TRAINING REGULATIONS

PROGRAMMING (.NET TECHNOLOGY) NC III

INFORMATION AND COMMUNICATION TECHNOLOGY (ICT) SECTOR

TABLE OF CONTENTS

ICT SECTOR

PROGRAMMING (.NET TECHNOLOGY) - NATIONAL CERTIFICATE LEVEL III

TRAINING REGULATIONS FOR
PROGRAMMING (.NET TECHNOLOGY)NC III

Section 1   PROGRAMMING (.NET TECHNOLOGY) NC III QUALIFICATIONS

The **PROGRAMMING(.NET TECHNOLOGY) NC III** Qualification consists of competencies that a person must achieve to develop or write program codes using a personal computer or workstation as part of a systems development team. Specifically, it includes specialized competencies on .NET technology programming language, such as: 1.) programming in HTML5 with JavaScript & CSS3 and 2.) developing ASP.NET MVC 5 web applications.

This Qualification is packaged from the competency map of the Information and Communication Technology Industry (Service sector) as shown in Annex A.

The units of competency comprising this qualification include the following:

CODE NO.	BASIC COMPETENCIES
5 00 311 1 09	Lead workplace communication
5 00 311 1 10	Lead small teams
5 00 311 1 11	Develop and practice negotiation skills
5 00 311 1 12	Solve problems related to work activities
5 00 311 1 13	Use mathematical concepts and techniques
5 00 311 1 14	Use relevant technologies

CODE NO.	COMMON COMPETENCIES
ICT315202	Apply quality standards
ICT311203	Perform computer operations

CODE NO.	CORE COMPETENCIES
ICT313357	Perform Programming in HTML5 with JavaScript and CSS3
ICT313358	Develop ASP.NET MVC 5 Web Applications

A person who has achieved this Qualification can be employed in any or more of the following:

- Front-end web developer
- Windows store application developer
- Web Application developer
- Instructor/Professor

SECTION 2   COMPETENCY STANDARDS

This section gives the details of the contents of the basic, common, core and specialized units of competency required in **PROGRAMMING (.NET TECHNOLOGY) NC III.**

BASIC COMPETENCIES

UNIT OF COMPETENCY: LEAD WORKPLACE COMMUNICATION

UNIT CODE   :   500311109

UNIT DESCRIPTOR   :   This unit covers the knowledge, skills and attitudes required to lead in the dissemination and discussion of ideas, information and issues in the workplace.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Communicate information about workplace processes	1.1. Appropriate *communication method* is selected
	1.2. Multiple operations involving several topics areas are communicated accordingly
	1.3. Questions are used to gain extra information
	1.4. Correct sources of information are identified
	1.5. Information is selected and organized correctly
	1.6. Verbal and written reporting is undertaken when required
	1.7. Communication skills are maintained in all situations
2. Lead workplace discussions	2.1. Response to workplace issues are sought
	2.2. Response to workplace issues are provided immediately
	2.3. Constructive contributions are made to workplace discussions on such issues as production, quality and safety
	2.4. Goals/objectives and action plan undertaken in the workplace are communicated
3. Identify and communicate issues arising in the workplace	3.1. Issues and problems are identified as they arise
	3.2. Information regarding problems and issues are organized coherently to ensure clear and effective communication
	3.3. Dialogue is initiated with appropriate personnel
	3.4. Communication problems and issues are raised as they arise

RANGE OF VARIABLES

VARIABLE	RANGE
1. Methods of communication	1.1. Non-verbal gestures 1.2. Verbal 1.3. Face to face 1.4. Two-way radio 1.5. Speaking to groups 1.6. Using telephone 1.7. Written 1.8. Internet

EVIDENCE GUIDE

1. Critical aspects of Competency	Assessment requires evidence that the candidate: 1.1. Dealt with a range of communication/information at one time 1.2. Made constructive contributions in workplace issues 1.3. Sought workplace issues effectively 1.4. Responded to workplace issues promptly 1.5. Presented information clearly and effectively written form 1.6. Used appropriate sources of information 1.7. Asked appropriate questions 1.8. Provided accurate information
2. Underpinning knowledge	2.1. Organization requirements for written and electronic communication methods 2.2. Effective verbal communication methods
3. Underpinning Skills	3.1. Organize information 3.2. Understand and convey intended meaning 3.3. Participate in variety of workplace discussions 3.4. Comply with organization requirements for the use of written and electronic communication methods
4. Resource Implications	The following resources **MUST** be provided: 4.1. Variety of Information 4.2. Communication tools 4.3. Simulated workplace
5. Methods of Assessment	Competency may be assessed through: 5.1. Competency in this unit must be assessed through 5.2. Direct Observation 5.3. Interview
6. Context for Assessment	6.1. Competency may be assessed in the workplace or in simulated workplace environment

