Test Project WEB DESIGN

Client Side

## Contents

This Test Project proposal consists of the following documentation/files:

1. WSC2017\_TP17\_CLIENT\_SIDE\_EN.PDF
2. WSC2017\_TP17\_CLIENT\_SIDE\_MEDIA.ZIP – Media Files

## Introduction

In recent years, the internet has become an integral part of our daily lives, enabling the dissemination of information in an inexhaustible source of content and interaction. Every day the use of games has gained a prominent role in this universe, allowing millions of people to get access to fun and entertainment quickly and free.

You should design a game called **Jewel Crush**, develop the layout using HTML and CSS and develop client-side programming using JavaScript and its open source libraries. 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 centered in the screen both horizontally and vertically.

## Description of project and tasks

This is a module of 5 hours. Your first 2.5 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 2.5 hours you will create the functionality of game using JavaScript that allows the game to work correctly in different web browsers.

Jewel Crush game board elements are described below:

1. Jewels: Elements that are spread across the 9x9 board. Player needs to align three or more jewels of the same color and shape to break it. You need to design at least 5 types of jewel.
2. Power-up Jewels: Jewels that are formed when the player successfully chains four jewels. If the chain is formed horizontally, the power-up should be a horizontal power-up jewel, while vertically formed chain should produce a vertical power-up jewel.
3. Bomb Jewels: jewels that are formed when the player successfully chains five or more.
4. Player Score: Element that shows the player’s score acquired from creating chains and destroying jewels.
5. Timer: Element that shows how many seconds the player has until the game is over.
6. Sound Button: Element that enable and disable the sounds in the game.
7. Font Size Buttons: Elements that increase and decrease the font size.
8. Logo: Add the provided logo in the game.
9. Score Gain: Element that shows when the user destroys jewels and receives score.
10. Name: Element that shows the name of the current player.

**FIRST 2.5 HOURS – DESIGN AND INITIAL LAYOUT:**

1. **Deliver at least 3 PNG image files that present:**
   1. Game Instructions: The first screen of the game presents the instructions to the player, a text field for player’s name, and the “Start Game” button. The instructions for the game are included in the media files. The “Start Game” button should be disabled if the text field is empty.
   2. Game board layout: It must present all 10 elements described above in the game screen.
   3. Ranking Table presentation: This design must present the logo of the game, “Back” button, and ranking table in the following column order: position, name, and score.
2. Develop the initial markup (HTML + CSS) of your game application. When the address is accessed (http://competitorYY.wsad.local/XX\_Client\_Side) the game is presented to the user with the game instructions and the button “Start Game”. The instructions must be presented in an animated way.

XX is your country code. YY is your workstation

1. “Start Game” buttons must have active and hover effects. The background of the buttons in hover state must be: #f19e0d. The active state must follow the example called ripple which is provided in the media files.
2. The HTML and CSS code must be valid in the W3C standards for HTML 5 and CSS 3 rules.

**FINAL 2.5 HOURS – GAME FUNCTIONALITIES:**

1. Pressing the “Start Game” button in the initial screen will begin the countdown from 3 before the game start. Player cannot do anything in this state. After the countdown reaches 0, player can start playing the game with the 9x9 board filled with random jewel in random position.
2. Player can click on any jewel to choose it. The chosen jewel should be highlighted and animated to indicate that the jewel is chosen by the player. When the player clicks on non-jewel element, the chosen jewel should be unselected and return to its normal state.
3. After player chooses a jewel, he/she can choose another adjacent jewel to swap the place of the two jewels. If three or more jewels of the same color and shape are adjacent to each other in the same line (horizontal only or vertical only), all of them will disappear (with animation), the player’s score will increase by the square number of jewels in the chain, new jewels will be generated on top of the first row, and the jewels on top of the previous ones will fall slowly, filling the empty spaces. The chaining process will repeat until no more chains are found in the game board.
4. If player fails to chain jewels, the jewels should go back to their previous position in an animated manner.
5. If player successfully chain four jewels, the game generates a power-up jewel. If the chain is formed horizontally, the power-up generated is a horizontal power-up jewel. If the chain is formed vertically, a vertical power-up jewel appears. If player successfully chain at least three jewels in both horizontal line and vertical line (T shape, L shape, or + shape) in the same jewel chain, the cross power-up will be generated.
6. If player successfully chain five or more jewels, the game generates a bomb jewel.
7. When player clicks on a vertical power-up, the whole column of where the jewel resides will get destroyed. When player clicks on a horizontal power-up jewel, the whole row of where the jewel resides will get destroyed. When player clicks on a cross power-up jewel, the whole row and column of where the jewel resides will get destroyed. Each jewel destroyed is worth 3 points.
8. When player clicks on a bomb jewel, a 5x5 square with the bomb jewel in the center of it will be destroyed. Each jewel destroyed is worth 3 points.
9. Every time the player receives points, the game must show the score received in an animated and informative manner.
10. The timer starts at 30 seconds. Each second, the timer is reduced by 1. When the timer reaches 0, the game is over and player will not be able to play the game anymore. The game will show the text “Game Over” and the player’s score, following with “View Ranking” button and “Play Again” button.
11. If the player clicks on “Play Again” button, return the player to the instruction board. If the player clicks on “View Ranking” button, show the ranking board described in the design.
12. Use the file “ranking.sql” in the media folder “php” to create the table inside your database.
13. The player name and the score need to be registered in the server through an AJAX request. Send this information to the server address and using these variables:  
    1. http://competitorYY.wsad.local/XX\_Client\_Side /register.php  
        XX is your country code. YY is your workstation number.
    2. Use the register.php file available in media folder “php”. Update the connection information in line 3 to use your username and password, update host and dbname to connect to your database.
    3. Method: post
    4. name: name of the user
    5. score: number (integer) of points earned during the game
14. The register.php will register your information in one database and will return a JSON information with a ranking of users. An example of the JSON structure is presented below:

[  
{"id":"1","name":"Player 1","score":"10"},  
{"id":"2","name":"Player 2,"score":"8"}  
]

1. The ranking needs to be ordered by the score in descending order. If more than one user has the same score, they receive the same position in the ranking. The JSON data returned by the server is not ordered, it is your task to order the data correctly to present to the user on the client side.
2. To increase the game interaction, some sound effects need to be used:
   1. background.mp3: used during the game;
   2. destroyed.mp3: played when chained jewels disappear;
   3. bomb.mp3: played when the bomb jewel activates.
3. Use your talent to increase the usability of the game as much as possible to permit a better experience for the user.
4. To improve the accessibility of the game you must have options to increase/decrease the font size in the screen for timer and score counter.
5. There must be an option to disable/enable game sounds. If the sound is disabled, none of sounds should be played. If sound is enabled all sounds must be played.
6. Your game should work without JavaScript errors or messages shown in the browser console.
7. 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
8. The game needs to work correctly in two browsers, Google Chrome and Mozilla Firefox. The game requirements will be checked in Google Chrome and compatibility will be checked in Mozilla Firefox.

**Instructions to the Competitor**

* 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. You can use any supplied JavaScript framework if you find it necessary.
* Save your design files in a folder call"**XX\_Client\_Side/XX\_design**" where XXis your country code.
* File names:
  + Instructions: XX\_instructions.png
  + Game board: XX\_game\_board.png
  + Ranking: XX\_ranking.png
* You should create additional images for each of the requested resolution to highlight hidden elements, animations, interactions, or any additional information that will assist in the presentation of the game design.
* Additional file names
  + Instructions: XX\_instructions\_2.png, XX\_instructions\_3.png…
  + Game board: XX\_game\_board\_2.png, XX\_game\_board\_3.png …
  + Ranking: XX\_ranking.png, XX\_ranking\_2.png, XX\_ranking\_3.png …
* Save any image source files to a folder named "**XX\_source**" inside the "**XX\_Client\_Side/XX\_design**" folder. The source files are the files that contain the layers, development files, ie .psd, .ai, .svg, .jpg.
* Save the working game to the directory on the server named "**XX\_Client\_Side**". Be sure that your main file is called index.html.
* The register of the game results in the server are made for a same service for all competitors. Be sure that the address is correct and the variables and formats are correct to permit a correct registration in the database. In case of error the JSON error message is returned as follows.

[

{"error":"Error Message"}

]

* You are responsible for the time management in your development. If you finalize some tasks you can continue to other tasks. The initial 2.5 hours only define what will be evaluated first.

## Marking Scheme SUMMARY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SECTION | CRITERION | JUDGEMENT marks | MEASUREMENT marks | TOTAL |
| E1 | Client Side General | 0.70 | 1.85 | 2.55 |
| E2 | Client Side Design | 0.50 | 1.50 | 2.00 |
| E3 | Client Side Welcome Screen | 0.50 | 1.65 | 2.15 |
| E4 | Client Side Animation | 1.50 | 0.50 | 2.00 |
| E5 | Client Side Ranking | 1.10 | 0.70 | 1.80 |
| E6 | Client Side Code Quality | 0.70 | 1.00 | 1.70 |
| E7 | Client Side Elements | 0.60 | 1.20 | 1.80 |
| F1 | Client Side Fuels | 1.30 | 0.80 | 2.10 |
| F2 | Client Side Game Constraints | 0.50 | 2.30 | 2.80 |
| F3 | Client Side Game Scene | 0 | 3.00 | 3.00 |
| F4 | Client Side Enemies | 0.60 | 2.20 | 2.80 |
| F5 | Client Side Final Design | 0 | 2.40 | 2.40 |
| F6 | Client Side Game Over | 0 | 2.40 | 2.40 |
| F7 | Client Side Game Quality | 0.70 | 1.80 | 2.50 |
| **Total** |  | **8.70** | **23.30** | **32.00** |