Acknowledgement of Country

We acknowledge and pay our respects to the Kaurna people, the traditional custodians whose ancestral lands we gather on. We acknowledge the deep feelings of attachment and relationship of the Kaurna people to country and we respect and value their past, present and ongoing connection to the land and cultural beliefs.

(and to those more senior, too)

How we can help

Student Life

Counselling and support

8313 5663

Safer Campus Community

Report harassment or anti-social behaviour

8313 5663

UniCare

Healthcare on campus

8313 5050

https://www.adelaide.edu.au/covid-19

Faculty of SET / School of Computer Science

Software Engineering & Project
Course Introduction

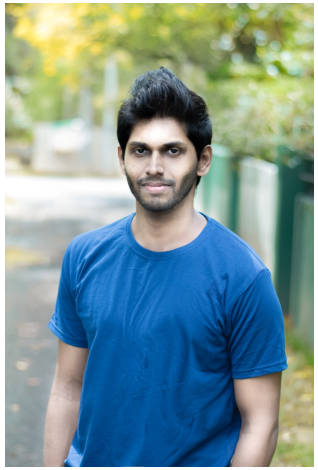
Teaching Team

Amali Weerasinghe (Course Coordinator)

Chuyue (Angela) Qin

Teaching Team: Tutors

Abdul Kareem



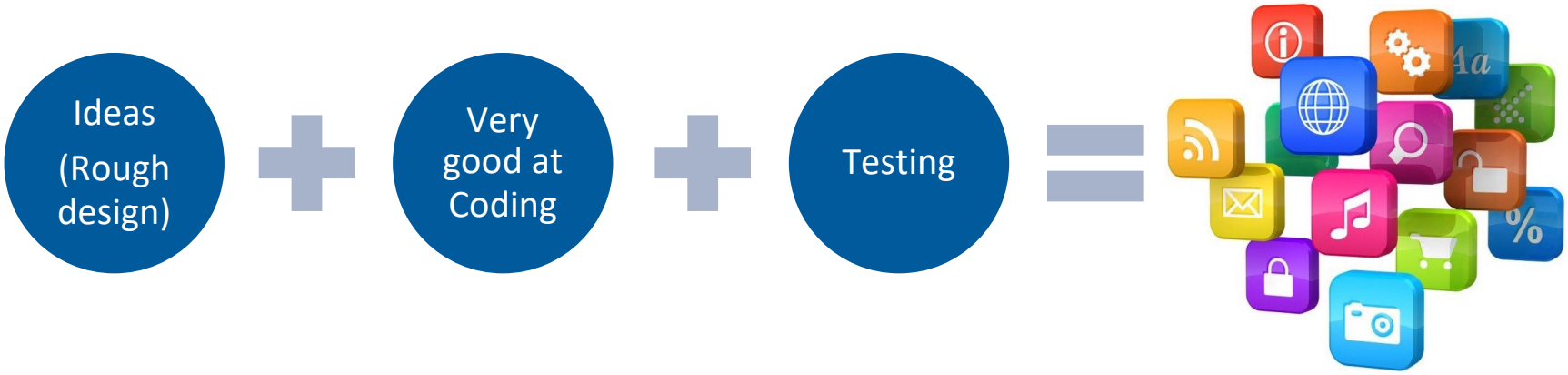
Dileepa

Isuru

Roshan

Why Software Engineering?

How to make a piece of software (in non-expert view)



Why Software Engineering?

- Simple iPhone apps
  - **20,000+ Lines of code**

- Google Chrome
  - **6M+ Lines of code**

- Facebook
  - **100M+ Lines of code**

- Google (All internet services)
  - **2 Billion Lines of code**

https://informationisbeautiful.net/visualizations/million-lines-of-code/ and other sources
10 LOC per day: https://www.google.com/search?q=lines+of+code+written+per+day

Why Software Engineering?

Singapore govt health database hacked | ATO to ditch AUSkey for myGovID | UTS creating middleware to protect international students' data | Hawaiki subsea cable comes online | Sydney man … AWS Snowba…

itnews

GOVERNMENT IT | SECURITY | FINANCE IT | TELCO | BENCHMARK AWARDS | LOG IN | SUBSCRIBE

ATO reveals cause of SAN failure

By Allie Coyne
May 30 2017
9:18AM

34 Comments

Gets costs back in settlement with HPE.