UNIT OF COMPETENCY: LEAD SMALL TEAMS

UNIT CODE   :   500311110

UNIT DESCRIPTOR   :   This unit covers the knowledge, skills and attitudes to lead small teams including setting and maintaining team and individual performance standards.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Provide team leadership	1.1. ***Work requirements*** are identified and presented to team members 1.2. Reasons for instructions and requirements are communicated to team members 1.3. ***Team members' queries and concerns*** are recognized, discussed and dealt with
2. Assign responsibilities	2.1. Duties, and responsibilities are allocated having regard to the skills, knowledge and aptitude required to properly undertake the assigned task and according to company policy 2.2. Duties are allocated having regard to individual preference, domestic and personal considerations, whenever possible
3. Set performance expectations for team members	3.1. Performance expectations are established based on client needs and according to assignment requirements 3.2. Performance expectations are based on individual team members duties and area of responsibility 3.3. Performance expectations are discussed and disseminated to individual team members
4. Supervise team performance	4.1. ***Monitoring of performance*** takes place against defined performance criteria and/or assignment instructions and corrective action taken if required 4.2. Team members are provided with **feedback**, positive support and advice on strategies to overcome any deficiencies 4.3. ***Performance issues*** which cannot be rectified or addressed within the team are referenced to appropriate personnel according to employer policy 4.4. Team members are kept informed of any changes in the priority allocated to assignments or tasks which might impact on client/customer needs and satisfaction 4.5. Team operations are monitored to ensure that employer/client needs and requirements are met 4.6. Follow-up communication is provided on all issues affecting the team 4.7. All relevant documentation is completed in accordance with company procedures

RANGE OF VARIABLES

VARIABLE	RANGE
1. Work requirements	1.1. Client Profile 1.2. Assignment instructions
2. Team member's concerns	2.1. Roster/shift details
3. Monitor performance	3.1. Formal process 3.2. Informal process
4. Feedback	4.1. Formal process 4.2. Informal process
5. Performance issues	5.1. Work output 5.2. Work quality 5.3. Team participation 5.4. Compliance with workplace protocols 5.5. Safety 5.6. Customer service

EVIDENCE GUIDE

1. Critical Aspects of Competency	Assessment requires evidence that the candidate:	
	1.1.	Maintained or improved individuals and/or team performance given a variety of possible scenario
	1.2.	Assessed and monitored team and individual performance against set criteria
	1.3.	Represented concerns of a team and individual to next level of management or appropriate specialist and to negotiate on their behalf
	1.4.	Allocated duties and responsibilities, having regard to individual's knowledge, skills and aptitude and the needs of the tasks to be performed
	1.5.	Set and communicated performance expectations for a range of tasks and duties within the team and provided feedback to team members
2. Underpinning Knowledge	2.1.	Company policies and procedures
	2.2.	Relevant legal requirements
	2.3.	How performance expectations are set
	2.4.	Methods of Monitoring Performance
	2.5.	Client expectations
	2.6.	Team member's duties and responsibilities
3. Underpinning Skills	3.1.	Communication skills required for leading teams
	3.2.	Informal performance counseling skills
	3.3.	Team building skills
	3.4.	Negotiating skills
4. Resource Implications	The following resources **MUST** be provided:	
	4.1.	Access to relevant workplace or appropriately simulated environment where assessment can take place
	4.2.	Materials relevant to the proposed activity or task
5. Methods of Assessment	Competency may be assessed through:	
	5.1.	Direct observations of work activities of the individual member in relation to the work activities of the group
	5.2.	Observation of simulation and/or role play involving the participation of individual member to the attainment of organizational goal
	5.3.	Case studies and scenarios as a basis for discussion of issues and strategies in teamwork
6. Context of Assessment	6.1.	Competency assessment may occur in workplace or any appropriately simulated environment
	6.2.	Assessment shall be observed while task are being undertaken whether individually or in-group

UNIT OF COMPETENCY: DEVELOP AND PRACTICE NEGOTIATION SKILLS

UNIT CODE : **500311111**

UNIT DESCRIPTOR : This unit covers the skills, knowledge and attitudes required to collect information in order to negotiate to a desired outcome and participate in the negotiation.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Plan negotiations	1.1 Information on *preparing for negotiation* is identified and included in the plan 1.2 Information on creating *nonverbal environments* for positive negotiating is identified and included in the plan 1.3 Information on *active listening* is identified and included in the plan 1.4 Information on different *questioning techniques* is identified and included in the plan 1.5 Information is checked to ensure it is correct and up-to-date
2. Participate in negotiations	2.1 Criteria for successful outcome are agreed upon by all parties 2.2 Desired outcome of all parties are considered 2.3 Appropriate language is used throughout the negotiation 2.4 A variety of questioning techniques are used 2.5 The issues and processes are documented and agreed upon by all parties 2.6 Possible solutions are discussed and their viability assessed 2.7 Areas for agreement are confirmed and recorded 2.8 Follow-up action is agreed upon by all parties

