

CSEN503 Introduction to Communication Networks
Winter Term 2022
Project Description

1. Introduction

The objective of this project is to give you an experience in developing a network application based on the client/server architecture. You are required to build a web application that is used as a simple book database. The website allows the users to lookup books abstracts, genres, authors, reviews... etc. Users should be allowed to create an account, add books to their “want to read” lists and search for books. The website should initially be hosted on your PCs “local host” and when the site development phase is done, you should host it online on the cloud platform heroku.com.

2. Components:

- **Users Login (Main Page):**
Registered users should be allowed to log in to their accounts using their stored username and password. If an unregistered user tries to log in an error message should be displayed.
- **User Registration:**
Users should be allowed to create an account using a unique username and a password and the users’ information should be stored in a JSON file that represents a simple database. If the user tried to register using an already taken username, an error message should be displayed.
- **Home Page:**
The home page is the first page that should be encountered by the users when they log in to their accounts. It contains several book genres and a button to view the user’s “want to read” list. When the user clicks on any book genre, they should be redirected to that genre’s page.
- **Genre Page:**
The genre page contains all the books within this genre. When a user clicks on any book’s name, they should be redirected to that book’s page.
- **Book Page:**
The book page contains an abstract for the book. The page should also contain an embedded video describing the book which can be streamed by the user. Finally, an “add to want to read” button should be added. The button adds this book to the user’s “want to read” list in the database.

- **Want to Read List Page:**

The want to read list page contains the books that the user previously added using the “add to want to read list” button. A “view want to read list” button should be added to the home page that directs the user to their own want to read list page.

- **Search:**

A search bar should be displayed in all pages except for the registration and login pages. The search will be done using books names only. The search result is either a “book not found” message if the book was not available in the database or a list of the books that contain the search keyword in their names. The search results should be clickable and they direct you to that specific book’s page.

3. Deployment:

After developing your web application locally on your pc, you should deploy the website on an online cloud. We will be using the following technologies for the deployment phase:

3.1 Heroku

Heroku is a cloud platform that gives the developers all of the tools and infrastructure needed to deploy their applications to a production ready server. You should create an account on heroku.com. In addition, you should download and install the Heroku Command Line Interface (CLI) to make it easy for you to create and manage your Heroku applications directly from the terminal.

You can download the Heroku CLI through the following link:

<https://devcenter.heroku.com/articles/heroku-cli#download-and-install>

There are different strategies used for deploying the applications on Heroku. The simplest and most common approach is to push your code from a Git repository to a Heroku app.

3.2 Git

Git is a free and open source distributed code management software. It is a version-control system for tracking changes in the source code during software development. You can download and install it through the following link:

<https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

To set up Git for the first time, please follow the instructions on the following link:

<https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup>

4. Technologies:

- **Node.js:**

Node.js is an open source platform for executing JavaScript language outside the browser (JavaScript run-time environment). You can install Node.js through the following page:

<https://nodejs.org/en/download/current/>

When the installation is done, a tool called NPM (node package manager) is also installed. NPM is used to install node packages directly through the terminal (command prompt). To know more about NPM and the available packages visit npmjs.com/

- **Express:**

Express is a node package that is used as a web application framework. Express can be installed directly through the terminal using the command “*npm install express*”. Express is the package responsible for running the web server.

- **Visual Studio Code (VSCode):**

VSCode is the IDE that will be used for developing the web application. VSCode has built-in support for JavaScript, HTML and several other languages. Furthermore, it has a built-in terminal that can be used directly to execute commands.

- **File System:**

File system is a core module in NodeJS. It is an API that allows you to interact with the file system in your computer. You will be using the File System to create a local database for your website as a simple storage files for your data.

- **Embedded JavaScript (EJS):**

EJS is a template engine that allows the user to generate HTML with plain JavaScript. It allows you to load data from your application in the view. After the template is rendered, it generates an HTML file for the browser.

5. Provided Files:

You are provided with the .ejs (HTML) files containing the basic view for your website. The files will contain plain HTML. Therefore, it is your responsibility to connect these views (frontend) with the server logic (backend) by writing JavaScript code.

The files are uploaded on the CMS-Berlin. You should download the zipped file and then add the provided files to your project by following the instructions provided within the zip file.

6. Submission Phase:

You are required to develop a fully functional website that is hosted locally on your computers with all the features of the books website. You should use the “File System” module to save the users’ information in a JSON file as a database. Check the previous sections for all the details. After finishing the development phase of your application, you need to deploy it on Heroku cloud (please refer to the deployment section). You should be able to test and debug your work before submission to make sure it is working.

Submission Guideline: Please upload your zipped project (without the *node_modules* folder) on your google drive and share the link (**make sure to enable the link sharing for the file**) together with the Heroku link of your website and send it to: mennaallah.khalifa@guc.edu.eg

7. Resources:

- JavaScript Tutorials:
<https://www.w3resource.com/course/javascript-course.html#/>
<https://www.w3schools.com/js/>
- NodeJS and Server Installation:
<https://www.pluralsight.com/guides/getting-started-with-nodejs>
- Handling Requests to Server:
<https://www.sitepoint.com/build-a-real-time-voting-app-with-pusher-node-and-bootstrap/>

- HTML Basics:
<https://firstsiteguide.com/html-for-beginners/>
- Creating a Simple Website Using Express:_
<https://codeforgeek.com/express-nodejs-tutorial/>
- Recommended YouTube Channel:
<https://www.youtube.com/c/WebDevSimplified>