

CHK302CEM: Agile Development

Module size	Single
Total student study hours	200
Pre-requisites and co-requisites	CHK220CT or equivalent
Excluded Combinations	None

Aims and Summary

This module aims to provide students with sound understanding and experience in Development methodologies, techniques and tools. Agile and rapid approaches to developing information systems and software are critical for responding to changing business environment. The module will examine in detail different methodologies for rapid development and their application in practice. Specific industry strength type of technology for RAD will be utilised for solving real-world problems.

Intended Module Learning Outcomes

On completion of this module the student should be able to:

1. Demonstrate a sound understanding of how Agile Methodologies can be used to define users' requirements, analysis and design of information systems.
2. Compare and contrast a range of current and emerging agile methodologies.
3. Evaluate the methods, techniques and tools for rapid development of various types of information systems, and the reasons for their selection and use.
4. Use a range of appropriate tools to contribute to the development of a solution to a real-world problem.

Indicative Content

Overview of Agile Methodologies ; Agile development model; and phases of development. Major agile methodologies; strengths and weaknesses;

Agile approaches: Dynamic Systems Development Method, Extreme Programming, Scrum, Lean Software Development, pair programming. Distributed Version Control: forking, merge requests, coding standards, branching, merging, code reviews

Automated Testing: unit testing, UI testing, acceptance testing, regression testing, continuous development, continuous deployment.

Problems and Challenges including technical, ethical and security concerns related to RAD

Teaching and Learning

The module is divided into 10 topics, each running for one week. Each week there will be a single 1 hour lecture plus a two and a three hour lab (total 6 hours per week).

Student activity and time spent on each activity comprises:

Lecture	11 hours	(5.5%)
Laboratory	44 hours	(22%)
<u>Self guided</u>	<u>145 hours</u>	<u>(72.5%)</u>
Total	200 hours	

Method of Assessment (normally assessed as follows)

Assessment	Weighting	Learning Outcomes			
		1	2	3	4
Coursework 1: Reflective report	50%		✓	✓	✓
Coursework 2 – 1 hour phase test	50%	✓	✓		

Practical assessment is performed during the academic year according to the schedule set out in the module guide. While coursework is prepared, submitted and then marked and returned at a later date, practical assessments are carried out in lab or tutorial sessions and have instant feedback.

Re-assessment is by new coursework

Assessment

Composition of module mark: 100% Coursework

Pass requirements: Coursework must be at least 40% and Module
Mark must be at least 40%

Essential Reading

Rasmusson, J. (2010) *The Agile Samurai: How Agile Masters Deliver Great Software*. The Pragmatic Bookshelf, North Carolina.

Recommended Reading

Evans, E. (2004) *Domain-driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley, Pearson Education.

Hogbin, E. (2015) *Git for Teams: a User-centered Approach to Creating Efficient Workflows in Git*. O'Reilly.

Required Equipment

None