



# EzemKofi

LKS Nasional 2023  
Test Project

## Introduction



EzemKofi is a coffee shop that runs in near downtown. The business has become a big success when they opened up the seventh branch last month. Their customers are always queueing to get their favourite coffee. Due to the high demand for coffee, the baristas are overwhelmed with the orders that keep coming in all the time.

In order to overcome that problem, EzemKofi's CEO plans to create a mobile app that can handle customers order online. The customer should have an account in their application in order to order a coffee. Customers can browse their favourite coffee based on their preferences. You are requested to help EzemKofi's CEO to make it happen. Below are some of the requirements that has been arranged by the analysts and designers.

## Contents

In this test project there are some files and usable resources:

- |                       |  |
|-----------------------|--|
| 1. EzemKofi_TP.pdf    | (Test project file)                        |
| 2. EzemKofi_Style.pdf | (Style guide file)                         |
| 3. Backend API        | (Folder that contains the API executables) |
| 4. Resources          | (Usable resources folder)                  |

## Project Description and Tasks

When developing the project, ensure all the deliverables meet the general requirements provided by the project designers:

- Project Criteria:
  - The project name should be “MOBILE\_II\_[XX]”, where XX is your PC number.
  - Time management is a crucial factor for the success of this project. So, manage your time wisely and submit all your works on time.
- Application User Interface:
  - Use the style guide provided consistently in the entire project.
  - The application wireframe is already provided by the UI designer.
    - You are allowed to modify the application interfaces as long as it doesn't interfere the business and application flow.
  - The scoring will be focused on the features of the application.
- Validation and Error Messages:
  - The application must runs well with a proper validation and shows a useful error messages to the user.
  - Analyze the required validation(s) that is/are needed in the application. This is one of the scoring criteria for your analytical skills for the project.
  - Any specific validation will be given in the project description.

## Instructions for Competitors

### 1. Run the API

Run the API that already given in the “**Backend API**” folder. **Click twice** on an executable file named “**EzemKofi.API.exe**”. This will be the data source that you will be using throughout the entire project development process.

### 2. Access the API Documentation

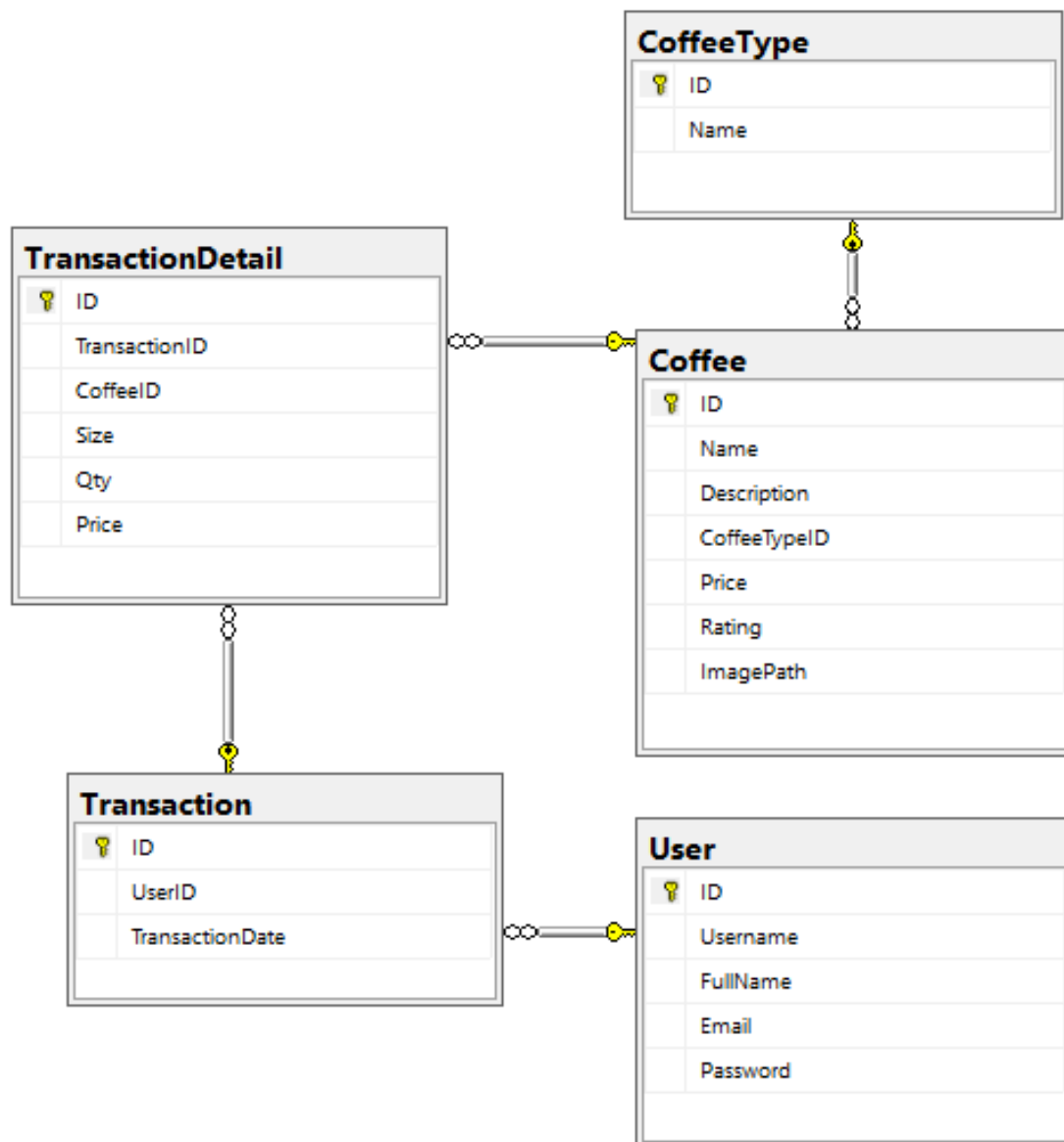
You can access the API documentation (Swagger OpenAPI) through this link:

<http://localhost:5000/swagger/index.html>

### 3. Develop the application and submit the project

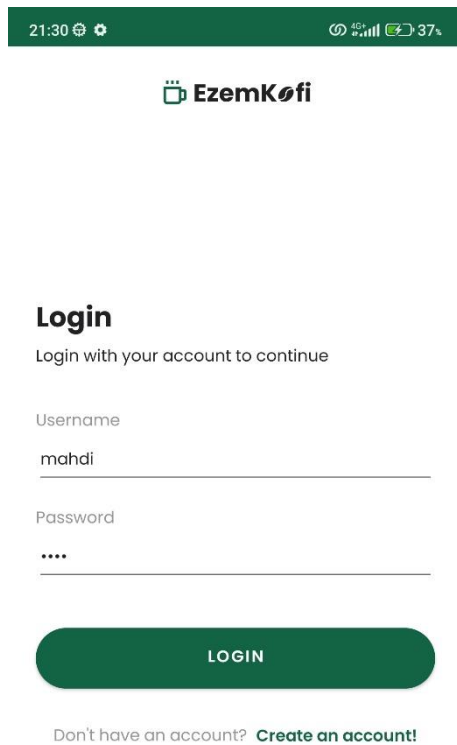
Develop the mobile application based on the description in the next section. And submit all your projects to the jury.

## Entity Relationship Diagram



To make it easier for you to understand the database structure, Database Designers from EzemKofi has provided you an Entity-Relationship Diagram (ERD). This diagram will explain the concept and representation of the data model that is used in the database.

## 1. Login Screen



21:30 4G+ 37%

**EzemKofi**

**Login**

Login with your account to continue

Username

mahdi

Password

....

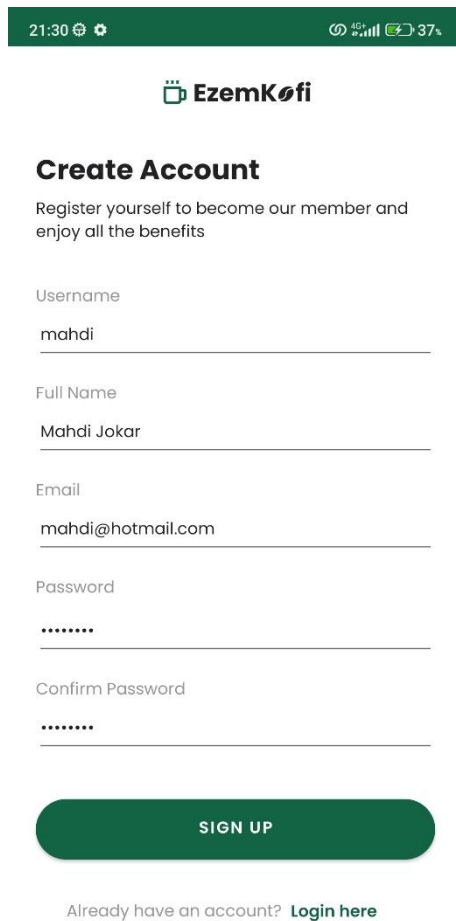
**LOGIN**

Don't have an account? [Create an account!](#)

This is the first screen that will be used by the user to login to the application. Registered users can login to the system with their username and password combination.

