**CLIENT SIDE MODULE**

# CONTENTS

## This module has the following files:

1. MODULE\_CLIENT\_SIDE.doc
2. MODULE\_CLIENT\_SIDE\_MEDIA.zip

INTRODUCTION

In recent years, people are intrigued to know about playing music in their devices. But, sometimes just playing music out loud is not enough to entertain people nowadays. People are more exhilarated if they can play a music game on their devices.

You are asked to develop a game called **Piano** using HTML and CSS and develop client-side programming using JavaScript. Some media files are available to you in a zip file. You can create more media and modify anything in the media if you want. Your game needs to be developed in a tablet resolution (960x600 pixels). In bigger resolution, the game must be centred in the screen both horizontally and vertically.

# DESCRIPTION OF PROJECTS AND TASKS

This is a module of 4 hours. Your first 2 hours must be used to create the design of the game in three PNG images and the initial layout using HTML/CSS. Your layout should follow the design that you created. The final 2 hours you will create the functionality of game using JavaScript that allows the game to work correctly in different web browsers.

**Piano** game screen should have meet these requirements below:

1. Game title
2. 4 lane block
3. Piano keys in the bottom
4. A border between keys and lane
5. Total time elapsed
6. Total score percentage
7. Timer

**Design and Initial layout**

1. **Deliver at least 3 PNG images that present:**
   1. Game instruction: The first screen of the game presents the game icon, instructions of gameplay, a select option to choose the music to play, and the Play button.
   2. Game screen: It must present all the elements described above in the game screen.
   3. Game icon which represents the game immediately.
2. **Develop the initial markup (HTML + CSS) of your game application.** Overall screen must be within 1024x1024 pixels and centered on the screen.
3. **The design should be delivered in dark mode color.** You are free to choose dark color as long as it has the user convenience.
4. **Piano keys should be blue colored**. The background for the keys must be #3e8db1 and the hover state must be #66CFFF.
5. **You are free to decorate** the game screen design as long as it meets the requirements.
6. **The HTML and CSS** code must be valid in the W3C standards for HTML5 and CSS3 rules in accordance with the WCAG and standard ARIA (Accessible Rich Internet Applications Suite)

You are given music.json file for every song provided.

**Game functionalities**

1. **Show the game icon** as a favicon.
2. **Show game instruction** in the centre after page are loaded.
3. **Show select field for player** to choose song. The song list are included in media files.
4. **Player can start game** after click “Play” button at the bottom of game instruction.
5. **The “Play” button should be disabled** if the user did not select the song.
6. **Show countdown for three seconds in the center of screen** after user clicked the play button before the game started playing.
7. **Tiles should move and hit** the border at the exact given time.
8. **To click the keys**, player can use “DFJK” keys with the following (from the left key):
   1. ‘D’ for the first key
   2. ‘F’ for the second key
   3. ‘J’ for the third key
   4. ‘K’ for the fourth key
9. **The score will be increased** if player click the keys when the tiles is touched the border.
10. **Score** is served as percentage from total tiles clicked divided by total tiles passed.
11. **The tile will be vanished** when player clicked at the right timing or it passed the border without clicked.
12. **Click tolerance** for click distance is permitted for 100ms back and forth from the tile.
13. **Lights appeared above the border** when player clicked the key. The light should be appeared which is a gradient from transparent to solid color vertically. The light is located above the border which the key is clicked.
14. **Player fails** if the score below 50% in the end.
15. **Show alert after song finished** to display the score, high score, and winning status.
16. **Save the score to the browser**.
17. **Maintain your HTML/CSS and JavaScript code organized and clean to facilitate future maintenance**. Use correct indentation and comments. Use meaningful variable names and document your code as much as possible so another developer would be able to modify your work in the future.
18. The game needs to work correctly in **Google Chrome** and **Mozilla Firefox.**
19. **Player can change the tile move speed**
20. Even if the move speed increased, the tile must be **consistently** hit the border at the given time.
21. **Player can increase or decrease the speed** by clicking CTRL and (+) or CTRL and (-). An indicator should appear to indicate that the speed is changed.
22. **Player can pause** the game
23. **Press Esc** to open the pause popup. Player can change the volume by moving the range. The game should be in paused state when opening the rewind popup.
24. **Press Esc again** to continue or click the continue button.
25. **Restart button is exists in the popup** for user to restart the game.

**Display countdown from 3 at the center of screen** before playing after user continue or restart the game.

# INSTRUCTION FOR COMPETITORS

1. The media files are available in the ZIP file. You can modify the supplied files and create new media files to ensure the correct functionality and improve the application.
2. Save your design files in a folder called "**client-side/design**".
3. The entry file should be ‘**client-side/index.html’**
4. You should create additional images for each of the requested resolutions to highlight hidden elements, animations, interactions, or any additional information that will assist in the presentation of the game design.
5. Additional file names
   1. Instructions: province\_instructions\_1.png, XX\_instructions\_2.png
   2. Game board: province\_game\_screen\_1.png, XX\_game\_screen\_2.png
   3. Game icon: province\_game\_icon.png
6. Save any image source files to a folder named "**source**" inside the "**client-side/design**" folder. The source files are the files that contain the design layers, development files, i.e. xd, fig, psd, ai, or svg.
7. Save the working game to the directory on the server named "**client-side**". Be sure that your main file is called index.html.
8. You are responsible for the time management in your development. If you finalize some tasks you can continue to other tasks.

**Example**

These following images are for example purpose only. You may design your own game layout.

**Graphical user interface, chart, histogram

Description automatically generated**

*Figure 1 Gameplay Example*

Marking Scheme

|  |  |  |
| --- | --- | --- |
| SECTION | CRITERION | SCORE |
| E1 | Client Side General | 5.5 |
| E2 | Client Side Design | 1.5 |
| E3 | Game Screen | 1.25 |
| E4 | Code Quality | 1.5 |
| E5 | Client Side Elements | 1.5 |
| F1 | Game Instructions | 2 |
| F2 | Game Functionality | 13.25 |
| F3 | Game Tiles | 4 |
| F4 | Game Miscellaneous | 3.25 |
| F5 | Game Quality | 3 |