PA1
Simple Wordle

Mobile App Programming

Wordle

- Guess a five-letters English word.
  - Letter not included / Included but wrong pos / Right
  - Similar with Bulls and Cows(숫자야구)
- https://www.nytimes.com/games/wordle/index.html

Wordle

- **Green**: Right letter ON Right position
  - Same letter could be elsewhere too
- **Yellow**: Right letter **BUT** Wrong position
- **Gray**: That letter is **NOT** included

PA1 Goal

- Make Wordle Application
  - We will give you a dictionary file.
  - User input must be typed on **EditText**.
  - User can only guess with a valid word.
    exists in the dictionary file.

  - Guess list must be shown.
  - Gray/Yellow/Green **letter list** must be shown.
  - Lists must be implemented using **ListView** (RecyclerView is also allowed…)
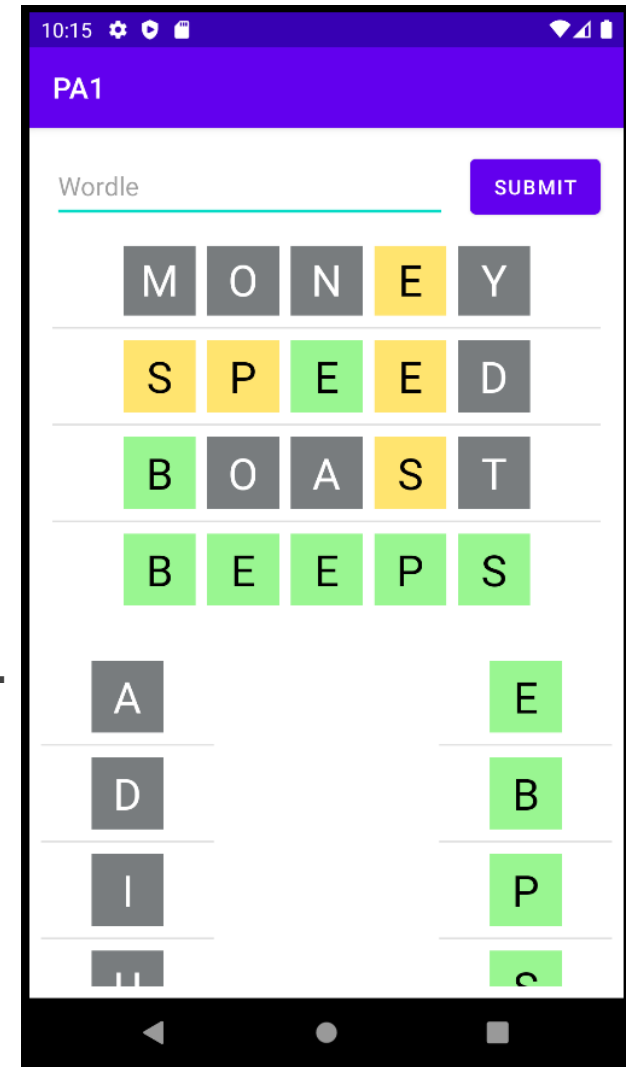
PA1 Goal

- UI
  - Score will be deducted when
    - You did not follow guideline
    - AND seemingly bad
      - No objection allowed
  - TAs will score your apps
    - **Pixel 5 API 29**
      - **1080 x 2340** : 440dpi
      - Android 10.0 (API 29)
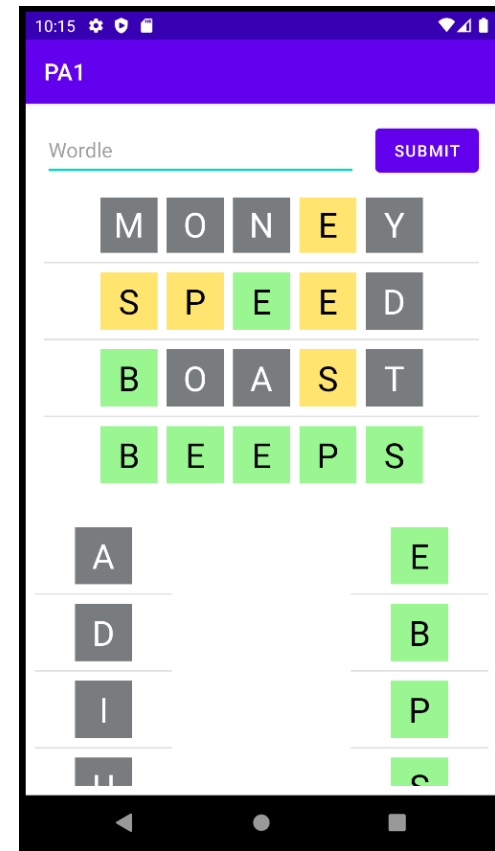
PA1 Specification

- Project Settings
  - Minimum SDK: Must be **29** (Android 10.0)
  - Target & Compile SDK: Must be **34** (Android 14.0)
  - Application ID(Package): Must be **edu.skku.cs.pa1**
- Application Execution
  - The wordle screen must be shown without any action.
  - The application must be started in 10 seconds.
  - UI must not stop more than 5 seconds.
  - No error while build/execution.

