Course: PROG 8790-23F-Sec 2
Course Name: Android Wear Development

Mid-Term Examination

Total Grade Points: 35
Time of Examination: 2 Hours 45 Minutes

Problem Specification

- Please read this requirement specification document completely to understand the application that you need to create.
- **It's your responsibility to read and understand all the requirements.**
- No kind of help in terms of solution or error fixing will be provided during the examination.
- Late submission will not be considered for evaluation under any circumstances
- Any remote attempt will be graded 0
- Don't share your solution with any one
- Do not use any automated tool like chatGPT
- You are allowed to refer all the materials posted under Week 1 to Week 6, Assignment 1, and other class activities
- The marking sheet is provided at the end of this document.

Create your Android Wear app project with the following specifications:

- The task is to create an Android Wear application for tracking the workout sessions.
- Project Name: XXWorkout, where xx is your initials. (For example, KPWorkout for Kevin Peterson)
- Project Type: No Activity
- Minimum SDK: API 30: Android 11
- Language: Java
- Create the strings, colors, themes, and dimens as needed for the UI
- ViewBinding must be enabled and implemented throughout the application
- Listener interfaces (if any) must be implemented at the class level
- Create separate packages for different classes for better organizing the code (for example, activity, model, utility, adapter etc.)
- Implement validations wherever needed
- Few of the resources are given to you as downloadable**. You can find these resources in the folder 'Midterm Exam – Resources' under Week 7.** You may use the provided resources or you are free to design of your own. Either way, the design of the pages should look exactly same as the screens images provided with the widgets as specified.

Build your prototype:

- Design the UI, if viewed on a wear emulator or real Android wear device will match EXACTLY the screen shots provided, except:
    - You can use the color themes of your choice following material design guidelines

- The design should be optimized for round and square displays
- Use camelCase naming conventions for id's, variables, and methods

Functional Requirements:

Note: Use SharedPreferences for any local data storage.

Screen 1: Add a Workout Session

1. Design the launcher screen for entering the workout details as seen in figure 1 below.
2. User should be able to input one of the workout type from '**Cycling**', '**Walking**' and '**Running**' as seen in figure 2 below in an EditText.
3. A timer should start running when the 'Start' button is clicked to track the workout session. When the timer starts running, the button should change to 'Stop' as seen in figure 3 below.
4. When the 'Stop' button is clicked, load another screen which lists all the workout sessions added along with the duration and the calories burned as seen in Figure 4.

   Hint: You may save the current workout session along with the duration and the calories burned in a SharedPreference when the 'Stop' button is clicked (Please note that you do not have to implement any sensors).

   Calculate the calories burned for each session using the below formula

   Calories Burned:

   Calory burned in one minute (CAL) = MET × 3.5 × (Weight in Kg)/200
   So, Calory burned during the entire period = CAL * Number of minutes entered

   Please assume weight as 70 Kg. as we are not accepting weight from the user.

   where MET (Metabolic Equivalent of Task) values for different types of activity are as follows: (Please note that we use approximate values)

   - MET for Walking : 3.5
   - MET for Running : 9.8
   - MET for Cycling: 11.5

   Also when the 'Stop' button is clicked, the timer should be stopped and reset to 00:00:00

