CET343 Android Mobile Development Assignment

This assignment contributes 100% to your final module mark and tests learning outcomes 1,2, 3 & 4. The assessment has been designed to enable completion within a **maximum** time frame of **42 hours**.

1. Advanced understanding of the various methodological approaches to native mobile design, including different programming languages, storage designs and appropriate native mobile standards.

2. Comparative knowledge of design and development techniques for the design of native mobile applications and storage systems

3. Ability to plan, analyse, design, build and implement a media and interaction rich native mobile application in a chosen domain

4. Ability to plan, develop and implement an integrated testing and evaluation strategy accounting for user experience, data integrity and network performance issues.

This assignment is split into three parts. Part A is worth 30 marks and is an analysis, design, functionality, testing and evaluation report which offers you the opportunity to discuss your app's life cycle, functionality against the requirements, present your testing results and evaluate the overall success. Part B is worth 60 marks and this offers you the opportunity to create the app you have designed. The final section, Part C, is worth 10 marks and gives you the opportunity to demonstrate your app.

The deadline for this assignment is available on **Canvas**. Please submit to the assignment space on canvas.

Scenario

FitLife

You have been commissioned to create an app that assists users in planning their weekly workouts while allowing them to create a personalized fitness checklist. This app enables users to input their favourite workout routines, including exercises, equipment needed, instructions, and pictures. It also features a workout checklist generator that consolidates all exercises and equipment required for the selected routines. Users can delegate tasks such as training reminders to friends, track expenses (e.g., gym fees, supplements), and even integrate with fitness platforms or local gyms for class availability.

Typical User Interactions

John is a busy professional who wants to stay consistent with his fitness goals despite his busy schedule. He searches for workout planning apps on the Play Store and discovers **FitLife**. After downloading the app, he registers for an account and logs in.

Inside the app, John begins by browsing a selection of workout routines. He selects several sessions for the week, including "*Full-Body Strength Training"*, "*HIIT Cardio"*, and "*Morning Yoga Flow"*. For each routine, he reviews the required equipment (dumbbells, yoga mat, resistance bands) and saves them to his weekly workout plan. The app automatically generates a checklist that includes all the necessary equipments, organized by category (e.g., strength equipment, mats, accessories).

Knowing that he won't always have time to prepare his equipments, John decides to delegate part of the task to his friend Mark, who often trains with him. He selects the checklist and uses the delegation feature to send Mark a text message containing the list of equipment to bring along, along with a reminder of which gym provides which equipment.

A few days later, while browsing a fitness blog, John discovers a new workout routine for "*Core Strengthening Pilates"*. He shares the link directly to FitLife, which pulls in the exercises and adds them to his workout plan and equipment checklist.

John also uses the app's geotagging feature to mark his favourite workout spots like the university gym, a nearby park, and a yoga studio, so he can quickly decide where to train. As he completes his sessions, he marks exercises as "done" in the checklist, ensuring he stays on track. The app also provides reminders for upcoming workout sessions and rest days, helping him stay consistent.

By the end of the week, John is proud of how easy it was to plan and follow his fitness schedule, manage his equipment, and stay motivated using FitLife. The app gave him more structure in his training, making fitness a natural part of his routine.

You are required to develop **a prototype application** for the Android operating systems which will include all the core requirements.

The prototype

Core requirements of the app:

- Home screen - the point of entry for your app.
- User registration and login
- Manage my items (the ability to manage exercises and workout routines):
    - Delete items – e.g. remove unwanted workouts or exercises.
    - Edit items – e.g. update exercise details (e.g., number of sets, reps, or instructions).
    - Mark items as completed (done) – check off exercises or workouts once finished.
- Create a workout routine – users can create custom workouts by inputting exercises, required equipment, and instructions (with the option of adding images).
- Item Delegation - the ability to send a workout equipment list or exercise checklist or routine reminders as an SMS to another phone/contact
- Implement only **one** desirable feature from the following list below:
    - **Geotagging** – the ability to geotag gyms, yoga studios, or parks so workout routines and checklists can be linked to a location and viewed on a map.
    - **Integration with other apps** – allow integration with fitness or wellness apps/blogs so that exercises or routines from other apps can be imported into FitLife.
    - **Gesture controls** – include at least three gestures to improve management of exercises (e.g., swipe left to delete an exercise, swipe right to mark as complete, shake to reset the workout list).

