

General Assembly

Project 1 - Get'a Wave



Get'a Wave is a game based on the game Frogger. You must travel from home to the beach without hitting the obstacles, drowning in the river or getting hit by the cleaguels. Once the first surfer arrives on the beach, the second is able to start. After each round the game runs faster until all of the 3 surfers arrive on the beach.

Game's Link

[Get'a Wave](#)

Overview and Concept

I had one week to build a grid-based game using JavaScript, HTML and CSS. My own version of frogger has three different characters, who must navigate the course using the keyboard arrow buttons while avoiding obstacles. The game increases in difficulty after each round.

Technologies used

HTML

- Header with logo.
- Grid with Cells including 8 rows and 11 columns in a total of 88 cells.
- Intro, game over and Win overlays with 'start' buttons.
- Audio element for my background music and sounds effects.
- Favicon to display in the browser tab.

CSS

- Grid using flex-box.
- CSS transitions for the overlay.
- CSS Animation for active character, start and restart buttons and "you win" text.
- `position: fixed` the overlay with a transparent background.

JavaScript

- keyUp event to move the characters.
- setInterval to move obstacles.
- Play and pause sounds.
- Hide and show overlays.
- Click events to start the game.

The approach taken:

Day 1 - As this was my first project, I didn't have much planned, I knew what I wanted, but I didn't know yet how to plan or design a wireframe.

I decided to start from searching for surfer's images, sound effects, background images, videos and Gifs for the game-over and win pop-ups.

Day 2 - I built the grid using flex-box and added the first surfer to the grid, a starting position, and keyUp function to move the surfer around.



```
function keyUp(event) {
    let newPosition = characterPositions[activeCharacter]
    //let characters move around inside the grid.
    if (event.code === 'ArrowUp' && characterPositions[activeCharacter] >= width) {
        newPosition -= width
    } else if (event.code === 'ArrowDown' && characterPositions[activeCharacter] < cells.length - width) {
        newPosition += width
    } else if (event.code === 'ArrowLeft' && characterPositions[activeCharacter] % width !== 0) {
        newPosition--
    } else if (event.code === 'ArrowRight' && (characterPositions[activeCharacter] + 1) % width !== 0) {
        newPosition++
    }
}
```

Day 3 - I added the obstacles and made them move at one specific speed using setInterval. I added the other non-moving obstacles where it was blocked for the surfer to enter and also the water obstacle where the surfer would drown and the game would be over.

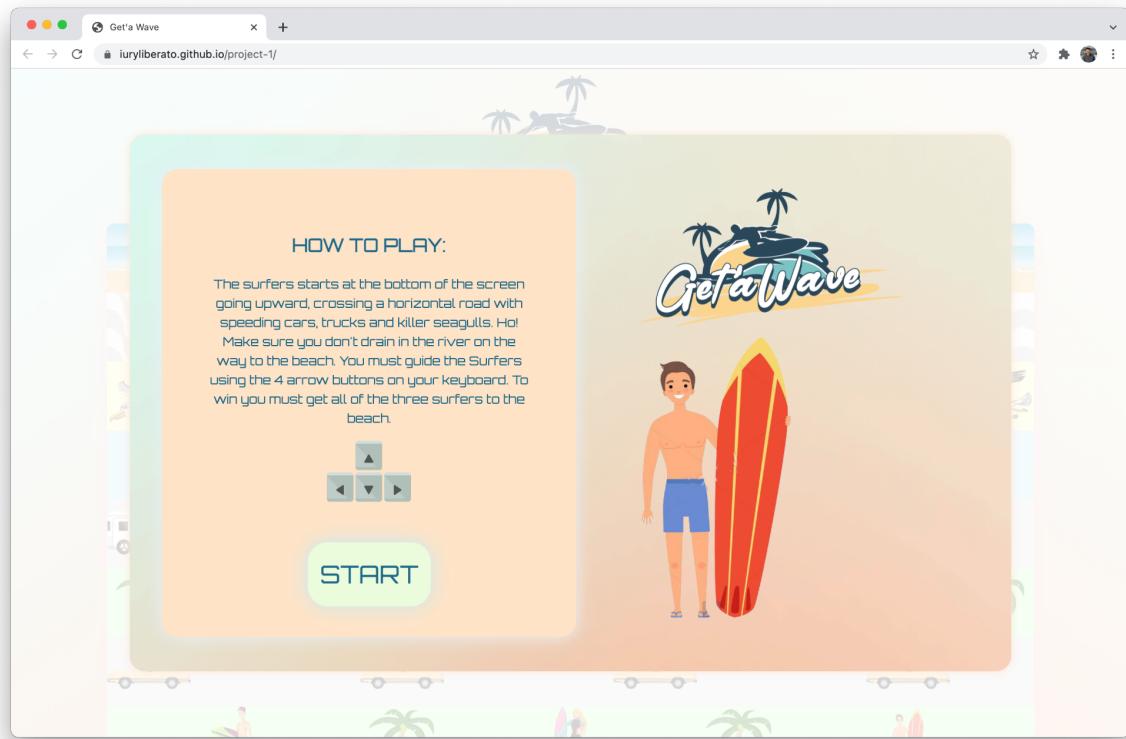
Day 4 - I added the sound effects for when the surfer moves, when the game is over caused by car traffic, seagulls or water, and gifs for game over and win. On the same day, I started building the pop-up pages for when you land on the page, when you get a game over and when you win.

```
if (!cells[newPosition].classList.contains('no-Entry')) {  
    characterPositions[activeCharacter] = newPosition  
    movesound()  
  
}  
if (cells[newPosition].classList.contains('water')) {  
    gameOver()  
    gameOverThemee()  
}  
if (cells[newPosition].classList.contains('car')) {  
    gameOver()  
    carHornPlay()  
}  
if (cells[newPosition].classList.contains('seagull')) {  
    gameOver()  
    seagullSounds()  
}  
if (cells[newPosition].classList.contains('truck')) {  
    gameOver()  
    carHornPlay()  
}
```

