Course: Web Application Development
Lab Instructor: Lam Pham
Email: pqslam@hcmiu.edu.vn

Lab 7 – NodeJS and React

MySQL Server:
Create following tables

```
-- --------------------------------------------------------

--
-- Table structure for table 'course'
--

CREATE TABLE IF NOT EXISTS 'course' (
  'CourseID' bigint(20) NOT NULL auto_increment,
  'CourseName' varchar(255) NOT NULL,
  PRIMARY KEY  ('CourseID'),
  KEY 'Course_CourseID_INDEX' ('CourseID')
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

-- --------------------------------------------------------

--
-- Table structure for table 'student'
--

CREATE TABLE IF NOT EXISTS 'student' (
  'StudentID' int(11) NOT NULL auto_increment,
  'StudentName' varchar(255) NOT NULL,
  PRIMARY KEY  ('StudentID')
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;

-- --------------------------------------------------------

--
-- Table structure for table 'studentcourse'
--

CREATE TABLE IF NOT EXISTS 'studentcourse' (
  'StudentID' bigint(20) NOT NULL,
  'CourseID' bigint(20) NOT NULL,
  PRIMARY KEY  ('StudentID','CourseID'),
  KEY 'StudentCourse_StudentID_INDEX' ('StudentID'),
  KEY 'StudentCourse_CourseID_INDEX' ('CourseID')
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

NodeJS Server:

- Create a connection to your MySQL database.
- Create new api for signin (example: POST /signin/). In this assignment, the user will login successfully if username = password = StudentID.
- Create new api to retrieve all courses in table 'course'.
- Create new api to register courses.
- Create new api to retrieve all current user's registered courses.
- Create new api to update current user's registered courses.

React Server:

- Create a login page
- Create a page to register courses.
- Create a page to display current user information (student id , name, and registered courses)

Reference/Guide:

Nodejs (Backend)

What is Node.js?

- Node.js is an open source server environment
- Node.js is free
- Node.js runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- Node.js uses JavaScript on the server

Why Node.js?

Node.js uses asynchronous programming!

A common task for a web server can be to open a file on the server and return the content to the client.

Here is how PHP or ASP handles a file request:

1. Sends the task to the computer's file system.
2. Waits while the file system opens and reads the file.
3. Returns the content to the client.
4. Ready to handle the next request.

Here is how Node.js handles a file request:

1. Sends the task to the computer's file system.
2. Ready to handle the next request.
3. When the file system has opened and read the file, the server returns the content to the client.

Node.js eliminates the waiting, and simply continues with the next request.

Node.js runs single-threaded, non-blocking, asynchronous programming, which is very memory efficient.

What Can Node.js Do?

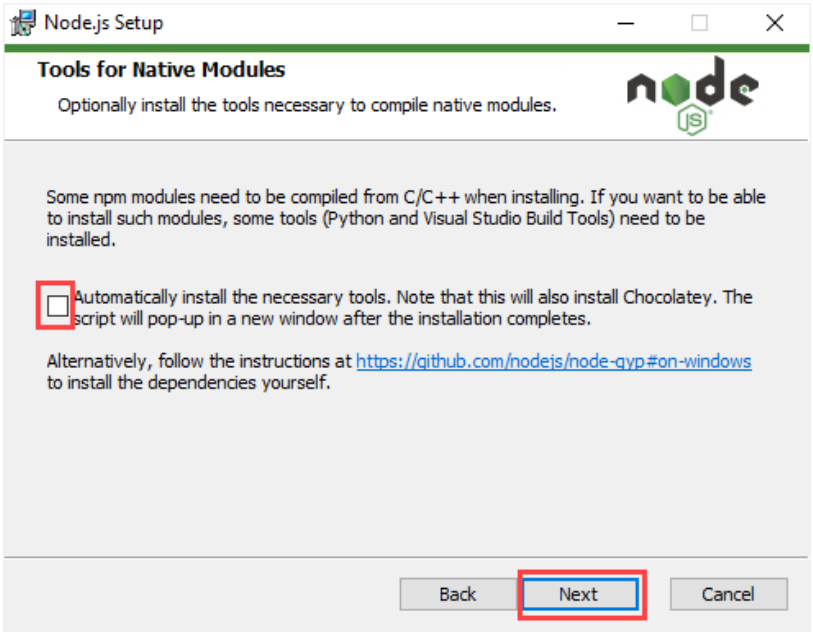
- Node.js can generate dynamic page content
- Node.js can create, open, read, write, delete, and close files on the server
- Node.js can collect form data
- Node.js can add, delete, modify data in your database

What is a Node.js File?

- Node.js files contain tasks that will be executed on certain events
- A typical event is someone trying to access a port on the server
- Node.js files must be initiated on the server before having any effect
- Node.js files have extension ".js"

Here is the very basic start of NodeJS:

1. Download Node.js
    a. The official Node.js website has installation instructions for Node.js (https://nodejs.org)
    b. If you plan to develop further with NodeJS, you can tick this box. If you just want to learn only then untick the box:



2. Getting Started

Once you have downloaded and installed Node.js on your computer, let's try to display "Hello World" in a web browser.

Create a Node.js file named "myfirst.js", and add the following code:

```
var http = require('http');

http.createServer(function (req, res) {

  res.writeHead(200, {'Content-Type': 'text/html'});

  res.end('Hello World!');

}).listen(8080);
```

Save the file on your computer: C:\Users*Your Name*\myfirst.js

The code tells the computer to write "Hello World!" if anyone (e.g. a web browser) tries to access your computer on port 8080.

For now, you do not have to understand the code. It will be explained later.

3. Command Line Interface

Node.js files must be initiated in the "Command Line Interface" program of your computer.

How to open the command line interface on your computer depends on the operating system. For Windows users, press the start button and look for "Command Prompt", or simply write "cmd" in the search field.

Navigate to the folder that contains the file "myfirst.js", the command line interface window should look something like this:

C:\Users\Your Name>_

4. Initiate the Node.js File

The file you have just created must be initiated by Node.js before any action can take place.

Start your command line interface, write `node myfirst.js` and hit enter:

C:\Users\Your Name>node myfirst.js

Now, your computer works as a server!

If anyone tries to access your computer on port 8080, they will get a "Hello World!" message in return!

Start your internet browser, and type in the address: http://localhost:8080

Please take your time to follow all of tutorials of w3schools for NodeJS:

- https://www.w3schools.com/nodejs/nodejs_get_started.asp
- https://www.tutorialspoint.com/nodejs/index.htm
- https://nodejs.dev/learn/introduction-to-nodejs (Official tutorials from NodeJS)

Here is tutorials to connect NodeJS with MySQL:

- https://www.w3schools.com/nodejs/nodejs_mysql.asp

Reactjs (FrontEnd)

What is React?

- React is a JavaScript library for building user interfaces.
- React is used to build single-page applications.

- React allows us to create reusable UI components.
- React, sometimes referred to as a frontend JavaScript framework, is a JavaScript library created by Facebook.
- React is a tool for building UI components.

To setup ReactJS, you need to have nodeJS installed first, then follow step 1 to 6 to have ReactJS and dependencies installed in your project:

- https://www.tutorialspoint.com/reactjs/reactjs_environment_setup.htm

Here are more tutorials of ReactJS:

- https://www.w3schools.com/REACT/default.asp
- https://www.tutorialspoint.com/reactjs/index.htm
- https://reactjs.org/tutorial/tutorial.html (Official tutorials from ReactJS)