The Australian Taxation Office has reached a settlement with storage vendor HPE that will see it recoup its costs from its damaging storage network failure and provided a first glimpse into what went wrong.

The ATO's 3PAR storage area network (SAN) problems started last December when an unexpected three-day outage took all its online services down.

The failure of the 3PAR SAN was the result of a confluence of events: the fibre optic cables feeding the SAN were not optimally fitted, software bugs on the SAN disk drives meant stored data was inaccessible or unreadable, back-to-base HPE monitoring tools weren't activated, and the SAN configuration was more focused on performance than stability or resilience, Jordan said.

The collapse of the SAN caused the loss of 1PB of data.

Why Software Engineering & Project?

ATO $4.2 billion short of budget revenue target

By Nassim Khadem

Updated 30 October 2017 – 4:22pm, first published at 12:34pm

The Tax Office has fallen short of budget revenue targets by $4.2 billion, its 2017 annual report

Collectible debt

Collectable tax debt was $20.9 billion, up from $19.2 billion in 2015–16. The majority was owed by small business. Small businesses owed nearly $13.9 billion in collectable tax debt, an increase of 7 per cent from last year.

The 12-month rolling average of the ratio of total collectable debt to net tax collections was 5.6 per cent, "not quite achieving the target of 'below 5.5 per cent'", the ATO said.

These were "reasonable results", it said, given a year-on-year increase of $1.8 billion in audit-raised liabilities, including liabilities flowing from the Tax Avoidance Taskforce, the Serious Financial Crime Taskforce, Operation Elbrus and Operation Nosean.

The decrease in debt collection activities was due to the lead-up to an ATO system upgrade in November 2016 and IT outages late last year and early this year.

https://www.smh.com.au/business/the-economy/ato-42-billion-short-of-budget-revenue-target-20171030-gzauc0.html

Learning Outcomes

On successful completion of this course
you will be able to:

- Describe and discuss software development techniques and methodologies
- Apply various computer science methods and algorithms
- Conduct a group-based software development
- Demonstrate professional conduct in software development and meeting participation
- Demonstrate professional codes of computer scientists and engineers
- Investigate, analyse, and use software tools to increase the productivity of software development

Textbooks (either is fine)

Activities

Lectures: *Provide fundamental knowledge for software development techniques and methodologies*

- *Run from Week 1 to Week 10*
  - Monday 4.10pm-5pm [The Braggs]
  - Thursday 3.10pm-4pm [The Braggs]
  - Friday 4.10pm-5pm [The Braggs]
- Check detailed schedule on MyUni
- Delivered hybrid: face-to-face + via Echo360 (see details on the course page)

Assignments in General: *Analysis and critique on the main components of software development process and ethics.*
→ not just to keep you busy… but to let you practise!

The schedule and deadlines are available on myUni.

Activities

Group Project: Work *in a group of 6-9*

- Work with a client (read: a tutor will be the proxy for a project provider from industry)
- Develop a system
- Fortnightly client meeting: 25 minutes meeting with a client start from Week 2 – 11 (Week 2 meeting is going to be longer; then meetings in Weeks 5/7/9/11)
- **Clarification:** Your contribution to the group affects the group mark! (well… it is a group project, remember?)
- Individual contributions will be tracked via
  - GitHub repository
  - Group sprint reports
  - Individual sprint reports

Assessments

- Initial Report: 10%, Group
- Retrospective 1/2/3/4: 10%, Group + Individual
- Final Report: 20%, Group + Individual
- Quizzes: 5%, Individual
- Final Presentation: 25%, Group + Individual
- No exam.

There are no “minimum performance” hurdles.

