# Artefact 2:

The backend is managed using node.js – a Javascript backend.

The server starts in the index.js file in the root directory, which near the end of the project looked like appendix 1.

Setting this up required research into node.js as the technology was not familiar to anyone in the group.

Node.JS has various modules that can be imported and used, which made it very flexible and usable, which is why we chose to use it for our project.

The modules we used were:

1. Express – Runs the server
2. bodyParser – Gets the data from GET and POST requests
3. nodemailer – Sends emails to users
4. pg – Manages the connection to the PostgreSQL Database

To declutter the files as much as possible, I decided to give each “feature” its own backend routing file, as well as its own website folder.

By the end of the design, the folder structure was something like below:

|Index.js

|studentLoginRouting.js

|studentRegistrationRouting.js

└───website

├───shared

│ ├───angular

│ ├───img

│ ├───stylesheets

│ │ └───font-awesome

│ │ ├───css

│ │ └───fonts

│ └───views

├───studentLogin

│ ├───css

│ └───views

├───studentRegistration

│ ├───css

│ └───views

…

This level of separation assisted the group in creating new features quite easily, as it was clear where to make changes and what to do.

## Angular 2 app:

Since Angular development was also completely new to the team, I had to go through a tutorial to figure out how it works.

The result of this tutorial can be seen in the angularTest folder contained in the root directory of the repository. This was to ensure that node and angular will work together properly and that our framework was suitable.

Angular app structure designed was as followed:

/website/featureFolder/

Feature.main.js

Feature.module.js

Feature.component.js

Feature.service.js

/website/featureFolder/views/

Feature.component.ejs

/website/featureFolder/css/

Feature.component.css

Where there can exist multiple .component.js or .service.js files.

This structure allows consistency everywhere and also means that the folders do not get too cluttered in terms of organization and finding what you’re looking for.

## Appendix 1:

