Project Plan

Project framing:

The Project is assigned to us is an individual project with few conditions.
We are free to choose what ever topic we like.
We can either work alone, in pairs or in a group of maximum three participants.

The conditions are :

1. Build a front-end application with Vanilla JavaScript
2. Apply what we have learned on this project
3. Search for more resources and use them on the project
4. Allowed to use JSON and Fonts-libraries
5. Have a user friendly UX, UI Design
6. Use ES6
7. Use GitHub and GitHub Pages

Start:

The project starts from the 20.04.2020

The deadline is 30.04.2020

The presentation which will take maximum 10 minutes for each project

Project Team :

We decided to work as a pair on this project.

Team members are : Felix Wurst & Neda Dehghan

Step 1:

Deciding and brain-storming on an idea.

Each of us will gather ideas which we like by ourselves at first.
Make research and find what we like and think is useful for us to make.
When we individually have decided on maximum a handful, we meet up with the other and present the ones we like most, to our counterpart.
When we have done this we will see which one we both like the best and can agree on.
After agreeing on the main Idea, we discuss the framework and things we like the project to have.

All of this happens on the first day.

The framework and precising the the single ideas of the Project can go into the second day.

On the second day the idea and the main actions of the project should be standing and ideally not be changed anymore.
Details of the project can and probably will be changed along the way.

Step 2:

We choose the interface we want to work on.
In our case it is Visual Studio Code.

Whatever is chosen for the project a game, an app or a website it makes sense to decide on the Design before starting.

Decide on:

- Appearance and look
- the features it will include
- how the user will interact with it
- how the game / app / website shall work

We have decided on a Jump and Run Game with a Retro Design of the 2D 8-Bit Games of the 80’s.
The graphics and colors will be designed and styled in reference to Roy Lichtenstein and Piet Mondrian.
The Player will have to conquer different levels in which she/he will have to collect all coins and avoid being burned by lava to be able to get into the next level.
If the Player touches lava, the current level is restored to its starting position, and the player may start again.

Step 3:

Planing our code.

We want to create a structured foundation for a jump and run game.
The game should be extendable.
For example if you want a different design, it should be possible to use canvas to add on a different player or background.
Or if you want special features or extra monsters, the structure is standing and you can add these on.
Our game shall be developed without canvas. Canvas shall be used as an addition if wanted.
But the main structure which is the foundation and can be reused again and again to build upon, even in creating the levels.
There shall be different moving actors in the game and some which are static.
We want the player token to be moved with the arrow keys, when they are pressed down.

Our game requires :

1. setting of the chars and their meanings for the level / actors
2. create a simple level plan
3. create a level reader
4. create a function to map through the level and be able to read it
5. that function should be able to differ between the moving and static actors of the game
6. set a status, create a state class to track the state of the running game
7. be able to detect the positions of all game actors
8. creations and settings for all the game actors
9. create function for base element creation and attributes
10. add and draw levels
11. function for scaling of the coordinates
12. drawing of the single actors with scaling and position
13. show current position of actor and redraw in new position
14. detect position of the actors to set the position of the game screen
15. set and control the movement of all single moving actors of the game
16. setting update methods so the position of the moving actors can be updated and calculated
17. set collide, overlap and touch functions for the game actors that need to do so (for example: “eating” coins)
18. set the tracking keys
19. set function to be able to run the animation and erase the frame delay
20. set function to run the level
21. set function to run the game
22. create score bar and score items (lives, coins, level, score)
23. create function to access local storage and create high score list
24. put sounds on to the single movements
25. game complete

Step 4:

Our Schedule

We have 10 days to realize our project. Including a weekend.
Day 11 is the presentation.
We have decided to work on all days including the weekend to have the maximum amount of time.

Our working hours together will be from 9:30 am – 20:00 pm
If difficulties appear or we drop back in schedule we work beyond that time, but at home by ourselves and merge on the next day.
Sunday will be working individually at home on an assigned task for that day.

When finishing the day, we brainstorm about what the steps will be for the next day, what has been good, bad or missing on this day and what we can do more efficient on the next.

Schedule :

Day 1-2 : finding a project, agreeing on it and doing the framework.

Day 3-4 : analyzing and constructing the frame code and steps for the game and start coding. Start documentation of the code.

Day 5-7 : still coding and hopefully be finished with the main code on day 7. Do testings and start doing the add-ons and extras

Day 8-9 : finish add-ons, sounds, designs, eliminate last bugs

Day 10 : final testings, beautify, create read-me and plan the presentation

Day 11 : PRESENTATION