# INTRODUCTION

Websites have become something that is frequently reoccurring in this day and age. When any one utters the word “Internet”, it is no exaggeration that most people would first think about a website, such as Google, Facebook, or even YouTube. This shows that the usage of websites is very dominant compared to a lot of other aspects of the Internet. There are many uses for websites, while some use it as a social network, a search engine, a video hosting site, as for the mentioned websites above, there are others who will use it for advertising, marketing, reviewing or even writing diaries called blogs. With the usage of websites multiplying each day, we will discover that most of the widely-used websites all have astonishing web designs, yet most people do not know about the structures and systems of how the website looks, or even how a website is built.

In this assignment we will learn and dive into the steps and systems of creating and designing a website. 3 languages will be used to develop the website, and they are HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), and JS (JavaScript). The website will be called “BadmintonGeek”, which is a mockup of a forum/tutorial website that is mainly focused on the sport of Badminton. The development of the website should display and elaborate basic and fundamental knowledge of web designing, which is tantamount to a construction of a good-looking and well-structured website.

# WEBSITE DEVELOPMENT METHOD

## LOGIN FORM

### HTML

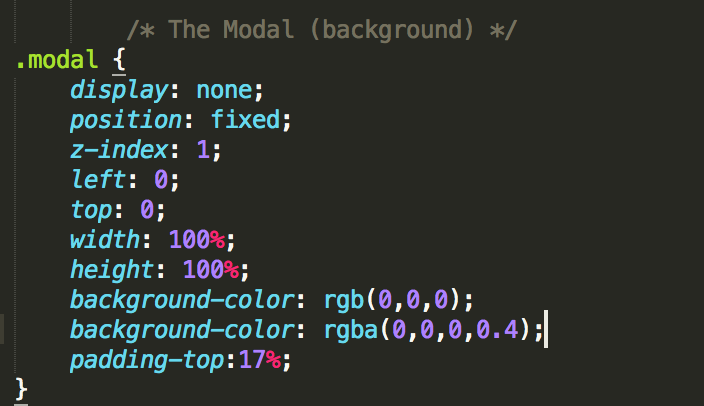


*Figure 2.1*

The login form is a floating section of the website that will be activated on the click of the login button in the top navigation bar. Once that is clicked, the form with the code as shown above will be displayed. The form has a large background made with a div that possesses the class of “modal” and the id of “id01”. Within that will be the main contents of the form that is also made by a div and possesses that class of “squarefloat” and “animate”. Within this div, there is an img element that shows the website logo and a table where all the input types are held to keep the form in order. After the table is the login button, which uses the button element and possesses the id of “btnlogin”.

### CSS



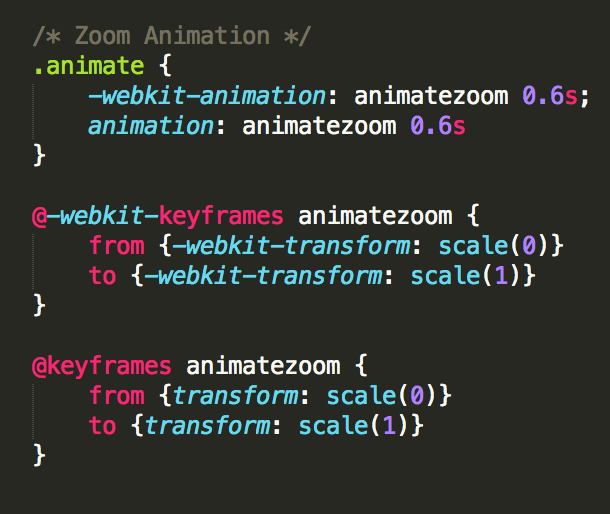


*Figure 2.2*

*Figure 2.3*

These are the visual adjustments made for the elements that are mentioned in the HTML section above. In Figure 2.2, the modal CSS class selector sets the Display property to “none”. So that the login form is hidden by default until the login button is pressed in the top navigation bar. The position is set to fixed so that the position of the login form does not change, and the z-index makes sure that the login form is above the current webpage, which makes the login form have a “floating” effect on the page. Other than that, is the width and height properties that makes sure the floating div covers the entire viewport. The background color sets up the color black and the padding-top property adds a little space on the top of the div where it is needed.

In Figure 2.3, the code deals with the main login form and its alignments, with the properties similar to the above mentioned, other different adjustments are the font-family property, which deals with a special selection of fonts, the font size, which changes the size of the wordings, and the border radii, which smoothens out and makes the borders of the form round.



*Figure 2.4*

In Figure 2.4, there is the animation of the login form that executes every time the login form shows up. The animate class calls the animation called “animatezoom”, which is defined by the keyframe to transform the scale of the form from 0 to 1.

### JavaScript



*Figure 2.5*

The code snippet as show in Figure 2.5 is an example of how the login form displays itself. When the login button in the top navigation bar gets clicked, it calls the javascript function “loginPls()”, which then sets the display property of the element by the ID of “id01” to ‘table’ instead of ‘none’. The login form will then show itself on the page and be hiding no longer. The code below is then used to close (hide) the form once again. It listens to the clicking event on the window, when the user clicks on anywhere in the window, it will check if the event target is the modal variable, which is set to be the element by the id of “id01”. If this following condition is true, then the display property of the element will be set to “none”