RANGE OF VARIABLES

VARIABLE	RANGE

1. Preparing for negotiation	1.1 Background information on other parties to the negotiation
	1.2 Good understanding of topic to be negotiated
	1.3 Clear understanding of desired outcome/s
	1.4 Personal attributes
	    1.4.1 Self-awareness
	    1.4.2 Self esteem
	    1.4.3 objectivity
	    1.4.4 empathy
	    1.4.5 respect for others
	1.5 Interpersonal skills
	    1.5.1 listening/reflecting
	    1.5.2 nonverbal communication
	    1.5.3 assertiveness
	    1.5.4 behavior labeling
	    1.5.5 testing understanding
	    1.5.6 seeking information
	    1.5.7 self-disclosing
	1.6 Analytic skills
	    1.6.1 observing differences between content and process
	    1.6.2 identifying bargaining information
	    1.6.3 applying strategies to manage process
	    1.6.4 applying steps in negotiating process
	    1.6.5 strategies to manage conflict
	    1.6.6 steps in negotiating process
	    1.6.7 options within organization and externally for resolving conflict

2. Non-verbal environments	2.1 Friendly reception 2.2 Warm and welcoming room 2.3 Refreshments offered 2.4 Lead in conversation before negotiation begins
3. Active listening	3.1 Attentive 3.2 Don't interrupt 3.3 Good posture 3.4 Maintain eye contact 3.5 Reflective listening
4. Questioning techniques	4.1 Direct 4.2 Indirect 4.3 Open-ended

EVIDENCE GUIDE

1. Critical Aspects of Competency	Assessment requires evidence that the candidate: 1.1 Demonstrated sufficient knowledge of the factors influencing negotiation to achieve agreed outcome 1.2 Participated in negotiation with at least one person to achieve an agreed outcome
2. Underpinning Knowledge and Attitude	2.1 Codes of practice and guidelines for the organization 2.2 Organizations policy and procedures for negotiations 2.3 Decision making and conflict resolution strategies procedures 2.4 Problem solving strategies on how to deal with unexpected questions and attitudes during negotiation 2.5 Flexibility 2.6 Empathy
3. Underpinning Skills	3.1 Interpersonal skills to develop rapport with other parties 3.2 Communication skills (verbal and listening) 3.3 Observation skills 3.1 Negotiation skills
4. Resource Implications	The following resources **MUST** be provided: 4.1 Room with facilities necessary for the negotiation process 4.2 Human resources (negotiators)
5. Methods of Assessment	Competency may be assessed through: 5.1 Observation/demonstration and questioning 5.2 Portfolio assessment 5.3 Oral and written questioning 5.4 Third party report
6. Context for Assessment	6.1 Competency to be assessed in real work environment or in a simulated workplace setting.

UNIT OF COMPETENCY: **SOLVE PROBLEMS RELATED TO WORK ACTIVITIES**

UNIT CODE : **500311112**

UNIT DESCRIPTOR : This unit of covers the knowledge, skills and attitudes required to solve problems in the workplace including the application of problem solving techniques and to determine and resolve the root cause of problems.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables	
1. Identify the problem	1.1.	Variances are identified from normal operating parameters; and product quality
	1.2.	Extent, cause and nature are of the problem are defined through observation, investigation and *analytical techniques*
	1.3.	*Problems* are clearly stated and specified
2. Determine fundamental causes of the problem	2.1.	Possible causes are identified based on experience and the use of problem solving tools / analytical techniques.
	2.2.	Possible cause statements are developed based on findings
	2.3.	Fundamental causes are identified per results of investigation conducted
3. Determine corrective action	3.1.	All possible options are considered for resolution of the problem
	3.2.	Strengths and weaknesses of possible options are considered
	3.3.	Corrective actions are determined to resolve the problem and possible future causes
	3.4.	**Action *plans*** are developed identifying measurable objectives, resource needs and timelines in accordance with safety and operating procedures
4. Provide recommendation/s to manager	4.1.	Report on recommendations are prepared
	4.2.	Recommendations are presented to appropriate personnel.
	4.3.	Recommendations are followed-up, if required

RANGE OF VARIABLES

VARIABLE	RANGE
1. Analytical techniques	1.1. Brainstorming 1.2. Intuitions/Logic 1.3. Cause and effect diagrams 1.4. Pareto analysis 1.5. SWOT analysis 1.6. Gant chart, Pert CPM and graphs 1.7. Scatter grams
2. Problem	2.1. Non–routine process and quality problems 2.2. Equipment selection, availability and failure 2.3. Teamwork and work allocation problem 2.4. Safety and emergency situations and incidents
3. Action plans	3.1. Priority requirements 3.2. Measurable objectives 3.3. Resource requirements 3.4. Timelines 3.5. Coordination and feedback requirements 3.6. Safety requirements 3.7. Risk assessment 3.8. Environmental requirements