- Users can **login** to the application by **filling their username** and **password**.
  - If the **combination exists** in the database, the user will be **directed to** the **main screen** of the application.
  - If the **combination doesn't exists** in the database, **let the user know**.
- New users can **register** themselves by clicking the **“Create an account!”** link below.

## 2. Register Screen



The screenshot shows the 'Create Account' screen of the EzemKofi app. At the top, there is a status bar with the time 21:30, signal strength, and battery level at 37%. Below the status bar is the EzemKofi logo. The main heading is 'Create Account', followed by the text 'Register yourself to become our member and enjoy all the benefits'. There are five input fields: 'Username' with the value 'mahdi', 'Full Name' with the value 'Mahdi Jokar', 'Email' with the value 'mahdi@hotmail.com', 'Password' with masked characters '\*\*\*\*\*', and 'Confirm Password' with masked characters '\*\*\*\*\*'. A green 'SIGN UP' button is located below the input fields. At the bottom, there is a link that says 'Already have an account? Login here'.

21:30 4G+ 37%

**EzemKofi**

### Create Account

Register yourself to become our member and enjoy all the benefits

Username  
mahdi

Full Name  
Mahdi Jokar

Email  
mahdi@hotmail.com

Password  
\*\*\*\*\*

Confirm Password  
\*\*\*\*\*

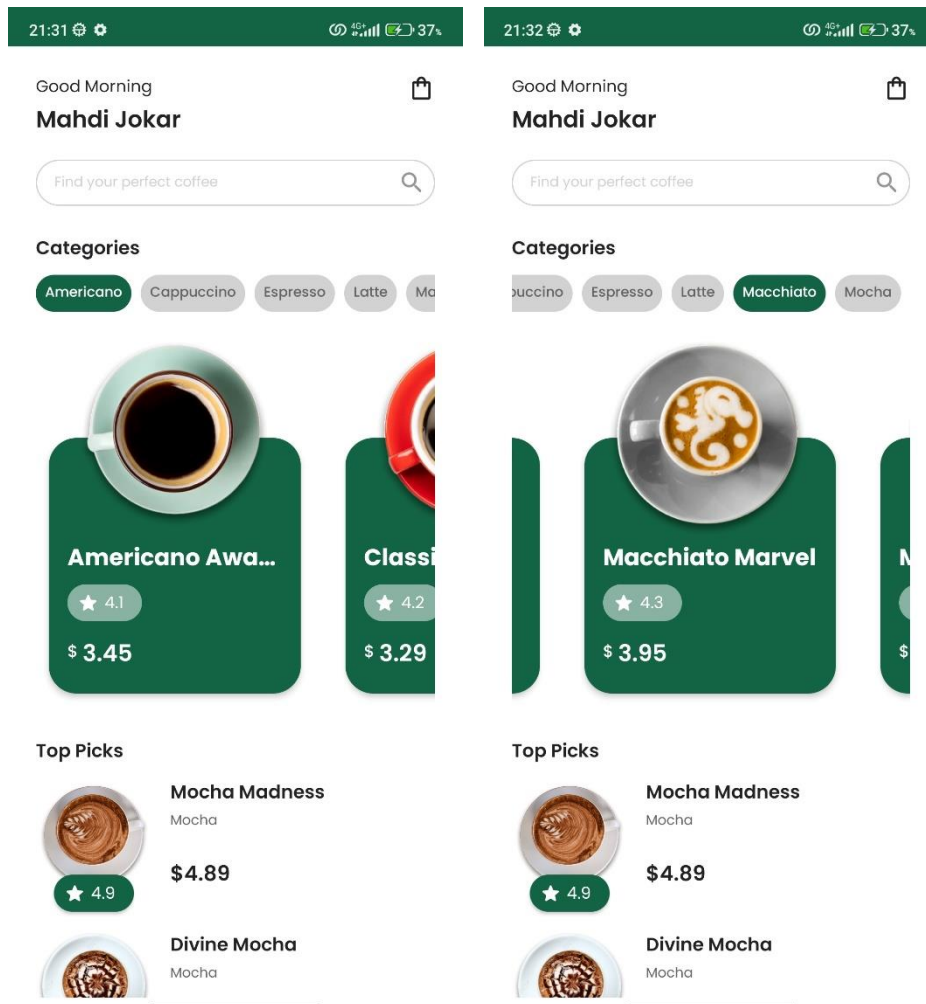
