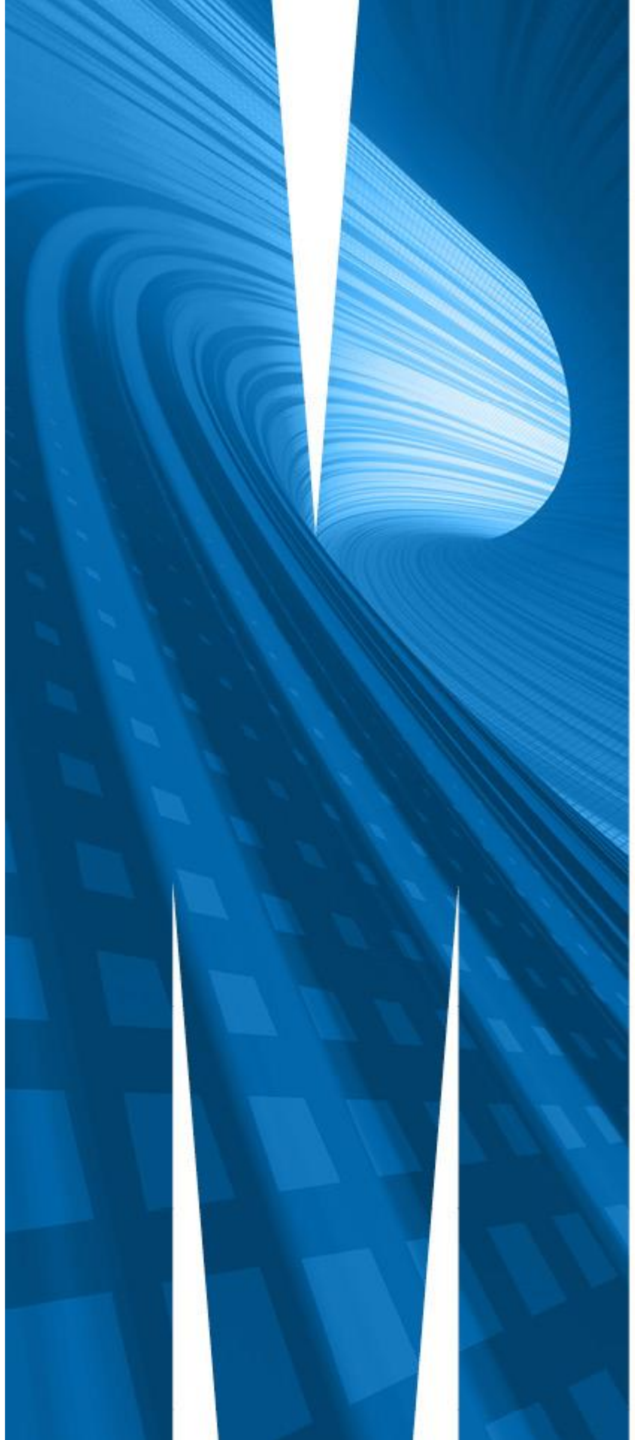
MONASH University

FIT2107 Semester 2 2022

Week 1 – Introduction to the Unit

25-07-2022

Dr Najam Nazar



TEACHING STAFF

Email: firstname.lastname@monash.edu

Role Account: fit2107.clayton-x@monash.edu
Malaysia students please contact your respective lecturer.

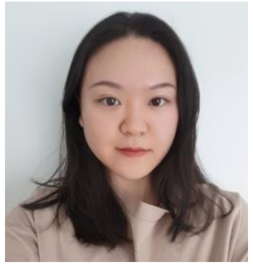
Dr. Najam Nazar
Lecturer @ Clayton

Dr. Lillian Wang
Lecturer @ Malaysia

Dr. Chakkrit Tantithamthavorn
Chief Examiner

Aaron Zheng
Head Tutor

Antoni Erdag
Tutor



Neda Wang
Tutor

Zhiming Deng
Tutor

TingTing Bi
Tutor

UNIT SCHEDULE

Week	Topic	Workshop	Assessment
1	Software Quality - Introduction	Quality Attributes	
2	Quality Plans and Methods	Artefact Quality Plans	
3	Software Testing - Introduction	Faults, Errors and Mistakes	
4	Black-Box Testing – I	Black-Box Testing - I	
5	Black-Box Testing - II	Black-Box Testing – II + Assignment Support	Assignment 1 Due
6	White-Box Testing – I	White-Box testing - I	
7	Unit Testing White-Box Testing - II	White-Box Testing - II	
8	TDD + Unit Testing + Continuous Integration	Unit Testing	
9	Mocking Tests	Mocking + Assignment Support	Assignment 2 Due
10	Back to Software Quality - Code Reviews	Code Review	
11	GUEST LECTURE	Python metrics	
12	Review		Assignment 3 Due

