worldskillsuk
excellence at work

WSUK – National Finals Competition
Web Development

2024

MODULE C
BACKEND DEVELOPMENT

Val Adamescu
Lewis Newton

Introduction

In this module, you are required to add functionality to an existing web application. You can write your solution using both front- and back-end technologies, including HTML, CSS, JavaScript, and PHP. You can also use any of the provided frameworks, should you wish.

The test project can be completed without using any framework; it is your decision whether you use one. If you do use any of the frameworks, the submitted solution ***must be functional as-is****;* ***no attempts will be made by the judges to fix or manipulate the code in any way.***

Please put your files in ***`XX_module_c`*** *folder where XX is your station number.*

If specific steps are required for initialising your project, please make sure that you are including a read.me file with the basic initialisation information.

You have three (3) hours to complete this task. You need to submit it before the time runs out. No additional time will be given for submission.

Requirements

WorldSkills is a global charity dedicated to promoting excellence in technical, vocational, and general education through its renowned biennial competitions, often called the "*Skills Olympics*". WorldSkills brings together experts, competitors, industry leaders, and visitors from around the world to celebrate and elevate standards across all educational fields, fostering a skilled and adaptable workforce for the future.

This year, France hosted several major events, including:

- Olympic Games Paris 2024
- 2024 Summer Paralympics
- WorldSkills Lyon 2024

These events welcomed thousands of spectators and generated a huge buzz across the country. Inspired by all three, WorldSkills hope to carry on that excitement into its future events. Following the success of *WorldSkills Lyon 2024*, WorldSkills is now preparing for its next edition: ***WorldSkills Shanghai 2026***.

User Interface Requirements

Before they can share the outcome of the competition, WorldSkills need a way of entering competitor results. To achieve this, they are building a new Competition Information System (CIS), which is slowly being developed over multiple phases.

In its first phase, the CIS should allow its target users — administrators — to manage competitor results and register new administrators to do the same. To start with, there is a minimal set of requirements. Namely, the CIS must support:

- Basic authentication (logging in and logging out)
- Registering new administrators
- Reading competitor results
- Adding competitor results
- Editing competitor results
- Deleting competitor results

You are provided with a set of routes to implement, as follows:

Route	Path	Description	Access
Landing (Index)	`/`	An introductory page, with a link to access the dashboard.	`Everyone`
Login	`/login`	A page for logging into the application.	`Guest`
Dashboard	`/dashboard`	A page providing an overview of the CIS' main functions.	`Administrator`
Administrator Registration	`/register`	A page for creating a new administrator.	`Administrator`
Competitor Results	`/results`	A page for viewing, adding, editing and deleting competitor results.	`Administrator`

Guest access means that users *do not* need to be logged in to access the route, whereas ***Administrator*** access means that users *do* need to be logged in to access the route. Both administrators and guests can access *Everyone* routes.

If a guest tries to access an ***Administrator*** route, they must be redirected to the login page (`/login`).

Each route must be accessible using the provided path, e.g., `https://{DOMAIN}/dashboard`. Ideally, the paths should match exactly, although file extensions are permitted for this initial phase, e.g., `/dashboard.html` or `/dashboard.php`.

Design

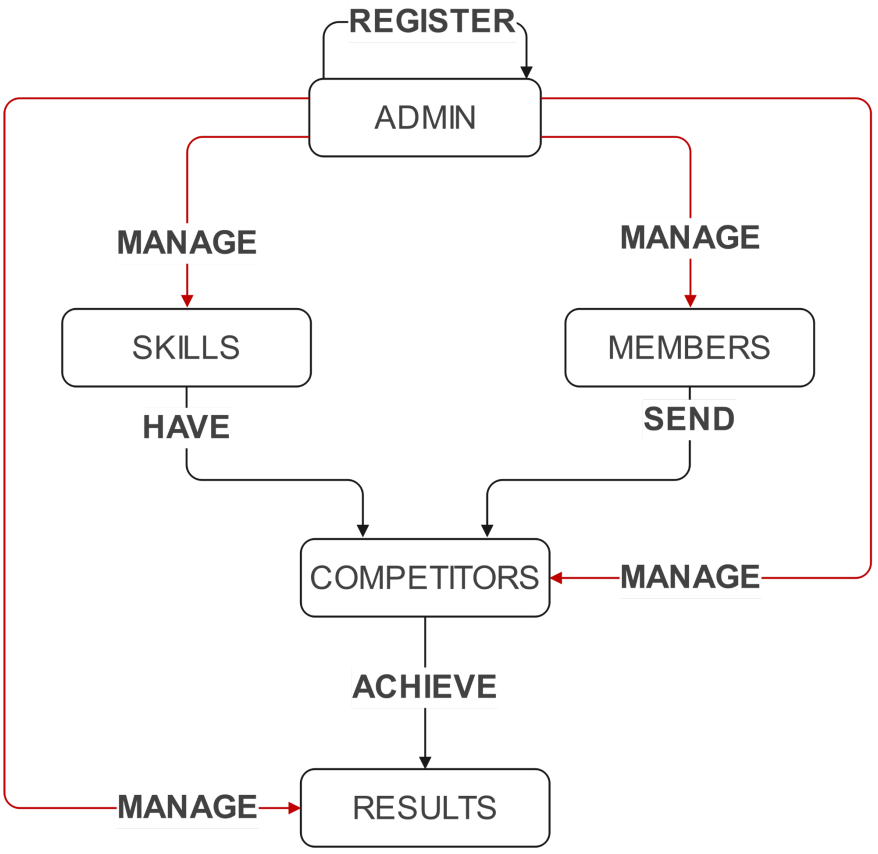
You are provided with some minimal assets, including HTML and CSS files, to get you started with some example pages.

