Contents

General Information .................................................. 2
Learning Outcomes .................................................... 3
Assessment Tasks ....................................................... 3
Delivery and Resources ............................................... 6
Unit Schedule .......................................................... 6
Policies and Procedures .............................................. 7
Graduate Capabilities ................................................ 8
Changes from Previous Offering ................................. 12
Changes from Last Year .............................................. 12
Standards and Grading ............................................... 12
General Information

Unit convenor and teaching staff
Lecturer
Matthew Mansour
matthew.mansour@mq.edu.au
Contact via matthew.mansour@mq.edu.au

Unit Convenor
Stephen Smith
stephen.smith@mq.edu.au
Contact via stephen.smith@mq.edu.au
E6A 381
30 minutes prior to lectures

Lecturer
Deborah Richards
deborah.richards@mq.edu.au
Contact via deborah.richards@mq.edu.au
E6A 328
email for an appointment

Credit points
3

Prerequisites
ISYS114(P) or COMP114(P) or ISYS154(P) or COMP154(P)

Corequisites

Co-badged status

Unit description
This unit is an intermediate unit to deliver a solid foundation in concepts, methods, tools and techniques that organisations use to control the information they use in their day-to-day business, with a particular focus on how computer-based technologies can most effectively contribute to the way business is structured. The units focuses on the fundamental concepts and models of applications development so that they can understand the key processes related to building functioning applications and appreciate the complexity of applications development. The unit emphasises program development and incorporates the software development life cycle, requirements gathering, designing a solution, and implementing and testing a solution in a programming language.
Important Academic Dates
Information about important academic dates including deadlines for withdrawing from units are available at [https://students.mq.edu.au/important-dates](https://students.mq.edu.au/important-dates)

Learning Outcomes

1. Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
2. Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
3. Demonstrate ability to communicate software requirements and designs clearly and effectively.

Assessment Tasks

<table>
<thead>
<tr>
<th>Name</th>
<th>Weighting</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>Early Diagnostic Assessment</td>
<td>5%</td>
<td>Week 4</td>
</tr>
<tr>
<td>Requirements and Modelling</td>
<td>20%</td>
<td>Week 7</td>
</tr>
<tr>
<td>App. Dev. Proj - Part 1</td>
<td>10%</td>
<td>Week 10</td>
</tr>
<tr>
<td>App. Dev. Proj - Part 2</td>
<td>15%</td>
<td>Week 12</td>
</tr>
<tr>
<td>Final Examination</td>
<td>50%</td>
<td>TBA</td>
</tr>
</tbody>
</table>

Early Diagnostic Assessment

Due: **Week 4**  
Weighting: 5%

You will need to form into groups of 5. Use iLearn to put yourself into a group by the end of week 2. Instructions for doing this will be given in class and in the news forum.

You will be required to submit a document in iLearn which outlines the progress of your group in achieving assignment 1. This task is to ensure teams have made contact, and gained an initial understanding of the problem to be elaborated further in part B.

Assignment specification will be available on the iLearn site. Submission will be via iLearn.

This Assessment Task relates to the following Learning Outcomes:
• Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.

Requirements and Modelling

Due: Week 7
Weighting: 20%

This group assignment will develop your skills in requirements specification, analysis and design.

Assignment specification and submission via iLearn.

This Assessment Task relates to the following Learning Outcomes:
• Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
• Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
• Demonstrate ability to communicate software requirements and designs clearly and effectively.

App. Dev. Proj - Part 1

Due: Week 10
Weighting: 10%

Application Development Project - Part 1

Assignment two will have two parts. The first part will assess your design and documentation skills. You will create dataflow, database and screen designs and diagrams.

Submission via iLearn.

This Assessment Task relates to the following Learning Outcomes:
• Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
• Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
• Demonstrate ability to communicate software requirements and designs clearly and
effectively.

App. Dev. Proj - Part 2

Due: **Week 12**
Weighting: **15%**

Application Development Project - Part 2

The second part of assignment 2 will assess your development and testing skills. You will create a database, html forms, php code and test case.

