STANDARD KEMAHIRAN PEKERJAAN KEBANGSAAN
(NATIONAL OCCUPATIONAL SKILLS STANDARD)

APPLICATION DEVELOPMENT
LEVEL 3

JPK

Jabatan Pembangunan Kemahiran
Kementerian Sumber Manusia, Malaysia

Department of Skills Development (DSD)

Ministry of Human Resources

62530 PUTRAJAYA, MALAYSIA

STANDARD KEMAHIRAN PEKERJAAN KEBANGSAAN (NATIONAL OCCUPATIONAL SKILLS STANDARD)

FOR

APPLICATION DEVELOPMENT LEVEL 3

LIST OF ABBREVIATIONS

CSS	Cascading Style Sheets
DML	Data Manipulation Language
EAI	Enterprise Application Integrator
ERD	Entity Relation Diagram
EULA	End User License Agreement
FDS	Functional Design Specification
FTP	File Transfer Protocol
HTML	Hypertext Mark up Language
HMVC	Hierarchical Model View Controller
HTTP	Hypertext Transfer Protocol
IDE	Integrated Development Environment
JDK	Java Development Kit
LAMP	Linux, Apache, MySQL, PHP
MVA	Model View Adapter
MVC	Model View Controller
MVP	Model View Presenter
OAuth	Open Authorization
OOP	Object Oriented Programming
PHP	PHP: Hypertext Pre-processor
RFC	Request For Comments
SSL	Software Security License
SRS	Software Requirements Specification
SOP	Standard Operating Procedure
SCM	Source Code Management

SDD	Software Design Document
SDK	Software Development Kit
SDLC	Software Development Life Cycle
SQL	Structured Query Language
SSH	Secure Shell Protocol
SSO	Single Sign On
TDD	Test Driven Development
UI	User Interface
UML	Unified Modelling Language
UX	User Experience
WAMP	Windows, Apache, MySQL, PHP

TABLE OF CONTENTS

STANDARD PRACTICE

NATIONAL OCCUPATIONAL SKILLS STANDARD (NOSS) FOR

APPLICATION DEVELOPMENT

LEVEL 3

1. INTRODUCTION

The government has strengthened the development of the Information and Technology (ICT) sector through the Economic Transformation Programme (ETP), which is driven by twelve (12) National Key Economic Areas (NKEA). In the NKEA of Communications Content and Infrastructure, the first Entry Project Point (EPP 1) is nurturing Malaysia's Creative Content. This EPP aims to enhance capacity, capability and competency in Malaysia's creative industry to produce world-class content and make the country a regional hub for digital content.

Digital Malaysia aims to turn Malaysia into a developed digital economy by 2020, which connects and empowers the government, businesses and citizens. Today, both MSC Malaysia and Digital Malaysia run con-currently to spur Malaysia's ICT industry development and digital economic growth, under the purview of Malaysian Development Corporation (MDeC).

In addition, the pioneer Human Capital Report 2013 produced by WEF ranks Malaysia 22nd out of 122 economies surveyed in terms of best performance by a higher to middle-income nation in the area of workforce and employment. Multimedia Super Corridor (MSC) Malaysia is estimated to provide 50,000 total jobs and MYR$ 5.5bil investments by 2015 with cloud based computing services set to become the next engine of growth.

Occupational Overview

An application programmer is the person who writes, tests and maintains the details of instructions that computers must follow to perform their functions. They write the step-by-step instructions that direct computers and mobiles to process information in a series of logical steps. This involves establishing a detailed specification and clarifying exactly what the programme needs to do, breaking the specification down to its simplest elements and translating the information into an appropriate programming language. Different tasks require different programming languages such as Java (J2EE), C++, etc. Most application programmers specialise in different languages and their work nature depends on their respective employers.

An application programmer is also responsible for designing and modifying computer applications. Related tasks include testing, debugging and maintaining the source code. They must be familiar with computer hardware, computing systems theories, programming languages and software structure. As technology is ever changing, application programmers must be very adaptable and willing to learn new techniques.

Justification and Rational of NOSS development

Employment opportunities for programmers are expected to grow as programming tasks become increasingly sophisticated and employers are demanding high competency of skills and experiences from employees. Job prospects will provide more opportunity for programmers who possess formal education as well as having the ability to do programming in several different languages and tools – including C++ and other object-oriented languages such as Java.

Courses in educational and training institutes must be designed within a framework that is aligned with nation's agenda to benefit the growth and sustainability of the country. This NOSS development is an initiative to support the high demand for skilled personnel in the area of application development and to ensure that training in this occupational area meets industry demand and requirements.

Authority and Regulatory/Statutory Bodies Related to Industry

Currently, there are a number of authorities and regulator in Malaysia. They are as follow:

- Ministry of Communications and Multimedia Malaysia (MCMM)
- Malaysian Development Corporation (MDeC)
- Malaysian Multimedia and Communication Commission (MCMC)
- Malaysian Technology Development Corporation (MTDC)
- Malaysian Intellectual Property Organisation (MyIPO)
- Malaysian Creative Content Association (MCCA)

2. OCCUPATIONAL STRUCTURE

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**				
SUB SECTOR	**SOLUTION DEVELOPMENT**				
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**
L5	SYSTEM ANALYST / TECHNICAL MANAGER / SOLUTION ARCHITECT	SYSTEM ANALYST / TECHNICAL MANAGER / SOLUTION ARCHITECT	WEB SYSTEM ANALYST / TECHNICAL MANAGER / SOLUTION ARCHITECT	DATABASE SYSTEM ANALYST / TECHNICAL MANAGER / SOLUTION ARCHITECT	SERVER SYSTEM ANALYST / TECHNICAL MANAGER / SOLUTION ARCHITECT
L4	MOBILE APPLICATION LEAD PROGRAMMER	APPLICATION LEAD PROGRAMMER	WEB LEAD PROGRAMMER	DATABASE LEAD PROGRAMMER	SERVER LEAD PROGRAMMER
L3	APPLICATION PROGRAMMER				
L2	NO LEVEL				
L1	NO LEVEL				

Fig. 1.1 Existing Occupational Structure for Information & Communication Technology (ICT) sub-sector Solution Development in Malaysia

OCCUPATIONAL AREA STRUCTURE

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**				
SUB SECTOR	**SOLUTION DEVELOPMENT**				
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**
L5	MOBILE APPLICATION DEVELOPMENT MANAGEMENT	SYSTEMS IMPLEMENTATION INTEGRATION	WEB DEVELOPMENT	DATABASE PROGRAMMING	SERVER APPLICATION DEVELOPMENT AND MANAGEMENT
L4	MOBILE APPLICATION DEVELOPMENT ADMINISTRATION	SYSTEMS MODULE DEVELOPMENT	WEB DEVELOPMENT	DATABASE PROGRAMMING	SERVER APPLICATION DEVELOPMENT AND MANAGEMENT
L3	APPLICATION DEVELOPMENT				
L2	NO LEVEL				
L1	NO LEVEL				

Fig. 1.1 Occupational Area Structure (OAS) for Information & Communication Technology (ICT) sub-sector Solution Development in Malaysia

3. DEFINITION OF COMPETENCY LEVELS

The NOSS is developed for various occupational areas. Candidates for certification must be assessed and trained at certain levels to substantiate competencies. Below is a guideline of each NOSS Level as defined by the Department of Skills Development, Ministry of Human Resources, Malaysia.

Level 1:	Competent in performing a range of varied work activities, most of which are routine and predictable.
Level 2:	Competent in performing a significant range of varied work activities, performed in a variety of contexts. Some of the activities are non-routine and required individual responsibility and autonomy.
Level 3:	Competent in performing a broad range of varied work activities, performed in a variety of contexts, most of which are complex and non-routine. There is considerable responsibility and autonomy and control or guidance of others is often required.
Level 4:	Competent in performing a broad range of complex technical or professional work activities performed in a wide variety of contexts and with a substantial degree of personal responsibility and autonomy. Responsibility for the work of others and allocation of resources is often present.
Level 5:	Competent in applying a significant range of fundamental principles and complex techniques across a wide and often unpredictable variety of contexts. Very substantial personal autonomy and often significant responsibility for the work of others and for the allocation of substantial resources features strongly, as do personal accountabilities for analysis, diagnosis, planning, execution and evaluation.

4. AWARD OF CERTIFICATE

The Director General shall award, to any person upon completing successfully the NOSS program following skills level qualifications:

a) Malaysia Skills Certificate / Sijil Kemahiran Malaysia (SKM) Level 1, 2 & 3
b) Malaysia Skills Diploma / Diploma Kemahiran Malaysia (DKM) Level 4
c) Malaysia Skills Advanced Diploma / Diploma Lanjutan Kemahiran Malaysia (DLKM) Level 5
d) Statement of Achievement / Penyata Pencapaian (PC)

No person shall be awarded a Certificate unless he / she satisfy the requirements set by Malaysia Skills Certification System.

5. JOB COMPETENCIES

An Application Programmer Level 3 is competent in performing:

- Application Prototype Development
- Application Module Development
- Application Module Integration
- Development Environment Deployment
- Application Bug Fixing
- Application System Documentation Compilation
- Application Development Supervision

6. WORKING CONDITIONS

Programmers generally work in offices in comfortable surroundings. Many programmers may work long hours or weekends to meet deadlines or fix critical problems that occur during off hours. Telecommuting is becoming common for a wide range of computer professionals, including computer programmers. As computer networks expand, more programmers are able to make corrections or fix problems remotely using modems, e-mail, and the Internet to connect to a customer's computer. Software development often involves working in a group, so application programmers must have a team-oriented attitude. The ability to write instruction manuals and maintenance instructions is also required.

7. EMPLOYMENT PROSPECTS

Programmers are employed in almost every industry, but the largest concentration is in computer systems design and related services. Large numbers of programmers also work for telecommunications companies, software publishers, financial institutions, insurance careers, educational institutions and government agencies.

Programmers may also work as an independent consultant, providing expertise with new programming languages or specialised areas of application. Programmers can outsource their jobs to other countries. They can perform their job function from anywhere in the world and can digitally transmit their programs to any location via e-mail.

Other related occupations with respect to employment opportunities are:

- Application Development Executive
- System Software Engineer
- System Administration
- Database Administrator
- System Analyst
- Software Architect

- Computer Support Specialist

Other related industries with respect to employment opportunities are:
- Telecommunications Companies
- Software Publishers
- Financial Institutions
- Insurance Careers
- Educational Institutions
- Government Agencies

8. CAREER ADVANCEMENT

As for career advancement, most competent application programmers' level 3 gains their competency on the job. They gradually learn new skills as they gain experience for career advancement. They may also enhance their knowledge and skills by attending professional courses provided by their employer or by software vendors.

