TODO 1: ASCII hangman

Question:

Using Android Studio, you are required to design and implement an ASCII hangman game application, which works as follows: The player attempts to guess the word "birzeit". The Hangman is invisible at the beginning of the game, and when the player submits a wrong letter, parts of the figure should appear step by step in the following order:

head → body → right arm → left arm → right leg → left leg → LOSE GAME

Instructions:

Main activity Design:

1. The main activity contains a form for the player to register in the game before playing.
2. Required information are First name, Last name, and Student ID.
3. A button to start the game which changes the activity to the Hangman Activity where the game starts.

Hangman activity Design:

4. Display a blank Hangman figure at the beginning (no visible parts).
5. Provide an input field for the player to enter their guesses.
6. Display the Hangman word with hyphens ("-") for missing letters, initially set as "-------" to represent the word "birzeit."
7. Display a TextView to inform the player about their progress.

Game Logic:

1. The word to be guessed is "birzeit", the game should be case-insensitive (both uppercase and lowercase letters are considered the same).
2. Keep track of the Hangman figure's state (head, body, arms, legs) as the player submits wrong letters.
3. Allow a maximum of six incorrect guesses before displaying the complete Hangman figure.
4. Update the Hangman word display with correctly guessed letters.

Displaying the Hangman Figure: As the player submits incorrect letters, reveal parts of the Hangman figure:

1. First incorrect guess: Head
2. Second incorrect guess: Body
3. Third incorrect guess: Right arm
4. Fourth incorrect guess: Left arm
5. Fifth incorrect guess: Right leg
6. Sixth incorrect guess: Left leg

Winning and Losing Conditions:

1. Winning: When the player correctly guesses the word "birzeit", you should tell the user that he won and end the program.
2. Losing: When the player makes more than six incorrect guesses and displays the complete Hangman figure, you should tell the user that he lost and end the program.

User Feedback:

1. Inform the players when they win or lose the game. Remember to use student information filled out in the main activity.

ex ("Congratulations, Tarek Zidan! You won the game").
ex ("Sorry Tarek Zidan! You lost the game").

Restarting the Game:

1. Allow the player to restart the game after winning or losing, which clears the guessed letters, the number of wrong guesses, and the Hangman.

Note: Please ensure your code follows good coding practices, including appropriate variable names and error handling.

Use any design you want in the Main activity to submit your information, while your Hangman activity should look something close to the following:
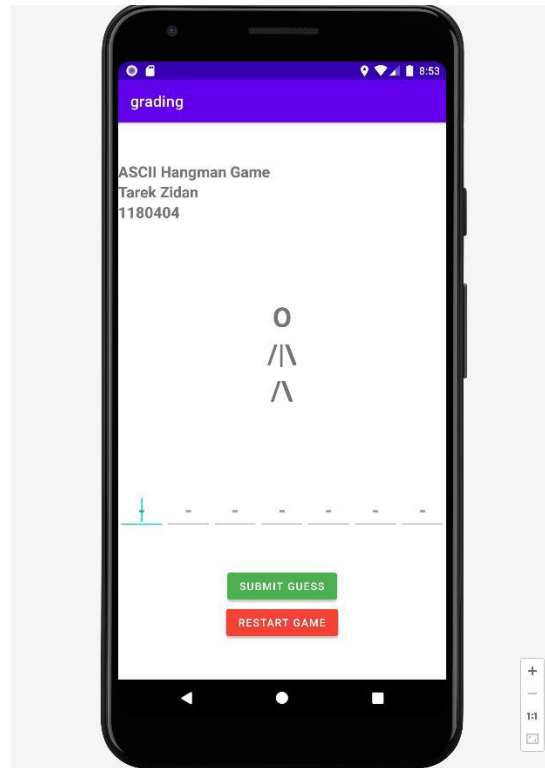
Notes:

- You can create any design as long as it is clear, arranged, and understandable.

- Marks are distributed as follows: (3 for the design, and 7 for **the functionality**).

- The name of the application must follow the convention **ID_FirstName_FamilyName.**

- I will use a Pixel 3a XL device with API Level 26 (Graphic=Software). So, please match these requirements to avoid unexpected issues once I test your solution.

- ToDo is individual: Cheating, using ChatGPT, is not allowed (it will lead to a 0 mark!). **We'll be very strict about that**.

What to submit?

1. Project.zip file (Size in KB)

From Android Studio: File → Export to zip file

2. APK file

From Android Studio: Build → Build Bundle(s) / APK(s) → Build APK(s)

3. Record your screen in a video that does not exceed 1 minute, show all possible test cases, upload your video to Google Drive, and attach your link (make sure it is accessible using the link), or just send the Video as an attachment if possible.

Where to submit?

• Send all the APK file and the ZIP file and drive link as a reply to my message

Deadline?

• Friday 4/10/2024 at 11:59 PM
• Late policy: You will receive a 10% reduction of your mark for each late day.