Students Handout

MVC-EF(Model View Controller)Workshop2

**PublicController - User Registration and Login**

**Overview:**

The PublicController is responsible for user-related actions, such as registration and login. It interacts with the IPublicService for user-related operations.

**Actions:**

**JobProviderRegistration:**

Handles user registration.

Invokes the Register method from IPublicService.

Redirects to the login page on successful registration.

**Login:**

Handles user login.

Invokes the LoginJobSeeker method from IPublicService.

Sets session variables (UserId and CompanyId) on successful login.

Redirects to the "PostJob" action in the "JobProvider" controller.

Displays an error message for invalid login attempts.

**JobProviderController - Posting and Viewing Jobs**

**Overview:**

The JobProviderController manages actions related to job posting and viewing. It uses AutoMapper to map DTOs to domain models and interacts with job-related services and repositories.

**Actions:**

**PostJob:**

Handles the posting of a job.

Uses AutoMapper to map a JobDto to a Job domain model.

Invokes the Create method from the IJobRepository.

Displays a success or failure message in TempData.

Redirects to the "AllJobs" action.

**AllJobs:**

Retrieves and displays all jobs posted by the logged-in user's company.

Uses the IJobService to get jobs based on the user's company ID.

**Key Points for Students:**

**Controller Responsibilities:**

Controllers handle user requests and interact with services and repositories.

**Action Methods:**

Understand the purpose of each action method, such as user registration, login, job posting, and viewing.

**Session Management:**

Learn how to use HttpContext.Session for storing and retrieving data between requests.

**AutoMapper Integration:**

Comprehend the integration of AutoMapper to simplify the mapping of DTOs to domain models.

**Error Handling:**

Observe how error handling is implemented in actions to handle exceptions and validation errors.

This handout provides an overview of the responsibilities and actions of the PublicController and JobProviderController in an ASP.NET MVC application. Students are encouraged to explore and test these controllers within the context of a complete MVC application.