

# Please do not request edit access to this template

Refer to instructions on how to make a copy of the presentation template on the Appathon website: <https://appathon.appinventor.mit.edu/#:~:text=Presentation%20Information>

# YOUR APP NAME

Ammar Umar Elfaruk



# OVERVIEW

- **App Name:** (word puzzle&math puzzle)
- **App Description:** (a word puzzle that helps to know how to write and spell English words with a funny way and easy to understand,a math puzzle that you can mix number to get the double of this two numbers )
- **Track:** (Youth Individual,)
- **Category:** (Education & research)



# THEME

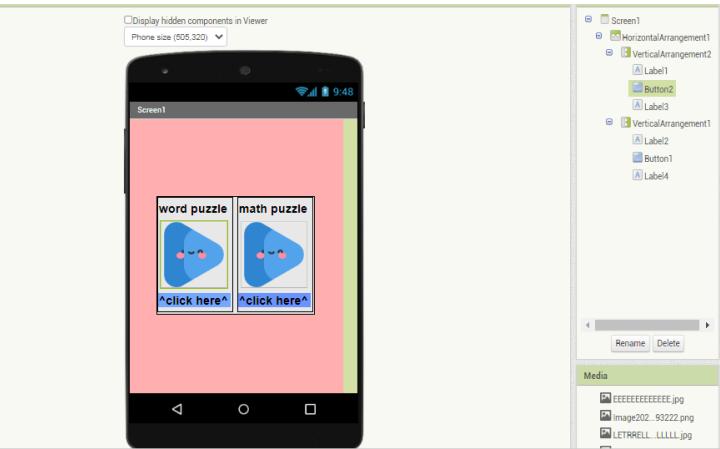
**In 100 words or less, describe the problem that you chose to address, and why you were interested in solving it. Please note that this portion of your submission will be published if you are selected as a finalist or winner.**

(this puzzle is made for fun and learning in the same time, and help to waste time in usefle lerning english and help to increase the ability for writing and spelling and arrange words, ,a math puzzle that you can mix number to get the double of this two numbers)

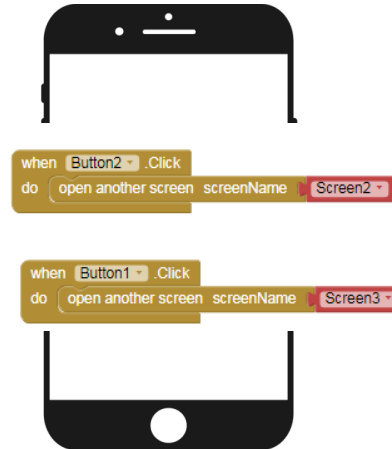


# INSTRUCTIONS

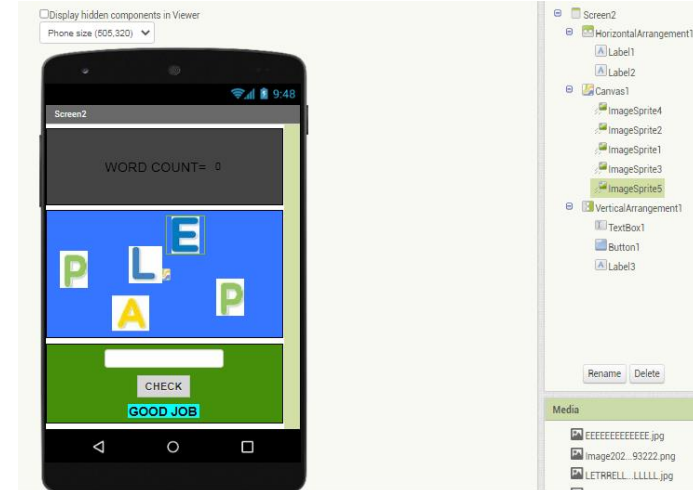
Please provide step-by-step instructions for using your app, along with high-quality screenshots of the app. Feel free to add as many slides as needed.



