Ryerson University
Faculty of Engineering & Architectural Science

Department of Mechanical and Industrial Engineering

Program: Mechanical Engineering

Course Number	CCPS 530
Section Number	01
Course Title	Web Systems Development
Semester/Year	Fall 2021

Instructor	Dr. Ghassem Tofighi

Report NO.	8

Report Title	Express and MongoDB

Group No.	N/A
Submission Date	November 30, 2021
Due Date	November 30, 2021

Name	Student ID	Signature*
Fareed Syed	xxxx19438	

(Note: Remove the first 4 digits from your student ID)

**By signing above you attest that you have contributed to this submission and confirm that all work you have contributed to this submission is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at http://www.ryerson.ca/senate/policies/pol60.pdf.*

Web Application Snips

app.js

```
var express = require("express");
var app = express();

var MongoClient = require("mongodb").MongoClient;
var url = "mongodb+srv://ryerson:123456a@chatapp.pllqz.mongodb.net/WeatherMapDB?retryWrites=true&w=majority"

app.use(express.json());       // to support JSON-encoded bodies
app.use(express.urlencoded({     // to support URL-encoded bodies
    extended: false
}));
app.use(express.static('Static'));

app.get('/', function (req, res) {
    res.write("<h1>Welcome to LAB8</h1>");
    res.write("<h1>Fareed Syed</h1>");
    res.write("<a style='color: blue' href=\"/bookinventory/list\">Inventory List</a><br>");
    res.write("<a style='color: blue' href=\"/bookinventory/add\">Add book</a><br>");
    res.end();
})

app.get('/bookinventory/list', function (req, res) {

    MongoClient.connect(url, function (err, db) {
        if (err) throw err;
        var dbo = db.db("mydb");


        db.on('error', () => console.log("Error in connecting to the Database."));
        db.once('open', () => console.log("Connected to the Database!"));

        dbo.collection("books").find().toArray(function (err, books) {
            if (err) throw err;

            if (books.length === 0) {
                console.log("User not found. Please signup.");
                res.redirect('/login')
            } else {
                console.log("Book Inventory found Successfully!");
                var html = '<p>'
                for (var i = 0; i < books.length; i++) {
                    html = html + '<b>Title: </b>' + books[i].title + '<br>';
                    html = html + '<b>Author: </b>' + books[i].author + '<br>';
                    html = html + '<b>Publisher: </b>' + books[i].publisher + '<br>';
                    html = html + '<b>Date: </b>' + books[i].date + '<br>';
                    html = html + '<b>Website: </b>' + books[i].website + '<br><br>';
                }
                html += '</p>'
                res.send('<h1>Fareed Syed: LAB8</h1><h2>List of Books: </h2>' + html);
            }
        });
    });
})
```

```
app.get('/bookinventory/add', function (req, res) {

    var html = '<br><form action="/bookinventory/addbooks" method="post">' +
        '<label for="btitle">Title:</label><br><input type="text" id="btitle" name="btitle"><br>' +
        '<label for="bauthor">Author:</label><br><input type="text" id="bauthor" name="bauthor"><br>' +
        '<label for="bpublisher">Publisher:</label><br><input type="text" id="bpublisher" name="bpublisher"><br>' +
        '<label for="bdate">Date:</label><br><input type="text" id="bdate" name="bdate"><br>' +
        '<label for="bweb">Website:</label><br><input type="text" id="bweb" name="bweb">' +
        '<br><input type="submit" value="Submit"><br>' +
        '</form>'

    res.send('Add a book: ' + html);
});

app.post('/bookinventory/addbooks', function (req, res : SuccessTextStatus ) {

    console.log(req.body);

    var new_title = req.body.btitle;
    var new_author = req.body.bauthor;
    var new_publisher = req.body.bpublisher;
    var new_date = req.body.bdate;
    var new_web = req.body.bweb;

    MongoClient.connect(url, function (err, db) {
        if (err) throw err;
        var dbo = db.db("mydb");
        var new_json = {
            'title': new_title,
            'author': new_author,
            'publisher': new_publisher,
            'date': new_date,
            'website': new_web
        };

        db.on('error', () => console.log("Error in connecting to the Database."));
        db.once('open', () => console.log("Connected to the Database!"));

        dbo.collection('books').insertOne(new_json, function (err, collection) {
            if (err) throw err;
            console.log("Book Added Successfully!")
            db.close();
        });
    });

    // books.push(new_json);

    res.send('New book is added!<br><br><a href="/bookinventory/list">List of books.</a>');
});

app.listen(3000, () => {
    console.log("Server is working!")
})
```

Testing the RESTful APIs

REST API Testing is an open-source web automation testing technique that is used for testing RESTful APIs for web applications. The purpose of rest api testing is to record the response of rest api by sending various HTTP/S requests to check if rest api is working fine or not. Rest api testing is done by GET, POST, PUT and DELETE methods. However the common tests learnt so far are GET and POST methods in this course.GET– The GET method is used to extract information from the given server using a given URI. While using GET requests, it should only extract data and should have no other effect on the data. POST– A POST request is used to create a new entity. It can also be used to send data to the server, for example, customer information, file upload, etc. using HTML forms.

Conclusion

Total time spent on this LAB was around 1 hours. 30 min was dedicated to readings, researching and learning using the Module8, Stack Overflow and blog posts. And the rest of the time was spent in writing the code. Another 15 minutes was spent on debugging the code to make sure everything works fine. Approximately 15 Minutes were spent on writing the report for this LAB. Over all the Lab was a successful and fun learning experience.

References

Thomas Hamilton, “REST API Testing Tutorial: Sample Manual Test Cases & Rest API for Testing,” *Guru99*, 8-Oct-2021. [Online]. Available: https://www.guru99.com/testing-rest-api-manually.html. [Accessed: 16-Nov-2021].