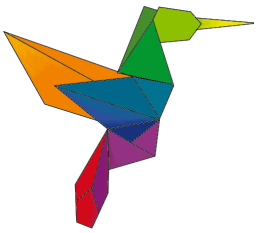


COMS32500 Web Technologies

Sandra Borlovan(sb 14766)

Introduction

The FunCode website, developed as part of this unit, consists of a learning platform targeted towards young students that wish to develop their coding skills. The website it is build not to require any prior knowledge and offers the advantages of learning an up to date programming language while getting engaged in simple and enjoyable tasks. An interpreted language, Python has a design philosophy which emphasizes code readability, its resemblance with natural english making into perfectly suitable for children. Its abstraction offers the ability to focus more on the process understanding rather than syntax.

1.Website description

1.1 Running the website

Given the requirement that certain basic frameworks depend upon, the structure of the website is as follows:

- **node modules + npm modules**
- **site**
    - **public**
        - **css files**
        - **database(db)**
        - **favicons(icon images for different operating systems)**
        - **images**
        - **js files**
    - **views (ejs/html pages)**
    - **server.js**

, the website being executed by running the command **node server.js** in the **site** folder.

1.2 Browsing the website

The website is composed of three main pages that all contain a header for navigation. In addition, the header also provides the user the ability to log in/sign in by clicking the profile icon. The **About page** (index.ejs) is designed to give a general

view of the usability and targeted audience. The **Profile page** (profile.ejs) contains the user information that is presented under three fields: User details that allows modifications of a user's data, Status of tutorials that keeps track of your work and Account Personalisation that enables the modifications of the profile picture given some alternatives. The third web page mentioned above is the **Tutorials page** (tutorials.ejs) that present the user with the tutorials available and acts as a routing center by redirecting the user to the specific tutorial selected, all pages of this type following the tutorial-structure.ejs model while the content being filled using templating.

Resources used

-express - server side framework
-ejs - templating system for dynamic pages
-express-session - user sessions (node module)
-sqlite3 - node database module
-p5.js - JavaScript library for canvas drawing
-grid system from bootstrap - front end framework features
-jquery - as a bootstrap dependency

Extra resources

-Skulpt - in-browser implementation of a Python interpreter.

2 Estimated Marks

2.1 HTML (Estimated mark: A)

The breadth and depth of the techniques used cover a broad variation that can prove the skills acquired. The HTML pages were delivered using XHTML to insure their validity along with testing it on various browsers such as Chrome, Firefox and Microsoft Edge for consistency.

2.2 CSS (Estimated mark: A)

The HTML structure was accompanied by separate CSS stylesheets that given the variations and complexity of the pages extended a big range of properties such as positioning, animations, hovering effects and additional aspects that can be viewed in the **public/css** folder.

2.3 Client-Side Javascript (Estimated mark: A)

The breadth and depth on this topic were covered by the implementation of tutorial associated games such as the Memory Game on Tutorial 2 page and Pacman Game on Tutorial 1 and 3 pages. The later game was developed using only the core p5.js JavaScript library, that is however the only part including extra resources. In addition, I implemented certain CSS styling transitions depending on the user input and the processing of forms that enable users to either log in, sign in or modify their details or profile image and sends specific POST/GET requests to the server.

2.4 Server Side Javascript (Estimated mark: B)

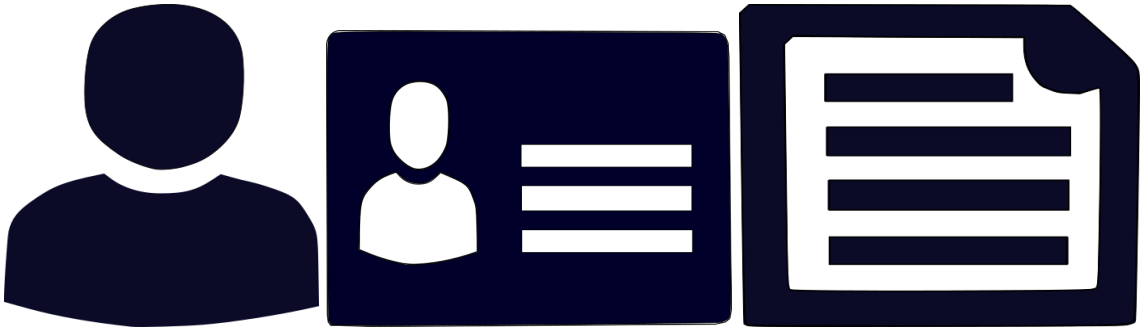
In this case the mark reflects the additional security considerations that could have been implemented to improve the server's robustness. However, the server manages, using express and express-sessions frameworks, the consistency of the user information during the user activity and its delivery on the front end part. In addition, the server supports the creation, delivery and update of data that is securely stored in a database, the access being granted by the use of sqlite3 database module.

2.5 SVG and PNG (Estimated mark: A)

The SVGs included in the website were realised in Inkscape using techniques such as:

- Drawing bezier curves tool;
- Edit paths tool
- Fill and stroke tools;
- Mirroring.

Here are the SVGs created:



The PNG were created in Inkscape for contouring and GIMP2 for filling , gradient, shades and enhanced contouring. The images created include:

-the website icon;

-the background image on the About Page;

-the gears on the About Page;

-the profile pictures available on the Profile Page

, where their designs were inspired from researching online.