PA1 Specification
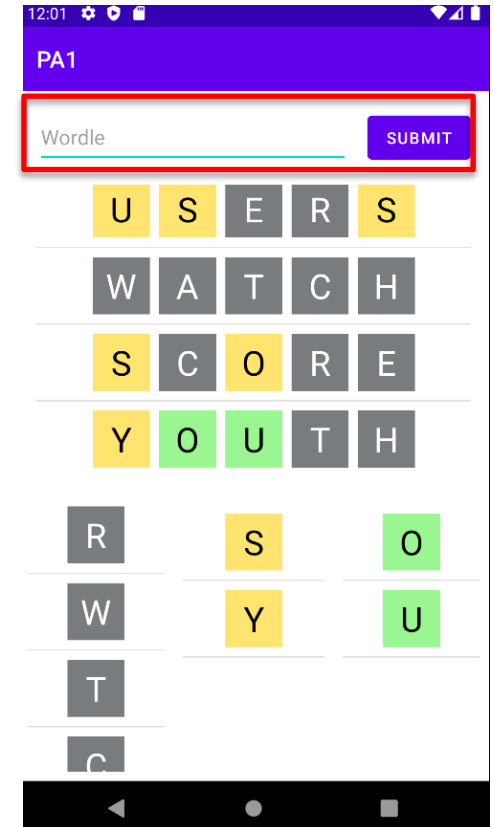
- ## User Interface: Screen

  - ### EditText

    - **Height**: 50dp
    - **Width**: match_constraint.
    - **Margin** (top, bottom, start) : **16dp**
    - **Text**: None (empty)
    - **Hint**: Wordle

  The margin btw EditText and button : 16dp
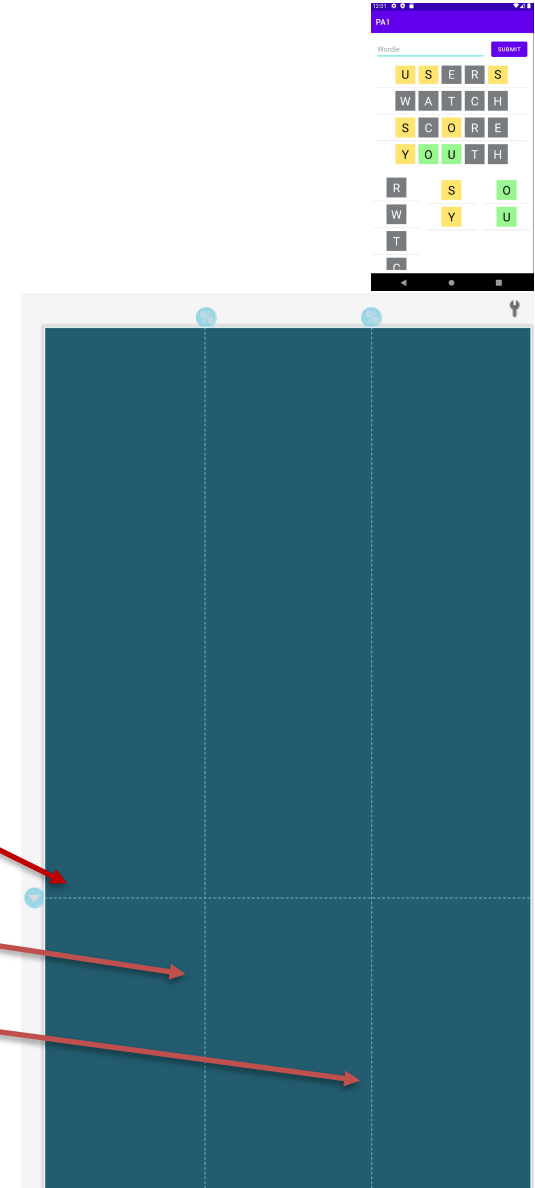
  - ### Button

    - **Height** : 50dp
    - **Width** : wrap_contents
    - **Margin** (top, bottom, end) : **16dp**
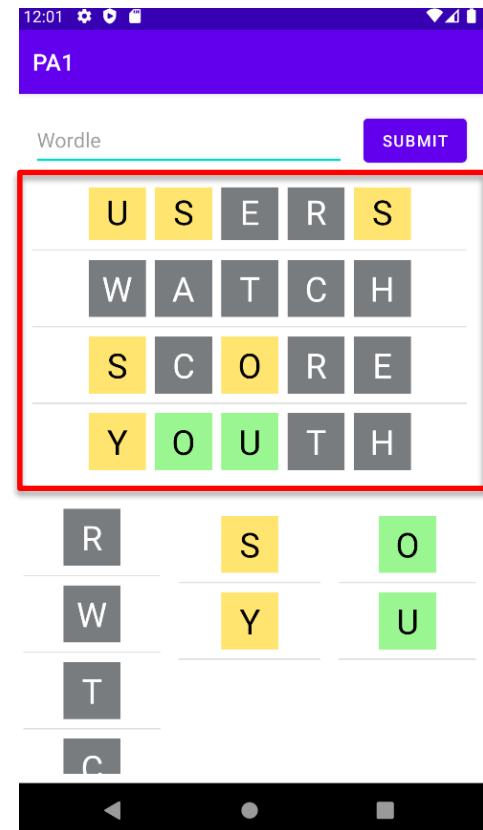   - **Text** : **SUBMIT**