**SIGN UP**

Already have an account? [Login here](#)


In this screen, user can register an account by filling some informations.

- **Username** must be **unique** (must **not registered to the database** before).
- **Password** should have **minimal four characters**.
- When users **successfully registered** their account, **direct** user to the **main screen**.
- Users can also **go back** to the **login form** by clicking “**Login here**” link.

### 3. Main Screen

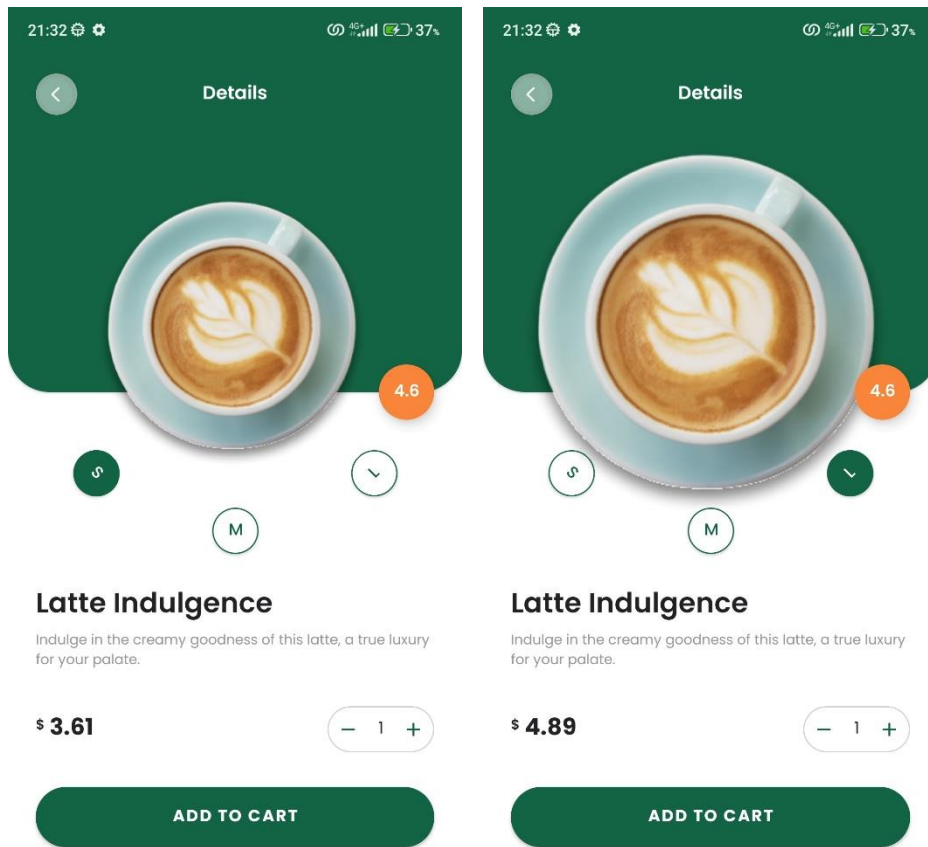


This is the main screen that will shows a list of coffee that user can choose. Users can search the coffee they wanted. User also can filter out the coffee list by its categories. There are also the recommended top picks coffee of the day below.

- The **shopping bag icon** () will **direct** user to **cart screen**.
- When user **clicks** on the **search field**, **direct** user to the **search screen**.
- **Show** the **category** list in **horizontal form** and make it **scrollable**.
  - The **default** selected category is **Americano**.
  - The **selected category** is **shown** in a **different colour**.
- **Show** the **coffee** list in **horizontal form** and make it **scrollable**.
- **Show** the **top picks coffee** in a **list** below.
- When user **clicks** on **one of the listed coffee**, user will be **directed** to the **coffee detail screen**.



