

**CS102**

**Fall 2023/24**

Instructor:

**Uğur Gündükbay**

Project  
Group

**1C**

Assistant:

**Sinan Sonlu**

Criteria	TA/Grader	Instructor
Presentation		
Overall		

## Bilkent Connect

Sensin

**Başar Çubukçuoğlu & Alperen Karslıoğlu  
& Halil Arda Özongun & Yunus Emre Erkan**

### Detailed Design Report

( version 2.0)

**23 December 2023**

## Introduction

Bilkent Connect is a multifunctional social media Android application that aims to satisfy the social needs of a Bilkent student. The typical social media applications usually do not focus on a specific community. Therefore, they cannot be entirely suitable for these specific communities. Bilkent Connect provides a social media platform exclusive to Bilkent University students. Hence, it has diverse features that are designed specifically for

Bilkent students. The main feature of Bilkent Connect is sharing posts with other students, whether anonymously or not, in order to create an accessible environment where students can share their ideas. The second feature is a second-hand sale page where students can buy and sell second-hand school-related items. The third feature is the BBC Cafeteria page, where students can check that day's cafeteria meal and rate or comment on that meal to interact with other students. With all of these features, Bilkent Connect offers a unique experience that is specifically designed for Bilkent students by Bilkent students.

## System Overview

### 1. Frontend

#### 1.1. Android Studio

The development of the user interface and front-end for the Bilkent Connect Android social media application will be carried out using Android Studio, a powerful integrated development environment (IDE) specifically designed for Android application development. Android Studio offers an extensive set of tools and resources, empowering us to craft an intuitive and visually appealing user interface that aligns with the branding and user experience goals of Bilkent Connect. Our primary focus will be on utilizing Android's native UI components, including layouts and views, to design screens showcasing various functionalities such as posts, comments, profile pages, feature menus, second-hand sale products, and daily meal information. We aim to customize these components to enhance the overall engagement of users of our app. The layout editor in Android Studio will be employed to visually design XML-based layout files, allowing us to define the structure and positioning of UI elements. Furthermore, we plan to integrate appropriate animations and transitions to enhance the visual experience and provide responsive feedback to users. Leveraging Android Studio's capabilities for user interface and front-end development ensures that we can capitalize on powerful tools and resources, resulting in the creation of a high-quality application developed for the specific social media needs of Bilkent students.

## **1.2. Firebase**

Firebase is a popular mobile and web application development platform offered by Google. We plan to utilize the Authentication service in Firebase in the frontend of our application.

### **1.2.1. Firebase Authentication**

Firebase Authentication is a system that makes it easy for users to log in with an e-mail, password, and phone number. In addition, Firebase provides convenience for users and developers to log in with third-party identity providers such as Google, Facebook, and Github. We will use Firebase Authentication to build a secure login system in our application. Firebase provides a user and developer-friendly authentication system.

## **2. Data**

### **2.1. Firebase**

#### **2.1.1. Cloud Firestore**

Cloud Firestore is a serverless NoSQL database service that offers high scalability and flexibility. Data is organized as documents and collection and customize them according to our needs. In particular, Cloud Firestore, which is ideal for real-time applications, can instantly listen to data changes in the Bilkent Connect application and can be automatically reflected in the application interface. On the other hand, firestore can limit access to data with security rules. Thus, we can ensure that users only access the data they are authorized to access. Another feature that makes Firestore more unique for Bilkent Connent is that it can work integrated with other services of Firebase (for example, Authentication and Storage).

#### **2.1.2. Cloud Storage:**

Cloud Storage is designed for the storage and management of large files. It is especially suitable for storing media files such as images and videos uploaded by users. In this way, we will keep the photo or video parts of the posts posted by users in this way. Also, we can keep other kinds of photos or videos, such as advertisement photos, profile photos, etc. Another feature that makes Cloud Storage

unique is that file upload and download processes are fast and efficient thanks to Firebase's powerful infrastructure, which will make our application work as efficiently as possible. Also, flexible permissions and security rules can be defined to manage file access in Cloud Storage. Thus, users can be ensured access to only the content they are authorized to.