EVIDENCE GUIDE

1. Critical Aspects of Competency	Assessment requires evidence that the candidate:
	1.1. Identified the problem
	1.2. Determined the fundamental causes of the problem
	1.3. Determined the correct / preventive action
	1.4. Provided recommendation to manager
	These aspects may be best assessed using a range of scenarios / case studies / what ifs as a stimulus with a walk through forming part of the response. These assessment activities should include a range of problems, including new, unusual and improbable situations that may have happened.
2. Underpinning Knowledge	2.1. Competence includes a thorough knowledge and understanding of the process, normal operating parameters, and product quality to recognize non-standard situations
	2.2. Competence to include the ability to apply and explain, sufficient for the identification of fundamental cause, determining the corrective action and provision of recommendations
	2.2.1. Relevant equipment and operational processes
	2.2.2. Enterprise goals, targets and measures
	2.2.3. Enterprise quality, OHS and environmental requirement
	2.2.4. Principles of decision making strategies and techniques
	2.2.5. Enterprise information systems and data collation
	2.2.6. Industry codes and standards
3. Underpinning Skills	3.1. Using range of formal problem solving techniques
	3.2. Identifying and clarifying the nature of the problem
	3.3. Devising the best solution
	3.4. Evaluating the solution
	3.5. Implementation of a developed plan to rectify the problem

4. Resource Implications	4.1. Assessment will require access to an operating plant over an extended period of time, or a suitable method of gathering evidence of operating ability over a range of situations. A bank of scenarios / case studies / what ifs will be required as well as bank of questions which will be used to probe the reason behind the observable action.
5. Methods of Assessment	Competency may be assessed through: 5.1. Case studies on solving problems in the workplace 5.2. Observation The unit will be assessed in a holistic manner as is practical and may be integrated with the assessment of other relevant units of competency. Assessment will occur over a range of situations, which will include disruptions to normal, smooth operation. Simulation may be required to allow for timely assessment of parts of this unit of competency. Simulation should be based on the actual workplace and will include walk through of the relevant competency components.
6. Context of Assessment	6.1. In all workplace, it may be appropriate to assess this unit concurrently with relevant teamwork or operation units.

UNIT OF COMPETENCY: **USE MATHEMATICAL CONCEPTS AND TECHNIQUES**

UNIT CODE   **:**   **500311113**

UNIT DESCRIPTOR   **:**   This unit covers the knowledge, skills and attitudes required in the application of mathematical concepts and techniques.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Identify mathematical tools and techniques to solve problem	1.1 Problem areas are identified based on given condition 1.2 *Mathematical techniques* are selected based on the given problem
2. Apply mathematical procedure/solution	2.1 Mathematical techniques are applied based on the problem identified 2.2 Mathematical computations are performed to the level of accuracy required for the problem 2.3 Results of mathematical computation is determined and verified based on job requirements
3. Analyze results	3.1 Result of application is reviewed based on expected and required specifications and outcome 3.2 *Appropriate action* is applied in case of error

RANGE OF VARIABLES

VARIABLE	RANGE
1. Mathematical techniques	May include but are not limited to: 1.1 Four fundamental operations 1.2 Measurements 1.3 Use/Conversion of units of measurements 1.4 Use of standard formulas
2. Appropriate action	2.1 Review in the use of mathematical techniques (e.g. recalculation, re-modeling) 2.2 Report error to immediate superior for proper action

EVIDENCE GUIDE

1. Critical Aspects of Competency	Assessment requires evidence that the candidate: 1.1 Identified, applied and reviewed the use of mathematical concepts and techniques to workplace problems
2. Underpinning Knowledge	2.1 Fundamental operation (addition, subtraction, division, multiplication) 2.2 Measurement system 2.3 Precision and accuracy 2.4 Basic measuring tools/devices
3. Underpinning Skills	3.1 Applying mathematical computations 3.2 Using calculator 3.3 Using different measuring tools
4. Resource Implications	The following resources **MUST** be provided: 4.1 Calculator 4.2 Basic measuring tools 4.3 Case Problems
5. Methods of Assessment	Competency may be assessed through: 5.1 Authenticated portfolio 5.2 Written Test 5.3 Interview/Oral Questioning 5.4 Demonstration
6. Context of Assessment	6.1 Competency may be assessed in the work place or in a simulated work place setting

UNIT OF COMPETENCY: USE RELEVANT TECHNOLOGIES

UNIT CODE   :   500311114

UNIT DESCRIPTOR   :   This unit of competency covers the knowledge, skills, and attitude required in selecting, sourcing and applying appropriate and affordable technologies in the workplace.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Study/select appropriate technology	1.1 Usage of different **technologies** is determined based on job requirements 1.2 Appropriate technology is selected as per work specification
2. Apply relevant technology	2.1 Relevant technology is effectively used in carrying out function 2.2 Applicable software and hardware are used as per task requirement 2.3 **Management concepts** are observed and practiced as per established industry practices
3. Maintain/enhance relevant technology	3.1 Maintenance of technology is applied in accordance with the **industry standard operating procedure**, **manufacturer's operating guidelines** and **occupational health and safety procedure** to ensure its operative ability 3.2 Updating of technology is maintained through continuing education or training in accordance with job requirement 3.3 Technology failure/ defect is immediately reported to the concern/responsible person or section for **appropriate action**