Assumed knowledge

We assume that you have the knowledge of COMP SCI 2103/2202 Algorithm Design and Data Structures/Foundations:

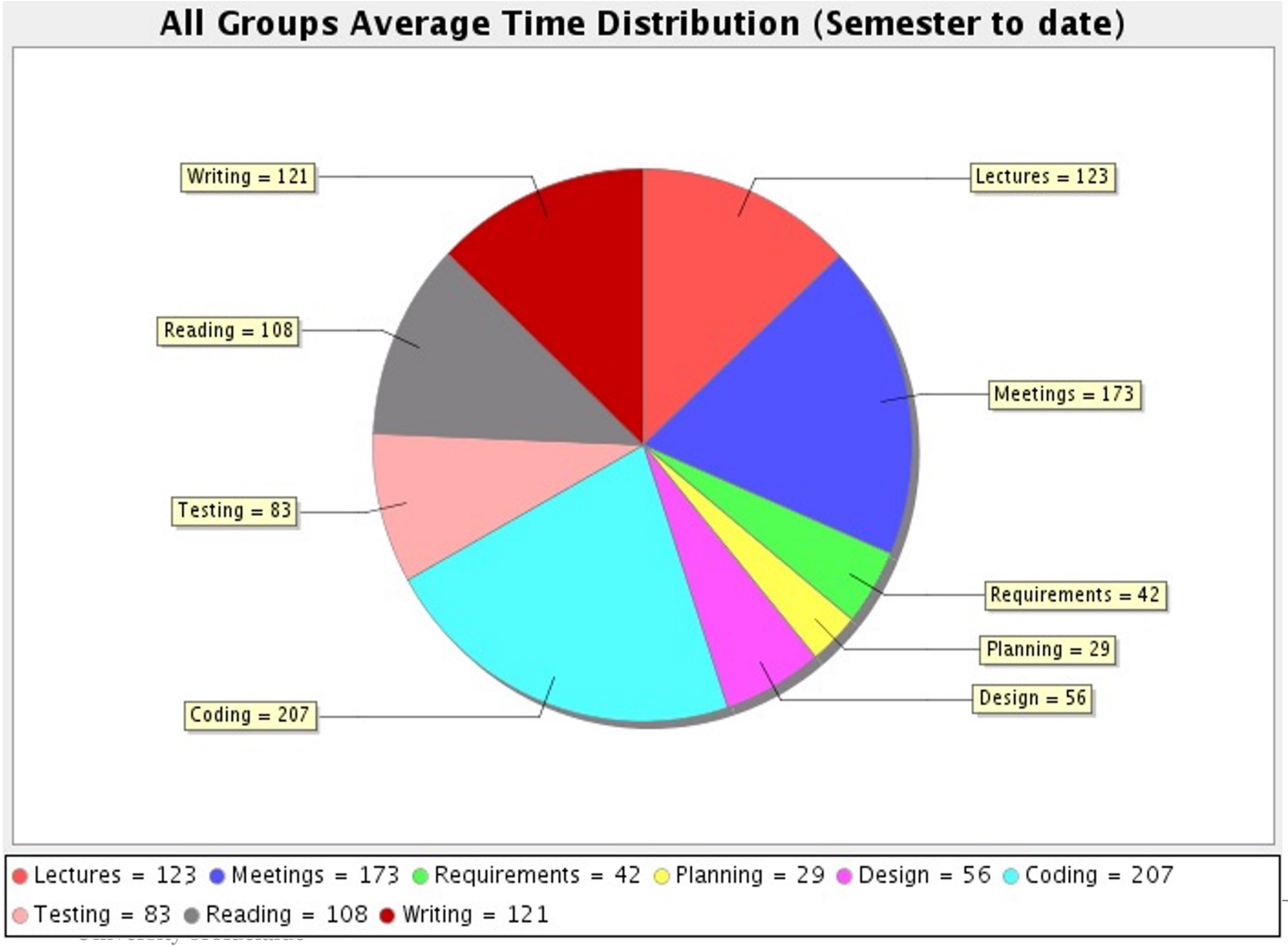
- The fundamental concepts of procedural programming.
- Fundamental programming constructs.
- Approaches to Problem Solving. Notion of abstract data type, representation of lists, stacks, queues, sets, trees and hash tables. Graphs and Graph Traversal.