#### **2.1.3. Firebase Analytics:**

Firebase Analytics gives us insight into how our application is being used. This analytics service provides a wide range of data, from which features are used more to user demographics and in-app behavior patterns. We can use this information to improve the user experience.

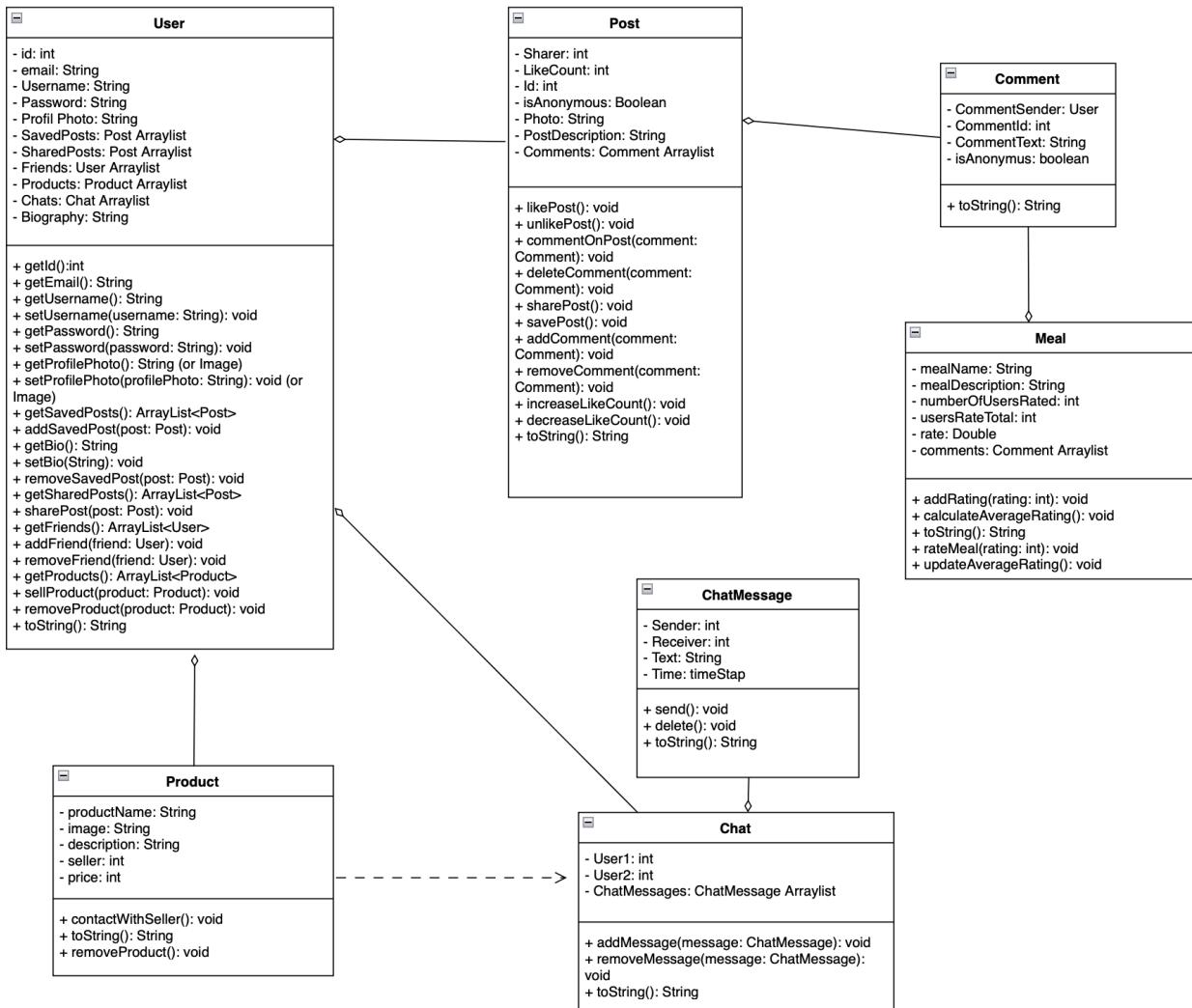
### **2.2. Jsoup**

In our project, we utilized web scraping techniques to gather real-time information from the Bilkent Cafeteria website. The goal was to retrieve dynamic data such as daily menus and meal content. Using Jsoup, we navigated through the HTML structure of the Bilkent Cafeteria website, targeting specific elements that contained relevant information. These elements included daily menus and meal descriptions. After retrieving the target HTML parts from the website, we use Jsoup for parsing the HTML code and convert it to a string in order to make it usable for our application.

### **2.3. Picasso**

Picasso is a popular image loading library in Android development, created to simplify the process of loading images from external sources and displaying them in your application. It's particularly useful in applications where displaying images from the internet or storage is a common task, as is likely in a social media application like Bilkent Connect.