9. SOURCES OF ADDITIONAL INFORMATION

LOCAL

- **Department of Skills Development**
  Kementerian Sumber Manusia, Aras 7 & 8
  Parcel D, Blok D4 Pusat Pentadbiran Kerajaan Persekutuan
  Putrajaya, Malaysia
  Tel: 03-8886 5000
  Website: http://www.dsd.gov.my

- **Multimedia Development Corporation (MDeC)**
  MSC Malaysia Headquaters
  Persiaran APEC, 63000, Cyberjaya
  Selangor Darul Ehsan, Malaysia
  Tel: 1-800-88-8338
  Fax: +603-83153115
  Website: www.mscmalaysia.my

- **Persatuan Industri Komputer dan Multimedia Malaysia (PIKOM)**
  E1, Empire Damansara
  E-01-G, No,2, Jalan PJU 8/8A, Damansara Perdana
  47820 Petaling Jaya, Selangor
  Tel: 603-40650078
  Fax: 603-40650079
  Website: http://www.pikom.org.my/

- **Malaysian Communications and Multimedia Commission (MCMC)**
  MCMC Tower 1
  Jalan Impact, Cyber 6
  63000 Cyberjaya
  Selangor Darul Ehsan
  Tel: +60 3 8688 8000
  Fax: +60 3 8688 1000
  Website: www.mcmc.gov.my

INTERNATIONAL

- **W3C (World Wide Web Consortium)**
  32 Vassar Street
  #32G515, Cambridge, MA 02139
  Tel: (617) 253-2613
  Website: http://www.w3.org/

- **Association for Computing Machinery**
  2 Penn Plaza, Suite 701
  New York, NY 10121-0701
  Phone: +1 212 626-0500 (Global)
  Fax: +1 212 944-1318 (Global)
  Email: acmhelp@acm.org

- **IETF (Internet Engineering Task Force)**
  IETF Secretariat
  c/o Association Management Solutions, LLC (AMS)
  48377 Fremont Blvd., Suite 117
  Fremont, California 94538, USA
  Phone: +1-510-492-4080
  Fax: +1-510-492-4001
  Website: www.ietf.org

10. ACKNOWLEDGEMENT

The Director General of DSD would like to extend his gratitude to the organisations and individuals who have been involved in developing this standard; especially members of Standard Technical Evaluation Committee (STEC) for validated this document.

PANEL PENILAI		
1	Situah Ariff bin Zakaria	MMSC System Sdn Bhd
2	Syamsul Hani bin Hasran	Nusantara Software Sdn Bhd
3	Shafful Shazwan Bin Sahar	Malaysian Academy of Creative Technology Sdn Bhd

11. COMMITTEE MEMBERS FOR DEVELOPMENT OF STANDARD PRACTICE (SP), COMPETENCY PROFILE CHART (CPC), COMPETENCY PROFILE (CP) AND CURRICULUM of COMPETENCY UNIT (CoCU)

APPLICATION DEVELOPMENT- LEVEL 3

DEVELOPMENT COMMITTEE		
1.	Arbaayah Binti Kamarudin	System Executive Intergrated Knowledge & Campus Management Sdn Bhd
2.	Muhammad Ikmal Ezzani Bin Ruslan	Lead Mobile Developer Ikez Network Sdn Bhd
3.	Hamizah Binti Abdul Halim	System Executive Intergrated Knowledge & Campus Management Sdn Bhd
4.	Nurul Huda Binti Zahari	Consultant IFCA MSC Berhad
5.	Mohd Amirullah Bin Mohd Ali Napiah	Java Developer MMSC Group Company Sdn Bhd
6.	Abdul Hakim Bin A.Gafa	Senior Programmer Theta-Edge Berhad
7.	Suhaidee Bin Ahmed	Penolong Pegawai Teknologi Maklumat Jabatan Pendaftaran Negara
8.	Zairul Rizwan Bin Ruslan	Programmer Konsortium Jaya Sdn Bhd
FACILITATOR		
1.	Mastura Liza binti Muhammad	PFH Resources (M) Sdn Bhd

COMPETENCY PROFILE CHART (CPC)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**				
SUB SECTOR	**SOLUTION DEVELOPMENT**				
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**
NOSS TITLE	**APPLICATION DEVELOPMENT**				
JOB LEVEL	**THREE (3)**		**NOSS CODE**	**IT-010-3:2016**	

COMPETENCY ← **COMPETENCY UNIT** →

CORE	**APPLICATION PROTOTYPE DEVELOPMENT**	**APPLICATION MODULE DEVELOPMENT**	**APPLICATION MODULE INTEGRATION**	**DEVELOPMENT ENVIRONMENT DEPLOYMENT**
	IT-010-3:2016 –C01	**IT-010-3:2016 –C02**	**IT-010-3:2016 –C03**	**IT-010-3:2016 –C04**
	APPLICATION BUG FIXING	**APPLICATION SYSTEM DOCUMENTATION COMPILATION**	**APPLICATION DEVELOPMENT SUPERVISION**	
	IT-010-3:2016 –C05	**IT-010-3:2016 –C06**	**IT-010-3:2016 –C07**	

COMPETENCY PROFILE (CP)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**				
SUB SECTOR	**SOLUTION DEVELOPMENT**				
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**
NOSS TITLE	**APPLICATION DEVELOPMENT**				
LEVEL	**THREE (3)**		**NOSS CODE**	**IT-010-3:2016**	

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
1. Application Prototype Development	IT-010-3:2016 – C01	Application prototype development describes the competencies required to build/create application prototype that simulates a few aspects of the final product such as user interface or application flow. The person who is competent in this CU shall be able to interpret application prototype development requirement, setup local environment, implement application prototype mock up flow, conduct user interface and user experience functionality test and commit prototype source code. The outcome of this competency is a prototype that resembles the final product's user interface or application flow as per Software	1. Interpret application prototype development requirement 2. Setup local environment	1.1 Development timeline confirmed according to Software Requirement Specification (SRS) 1.2 Modules number to be developed confirmed according to job brief 1.3 Task assignation confirmed according to job brief 1.4 Application prototype mock up design interpreted according to storyboard 1.5 Application prototype flow interpreted according to storyboard 2.1 Local environment requirement interpreted according to job brief 2.2 Programming language to be used confirmed according to job brief 2.3 Local server access

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
		Requirement Specification (SRS).		configuration obtained according to job brief 2.4 Development kit installed according to installation procedure 2.5 Development stack installed according to installation procedure 2.6 Local server installed according to installation procedure
			3. Implement application prototype mock up flow	3.1 User interface and user experience flow interpreted according to storyboard/functional specification 3.2 Prototype module codes written according to application flow
			4. Conduct user interface and user experience functionality test	4.1 User interface function tested according to application flow 4.2 User interface conformed to comply with user experience flow
			5. Commit prototype source code	5.1 Source code repository destination determined according to job brief 5.2 Prototype source code uploaded to source code repository 5.3 Work progress report updated according to company's

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
				Standard Operating Procedure (SOP) 5.4 Prototype source code submission reported to superior
2. Application Module Development	IT-010-3:2016 – C02	Application module development is the process of coding, testing, debugging & maintaining the source code. The person who is competent in this CU shall be able to interpret application module development requirement, setup local environment, plan module expected behaviour, write module code and commit source code. The outcome of this competency is to develop an application that functions, deployment ready, error free and as per Software Requirement Specification (SRS).	1. Interpret application module development requirement 2. Setup local environment	1.1 Module function and specification interpreted according to Software Requirement Specification (SRS) 1.2 Development timeline confirmed according to job brief 1.3 Task assignation confirmed according to job brief 1.4 Application flow interpreted according to Software Requirement Specification (SRS) 1.5 Application mock up design interpreted according to storyboard 1.6 Third party component requirement (payment gateway, security certificate) interpreted according to Software Requirement Specification (SRS) 2.1 Local environment requirement interpreted according to job brief 2.2 Database structure interpreted according to job

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
				brief 2.3 Language programming to be used confirmed according to job brief 2.4 Local server access configuration confirmed according to job brief 2.5 Development server access configuration confirmed according to job brief 2.6 Source Code Management (SCM) access confirmed according to job brief 2.7 Development kit (Integrated Development Environment IDE)) installed according to installation procedure 2.8 Development stack (WAMP, LAMP) installed according to installation procedure 2.9 Local server installed according to installation procedure 2.10 Local database server installed according to installation procedure 2.11 Source Code Management (SCM) software installed according to installation procedure
			3. Plan module expected behaviour	3.1 Module expected scenarios listed out according to Software Requirement Specification (SRS)

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
				3.2 Module expected input and output listed out according to Software Requirement Specification (SRS) 3.3 Module test script written according to Software Requirement Specification (SRS)
			4. Write module code	4.1 Variables named according to coding/naming convention 4.2 Instruction code written to perform module function according to Software Requirement Specification (SRS) 4.3 Database connection associated with source code in local server 4.4 Structured Query Language (SQL) statement written according to data flow in the Software Requirement Specification (SRS) 4.5 Coding developed according to test script 4.6 Module output verified according to Software Requirement Specification (SRS) 4.7 Module code debugged according to module output
			5. Commit module source code	5.1 Source code repository destination determined

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
				according to job brief 5.2 Module source code uploaded to source code repository 5.3 Work progress report updated according to company's Standard Operating Procedure (SOP) 5.4 Application source code submission reported to superior
3. Application Module Integration	IT-010-3:2016 – C03	Application module integration is a process where modules are combined and tested as a group. Integration testing focuses on checking data communication amongst these modules. The person who is competent in this CU shall be able to interpret module integration requirement, perform module integration, test module integration code and commit module integration code. The outcome of this competency is to ensure that all modules are integrated correctly, functional and ready for system testing.	1. Interpret module integration requirement 2. Perform modules integration	1.1 Module to be integrated identified according to Software Requirement Specification (SRS) 1.2 Source code repository location confirmed according to job brief 1.3 Module source code retrieved from source code repository 2.1 Related module associated with database table according to Software Requirement Specification (SRS) 2.2 Folder asset data extraction established according to Software Requirement Specification (SRS) 2.3 Integration code between modules wrote according to Software Requirement Specification (SRS) 2.4 Integration between modules executed according to

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
				Software Requirement Specification (SRS)
			3. Test module integration code	3.1 Modules integration functionality test conducted according to Software Requirement Specification (SRS) 3.2 Module integration output verified according to Software Requirement Specification (SRS) 3.3 Module integration code debugged according to Software Requirement Specification (SRS)
			4. Commit module integration code configuration	4.1 Integrated module source code uploaded to source code repository 4.2 Work progress report updated according to company's Standard Operating Procedure (SOP) 4.3 Integrated module source code submission reported to superior