Step 1: (fill in description)



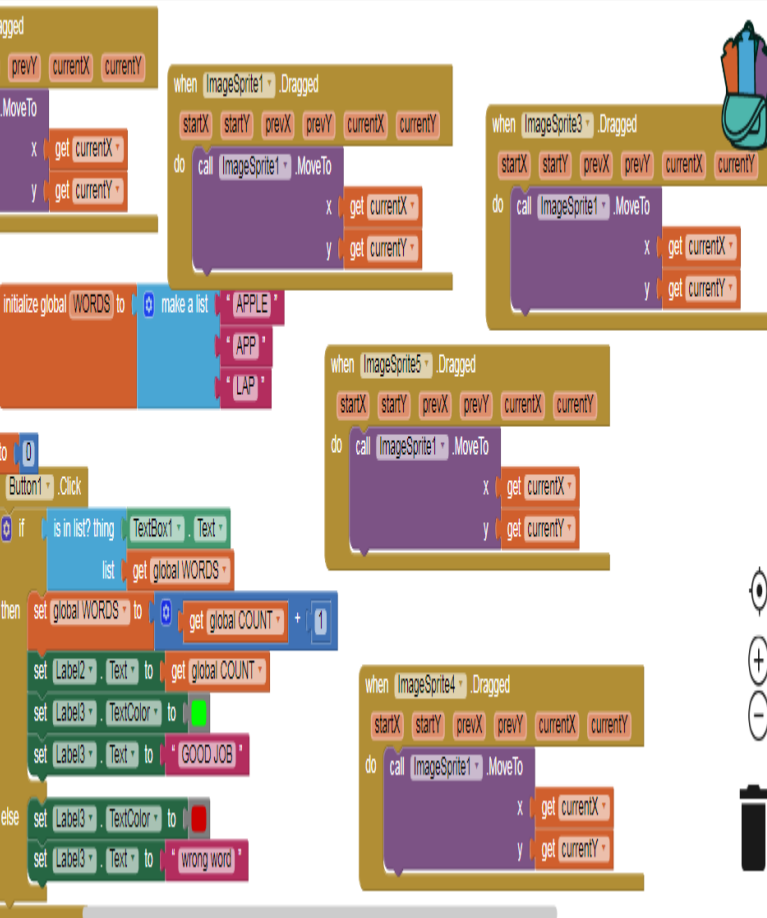
Step 2: (fill in description)



Step 3: (fill in description)



