**The step-by-step processes that I followed throughout the project:**

* First I created a basic template for the whole project using HTML 5 and CSS 3. The template had a header, footer, body; a section for navigation and also a banner section.
* Next I added NIVO slider in the banner section and drop-down menu functionality in the navigation section using JQuery and JavaScript.
* Next I prepared two styles for the body section. One for general content covering the whole area and the other for products view with 3products in a row.
* Next I started designing the logo using HTML 5 Canvas. I decided on ‘half a start’ and the letters ‘M’ and ‘P’ in an interesting combination for “Movie Props”.
* After deciding up on the logo, I placed it just before the navigation section and adjusted the positions using CSS 3.
* Next I surfed the web for useful contents and suitable pics to be used as dummy text and product images. After finding 16 such products and their information and some other related dummy texts, I just had to stop my search.
* Next I started working on the database design. I named the database and decided upon 4 tables. I designed the tables and the necessary columns as I thought would work properly.
* Next I reflected my database design on my local MySQL server with the help of PHPMyAdmin. Then I populated the database with the dummy data I collected from the Internet.
* Next I started to think about modularity of design with PHP for my project. I decided upon a customized, much slimmed down version of MVC design pattern; where I could manage presentation, business and data logic as separately as possible and as simply as possible. I was very confused at the beginning of the project, whether this idea of mine would work or not; however it turned out okay after first few modules.
* Thus after already having the ‘js’, ‘css’ and ‘images’ folders; I created the ‘model’, ‘view’ and ‘controller’ folders and also 2 more folders: ‘config’ and ‘template’. The ‘template’ folder contains all the static parts of any page: ‘header.php’, ‘banner.php’, ‘footer.php’ and also ‘extrabodysection.php’ files created by splitting the project template designed earlier; the ‘config’ folder contains a ‘config.php’ file having some custom project-related configuration info.
* Next I started creating different modules and related pages one-by-one. According to my design, each page had one view and all requests had to pass through ‘controller.php’ in the ‘controller’ folder. In the ‘controllerhelper.php’ from the same folder; server-side logic of fetching JSON data from a web-service and some other helpful variables and functions are implemented. The ‘model.php’ file from the ‘model’ folder has all the database interaction logic implemented along with some database configuration info. All the ‘post’ and ‘get’ requests, even the AJAX requests are handled in the ‘controller.php’ file.
* Finally via section-by-section development and careful testing, I finished the project.

|  |  |
| --- | --- |
| **Technology** | **The way I have addressed them** |
| HTML 5 | I aimed to create only one page that would work as a template for the whole project. I took help from [www.w3schools.com](http://www.w3schools.com) and many other personal and web resources. |
| CSS 3 | I also took help from [www.w3schools.com](http://www.w3schools.com) here. I took ‘reset.css’ from online to start with clean slate. I applied styles section-by-section and slowly created ‘style.css’ and also took help from different resources as I felt the need to. NIVO slider had its own styles in ‘nivo-slider.css’. |
| JavaScript | I used JQuery for easier JavaScript. NIVO slider came with its own JS pack. I made use of ‘superfish.js’ for making an attractive drop-down menu effect. In ‘main.js’, I wrote my own JS code. |
| HTML 5 Canvas | In ‘main.js’, I wrote the logic for creating a logo with HTML 5 Canvas. What I did was; roughly draw-up the logo on paper with dummy initial co-ordinates; next, translate and implement using JS/HTML 5 Canvas; iteratively readjust and try-out till satisfaction. Here, I took help form different online resources including [www.w3schools.com](http://www.w3schools.com). |
| AJAX | In my products-view-page (‘products.php’), customers can like any product or even unlike it; only after logging-in. These interactions demand to be processed in the background automatically. Thus I used AJAX to ‘post’ necessary info to my ‘controller.php’ and also received and processed responses sent-back by ‘controller.php’; all of these inside a simple script-block of type JavaScript, residing in the same page. |
| PHP and MySQL | I used a personalized way of modularizing PHP code with the goal to simplify and quicken development of the project. This modularization follows a very simple MVC pattern. For MySQL, I made use of ‘mysqli’ library from PHP; mostly in Object-Oriented style. |
| Web Services | I used server-side scripting to receive random movie-trivia in JSON format from the web-service: <http://webdev.student.uws.ac.uk/movie-trivia.php>. Here, I made use of PHP given CURL functions. This is implemented as a function in ‘controllerhelper.php’ within ‘controller’ folder of my project. |