CITS5502 Software Processes
Introduction

Unit coordinator: Arran Stewart
Focus of the unit: this unit focuses on how we can

- describe;
- analyse;
- measure; and
- improve

software processes. We look at software development life-cycles; meta-processes (processes to do with processes); and software processes at the team and individual level.
Whenever we perform software-related activities, we’re making use of a software process (even though it may be ad hoc and undocumented).

Making our processes explicit allows us to assess and improve them.
Unit Information

Unit coordinator: Arran Stewart
Contact: arran.stewart@uwa.edu.au
Phone: +61 8 6488 1945
Office: Rm G.08 CSSE Building
Consultation: Thurs 4-5pm

Teaching staff: Arran Stewart, Terry Woodings

Unit webpage: http://teaching.csse.uwa.edu.au/units/CITS5502/

Contact Hours:

- Lecture: Tuesday 3–5pm Robert Street L.T.
- Workshop: Thursday 1–2pm in Blakers LT, starting week 2
Getting information about the unit

- **Unit webpage.** Your first point of call for information.
- **HELP5502.** The discussion and question forum for the unit – if you have questions not answered on the webpage, and your question would benefit other students, ask it here.
- **Unit coordinator.** If you have questions that are not appropriate for the discussion forum, the next best avenue is to email the unit coordinator, or come to regular consultation hours.
Unit contact hours – details

Lectures:
- Recorded – but recordings are not always reliable.

Workshop:
- Work through practical exercises
- Bring a laptop if you can.
• 2 projects – worth 30% and 20%
• exam (50%)
Assessment – exam

- Worth 50%; practice questions will be made available before the November examination period.
General overview of topics:

- Defining and analysing software processes
- Measuring and improving process effectiveness
Programming knowledge required

- The prerequisite for the unit is 12 points worth of programming units
- For the projects that require calculation, you *may* use any language you like (Python, Matlab, R, Java, Haskell, etc)
- But it is also possible to complete them using nothing more than the built-in statistical analysis features of any modern spreadsheet application (e.g. Microsoft Excel\(^1\), or LibreOffice Calc\(^2\))

---

\(^1\) Available for MS Windows, MacOS X and Android; an online version is also available at [https://office.live.com/start/Excel.aspx](https://office.live.com/start/Excel.aspx)

\(^2\) Available for MS Windows, MacOS X and Linux.
Various readings – but a copy of one of the software engineering texts by Pressman or Sommerville would be useful. (Expensive if bought locally; under $50 bought secondhand.) See the web page for details.