

Course: (WEB322) -Web Programming Tools and Frameworks	Assignment 3 of 6	Contribution:9%(each)-30% of course
Instructor: Haytham Qushtom	Date Given:	Date Due

Notes for the Student: This Project corresponds to Assignment 3 and is a continuation of the work you have already completed in Assignments 1 and 2.

Background: You will need to have access to an IDE or text editor and have a thorough understanding of Node.js, Express, Git, GitHub, Heroku.

Assignment Submission Requirements

- Submit the link of web application to your **new** GitHub
- Submit the link of your **new** hosted Heroku project in Blackboard. Your app name under Heroku should use the following code:

YOUR FIRST NAME+LAST 3 DIGIT OF YOUR STUDENT ID+ASSIGNMENT CODE
Ex. Haytham567A3.herokuapp.com

If you are not following this code, your submission will not be marked.

- If your application is not pushed to GitHub, it will not be marked.
- **If your web application does not function or work on Heroku, it will not be marked.**
- Marks will be deducted for any projects committed to GitHub after the deadline date.

Assignment Regulations

- This project must be done individually.
- You must create a private GitHub repository and add me as a collaborator.
- Please review Seneca's policies on Academic Integrity, specifically:

*"Each student should be aware of the College's policy regarding Cheating and Plagiarism. Seneca's Academic Policy will be strictly enforced. To support academic honesty at Seneca College, all work submitted by students may be reviewed for authenticity and originality, utilizing software tools and third party services. Please visit the Academic Honesty site on <http://library.senecacollege.ca> for further information regarding cheating and plagiarism policies and procedures.." **Thus, ensure that your code or any part of it is not duplicated by another student(s). This will result in a percentage of zero (0%) assigned to all parties involved.***

Technical Requirements

- All back-end functionality **MUST** be done using Node JS and Express.
- Your views **MUST** be created with Express-Handlebars
- You can use a CSS Framework such as **Bootstrap** to make your website responsive and aesthetically pleasing.
- You are **NOT** allowed to use any Front-End JavaScript Frameworks, specifically, **No React, No Angular**.

Detailed App Specification

This assignment is a continuation of Assignment 1 and 2, thus all the requirements for this assignment is to be made “on top” of your Assignment 1 and 2.

Note, **no database connectivity is required for Assignment 3.**

REMINDER:

All back-end functionality **MUST** be done using **Node JS and Express**.
Your views **MUST** be created with **Express-Handlebars**

Assignment 3 (9%)

Application Architecture

Your application **MUST** be structured **FULLY** according to the views, models and controllers must be separated as per the design pattern requirements.

Responsive Design

Ensure that your entire website renders well on a **variety of devices**, specifically on desktops, tablets and smartphones. To accomplish this you will need to use Bootstrap CSS Layout Module (CSS Grids).

Server-Side Validation

- You are required to implement Server-Side validation for both the login and registration form. **NO HTML5 VALIDATION OR CLIENT SIDE JS VALIDATION IS ALLOWED**
- For the login form, you are required to check for nulls (i.e, check to see if the user entered a value in the respective text fields) and the username does not contain special character.
- For the registration form, you have to check for nulls AND implement at least 3 complex validation criteria using regular expressions on all the fields (For example, enforcing that the user must enter a password that is 6 to 12 characters and the password must have letters and numbers only but the phone number must be only numbers)

Note: **NO VALIDATION LIBRARY IN SERVER SIDE IS ALLOWED. NO HTML5 VALIDATION OR CLIENT SIDE JS VALIDATION IS ALLOWED**

- Your form should not clear the data entered in the form if there are validation errors.
- **All error messages must be rendered on their respective areas near to the form element and must be styled properly.**

User Registration Form

When a user fills out the registration form and then hits the submit button, provided that all the validation criteria were not violated, your website must then **redirect the user to a dashboard page**. For now, the dashboard page should contain information welcoming the user and should be properly styled.

Git, GitHub and Heroku

Your web application must be pushed to a remote GitHub repository on your account. Please note that you must set your remote repository to private and then add me as a collaborator so I can view your web application. Lastly, you are required to deploy the working web application to Heroku.

Note: please use my email (haytham.gushtom@senecacollege.ca) to add me to your private GitHub repository.

Rubric

Criteria	Not Implemented	Partially Implemented	Fully Implemented
<ul style="list-style-type: none">Login form ValidationError messages are styled and are styled properly	0		2.5
<ul style="list-style-type: none">Registration form ValidationError messages are styled and are styled properly	0		2.5
GitHub <ul style="list-style-type: none">You made at least 4 reasonable commitsWebsite was deployed to Heroku	0		2
Responsive Design <ul style="list-style-type: none">Overall site is looks polished on all devices, specifically on smartphones, tablets and large screens	0		2

THE END