BIRZEIT UNIVERSITY
Electrical and Computer Engineering Department
ENCS5150 – Advanced Computer Systems Engineering Laboratory
Android Project - Smart Grocery Store App
First Summer Semester - 2025

Submission Deadline:
Discussion Date: 12 days after the submission deadline.
Team Work: Teams of two students
Total Marks: 100; 40% out of the final grade

Project Description:

A grocery store wants an Android application that allows customers to browse products, reserve/order items, view special offers, and contact the store.
The app works with online data (REST API) and also has an offline mode using SQLite.
There will be two roles: Customer and Admin.

The application should include the following features:

1. Introduction Layout (5 Marks)

❖ Create an animated screen that displays the grocery store logo (use any logo from the internet or create your own). Display it for 2-3 seconds.
❖ After animation, show an Introduction layout with:
  • A short description of the app.
  • A Connect button.

Connect Button behavior:
❖ Use REST API to fetch product categories.
  Dataset example (you can use this or create your own data or generate other random one using ChatGPT or any other method):
  https://mocki.io/v1/cede0a18-239c-4370-a84f-93fd197c5111
  Note: You need to replace the image links with real images.
❖ If the connection succeeds → navigate to Login/Registration screen.
❖ If it fails → show a toast message error and stay on this page.

2. Login and Registration (20 Marks)

❖ **2.1 Login (10 Marks)**

- The user can enter their email and password to log in.
- "Remember Me" checkbox to save the email in **SharedPreferences**.
- Validate email and password before login.
- If login succeeds → navigate to **Home**.
- If login fails → show an error.

❖ **2.2 Registration (10 Marks)**

Fields to fill:
❖ A Sign-Up button on login page opens a Registration layout.
❖ Fields:
1. Email (unique, must check that it is in email format)
2. First Name (≥3 characters)
3. Last Name (≥3 characters)
4. Password (≥5 characters, at least 1 letter, 1 number, 1 special character)
5. Confirm Password (must check that it matches Password)
6. Gender (Spinner)
7. City (Spinner with at least 4 palestinian cities)
8. Phone number (Auto-fill country code based on city)
❖ Validate inputs before submission.
❖ Password must be stored securely (you may use any **simple** method: eg, ceaser cipher).
❖ After successful registration → redirect to Login page.

3. Main Layout (10 Marks)

❖ Use a Navigation Drawer with:
- Home is the default screen (Has a short description 'about us').
- My orders.
- My Favorites.
- Offers.
- Profile Management.
- Contact Us (Call / Map / Email) use any dummy email/phone number.
- Logout.

4. Products Section (20 Marks)

❖ Display a list of products (from API or local DB).
❖ Each product must show:

1. Name
2. Category
3. Price
4. Stock status
5. Image

❖ Use Fragments to display products.
❖ Features:

    1. Search Bar: Search by name.
    2. Filters: Filter by category and price range.
    3. Add to Favorites Button: Save the product in favorites.
    4. Order Button: Open a form:

        • Quantity
        • Delivery method (pickup/home delivery)
        • Confirm button to place the order.

❖ Save order to SQLite and also sync with the server (if connected).

5. My Orders Section (5 Marks)

❖ Show all previous and current orders.
❖ Each order must include:
    1. Product name
    2. Quantity
    3. Date & Time
    4. Status (Pending, Approved, Delivered)

6. Favorites section (5 Marks)

❖ Show a list of favorite products added by the user.
❖ Allow ordering directly from this list.

7. Offers Section (5 Marks)

❖ Display special offers (discounted items).
❖ These products should also allow:
    • Add to favorites
    • Place an order

8. Profile Management (10 Marks)
❖ Display user information.
❖ User can:
    • View/update their first name, last name, password, and phone number.
    • Add profile picture.
❖ Password update must respect the same security conditions as registration.

9. Contact Us Layout (5 Marks)
❖ Three buttons:
    • **Call Us**: Call the store number (e.g., +970599000000), which opens the phone application and puts this number automatically.
    • **Locate Us**: Opens Google Maps location of the store. (opens BZU location)
    • **Email Us**: Opens Gmail with prefilled "to" field (YourStoreName@store.com).

10. Logout (5 Marks)
  ❖ Logs out the user and returns to the login page.

11. Admin Home Layout (10 Marks)
  ❖ Admin has a special Navigation Drawer with:
  - **Delete Customers**: Delete any customer.
  - **Add New Admin**: Add new admin (same rules as regular sign-up).
  - **Manage Products**: Add, edit, or delete products.
  - **Manage Orders**: View all orders and change their status.
  - **View Users**: See list of registered users.
  - **Logout**: Logs the Admin out.

  ❖ To differentiate between the admin profile and the regular user, each must have a different role.
  ❖ The application must have a pre-created static Admin account (Email: admin@admin.com, Password: Admin123!) to allow access to admin features.

Technical notes to be considered in your application:

  ❖ Android Layouts (dynamic and static).
  ❖ Intents and Notifications (e.g., Toast).
  ❖ SQLite Database.
  ❖ Animations (frame or tween animation when switching screens or loading data) for example, make a heart shape get larger, then back to its original size when adding something to favorites, or a bounce animation to Reserve buttons.
  ❖ Fragments (especially for product details, reservations, and favorites).
  ❖ Shared Preferences (for login persistence).
  ❖ RESTful API Integration.
  ❖ Data Validation and Exception Handling.

Notes:

  ❖ The application must be named following the format: FirstStudentID_SecondStudentID_CourseProject.
  ❖ Testing will be done on a **Pixel 3a XL device with API Level 26** (Graphics = Software). Ensure compatibility.
  ❖ Any form of **auto-generated code using AI tools (e.g., ChatGPT) will result in a zero mark**.
  ❖ You are allowed to use internet resources for learning, but plagiarism is strictly prohibited.

What to Submit?

1. **Project.zip file** (Export from Android Studio: File → Export to zip file)
2. **APK file** (Export from Android Studio: Build → Build Bundle(s) / APK(s) → Build APK(s))

Where to submit?

Submit both the APK file and ZIP file as a reply to the message on Ritaj.

Discussion

A discussion session will be scheduled after submission. Both team members must be present and ready to explain their code, make modifications, or add additional features if requested. Not attending the discussion will result in a grade of zero.