Code assignment – Mobile developer

General Requirements:

- Task should be solved by using HTML5 and Javascript.
- Using the PIXI.js library (http://www.pixijs.com/) for game UI drawing.
- No external framework should be used.
- Your own implementation of MVC Pattern would be a great plus.
- Implement Reels Spin animation (animation you see after clicking the "Play" button and before the result of the round is displayed), it can be the simplest animation.
- Divide game client on 2 parts: framework and game. It will show your architectural skills.

Preferable (but not required):

- Designed for mobile devices (or Chrome emulation) and desktop browser.
- Implement Win Scenario animation (played in case of winning something), it can be the simplest animation.

Idea: Create the simplest slot game: 3 columns, 1 symbol each (see "The simplest layout" image below) and a "Play" button. Use image assets attached. Clicking the "Play" button randomly updates the symbols selection (3 symbols) on the screen. If they all are the same - you win (show some greetings for 1-3 sec)! You can continue playing by pressing "Play" button again and again.

Before start:

- Read about Slot Games theory, it can be something general.
- Read about Reel Strips.
- Play some Slot games (google "slot games", these games always have the "play for fun" mode), it will help you getting the general idea of what the Slot Game is.
- Slot Game basic principle: pressing the "Play" button triggers client request of the outcome from a server and starts the symbols spinning animation.

The server responds instantly, hence the outcome is known at the very beginning of the spin animation. Eventually the spin animation ends and the spin result is displayed, according to the outcome from the server response. Think how it can be used in your game.

Specification:

General

The purpose of this assignment is to examine your skills, both in terms of how you solve a problem, structure code as well as your eye for visual design (layout, animations). Feel free to add stuff you feel would enhance the game (e.g. audio, win animations etc.). The minimum requirement is a fully functional "game flow" (i.e. loading, gameplay, result), and it is really the only requirement, the way you present the game is fully up to you.

Loading phase

Resource fetching: The symbol images (path and filename) must be specified in an "external" file (for example JSON or XML), upon loading the game, the file names will be fetched from the external file by AJAX request and loaded into the game.

Game phase:

Display area:

3 symbols (the simplest layout) should be shown in the display area + "Play" button.

Play button: The "Play" button should not be enabled until the loading phase has completed, preferably this is indicated in some way. When enabled and clicked it starts the symbol switch.

Symbols switch phase: The display area should switch between the preloaded symbols, preferably some kind of animation indicates the switch (i.e. fade or scale animation) but it's not required. After some arbitrary time the switching stops and the result symbols are shown.

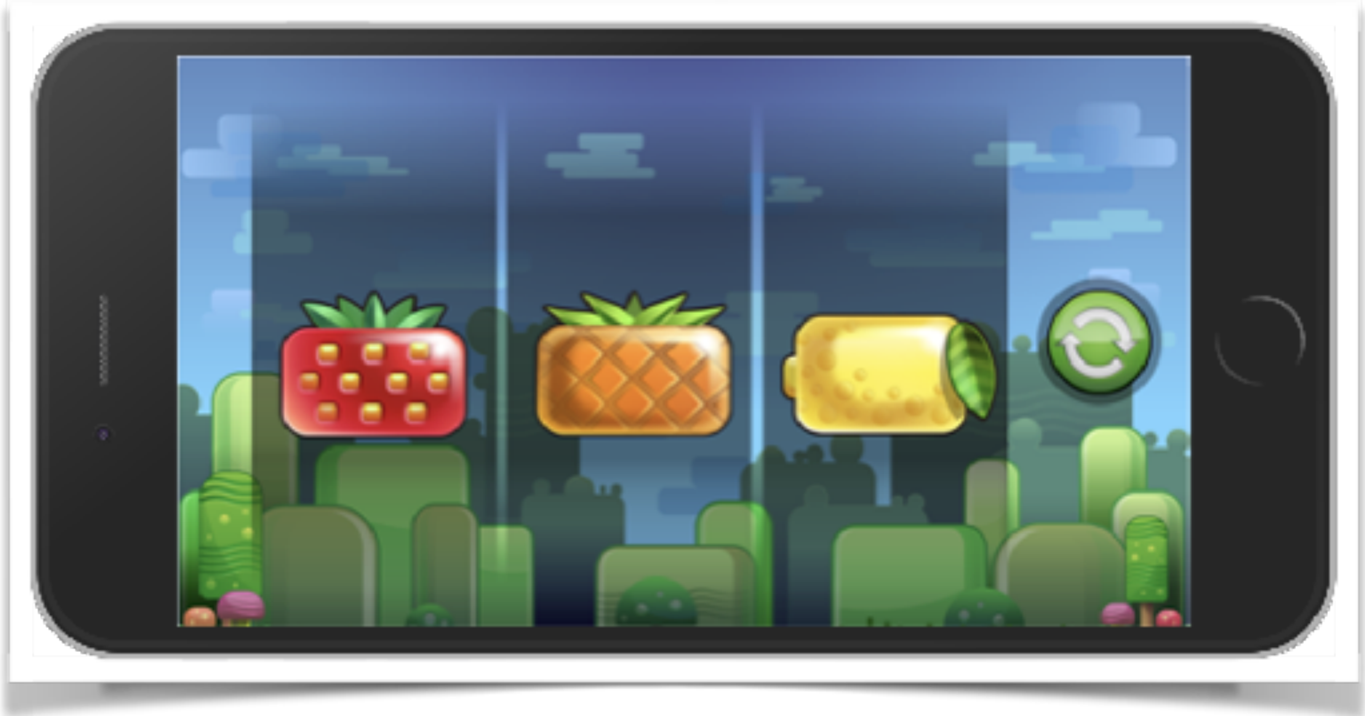
Result phase: If result symbols are all the same, we have a win situation, this should be indicated in some way.

