Game 15

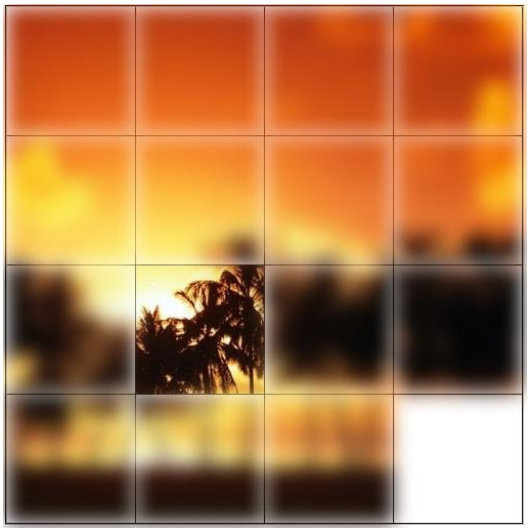
Web technologies coursework

COMSM0104

Bristol University

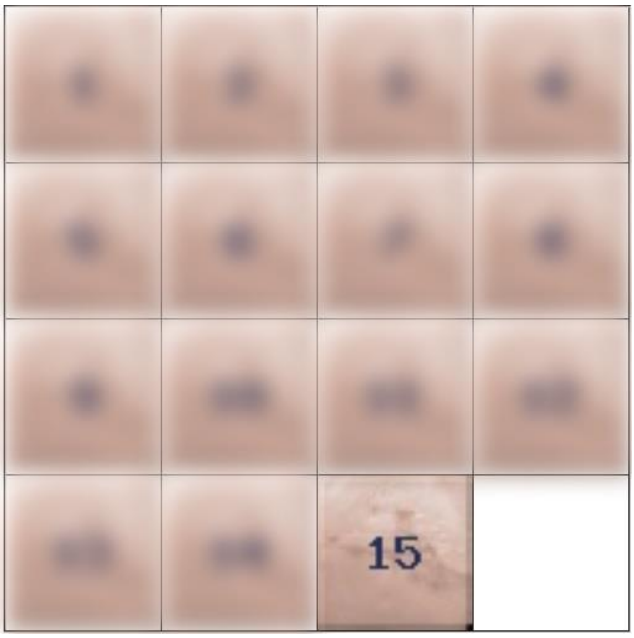
Martin V. Ivanov

mi16873



Game 15

I was inspired by this physical game of numbers we've all seen as a toy and decided to transfer it to the digital world. There are a few differences than the original game. First of all I make use of some cool css features such as blurring and transitions. Futhermor, I make few boards with numbers not available so it tests the memory of a person as well as their logical thinking. It is also a good example that covers most of the material that we studied in the Web Technologies course.



Group Work

I worked on this project on my own. Initially I had a partner, but at one point he decided to change team. He hasn't done work on this codebase.

Self Assessment

- HTML – B – Pages are well formatted with proper mark-up, but some of them have validation problems. I decided to leave some of the warning of the validator because of the large amount of java script I have to rewrite for some of the problems (for example replacing names with Id's. Or empty src of an image is also invalidated, but I populate this with java script)
- CSS – A Worked with transitions and effects. Everything works perfectly on Chrome and firefox, not so good on IE and Edge
- JS – B I wrote a lot of java script, it works well on all browsers, it is a bit messy, should have gone for framework like angular.js but for the scope of this course should be sufficient.
- PNG – B Cropped, trimmed, sliced images with Paint.NET. It wasn't so complex tasks but there were a lot of images required for the 6 different puzzles. Initially I wanted to make it automatic with node modules for image slicing, but it tuned out to be too much work so I did it by hand.
- SVG – C I first heard of Inkscape on the lectures and decided to give it a try. Generated myself a simple star logo with text inside using calligraphic writing. Its placed on the index page left side.
- Server – B I Wrote the minimum functionality for a backend node server. It's not a lot but it's nice structured and well organized code. It uses tokens for authentication and hashes the users passwords as required.

- Database – B Only 4 tables. Wanted to go a bit further but all of them are generated by the node code. The quality of the queries is good.
- Dynamic pages – X Didn't have time to make use of this otherwise good feature.
- Depth – 24 In this current state of the application I think I cover most of the requirements but I couldn't complete my initial idea – to support 2 players having a parallel game and each seeing the other player board update by signalR requests and responses. Also I want to implement high score by time of competition and few other features but for the 50 hours for this course and just before examination period this is what I could do.