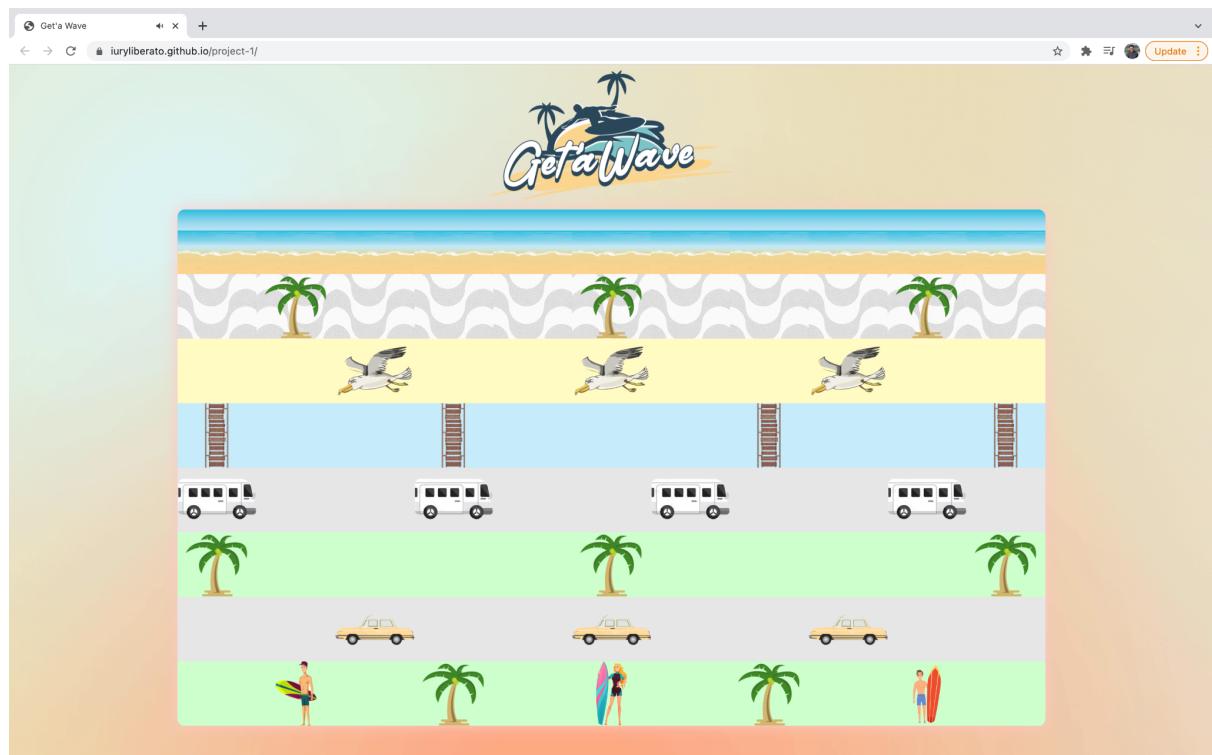
Day 5 - I finished building the pop-up pages and decided to add another two surfers to get the game going for longer and implemented faster speed after each round.

When load the page:

- Initially there is a start overlay with some instructions and the start button.