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
4. Development Environment Deployment	IT-010-3:2016 – C04	Development environment deployment is a process of deploying and executing source code, data and third party component before it goes for staging. Integration system testing focuses on checking third party components compatibility with application modules. The person who is competent in this CU shall be able to deploy source code to development server, perform data population to development server and perform integration system testing. The outcome of this competency is to establish an environment that is as identical to the production environment, which is reliable and error free.	1. Deploy source code to development server 2. Perform data population to development server 3. Perform integration system testing	1.1 Source code compiled according to application technical specification 1.2 Source code uploaded to development server according to application technical specification 1.3 Development database connection associated with source code in development server 2.1 Development database populated with required data set 2.2 Application behaviour and data flow tested with functional specification 3.1 Third party component installed according to Software Requirement Specification (SRS) 3.2 Third party component integrated with application module 3.3 Third party component compatibility with application module tested according to Software Requirement Specification (SRS)

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
5. Application Bug Fixing	IT-010-3:2016 – C05	Application bug fixing is a process of tracking and removing error, flaw, failure, or fault in an application that causes it to produce an incorrect or unexpected result, or to behave in unintended ways. The person who is competent in this CU shall be able to interpret bug report, simulate bug/error scenario, debug application code, and commit fixed source code. The outcome of this competency is to ensure that the application developed is error free.	1. Interpret bug report	1.1 Classification of error (unresponsive input, cosmetic error, navigation error, out of memory error, unexpected error or server error) identified according to bug report 1.2 Cause of error (server down time, unexpected usage or unexpected application behaviour) confirmed according to bug report
			2. Simulate bug/error scenario	2.1 Application error scenario reproduced according to bug report 2.2 Application error occurrence confirmed according to bug report
			3. Debug application code	3.1 Application source code retrieved from source code repository 3.2 Cause of error identified according to bug report 3.3 Affected source code fixed according to debugging procedure 3.4 Initial error scenario reproduced according to bug report 3.5 No bug occurrence confirmed according to simulation result 3.6 Bug fixes confirmed according to simulation result

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
			4. Commit fixed source code	4.1 Fixed source code published to source code repository 4.2 Bug fixes status reported to superior
6. Application System Documentation Compilation	IT-010-3:2016 – C06	Application system documentation compilation is a process of collecting and arranging source code documents and user manual materials in good order. The person who is competent in this CU shall be able to interpret system documentation requirement, compile system documentation and store system documentation. The outcome of this competency is accurate information and well organized documents compiled according to application module. System documentation compiled are secured and kept safely in source code repository.	1. Interpret system documentation requirement 2. Compile system documentation material 3. Upload system documentation	1.1 Application system documentation requirement identified according to job brief 1.2 Company's guideline of application system documentation referred 2.1 Source code written according to company's document control guideline 2.2 Source code documentation generated according to company's document control guideline 2.3 User manual material (screenshot, menu) compiled according to company's document control guideline 2.4 User manual material organized according to application module 3.1 System documentation storage destination confirmed in source code repository 3.2 System documentation uploaded to source code repository 3.3 System documentation

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
				access granted to technical documentation writer
7. Application Development Supervision	IT-010-3:2016 – C07	Application development supervision is a work process of executing administrative responsibilities and to enforce SOP. This competency unit outlines work scope of administrative functions as stipulated in company's job descriptions and SOP. The person who is competent in supervisory functions shall be able to confirm facilities and equipment functionality, prepare job schedule and perform unit meeting / briefing. The outcome of this competency is to enable trainee to perform supervisory skills to support operation according to company's requirements.	1. Confirm facilities and equipment functionality 2. Prepare job schedule	1.1 Type of facilities and equipment determined as per checklist 1.2 Facilities and equipment availability & functionality confirmed as per checklist 1.3 Work requisition on facilities and equipment malfunction raised as per company's SOP 1.4 Status of malfunction facilities and equipment followed up 1.5 Status of facilities and equipment record updated 2.1 Types and function of scheduling determined as per company's scheduling procedure 2.2 Scope of work & job descriptions interpreted as per company's scheduling procedure 2.3 Subordinates competency status validated as per operation requirements 2.4 Number of manpower verified as per staffing record 2.5 Personnel assigned for duty as per operations requirements

CU Title	CU Code	CU Descriptor	CU Work Activities	Performance Criteria
				2.6 Duty roster / jobs schedule generated based on operations requirements
			3. Conduct unit meeting / briefing	3.1 Daily staff briefing conducted as per operation requirement 3.2 Current operational issues communicated during daily staff briefing 3.3 Unit meeting conducted as per meeting procedure 3.4 Agenda of meeting discussed as per meeting procedure 3.5 Internal communication activities documented for future reference

CURRICULUM of COMPETENCY UNIT (CoCU)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**								
SUB SECTOR	**SOLUTION DEVELOPMENT**								
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**				
NOSS TITLE	**APPLICATION DEVELOPMENT**								
COMPETENCY UNIT TITLE	**APPLICATION PROTOTYPE DEVELOPMENT**								
LEARNING OUTCOME	The person who is competent in this CU shall be able to build/create application prototype that resembles the final product's user interface or application flow as per SRS. Upon completion of this CU trainee shall be able to: • Interpret application prototype development requirement • Setup local environment • Implement application prototype mock up flow • Conduct user interface and user experience functionality test • Commit prototype source code								
PRE-REQUISITE (if applicable)	Nil								
COMPETENCY UNIT ID	IT-010-3:2016 –C01	**LEVEL**	3	**TRAINING DURATION**	180	**SKILL CREDIT**	18		

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
1. Interpret application prototype development requirement	i. Function of Software Requirement Specification (SRS) ii. Introduction to scripting language such as: • Cascading Style Sheets (CSS) • Hyper Text Markup Language (HTML) • JavaScript • PHP: Hypertext PreProcessor	i. Obtain project brief ii. Identify development timeline iii. Determine modules number to be developed iv. Identify task assignation v. Check application prototype mock up design vi. Check application	Attitude: i. Proactive when interpret application prototype development requirement ii. Resourceful when interpret application prototype development	Related Knowledge 7 Related Skills 11	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Function of Software Requirement Specification (SRS) explained ii. Types of scripting language listed iii. Function of Integrated Development Environment

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	(PHP) • Structured Query Language (SQL) iii. Introduction to Integrated Development Environment (IDE) iv. Project brief content such as: • Development timeline • Modules number • Task assignation v. Fundamentals of application prototype vi. Awareness of End User License Agreement (EULA) vii. Software licensing such as: • Proprietary • Open source	prototype flow	requirement iii. Committed when interpret application prototype development requirement iv. Analytical thinking when interpret application prototype development requirement v. Adhere to End User License Agreement (EULA) <u>Safety:</u> i. Adhere to workplace ergonomics practice			(IDE) explained iv. Types of project brief content listed v. Purpose of End User License Agreement (EULA) explained vi. Types of software licensing listed vii. Application prototype development requirement confirmed viii. Workplace ergonomic practice explained
2. Setup local environment	i. Introduction to development environment ii. Introduction to local server such as: • Local server access configuration • Installation	i. Interpret local environment requirement ii. Determine programming language to be used iii. Obtain local server access configuration iv. Install development kit	<u>Attitude:</u> i. Proactive when setup local environment ii. Resourceful when setup local environment	<u>Related Knowledge</u> 22 <u>Related Skills</u> 32	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Definition of development environment explained ii. Definition of local server explained iii. Definition of Source Code Management

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	procedure iii. Introduction to Source Code Management (SCM) such as: • Source code repository • Branching • Revision • Access control • Distribution iv. Development kit installation procedure for: • Java Development Kit (JDK) • Software Development Kit (SDK) v. Development stack installation procedure for: • Linux, Apache, MySQL, PHP (LAMP) • Windows, Apache, MySQL, PHP (WAMP) • Text-based application vi. Database administration such as: • Database access • Data manipulation • Database	v. Install development stack vi. Install local server	iii. Committed when setup local environment iv. Analytical thinking when setup local environment v. Follow company's installation guideline <u>Safety:</u> i. Adhere to workplace ergonomics practice			(SCM) explained iv. Development kit installation procedure explained v. Development stack installation procedure explained vi. Database administration explained vii. Integrated Development Environment (IDE) installation and functionality confirmed viii. Software Development Kit (SDK) installation and functionality confirmed ix. Development stack installation and functionality confirmed x. Local server installation and functionality confirmed xi. Local database server installation and

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	maintenance					functionality confirmed xii. Source Code Management (SCM) software installation and functionality confirmed
3. Implement application prototype mock up flow	i. Types of scripting language such as: • Cascading Style Sheets (CSS) • Hyper Text Markup Language (HTML) • JavaScript • PHP: Hypertext PreProcessor (PHP) • Structured Query Language (SQL) ii. Software Development Life Cycle (SDLC) such as: • Agile development • Scrum development iii. Software architectural pattern such as: • Model View Controller (MVC) • Hierarchical Model View Controller (HMVC) • Model View	i. Interpret Software Requirement Specification (SRS) document ii. Interpret user interface and user experience flow iii. Select programming language iv. Write prototype module codes	Attitude: i. Proactive when implementing application prototype mock up flow ii. Resourceful when implementing application prototype mock up flow iii. Committed when implementing application prototype mock up flow iv. Analytical thinking when implementing application prototype mock up flow v. Meticulous	Related Knowledge 22 Related Skills 32	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Types of scripting language listed ii. Definition of Software Development Life Cycle (SDLC) explained iii. Software architectural pattern defined iv. Programming language model explained v. Definition of user interface and user experience explained vii. Function of prototype mock up explained viii. Prototype module codes expected functionality

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	Presenter (MVP) • Model View Adapter (MVA) iv. Programming language model such as: • Object Oriented Programming (OOP) • Structured programming • Procedural programming v. Definition of user interface and user experience vi. Prototype mock up		when implementing application prototype mock up flow vi. Follow company's coding guideline *Safety:* i. Adhere to workplace ergonomics practice			confirmed
4. Conduct user interface and user experience functionality test	i. Fundamentals of user interface (UI) ii. Fundamentals of user experience (UX) iii. Application prototype functionality test procedure iv. Types of test such as: • Unit testing • Usability testing	i. Run application prototype ii. Test user interface function iii. Compare user interface to comply with user experience flow	Attitude: i. Proactive when conducting user interface and user experience functionality test ii. Resourceful when conducting user interface and user experience functionality test	Related Knowledge 18 Related Skills 27	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Fundamentals of user interface (UI) explained ii. Fundamentals of user experience (UX) explained iii. Application prototype functionality test procedure explained iv. User interface conformed to comply with user experience flow