PA1 Specification

- User Interface: Screen
  - Guideline
    - androidx.constraintlayout.widget.Guideline
    - Horizontal
      - 250dp from bottom
    - Vertical
   - 33%
      - 67%

PA1 Specification

- User Interface: Screen
  - Wordle Word List
    - Listview (or RecyclerView allowed)
    - ListView Item Layout: **Refer P11**
    - ListView's Margin (**All direction**): **16dp**
    - Height & Width: match_constraint.
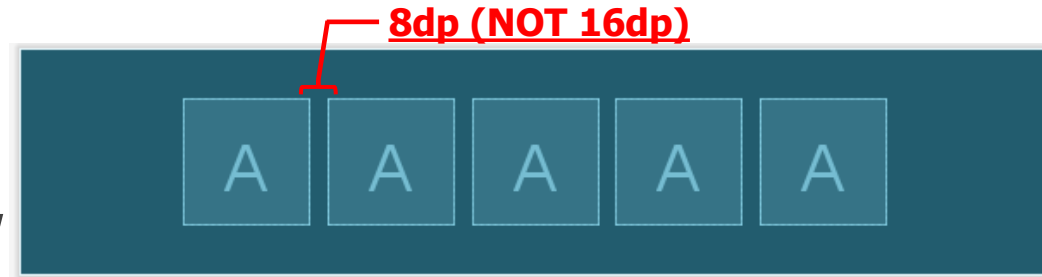
PA1 Specification

- User Interface: Screen
  - Wordle Letter List
    - Listview (or RecyclerView allowed)
    - ListView Item Layout: **Refer P12**
    - ListView's Margin (**All direction**): **8dp**
    - Height & Width: match_constraint
    - Leftmost contains (**Gray**) letters
    - Middle contains (**Yellow**) letters
    - Rightmost contains (**Green**) letters

# PA1 Specification

8dp (NOT 16dp)
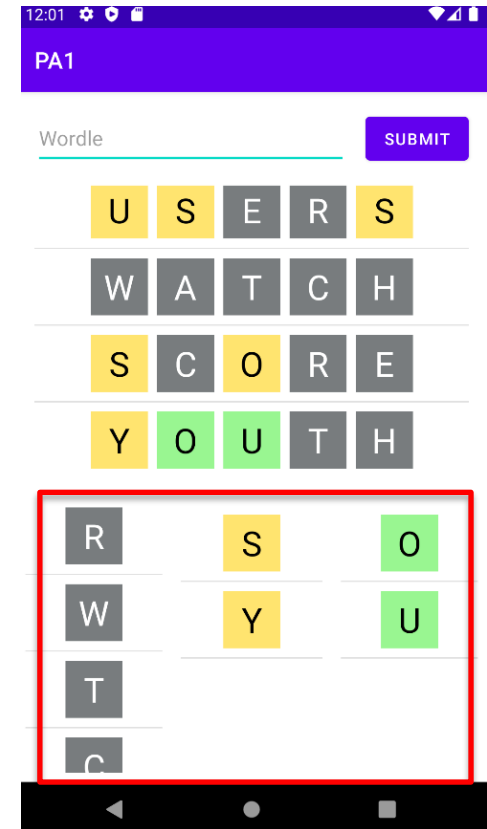
- User Interface: Listview

  - Item of **Wordle Word List**

    - All TextViews must be **centered.**

    - Vertical Margin: **8dp**

    - **Horizontal margin** between TextViews : 8dp

    - **Text** should be **centered** vertically & horizontally inside a TextView.

    - Letter Textview Attributes

      - **Width & Height** : 50dp

      - **Text Size** : 30sp

PA1 Specification

- User Interface: Listview
  - Item of Wordle Letter List
    - Must be centered horizontally.
    - Letter Textview Attributes
      - **Width & Height**: 50dp
      - **Text Size**: 30sp
    - **Margin (top and bottom)**: 8dp
   - **Text** must be **centered** vertically or horizontally inside a TextView.