RANGE OF VARIABLES

VARIABLE	RANGE
1. Technology	May include but are not limited to: 1.1 Office technology 1.2 Industrial technology 1.3 System technology 1.4 Information technology 1.5 Training technology
2. Management concepts	May include but not limited to: 2.1 Real Time Management 2.2 KAIZEN or continuous improvement 2.3 5s 2.4 Total Quality Management 2.5 Other management/productivity tools
3. Industry standard operating procedure	3.1 Written guidelines relative to the usage of office technology/equipment 3.2 Verbal advise/instruction from the co-worker
4. Manufacturer's operating guidelines/ instructions	4.1 Written instruction/manuals of specific technology/ equipment 4.2 General instruction manual 4.3 Verbal advise from manufacturer relative to the operation of equipment
5. Occupational health and safety procedure	5.1 Relevant statutes on OHS 5.2 Company guidelines in using technology/equipment
6. Appropriate action	6.1 Implementing preventive maintenance schedule 6.2 Coordinating with manufacturer's technician

EVIDENCE GUIDE

1. Critical Aspects of Competency	Assessment requires evidence that the candidate: 1.1 Studied and selected appropriate technology consistent with work requirements 1.2 Applied relevant technology 1.3 Maintained and enhanced operative ability of relevant technology
2. Underpinning Knowledge	2.1 Awareness on technology and its function 2.2 Repair and maintenance procedure 2.3 Operating instructions 2.4 Applicable software 2.5 Communication techniques 2.6 Health and safety procedure 2.7 Company policy in relation to relevant technology 2.8 Different management concepts 2.9 Technology adaptability
3. Underpinning Skills	3.1 Relevant technology application/implementation 3.2 Basic communication skills 3.3 Software applications skills 3.4 Basic troubleshooting skills
4. Resource Implications	The following resources **MUST** be provided: 4.1 Relevant technology 4.2 Interview and demonstration questionnaires 4.3 Assessment packages
5. Methods of Assessment	Competency must be assessed through: 5.1 Interview 5.2 Actual demonstration 5.3 Authenticated portfolio (related certificates of training/seminar)
6. Context of Assessment	6.1 Competency may be assessed in actual workplace or simulated environment

COMMON COMPETENCIES

UNIT TITLE   :   **APPLY QUALITY STANDARDS**

UNIT CODE   :   **506315202**

UNIT DESCRIPTOR   :   This unit covers the knowledge, skills, attitudes and values needed to apply quality standards in the workplace. The unit also includes the application of relevant safety procedures and regulations, organization procedures and customer requirements.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Assess quality of received materials	1.1. Work instruction is obtained and work is carried out in accordance with standard operating procedures. 1.2. Received **materials** are checked against workplace standards and specifications. 1.3. Faulty materials related to work are identified and isolated. 1.4. **Faults** and any identified causes are recorded and/or reported to the supervisor concerned in accordance with workplace procedures. 1.5. Faulty materials are replaced in accordance with workplace procedures.
2. Assess own work	2.1. **Documentation** relative to quality within the company is identified and used. 2.2. Completed work is checked against workplace standards relevant to the task undertaken. 2.3. **Errors** are identified and isolated. 2.4. Information on the quality and other indicators of production performance are recorded in accordance with workplace procedures. 2.5. In cases of deviations from specific **quality standards**, causes are documented and reported in accordance with the workplace' s standards operating procedures.
3. Engage in quality improvement	3.1. Process improvement procedures are participated in relative to workplace assignment. 3.2. Work is carried out in accordance with process improvement procedures. 3.3. Performance of operation or quality of product of service to ensure **customer** satisfaction is monitored.

RANGE OF VARIABLES

VARIABLE	RANGE

1 Materials	1.1 Materials may include but not limited to:
	    1.1.1. Manuals
	    1.1.2. Job orders
	    1.1.3. Instructional videos

2 Faults	2.1 Faults may include but not limited to:
	    2.1.1. Materials not to specification
	    2.1.2. Materials contain incorrect/outdated information
	    2.1.3. Hardware defects
	    2.1.4. Materials that do not conform with any regulatory agencies

3 Documentation	3.1 Organization work procedures 3.2 Manufacturer's instruction manual 3.3 Customer requirements 3.4 Forms

4 Errors	4.1 Errors may be related but not limited to the following:
	    4.1.1. Deviation from the requirements of the Client
	    4.1.2. Deviation from the requirement of the organization

5 Quality standards	5.1 Quality standards may be related but not limited to the following:
	    5.1.1. Materials
	    5.1.2. Hardware
	    5.1.3. Final product
	    5.1.4. Production processes
	    5.1.5. Customer service

6 Customer	6.1 Co-worker 6.2 Supplier/Vendor 6.3 Client 6.4 Organization receiving the product or service

EVIDENCE GUIDE

1	Critical aspect of competency	Assessment must show that the candidate: 1.1 Carried out work in accordance with the company's standard operating procedures 1.2 Performed task according to specifications 1.3 Reported defects detected in accordance with standard operating procedures 1.4 Carried out work in accordance with the process improvement procedures
2	Underpinning knowledge	2.1 Relevant production processes, materials and products 2.2 Characteristics of materials, software and hardware used in production processes 2.3 Quality checking procedures 2.4 Workplace procedures 2.5 Safety and environmental aspects of production processes 2.6 Fault identification and reporting 2.7 Quality improvement processes
3	Underpinning skills	3.1 Reading skills required to interpret work instruction 3.2 Communication skills needed to interpret and apply defined work procedures 3.3 Carry out work in accordance with OHS policies and procedures 3.4 Critical thinking 3.5 Solution providing and decision-making
4	Method of assessment	The assessor must select two of the following to objectively evaluate the candidate: 4.1 Observation and oral questioning 4.2 Third party report 4.3 Portfolio 4.4 Practical demonstration
5	Resource implication	Materials, software and hardware to be used in a real or simulated situation
6	Context of Assessment	Assessment may be conducted in the workplace or in a simulated environment