# Core Design Details



## 1. Model Classes and Interfaces

User is a class representing users using the app, storing their ID, name, email, biography, profile photo, and password. User class will also store the information, including saved posts, shared posts, friends, products that the user is currently selling, and private chats with other users. This class will have a composition relationship with the classes, Post, and Chat. Besides, this class will have an aggregation relationship with the class Products.

Product class will represent an item listed by a producer, storing the name, the producer that owns the product, an example image of the product, the price, and a description of the product. This class will have a dependency relationship with Chat class.

The Chat class will represent a conservation between two users. It will include two User objects and ChatMessages objects. An object of this class will be created every time one user starts making new conversations with different users.

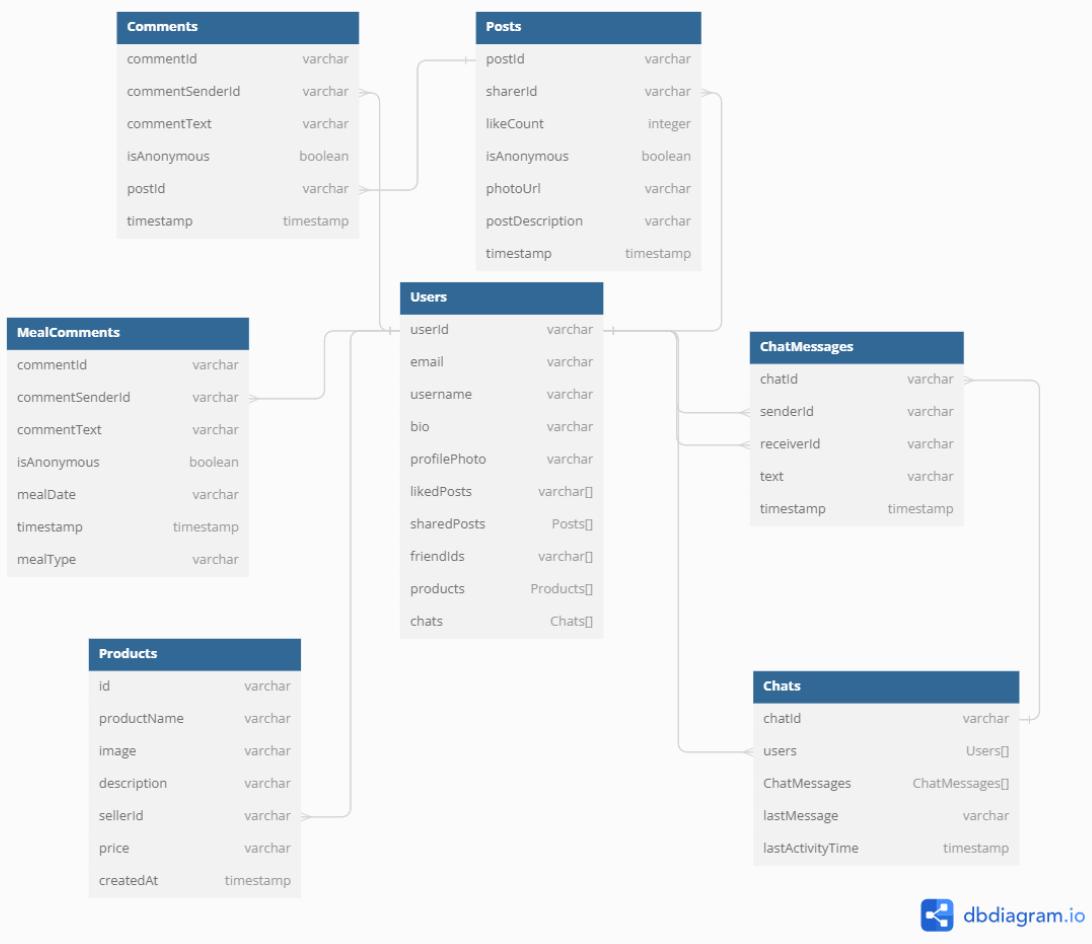
The ChatMessage class will represent a single message between two users. It will store the sender and receiver user ID, the message written by the sender, and the time when that single message was written.