PA1 Specification
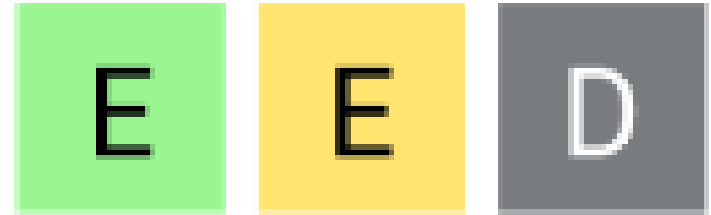
- **User Interface: Others**
  - Wordle Letter
    - Right Letter & Right position (Green)
      - Background = **#FF99F691** (#99F691)
      - Text = **#FF000000** (#000000)
    - Right Letter & Wrong position (Yellow)
      - Background = **#FFFFE46F** (#FFE46F)
      - Text = **#FF000000** (#000000)
    - Wrong Letter (Gray)
      - Background = **#FF787C7E** (#787C7E)
      - Text = **#FFFFFFFF** (#FFFFFF)

PA1 Specification
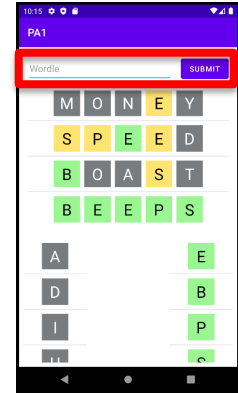
- Functionalities
  - Wordle
    - On app execution, generates a five-letter random answer from **the dictionary file**.
  - Guess
    - If a user types word in EditText and presses **SUBMIT** Button,
      - Checks if it is in the dictionary,
        - » If not, shows **toast message**: "Word '*<User_Input>*' not in dictionary!"
        - » If in, adds that word into below lists and clear EditText.

PA1 Specification
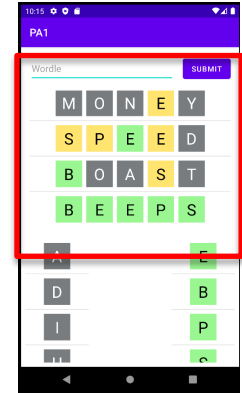
- Functionalities
  - Inserts word into **Wordle Word List**
    - Once user guessed a word from dictionary, **it should be added into Wordle Word List.**
    - The guessed five-letter word should be shown in one line.
    - Newly added word must insert into the bottom-most.
    - **Set background colors and text colors** according to the rules and color codes in **Page 13**.
      - (Green), (Yellow), (Gray)
   - It should be scrollable.

PA1 Specification
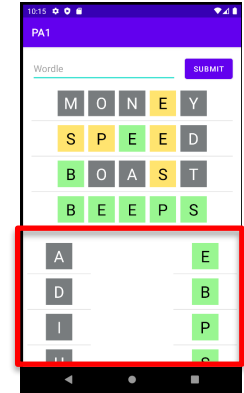
- Functionalities
  - Inserts a letter into **Wordle Letter list**
    - Once user guessed a word from dictionary,
      all the guessed five letters should be **inserted into proper list**.
    - Set the **background color and text color** according to
      the rules and color codes in **Page 13**.
      - (Green), (Yellow), (Gray)
    - Since all threes are ListView, each ListView Should be scrollable.
    - Newly added letter must be inserted at the bottom of the list.

PA1 Specification

- Functionalities
  - Wordle Letter list
    - If a letter becomes **Gray**,
      - Adds to **Left List** if **NOT already in it**.
    - If a letter becomes **Yellow**,
      - Adds to **Middle List** if **NOT already in Middle or Right list**.
    - If a letter becomes **Green**,
      - Add to **Right List** if **NOT already in Right List**.
      - This letter should **be REMOVED from other lists**.
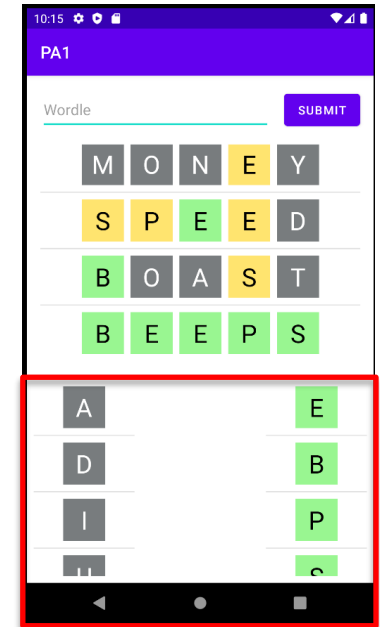
PA1 Specification

- Functionalities

  - **CAUTIONS!** - Wordle Letter list

    - If a letter **exists in multiple lists more than once**, **insert** it into **the list located further to the right** **(Gray < Yellow < Green).**

    - If the **color of the letter needs to change** in the next input of the user, **this change should also be reflected in Wordle Letter List.**

PA1 Specification

- Functionalities

  - Prints **the minimum number of attempts**

    - When the application starts, a **Toast Message** should display the "**minimum number of attempts to reach the correct word**". (The minimum attempts across all games since app was installed.)

    - Because we don't have a reset button all the number of attempts should be **stored somewhere (& somehow)** and should be loaded when the application starts.