| UNIT TITLE | : | **PERFORM COMPUTER OPERATIONS** |

| UNIT CODE | : | **506311203** |

| UNIT DESCRIPTOR | : | This unit covers the knowledge, skills, attitudes and values needed to perform computer operations which include inputting, accessing, producing and transferring data using the appropriate hardware and software. |

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Plan and prepare for task to be taken undertaken	1.1. Requirements of task are determined in accordance with the required output. 1.2. Appropriate **hardware** and **software** are selected according to task assigned and required outcome. 1.3. Task is planned to ensure that **OH & S guidelines and** procedures are followed. 1.4. Client -specific guidelines and procedures are followed. 1.5. Required data security guidelines are applied in accordance with existing procedures.
2. Input data into computer	2.1. Data are entered into the computer using appropriate program/application in accordance with company procedures. 2.2. Accuracy of information is checked and information is saved in accordance with standard operating procedures. 2.3. Inputted data is stored in **storage media** according to requirements. 2.4. Work is performed within **ergonomic guidelines.**
3. Access information using computer	3.1. Correct program/application is selected based on job requirements. 3.2. Program/application containing the information required is accessed according to company procedures. 3.3. **Desktop icons** are correctly selected, opened and closed for navigation purposes. 3.4. Keyboard techniques are carried out in line with OH & S requirements for safe use of keyboards.

4. Produce output/ data using computer system	4.1. Entered data are processed using appropriate software commands.
	4.2. Data are printed out as required using computer hardware /peripheral devices in accordance with standard operating procedures.
	4.3. Files and data are transferred between compatible systems using computer software, hardware/peripheral devices in accordance with standard operating procedures.
5. Use basic functions of a www-browser to locate information	5.1. Information requirements for internet search are established.
	5.2. Browser is launched.
	5.3. Search engine is loaded.
	5.4. Appropriate search criteria/or URL of site is entered.
	5.5. Relevant links are followed to locate required information.
	5.6. Useful pages are bookmarked or printed as required.
6. Maintain computer equipment and systems	6.1. Procedures for ensuring security of data, including regular back-ups and virus checks are implemented in accordance with standard operating procedures.
	6.2. Basic file maintenance procedures are implemented in line with the standards operating procedures.

RANGE OF VARIABLES

VARIABLE	RANGE
1 Hardware and peripheral devices	1.1 Personal computers 1.2 Networked systems 1.3 Communication equipment 1.4 Printers 1.5 Scanners 1.6 Keyboard 1.7 Mouse 1.8 Voice/Data logger
2 Software	Software includes the following but not limited to: 2.1 Word processing packages 2.2 Database packages 2.3 Internet 2.4 Spreadsheets 2.5 Client Specific Software
3 OH & S guidelines	3.1 OHS guidelines 3.2 Enterprise procedures
4 Storage media	Storage media include the following but not limited to: 4.1 Diskettes 4.2 CDs 4.3 Zip disks 4.4 hard disk drives, local and remote 4.5 Optical drives
5 Ergonomic guidelines	5.1 Types of equipment used 5.2 Appropriate furniture 5.3 Seating posture 5.4 Lifting posture 5.5 Visual display unit screen brightness
6 Desktop icons	6.1 Icons include the following but not limited to: 6.2 Directories/folders 6.3 Files 6.4 Network devices 6.5 Recycle bin 6.6 Program icons
7 Maintenance	7.1 Creating and managing more space in the hard disk and other peripherals 7.2 Reviewing programs 7.3 Deleting unwanted files 7.4 Backing up files 7.5 Checking hard drive for errors 7.6 Using up to date anti-virus programs 7.7 Cleaning dust from internal and external surfaces

EVIDENCE GUIDE

1	Critical aspects of competency	Assessment must show that the candidate: 1.1 Selected and used hardware components correctly and according to the task requirement 1.2 used basic software applications to create new files and documents 1.3 Produced accurate and complete data in accordance with the requirements 1.4 Used appropriate devices and procedures to transfer files/data accurately 1.5 Used basic functions of a www-browser to locate information.
2	Underpinning knowledge	2.1 Basic ergonomics of keyboard and computer user 2.2 Main types of computers and basic features of different operating systems 2.3 Main parts of a computer 2.4 Storage devices and basic categories of memory 2.5 Relevant types of software 2.6 General security, privacy legislation and copyright 2.7 Viruses 2.8 OH & S principles and responsibilities 2.9 Calculating computer capacity 2.10 Productivity Application 2.11 Business Application 2.12 System Software
3	Underpinning skills	3.1 Reading and comprehension skills required to interpret work instruction and to interpret basic user manuals. 3.2 Communication skills to identify lines of communication, request advice, follow instructions and receive feedback. 3.3 Technology skills to use equipment safely including keyboard skills.
4	Method of assessment	The assessor may select two of the following assessment methods to objectively assess the candidate: 4.1 Direct Observation and Oral Questioning 4.2 Practical demonstration
5	Resource implication	5.1 Computer hardware with peripherals 5.2 Appropriate software
6	Context of Assessment	6.1 Assessment may be conducted in the workplace or in a simulated environment

