SCCT Development Environment (WIP)

A proof of concept for a development environment to provide a common cross-platform setup for students working either on university machines or on their own computers.

The latest version of the development environment can be downloaded at: https://owen.panel.uwe.ac.uk/scct/scct-dev-environment.zip

The source for the container images and build can be found at: https://github.com/owenjones/scct-dev-container

While testing please report any issues to owen.jones@uwe.ac.uk.

Required Software

To use the development environment Visual Studio Code and the Docker engine must be installed:

UWE Lab Computers

Visual Studio Code and Docker Desktop are already available, search for them from the Start menu.

See note in WIP issues relating to lab computer issues.

UWE Staff Computer

Visual Studio Code and Docker Desktop are available to install from the Software Center. After installing Docker you’ll need to logout and login again for post-installation changes to take effect.

Your Own Computer (Windows, macOS, Linux)

You will need to download and install the software yourself:

- Visual Studio Code: https://code.visualstudio.com/
- Docker: https://www.docker.com/products/docker-desktop/

Launching the Development Environment

- Start Docker and open a new Visual Studio Code window.
- Click the “Remote Connection” button (><) in the bottom left of the window – if the button is missing, you need to install the Microsoft “Dev Containers” extension first

- Select the “Open Folder in Container…” option and navigate to the development environment workspace (if you’re viewing this instruction file as a PDF, it’s the directory you found it in)
- Visual Studio Code will start building the environment, this may take a while the first time you use it on a computer as there are several container images to be downloaded
- Once the development environment is setup you will find yourself in a Visual Studio Code workspace running inside the development environment

Using the Development Environment

The development environment is a container running a Linux distribution (Ubuntu), when you first start the environment your file explorer will be in your home directory (/home/scct) in the container. Here you can create new files and folders and use these for active workspaces.

To interact with the development environment through the command line use the “Terminal > New Terminal” option from the top menu.

The development environment comes with several additional services to make development easier:

- PHPMyAdmin, an interactive mysql tool: http://localhost:8080
- Mongo-express, an interactive mongodb tool: http://localhost:8081

MySQL Server

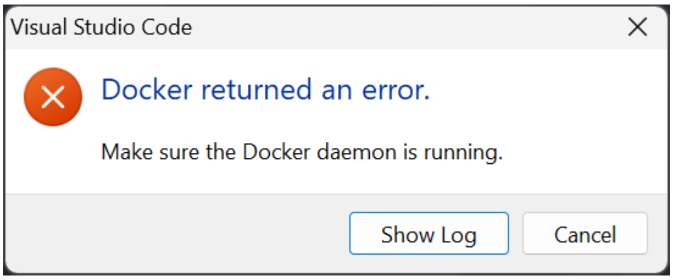
The development environment comes with a MySQL server, this can be accessed through the command line using the ‘mysql’ executable or through Python scripts using the details below. You can

- Username: scct
- Password: scctdev
- Hostname: mysql
- Port: 3306

If you’re connecting to the MySQL server with locally installed software (e.g. MySQL Workbench) rather than using software running on the container, you should use ‘localhost’ as the hostname.

Common Errors

“Make sure the Docker daemon is running”



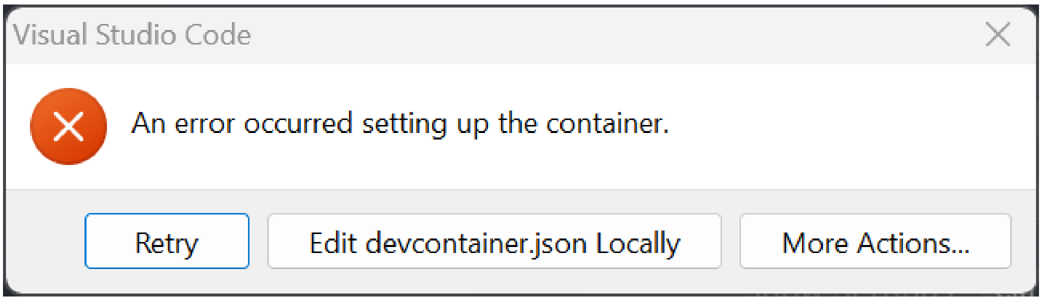
Visual Studio Code

Docker returned an error.

Make sure the Docker daemon is running.

You need to start Docker Desktop (or install it if you haven’t already) before trying to launch the development environment.

“An error occurred setting up the container”



Visual Studio Code

An error occurred setting up the container.

There are several issues that can cause the container setup to fail, the most common is that you already have Docker containers or other locally installed software (e.g. MySQL server) running using ports required by the development environment.

Check you have nothing else bound to ports **3306**, **8080** or **8081** – if you do stop these Docker containers or close the running software, then try launching the development environment again.

WIP Known Issues (0.1)

Trying to run environment on UWE lab machines from the Documents folder or H drive fails as the location user files are kept in is a network share and docker cannot handle the docker compose files being on a network share

~~Solution is to drop the environment folder into your OneDrive folder as this is a locally mapped path – will this sync student work properly??~~

No and it locks up after a short time, how do we get this to run if students only have network drives – can they use removable drives?