4.1 Mainframe Application Development 1 (COBOL)

Module Code:	
Credits:	10
Credit Level:	8
Prerequisite Modules:	None
Co-requisite Modules:	None
Anti-requisite Modules:	None

Total Hours	
Classroom	48
Tutorials	-
Practicals	
Virtual-Class	24
Total	**72**

MODULE AIMS:

- To develop legacy mainframe applications using the COBOL programming language.
- To employ standard interactive mainframe tools such as TSO, ISPF and SDSF.
- To develop and execute mainframe scripts using the Job Control Language

MODULE LEARNING OUTCOMES:

A learner who successfully completes this module will be able to:

1. Apply the appropriate primary and line commands primary and line commands, to create and maintain mainframe datasets and members using the end-user interface programs TSO and ISPF.
2. Create, edit and submit a batch Job on the mainframe using Job Control Language (JCL), and provide proof of the result using the SDSF tool.
3. Evaluate the three primary programming principles of sequence, selection and iteration.
4. Represent a problem using a Jackson's Structured Diagram.
5. Design and develop a series of programs that apply the report-generation and master file processing concepts using the high-level COBOL programming language.

SYLLABUS CONTENT:

Mainframe Environment

- Introduction to the IBM Zeus mainframe environment.
- Mainframe concepts and terms.
- Using Interactive System Productivity Facility (ISPF) and System Display and Search Facility (SDSF) within the Time Sharing Option (TSO) mainframe environment.
- The basics of Job Control Language (JCL)
- How to use JCL and JES facilities to manage program execution.

Problem Analysis, Design Concepts using Jacksons Structured Design

- Applying a structured approach to problem design.
- Jacksons Structured Design (JSD) rules and principles.
- Using a software tool to represent JSD.

COBOL Mainframe Programming Language

- Introduction to COBOL programming.
- Compiling, testing and debugging a COBOL program.
- How to write a program that prepares a report.
- How to design, code, and test a structured program
- Using recognised COBOL features for structured coding.
- Working with files.
- Using datasets to create Sequential Access Method (SAM) file types.
- Writing a COBOL program to process input and output data files.

MODULE ASSESSMENT:

Continuous Assessment: 100%

Learning Outcome	Final Exam	Continuous Assessment	Project	Lab Exercise	Quiz
1		√		√	√
2		√		√	√
3		√		√	√
4		√		√	√
5		√	√	√	√

Coursework may comprise a mix of assessment approaches, such as: on-the-job practical work and/or project, reports, practicals, presentations, portfolios, class tests, quizzes, group work and integrated assessment. Details of the nature of assessment and submission dates are contained in the annual Programme Assessment Schedule.

Continuous Assessment:

Continuous Assessment will comprise of practical work using remote access to IBM Zeus Mainframe, assignments and VLE quizzes over the semester. The student will be required to submit **3** items of continuous assessment with the following weightings and complete VLE quizzes:

Assignment Number and Topic	**Weighting**
Assignment 1 Maintain a Cobol program	15%
Assignment 2 Write Cobol program (files)	35%
Assignment 3 Work Based (company specific) Cobol update program using SAM and VSAM file types.	45%
VLE – Lesson Quizzes	5%

RECOMMENDED READING AND OTHER MEDIA:

Essential:				
Author	**Year**	**Title**	**Publisher**	**ISBN**
Raul Menendez, Doug Lowe	1998	Murach's OS/390 and OS JCL	Murach	1-890774-14-6
Doug Lowe	2000	Murach's MVS TSO, Concepts and ISPF(MVS TSO)	Murach	978-0-911625-56-1
Mike Murach	2000	Murach's Mainframe COBOL	Murach	1-890774-24-3
David Stephens	2008	What On Earth is a Mainframe	Longpela Expertise	978-14092-2535-5