CORE COMPETENCIES

UNIT OF COMPETENCY :	**PERFORM PROGRAMMING IN HTML5 WITH JAVASCRIPT AND CSS3**
UNIT CODE	: **ICT313357**
UNIT DESCRIPTOR	: This unit provides an introduction to HTML5, CSS3, and JavaScript. This course helps students gain basic HTML5/CSS3/JavaScript programming skills. This course is an entry point into both the Web application and Windows Store apps training paths. The course focuses on using HTML5/CSS3/JavaScript to implement programming logic, define and use variables, perform looping and branching, develop user interfaces, capture and validate user input, store data, and create well-structured application. Further, this unit covers the skills, knowledge and attitude required in developing HTML5 websites. This also includes an introduction to the Windows 8 platform and the Windows Store apps.

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Develop basic HTML document using HTMP5 and CSS 3 syntax	1.1 Domain model is produced to illustrate different ***business domain features***. 1.2 Business processes are documented using agreed ***representation format***. 1.3 ***Interaction diagrams*** are produced that describe in detail how the objects will interact with the business logic to perform the methods or functions of the business processes. 1.4 Interaction diagrams are documented using agreed ***representation format***. 1.5 An HTML5 document is created and configured according to ***user's specification***. 1.6 The HTML5 document is designed using cascading style sheet 3 (CSS3).
2. Create HTML5 document using advanced techniques with JavaScript and CSS3	2.1 Static pages are created using the ***new features*** available in HTML5. 2.2 CSS3 is used to apply basic styling to the elements in an HTML5 page. 2.3 Syntax of JavaScript is explained and how to use JavaScript with HTML5 is described. 2.4 JavaScript code that manipulates the HTML DOM is written and events are handled. 2.5 How to use jQuery is described to simplify code that uses many common JavaScript APIs. 2.6 Forms that use the new ***HTML5 input types*** are created. 2.7 User input is validated and feedback is provided by using the new HTML5 attributes.

	2.8	JavaScript code is written to validate user input and feedback is provided in cases where it is not suitable to use HTML5 attributes
	2.9	Text elements are styled on an HTML5 page by using CSS3.
	2.10	Styling is applied to block elements by using CSS3.
	2.11	CSS3 selectors are used to specify the elements to be styled in a Web application.
	2.12	Graphical effects and transformations are implemented by using the new CSS3 properties
3. Communicate with Remote Data Source and Create Objects and Methods using JavaScript	3.1	Data are serialized, deserialized, sent and received by using XMLHTTPRequest object.
	3.2	Code that serializes, deserializes, sends, and receives data is simplified by using the jQuery ajax method.
	3.3	Benefits of structuring JavaScript code is described carefully to aid maintainability and extensibility.
	3.4	Best practices are explained for creating custom objects in JavaScript.
	3.5	How to extend custom and native objects is described to add functionality.
4. Create intuitive user interface and add offline support for web applications	4.1	Drag and Drop and File APIs are used to interact with files in a Web application.
	4.2	Audio and video are incorporated into a Web application.
	4.3	Location of the user running a Web application is detected by using the Geolocation API.
	4.4	How to debug and profile a Web application is explained by using the Web Timing API.
	4.5	The need to detect device capabilities is described and reacted to different form factors in a Web application.
	4.6	Web page that can dynamically adapt its layout to match different form factors is created.
	4.7	Data is saved and retrieved locally on the user's computer by using the Local Storage API.
	4.8	Offline support is provided for a Web application by using the Application Cache API.
5. Use CSS3 to create scalable graphics and animated user interface	5.1	Scalable Vector Graphics are used to add interactive graphics to an application.
	5.2	Complex graphics are drawn on an HTML5 Canvas element by using JavaScript code.
	5.3	CSS transitions are applied to elements on an HTML5 page, and write JavaScript code to detect when a transition has occurred.
	5.4	Different types of 2D and 3D transitions available with CSS3 are described.
	5.5	Complex animations are implemented by using CSS key-frames and JavaScript code.

