

University of Wolverhampton
Faculty of Science and Engineering
Department of Mathematics and Computer Science

Module Assessment

Module	4CS017 – Internet Software Architecture
Module Leader	Deepson Shrestha
Semester	2
Year	2022-23
Assessment	Portfolio
% of module mark	100%
Due Date	See below
Hand-in – what?	
Hand-in- where?	Canvas
Pass mark	40%
Method of retrieval	Submit the assignment reworked either before the end of the semester, or by the July resit deadline.
Feedback	Individual feedback via Canvas, in addition verbal feedback is available in class.
Collection of marked work	N/A

Learning Outcomes:

1. LO1 Understand Internet and Cloud-based software architectures and related technologies, tools and techniques.
2. LO2 Design a suitable architecture depending on a set of requirements and constraints.
3. LO3 Implements a simple distributed application using an appropriate architecture and suitable technologies.

IMPORTANT – This is an **individual** assessment.

Assignment Overview:

You are required to produce a Weather Forecast app and reflect on best practices with regards to the architecture of the application. As such you will produce 3 prototypes, reflect on the progress made, and submit a final version at the end of the module.

Important: This is an **individual** assessment.

The various prototypes should be submitted as follow:

Week 6	Application prototype 1 (20 points)
Week 9	Application prototype 2 (50 points)
Week 11	Application prototype 3 (30 points)
Assessment week	Final prototype

All the material required to complete the tasks will be covered during the lectures and workshop sessions. Detailed marking schemes (rubrics) for each prototype are available on Canvas, and are also included in the document.

Details of what is expected in each prototype is described below.

IMPORTANT!

Following weather data must be included in all the prototypes;

- 1. City name (eg: Kathmandu, Nepal)**
- 2. Day and Date (eg: Jan 20, 2023)**
- 3. Weather condition (eg: Broken clouds, scattered rain)**
- 4. Weather icon (as per the weather conditions)**
- 5. Temperature**
- 6. Rainfall**
- 7. Wind speed**
- 8. Humidity**

The Units of measurement should be metric units.

Week 6 - Application prototype 1 (20 points)

In this first version of your app, you should get a simple HTML/JavaScript page to **retrieve live weather data from the OpenWeatherMap API** and present the information suitably on a webpage.

When the weather app loads for the first time, **it should display the weather info of your assigned city**. You have to add a **search bar** to search for any desired city afterwards.

Important: the exact data to be retrieved will vary from student to student, e.g. different cities, different weather information such as temperature, rainfall, wind etc. Please refer to the City list for the data YOU need to retrieve.

You should also draw an **UML Deployment Diagram, Activity Diagram and Sequence Diagram** of your weather app's architecture.

Summary of what needs to be submitted:

- Your HTML / JavaScript page (the actual file containing the code, NOT a screenshot)
- Your UML Deployment Diagram (e.g. PNG image)
- UML Activity Diagram
- UML Sequence Diagram

Week 9 - Application prototype 2 (50 points)

In this second version of your prototype, you will implement **server-side caching** of the weather data, in order to avoid accessing the OpenWeatherMap API too often. This will be achieved with PHP and MySQL, as covered in the workshops.

Furthermore your app should be able to display the weather information of the past week, **all the weather data of the past week should be loaded from your database.**

You should also draw an **UML Deployment Diagram, Activity Diagram and Sequence Diagram** of your **new** prototype's architecture

Summary of what needs to be submitted:

- Your **updated** HTML / JavaScript page
- **Your PHP scripts**
- **Your MySQL table** (SQL script or screenshot of its structure + data)
- Your **updated** UML Deployment Diagram (e.g. PNG image), which should now include your server-side node.
- UML Activity Diagram
- UML sequence Diagram

Week 11 - Application prototype 3 (30 points)

In this third and final version of your prototype, you will implement **browser caching** of the weather data, in order to avoid network / server access when not required. This can be achieved with the localStorage JavaScript API, as covered in the workshops. You should host your weather app on a free web hosting platform and you should also enable client side caching to make the webpage accessible in **offline mode**.

You should also draw an **UML Deployment Diagram, Activity Diagram and Sequence Diagram** of your **new** prototype's architecture

Summary of what needs to be submitted:

- Your **updated** HTML / JavaScript page, showing browser caching JavaScript code.
- A screenshot of your data stored in the browser (i.e. Chrome console)
- Your PHP scripts
- Your **updated** UML Deployment Diagram (e.g. PNG image), which should now include your browser cache node.
- UML Activity Diagram
- UML sequence Diagram

Assessment week

As your final summative submission, you should submit:

- A reflective report describing your app's **final** architecture along with the UML diagrams and a link to your weather app.

Submission of work

Your completed work for assignments must be handed in on or before the due date. ***You must keep a copy or backup of any assessed work that you submit. Failure to do so may result in your having to repeat that piece of work.***

Penalties for late submission of coursework

Standard Faculty of Science and Technology arrangements apply.

ANY late submission (without valid cause) will result in 0 marks being allocated to the coursework.

Procedure for requesting extensions

If you have a valid reason for requiring an extension you must request an extension using e:vision. **Requests for extension to assignment deadlines should normally be submitted at least one week before the submission deadline** and may be granted for a maximum of seven days (one calendar week).

Retrieval of Failure

A pass of 40% or above must be obtained overall for the module (but not necessarily in each assessment task).

Where a student fails a module they have the right to attempt the failed assessment(s) once, at the next resit opportunity (normally July resit period). If a student fails assessment for a second time they have a right to repeat (i.e. RETAKE) the module.

NOTE: STUDENTS WHO DO NOT TAKE THEIR RESIT AT THE NEXT AVAILABLE RESIT OPPORTUNITY WILL BE REQUIRED TO REPEAT THE MODULE.

Mitigating Circumstances (also called Extenuating Circumstances).

If you are unable to meet a deadline or attend an examination, and you have a valid reason, then you will need to request via e:vision **Extenuating Circumstances**.

Feedback of assignments

You will be given feedback when you demonstrate your work.

You normally have **two working weeks** from the date you receive your grade and feedback to contact and discuss the matter with your lecturer. See the Student's Union advice page <http://www.wolvesunion.org/adviceandsupport/> for more details.

Registration

Please ensure that you are registered on the module. You can check your module registrations via e:Vision You should see your personal tutor or the Student Support Officer if you are unsure about your programme of study. The fact that you are attending module classes does not mean that you are necessarily registered. A grade may not be given if you are not registered.

Cheating

Cheating is any attempt to gain unfair advantage by dishonest means and includes **plagiarism** and **collusion**. Cheating is a serious offence. You are advised to check the nature of each assessment. You must work individually unless it is a group assessment.

Cheating is defined as any attempt by a candidate to gain unfair advantage in an assessment by dishonest means, and includes e.g. all breaches of examination room rules, impersonating another candidate, falsifying data, and obtaining an examination paper in advance of its authorised release.

Plagiarism is defined as incorporating a significant amount of un-attributed direct quotation from, or un-attributed substantial paraphrasing of, the work of another.

Collusion occurs when two or more students collaborate to produce a piece of work to be submitted (in whole or part) for assessment and the work is presented as the work of one student alone.