

# Project 3: Fifteen Puzzle Game

CSC 4370: Web Programming

Seung Lee

George Soetan

Chowdhury Mobin

## Use

Problem statement: Demonstrate the use of JavaScript's event handling by creating a 4x4 **fifteen puzzle** game using the DOM, along with CSS, and HTML.

Requirements: To play, the user must click “shuffle” which scrambles the tiles, and click on the tiles near the empty space to slide them. The objective is to restore the tiles to their original position within 8 minutes in order to win.

# Design: Overview of solution

To demonstrate JavaScript's event handling, we started with a separate JavaScript file that would interact with our HTML file.

-JS functions would execute events primarily through element id

```
function start_timing(elemId,endtime){  
    var timer = document.getElementById(elemId);  
    function update_clock(){  
        var time = countdown(endtime);  
        timer.innerHTML = "Time left: " + time;  
        if(time > 0){  
            setTimeout(update_clock, 1000);  
        }  
    }  
    update_clock();  
}
```

```
<script type="text/javascript">  
    window.onload = function () { start_game(); }  
    const music = new sound("sounds/timing.mp3");  
</script>
```

# Design: Key design features

Shuffle: Tiles of puzzle are randomized.

Animations (2): Sliding motion of tiles when a moveable tile is clicked.  
Puzzle changing to winning image with style changes if game is won

Game Time With Music File: Music plays at the 5 minute mark.

End of Game Notification: Once game is won, a “You Won” notification appears with total moves taken, total time taken, and the best time so far

# Design: User Interface

Mouse clicking is the only actions needed for this game.

The tiles start off already in the correct position. “Shuffle” to unscrambles the tiles. The moveable tiles will be highlighted green. Clicking the moveable tiles will show them sliding to the empty position. Music will play 5 minutes. If the game is won, it displays a blue square with a mario image.

