**PROJECT REPORT**

**ON**

**CORE CURRICULUM COURSES TABLE DESIGN**

Instructor: John F. Santore

Designed

By,

Harpreetkour Pujari

**Objective**

To make a reusable application which could make a dynamic table by mapping courses to their majors and display the details of courses.

**Introduction**

The Core Curriculum Page Table Design project has been made to make core curriculum course selection convenient for students. The application has been designed using JavaScript and Node.js framework. The current core curriculum webpage of the university website has some problems like redundancy, obsolete data. In this project, I am making a more convenient core curriculum webpage for students without using any protected university database. The data sources used/required are Bridgewater State University website’s core curriculum page and next semester course details document in text format. The first part of this project involves parsing and scraping university website to get required data. The second part is to process the data to make a dynamic table from the contents in university website. Third part is to parse a file to get information of how many of these courses are actually being offered in upcoming semester. As the required data is being dynamically fetched this project could be reused every semester.

**Requirements**

**Software required:**

**Node.js**, in this you must also install,

1. Express framework
2. Handlebars view engine
3. Cheerio module
4. Request module
5. Collections Package
6. Fs module

**Special Requirements:**

1. Must be connected to internet
2. Course status file in text format

**Files Description**

**App.js**

This is a configuration file which associates all modules, routes and view engine

**Index.js**

Folder: routes

This is the file containing all the working code. This file generates the table and sends to view engine to be displayed.

**Index.hbs**

Folder: views

Displays the table as HTML contents

**Layout.hbs**

Folder: views

This file has stylesheets associated and displays the index.hbs file contents in specified format and layout (Basically, it is layout for index.hbs file HTML data).

**Error.hbs**

Folder: views

Displays error details when error occurs

**Form.hbs**

Folder: views

This file displays the form with details of courses (like title, faculty, timing, days, status)

**Code Logic steps**

* Fetches Data from University Website’s Core Curriculum page using request module
* Loads the data using cheerio module
* Using JQueries Sorts the courses according to majors they belong to
* Saves the courses in associated major collections
* Opens the AllCourseList.txt file
* Using the course code finds all the course sessions of every single course that will be offered next semester
* Creates tables for each course description
* Maps every course to its detail table
* Creates contents to be displayed on Main table page
* Displays the Table on webpage
* Takes a Detail Content display request of a particular course
* Displays the details

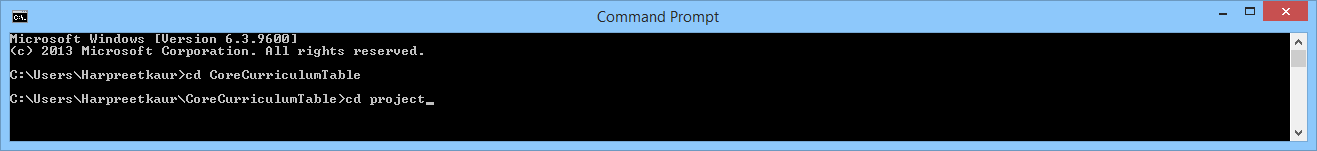
**Control Flow**

* User requests for the Core Curriculum Table Page
* Request comes to App.js, it then forwards it to index.js in routes
* The actions get performed according to logic and then the resulted contents get forwarded to view engine.
* The Main courses page gets displayed
* When user clicks a “About the course” button for a particular course.
* A post request comes to index.js
* The response gets generated for that course’s detail and sent back

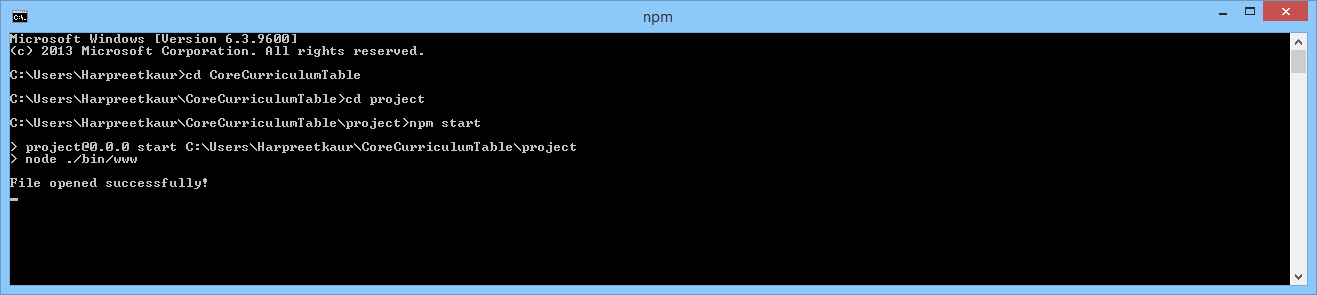
**Steps to run the application**

1. Install node.js
2. Install all the modules and packages specified in software requirement section.
3. Then open command prompt or terminal
4. Set path to project folder

* **Project** folder is inside **CoreCurriculumTable** folder. Set path to it.



1. Type in: **npm start**
2. Wait for the server to start



1. Open your browser and type in the port number(**currently 3000**) the application is running on

**Future scope**

For the purpose of more convenience, courses in the table can be associated directly with the infobear course description and registration page.

**Conclusion**

A project designed to provide ease with core curriculum selection. As the required data is being dynamically fetched this project could be reused every semester.