The Comment class is structured to encapsulate created comments. It is configured to store a User object (representing the sender user), a comment ID for convenient database retrieval, the text of the comment itself, and an indicator demonstrating whether the comment was made anonymously.

Meal class will represent daily meal information by storing the names of the meals and information about their content. This class also stores the rates and comments of the users about the meals. This class will have an aggregation relationship with the Comment class.

The Post class will represent a post. It will store the sharer of the post, count of likes, ID of the post for having an organized database, boolean data of whether the post is anonymous or not, photo, description of the post, and comments

## 2. Database Schema



This SQL schema is designed to manage the functionalities of Bilkent Connect. The schema comprises various tables including Users, Posts, Comments, Products, Chats, ChatMessages, MealComments.

**Users:** This table is central to user-related interactions within the app. It stores information about users such as id, email, username, bio, and profile photo. Additionally, it now includes arrays for liked posts, shared posts, friends' IDs, products, and chats, reflecting a user's activities and connections in the app.

**Posts:** This table contains details of user-generated posts. Each record includes the post id, the id of the user who shared it (linked to the Users table), like count, anonymity status, photo URL, post description, and timestamp.

**Comments:** Used for storing comments on posts, including the comment's id, the id of the sender (linked to Users), comment text, anonymity status, and timestamp.

**Chats:** Manages chat sessions between users, with each record containing a chat id, participants' ids (linked to Users), an array of chat messages, the last message, and the last activity timestamp.

**ChatMessages:** Stores individual messages within chat sessions, including the chat id (linked to Chats), sender id, receiver id (both linked to Users), message text, and timestamp.

**MealComments:** Stores comments on meal posts, with fields for comment id, commenter's id (linked to Users), comment text, anonymity status, meal date, timestamp, and meal type.

**Products:** Catalogs products available in the app, with details such as product id, name, image, description, seller's id (linked to Users), price, and creation timestamp.

## Task assignment

UI Design	Başar Çubukçuoğlu & Alperen Karslıoğlu & Halil Arda Özongun & Yunus Emre Erkan
Frontend	Başar Çubukçuoğlu & Alperen Karslıoğlu & Halil Arda Özongun
Database	Halil Arda Özongun
Web Scraping	Yunus Emre Erkan

# In App Screenshots

The first screenshot shows the "Meal Comments" screen with a header "12:52". It displays comments for "Öğlen Yemeği/Lunch" from users "basarc" and "Ghost". The second screenshot shows the "BCC Cafeteria" screen with a header "12:51". It displays the menu for "Öğle Yemeği/Lunch" and nutritional information. The third screenshot shows the "Bilkent Connect" screen with a header "12:46". It displays a profile picture of a group of people, the user's name "basarc", and their email "basar.cubukcuoglu@ug.bilkent.edu.tr". A "new bio" placeholder is visible.

**Meal Comments**

Öğlen Yemeği/Lunch

**basarc**  
today's lunch was delicious

**Ghost**  
No way! There was a worm on my salad!

**BCC Cafeteria**

Öğle Yemeği / Lunch Minestrone Çorba / Minestrone soup Olimpiyat Köfte (Dilim Patates) / Meatballs topped with cheese and vegetables served with fried potatoes veya / or Vegan Fırın Patates / Baked potatoes (Vegan) Amasra Salata / Amasra salad Muhallebili Cezerye / Carrot and hazelnut dessert with vanilla pudding

Information about the meal

Enerji (k.cal.) / Energy (Cal.): 1046  
Karbonhidrat / Carbohydrate: %54 Protein / Protein: %15 Yağ / Fat: %31

**Akşam Yemeği/Dinner**

Akşam Yemeği / Dinner Misir Çorba / Corn soup Avcı Usulü Tavuk / Roasted chicken served with vegetables veya / or Vegan Mantar Sote / Sautéed mushrooms (Vegan) Arpa Şehriye Pilavi / Orzo Ayran / Ayran

Information about the meal

Öğle Yemeği / Lunch Mısnaslı Çorba / Mung bean soup Karışık Sebze Gratin / Vegetable gratin veya / or Vegan Sebze Kavurma / Braised vegetables (Vegan)

**Bilkent Connect**

basarc

basar.cubukcuoglu@ug.bilkent.edu.tr

new bio

**Edit Profile**

**Your Posts**

**Liked Posts**

**Security**

**Log Out**

The first screenshot shows the "Edit Profile" screen with a placeholder profile picture and sections for "Edit Username" and "Edit Biography". The second screenshot shows the "Security" screen with a "Change Password" button and fields for "Current Password", "New Password", and "Confirm New Password". The third screenshot shows the "Your Posts" screen displaying a post by user "Ghost" with a photo, like and comment icons, and the caption "same guys but private". Another post by "Ghost" is partially visible below it.