Submission via iLearn.

This Assessment Task relates to the following Learning Outcomes:

- Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
- Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
- Demonstrate ability to communicate software requirements and designs clearly and effectively.

Final Examination

Due: **TBA**
Weighting: **50%**

Closed Examination

This Assessment Task relates to the following Learning Outcomes:

- Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
- Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
- Demonstrate ability to communicate software requirements and designs clearly and effectively.
Delivery and Resources

CLASSES

Each week you should attend 2 hours of lectures and a Mixed tutorial Class (starting in week 2). For details of days, times and rooms consult the timetables webpage.

Note that mixed classes commence in week 2.

Please note that you are required to attend most of the mixed classes and hand in prepared work each week. Failure to do so may result in you failing the unit or being excluded from the exam.

REQUIRED AND RECOMMENDED TEXTS AND/OR MATERIALS

Textbook

The textbooks for ISYS254 used this semester is:


Note: The book can be found at the co-op bookshop.

UNIT WEBPAGE AND TECHNOLOGY USED AND REQUIRED

Websites

The web page for this unit can be found at: here

Digital recordings of lectures are available. Read instructions here.

Discussion Boards

The unit makes use of discussion boards hosted within ilearn Please post questions relevant to the unit there. They are monitored by the staff on the unit.

Unit Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
<th>Reading</th>
<th>Lecturer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Unit Overview - Systems, Roles, and Development Methodologies.</td>
<td>Chp1</td>
<td>Smith</td>
</tr>
<tr>
<td>2</td>
<td>Understanding and Modelling Organizational Systems + Use Case Diagrams</td>
<td>Chp 2 and online resources</td>
<td>Richards</td>
</tr>
<tr>
<td>3</td>
<td>Information Gathering</td>
<td>Chp 4 &amp; 5</td>
<td>Richards</td>
</tr>
<tr>
<td>4</td>
<td>O-O Analysis and Design - Sequence Diagrams &amp; State Diagrams</td>
<td>Chp 10 and online resources</td>
<td>Richards</td>
</tr>
<tr>
<td>5</td>
<td>O-O Analysis and Design - Package Diagrams &amp; Model Consistency</td>
<td>Chp 10 and online resources</td>
<td>Richards</td>
</tr>
</tbody>
</table>
### Policies and Procedures

Macquarie University policies and procedures are accessible from [Policy Central](http://mq.edu.au/policy/docs/). Students should be aware of the following policies in particular with regard to Learning and Teaching:


In addition, a number of other policies can be found in the **Learning and Teaching Category** of Policy Central.

### Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: [https://students.mq.edu.au/support/student_conduct/](https://students.mq.edu.au/support/student_conduct/)

### Results

Results shown in *iLearn*, or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in *eStudent*. For more information visit [ask.mq.edu.au](http://ask.mq.edu.au).


### Student Support

Macquarie University provides a range of support services for students. For details, visit [http://stu](http://stu)
Our graduates will take with them the intellectual development, depth and breadth of knowledge, scholarly understanding, and specific subject content in their chosen fields to make them competent and confident in their subject or profession. They will be able to demonstrate, where relevant, professional technical competence and meet professional standards. They will be able to articulate the structure of knowledge of their discipline, be able to adapt discipline-specific knowledge to novel situations, and be able to contribute from their discipline to inter-disciplinary solutions to problems.

This graduate capability is supported by:

Learning outcomes

- Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
- Demonstrate an understanding of the concepts and tools needed to successfully design
and build database-centric application programs using object-oriented and traditional methods and project management techniques.

**Assessment tasks**

- Early Diagnostic Assessment
- Requirements and Modelling
- App. Dev. Proj - Part 1
- App. Dev. Proj - Part 2
- Final Examination

**Problem Solving and Research Capability**

Our graduates should be capable of researching; of analysing, and interpreting and assessing data and information in various forms; of drawing connections across fields of knowledge; and they should be able to relate their knowledge to complex situations at work or in the world, in order to diagnose and solve problems. We want them to have the confidence to take the initiative in doing so, within an awareness of their own limitations.