    - **Toast Message Format:**
      - » On the **first** trial:
        "This is the first attempt."
      - » Else,
        "The MIN attempts : <*Word*> - <*# of attemps*>"

PA1 Others

- **We will not care about…**
  - # of Adapter classes
  - # of Adapter objects
  - NO NEED TO make reset button : just restarts app
  - Successful guess: NO NEED to do something else.
- **Questions?**
  - https://docs.google.com/spreadsheets/d/1TWuGM4wPLUg2rK-XUCz17M-ggpVf9QnnekdHP1eUINA/edit?usp=sharing
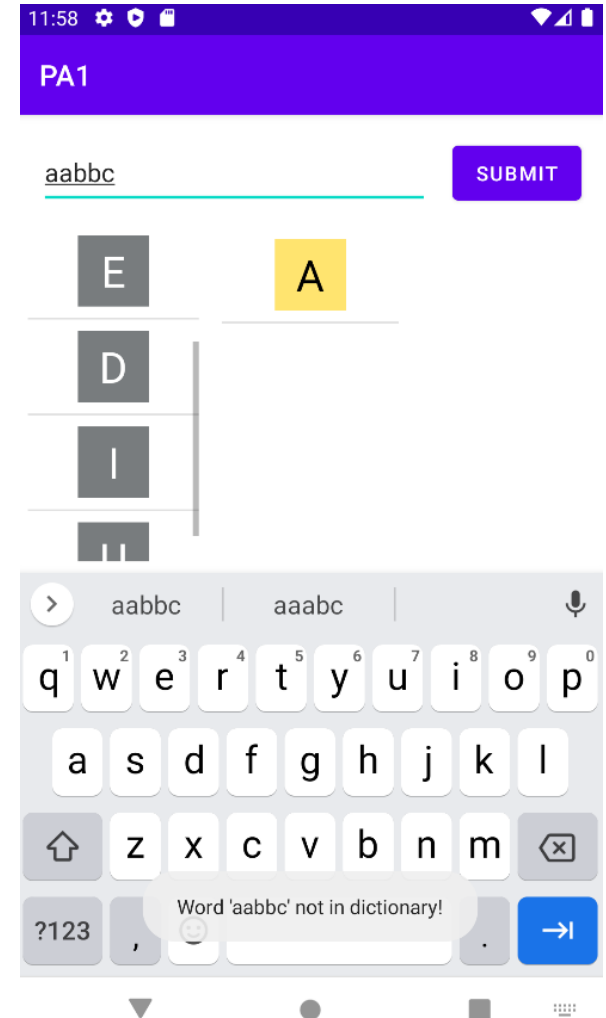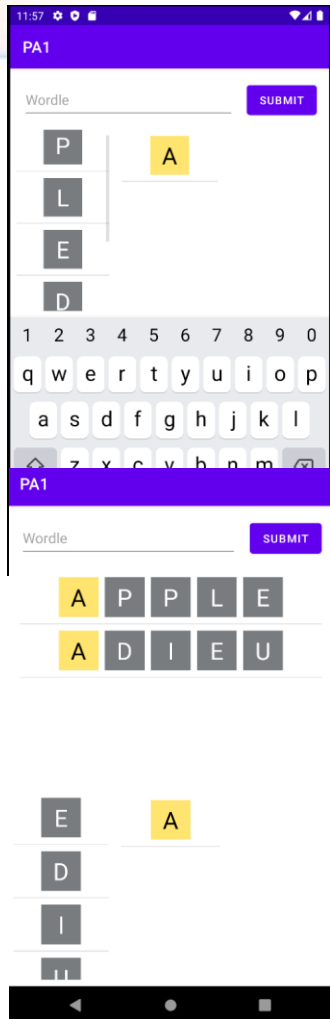 - See this slide until the end.
  - Tips

