Developing Web Applications - Ex. 2
Domino Game - Part 1

Daniel Cohen
ID: 204106652
Email: danielcoe@mta.ac.il

Tal Koren
ID: 201168366
Email: talkn@mta.ac.il

Basic Explanation:

The game is divided into these components:

Game - responsible for the whole game and the game state - has the playerTiles which is the array of tiles that the player holds, boardTiles which is the array of tiles that the board holds, and gameTiles which is the array of the other tiles. The Game component is responsible to check if the game is over, when and how to render the tiles on the bored, the “placeholders” which are the red square that tell the user if the place on the board is legal etc. The Game component gives props to all the other child components with the appropriate value and current state, they only render the other components according to the given props.

Board - Responsible to render the board according to boardTiles array.

Tile - Most used component in the game. Has several given props such as rotated, placeholder, rendered, empty etc. Eventually it is rendered as a div that contains two divs and a divider, each div gets a a number which is mapped to the 2 values of the tile (with a TilesMap json). Each internal div of a tile gets it’s value from the map to a certain className, then it gets the white dots with CSS.

Stock - Responsible for the stock of the game. Any tile that is not hold by the player or on the board is in the stock.

Toolbar - Holds the toolbar of information and actions of the game. It has the statistics, the actions buttons (New Game, Prev, Next, Undo) and the game timer. It also contains a message section for better UX, during the game the player is being notified with several messages of the progress of the game (Game over, Game starts) and also warning message if a certain illegal action is being made (like placing a tile before choosing a tile first).

PlayerStack - Holds the tiles that the player has. Every time the player withdraws a tile from the stock it gets to the player stack so he can place it on the board. If the stack is empty the player wins.

Implementation:

The player clicks on a tile from the stack, then the board shows red squares where he can place the chosen tile. By clicking on a certain red square the chosen tile is being placed on the board. The player can choose another tile from his stack at any time and the board will update accordingly.

Bonuses:

1. **Undo** - At any point of the game the player can choose to undo a certain move until he reaches the first move of the game.

Known Issues:

The implementation of placing doubles is not fully completed due to lack of time. When placing a double and then a regular tile vertically, others tiles will be placed horizontally.