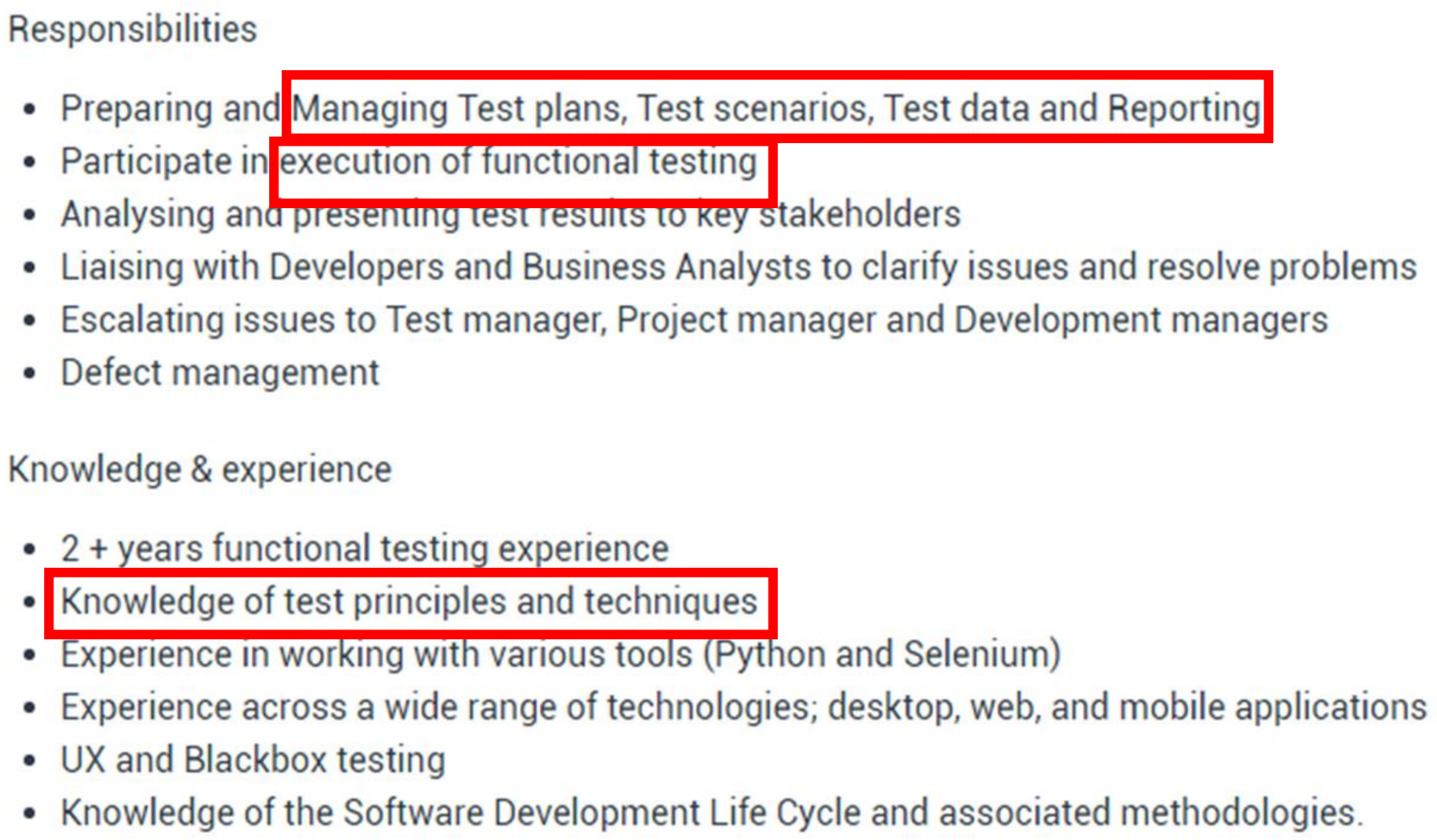
UNIT CHANGES 2022

- Based on the Industry trend as well as on feedback from students this course is revised each year
- Key Changes
  - More focus on Software Testing
  - More programming
  - Code Reviews are introduced
  - No Quizzes
- Lecture slides
- Mandatory Reading on Ed (Assessed)
- Workshops
- Assignment released one week before the deadline of previous Assignment on Ed.

WHY ARE WE DOING THIS UNIT?