This graduate capability is supported by:

**Learning outcomes**

- Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
- Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.

**Assessment task**

- Final Examination

**Creative and Innovative**

Our graduates will also be capable of creative thinking and of creating knowledge. They will be imaginative and open to experience and capable of innovation at work and in the community. We want them to be engaged in applying their critical, creative thinking.

This graduate capability is supported by:

**Learning outcome**

- Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
Effective Communication

We want to develop in our students the ability to communicate and convey their views in forms effective with different audiences. We want our graduates to take with them the capability to read, listen, question, gather and evaluate information resources in a variety of formats, assess, write clearly, speak effectively, and to use visual communication and communication technologies as appropriate.

This graduate capability is supported by:

**Learning outcomes**

- Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
- Demonstrate ability to communicate software requirements and designs clearly and effectively.

**Assessment tasks**

- Early Diagnostic Assessment
- Requirements and Modelling
- App. Dev. Proj - Part 1
- App. Dev. Proj - Part 2
- Final Examination

Engaged and Ethical Local and Global citizens

As local citizens our graduates will be aware of indigenous perspectives and of the nation's historical context. They will be engaged with the challenges of contemporary society and with knowledge and ideas. We want our graduates to have respect for diversity, to be open-minded, sensitive to others and inclusive, and to be open to other cultures and perspectives: they should have a level of cultural literacy. Our graduates should be aware of disadvantage and social justice, and be willing to participate to help create a wiser and better society.

This graduate capability is supported by:

**Learning outcome**

- Demonstrate ability to communicate software requirements and designs clearly and effectively.

Socially and Environmentally Active and Responsible

We want our graduates to be aware of and have respect for self and others; to be able to work with others as a leader and a team player; to have a sense of connectedness with others and country; and to have a sense of mutual obligation. Our graduates should be informed and active...
participants in moving society towards sustainability.

This graduate capability is supported by:

**Learning outcome**

- Demonstrate ability to communicate software requirements and designs clearly and effectively.

**Capable of Professional and Personal Judgement and Initiative**

We want our graduates to have emotional intelligence and sound interpersonal skills and to demonstrate discernment and common sense in their professional and personal judgement. They will exercise initiative as needed. They will be capable of risk assessment, and be able to handle ambiguity and complexity, enabling them to be adaptable in diverse and changing environments.

This graduate capability is supported by:

**Learning outcomes**

- Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.
- Demonstrate ability to communicate software requirements and designs clearly and effectively.

**Critical, Analytical and Integrative Thinking**

We want our graduates to be capable of reasoning, questioning and analysing, and to integrate and synthesise learning and knowledge from a range of sources and environments; to be able to critique constraints, assumptions and limitations; to be able to think independently and systemically in relation to scholarly activity, in the workplace, and in the world. We want them to have a level of scientific and information technology literacy.

This graduate capability is supported by:

**Learning outcomes**

- Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.
- Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.

**Assessment tasks**

- Requirements and Modelling
Commitment to Continuous Learning

Our graduates will have enquiring minds and a literate curiosity which will lead them to pursue knowledge for its own sake. They will continue to pursue learning in their careers and as they participate in the world. They will be capable of reflecting on their experiences and relationships with others and the environment, learning from them, and growing - personally, professionally and socially.

This graduate capability is supported by:

**Learning outcome**

- Demonstrate ability to communicate software requirements and designs clearly and effectively.

**Assessment tasks**

- Requirements and Modelling
- App. Dev. Proj - Part 1
- App. Dev. Proj - Part 2

**Changes from Previous Offering**

The weekly structure has changed to focus on requirements and UML modelling in the first half and design, project management and quality in the second half.

Students will work on examples and problems in mixed classes to prepare them for the assignments which will target similar (but not the same) problem.

Students will learn how OO analysis and design relates to OO software development and also create non-OO analysis and designs and associated code.