*If you have no/little background in the topics, you *need* to allow for more than **12 hours** per week for the course.*

Workload

- This course involves many group and individual tasks
- You should expect to spend (at least) 156 hours on this course during the semester
  (see https://www.adelaide.edu.au/policies/669/?dsn=policy.document;field=data;id=977;m=view
  → 12 hours/week * 13 weeks)
- Practical work should take you about **130 hours** → put in 10 hours of quality work, and you increase your chances (and the chances of the team) of passing with a good mark.
- You are expected to meet in your groups outside of the project sessions. It is recommended to have a meet twice a week for a minimum of 1 hour… even if it is just blocked time to sit down to do some work.

Time Distribution in an older Software Engineering Course: hours a team has put in during a semester



Team total, here: about 940 hours

Project: Specification

- **Task:** You will make a software system per the request from an industry company.
- Specification is *intentionally* vague and incomplete.
- You have to work with a client (= a tutor) to retrieve requirements
  - Requirements are highly uncertain, changeable at this stage which are the reality of many software projects in industry.
- Hardware and Software
  - Depends on the project.
  - If special hardware is needed, the company and the school will try to provide it - but be prepared to use simulators instead.
  - Mostly open source software. If licensed software is required, the company and the school will try to provide it - but be prepared to work with alternatives instead.

Project: Meetings

- Client meetings can held in the Software Engineering lab (Ingkarni Wardli Room 4.62), but we strongly encourage online/hybrid meetings, as the actual provider (who might be from industry) can then join more easily from time to time.
- Times: you organise these with your tutor.
- One from our teaching team will act as your client → she/he is your first point of contact. Technical challenges are for your team to sort out!
- **Meeting attendances are compulsory**, all group members must attend
  - Absences will require a documentation: Medical certificate, etc.
- Meeting minutes:
  - It will be ***extremely*** helpful to you if all meetings (e.g., client meetings, group meetings) are accurately recorded.

s

Project: Tools

- Tooling will depend on the project requirements and the tech stack your group decides to use (e.g., IntelliJ, Maven, JUnit, etc. in case you decide to use the Java ecosystem)
- All groups **must** use GitHub for version control and project management (https://github.com/features/project-management)
- Become familiar with the tools very early on!
- More details on exemplary tools in dedicated lectures.

Project: Groups

- You work in groups of 6-9
  - It is not a simulation, but a real industry project.
  - Often your team mates come from different backgrounds, both academic and culture
  - Developing skills on how to work with people
- Self-allocating to groups will start soon
  - ***We will let you self-organise on a first-come-first-serve basis.***
  - But there will be constraints:
    - Postgraduates separate, as they have one more component in the final report.
    - Some projects will be available only to certain majors.

Communication (1/3)

Course website
https://myuni.adelaide.edu.au/courses/75055

There will be ***Announcements*** from time to time - these tend to be really important, so please read them when you get the notification emails.

Use the ***Discussions*** to ask your questions: with one reply there, we can help you and possibly many of your friends. Also, this way we get to see what is happening in the course.

Communication (2/3)

Also, let us know (via the tutors or via the ***Discussions***) what seems to be going okay and not okay with the course.

Why?

You influence how this course will be delivered in the future!

Proof: based on 2019 feedback:
no exam (yeah!?), self-enrolment, assessments aligned with agile development, and plenty of fine-tuning…
… and based on 2020/2021 feedback:
more quizzes, more details on assessment criteria, …

Communication (3/3)

http://roehamptonlearning.com/too-many-emails-how-to-do-as-much-with-less/

Let's be honest, efficient, and effective:

- we can't micromanage 400 students
- we can't micromanage 50 teams

... but

- ***we trust you to self-organise!***

Your first point of contact is your client (=the tutor). If needed, they can escalate issues to the lecturers.

If you need an extension (e.g. due to sickness), then please contact course coordinator via email, along with proof of your circumstances (e.g. a doctor's certificate). Contact me ASAP, not after Week 13 for something that has happened in Week 2.

What to do today?

Check details, schedule, deadlines on myUni.
Self-signup will start today.
Once the groups are allocated, contact your teammates and start your the work ASAP.