PA1 Others

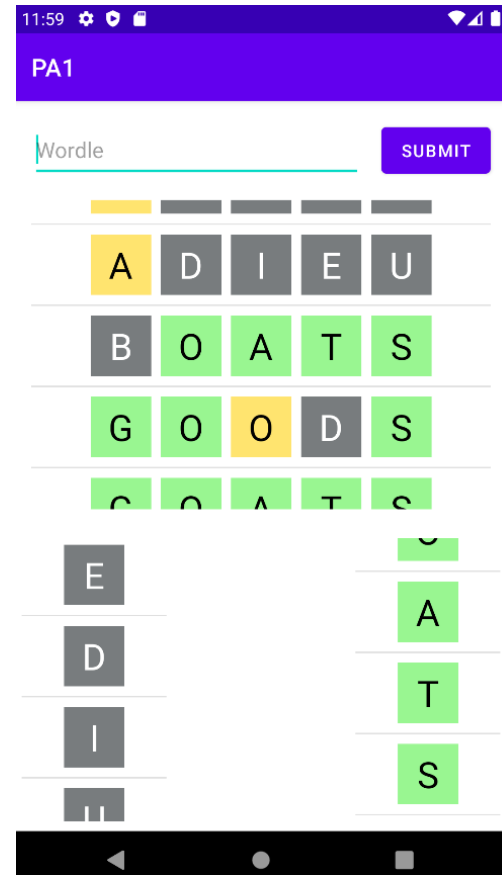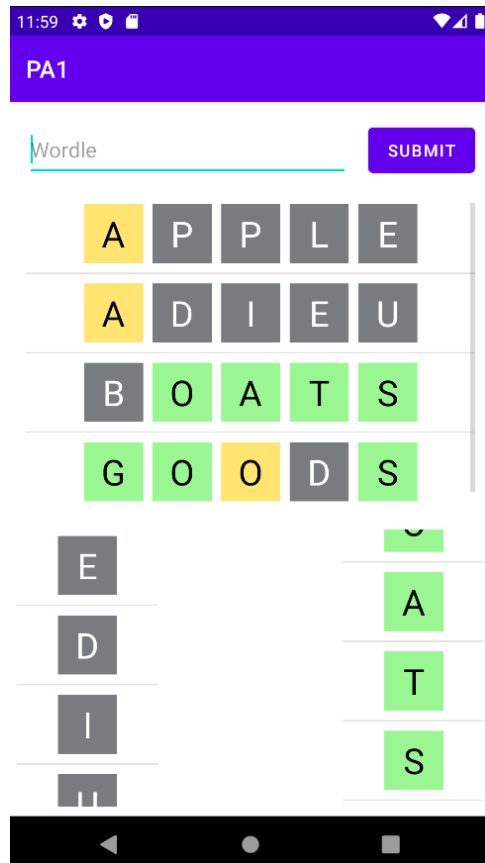**We DO NOT care**
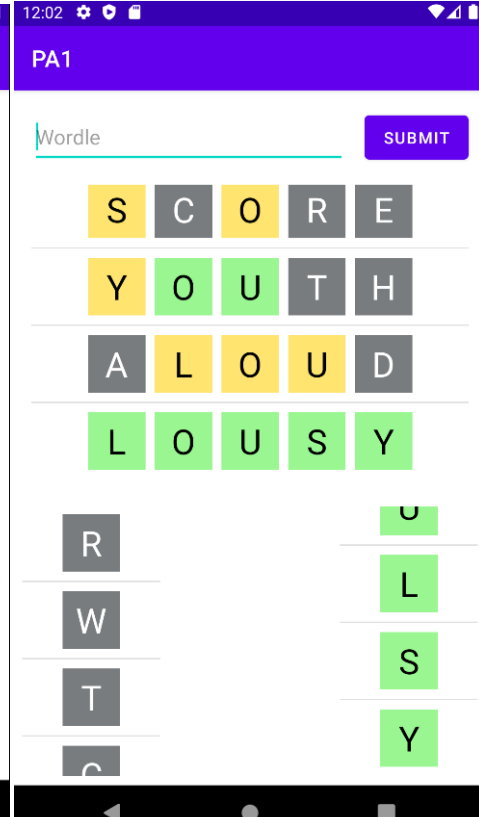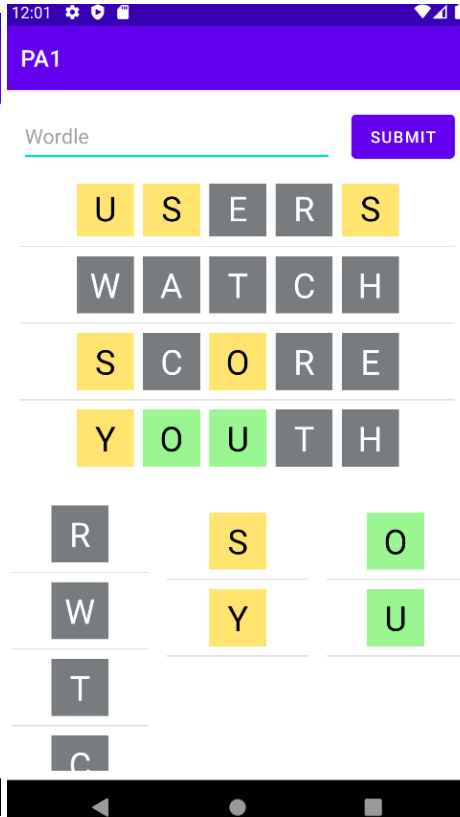**the existence of Application Title Bar**
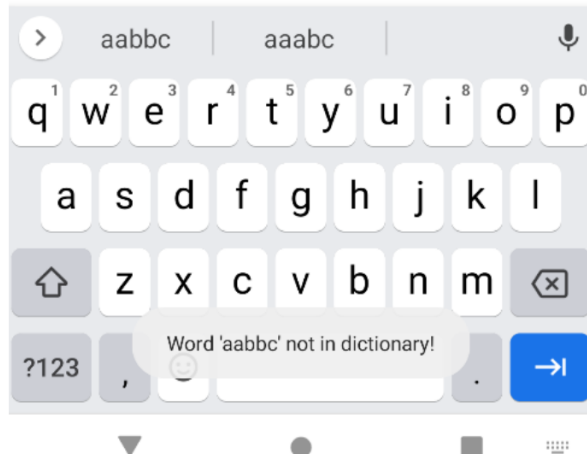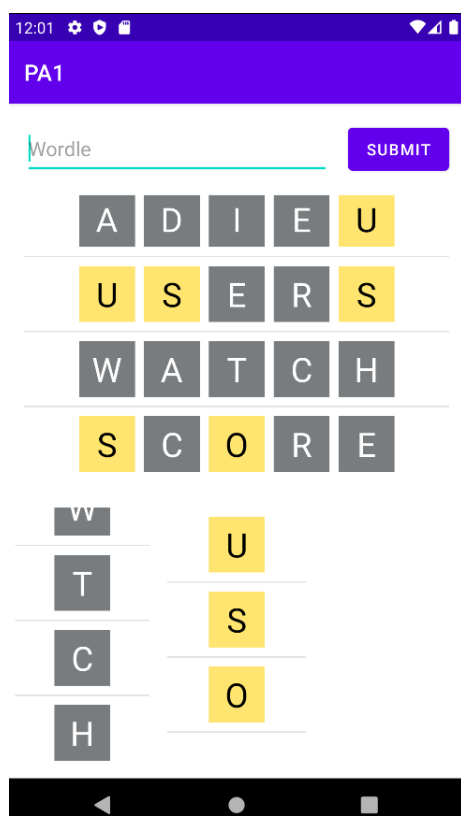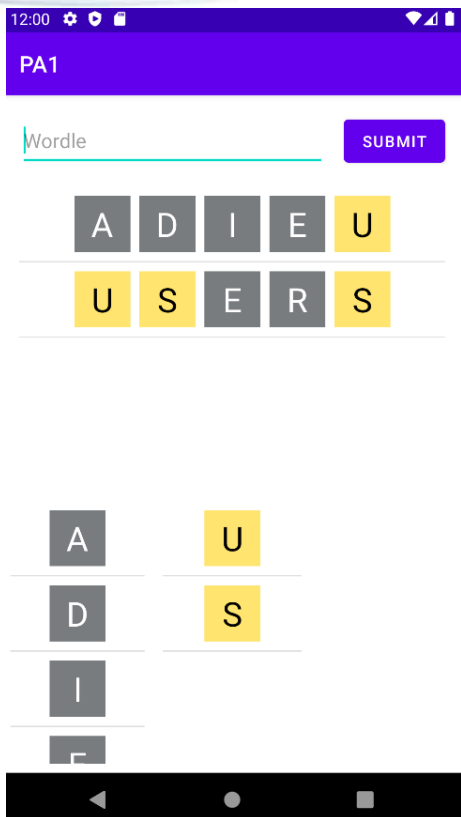
DO NOT use "enableEdgeToEdge()"