You are expected to modify these pages to improve their usability. Aesthetics are less important, but intuitiveness is crucial.

Ensure that the interface layout is complete, consistent, and intuitive for administrators. Use clear, readable text and maintain a cohesive design across all pages to enhance usability. The interface should be fully responsive, providing an accessible experience on both desktop and mobile devices

Database design and structure

To speed up the development process, you are provided with an entity diagram (ED). Use this to get an idea of how the CIS' data can be organised.



You are also provided with SQL for creating some of the tables. However, there are issues that you need to address. Pay attention to the columns and ensure their data types and lengths are correct.

Not all tables and columns have been created yet; it is your task to add them to your own database. You must ensure that your database is in the third normal form (3NF).

Some dummy data has been prepared already. The rest is expected to be created when testing your solution.

Design your database to ensure a clear and logical structure, following best practices for data organization and accuracy. Use the provided entity diagram as a guide, ensuring that essential tables, primary keys, and relationships are correctly implemented. Choose appropriate data types and field lengths for each column, and apply constraints as needed to maintain data accuracy and referential integrity.

Security and Privacy

Security and privacy are essential considerations for the CIS database. You must store passwords securely using a cryptography method, e.g., hashing or encryption.

You must also sanitise user inputs to protect against common security threats such as cross-site scripting (XSS) or SQL injection attacks.

Pages

★ *Landing (Index)*

Path: `/`

Description: This is the introductory page of the application. It provides a brief overview and includes a link to access the dashboard.
Access: Everyone

★ *Login*

Path: `/login`

Description: This page allows users to log into the application. It includes fields for entering a username and password, and a button to submit the login form.
Access: Guest

★ *Dashboard*

Path: `/dashboard`

Description: This page provides an overview of the main functions of the Competitor Information System (CIS). It includes links to various administrative tasks.
Access: Administrator

★ *Administrator Registration*

Path: `/register`

Description: This page allows for the creation of new administrator accounts. It includes fields for entering the necessary information to register a new administrator, such as their username and password.
Access: Administrator

★ *Competitor Results*

Path: `/results`

Description: This page allows administrators to view, add, edit, and delete competitor results. It includes a table displaying the current results and forms for managing the results.
Access: Administrator

Access Control

Implement robust access control measures to ensure secure and role-specific access. Key requirements include:

- **Role-based Restrictions**: Enforce access restrictions for all pages and actions based on user roles. Redirect unauthorized users (e.g., guests) attempting to access administrator-only areas.
- **Session Verification**: Consistently verify user sessions before granting access to restricted routes, ensuring access remains secure.
- **Session Timeout**: Implement session timeouts to automatically log users out after periods of inactivity.
- **Access Logging**: Log access attempts to restricted pages, particularly any unauthorized access attempts, for enhanced security monitoring.

Authentication

The application requires session-based authentication to manage user access securely.

You are provided with specific credentials to test your application. Ensure that all passwords are securely stored in the database using appropriate encryption methods.

Registered admins created on the `/register` page must also be able to log in using the same steps.

➢ *Log In*

Upon submitting the login form, initiate a session and redirect authenticated users to the dashboard. If the login attempt fails (e.g., incorrect username or password), display an error

message informing the user of the unsuccessful attempt. Ensure error messages do not reveal sensitive information.

➢ ***Log Out***

Authenticated users should have a straightforward way to end their session. Ensure they are redirected to the login or landing page after logging out. Implement secure session handling to protect against unauthorized access after logout.

Input Validation and Data Handling

Ensure that all user inputs are validated to maintain data integrity and security across the application. Use best practices to verify data types and input length where appropriate, and safeguard against common vulnerabilities. Store sensitive data, such as passwords, securely, and design error messages to be informative without exposing system details. Additionally, implement secure session management to protect user sessions effectively.

System Management

The system should efficiently handle core tasks, including administrator registration, competitor result management, and access control. Ensure that unauthorized access attempts are seamlessly redirected to the login page. Implement clear, user-friendly error messages across all functions, including login, registration, and competitor result actions, to inform users of any issues.

Conditional Elements

Some elements do not always need to be shown. Hide these elements where they are not needed and instead only show them when it makes sense to do so.

For example, the log out button only needs to show when users are logged in.

Error messages only need to be shown if something has gone wrong, such as validation or authentication failing.

Code Quality and Maintainability

Write your code in a way that is clear, structured, and easy to maintain. Use consistent naming conventions and add comments to explain key sections or complex logic. Organize your code into functions or components where possible to enhance readability and support future updates. Avoid hard-coded values by using variables or configuration files to allow for flexibility.

Media Files

Media files are provided in the media directory. These include assets and database dummy data.

You are free to modify the assets, including the HTML and CSS files.

Do not modify the dummy data; this will be used to test your application.

You are expected to modify and add to the SQL to complete your solution.

General Guidance

Copy the provided assets to quickly get started. Focus on functionality rather than presentation.

Make sensible changes to the UI to improve usability; do not spend too much time on aesthetics.

Task Submission

Archive your project using native archive software on Windows or 7ZIP, naming the archive with your station number and module letter (e.g., ***`XX_module_c.zip`***). You need to submit it before the time runs out. No additional time will be given for submission.

If specific steps are required for initialising your project, please make sure that you are including a read.me file with the basic initialisation information.

Marking Scheme Summary

SECTION	CRITERIA	JUDGEMENTS MARKS	MEASUREMENTS MARKS	TOTAL
C1	User Interface	1	1	2
C2	System Management	1	4	5
C3	Authentication	0	3	3
C4	Access Control	1	1	2
C5	Data Validation and Security	0	3	3
C6	Database	1	2	3
C7	Maintainability	1	1	2
			TOTAL ALLOCATE MARKS	**20**