Assignment 1 will be group based and Assignment 2 will be an individual project. Both assignment will be delivered in two phases.

A different textbook is being used.

**Changes from Last Year**

**Standards and Grading**

Firstly, in order to pass the unit, you must obtain a total mark of 50% or higher and a mark of 40% or higher in the final examination; submit at least one of the assignments and attend 75% of
mixed classes.

Secondly, ISYS254 will be graded according to the following general descriptions of the letter grades as specified by Macquarie University.

- High Distinction (HD, 85-100): provides consistent evidence of deep and critical understanding in relation to the learning outcomes. There is substantial originality and insight in identifying, generating and communicating competing arguments, perspectives or problem solving approaches; critical evaluation of problems, their solutions and their implications; creativity in application as appropriate to the discipline.

- Distinction (D, 75-84): provides evidence of integration and evaluation of critical ideas, principles and theories, distinctive insight and ability in applying relevant skills and concepts in relation to learning outcomes. There is demonstration of frequent originality in defining and analysing issues or problems and providing solutions; and the use of means of communication appropriate to the discipline and the audience.

- Credit (Cr, 65-74): provides evidence of learning that goes beyond replication of content knowledge or skills relevant to the learning outcomes. There is demonstration of substantial understanding of fundamental concepts in the field of study and the ability to apply these concepts in a variety of contexts; convincing argumentation with appropriate coherent justification; communication of ideas fluently and clearly in terms of the conventions of the discipline.

- Pass (P, 50-64): provides sufficient evidence of the achievement of learning outcomes. There is demonstration of understanding and application of fundamental concepts of the field of study; routine argumentation with acceptable justification; communication of information and ideas adequately in terms of the conventions of the discipline. The learning attainment is considered satisfactory or adequate or competent or capable in relation to the specified outcomes.

- Fail (F, 0-49): does not provide evidence of attainment of learning outcomes. There is missing or partial or superficial or faulty understanding and application of the fundamental concepts in the field of study; missing, undeveloped, inappropriate or confusing argumentation; incomplete, confusing or lacking communication of ideas in ways that give little attention to the conventions of the discipline.

| L.O 1 | Developing | Functional (P) | Proficient (Cr-D) | Advanced (D-HD) |
Practice the key phases of the software development life cycle (SDLC) including requirements engineering, analysis, design, basic development and testing and demonstrate understanding of alternative SDLC lifecycle models.

<table>
<thead>
<tr>
<th>L.O. 2</th>
<th>Developing</th>
<th>Functional</th>
<th>Proficient</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demonstrate an understanding of the concepts and tools needed to successfully design and build database-centric application programs using object-oriented and traditional methods and project management techniques.</td>
<td>Ability to use a CASE tool to create models but with limited understanding of how the models connect and how the tool can be effectively used to ensure model consistency.</td>
<td>Ability to use a CASE tool to create models, understanding of how the models connect and some understanding of how the tool can be effectively used to ensure model consistency.</td>
<td>Ability to use a CASE tool to create models, understanding of how the models connect and how the tool can be effectively and efficiently used to ensure model consistency.</td>
<td>Ability to use a CASE tool to create models, understanding of how the models connect and how the tool can be effectively and efficiently used to ensure model consistency. Familiarity with extended features of the tools and the role that these features can play in supporting the development of software.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>L.O.3</th>
<th>Developing</th>
<th>Functional</th>
<th>Proficient</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demonstrate ability to communicate software requirements and designs clearly and effectively.</td>
<td>Aware of documentation standards and notations but not able to apply them to the appropriate situation.</td>
<td>Able to follow a standard (e.g. IEEE standard for SRS) and appropriately use modelling notations. Able to document and read documentation concerning a software application.</td>
<td>Able to communicate at a functional level and also able to verify and validate documentation produced by themselves and others.</td>
<td>Able to communicate at a proficient level and also able to demonstrate a deep understanding of the role and relationship of various documents, activities, processes and roles which make up software development teams, projects, processes and products.</td>
</tr>
</tbody>
</table>