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
			iii. Committed when conducting user interface and user experience functionality test iv. Analytical thinking when conducting user interface and user experience functionality test v. Meticulous when conducting user interface and user experience functionality test vi. Ensure application prototype to meet SRS <u>*Safety:*</u> i. Adhere to workplace ergonomics practice			

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
5. Commit prototype source code	i. Maintenance of source code such as: • SCM • Version control ii. Source code distribution method such as: • Pull • Push • Check out • Commit • Update iii. Format of work progress report	i. Determine source code repository destination ii. Upload prototype source code to source code repository iii. Update work progress report iv. Report prototype source code submission to superior	<u>Attitude:</u> i. Proactive when committing prototype source code ii. Resourceful when committing prototype source code iii. Committed when committing prototype source code iv. Meticulous when committing prototype source code <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 4 <u>Related Skills</u> 5	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Function of source code maintenance explained ii. Source code distribution method listed iii. Prototype source code transferred to source code repository

Employability Skills

Core Abilities	Social Skills
01.01 Identify and gather information. 01.02 Document information procedures or processes. 01.03 Utilize basic IT applications. 02.01 Interpret and follow manuals, instructions and SOP's. 02.03 Communicate clearly. 02.04 Prepare brief reports and checklist using standard forms. 02.05 Read/Interpret flowcharts and pictorial information. 03.02 Demonstrate integrity and apply practical practices. 03.03 Accept responsibility for own work and work area. 03.04 Seek and act constructively upon feedback about work performance. 03.06 Respond appropriately to people and situations. 03.07 Resolve interpersonal conflicts. 06.01 Understand systems. 06.02 Comply with and follow chain of command. 06.03 Identify and highlight problems. 06.04 Adapt competencies to new situation systems. 01.04 Analyze information. 01.05 Utilize the Internet to locate and gather information. 01.06 Utilize word processor to process information. 02.07 Utilize Local Area Network (LAN)/Intranet to exchange information. 02.08 Prepare pictorial and graphic information. 03.08 Develop and maintain a cooperation within work group. 04.01 Organize own work activities. 04.02 Set and revise own objectives and goals. 04.03 Organize and maintain own workplace. 04.04 Apply problem solving strategies. 04.05 Demonstrate initiative and flexibility. 06.05 Analyse technical systems. 06.06 Monitor and correct performance of systems. 01.07 Utilize database applications to locate and process information. 01.08 Utilize spreadsheets applications to locate and process information. 01.10 Apply a variety of mathematical techniques. 01.11 Apply thinking skills and creativity. 02.09 Prepare flowcharts.	1. Communication skills 2. Conceptual skills 3. Interpersonal skills 4. Learning skills 5. Leadership skills 6. Multitasking and prioritising 7. Self-discipline 8. Teamwork

02.10 Prepare reports and instructions. 02.11 Convey information and ideas to people. 03.15 Liaise to achieve identified outcomes. 05.01 Implement project/work plans. 05.02 Inspect and monitor work done and/or in progress.	

Tools, Equipment and Materials (TEM)

ITEMS	RATIO (TEM : Trainees)
1. Computer set	1:1
2. Internet connection	As required
3. Source Code Management (SCM) software	1:1
4. IDE software	1:1
5. Software Development Kit (SDK)	1:1
6. Database Management System (DBMS)	1:1
7. Word processing software	1:1
8. Computer with server role	1:25
9. Stationeries	As required

Reference

REFERENCES

1. Todd Zaki Warfel (2009), Prototyping: A Practitioner's Guide, Louis Rosenfeld Media, LLC, ISBN I-933820-21-7
2. Steve McConnell (2004), Code Complete: A Practical Handbook of Software Construction, (2nd Edition),Microsoft Press, ISBN: 978-0-7356-1967-8
3. Michael L. Scot (2009) , Programming Language Pragmatics, Third Edition, Morgan Kaufmann, ISBN-13: 978-0123745149
4. Benjamin C. Pierce (2002), Types and Programming Languages (1st edition), The MIT Press, ISBN-13: 978-0262162098
5. Simon Marlow (2013), Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming (1st Edition), O'Reilly Media, ISBN-13: 978-1449335946
6. Scott Meyers (2005), Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition), Addison-Wesley Professional, ISBN-13: 978-0321334879
7. Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins (2009), Linux in a Nutshell (6th Edition), O'Reilly Media, ISBN-13: 978-0596154486

CURRICULUM of COMPETENCY UNIT (CoCU)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**						
SUB SECTOR	**SOLUTION DEVELOPMENT**						
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**		
NOSS TITLE	**APPLICATION DEVELOPMENT**						
COMPETENCY UNIT TITLE	**APPLICATION MODULE DEVELOPMENT**						
LEARNING OUTCOME	The person who is competent in this CU shall be able to develop an application that functions, deployment ready, error free and as per SRS. Upon completion of this CU trainee shall be able to: • Interpret application module development requirement • Setup local environment • Plan module expected behaviour • Write module code • Commit module source code						
PRE-REQUISITE (if applicable)	Nil						
COMPETENCY UNIT ID	IT-010-3:2016 –C02	**LEVEL**	3	**TRAINING DURATION**	540	**SKILL CREDIT**	54

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
1. Interpret application module development requirement	i. Types of functional specification document such as: • Functional Design Specification (FDS) • Software Design Document (SDD) ii. Function of Software Requirement Specification (SRS) iii. Introduction to scripting	i. Obtain project brief ii. Identify module function and specification iii. Identify development timeline iv. Identify task assignation v. Check application flow vi. Check application mock up design vii. Check third party	Attitude: i. Proactive when interpreting application module development requirement ii. Resourceful when interpreting application module	Related Knowledge 22 Related Skills 32	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Function of Software Requirement Specification (SRS) explained ii. Types of scripting language listed iii. Function of Integrated Development

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	language iv. Introduction to Integrated Development Environment (IDE) v. Introduction to Test Driven Development (TDD) vi. Project brief content such as: • Development timeline • Modules number • Task assignation vii. Application mock up viii. Third party component such as: • Payment gateway • Security certificate • Single Sign On (SSO) ix. Awareness of End User License Agreement (EULA) x. Software licensing such as: • Proprietary • Open source xi. Unified Modelling Language (UML)	component requirement	development requirement iii. Committed when interpreting application module development requirement iv. Analytical thinking when interpreting application module development requirement v. Adhere to End User License Agreement (EULA) <u>*Safety:*</u> i. Adhere to workplace ergonomics practice			Environment (IDE) explained iv. Function of Test Driven Development (TDD) explained v. Types of project brief content listed vi. Function of application mock up explained vii. Third party component listed viii. Purpose of End User License Agreement (EULA) explained ix. Types of software licensing listed x. Application prototype development requirement confirmed xi. Workplace ergonomic practice explained

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
2. Setup local environment	i. Introduction to development environment ii. Introduction to local server such as: • Local server access configuration • Installation procedure iii. Introduction to Source Code Management (SCM) such as: • Source code repository • Branching • Revision • Access control • Distribution iv. Development kit installation procedure for: • Java Development Kit (JDK) • Software Development Kit (SDK) v. Development stack installation procedure for: • Linux, Apache, MySQL, PHP (LAMP) • Windows, Apache, MySQL, PHP	i. Interpret local environment requirement ii. Interpret database structure iii. Select programming language iv. Check local server access configuration v. Check development server access configuration vi. Check Source Code Management (SCM) access vii. Install Integrated Development Environment (IDE) viii. Install Software Development Kit (SDK) ix. Install development stack (i.e. WAMP, LAMP) x. Install local server xi. Install local database server xii. Install Source Code Management (SCM) software	Attitude: i. Proactive when setup local environment ii. Resourceful when setup local environment iii. Committed when setup local environment iv. Analytical thinking when setup local environment v. Follow company's installation guideline *Safety:* i. Adhere to workplace ergonomics practice	Related Knowledge 43 Related Skills 65	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Definition of development environment explained ii. Definition of local server explained iii. Definition of Source Code Management (SCM) explained iv. Development kit installation procedure explained v. Development kit installation procedure explained vi. Development stack installation procedure explained vii. Database administration explained viii. Integrated Development Environment (IDE) installation and functionality confirmed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	(WAMP) • Text-based application vi. Database administration such as: • Database access • Data manipulation • Database maintenance					ix. Software Development Kit (SDK) installation and functionality confirmed x. Development stack installation and functionality confirmed xi. Local server installation and functionality confirmed xii. Local database server installation and functionality confirmed xiii. Source Code Management (SCM) software installation and functionality confirmed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
3. Plan module expected behaviour	i. Definition of module expected behaviour ii. Function of module test script iii. Test Driven Development (TDD) method such as: • Pseudo code • Data flow diagram • Module process flow • Decision table	i. List out module expected scenarios ii. List out module expected input and output iii. Write module test script	Attitude: i. Proactive when planning module expected behaviour ii. Resourceful when planning module expected behaviour iii. Committed when planning module expected behaviour iv. Analytical thinking when planning module expected behaviour *Safety:* i. Adhere to workplace ergonomics practice	Related Knowledge 54 Related Skills 81	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Definition of module expected behaviour explained ii. Function of module test script explained iii. Test Driven Development (TDD) method listed iv. Module expected input and output listed v. Module test script produced and expected functionality confirmed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
4. Write module code	i. Types of scripting / programming language such as: • Cascading Style Sheets (CSS) • Hyper Text Markup Language (HTML) • JavaScript • Java • Objective C • PHP: Hypertext PreProcessor (PHP) • Structured Query Language (SQL) ii. Software architectural pattern such as: • Model View Controller (MVC) • Hierarchical Model View Controller (HMVC) • Model View Presenter (MVP) • Model View Adapter (MVA) iii. Programming language model such as: • Object oriented programming (OOP) • Structured programming	i. Check naming convention ii. Write instruction code to perform module function iii. Associate database connection with source code in local server iv. Compose Structured Query Language (SQL) statement v. Develop coding according to test script vi. Execute unit testing vii. Verify module output viii. Debug module code	<u>Attitude:</u> i. Proactive when writing module code ii. Resourceful when writing module code iii. Committed when writing module code iv. Analytical thinking when writing module code v. Meticulous when writing module code vi. Follow company's coding guideline <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 86 <u>Related Skills</u> 130	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Types of scripting language listed ii. Software architectural pattern listed iii. Programming language model listed iv. Types of functional specification document listed v. Definition of user interface and user experience explained vi. Function of application mock up explained vii. Definition of unit testing explained viii. Debugging procedure explained ix. Instruction code produced and functionality checked x. Database connected