- IT graduates think testing doesn’t require any programming skills
- Only unit testing requires programming skills
- I am excellent in programming
    - Hmm….I don’t want to be tester.
- IT industry is very dynamic and rapidly changing.
    - Software Quality is critical
    - Software testing is essential.
- Being software engineer, knowing testing is a critical skill
- …and coding is mandatory for this unit 😉

WHY ARE WE DOING THIS UNIT?



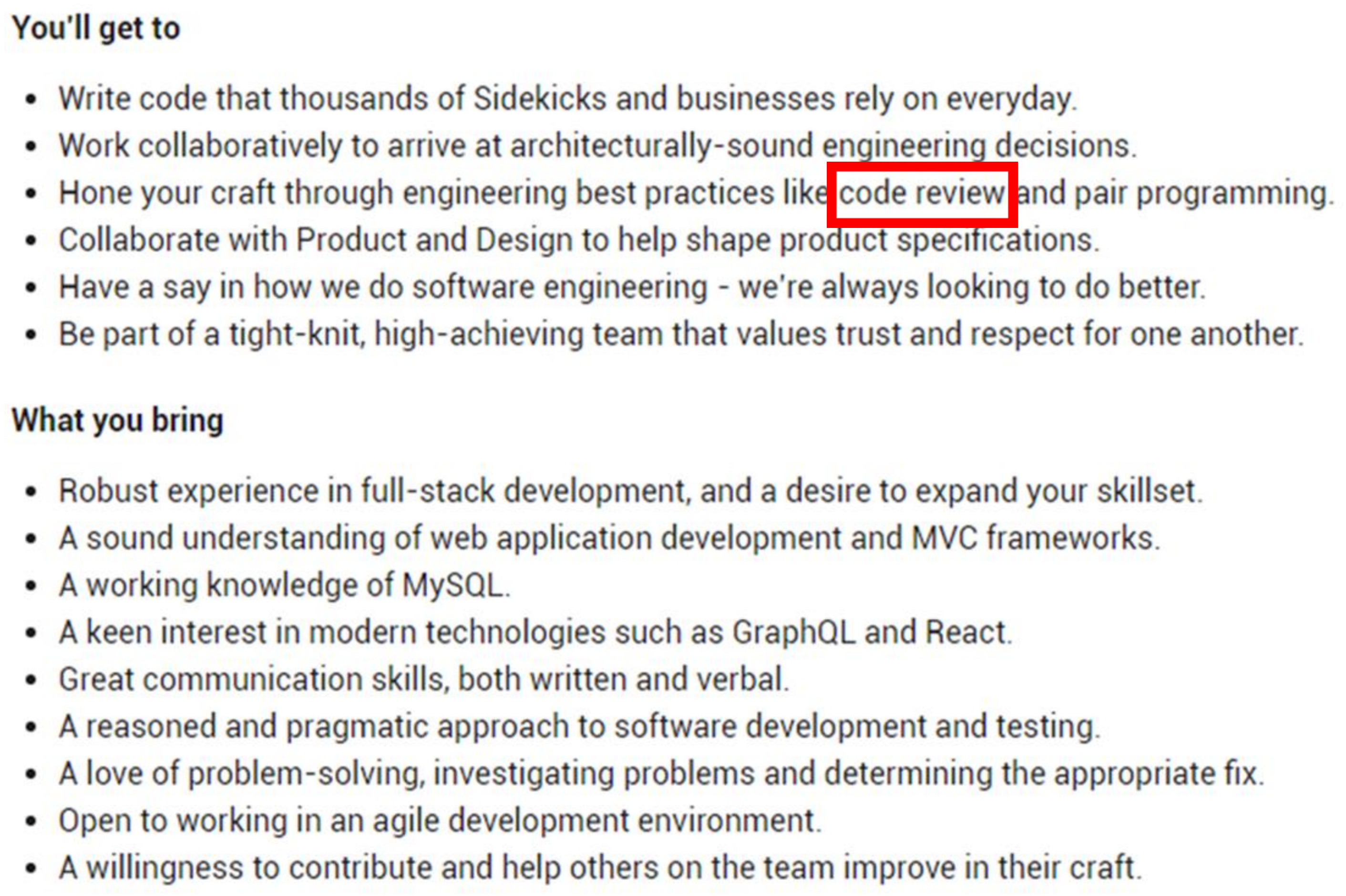
Responsibilities

- Preparing and Managing Test plans, Test scenarios, Test data and Reporting
- Participate in execution of functional testing
- Analysing and presenting test results to key stakeholders
- Liaising with Developers and Business Analysts to clarify issues and resolve problems
- Escalating issues to Test manager, Project manager and Development managers
- Defect management

Knowledge & experience

- 2 + years functional testing experience
- Knowledge of test principles and techniques
- Experience in working with various tools (Python and Selenium)
- Experience across a wide range of technologies; desktop, web, and mobile applications
- UX and Blackbox testing
- Knowledge of the Software Development Life Cycle and associated methodologies.

