Firestore](#), downloaded the JSON file and uploaded it to my local project, that is, within Android Studio. In Firestore, I created:
  - a. Real Time Database
  - b. Authentication
  - c. Storage
4. I created / modified the:
  - a. [Contacts.kt file](#):
    - This file defines the data model for contacts, including properties such as name, phone number, work schedule and image URI, allowing for structured data representation.
  - b. [RvContactsAdapter.kt file](#):
    - This adapter facilitates the binding of contact data to the RecyclerView, managing the display and interaction of contact items in the user interface.
  - c. [MainActivity.kt & activity\\_main.xml files](#):
    - I integrated navigation components and set up the main layout structure to support seamless transitions between fragments within the application.
  - d. [LoginFragment & fragment\\_login.xml file](#):

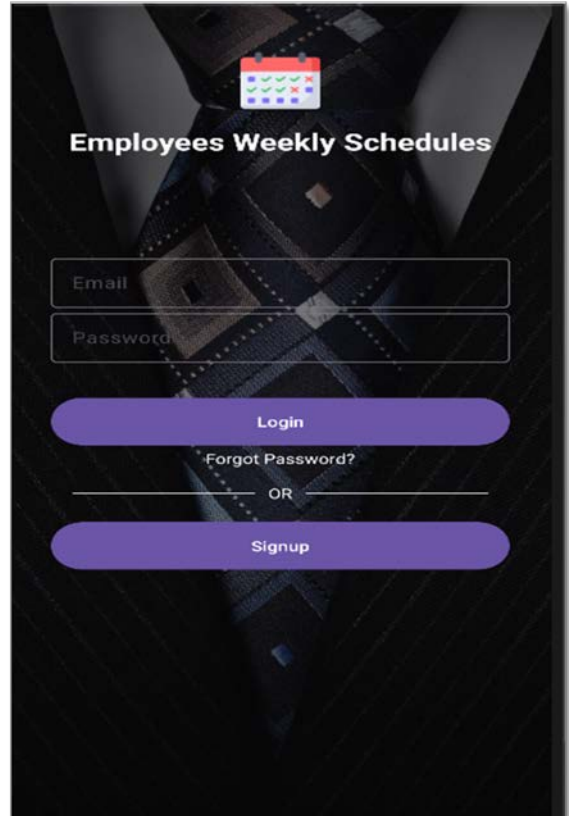
- I implemented the user authentication interface, allowing users to log in by entering their credentials and navigating to the appropriate screens upon successful login.
- e. SignupFragment & fragment\_signup.xml files:
  - This fragment enables new users to register by providing their details, with validation to ensure all fields are completed before submission.
- f. ForgotPasswordFragment & fragment\_forgot\_password.xml files:
  - I created a user-friendly interface for password recovery, guiding users through the process of resetting their passwords via email.
- g. UpdatePasswordFragment & fragment\_update\_password files:
  - This fragment provides functionality for users to update their passwords, ensuring a secure and straightforward process for maintaining account security.
- h. HomeFragment & fragment\_home files:
  - I developed the main interface for displaying contacts, integrating functionality to fetch, display, and manage contact information effectively.
- i. AddFragment & fragment\_add files:
  - This fragment allows users to add new contacts, providing a simple form for entering contact details and saving them to the database.
- j. UpdateFragment & fragment\_update files:
  - I implemented functionality for users to edit existing contacts, allowing for updates to information like name, phone number, work schedule, and image URI.
- k. nav-graph file:
  - I configured the navigation graph to manage the transitions between various fragments, ensuring a logical flow within the application.
- l. drawable folder:
  - Here, I added my images used within this project.
- m. strings.xml:
  - This file contains all static text elements used within the project, centralizing string management for easier localization and maintenance.


### Project Demo:

Current Username: [test@gmail.com](mailto:test@gmail.com)

Current Password: **12345678**

1) Open Application (if you don't have a login credential, click on 'Signup' button to create one else use the credential provided above)



  
**Employees Weekly Schedules**

Email

Password

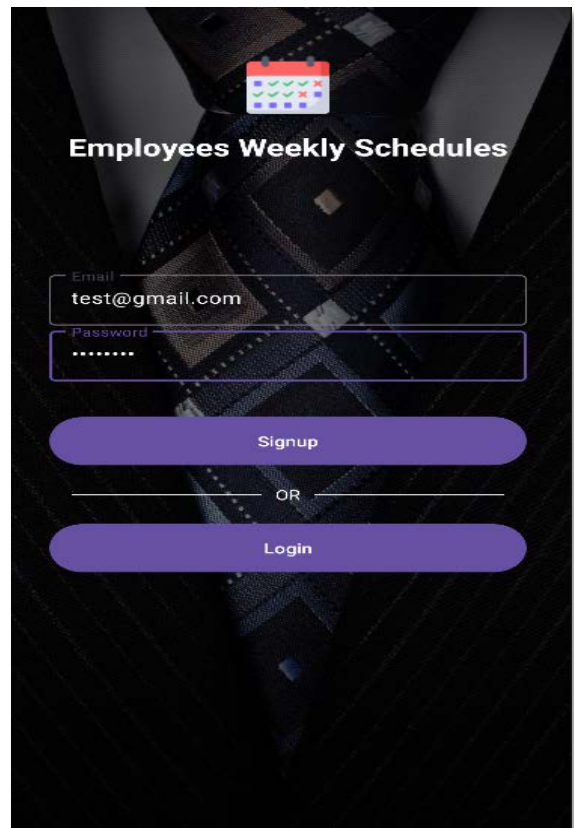
Login


[Forgot Password?](#)