There is no associated logo or house style for this app yet so you have free reign over the appearance, but pay particular attention to Material Design.

Part A - Analysis, design, functionality, testing and evaluation (30 marks)

A technical report should be submitted. The document should include the following sections:

- **Front report page**. Include name, userID, registration number and programme.
- **Analysis**. This section requires you to provide a comparative evaluation of the various approaches to mobile app design and development, including operating systems, options for different programming languages, storage designs and appropriate native mobile standards. Your comparison should consider iOS and Android based solutions at a minimum. You should support your discussion with academic references. This section should be in the region of 750 words.
- **Design**. This section should show your screen hierarchy, plus wireframe design and menu / layout details. You should critically evaluate how your design matches current Material Design principles and UX trends. Screenshots of XML layouts created in Android Studio are not acceptable and will receive no marks.
- **Functionality**. Provide detail on key elements of your application's functionality with a breakdown of key examples of coding logic, ie how it works. You should give a clear description of the implementation, including storage requirements and external APIs used (if relevant), and justify your design and implementation choice.
- **Test Strategy and Test Results**. Summarise your strategy for testing and include a testing table of tests and their results.
- **Evaluation**. Comment on the success of your implementation and make recommendations for improvements.

Part B - Build of the mobile Android app - (60 marks)

The completed prototype app should be submitted. The application should be native and work well on an Android phone or tablet device. For the purpose of this project all data persistence can be local but it is encouraged to implement cloud solutions for data persistence.

Your prototype should include all of the core requirements mentioned in the scenario. Please only attempt one desirable feature from the provided list only, as marks will only be awarded for one. You must make it clear which desirable feature you have attempted.

Please ensure that all code (XML and Java / Kotlin) is commented appropriately and that you have packaged the app successfully before submitting a zipped project file to canvas.

Part C - Demo (10 marks)

You are required to demonstrate your app with a screencast recording. This should be no longer than 10 minutes. The purpose of this demonstration is to:

- To showcase the application and how well you have met the core requirements and the desirable feature requirement.
- To demonstrate any issues identified during testing which have not yet been resolved.

Submission

The deadline for submission is available on Canvas.

Following the new requirements for assignment submissions, you are required to fill out the academic declaration form which is available on Canvas.

Generative AI may be used only to support your learning and understanding of concepts. Any use of Generative AI or automated tools to generate parts of this assignment (e.g., code, diagrams, recorded screencast or written sections) will result in 0 marks for each affected element and may be referred for an academic misconduct investigation.

CET343 Assignment Submission Requirements:

- Filled academic declaration form (filled using the link on Canvas Assignments section)
- A PDF for your report (Assignment Part A)
- A Zip file containing your android studio project for the Application (Assignment Part B)
- An MP4 video file of your screencast (Assignment Part C)

Marking Criteria

Part A

Analysis (4 marks)
Full marks will be awarded for a clear, detailed and accurate analysis and evaluation of the various methods and approaches to native app development.

Design (10 marks)
Full marks will be awarded for a full collection of wireframe designs that follow current material design standards and follow current trends in application design. User interaction needs to be fully considered to ensure a good user experience and marks will reflect this. An explanation of your design and how it matches material design must be included to achieve full marks for this section.

Functionality (4 marks)
Full marks for clearly explaining your app's core functionality including explaining the storage of data and any APIs you might have used.

Testing and Evaluation (8 marks)
Full marks awarded for detailing your chosen test strategy, presenting your findings and critically evaluating how successful your app has been.

Presentation and referencing (4 marks)
Full marks for this section will be awarded for a professional report including a front cover and accurate referencing.