## 4. Coffee Detail Screen



This screen will shows the detail of the selected coffee. Users also can add the coffee to their cart.

- The **back button** at the top left will **direct** users back to the **main screen**.
- Show the coffee detail information.
  - Image
  - Name
  - Description
  - Rating
  - Price

- Users can **choose** the **size** of the coffee (**Small, Medium, or Large**). The **default size** chosen is **Medium**.
  - When user choose the **small** size:
    - The **image** will **rotate 360 degrees** while the **size** is **shrinking** to **85%** from the **normal size**.
    - The **price** will become only **85%** from the **original price**.
  - When user choose the **medium** size:
    - The **image** will **rotate 360 degrees** and the **size** will become **normal**.
    - The **price** is as **same** as the **original price**.
  - When user choose the **large** size:
    - The **image** will **rotate 360 degrees** while the **size** is **increasing** to **115%** from the **normal size**.
    - The **price** will become **115%** from the **original price**.
  - For the **rotating** and **growing/shrinking animation** references are **provided** in the **resources folder**.
- Users can **modify** the **quantity** of the coffee they want to buy. The minimum quantity **can't be less than one**.
- When user **clicks** on the **add to cart** button:
  - If the user **hasn't added the coffee with that size to the cart** yet, **add the coffee** to the cart **normally**.
  - But if the user **has added the coffee with that size to the cart**, **update** the **quantity** by **adding previous quantity** with the **currently selected quantity**.
  - **Direct** user to **cart screen** after user **finish adding** to the cart/**updating** the cart.

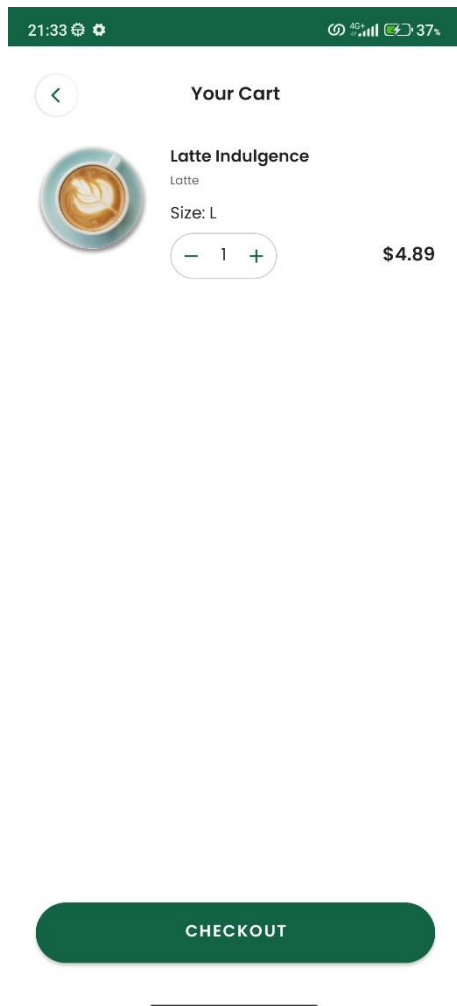
## 5. Search Screen



This is the screen to search a coffee. Users can filter the search result by entering the coffee name that they want to buy.

- The **back button** at the top left will **direct** users back to the **main screen**.
- The search results is displayed in a list.
  - Image
  - Rating
  - Name
  - Category
  - Price
- When user **clicks** on **one of the listed coffee**, **direct** user to **coffee detail screen**.

## 6. Cart Screen



This is the screen that will be used to check out all the cart items before. Users can view all the items that they put to the cart before. Users can also modify the quantity of an item from the cart.

- The **back button** at the top left will **direct** users back to the **main screen**.
- Show all the coffee information in the list.
  - The **price** is **calculated** based on the **quantity** and the **selected size**. You can see the size rules at the previous section (coffee detail section).
- Users can **update** the **quantity of an item** in the cart.
  - When the **quantity** is set to **zero**, then the item is **removed from the cart**.
  - **Recalculate** the **price** when **updating the quantity**.

- When user clicks on the **checkout** button, it will **do the checkout process** to the **database** and **clear** the **cart** so the user can start over with an empty cart.
  - Note that user **can't do checkout** with an **empty cart**.

~ Programming is thinking, not typing. But of course, why not both? ~