

ITEM	DETAILS
1. Title of subject	SYSTEMS ANALYSIS & DESIGN
2. Subject code	STC209
3. Status of subject	Core
4. Stage	Year 2
5. Credit Hour	4
6. Pre-Requisite	STC103 Structured Programming
7. Assessment	<p><b>40% Coursework</b>            Test 1 - 10%            Test 2 - 10%            Project – 20%</p> <p><b>60% Examination</b></p>
8. Semester	Semester 1
9. Objective of subject	<p>To enable students to:</p> <ul style="list-style-type: none"> <li>• Identify the best practices for developing the OO system in a more disciplined manner in order to fulfill the software engineering requirements.</li> <li>• Present a set of criteria, activities, and techniques to analyze business problems using use-case, structural and behavioural models.</li> <li>• Model OO systems in Unified Modelling Language (UML).</li> <li>• Design and implement system application architecture, user interface components, and object persistence components for an information system.</li> </ul>
10. Synopsis of subject	<p>Systems Analysis and Design is a software development process to organise the activities related to creation, delivery, and maintenance of software systems. The course provides a balanced coverage of Systems analysis and design activities using OO models and techniques. The course is divided into four parts. Part 1 introduces the system development life cycle (SDLC), Object Orientation and UML, Project planning, project initiation, and feasibility study. Part 2 presents techniques needed during the analysis phase. It focuses on requirements gathering that are used to create Use cases, Structural and Behavioural models. The Design phase is covered in part 3 of the course. It focuses on the system architecture</p>

	design, interface design and Object persistence design. The implementation phase is the final phase in SDLC. Implementation phase focuses on the system construction and systems Installation. It includes information about the test plan, user documentation, conversion plan, change management plan, support plan and project assessment.	
<b>11. Details of subject</b>	<b>Contents</b>	<b>Hours</b>
<b>Week 1 and Week 2</b>	<b>Topic:</b> <b>PART I</b> <b>SYSTEM DEVELOPMENT BASIC CONCEPTS</b> 1.0 Introduction to System Development 1.1 Different types of Information System 1.2 System Development Life Cycle (SDLC) 1.3 The evolution of System Development 1.4 Basic characteristics of OO Systems 1.5 Project Team Roles and Skills	8
	<b>Learning Outcomes:</b> At the end of the lessons, students will be able to: Comprehend the basic concepts of OO and Systems Development Life Cycle (SDLC)	
	<b>Further reading for this lesson:</b> Dennis, Wixon & Tegarden. (2002). Chapter 1.	
<b>Week 3</b>	2.0 Project Management 2.1 Creating the work plan 2.2 Staffing the project 2.3 Controlling and Directing the project 2.4 Tools to support System Development Process	4
	<b>Learning Outcomes:</b> At the end of the lessons, students will be able to: Understand in creating the work plan for the project, controlling and directing the project.	
	<b>Further reading for this lesson:</b> Dennis, Wixon & Tegarden. (2002). Chapter 3.	
<b>Week 4</b>	Planning Phase 2.5 Identifying Business Value 2.6 Feasibility Analysis	4
	<b>Learning Outcomes:</b> At the end of the lessons, students will be able to: Demonstrate the understanding of different feasibility analysis.	
	<b>Further reading for this lesson:</b> Dennis, Wixon & Tegarden. (2002). Chapter 2.	

<p><b>Week 5</b></p>	<p><b>PART II</b> <b>ANALYSIS PHASE</b> 3.0 Requirement Engineering     3.1 Requirement Analysis     3.2 Requirement Elicitation     3.3 Requirement Specification     3.4 Requirement Validation</p> <hr/> <p><b>Learning Outcomes:</b> At the end of the lessons, students will be able to: Understand the requirements for a new system, to analyze the information gathered and how to put it together into a form that documents how the To-Be system will operate.</p> <hr/> <p><b>Further reading for this lesson:</b> Dennis, Wixon &amp; Tegarden. (2002). Chapter 4 and Chapter 5.</p>	<p>4</p>
<p><b>Week 6</b></p>	<p>4.0 Object Modelling     4.1 An overview of UML     4.2 The Class Diagram         4.2.1 Creating Class-Responsibility-Collaboration (CRC) cards and Class Diagram         4.2.2 Data Dictionary</p> <hr/> <p><b>Learning Outcomes:</b> At the end of the lessons, students will be able to: Comprehend the data structure of an OO systems, class responsibilities and their collaborations, drawing of class diagrams.</p> <hr/> <p><b>Further reading for this lesson:</b> Dennis, Wixon &amp; Tegarden. (2002). Chapter 7.</p>	<p>4</p>
<p><b>Week 7</b></p>	<p><b>PART III</b> <b>DESIGN PHASE</b> 5.0 Moving On To Design     5.1 Evolving the Analysis Model to Design Model     5.2 Packages and Package Diagram     5.3 Design Strategies</p> <hr/> <p><b>Learning Outcomes:</b> At the end of the lessons, students will be able to: Comprehend the transition from analysis and design, use of factoring, partitions and layers and to understand the custom, packaged and outsource design.</p> <hr/> <p><b>Further reading for this lesson:</b> Dennis, Wixon &amp; Tegarden. (2002). Chapter 9.</p>	<p>4</p>