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	• Procedural programming iv. Types of functional specification document such as: • Functional Design Specification (FDS) • Software Design Document (SDD) v. Definition of user interface and user experience vi. Application mock up vii. Unit testing viii. Debugging procedure					with source code in local server xi. Structured Query Language (SQL) statement produced and expected functionality confirmed xii. Coding produced and expected functionality confirmed xiii. Module code error fixed and expected functionality confirmed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
5. Commit module source code	i. Maintenance of source code such as: • SCM • Version control ii. Source code distribution method such as: • Pull • Push • Check out • Commit • Update iii. Format of work progress report	i. Determine source code repository destination ii. Upload module source code to source code repository iii. Update work progress report iv. Report module source code submission to superior	<u>Attitude:</u> i. Proactive when committing source code ii. Resourceful when committing source code iii. Committed when committing source code iv. Meticulous when committing source code <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 11 <u>Related Skills</u> 16	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Function of source code maintenance explained ii. Source code distribution method listed iii. Module source code transferred to source code repository

Employability Skills

Core Abilities	Social Skills
01.01 Identify and gather information. 01.02 Document information procedures or processes. 01.03 Utilize basic IT applications. 02.01 Interpret and follow manuals, instructions and SOP's. 02.03 Communicate clearly. 02.04 Prepare brief reports and checklist using standard forms. 02.05 Read/Interpret flowcharts and pictorial information. 03.02 Demonstrate integrity and apply practical practices. 03.03 Accept responsibility for own work and work area. 03.04 Seek and act constructively upon feedback about work performance. 03.06 Respond appropriately to people and situations. 03.07 Resolve interpersonal conflicts. 06.01 Understand systems. 06.02 Comply with and follow chain of command. 06.03 Identify and highlight problems. 06.04 Adapt competencies to new situation systems. 01.04 Analyze information. 01.05 Utilize the Internet to locate and gather information. 01.06 Utilize word processor to process information. 02.07 Utilize Local Area Network (LAN)/Intranet to exchange information. 02.08 Prepare pictorial and graphic information. 03.08 Develop and maintain a cooperation within work group. 04.01 Organize own work activities. 04.02 Set and revise own objectives and goals. 04.03 Organize and maintain own workplace. 04.04 Apply problem solving strategies. 04.05 Demonstrate initiative and flexibility. 06.05 Analyse technical systems. 06.06 Monitor and correct performance of systems. 01.07 Utilize database applications to locate and process information. 01.08 Utilize spreadsheets applications to locate and process information. 01.10 Apply a variety of mathematical techniques. 01.11 Apply thinking skills and creativity. 02.09 Prepare flowcharts.	1. Communication skills 2. Conceptual skills 3. Interpersonal skills 4. Learning skills 5. Leadership skills 6. Multitasking and prioritising 7. Self-discipline 8. Teamwork

02.10 Prepare reports and instructions. 02.11 Convey information and ideas to people. 03.15 Liaise to achieve identified outcomes. 05.01 Implement project/work plans. 05.02 Inspect and monitor work done and/or in progress.	

Tools, Equipment and Materials (TEM)

ITEMS	RATIO (TEM : Trainees)
1. Computer set	1:1
2. Internet connection	As required
3. Source Code Management (SCM) software	1:1
4. IDE software	1:1
5. Software Development Kit (SDK)	1:1
6. Database Management System (DBMS)	1:1
7. Word processing software	1:1
8. Computer with server role	1:25
9. Stationeries	As required

Reference

REFERENCES

1. Keith Cooper, Linda Torczon(2011), Engineering a Compiler (2nd Edition), Morgan Kaufmann, ISBN: 978-0120884780
2. Todd Zaki Warfel (2009), Prototyping: A Practitioner's Guide, Louis Rosenfeld Media, LLC, ISBN I-933820-21-7
3. Steve McConnell (2004), Code Complete: A Practical Handbook of Software Construction, (2nd Edition),Microsoft Press, ISBN: 978-0-7356-1967-8
4. Michael L. Scot (2009) , Programming Language Pragmatics, Third Edition, Morgan Kaufmann, ISBN-13: 978-0123745149
5. Simon Marlow (2013), Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming (1st Edition), O'Reilly Media, ISBN-13: 978-1449335946
6. Scott Meyers (2005), Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition), Addison-Wesley Professional, ISBN-13: 978-0321334879
7. Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins (2009), Linux in a Nutshell (6th Edition), O'Reilly Media, ISBN-13: 978-0596154486

CURRICULUM of COMPETENCY UNIT (CoCU)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**					
SUB SECTOR	**SOLUTION DEVELOPMENT**					
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**	
NOSS TITLE	**APPLICATION DEVELOPMENT**					
COMPETENCY UNIT TITLE	**APPLICATION MODULE INTEGRATION**					
LEARNING OUTCOME	The person who is competent in this CU shall be able to ensure that all modules are integrated correctly, functional and ready for system testing. Upon completion of this CU trainee shall be able to: • Interpret module integration requirement • Perform modules integration • Test module integration code • Commit module integration code					
PRE-REQUISITE (if applicable)	Nil					
COMPETENCY UNIT ID	IT-010-3:2016 –C03	**LEVEL**	3	**TRAINING DURATION**	270	**SKILL CREDIT** 27

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
1. Interpret module integration requirement	i. Introduction to module integration ii. Purpose of module integration iii. Integration requirement such as: • Module to be integrated • Level of integration • Integration point iv. Enterprise Application Integrator (EAI) software	i. Interpret project brief ii. Identify module to be integrated iii. Identify level of integration iv. Identify module integration point v. Determine source code repository location vi. Retrieve module source code	Attitude: i. Proactive when interpreting module integration requirement ii. Resourceful when interpreting module integration requirement iii. Committed	Related Knowledge 11 Related Skills 16	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Module integration explained ii. Purpose of module integration explained iii. Types of module integration requirement listed iv. Enterprise

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
			when interpreting module integration requirement iv. Analytical thinking when interpreting module integration requirement v. Adhere to End User License Agreement (EULA) <u>*Safety:*</u> i. Adhere to workplace ergonomics practice			Application Integrator (EAI) software explained v. Module integration requirement confirmed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
2. Perform modules integration	i. Database architecture ii. Common data format method iii. Data Manipulation Language (DML) iv. Data integration frequency v. Module integration process vi. Integration method such as: • Shared folder • Cron job • Intermediate database • Trigger vii. Data extraction	i. Associate related module with database table ii. Establish shared folder for data extraction iii. Write integration code between modules iv. Execute integration between modules	<u>Attitude:</u> i. Proactive when performing modules integration ii. Resourceful when performing modules integration iii. Committed when performing modules integration iv. Analytical thinking when performing modules integration v. Adhere to End User License Agreement (EULA) <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 54 <u>Related Skills</u> 81	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Database architecture explained ii. Data Manipulation Language (DML) explained iii. Data integration frequency explained iv. Module integration process explained v. Integration method listed vi. Data extraction explained vii. Related module connected with database table v. Shared folder for data extraction created vi. Integration code between modules produced and expected functionality confirmed vii. Modules

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
						integration confirmed
3. Test module integration code	i. Types of unit interface such as: • Terminal • Input and output device ii. Types of testing such as: • Functional test • File test • Data integrity test iii. Data archiving procedure	i. Identify unit interface ii. Run module integration code iii. Verify module integration output iv. Debug module integration code v. Archive application data	<u>Attitude:</u> i. Proactive when testing module integration code ii. Resourceful when testing module integration code iii. Committed when testing module integration code iv. Analytical thinking when testing module integration code <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 38 <u>Related Skills</u> 57	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Types of unit interface listed ii. Types of testing listed iii. Data archiving procedure explained iv. Modules integration functionality confirmed v. Module integration code fixing result confirmed vi. Application data archived

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
4. Commit module integration code	i. Maintenance of source code such as: • SCM • Version control ii. Source code distribution method such as: • Pull • Push • Check out • Commit • Update iii. Format of work progress report	i. Upload integrated module source code to source code repository ii. Update work progress report iii. Report integrated module source code submission to superior	Attitude: i. Proactive when committing module integration code ii. Resourceful when committing module integration code iii. Committed when committing module integration code iv. Meticulous when committing module integration code *Safety:* i. Adhere to workplace ergonomics practice	Related Knowledge 5 Related Skills 8	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Function of source code maintenance explained ii. Source code distribution method listed iii. Integrated module source code source code transferred to source code repository iv. Updated work progress report submitted to superior

Employability Skills

Core Abilities	Social Skills
01.01 Identify and gather information. 01.02 Document information procedures or processes. 01.03 Utilize basic IT applications. 02.01 Interpret and follow manuals, instructions and SOP's. 02.03 Communicate clearly. 02.04 Prepare brief reports and checklist using standard forms. 02.05 Read/Interpret flowcharts and pictorial information. 03.02 Demonstrate integrity and apply practical practices. 03.03 Accept responsibility for own work and work area. 03.04 Seek and act constructively upon feedback about work performance. 03.06 Respond appropriately to people and situations. 03.07 Resolve interpersonal conflicts. 06.01 Understand systems. 06.02 Comply with and follow chain of command. 06.03 Identify and highlight problems. 06.04 Adapt competencies to new situation systems. 01.04 Analyze information. 01.05 Utilize the Internet to locate and gather information. 01.06 Utilize word processor to process information. 02.07 Utilize Local Area Network (LAN)/Intranet to exchange information. 02.08 Prepare pictorial and graphic information. 03.08 Develop and maintain a cooperation within work group. 04.01 Organize own work activities. 04.02 Set and revise own objectives and goals. 04.03 Organize and maintain own workplace. 04.04 Apply problem solving strategies. 04.05 Demonstrate initiative and flexibility. 06.05 Analyse technical systems. 06.06 Monitor and correct performance of systems. 01.07 Utilize database applications to locate and process information. 01.08 Utilize spreadsheets applications to locate and process information. 01.10 Apply a variety of mathematical techniques. 01.11 Apply thinking skills and creativity. 02.09 Prepare flowcharts.	1. Communication skills 2. Conceptual skills 3. Interpersonal skills 4. Learning skills 5. Leadership skills 6. Multitasking and prioritising 7. Self-discipline 8. Teamwork

02.10 Prepare reports and instructions. 02.11 Convey information and ideas to people. 03.15 Liaise to achieve identified outcomes. 05.01 Implement project/work plans. 05.02 Inspect and monitor work done and/or in progress.	

