# Final Report

**Names:**

Arun Harish Balasubramonian - 44340326
Tze Thong Khor - 44260394
Saisrikar Paruchuri - 44751575

# 1 Project Aim and Outcome

## **Aim**:

The proposed website, named “Student TM” is designed to support students from diverse backgrounds or levels, benefit them to organise their coursework in a well-ordered way and remind them of upcoming assignments. This website allowed students to categorise their coursework by creating reminders and sub reminders, which will assist the students in planning their coursework and allowing students to set their goals accordingly. Besides, all the reminders and sub reminders are showed to students on the main page, allowing students to check all the reminders at once.

This website supports the functionality of creating, deleting, editing reminders and, checking the due date of the reminder and, setting a location if exist for the reminder. Students can list the priority of the sub reminders by using the drag and drop features to move the sub reminders. On top, students are also able to check the milestone on their coursework by marking the sub reminder completed. In addition, this website is supported by mobile and perform the same functionalities as the desktop website. Students are required to register an account on Student TM before they can use the functionalities of the website. They must activate their account by clicking the activation link sent to their email.

## **Outcome:**

The outcome of the project is to make sure that the students can register for their courses and add notes related to their subjects. They can create reminders for their assessments also we are very careful about the security for storing passwords and user details in database. During creation of account we have used advanced technologies for showing the format of the fields if the student makes an error at any field while registering an account. Students can see all there notes that was created by them in the home page. This StudentTM website supports all the functionalities of creating, deleting, editing remainders. If their is a due for any assessment in home page it will show the assessment. Our website is responsive and it can open in any device like desktop, laptop, tablet, mobile. We have implemented an app for StudentTM it has all the features that are supported by the website.

# 2 Approach and technologies

### Database design:

Several techniques are used in database design. Entity relationship (ER) model is used for conceptual model design which is an important phase to design a successful database application. The complete ER is then map to Relational Schema. Hence the database design is done.

### General approach:

Login page of this website used post method to send the user details like username and password. The reason for using post method is because this method provides a secure way to hide the submitted data in the page address field. PHP is used to valid the submitted data, by checking the database. Similar method is used in register page. The only difference is PHP is used to store the submitted data in database. The user password is hashed and salted before it stores in database to ensure that the stored passwords have been sufficiently secured via cryptographic techniques. Besides, in PHP, prepared statement for database is used so that there are no SQL injection vulnerabilities.

### Technologies:

**Software Requirements**

**Coding Languages**

| | | |
|---|---|---|
| **Client-Side** | **:** | HTML, CSS3, JavaScript, Bootstrap 3 |
| **Web Technologies** | **:** | Ajax, JSON, JQuery |
| **Server-Side** | **:** | PHP |
| **Database** | **:** | MySQL |
| **Server** | **:** | Apache |
| **Tools** | **:** | Apache Server, PhpMyAdmin |
| **Database Server** | **:** | MySQL |

# 3 Implementation of approach/outcome

The application implemented MVC architecture to better manage the data flow. The model in the application was the MySQL (database), the controller was the PHP using inbuilt MySQLi for communication with the server and the Ajax JavaScript, the sub-controller that connects with the Model and the Views (HTML, CSS and JQuery). While, the JavaScript acts as a sub-controller it contained its own MVC architecture classes to further manage the User Interaction.

The JavaScript contained class called loadData that retrieved content from PHP and stored in as its property. The data are then represented to the users using View class that made sure the content and its location are stored (DOM Object). Any interaction made by the users on the DOM objects are listened using the controller class that inherited the model and the view. The controller sends in the ajax request to PHP based on the location within the DOM of user interaction (Events) and changes in the DOM Object, for which the HTML contenteditable was used, through the Model class. The model class fetches the content in the format of JSON. The view inherited by the controller was called on any data changes that were made and updated accordingly.

The view used different CSS3, Bootstrap and JQuery features to update the content and for general design. These features enabled the application to be responsive to different screen resolution and especially to be compatible with the mobile screen layout. The goal of the application was to use minimal JQuery/JavaScript and use CSS3 to achieve the desired outcome. For example, in order for the application to show the existing course list as an overlay a simple attribute change in the body tag made the overlay to be visible. This was achieved using event listeners and JQuery functions.

# Peer assessment:

| | |
|---|---|
| Arun Harish Balasubramonian | 33.33% |
| Tze Thong Khor | 33.33% |
| Saisrikar Paruchuri | 33.33% |

Total - 100%