OR

Signup

2) Enter your VALID email address and a password, then click on 'Login' (or 'Signup' button if you're on the signup page)



  
**Employees Weekly Schedules**

Email

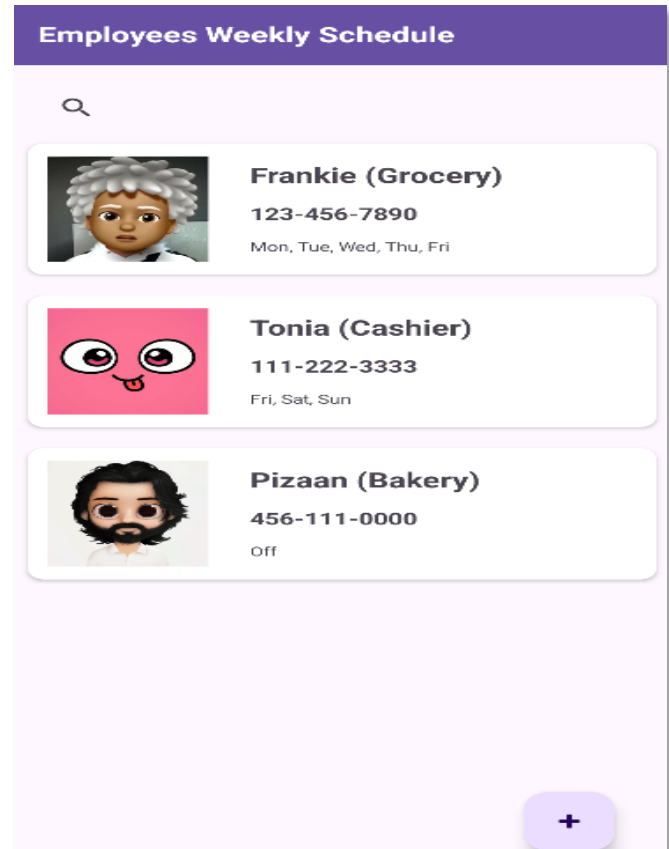
Password

Signup

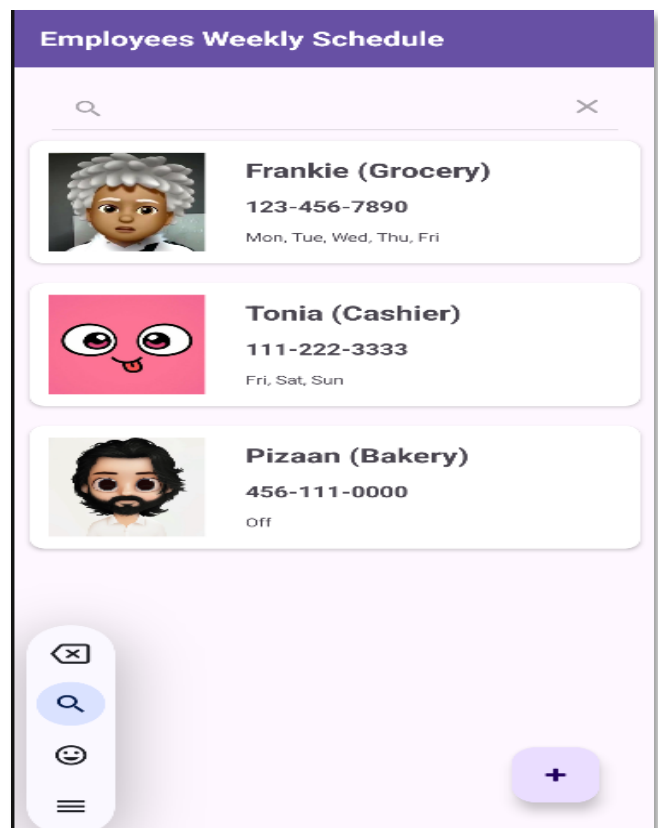
OR

Login

3) Verify that contacts are displayed on the Home Page




4) Click on the search icon to display a search bar




5) Enter a 3-letter day of the week or an employee's name to bring up related items.