PA1 Screenshots

PA1 Screenshots

PA1 Screenshots

PA1 Submission

- Submit single zip file with name **"<Student ID>_pa1.zip"**
  - Shift Twice ⟹ search "export" ⟹ Export to zip
    ⟹ Change file name and select location to save
  - Do not care about ending '-<Number>' (ex: 2023524288-1.zip)
- Submission Due
  - **10/31 23:59**
  - Delayed Submission
    - **~11/02 23:59**
    - Your score will be penalized by **25% per day**.
   - e.g. **70/100, 2 day late = 70*(1-0.25*2) ⟹ 35/100.**

PA1 Tips: File Read

- Download wordle dictionary txt file.
  - on iCampus: one word in single line, line by line
  - source: https://github.com/cwackerfuss/react-wordle/blob/main/src/constants/wordlist.ts
- Change project view to "Project", not "Android"

- Right click on PA1/app/src/main
- New > Directory
- Type 'asset' and choose one
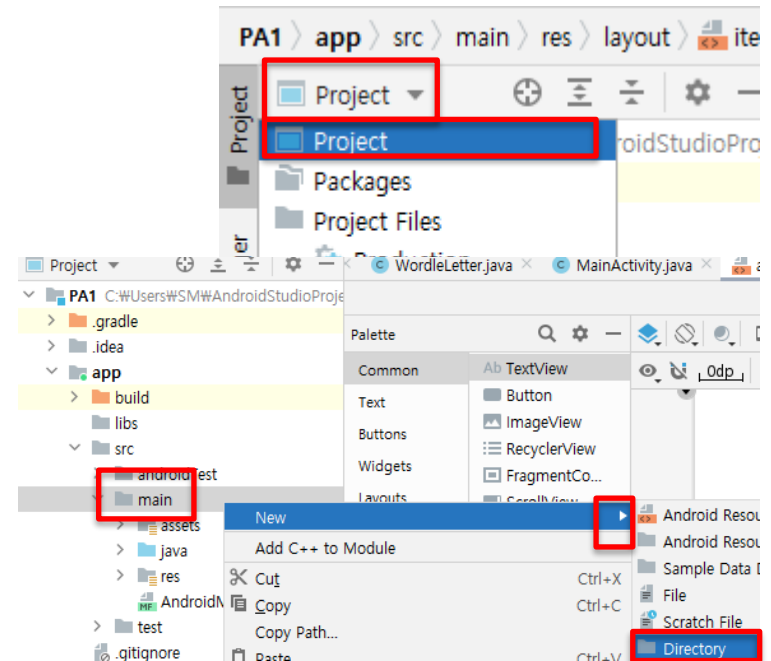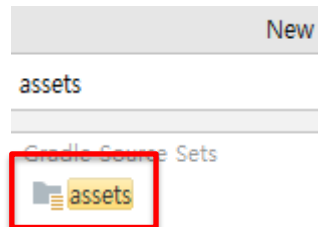
PA1 Tips: File Read

