

Web Development and Databases

Practical Aim

This practical session aims to cover practical aspects of web development. Also, your tutor will introduce Website project specification and share example(s) from previous year.

Exercise 1: What software are needed and why?

We'll be using the following software:

VS Code, Python 3.8.x+, Flask, MySQL, MongoDB, WorkBench, Postman, Web server, Web browser

Take few minutes to get yourself familiarise with the above software and tools. You may use a search engine. Once you have a clear understanding why you need the above software then you should move on to next exercise.

It is important to realise students that how important is to learn installation and setting up **software development environment** so that students can develop web applications and manage data through a suitable database management system. This will not only help them to learn different technologies in this module but they'll also be able to complete their website Project.

1. **VS Code** – code editor to write your scripts. You may use other editors if you're comfortable with those editors e.g., PyCharm, Sublime text editor, Atom. There are several extensions for VS Code which we'll use e.g., MySQL extension.
2. **Python** - for server side scripting. I am using Python 3.8.3+ (labs should have Python 3.9) and examples are tested using this version.
3. **Python Flask** – Python based micro web development framework. We'll use it to write server-side scripts to implement business logic.
4. **Python packages** – We'll use various python packages including: flask, requests, mysql-connector 8.0.20, Selenium, etc. We'll install these packages when we need them.
5. **MySQL Community Server 8.0.20** – as database management system to store data securely and in structured way
6. **MongoDB** – as NoSQL database system
7. **Various Graphical User Interface Clients:** We'll use various graphical clients to work with our apps and databases. For example, **MySQL Workbench Community 8.0.20** is a visual client to work with MySQL databases. Similarly, **Postman** is a visual client to test web APIs.
8. **Web Browsers:** Of course you would need a web browser. Our default choice will be **Google Chrome** but we'll also test our web pages in **Firefox**.
9. **Web Server:** This is optional. If needed we may need to use Apache Web server.

We'll introduce above software gradually when we'll be covering specific topics.

Exercise 2: Do you think different Web Browser can cause any incompatibility issues for web applications? Have a quick discussion with your tutor.

Yes, it is possible e.g., there can be Web browser incompatibilities (different behaviour for different HTML elements e.g., several CSS3 and JavaScript features may not be supported by older/specific versions of web browsers)

Exercise 3: In this exercise, you should quickly read Component A project specification (under Assignments link) and then have a discussion with your tutor to clarify any aspects. Use Discussion board to ask questions.

Students should start reading the Website project specification and post queries on Discussion Board. Also, run couple of demo/presentation videos from previous years to give students some inspiration. These videos will be available from BB.

Exercise 4: Discuss with your tutor what different options are to set up your web development environment? Also, take few minutes to find differences between **development environment** and **production environment** and discuss with your tutor.

There are many ways students can setup their web **development environment**. There is no good or bad approach – as long as they're able to access and use necessary technologies and are able to write, run and test their web applications then that should be good enough. Of course, students should need to be careful when they'll have to deploy their web application/site in a **production environment (i.e. in real business environment)**.

There are many ways students can install and configure their web development & databases environment. It's up to them to decide which approach suits them. They do not need to install everything now. At the moment their aim should be to adopt a suitable approach that suits them well.

All options would require students to take regular backup of their data/scripts.

These options can be:

1. *Native installation on their personal computer*

They'll have complete setup on their personal machine and it should work fast – depending upon their machine hardware configuration. If they use laptop then they may bring it to practical sessions and use them for practical exercises. This is the easiest approach and it should not take much space on their computers – I think simple installation of everything should be under 2 GB disk space. All required software are free/academic licenses.

2. Use lab machines.

Lab machines have necessary software already installed. Students may need to configure software and may need to download new packages e.g., python packages. There will be some limitations e.g., students will need to access DB from CSCT Cloud for which they'll need to setup a secure connection which would require creating and sharing SSH keys. Also, students will need to install python packages for specific user profile. They may also need to reconfigure VS Code when moving from one machine to another machine because VS Code settings are not synced with students' roaming profiles. We'll provide detailed guide to students for using/setting up lab machines.

3. A VM solution e.g., Windows 10 VM (or Linux VM)

They can copy a Virtual Machine on an external hard disk (minimum 64 GB. HDD (slow performance) or SSD (fast performance) with USB 3) and can use it on their personal machine as well as in computer labs. For their home machine, they will need to install hypervisor (e.g., VMWare Workstation or VMWare Player) to run the virtual machine. As UWE students they should be able to get VMWare Workstation license through UWE software for home page. In labs, VMWare Workstation should already be installed. Once VM is running, they can create complete setup (i.e., install and configure all software) on their VM.

Precautions: VMs may become slow if hardware (RAM, processors) are of low specs. They'll have to make sure they properly shut down VM and close VMWare and unplug the external drive from their host operating system.

4. CSCT Cloud solution

A fully configured environment accessible from home or from UWE computer lab. Students will need to download couple of client software and run few commands and setup environment when they use it first time. They should be able to write their scripts using VSCode on their computers and then save and execute their scripts on cloud server. Students will have restricted access to resources on the UWE Cloud server. We'll need to access Cloud server to perform DB related tasks.

5. Group work solution

1. Understand the problem and then perform requirements analysis, design, implementation, testing, deployment, maintenance
2. Entities: module, student, program, pathway etc.
3. You may consider something like a user comes to index page, and there is clicks on login button, provide login details and if these details are correct then SAT pathway or module choice page loads and so on...
4. Data storage is secure, data access is secure – i.e. only authorized users can access data and perform certain actions, data transfer is secure
5. Header, body, footer and what should be layout of body

Home work: Exercise 6: Let's assume we have the following URLs:

<http://www.uwe.ac.uk/>

<https://intranet.uwe.ac.uk/>

<https://www.uwe.ac.uk:443/search-results?q=Computer+Science>

<https://people.uwe.ac.uk/Person/Zaheer2Khan>

<https://www.bbc.co.uk/news>

<https://www.bbc.co.uk:443/news>

<https://www.bbc.co.uk:80/news>

Use a search engine to complete the following tasks:

Task 1: Identify protocols used in the above URLs

Task 2: Identify domain names in the above URLs

Task 3: Identify sub-domains in the above URLs

Task 4: Identify the top-level domain in the above URLs

Task 5: Identify the second level domain in the above URLs

Task 6: Identify Paths in the above URLs

Task 7: Identify Query String in the above URLs

Task 8: Identify ports in the above URLs

Task 9: Identify which one of the above URLs is invalid and why?

Task 1: http and https

Task 2: uwe.ac.uk and bbc.co.uk

Task 3: intranet, people

Task 4: .uk

Task 5: .ac and .co

Task 6: /Person/Zaheer2Khan , /search-results , /news

Task 7: search-results?q=Computer+Science

Task 8: 80 and 443

Task 9: <https://www.bbc.co.uk:80/news> is invalid because port 80 is for http protocol.