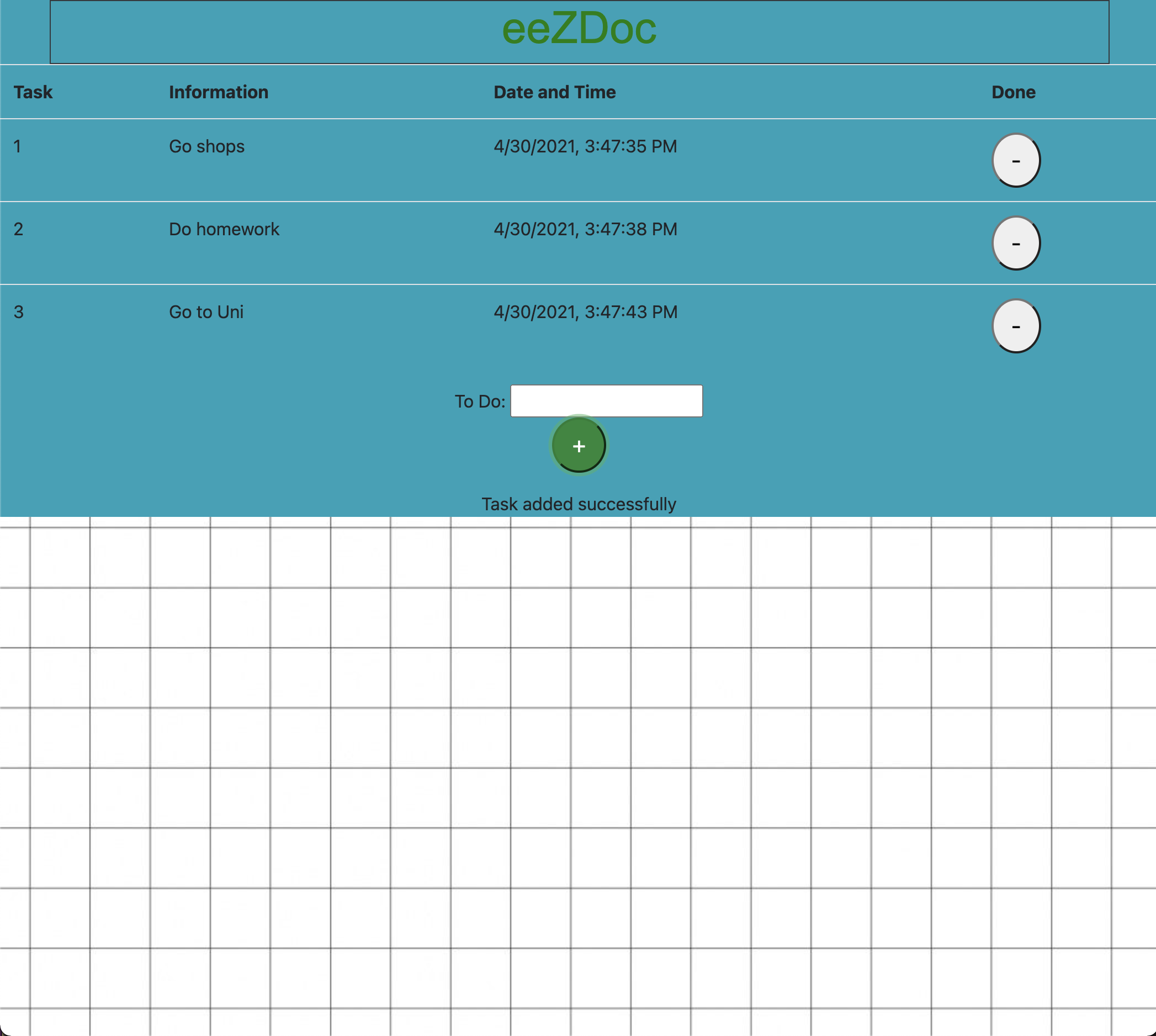
**Student Number: M00719709**

**Description**

The single page website I have built is a to-do list.

Below is a screenshot of the website I have built. The user can type a task to do in the input field and click the ‘+’ symbol to add a new to-do item. The user can also remove a to-do item by clicking the ‘-’ sign.



**Tests**

**Selenium IDE Chrome Extension**

I have used Selenium IDE Chrome Extension for tests. I have included a .side file in the ‘/tests’. I have also included screenshots in a folder in the ‘M00719709’ folder to demonstrate these tests.

I used the following tests:

* Assert Text
* Assert Not Text
* Assert Element Present
* Insert and Delete Special Characters
* Insert and Delete 3 Tasks

All the following test cases have passed and demonstrate the capabilities of the website.

**Mocha/Chai**

I have attempted to use Chai to run a Mocha test. I ran three tests. However, the tests have failed.

**Functionality**

The single page application is backed up to a MySQL database.

**Front End**

The front end will be a single html page. I used HTML/CSS and the client (JavaScript) for this.

I used **Bootstrap**. Bootstrap created a modern feel to the website.

I implemented **AJAX** for the to-do items to be loaded and exchanged from the client to the server. AJAX was used to perform the necessary get, insert and delete functions in the client.

Lastly, I used **jQuery**. jQuery calls functions upon events such as the ‘addTask()’ function when the ‘+’ button is clicked.

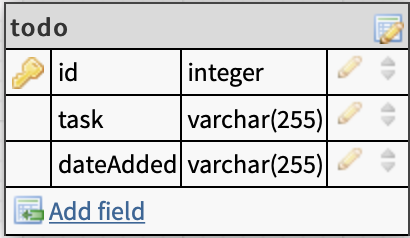
**Back end**

I used **Express**. Express was used to respond to the client’s get, insert, and delete requests of the to-dos.

In addition to this, I used **body-parser** parsed the body of the HTTP requests and perform the get, insert and delete functions. Commands such as ‘request.body.task’ reduced the time it took to build the website’s backend and made the code look cleaner.

**Below is the final database design.**

* Each to-do item has an id, task, dateAdded attribute. The primary key is the ‘id’ attribute to avoid duplication of task names



To conclude, the node website I built is complete and the HTML/CSS tests all passed. The website is simple and works with the database for, get, insert and delete functions.

A future approach to this task would be to pass the Mocha tests. I could have also included more complex functionalities such as a user login and registration and a personal to-do list for each user.