Tools, Equipment and Materials (TEM)

ITEMS	RATIO (TEM : Trainees)
1. Computer set	1:1
2. Internet connection	As required
3. Source Code Management (SCM) software	1:1
4. IDE software	1:1
5. Software Development Kit (SDK)	1:1
6. Database Management System (DBMS)	1:1
7. Word processing software	1:1
8. Computer with server role	1:25
9. Stationeries	As required

Reference

REFERENCES

1. W. F. Gielingh, W. J. de Bruijn, J. R. Halbert (1991), Computer Integrated Manufacturing: Implementation Levels for Semantic Integration of Open System CIM Modules, Springer London, ISBN 978-3-540-19695-2 / Online ISBN: 978-1-4471-3257-8
2. Keith Cooper, Linda Torczon(2011), Engineering a Compiler (2nd Edition), Morgan Kaufmann, ISBN: 978-0120884780
3. Steve McConnell (2004), Code Complete: A Practical Handbook of Software Construction, (2nd Edition),Microsoft Press, ISBN: 978-0-7356-1967-8
4. Michael L. Scot (2009) , Programming Language Pragmatics, Third Edition, Morgan Kaufmann, ISBN-13: 978-0123745149
5. Benjamin C. Pierce (2002), Types and Programming Languages (1st edition), The MIT Press, ISBN-13: 978-0262162098
6. Simon Marlow (2013), Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming (1st Edition), O'Reilly Media, ISBN-13: 978-1449335946
7. Scott Meyers (2005), Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition), Addison-Wesley Professional, ISBN-13: 978-0321334879

CURRICULUM of COMPETENCY UNIT (CoCU)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**								
SUB SECTOR	**SOLUTION DEVELOPMENT**								
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**				
NOSS TITLE	**APPLICATION DEVELOPMENT**								
COMPETENCY UNIT TITLE	**DEVELOPMENT ENVIRONMENT DEPLOYMENT**								
LEARNING OUTCOME	The person who is competent in this CU shall be able to establish an environment that is as identical to the production environment which is reliable and error free. Upon completion of this CU trainee shall be able to: • Deploy source code to development server • Perform data population to development server • Perform integration system testing								
PRE-REQUISITE (if applicable)	Nil								
COMPETENCY UNIT ID	IT-010-3:2016 –C04	**LEVEL**	3	**TRAINING DURATION**	360	**SKILL CREDIT**	36		

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
1. Deploy source code to development server	i. Maintenance of source code such as: • SCM • Version control ii. Source code distribution method such as: • Pull • Push • Check out • Commit • Update iii. Function of development server	i. Compile source code ii. Upload source code to development server iii. Associate development database connection iv. Associate server access	Attitude: i. Proactive when deploying source code to development server ii. Resourceful when deploying source code to development server iii. Committed when deploying source code to	Related Knowledge 43 Related Skills 65	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Purpose of source code maintenance explained ii. Source code distribution method listed iii. Function of development server explained iv. Function of database explained

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	iv. Function of database v. Module integration vi. Database administration such as: • Database access • Data manipulation • Database maintenance vii. Server administration such as: • Server access • Server security • Server maintenance		development server iv. Analytical thinking when deploying source code to development server <u>*Safety:*</u> i. Adhere to workplace ergonomics practice			v. Module integration explained vi. Database administration explained vii. Server administration explained viii. Source code transferred to development server ix. Data from database retrieved x. Server access established xi. Workplace ergonomics practice explained

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
2. Perform data population to development server	i. Types of data migration such as: • Data extraction • Data loading • Data massage / data cleaning ii. Types of testing such as: • Functional test • File test • Data integrity test iii. Types of functional specification document such as: • Functional Design Specification (FDS) • Software Design Document (SDD) iv. Structured Query Language (SQL)	i. Populate database with required data set ii. Test application behaviour and data flow with functional specification iii. Check application behaviour and data flow comply with functional specification	<u>Attitude:</u> i. Proactive when performing data population to development server ii. Resourceful when performing data population to development server iii. Committed when performing data population to development server iv. Analytical thinking when performing data population to development server <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 43 <u>Related Skills</u> 65	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Types of data migration listed ii. Types of testing explained iii. Types of functional specification document listed iv. Dummy or actual data keyed in database v. Application behaviour and data flow functionality confirmed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
3. Perform integrity system testing	i. Types of third party component such as: • Payment gateway • Software Security License (SSL) ii. Third party component integration procedure iii. Function of Enterprise Application Integrator (EAI) software iv. Type of Request For Comments (RFC) protocol such as: • Open Authorization (OAuth) • File Transfer Protocol (FTP) • HyperText Transfer Protocol (HTTP) • Secure Shell Protocol (SSH)	i. Install third party component ii. Integrate third party component with application module iii. Test third party component compatibility with application module iv. Check third party component compatibility	Attitude: i. Proactive when performing integrity system testing ii. Resourceful when performing integrity system testing iii. Committed when performing integrity system testing iv. Analytical thinking when performing integrity system testing *Safety:* i. Adhere to workplace ergonomics practice	Related Knowledge 58 Related Skills 86	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Types of third party component listed ii. Third party component integration procedure explained iii. Function of Enterprise Application Integrator (EAI) software explained iv. Type of Request For Comments (RFC) protocol explained v. Third party component installation and expected functionality confirmed vi. Third party component integration compatibility confirmed

Employability Skills

Core Abilities	Social Skills
01.01 Identify and gather information. 01.02 Document information procedures or processes. 01.03 Utilize basic IT applications. 02.01 Interpret and follow manuals, instructions and SOP's. 02.03 Communicate clearly. 02.04 Prepare brief reports and checklist using standard forms. 02.05 Read/Interpret flowcharts and pictorial information. 03.02 Demonstrate integrity and apply practical practices. 03.03 Accept responsibility for own work and work area. 03.04 Seek and act constructively upon feedback about work performance. 03.06 Respond appropriately to people and situations. 03.07 Resolve interpersonal conflicts. 06.01 Understand systems. 06.02 Comply with and follow chain of command. 06.03 Identify and highlight problems. 06.04 Adapt competencies to new situation systems. 01.04 Analyze information. 01.05 Utilize the Internet to locate and gather information. 01.06 Utilize word processor to process information. 02.07 Utilize Local Area Network (LAN)/Intranet to exchange information. 02.08 Prepare pictorial and graphic information. 03.08 Develop and maintain a cooperation within work group. 04.01 Organize own work activities. 04.02 Set and revise own objectives and goals. 04.03 Organize and maintain own workplace. 04.04 Apply problem solving strategies. 04.05 Demonstrate initiative and flexibility. 06.05 Analyse technical systems. 06.06 Monitor and correct performance of systems. 01.07 Utilize database applications to locate and process information. 01.08 Utilize spreadsheets applications to locate and process information. 01.10 Apply a variety of mathematical techniques. 01.11 Apply thinking skills and creativity. 02.09 Prepare flowcharts.	1. Communication skills 2. Conceptual skills 3. Interpersonal skills 4. Learning skills 5. Leadership skills 6. Multitasking and prioritising 7. Self-discipline 8. Teamwork

02.10 Prepare reports and instructions. 02.11 Convey information and ideas to people. 03.15 Liaise to achieve identified outcomes. 05.01 Implement project/work plans. 05.02 Inspect and monitor work done and/or in progress.	

Tools, Equipment and Materials (TEM)

ITEMS	RATIO (TEM : Trainees)
1. Computer set	1:1
2. Internet connection	As required
3. Source Code Management (SCM) software	1:1
4. IDE software	1:1
5. Software Development Kit (SDK)	1:1
6. Database Management System (DBMS)	1:1
7. Word processing software	1:1
8. Computer with server role	1:25
9. Stationeries	As required

Reference

REFERENCES

1. Craig S. Mullins, Database Administration: The Complete Guide to DBA Practices and Procedures (2nd Edition), ISBN-10: 0321822943, ISBN-13: 978-0321822949
2. W. F. Gielingh, W. J. de Bruijn, J. R. Halbert (1991), Computer Integrated Manufacturing: Implementation Levels for Semantic Integration of Open System CIM Modules, Springer London, ISBN 978-3-540-19695-2 / Online ISBN: 978-1-4471-3257-8
3. Keith Cooper, Linda Torczon(2011), Engineering a Compiler (2nd Edition), Morgan Kaufmann, ISBN: 978-0120884780
4. Steve McConnell (2004), Code Complete: A Practical Handbook of Software Construction, (2nd Edition),Microsoft Press, ISBN: 978-0-7356-1967-8
5. Michael L. Scot (2009) , Programming Language Pragmatics, Third Edition, Morgan Kaufmann, ISBN-13: 978-0123745149
6. Simon Marlow (2013), Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming (1st Edition), O'Reilly Media, ISBN-13: 978-1449335946

CURRICULUM of COMPETENCY UNIT (CoCU)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**					
SUB SECTOR	**SOLUTION DEVELOPMENT**					
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**	
NOSS TITLE	**APPLICATION DEVELOPMENT**					
COMPETENCY UNIT TITLE	**APPLICATION BUG FIXING**					
LEARNING OUTCOME	The person who is competent in this CU shall be able to ensure that the application developed is error free. Upon completion of this CU trainee shall be able to: • Interpret bug report • Simulate bug/error scenario • Debug application code • Commit fixed source code					
PRE-REQUISITE (if applicable)	Nil					
COMPETENCY UNIT ID	IT-010-3:2016 –C05	**LEVEL**	3	**TRAINING DURATION**	360	**SKILL CREDIT** 36

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
1. Interpret bug report	i. Definition of bug ii. Bug tracking software iii. Purpose of knowledgebase iv. End User License Agreement (EULA) v. Types of error classification such as: • Application based error • Server based error • User based error vi. Cause of error such as:	i. Obtain bug report ii. Identify types of error iii. Identify error triggered environment iv. Identify cause of error v. Identify severity of error	<u>Attitude:</u> i. Proactive when interpreting bug report ii. Resourceful when interpreting bug report iii. Committed when interpreting bug report iv. Analytical	<u>Related Knowledge</u> 7 <u>Related Skills</u> 11	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Definition of bug explained ii. Purpose of bug tracking software explained iii. Purpose of knowledgebase explained iv. End User License Agreement (EULA)