## Fifteen Puzzle

scramble the fifteen squares by repeatedly making moves that slide squares into



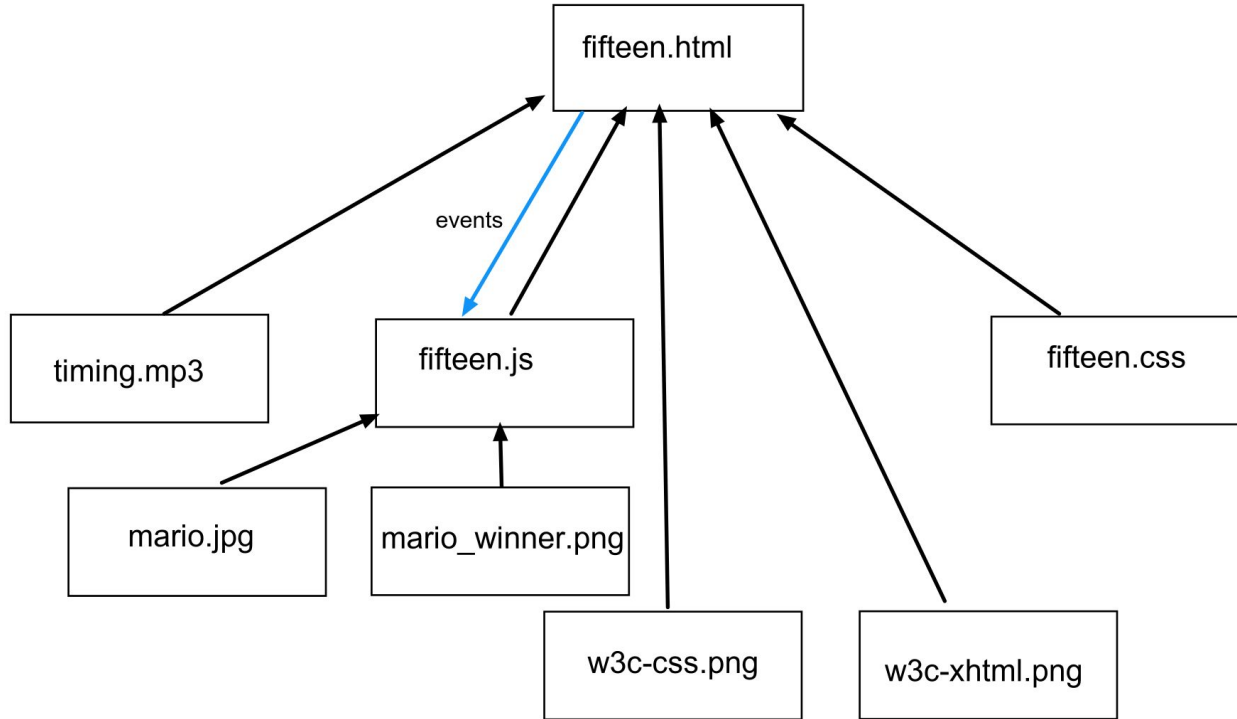
Shuffle

puzzle is to un-scramble the fifteen squares by repeatedly making moves into the empty space. How quickly can you solve it?

## You Won!



# Class Diagram



# Design: Pseudocode

## fifteen.js event handling example

Create startgame function

clearElement ('game')

//gets "game" id from HTML file and sets it to "

Set initial values including deadline

start\_timing("timer", deadline)

//using input, creates variable timer,  
sets it to the elemID "timer" id from HTML file

game = get "game" id from HTML file

//HTML file

Execute startgame using window.onload at

Beginning

...

Create a div with id = **timer**

Create a div with id = **game**

```
function clearElement(elemId) {  
    document.getElementById(elemId).innerHTML = '';  
}  
  
function start_game(){  
    clearElement('game');  
    total_moves_taken = 0;  
    total_time_taken = 0;  
    let current_time = Date.parse(new Date());  
    deadline = new Date(current_time + timer_amount*60*1000);  
    start_timing("timer", deadline);  
    game = document.getElementById('game');
```

```
function start_timing(elemId, endtime){  
    var timer = document.getElementById(elemId);
```

```
<body class="main">  
    <script type="text/javascript">  
        window.onload = function () { start_game(); }  
        const music = new sound("sounds/timing.mp3");  
    </script>
```

```
<div id="timer"></div>  
<div id="game"></div>
```

# Pseudocode

```
<div id="interface" class="buttons">
  <input type="button" id="shuffle" value="Shuffle" onclick="shuffle_tiles()">
  <p id="test"></p>
</div>
```

```
if ((x_diff) < 0 && (x_diff) != 0) { // if tile is moving to the right
  document.getElementById('test').innerHTML = "I moved right";
  if (be_fancy) { intv = setInterval(framesRight, switch_time); }
} else if ((x_diff) > 0 && (x_diff) != 0) { // if the tile is moving to the left
  document.getElementById('test').innerHTML = "I moved left";
}
```

## Under div **interface**

We create a button “**shuffle**” that calls **shuffle\_tiles()** from our js file which shuffles our tiles when we click.

“**test**” displays the direction we moved based on conditions in the js file



# Testing

## **Main test:**

- Verify appearance had no issues at start. Play game as intended for user
- Verify functionality of shuffle button, animation and color of tiles, timer, music file, and end game display

## **Bug fixing:**

- Was un-documented (other than in Github), if a problem arose – the person that discovered the issue would fix said issue.
- All tracking was done through discord and Github
- Use of the inspection tool on google chrome was heavily relied upon for verification of website functionality.

# Scrum Framework Benefits

**Scrum benefited our team through the following ways:**

- Meetings were held on discord and were around timed-boxed to 20 minutes in the evenings for 2 weeks, which kept members in touch and on track with plans
- Specific roles were discussed and assigned, which increased effective execution of tasks
- Review and retrospective of our work helped debugging and making improvements