WHY ARE WE DOING THIS UNIT?



You'll get to

- Write code that thousands of Sidekicks and businesses rely on everyday.
- Work collaboratively to arrive at architecturally-sound engineering decisions.
- Hone your craft through engineering best practices like code review and pair programming.
- Collaborate with Product and Design to help shape product specifications.
- Have a say in how we do software engineering - we're always looking to do better.
- Be part of a tight-knit, high-achieving team that values trust and respect for one another.

What you bring

- Robust experience in full-stack development, and a desire to expand your skillset.
- A sound understanding of web application development and MVC frameworks.
- A working knowledge of MySQL.
- A keen interest in modern technologies such as GraphQL and React.
- Great communication skills, both written and verbal.
- A reasoned and pragmatic approach to software development and testing.
- A love of problem-solving, investigating problems and determining the appropriate fix.
- Open to working in an agile development environment.
- A willingness to contribute and help others on the team improve in their craft.

TEACHING METHODS

- Semi Flipped Classroom
- Lectures
  - Two hour per week!
- Written Course Notes (Mandatory)
- Workshops (two hours)
  - Lots of working in small groups
  - Some at a computer, some not
  - Past exam questions so encouraged to attend

ASSESSMENTS

Assessment Task	Value	Due Date
Quality Check and Manual Blackbox Testing	20%	Week 5
TDD + Automated Blackbox + White Box Testing	25%	Week 9
Code Reviews	15%	Week 12
Exam	40%	TBA

ASSIGNMENT 1

- Assignments is about Quality check & Black Box Testing
- Due: Week 5
- Weight: 20%
- Students will be provided with a set of requirements, a running software and other related artefacts.
- Students will identify the quality of the software and apply appropriate black-box testing techniques manually to develop and document test cases.
- No Coding

ASSIGNMENT 2

- Assignments is about Test Driven Development and automated White Box Testing
- Due: Week 9
- Weight: 25%
- Students will be provided with an application, a partial source code and requirements.
- Students will apply Blackbox cases with appropriate white-box testing techniques to develop unit tests and document them.
- This assignment requires coding in Python.

ASSIGNMENT 3

- Assignments is about Code Review and Critique
- Due: Week 12 and Presentations week 12
- Weight: 15% (5% review + 5 % critique + 5% presentation/demonstration)
- Presentation is hurdle and failing to attend presentation means failing the hurdle to pass the assignment 3.
- Students will be given a code repo from other group.
- Students will perform code review and write a critique
- Students will present their code in the presentation and answer questions from the other group that has been assigned to critique

HURDLE

- To pass this unit a student must obtain:
  - 45% or more in the unit's examination, and
  - 45% or more in the unit's total non-examination assessment, and
  - an overall unit mark of 50% or more.
- If a student does not pass these hurdles, then a mark of no greater than 45NH will be recorded for the unit.

Academic Integrity

- Honest presentation of your academic work.
- Acknowledged words, data, diagrams, models, frameworks and/or ideas of others you have quoted (i.e. directly copied), summarised, paraphrased, discussed or mentioned in your assessment through the appropriate referencing methods,
- Provided a reference list of the publication details so your reader can locate the source if necessary. This includes material taken from Internet sites.
- Monash University treats plagiarism as a very serious offence constituting misconduct.
- Further instructions at https://www.monash.edu/students/study-support/academic-integrity

QUESTIONS + EMAILS

- Ask preferably during lectures, tutorials and consultations.
- Use ED discussion forums -> always encouraged.
- Emails
  - Always put FIT2107-S2-2022 at the beginning of the subject
  - No subject emails will be ignored….
  - Use your official Monash email to contact
  - Use your full name in the salutation area (at the end of the text)
  - For all sorts of questions the first person to contact is your tutor.
    - If it is not resolved there, you contact the Head Tutor, i.e. Aaron Zheng.
    - If necessary, he will involve the Lecturers and the Chief Examiner of the Unit.
- Contact through the role account

- I'll regularly make announcements on Ed Discussions
- Check regularly.
- I'll regularly send emails
  - I didn't check my email
  - Oh! it went into spam/junk…

MONASH University

What to do if you catch Covid?

- If you are required to isolate due to either testing positive with COVID-19 or any other sickness (flu/fever etc)
  - May choose to continue their learning activities online if you are well enough to do so.
  - Inform us and we add you to the online session.
- If you are unable to continue with studies during the isolation period.
  - Necessary learning arrangements may be required.
  - Please be patient and give us time to help you.
- Contact on Role Account Immediately and CC me or Admin TA.