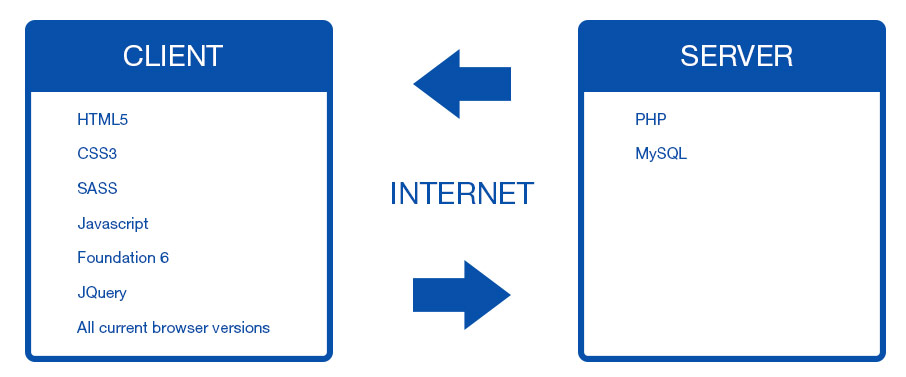
# System Design

Now that the system design has been detailed out and completed the next step in the development of Ani-Mate is the System Design of how each component will function within the overall project.

## Client Server Model

The Client Server Model is a renowned and popular model that is easy to understand of which is only one in an array of perspectives that can be used to show and layout the starting point for identifying technologies that will be used in a project.

This model is structured in way that shows you all the technologies that will be processed on the user’s client-side and what technologies will require the internet in order to process information stored on the server-side.



## Technologies

Having laid out the technologies that will be used in development and how they will correspond with each other via client and server, it is now time to explain their advantages, disadvantages and why these were chosen over other alternatives.

## Client-Side

**Foundation 6**

Is a front-end responsive framework targeted toward mobile first development that iterates and improves upon its’ previous versions such as a reduction in the amount of code by 50%, customizable SASS grid, faster prototyping, more semantic and accessible the list can go on. Its competitor could be said to be Bootstrap which is also a front-end responsive framework and from the research carried out, not all that much different from Foundation, fundamentally coming down to user preference.

Foundation is what this project will be developed with mainly down to the developers experience using the framework within the working environment and having already customized it to suit his/her workflow using node.js, npm, gulp and bower to help with setting up new projects and a great deal of flexibility.

**HTML5**

HTML5 (Hyper Text Markup Language Revision 5) was officially released as a stable W3C Recommendation on October 28th 2014 according to W3C’s timeline of the development of HTML5. It is the universal language used as the starting building blocks of any website, with its uses and implementation spreading farther and wider every day, with Apple refusing to use Flash Player back in 2011 with their line of iPhones it helped to carve the way for HTML5 development and how the web would be developed from then with regards as to what will be supported in the future. Without it the capabilities we have today would just not be, so the project will be developed upon the HTML5 guidelines for standard use and accessibility.

Since there is no other alternative to HTML5 for what it accomplishes and how universal it is, this will be used without a doubt.

**CSS3**

CSS3 (Cascading Style Sheet Revision 3) is the latest edition to CSS with it still being backward compatible with earlier versions. It has according to W3C been *“split into modules”* with new ones being added such as the Box Model, Animations, 2D/3D Transformations, Text Effects etcetera. The functionality and usage of CSS3 has been heavily implemented into today’s industry with it taking a firm seat in all of our most loved browsers like Firefox, Chrome, Opera etcetera.

Seeing as how the project is heavily based on the usage of CSS3 Animations it’s safe to say that this will be used.

**JavaScript/JQuery**

According to Wikipedia *“Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Web content production; the majority of websites employ it, and all modern Web browsers support it without the need for plug-ins.”* [Wikipedia, 2016].  
It first appeared in 1995 and has been used consistently since then in order to help make websites more interactive and a better overall user experience, nowadays with the use of JavaScript Library such as JQuery, it helps to streamline the development process down much like how SASS does the same with CSS.

The experience and materials that have been taught in regards to this language leads the developer to choose this as an implementation within the project.

## Server-Side

**PHP**

According to the PHP Website *“PHP (recursive acronym for* PHP: Hypertext Preprocessor*) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.”* [PHP Website, 2016].  
Essentially PHP is a Server Scripting Language that can be embedded in your HTML on the client but allows for communication to a server where database information may be stored and is essential to my project. My reasons for choosing PHP over another server scripting language is down to my personal experience with PHP, the fact that PHP is at the top of its field in regards to its goals and the limited timeframe available for the project itself where taking up and learning a whole new language would take up more time than what is available.

**MySQL**

MySQL (Structured Query Language) is the industry standard for accessing databases as it is open-source and allows for multiple users and great scalability. It was initially released back on May 23rd 1995 so it has had great longevity and implementation. MySQL offers paid editions of its service that have more functionally and features that are not really needed for small scale project like the one being created here.

MySQL will be the chosen Database Management System for this project because of its reputation, the experience the developer has with the platform and its features provided.

## Developer Tools

**Gulp**

Gulp will be used to help automate all the repetitive tasks that will be carried out throughout development such as with changes to the SASS file to compile any changes into ordinary CSS.

Having used Gulp in this manner with previous projects and in a workplace environment via the Foundation 6 CLI, the student is comfortable using this to help with workflow.

**SASS**

SASS (Syntactically Awesome Style Sheets) is an amazing innovation / extension on how ones workflow of CSS can be drastically improved. The introduction of variables, mix-ins and the ability to nest your CSS simply makes the whole process more streamlined. SASS’s competitor would be LESS, another CSS extension language, and from research into LESS it is not all that much different from SASS except in the way the markup is written.

The decision to ultimately go with SASS was an easy one due to the heavy use of experience with the language and its undeniable placement in today’s industry being firmly rooted.

## ER Diagram

In terms of Database design you have to initially understand what it is that needs to be created.  
An ER Diagram to the right has been made to show the relation of “Users Saved Animations”. Whenever a user saves an animation he/she has selected or customized their animation and save it, it will be stored as one of the many animations, the user can then go to their favourite animations section and choose to download or copy that animation.

## User Technical Flow Diagram

The above is a diagram detailing the flow that users will make when accessing the site along with the functionailities and features that they can interact with throughout the process.

The user can Sign-Up or Login, get directed to the main Animation Page and from here they can decide to visit their current saved animations or select an animation and either begin customizing it, download it, copy it to clipboard or just save it straight off for future use.