- Right click on assets
- Open in explorer

- Go into assets folder
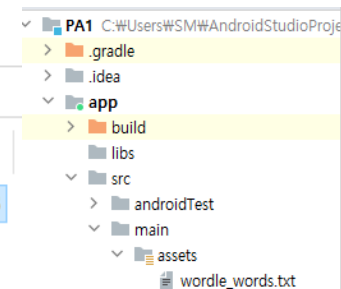- Paste that txt file in assets
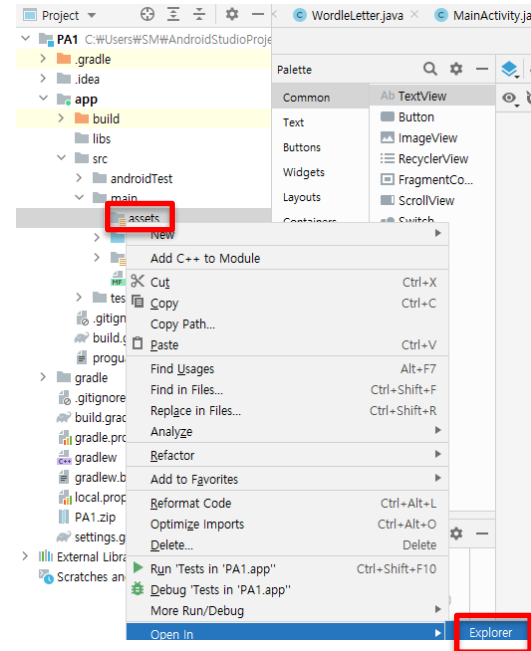
PA1 Tips: File Read

- Recommend to revert project view to "Android"
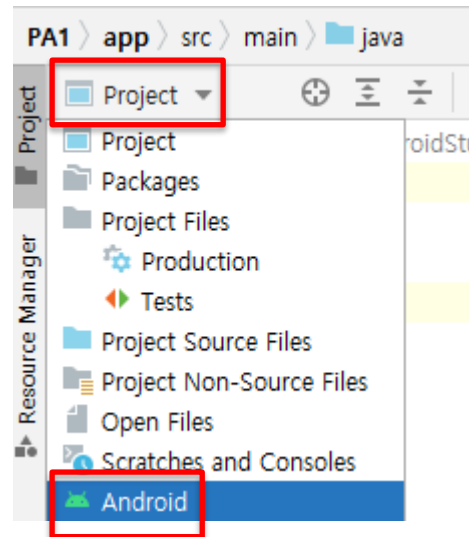
PA1 Tips: File Read

- Will open the InputStream

- Use Like
  - applicationContext.assets.open(fileName) will open file as InputStream
  - inputStream.readBytes().toString(Charsets.UTF_8) will read InputStream and convert it to String

- Further Tips:
  - https://www.baeldung.com/kotlin/inputstream-to-string
  - https://www.baeldung.com/kotlin/read-file

PA1 Tips: Others

```
■    <color name="background_out">#FF787C7E</color>
□    <color name="background_ball">#FFFFE46F</color>
■    <color name="background_strike">#FF99F691</color>
     <color name="text_out">#FFFFFFFF</color>
■    <color name="text_ball">#FF000000</color>
■  💡 <color name="text_strike">#FF000000</color>
```

- Color: app/res/values/colors.xml
  - R.color.name
  - ContextCompat.getColor(applicationContext, R.color.???)
- Adapter
  - this.notifyDataSetChanged()
- Others
  - textView.setBackgroundColor(~) / textView.setTextColor(~) / textView.text = ~
  - editText.text.toString() / editText.text = Editable.Factory.getInstance.newEditable(~)
  - android:hint

PA1 Tips: Others

- Guideline options with DP.

```xml
<androidx.constraintlayout.widget.Guideline
    android:id="@+id/guideline3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_percent="0.80" />
```

Percent

```xml
<androidx.constraintlayout.widget.Guideline
    android:id="@+id/guideline3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_end="585dp" />
```

DP