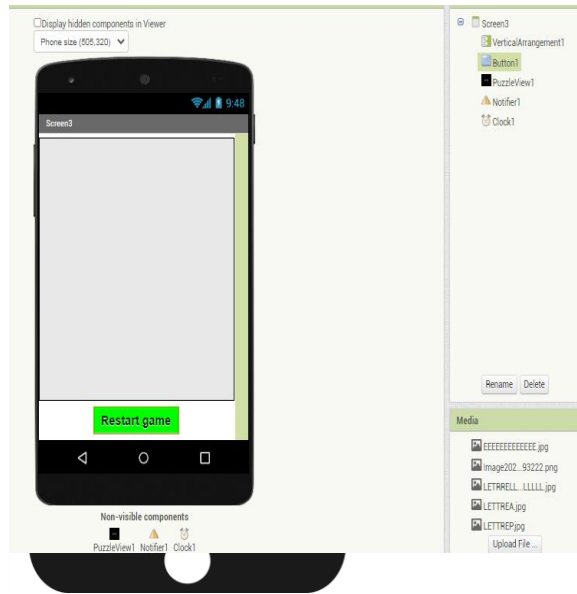
# INSTRUCTIONS



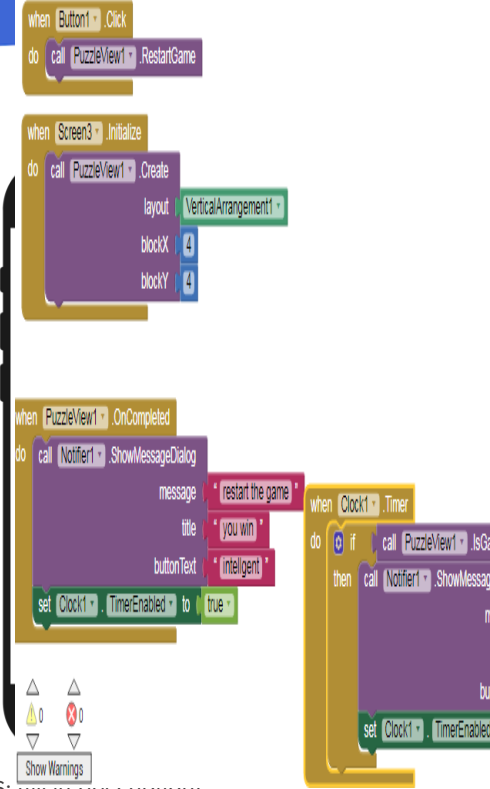
Step 5: (fill in description)

Code blocks for Step 5 include:

- When `ImageSprite1` is dragged: `startX`, `startY`, `prevX`, `prevY`, `currentX`, `currentY`; `do` call `ImageSprite1` MoveTo with `x` `get currentX` and `y` `get currentY`.
- When `ImageSprite3` is dragged: `startX`, `startY`, `prevX`, `prevY`, `currentX`, `currentY`; `do` call `ImageSprite1` MoveTo with `x` `get currentX` and `y` `get currentY`.
- When `ImageSprite5` is dragged: `startX`, `startY`, `prevX`, `prevY`, `currentX`, `currentY`; `do` call `ImageSprite1` MoveTo with `x` `get currentX` and `y` `get currentY`.
- When `ImageSprite4` is dragged: `startX`, `startY`, `prevX`, `prevY`, `currentX`, `currentY`; `do` call `ImageSprite1` MoveTo with `x` `get currentX` and `y` `get currentY`.
- Initialize global `WORDS` to `make a list` with `"APPLE"`, `"APP"`, and `"LAP"`.
- When `Button1` is clicked: `if` `is in list?` thing `TextBox1` .Text with `list` `get global WORDS`; `set global WORDS` to `get global COUNT` + `1`; `set Label2` .Text to `get global COUNT`; `set Label3` .TextColor to `green`; `set Label3` .Text to `"GOOD JOB"`; `else` `set Label3` .TextColor to `red`; `set Label3` .Text to `"wrong word"`.



Step 5: (fill in description)



Step 6: (fill in description)

Code blocks for Step 6 include:

- When `Button1` is clicked: `do` call `PuzzleView1` RestartGame.
- When `Screen3` is initialized: `do` call `PuzzleView1` Create with `layout` `VerticalArrangement1`, `blockX` `4`, and `blockY` `4`.
- When `PuzzleView1` is completed: `do` call `Notifier1` ShowMessageDialog with `message` `restart the game`, `title` `you win`, and `buttonText` `intelligent`; `set Clock1` .TimerEnabled to `true`.
- When `Clock1` is a timer: `do` `if` `call PuzzleView1` IsGameOver; `then` call `Notifier1` ShowMessageDialog with `message` `game over`, `title` `game over`, and `buttonText` `restart`; `set Clock1` .TimerEnabled to `false`.

Step 6: (fill in description)



# LIMITATIONS

In 100 words or less, describe the limitations of your app and what people should carefully consider when using it.

(Fill in your answer here)



# ACKNOWLEDGEMENTS

Please list the names of anyone who helped you with developing your app, and describe what type of help they provided.

- Person #1: (fill in how they helped you)
- Person #2: (fill in how they helped you)
- ...







# APPENDIX

If you have any supplementary information you wish to include, feel free to add it to the Appendix. **This section is entirely optional.**

It is recommended to add materials to the Appendix if you submitted a hardware project (e.g. with schematics, flow diagrams) or if your project included significant programming components outside of the MIT App Inventor platform.

Please be aware that judges reserve the right to exercise their discretion in reviewing materials within the Appendix, and they may not review its entirety if they consider it excessively lengthy.

# TITLE

Education&resrshes,english,math,word puzzle,math puzzle