When the game starts:



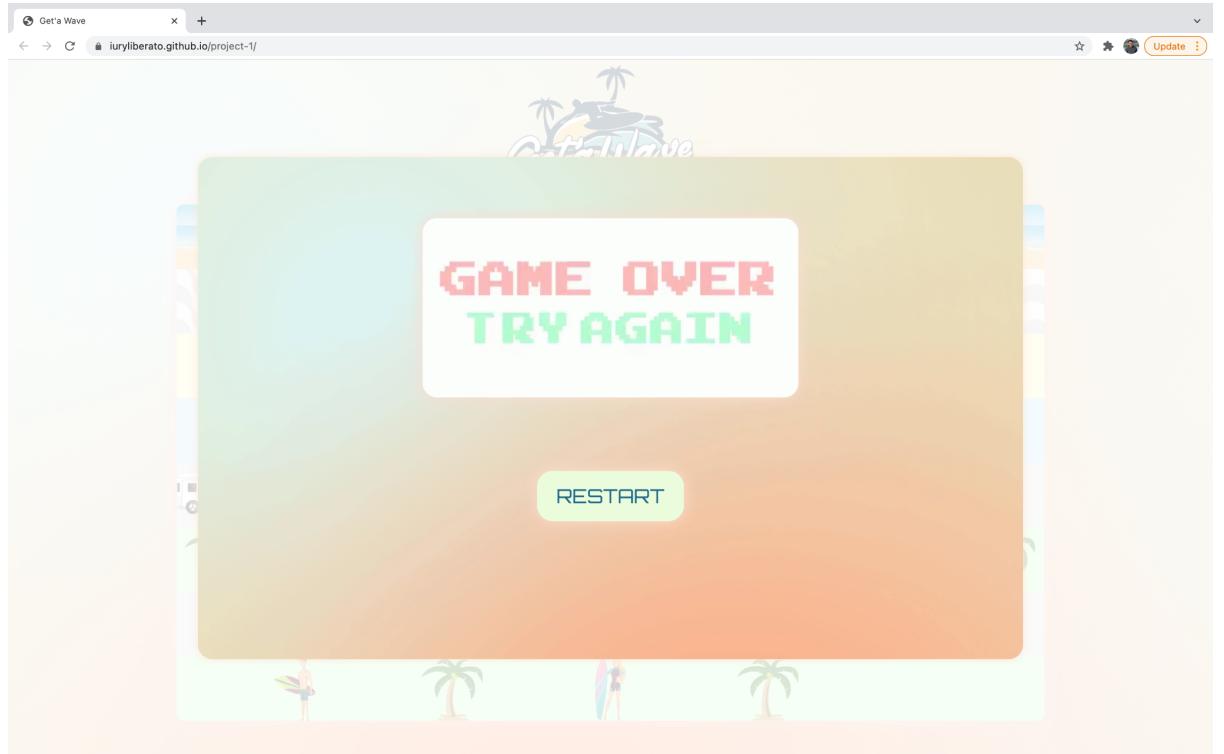
- The background music starts to play.
- The first surfer gets activated.

- The obstacles start running at a slow speed.
- Counter starts to be incremented which is used to control the movements of the obstacles, the faster you increment the counter, the faster the obstacles move.
- As the obstacles move, they compare with the character position to check if they have been hit.

When the character moves:

- The keyUp event is used to move the character up, down, left and right.
- It checks which direction the characters are going and doesn't let them go out of the grid.
- With `class="no-entry"` checks if the character is allowed to move into the new position, if not then the character does not move.
- Checks if the new position is an obstacle like water, car, truck or seagull, and if it is, then makes it Game Over.
- Once the character arrives in the first row, the character is safe and the next character gets activated.
- The traffic speeds up after each round by changing the interval time in the set interval function.
- Once all of the characters arrive on the beach (first row) is a win.
- When the user loses, a Game Over screen is displayed or if it is a win, a win screen is played.





- The last step was designing a logo to display nicely.



Key learnings:

- Learned how to use Flex-box.
- Learned how to use CSS animation.
- Learned how to create a pop-up using JavaScript, HTML and CSS.
- Learned how to use play() and pause() functions to control the audio.

Challenges:

- Disabling the other two surfers while the first one was active.
- Getting the right sound effect to stop playing after 4 seconds when winning or game over.
- Making the obstacles move at a higher speed after each round.

Future improvements:

- Create a pop-up after each of the surfers arrives on the beach, so the player knows to move their attention to the next surfer.

Bugs:

- There aren't any obvious bugs, but it would be nice to rewrite using React to see the difference between codes.