About Spikes

In this course, we will undertake “spike” activities, which may be new to some students.

Q: What is a Spike?
Unlike lab work which you have already done and typically has a very structured set of actions you must do, spike work is defined in a more open form and allows you to “close” the presented gap in your own way.
A Spike is an agile software engineering practice used to overcome some gap in knowledge or skills. In these cases it is difficult to build a plan or design in which you can have much confidence. The aim of the spike is to overcome these issues as quickly as possible.
We use spikes in this unit to close a number of “gaps” in knowledge and skill that align with the Intended Learning Outcomes (ILOs) of the Unit.

Q: So, what do I actually need to do for a Spike in this unit?
For each spike you need to "close" the gap in knowledge or skill presented. This gap is provided to you as a “Spike Plan”. You then close the gap, and present the outcome of your spike work using a “Spike Outcome Report”, which you must submit to Canvas and show to your tutor during your allocated lab session. If the tutor approves your report, they will record this in Canvas.

In summary, you need to:

1. do the **Work** (close the gap i.e. code, document etc.),
2. write a **Spike Outcome Report**,
3. **show** the work (running code) to your tutor, and
4. have the tutor **read and approve** your outcome report.

Your Spike Outcome Reports, and any code or resources you developed, form the basis of your final Portfolio. You must complete ALL **Core** Spikes to pass the unit. You do not need to do the “optional” spikes or spike “extensions” if present, but they can be the basis for extension work.

Q. What is the difference between a Spike and Lab?
You will generally have a Lab on a topic to establish a working starting point & the fundamentals before diving in further into the concepts with Spikes.

Labs	Spikes
are “follow the instructions” tasks – they will generally have existing code for you to modify in specific ways described by the lab notes	are more open and requires you to take some initiative in implementing key concepts
The outcome is the code – show the tutor your code	The outcome is the spike report and your code – show your tutor the spike report first, and they may ask to see your code
Lab will give you a specific outcome to upload to canvas – if no outcome is specified upload your commit history to canvas Keep code in your repo	Upload spike reports to canvas Keep spike reports in your repo (yes, they will be in two places) Keep code in your repo

Q. Can I just keep adding new code to my old code (say, from previous spikes?)
No! Keep a separate "clean" version of your code for each Spike Outcome that you achieve. Many spikes develop on prior spike work, but do not keep adding to old code directly. Create a separate new copy (folder) of code for each spike outcome.
Make sure your Extension work is also in a separate file. For example:

```
Repo/
      Task1/
            Code.py
            Code-Extension.py
      Task 2/
            Code.py           << if appropriate, copy Task1/code.py and then modify it
```