

Project Diary

Group: Simon, Emilia, Victor, Elin, Daniel, Youssef

Name: Youssef Laamim

Introduction:

Our group decided to work on the "**Grannar**" app, which aims to facilitate communication between neighbors in a specific area. The app allows users to create groups based on shared interests, chat with neighbors, and search for people with similar interests. We used tools like **Miro**, **Figma**, and **GitHub** to manage the project.

Daily Entries

Monday, 3/2

Morning Meeting (9:00 - 9:30):

- **Role Assignment:** We started the day by assigning roles within the team. A **Scrum Master** and **Product Owner** were selected.
 - **Tool Setup:** We created a repository on **GitHub** and set up a **Miro board** to track tasks.
 - **Figma Account:** A Figma account was created and linked to the group members' accounts.
 - **Meeting Schedule:** We determined the days we would meet, the daily Scrum, and communication channels.
-

Tuesday, 4/2

Morning Meeting (9:00 - 9:30):

- **Writing User Stories:** We wrote **User Stories** to clearly define the app's requirements.
 - **Design Assistance:** I helped design the **login**, **registration**, and **chat interfaces** on Figma.
-

Wednesday, 5/2

Morning Meeting (9:00 - 9:30):

- **Continuing Design:** I continued working on the design in Figma.
 - **Writing User Stories:** We wrote additional User Stories.
-

Thursday, 6/2

Morning Meeting (9:00 - 9:30):

- **Meeting with the Teacher (14:15 - 15:00):** We met with the teacher to present the designs and User Stories. The teacher provided feedback on improving the design and linking the UI with **Firebase**.
 - **Continuing Design:** I continued working on the design in Figma.
 - **Writing User Stories:** We wrote additional User Stories.
-

Friday, 7/2

Morning Meeting (9:00 - 9:30):

- **Cloning the Project from GitHub:** I cloned the initial project from GitHub.
 - **Fixing Android Studio Issues:** I resolved issues related to updating the Android version to match the project and fixing problems in the Gradle file.
-

Saturday, 8/2

Day off.

Sunday, 9/2

Day off.

Monday, 10/2

Morning Meeting (9:00 - 9:30):

- **Starting Task Implementation:** I began working on the first tasks for the project, focusing on the **chat functionality**:
 - Writing code for a functional chat.
-

Tuesday, 11/2

Morning Meeting (9:00 - 9:30):

- **Continuing Code Writing and Testing:** I continued writing and testing the code for the chat functionality.
-

Wednesday, 12/2

Morning Meeting (9:00 - 9:30):

- **Starting UI Design for Chat:** I began designing the chat interface by creating the **fragment_chat.xml** file for the main chat screen. I faced some difficulties in arranging the elements correctly, but after experimenting with different layouts, I achieved the desired design.
-

Thursday, 13/2

Morning Meeting (9:00 - 9:30):

- **Continuing UI Work:** I created the **fragment_home_chatt.xml** file for the main groups screen.
 - **Meeting with the Teacher (14:15 - 15:00):** We met with the teacher to present the progress. The teacher provided feedback on improving the design.
 - **Starting ChatMessage.kt Class:** I started writing the **ChatMessage.kt** class to store message data. I encountered issues with binding the data to the UI but resolved them using **RecyclerView**.
-

Friday, 14/2

Morning Meeting (9:00 - 9:30):

- **Starting the Group.kt Class:** I began working on the **Group.kt** class to store group data.
 - **Starting ChatAdapter.kt:** I started writing **ChatAdapter.kt** to bind data to the UI. I faced issues with displaying messages correctly but fixed them after debugging.
-

Saturday, 15/2

Day off.

Sunday, 16/2

Day off.

Monday, 17/2

Morning Meeting (9:00 - 9:30):

Continuing Work on GroupAdapter.kt: I continued working on **GroupAdapter.kt** to bind group data to the UI.

Implementing Message Sending and Receiving: I started implementing the function to receive messages. I faced issues with real-time message updates but resolved them using **addSnapshotListener**.

Tuesday, 18/2

Morning Meeting (9:00 - 9:30):

- **Implementing Group Search by Tags:** I started working on the group search functionality using tags.
-

Wednesday, 19/2

(Various tasks related to UI improvements, Firebase integration, and new group functionalities.)

Thursday, 20/2 till Friday, 1/3

Continuing Work

(Various tasks related to UI improvements, Firebase integration, and new group functionalities.)

group search functionality using tags.

function to receive messages. I faced issues with real-time message updates but resolved them using **addSnapshotListener**.

Tuesday, 4/3

Morning Meeting (9:00 - 9:30):

- **Updated Functions:**
 - Updated **fun updateProfile**.
 - Implemented function for users to send requests to the admin to join a group before becoming a member.
 - Added **JoinRequest** data.
 - Added notification icon.
 - Updated **addNewGroup** function with admin role.
 - Updated **item_group** design.
-

Wednesday, 5/3

Morning Meeting (9:00 - 9:30):

- Implemented function for admin to delete groups from chat and Firebase.
- Added **fun delete** in **FirestoreManager**.
- Implemented function to allow users to leave groups.
- Added a functionality that restricts group creation to registered users only.
- Implemented a feature where the group admin can approve or deny membership requests.

- Improved group search functionality by allowing users to filter by location.
-

Thursday, 6/3

Morning Meeting (9:00 - 9:30):

- Implemented function for admin to delete groups from chat and Firebase.
 - Added **fun delete** in **FirestoreManager**.
 - Added delete icon (red color).
 - Implemented swipe-to-delete for groups.
 - Added delete icon (red color).
-

Reflections

What Did I Learn from the Project?

- I learned how to use **Firebase** extensively, especially **Firestore** and **Authentication**.
- I also learned how to handle common issues like real-time data updates and managing state in **RecyclerView**.

What Went Well?

- Task division among team members was effective and helped us complete work quickly.
- Using **Firebase** simplified a lot of the work in managing data and users.
- Good communication within the team helped solve problems quickly.

What Could Be Improved?

- Next time, I will try writing **unit tests** to ensure code quality.
- The app's performance can be improved by optimizing **Firestore queries**.
- More features like **file sharing** or **private groups** could be added.

What Do I Want to Highlight?

- I want to highlight solving the issue of **real-time message updates**. The problem was that messages weren't updating the UI automatically. After research, we resolved it using **addSnapshotListener**, making the app run smoothly.
-

Attachments/Links

- **GitHub Repository Link:** <https://github.com/Daffnel/Grannar/tree/main>

- **Figma Link:**
<https://www.figma.com/design/RsobpCfVE3mihKi17l4Dbu/Grannar?node-id=0-1&p=f&t=S410xgODjw5MXutH-0>
- **Miro Link:** [Miro Link]
- **Firebase Link:**
<https://console.firebase.google.com/project/neighbors-d5393/overview>