<p><b>Week 8</b></p>	<p>5.4 Class Diagram at the Design Stage            5.4.1 Class and Method Design                5.4.1.1 Design Criteria                5.4.1.2 Object Design Activities                5.4.1.3 Constraints and Contracts                5.4.1.4 Method Specification</p> <hr/> <p><b>Learning Outcomes:</b>            At the end of the lessons, students will be able to:            Understand coupling, cohesion and connascence.            Specify restructure and optimise object designs and to specify constraints and contracts.</p> <hr/> <p><b>Further reading for this lesson:</b>            Dennis, Wixon &amp; Tegarden. (2002). Chapter 14.</p>	<p>4</p>
<p><b>Week 9</b></p>	<p>5.5 Relationships at the Design Stage            5.6 Sequence diagrams at the Design Stage</p> <hr/> <p><b>Learning Outcomes:</b>            At the end of the lessons, students will be able to:</p> <ul style="list-style-type: none"> <li>• Understand the usage of sequence diagrams at the design stage.</li> </ul> <hr/> <p><b>Further reading for this lesson:</b>            Dennis, Wixon &amp; Tegarden. (2002). Chapter 13.</p>	<p>4</p>
<p><b>Week 10 &amp; Week 11</b></p>	<p><b>PART IV</b>  <b>IMPLEMENTATION PHASE</b>            6.0 System Construction                6.1 Managing Programming                6.2 Designing Test                6.3 Developing Documentation                6.4 Case Study                    6.4.1 Project Example: Elevator Simulator                    6.4.2 Illustrate the relationship between Implementation and Design</p> <hr/> <p><b>Learning Outcomes:</b>            At the end of the lessons, students will be able to:</p> <ul style="list-style-type: none"> <li>• Understand different types of tests and when to use them and how to develop documentation.</li> </ul> <hr/> <p><b>Further reading for this lesson:</b>            Dennis, Wixon &amp; Tegarden. (2002). Chapter 15.</p>	<p>8</p>
<p><b>Week 12</b></p>	<p><b>PART V</b>  <b>TESTING</b>            7.0 An overview of Testing                7.1 Stages of testing                7.2 Object-oriented Testing</p> <hr/> <p><b>Learning Outcomes:</b>            At the end of the lessons, students will be able to:</p> <ul style="list-style-type: none"> <li>• Comprehend how to do different types of testing.</li> </ul>	<p>4</p>

	<b>Further reading for this lesson:</b> Dennis, Wixon & Tegarden. (2002). Chapter 15.		
<b>Week 13 &amp; Week 14</b>	<b>PART V</b> <b>TESTING</b> 8.0 An overview of Testing 8.1 Stages of testing 8.2 Object-oriented Testing  9.0 Report Writing Report Format		8
	<b>Learning Outcomes:</b> At the end of the lessons, students will be able to: <ul style="list-style-type: none"> <li>• Comprehend how to do different types of testing.</li> </ul>		
	<b>Further reading for this lesson:</b> Dennis, Wixon & Tegarden. (2002). Chapter 15.		
	<b>Total</b>		56
<b>12. Text</b>	<b>Compulsory</b>	<ol style="list-style-type: none"> <li>1. Hoffer, J. A. , George, J. F., &amp; Valacich, J. S. (2001). <i>Modern Systems Analysis &amp; Design</i> (3<sup>rd</sup> ed.). Prentice Hall.</li> <li>2. Britton, C., &amp; Doake, J. (2000). <i>Object-Oriented Systems Development: A Gentle Introduction</i>. McGraw Hill.</li> <li>3. Dennis, A., Wixon, B. H., &amp; Tegarden, D. (2002). <i>Systems Analysis and Design: An Object-oriented Approach with UML</i>. John Wiley.</li> </ol>	
	<b>Reference</b>	<ol style="list-style-type: none"> <li>1. Page, M. (1998). <i>Practical Guide to Structured Systems Design</i>. Prentice Hall.</li> <li>2. Kendall, P. A. (1995). <i>Introduction to Systems Analysis &amp; Design: A Structured Approach</i> (3<sup>rd</sup> ed.). McGraw Hill / Irwin.</li> <li>3. Yourdon, E. (2000). <i>Modern Structured Analysis</i> (2<sup>nd</sup> ed.). New Jersey: Prentice Hall.</li> <li>4. Bennett, S., McRobb, S., &amp; Farmer, R. (2001). <i>Object-oriented Systems Analysis and Design using UML</i> (2<sup>nd</sup> ed.). McGraw Hill.</li> </ol>	