Figure 1

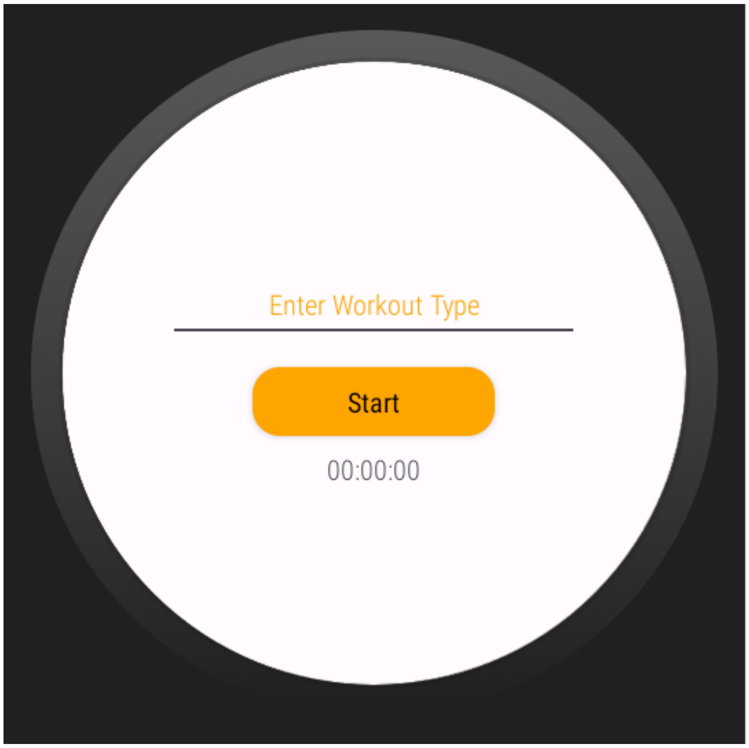


Figure 2

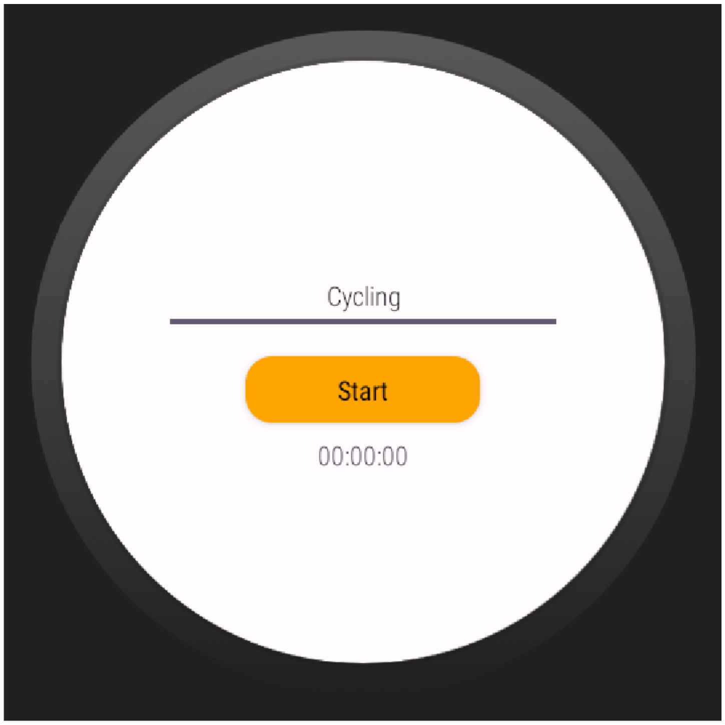
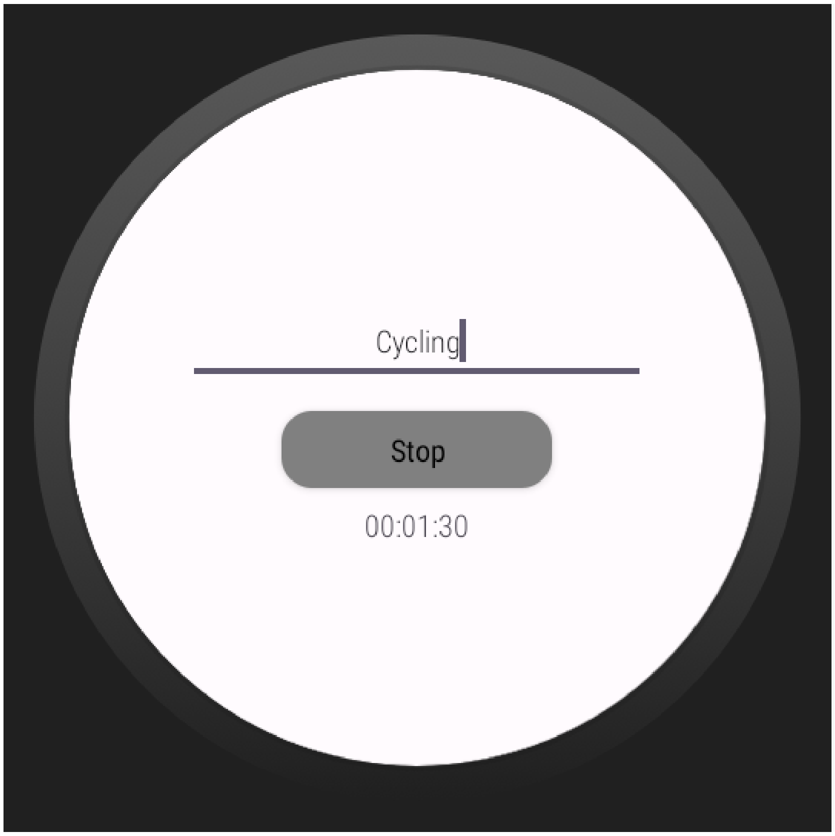