**Edit Profile**

**Edit Username**

New Username

**Edit Biography**

New Biography

**Change Password**

Current Password

New Password

Confirm New Password

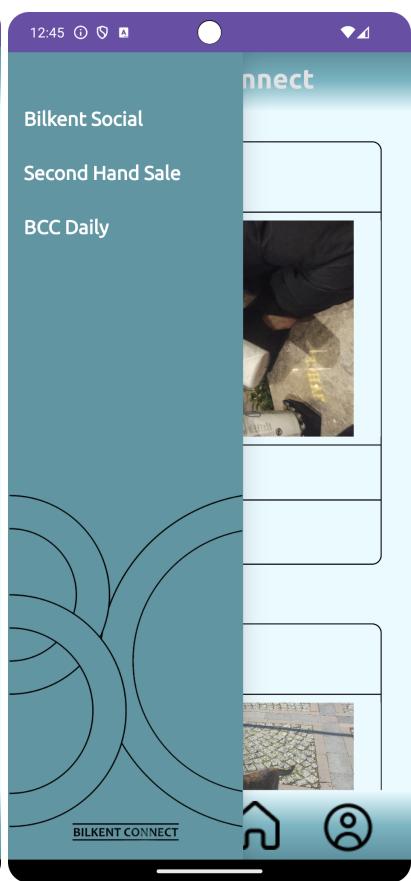