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	• Server down time • Unexpected user action • Unexpected application behaviour vii. Level of qualified severity such as: • Critical • High • Low • Blocker/show stopper • Trivial		thinking when interpreting bug report v. Adhere to End User License Agreement (EULA) <u>*Safety:*</u> i. Adhere to workplace ergonomics practice			explained v. Types of error classification listed vi. Cause of error listed vii. Level of qualified severity listed viii. Types of error confirmed ix. Error triggered environment confirmed x. Severity of error confirmed
2. Simulate bug/error scenario	i. Error reproduce procedure ii. Types of error triggered environment such as: • Hardware configuration • Software configuration • Network configuration iii. Types of unexpected user action such as: • Unexpected data input • Multiple user attempt • Unexpected	i. Reproduce application error scenario ii. Check application error occurrence iii. Identify impact of error	<u>Attitude:</u> i. Proactive when simulating bug/error scenario ii. Resourceful when simulating bug/error scenario iii. Committed when simulating bug/error scenario iv. Analytical thinking when simulating	<u>Related Knowledge</u> 65 <u>Related Skills</u> 97	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Error reproduce procedure explained ii. Types of error triggered environment listed iii. Types of unexpected user action listed iv. Types of error impact listed v. Bug/error scenario occurrence confirmed vi. Impact of error

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	navigation pattern iv. Types of error impact such as: • Data lost • System crash • System unresponsive • System malfunction		bug/error scenario <u>*Safety:*</u> i. Adhere to workplace ergonomics practice			confirmed
3. Debug application code	i. Debugging procedure ii. Purpose of custom error page iii. Types of custom error page such as: • Under maintenance • File not found • Network lost iv. Debugging tools v. Turnaround time vi. Types of testing such as: • Functional test • File test • Data integrity test	i. Retrieve application source code from source code repository ii. Identify affected source code iii. Refer knowledgebase iv. Isolate affected source code v. Apply custom error page during fixing period vi. Fix affected source code within turnaround time vii. Reproduce initial error scenario viii. Check no bug occurrence ix. Check bug fixed	<u>Attitude:</u> i. Proactive when debugging application code ii. Resourceful when debugging application code iii. Committed when debugging application code iv. Analytical thinking when debugging application code v. Follow fixing turnaround time <u>*Safety:*</u> i. Adhere to workplace	<u>Related Knowledge</u> 65 <u>Related Skills</u> 97	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Debugging procedure explained ii. Purpose of custom error page explained iii. Types of custom error listed iv. Debugging tools listed v. Turnaround time explained vi. Types of testing listed vii. Custom error page developed viii. Affected source code fixed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
			ergonomics practice			
4. Commit fixed source code	i. Maintenance of source code such as: • SCM • Version control ii. Source code distribution method such as: • Pull • Push • Check out • Commit • Update iii. Format of work progress report	i. Publish fixed source code to source code repository ii. Update knowledgebase iii. Update work progress report iv. Report fixed source code submission to superior	Attitude: i. Proactive when committing fixed source code ii. Resourceful when committing fixed source code iii. Committed when committing fixed source code iv. Analytical thinking when committing fixed source code *Safety:* i. Adhere to workplace ergonomics practice	Related Knowledge 7 Related Skills 11	Related Knowledge Lecture Related Skills Demonstration & Observation	i. Function of source code maintenance explained ii. Source code distribution method listed iii. Fixed source code transferred to source code repository iv. Updated work progress report submitted to superior

Employability Skills

Core Abilities	Social Skills
01.01 Identify and gather information. 01.02 Document information procedures or processes. 01.03 Utilize basic IT applications. 02.01 Interpret and follow manuals, instructions and SOP's. 02.03 Communicate clearly. 02.04 Prepare brief reports and checklist using standard forms. 02.05 Read/Interpret flowcharts and pictorial information. 03.02 Demonstrate integrity and apply practical practices. 03.03 Accept responsibility for own work and work area. 03.04 Seek and act constructively upon feedback about work performance. 03.06 Respond appropriately to people and situations. 03.07 Resolve interpersonal conflicts. 06.01 Understand systems. 06.02 Comply with and follow chain of command. 06.03 Identify and highlight problems. 06.04 Adapt competencies to new situation systems. 01.04 Analyze information. 01.05 Utilize the Internet to locate and gather information. 01.06 Utilize word processor to process information. 02.07 Utilize Local Area Network (LAN)/Intranet to exchange information. 02.08 Prepare pictorial and graphic information. 03.08 Develop and maintain a cooperation within work group. 04.01 Organize own work activities. 04.02 Set and revise own objectives and goals. 04.03 Organize and maintain own workplace. 04.04 Apply problem solving strategies. 04.05 Demonstrate initiative and flexibility. 06.05 Analyse technical systems. 06.06 Monitor and correct performance of systems. 01.07 Utilize database applications to locate and process information. 01.08 Utilize spreadsheets applications to locate and process information. 01.10 Apply a variety of mathematical techniques. 01.11 Apply thinking skills and creativity. 02.09 Prepare flowcharts.	1. Communication skills 2. Conceptual skills 3. Interpersonal skills 4. Learning skills 5. Leadership skills 6. Multitasking and prioritising 7. Self-discipline 8. Teamwork

02.10 Prepare reports and instructions. 02.11 Convey information and ideas to people. 03.15 Liaise to achieve identified outcomes. 05.01 Implement project/work plans. 05.02 Inspect and monitor work done and/or in progress.	

Tools, Equipment and Materials (TEM)

ITEMS	RATIO (TEM : Trainees)
1. Computer set	1:1
2. Internet connection	As required
3. Source Code Management (SCM) software	1:1
4. IDE software with debugging features	1:1
5. Software Development Kit (SDK)	1:1
6. Database Management System (DBMS)	1:1
7. Bug tracking software	1:1
8. Word processing software	1:1
9. Computer with server role	1:25
10. Stationeries	As required

Reference

REFERENCES

1. Kent Beck (2002), Test Driven Development: By Example (1st Edition), Addison-Wesley Professional, ISBN 978-0321146533
2. Paul Butcher (2009), Debug It!: Find, Repair, and Prevent Bugs in Your Code (Pragmatic Programmers) (1st Edition),Pragmatic Bookshelf, ISBN 978-1934356289
3. Andrew Hunt, David Thomas (1999), The Pragmatic Programmer: From Journeyman to Master (1st Edition), Addison-Wesley Professional, ISBN 078-5342616224
4. Matthew Linderman, Jason Fried (2004) , Defensive Design for the Web: How to improve error messages, help, forms, and other crisis points(1st Edition), New riders), ISBN 978-0735714106
5. Tobias Klein (2011), A Bug Hunter's Diary: A Guided Tour Through the Wilds of Software Security(1st Edition), No Starch Press ISBN 978-1593273859

CURRICULUM of COMPETENCY UNIT (CoCU)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**					
SUB SECTOR	**SOLUTION DEVELOPMENT**					
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**	
NOSS TITLE	**APPLICATION DEVELOPMENT**					
COMPETENCY UNIT TITLE	**APPLICATION SYSTEM DOCUMENTATION COMPILATION**					
LEARNING OUTCOME	The person who is competent in this CU shall be able to compile accurate information and documents according to application module and store the system documentation in source code repository. Upon completion of this CU trainee shall be able to: • Interpret system documentation requirement • Compile system documentation material • Upload system documentation					
PRE-REQUISITE (if applicable)	Nil					

COMPETENCY UNIT ID	IT-010-3:2016 –C06	**LEVEL**	3	**TRAINING DURATION**	90	**SKILL CREDIT**	9

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
1. Interpret system documentation requirement	i. System documentation requirement ii. Source code documenter software iii. Types of system documentation such as: • Unified Modelling Language (UML) • User manual • Source code manual • Data flow diagram	i. Identify application system documentation requirement ii. Obtain document template as per system documentation requirement iii. Refer company's guideline for application system documentation	<u>Attitude:</u> i. Proactive when interpreting system documentation requirement ii. Resourceful when interpreting system documentation requirement iii. Committed	<u>Related Knowledge</u> 7 <u>Related Skills</u> 11	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. System documentation requirement explained ii. Source code documenter software listed iii. Types of system documentation listed vii. Function of company's

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	• Entity relation diagram (ERD) • End User License Agreement (EULA) iv. Sample of user manual v. Sample of source code manual vi. Company's policy and guideline		when interpreting system documentation requirement iv. Analytical thinking when interpreting system documentation requirement v. Integrity when interpreting system documentation requirement <u>*Safety:*</u> i. Adhere to workplace ergonomics practice			policy and guideline explained iv. System documentation requirement confirmed

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
2. Compile system documentation material	i. Source code documentation comment ii. Types of user manual resource such as: • Screenshot image • Process flow diagram • Related article	i. Write source code documentation comment ii. Generate source code documentation iii. Compile user manual resource iv. Organize user manual material	<u>Attitude:</u> i. Proactive when compiling system documentation material ii. Resourceful when compiling system documentation material iii. Committed when compiling system documentation material iv. Analytical thinking when compiling system documentation material <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 22 <u>Related Skills</u> 32	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Source code documentation syntax explained ii. Types of user manual resource listed iii. Source code syntax produced iv. Source code documentation produced v. User manual resource and user manual material produced

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
3. Upload system documentation	i. Maintenance of source code such as: • SCM • Version control ii. Source code distribution method such as: • Pull • Push • Check out • Commit • Update	i. Determine system documentation storage destination in source code repository ii. Upload system documentation to source code repository iii. Grant system documentation access to technical documentation writer	<u>Attitude:</u> i. Proactive when uploading system documentation ii. Resourceful when uploading system documentation iii. Committed when uploading system documentation iv. Analytical thinking when uploading system documentation <u>*Safety:*</u> i. Adhere to workplace ergonomics practice	<u>Related Knowledge</u> 7 <u>Related Skills</u> 11	<u>Related Knowledge</u> Lecture <u>Related Skills</u> Demonstration & Observation	i. Function of source code maintenance explained ii. Source code distribution method listed iii. System documentation transferred to source code repository

Employability Skills

Core Abilities	Social Skills
01.01 Identify and gather information. 01.02 Document information procedures or processes. 01.03 Utilize basic IT applications. 02.01 Interpret and follow manuals, instructions and SOP's. 02.03 Communicate clearly. 02.04 Prepare brief reports and checklist using standard forms. 02.05 Read/Interpret flowcharts and pictorial information. 03.02 Demonstrate integrity and apply practical practices. 03.03 Accept responsibility for own work and work area. 03.04 Seek and act constructively upon feedback about work performance. 03.06 Respond appropriately to people and situations. 03.07 Resolve interpersonal conflicts. 06.01 Understand systems. 06.02 Comply with and follow chain of command. 06.03 Identify and highlight problems. 06.04 Adapt competencies to new situation systems. 01.04 Analyze information. 01.05 Utilize the Internet to locate and gather information. 01.06 Utilize word processor to process information. 02.07 Utilize Local Area Network (LAN)/Intranet to exchange information. 02.08 Prepare pictorial and graphic information. 03.08 Develop and maintain a cooperation within work group. 04.01 Organize own work activities. 04.02 Set and revise own objectives and goals. 04.03 Organize and maintain own workplace. 04.04 Apply problem solving strategies. 04.05 Demonstrate initiative and flexibility. 06.05 Analyze technical systems. 06.06 Monitor and correct performance of systems. 01.07 Utilize database applications to locate and process information. 01.08 Utilize spreadsheets applications to locate and process information. 01.10 Apply a variety of mathematical techniques. 01.11 Apply thinking skills and creativity.	1. Communication skills 2. Conceptual skills 3. Interpersonal skills 4. Learning skills 5. Leadership skills 6. Multitasking and prioritising 7. Self-discipline 8. Teamwork

02.09 Prepare flowcharts. 02.10 Prepare reports and instructions. 02.11 Convey information and ideas to people. 03.15 Liaise to achieve identified outcomes. 05.01 Implement project/work plans. 05.02 Inspect and monitor work done and/or in progress.	

