

American International University-Bangladesh (AIUB)

Course: Advanced Programming with Web Technologies (APWT)

Lecturer: Md. Al Amin (alamin@aiub.edu)

Lab Task - 03

Title: Company Management System

Description: Develop a Company Management System Application. In this project there are two types of user's. One is ADMIN and another is EMPLOYEE. An Admin can access all of the pages with page contents where Employees are not allowed to access any pages. Admin can also perform Adding, Updating & Deleting functions. Both users are need to login to access the application.

Functional Requirements:

Module Name	URL	Feature's
M1:Login	/Login	F1: Admin & Employee can login into the system through valid Username & Password. Complete validation (Express Validation) & Authentication approach will be applied.
M2:Admin	/Admin	F1: Its Admin Dashboard. Add Employee and AllEmpList button are there. Total EmpList,TypeOfEmpList are there.
	/Admin/AddEmployee	F2: Admin can Add New Employee here. Admin can set Employee's username and password. Before store into the DB perform:- <ul style="list-style-type: none">• Null Validation using JS or Express Validation• Phone Number not less or greater than 11 numbers.• Username must be greater than 8 character• Password should be equal or greater than 8 character and contains (A-Z, a-z, 0-9, and special sign like @, #, \$, & etc)
	/Admin/AllEmployeeList	F3: Admin can get all employee information in a table along with EmpID, Name, Phone, Gender, Designation and a Action field. In action field there are two links : Update & Delete.

	/Admin/Update/{id}	F4: Admin can update employee's information. Before updating perform:- <ul style="list-style-type: none"> Null Validation (using Express Validation only. Not JS Validation)
	/Admin/Delete/{id}	F5: Admin can delete any employee. Before deleting a employee a message will shown with employee's Name,Id and Phone. If Admin press Yes button then this employee will delete and if press No button then go back to <u>/Admin/AllEmployeeList</u> Url
	/Admin/Search	F6: Admin can search by EmpId or EmpName. Search result shown into a table. (Perform Ajax Searching). If search field is empty and hit the search button, then alert message will open with appropriate message.
M3: Employee	/Employee	F1: This is Employee Home page. There should be two buttons. MyProfile and UpdateProfile.
	/Employee/MyProfile	F2: Here employee can see his profile information like Name, Phone, Designation etc.
	/Employee/UpdateProfile	F3: Here employee can update his/her information. There should be a form along with form data. Employee can upload his/her profile picture. Before updating perform:- <ul style="list-style-type: none"> Null Validation using Express Validation only File operation. Uploading profile picture. Profile picture can't null Phone number should be 11 character. Not less or greater than 11 character Username must be greater than 8 character Password should be equal or greater than 8 character and contains (A-Z, a-z, 0-9, and special sign like @,#,\$,& etc.)

Technical Requirements:

1. The application should be developed using Express Web framework
2. The application should follow MVC architecture
3. Must use ejs view engine to design your Pages.
4. Write clean URLs
5. Use Different HTTP Methods
6. Create package.json file for dependency management
7. Use SESSION/COOKIE for user authentication
8. All the forms should be validated using express-validator.
9. Mysql/MongoDB can be used for data management

Topics will be covered:

- Database Connection
- HTTP methods
- Middleware
- Session/Cookie usage
- Express Validation
- CRUD operations

Submission Guideline:

1. Create a new repository name **"MID_LAB_EXAM"**.
2. Commit & Push in the repository every time you develop/fix any function/feature.
3. Your commit message should be your Lab task number +Module+Feature No. (Like: T2M1F1, T2M2F2). Here T2 means Lab Task , M1 Means Module M1 and F1 means Features 1.
4. **Your commit history will be reviewed. Maintain your commits properly.**