Cycling

Figure 3



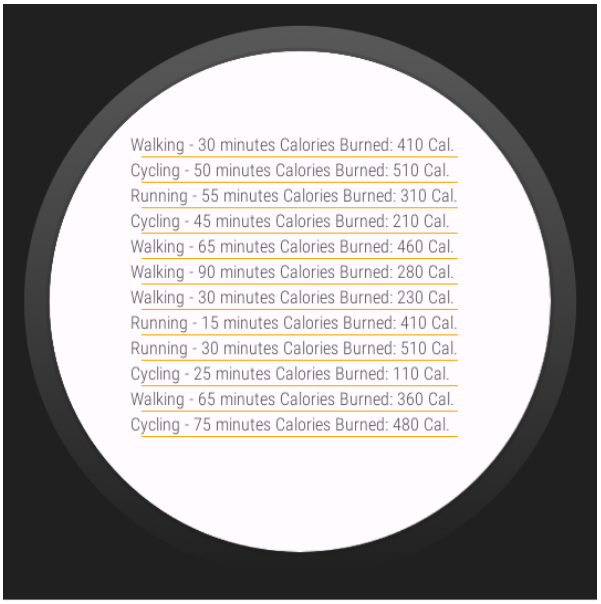
Cycling

00:01:30

Screen 2: List All Workout Sessions

1. Design the screen for listing all the workout sessions as seen in figure 4 below. Please note that the figure 4 shows a sample of the workout sessions only. You need to load the actual workout sessions you have added through the first screen.
2. List all the workout sessions that were added in a wearable recyclerview as shown

Figure 4



```
Walking - 30 minutes Calories Burned: 410 Cal.
Cycling - 50 minutes Calories Burned: 510 Cal.
Running - 55 minutes Calories Burned: 310 Cal.
Cycling - 45 minutes Calories Burned: 210 Cal.
Walking - 65 minutes Calories Burned: 460 Cal.
Walking - 90 minutes Calories Burned: 280 Cal.
Walking - 30 minutes Calories Burned: 230 Cal.
Running - 15 minutes Calories Burned: 410 Cal.
Running - 30 minutes Calories Burned: 510 Cal.
Cycling - 25 minutes Calories Burned: 110 Cal.
Walking - 65 minutes Calories Burned: 360 Cal.
Cycling - 75 minutes Calories Burned: 480 Cal.
```

What to Submit:

- **Comment the Code**: You must provide meaningful comments to every class, method, and every and any significant code segment.
- **Word Document:** Include the commented Java code and the screenshots of your applications in a **Word document**. Commented code should be pasted as text. Submission will not be considered for evaluation if commented code is pasted as an image. **Do not zip this document.**
- **Application Package:** Create a **zip file** of your entire application package.
- **Upload:** Upload the Word document and the zip file to the mid-term examination folder

Note: Please see marking sheet below:

Marking Sheet:

Description	Marks Allocated	Marks Achieved
‘Add Workout Session’ page designed as specified	2	
Screen 1: Functionalities implemented as per requirements: (Add Workout Session page) ➢ Input Workout Type ➢ Implementation of Start button ➢ Implementation of timer ➢ Implementation of ‘Stop’ button loading the second screen	 2 3 3 3	
‘List All Workout Sessions’ page designed as specified	2	
Screen 2: Functionalities implemented as per requirements: (List All Workout Sessions page) ➢ Loading all workout sessions added in the first screen in a Wearable Recycler View ➢ Calculation of calories burned for each session as per the formula provided	 7 4	
Organising the code into different packages	3	
Implementation of OOP principles	3	
Implementation of validations	3	
Total Grade Points	**35**	
Deduction		
Non submission of Word document file with the commented Java code pasted as text and screenshots	20	
Runtime errors	3 x --------	
Hardcoded text found in the UI	3	
Hardcoded color found in the UI	3	
Assignment Standard (proper project name, submission docs name etc)	1 x --------	
Programming Standards	1 x --------	
Bugs (including requirements mentioned in this specification)	1-5 based on severity	
Total		