Tools, Equipment and Materials (TEM)

ITEMS	RATIO (TEM : Trainees)
1. Computer set	1:1
2. Internet connection	As required
3. Source Code Management (SCM) software	1:1
4. IDE software	1:1
5. Software Development Kit (SDK)	1:1
6. Database Management System (DBMS)	1:1
7. Source code documenter software	1:1
8. Word processing software	1:1
9. Computer with server role	1:25
10. Stationeries	As required
11. Sample of user manual	1:1
12. Sample of source code manual	1:1

Reference

REFERENCES

1. Roger S. Pressman, Bruce Maxim (2010) Software Engineering: A Practitioners Approach (6th edition), Mc Graw Hill Education, ISBN 978-0073375977
2. Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Paulo Merson, Robert Nord, Judith Stafford (2010), Documenting Software Architectures: Views and Beyond (2nd Edition), Addison-Wesley Professional, ISBN 978-0321552686
3. Thomas T. Barker (2002), Writing Software Documentation: A Task-Oriented Approach (Part of the Allyn & Bacon Series in Technical Communication) (2nd Edition), Longman, ISBN 978-0321103284
4. Karl Wiegers, Joy Batty (2013), Software Requirements (3rd Edition) (Developer Best Practices), Microsoft Press, ISBN 978-0735679665

CURRICULUM of COMPETENCY UNIT (CoCU)

SECTOR	**INFORMATION & COMMUNICATION TECHNOLOGY (ICT)**					
SUB SECTOR	**SOLUTION DEVELOPMENT**					
JOB AREA	**MOBILE APPLICATION DEVELOPMENT**	**APPLICATION SYSTEM DEVELOPMENT**	**WEB DEVELOPMENT**	**DATABASE PROGRAMMING**	**SERVER PROGRAMMING**	
NOSS TITLE	**APPLICATION DEVELOPMENT**					
COMPETENCY UNIT TITLE	**APPLICATION DEVELOPMENT SUPERVISION**					
LEARNING OUTCOME	The outcome of this competency is to enable trainee to perform supervisory skills to support operation according to company's requirements. Upon completion of this competency unit, trainees will be able to:- • Confirm facilities and equipment functionality • Prepare job schedule • Perform unit meeting / briefing					
PRE-REQUISITE (if applicable)	Nil					

COMPETENCY UNIT ID	IT-010-3:2016 –C07	**LEVEL**	3	**TRAINING DURATION**	100	**SKILL CREDIT**	10

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
1. Confirm facilities and equipment functionality	i. Type of facilities such as • Server • PC ii. Type of equipment such as • Keyboard • Mouse • Printer iii. Facilities and equipment inventory checklist iv. Malfunction/	i. Check type of facilities and equipment against checklist ii. Ensure facilities and equipment availability iii. Ensure facilities and equipment functions iv. Arrange work order on malfunction of facilities and equipment v. Follow up status of malfunction facilities and equipment	Attitude: i. Responsible in assuring facilities and equipment functionality & operability ii. Timely in reporting facilities and equipment status	Related Knowledge 5 Related Skills 11	Related Knowledge Lecture & Case Study Related Skills Demonstration & Observation	i. Type of facilities and equipment listed and availability confirmed ii. Facilities and equipment functions specified and described iii. Facilities and equipment

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
	irregularities of facilities and equipment reporting procedure	vi. Confirm rectification on malfunction facilities and equipment done vii. Record status of facilities and equipment				inventory checklist updated iv. Malfunction/ irregularities of facilities and equipment recording /reporting procedure followed
2. Prepare job schedule	i. Scope of work and job specification ii. Type and function of scheduling • Daily • Weekly • Monthly iii. Job assignment and delegation iv. Duty roster format Standard Operating Procedure (SOP)	i. Determine type and function of scheduling ii. Check scope of work, job descriptions iii. Assign personnel for duty iv. Produce duty roster / jobs schedule	<u>Attitude:</u> i. Attentive to details in preparing duty roster ii. Non-bias in assigning job schedule	<u>Related Knowledge</u> 5 <u>Related Skills</u> 11	<u>Related Knowledge</u> Lecture & Case Study <u>Related Skills</u> Demonstration & Observation	i. Scope of work, job descriptions listed and described ii. Number of available personnel specified iii. Assignments confirmed and personnel to undertake job functions listed iv. Duty roster scheduled, formatted and generated

Work Activities	Related Knowledge	Related Skills	Attitude/Safety/ Environmental	Training Hours	Delivery Mode	Assessment Criteria
3. Perform unit meeting / briefing	i. Meeting / briefing requirements and preparation • Procedure & protocols of meeting • Type of meeting • Attendee / participant of meeting • Agenda of meeting • Minutes of meeting • Meeting documentation	i. Identify meeting / briefing requirements ii. Conduct daily staff briefing iii. Conduct unit meeting iv. Comply with briefing / meeting procedure v. Execute communication outcome / decision	Attitude: i. Organised and systematic in arranging meeting ii. Punctual for meeting iii. Sound decision making while in meeting	Related Knowledge 14 Related Skills 33	Related Knowledge Lecture & Case Study Related Skills Demonstration & Observation	i. Daily staff briefing agenda listed and discussed ii. Meeting protocols and procedure followed iii. Decision on meeting specified and executed

Employability Skills

Core Abilities	Social Skills
01.01 Identify and gather information. 01.02 Document information procedures or processes. 01.03 Utilise basic IT applications. 02.01 Interpret and follow manuals, instructions and SOP's. 02.02 Follow telephone/telecommunication procedures. 02.03 Communicate clearly. 02.04 Prepare brief reports and checklist using standard forms. 03.01 Apply cultural requirement to the workplace. 03.02 Demonstrate integrity and apply practical practices. 03.03 Accept responsibility for own work and work area. 03.04 Seek and act constructively upon feedback about work performance. 03.05 Demonstrate safety skills.	1. Communication skills 2. Conceptual skills 3. Interpersonal skills 4. Learning skills 5. Leadership skills 6. Multitasking and prioritising 7. Self-discipline 8. Teamwork

03.06 Respond appropriately to people and situations. 03.07 Resolve interpersonal conflicts. 01.04 Analyse information. 02.06 Write memos and letters. 03.08 Develop and maintain a cooperation within work group. 01.07 Utilise database applications to locate and process information. 01.08 Utilise spread sheets applications to locate and process information. 01.10 Apply a variety of mathematical techniques. 01.11 Apply thinking skills and creativity. 02.10 Prepare reports and instructions. 02.11 Convey information and ideas to people. 03.09 Manage and improve performance of individuals. 03.10 Provide consultations and counselling. 03.13 Develop and maintain team harmony and resolve conflicts. 03.14 Facilitate and coordinate teams and ideas. 03.15 Liaise to achieve identified outcomes. 03.16 Identify and assess client/customer needs.	

Tools, Equipment and Materials (TEM)

ITEMS	**RATIO (TEM : Trainees)**
1. Computer with internet and peripherals	1:2
2. Office facilities (printer, fax, machine)	As per requirements
3. Sample of work flow chart	1:1
4. Sample of company policies and various procedures manual (SOP,transaction, recording,reporting, documentation, facilities waste disposal)	1:1
5. Sample of duty roster format	1:1
6. Sample of inventory list	1:1
7. Training facilities (Audio Visual,rooms , materials/ modules)	1:1
8. Sample of Company Key Performance Index (KPI) document	As per requirements
9. Sample appraisal documentation (subordinates list,subordinate profiles, appraisal form)	1:1

REFERENCES

1. Asgar, J. 2008. The Organizational Role of Supervisors. Las, NV: Practical Management. ISBN: 9781599429694
2. Evans, D. 1999. Supervisory Management: Principles and Practice. London: Continuum. ISBN: 9780826457332
3. Leonard, E.C. 2013. Supervision: Concepts and Practices of Management. Cengage Learning. ISBN: 9781111969790
4. Mosley, D.C. & Pietri, P.H. 2011. Supervisory Management: The Art of Inspiring, Empowering, and Developing People. Cengage Learning. ISBN: 9780538737074

CU ID	COMPETENCY UNIT TITLE	WORKS ACTIVITIES	RELATED KNOWLEDGE	RELATED SKILLS	HOURS
			(A)	(B)	(A+B)
CU 1	APPLICATION PROTOTYPE DEVELOPMENT	Interpret application prototype development requirements	7	11	18
		Setup local environment	22	32	54
		Implement application prototype mock up flow	22	32	54
		Conduct user interface and user experience functionality test	18	27	45
		Commit prototype source code	4	5	9
CU 2	APPLICATION MODULE DEVELOPMENT	Interpret application module development requirements	22	32	54
		Setup local environment	43	65	108
		Plan module expected behaviour	54	81	135
		Write module code	86	130	216
		Commit source code	11	16	27
CU 3	APPLICATION MODULE INTEGRATION	Interpret module integration requirements	11	16	27
		Perform modules integration	54	81	135
		Test module integration code	38	57	95
		Commit module integration code	5	8	14
CU 4	DEVELOPMENT ENVIRONMENT DEPLOYMENT	Deploy source code to development server	43	65	108
		Perform data population to development server	43	65	108
		Perform integrity system testing	58	86	144
CU 5	APPLICATION BUG FIXING	Interpret bug report	7	11	18
		Simulate bug/error scenario	65	97	162
		Debug application code	65	97	162
		Commit fixed source code	7	11	18
CU 6	APPLICATION SYSTEM DOCUMENTATION COMPILATION	Interpret system documentation requirements	7	11	18
		Compile system documentation	22	32	54
		Upload system documentation	7	11	18
CU 7	APPLICATION DEVELOPMENT SUPERVISION	Confirm facilities and equipment functionality	5	11	16
		Prepare job schedule	5	32	37
		Perform unit meeting / briefing	14	33	47
		TOTAL HOURS	**744**	**1156**	**1900**