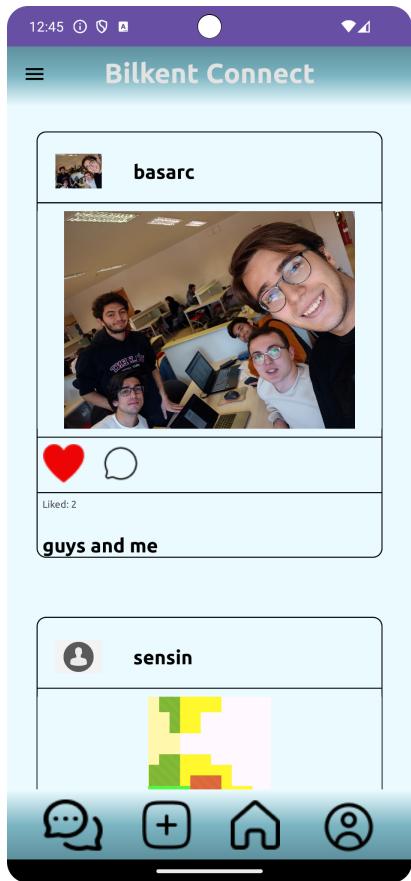
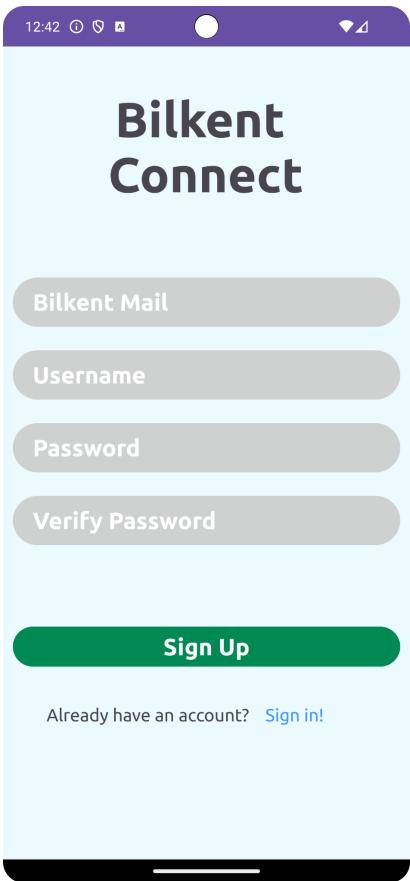
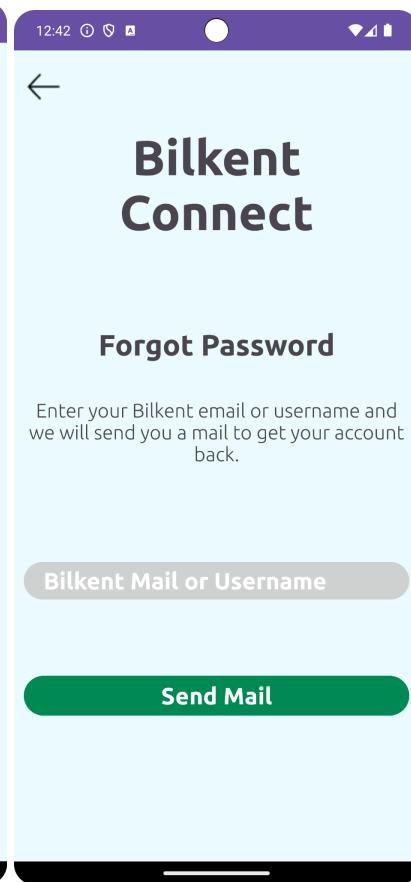
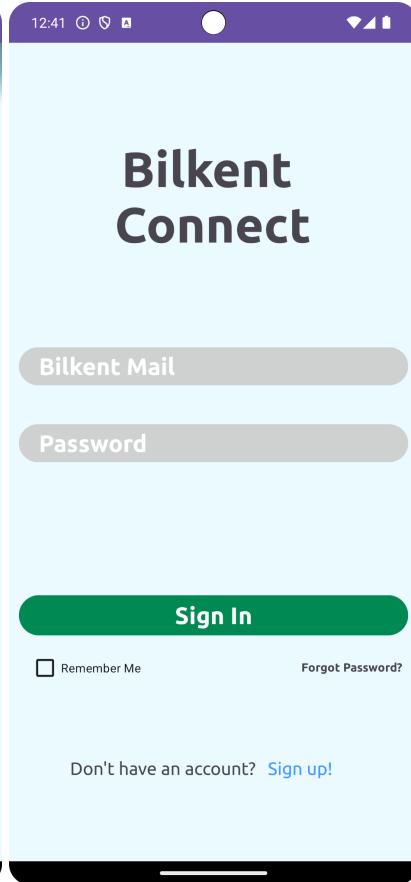
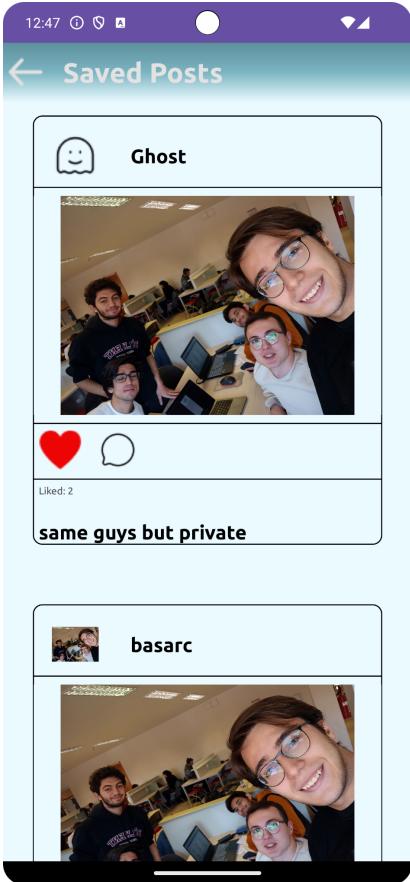
**Your Posts**