Resources provided:

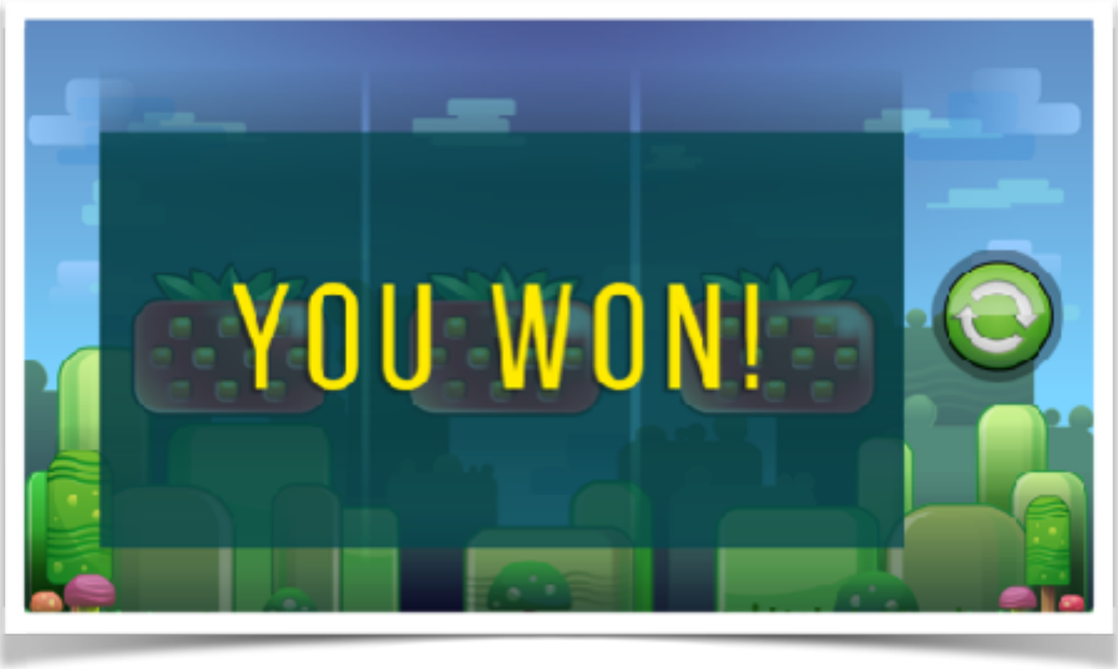
- **Img** folder containing symbol images, spin button and background.

Example layouts:

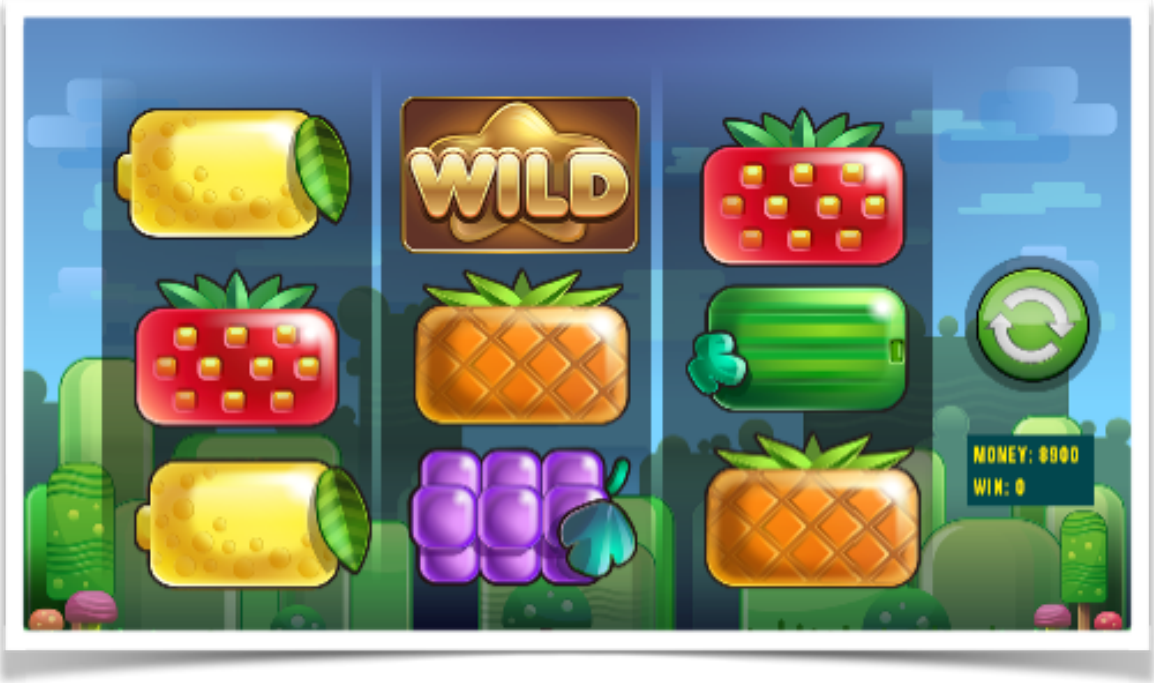
The simplest layout



Win situation layout



Advanced layout (not required)


WILD
MONEY: $900
WIN: 0