### Employees Weekly Schedule

fri



**Frankie (Grocery)**  
123-456-7890  
Mon, Tue, Wed, Thu, Fri



**Tonia (Cashier)**  
111-222-3333  
Fri, Sat, Sun

✕

🔍

😊

☰

+

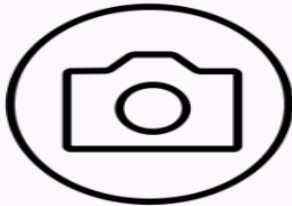
6) Click on the '+' button and verify that an Add Page is displayed

### Add Employee's Schedule

Name

Phone (ex. 123-456-7890)

Schedule (ex. Mon, Tue, Fri, Sat)



pick image

Send data


7) Enter a Name, Phone Number, Work Schedule and an Image, then click on 'Send data' button

### Add Employee's Schedule

Ronald (Store Manager)|

223-223-2233

Mon, Tue, Wed, Thu, Fri, Sat, Sun





pick image

Send data


8) Verify that you're returned to the Home Page and the new contact is visible

### Employees Weekly Schedule







**Frankie (Grocery)**  
123-456-7890  
Mon, Tue, Wed, Thu, Fri



**Tonia (Cashier)**  
111-222-3333  
Fri, Sat, Sun



**Pizaan (Bakery)**  
456-111-0000  
Off




**Ronald (Store Manager)**  
223-223-2233  
Mon, Tue, Wed, Thu, Fri, Sat, Sun

Image stored successfully

+

9) Click on the created contact and verify that the Update Page opens

**Update Employee's Schedule**



Ronald (Store Manager)


223-223-2233

Mon, Tue, Wed, Thu, Fri, Sat, Sun

Update data

10) Change phone to “000-000-0000” and click 'Update data' button

**Update Employee's Schedule**



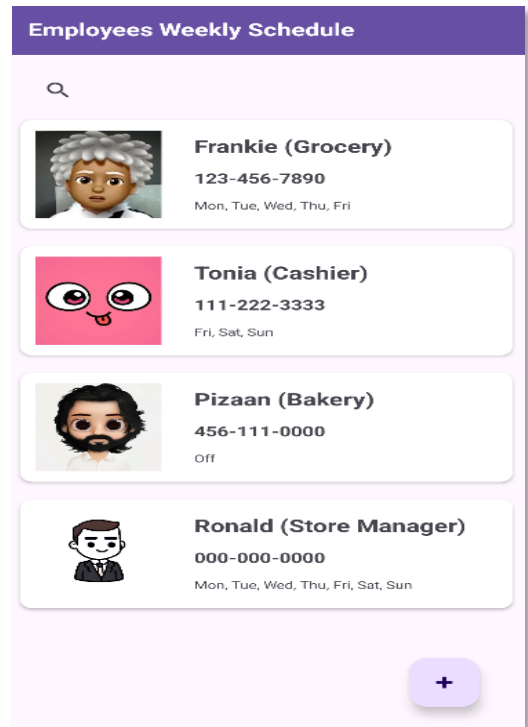
Ronald (Store Manager)

000-000-0000

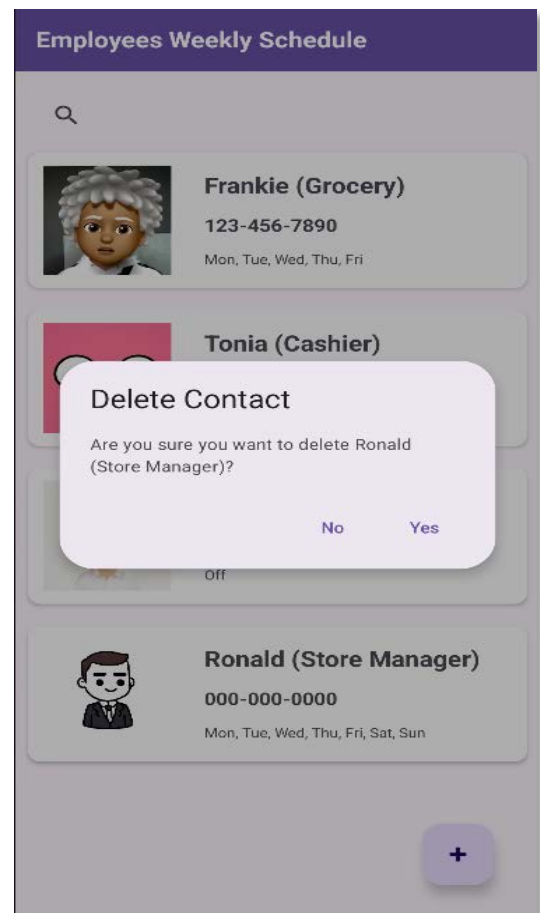
Mon, Tue, Wed, Thu, Fri, Sat, Sun

Update data

11) Verify that you're returned to the Home Page and the new contact is updated



12) Long-click on the created contact and verify a 'Delete' dialog box is shown





13) Click on 'Yes' and verify that the contact is no longer on the Home page.