Part B

Core functionality to include the following (maximum 48 marks):

- Home screen with Login or Signup - the point of entry for your app. (6 marks)
- User registration / login (8 marks)
- Manage my items
    - Delete items (4)
    - Edit items (8)
    - Mark as completed (4)
- Create a workout routine (8 marks)
- Item Delegation - The ability to send a workout equipment list or exercise checklist or routine reminders as an SMS to another phone/contact (10 marks)
- Desirable Feature (1 to be completed, maximum 12 marks)
    - **Geotagging** – the ability to geotag gyms, yoga studios, or parks so workout routines and checklists can be linked to a location and viewed on a map.
    - **Integration with other apps** – allow integration with fitness or wellness apps/blogs so that exercises or routines from other apps can be imported into FitLife.
    - **Gesture controls** – include at least three gestures to improve management of exercises (e.g., swipe left to delete an exercise, swipe right to mark as complete, shake to reset the workout list).

Part C

Demonstration (10 marks)

A short, successful and professional demo of your application explaining how you have met the core and desirable requirement and any issues you had along the way. This is to be submitted as a screencast (mp4) no longer than 10 minutes.

Marking Grid

Analysis	**4 marks** A fully comprehensive accurate analysis and evaluation that includes the various methods and approaches to native app development.	**2-3 marks** A good analysis and evaluation of the methods and approaches to native app development. This may be lacking in detail and critical evaluation.	**0-1 marks** A poor analysis of current methods and approaches to app development. This may be significantly lacking in detail and critical evaluation.
Design	**8-10 marks** A full collection of wireframe designs that follow current material design guidelines and current trends in app design, complete with a detailed critical evaluation of how your	**3-7 marks** A full collection of wireframe designs that partially conform to material design guidelines. Your critical evaluation might be lacking in detail and justification.	**0-2 marks** Partially completed wireframe designs with an evaluation that falls significantly short of what is required.

	design meets these guidelines.		
Functionality	**4 marks** A fully comprehensive accurate description of the functionality included in your app.	**2-3 marks** A description of most of the functionality of your app but some parts may be missing or not described.	**0-1 marks** A poor or missing description of the functionality of your app.
Testing and evaluation	**7-8 marks** A comprehensive test strategy where all tests are relevant and documented, including a detailed documentation of how successful your app has been.	**3-6 marks** A good test strategy where testing is mostly relevant and you have attempted to critically evaluate how successful your app has been.	**0-2 marks** Partially completed or missing test strategy with little or no evaluation.
Presentation and referencing	**3-4 marks** Professional presentation including contents page and front cover with accurate referencing	**0-2 marks** Presentation of the document might be lacking professionalism and references might not be accurate	**N/A**

Core app functionality (maximum 48 marks)		
Home screen	**4-6 marks** A fully or partially working activity.	**0-3 marks** An incomplete or missing activity.
User registration	**5-8 marks** A fully or partially working activity.	**0-4 marks** An incomplete or missing activity.
Delete items	**3-4 marks** A fully or partially working activity.	**0-2 marks** An incomplete or missing activity.
Edit my items	**5-8 marks** A fully or partially working activity.	**0-4 marks** An incomplete or missing activity.
Mark as completed	**3-4 marks** A fully or partially working activity.	**0-2 marks** An incomplete or missing activity.
Create an item	**5-8 marks** A fully or partially working activity.	**0-4 marks** An incomplete or missing activity.
Delegate an item	**6-10 marks** A fully or partially working activity.	**0-5 marks** An incomplete or missing activity.

Desired functionality (maximum 12 marks and only one to be attempted)

Location tagging of items	**9-12 marks** A complete or very good attempt at the functionality	**4-8 marks** A good or partially working attempt at the functionality	**0-3 marks** An incomplete or poor attempt at the functionality
App integration	**9-12 marks** A complete or very good attempt at the functionality	**4-8 marks** A good or partially working attempt at the functionality	**0-3 marks** An incomplete or poor attempt at the functionality
Range of gestures and sensors	**9-12 marks** A complete or very good attempt at the functionality	**4-8 marks** A good or partially working attempt at the functionality	**0-3 marks** An incomplete or poor attempt at the functionality

Presentation screencast (maximum 10 marks)		
Presentation	**6-10 marks** A professional demonstration of your app covering how the app meets the needs of the scenario with the core requirements and any desirable requirements.	**0-5 marks** A demonstration of your app that may fall short of explaining the core requirements of the app to the desired level