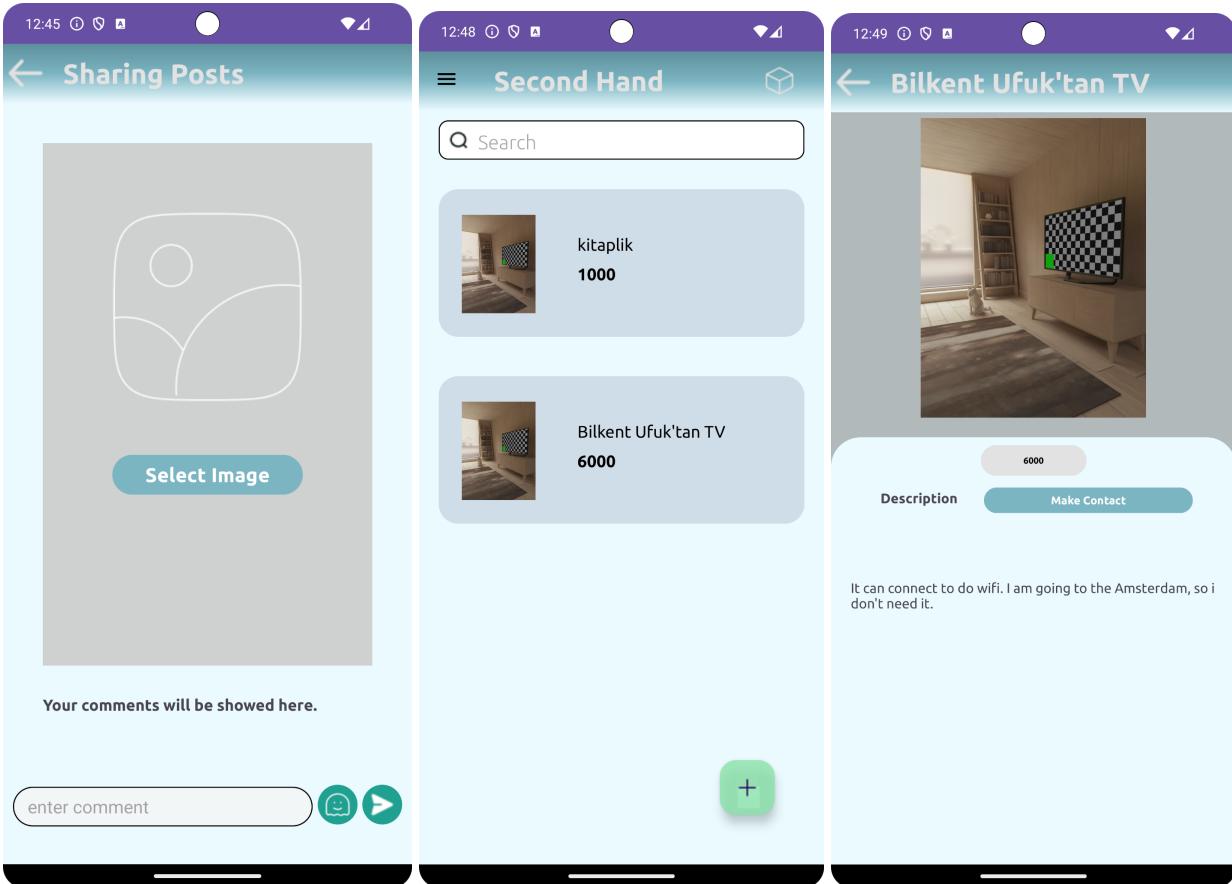
**Ghost**

same guys but private

Liked: 2

**Ghost**





## Reflections

### What did you dislike about project work?

Although there were times that we enjoyed observing the development of our project, it is undeniable that there were several parts of the project we disliked. For example, due to the concept of our application, We had to work on too many XML files, which was not so exciting. Additionally, the libraries we used for our project sometimes conflicted and created annoying bugs, which we disliked as a complete group.

### What was the most difficult aspect of it?

One of the most challenging aspects of the project was to work as a group in a large-scale project. Sometimes, it was hard to collaborate because of the schedule differences between the group members. Furthermore, this was the first time we had worked on something like this project, and we were all jumping into a new challenge. We were beginners in making an Android application. Therefore, there were several times that we just sat and thought about what we should do at this stage of the project.

### What would you do differently if you had to start over again?

We would certainly distribute the workload more equally. For example, at first, we thought one person would be enough to create a database for the application. However, as the project developed, we saw that the database workload had increased and got more

complicated. We would probably assign at least two people for the database if we could do the project again. Moreover, it would be better if we learned to use Android Studio before working on UI design because we used Figma, a collaborative design tool, for designing the UI. It seemed easy when we were working on Figma. However, implementing these designs on XML files took a lot of work. Thus, creating the XML files without using a design tool would take less time.

### **How much time did you spend on it?**

While we have not measured the time we worked on the project, we can say that it took approximately 15 hours per week. Of course, we have not worked for similar times each week due to the assignments for other courses. However, in sum, this project took around 200 hours during the 15-week semester.

### **Are you proud of what you have achieved?**

Although there are some minor parts we still need to complete, we have done a great job of developing a functional social media application with many functions. Seeing that we have accomplished most of the parts we imagined in the first stages of the project came as a relief in the end. Therefore, we can confidently say that we are proud of what we have achieved in this project.

### **What weren't you able to achieve?**

The only thing that we were not able to achieve is the chatting feature. We tried to implement it to our application, however, implementing it led to several bugs probably because of the conflict between the libraries we used.

## **Summary & Conclusions**

In conclusion, the system overview outlines the architecture and technologies employed, encompassing Android Studio for frontend development and Firebase for database services, including authentication, cloud storage, and analytics. The utilization of these technologies ensures a powerful and user-friendly application designed specifically for Bilkent students.

The frontend development using Android Studio emphasizes creating an intuitive and visually appealing user interface. By taking advantage of Android's native UI components and Firebase Authentication, the team aims to deliver a seamless and secure login system. The integration of animations and transitions further enhances the overall user experience.

Moreover, the database designed using Firebase wants to take advantage of the unique and fast database server provided by Google. Firebase provides a real-time database and Cloud Firestore, which ensures that user interactions such as messages and posts are updated in real-time across all devices, providing a dynamic and responsive user experience. In addition, Firebase offers high scalability and reliability. It effortlessly manages the fluctuating user traffic typical of social media platforms, ensuring smooth and uninterrupted service even during peak usage times.

The core design details include model classes representing users, products, chats, chat messages, comments, meals, and posts. These classes establish relationships to encapsulate relevant information and maintain a well-structured database.

Task assignments have been clearly outlined, with group members responsible for specific aspects such as UI design, frontend development, database management, and web scraping. This division of tasks ensures a collaborative and organized approach to the development of Bilkent Connect.

Overall, Bilkent Connect aims to provide a unique social media experience for Bilkent University students, facilitated by a well-designed architecture, thoughtful use of technologies, and a systematic division of development tasks.