6.	Implement real-time communications by using web-sockets and web worker process	6.1	How Web Sockets work is explained and how to send and receive data through a Web Socket is described.
		6.2	Web Socket API with JavaScript is used to connect to a Web Socket server, send and receive data, and handle the different events that can occur when a message is sent or received.
		6.3	The purpose of a Web Worker process and how it can be used to perform asynchronous processing as well as provide isolation for sensitive operations is described.
		6.4	Web Worker APIs from JavaScript code is used to create, run, and monitor a Web Worker process.
		6.5	Occupational health and safety standards in the workplace and 5S disciplines are strictly followed as per enterprise policies and procedures.
7.	Explore the Windows 8 platform and Windows Store Apps	7.1	Windows 8 platform, architecture, and features are described.
		7.2	Basics of the Windows 8 UI and Windows Store app experience and how it differs from Windows desktop apps are explained.
		7.3	New API model, how it supports building Windows Store apps, and how it supports multiple language-specific projections are explained.
		7.4	Single-page based Windows Store apps is described
		7.5	Windows store apps using **MVVM (model-view-view-model) design pattern with JavaScript** are designed and implemented.

RANGE OF VARIABLES

VARIABLE	RANGE
1. Business domain features	Business domain features may include but not limited: • Business domain objects • Inheritance relationships • Aggregation and Composition relationships • Methods or functions that identify the business processes that will be performed by the application
2. Representation format (business processes)	Representation format may be: • Class diagram • Class Responsibility cards or diagram
3. Interaction diagrams	May include: • Domain model • Business processes (see Analyze feature functionality)
4. Representation format (interaction diagrams)	Representation format may include and is not limited to: • Sequence diagrams • Communication diagrams • Interaction overview diagrams • Timing diagrams
5. User's Specifications	This may include configuring a window to control: • Accessibility • Appearance • Behavior • Configuration • Data • Design • Focus • Layout • Style
6. New Features	New features may include: • Audio • Section • Video • Article • Canvas • Nav • SVG graphics • Header • Storage • Footer • Geolocation • Aside
7. HTML5 input types	New input types may include: • date • telephone • datetime • time • datetime-local • URL • month • week • number • range • email • Search • color
8. MVVM design pattern with JavaScript	This can be implemented using, but not limited, to the following frameworks: • AngularJS • KnockoutJS

EVIDENCE GUIDE

1. Critical aspects of competency	Assessment requires evidence that the candidate:
	1.1. Developed basic HTML document using HTMP5 and CSS 3 syntax
	1.2. Created HTML5 document using advanced techniques with JavaScript and CSS3
	1.3. Communicated with Remote Data Source and Created Objects and Methods using JavaScript
	1.4. Created interactive and responsive user interface
	1.5. Added offline support to Web Applications
	1.6. Implemented an Adaptive User Interface
	1.7. Used CSS3 to create scalable graphics and animated user interface
	1.8. Implemented real-time communications by using web-sockets and web worker process
	1.9. Design and implement Windows Store applications using MVVM design pattern
2. Underpinning Knowledge and Attitude	2.1. Knowledge of UML or other modelling tools
	2.2. Create the document structure
	2.3. Write code that interacts with UI controls
	2.4. Apply styling to HTML elements programmatically
	2.5. Write code against HTML5 APIs
	2.6. Establish the scope of objects and variables
	2.7. Create and implement objects and methods
	2.8. Implement program flow
	2.9. Raise and handle an event
	2.10. Implement exception handling
	2.11. Implement a callback
	2.12. Create a web worker process
	2.13. Validate user input by using HTML5 elements
	2.14. Validate user input by using JavaScript
	2.15. Consume data
	2.16. Serialize, de-serialize and transmit data
	2.17. Style HTML properties
	2.18. Create a flexible content layout
	2.19. Create an animated and adaptive UI
	2.20. Find elements by using CSS selectors and jQuery
	2.21. Structure a CSS file by using CSS selectors
	2.22. OHS standards and 5S principles
3. Underpinning Skills	3.1. Domain modeling / Object-Oriented breakdown and encapsulation
	3.2. Function breakdown and encapsulation
	3.3. Interaction diagram modeling.
	3.4. Detailed / step-by-step process modeling
	3.5. Documentation
	3.6. Write website with HTML5, CSS3 and JavaScript
	3.7. Usage of IDE
	3.8. Debugging
	3.9. Testing
	3.10. Tracing
	3.11. Manipulating XML and JSON
	3.12. Creating an installation package
	3.13. Deploying and configuring installation package

4. Resource Implications	The following resources **MUST** be provided: 4.1. Access to PC work station and requisite software 4.2. Access to HTML5 development tools and hardware 4.3. Access to Windows store development tools and hardware 4.4. Materials relevant to the proposed activity or tasks 4.5. Internet connection (for setting up and familiarization with software sources)
5. Methods of Assessment	Competency may be assessed through: 5.1. Demonstration with questioning 5.2. Observation with questioning 5.3. Third party report 5.4. Assessment of produced documents 5.5. Deployment of "Programming in HTML5 with JavaScript and CSS3" certification exam
6. Context for Assessment	6.1. Competency may be assessed in workplace or in a simulated workplace setting 6.2. Assessment shall be observed while task are being undertaken whether individually or in group

UNIT OF COMPETENCY :   DEVELOPING ASP.NET MVC WEB APPLICATIONS

UNIT CODE   :   ICT313358

**UNIT DESCRIPTOR   :   In this course, students will learn to develop advanced
                             ASP.NET MVC 5 (or higher) applications using .NET
                             Framework 4.5 (or higher) tools and technologies. The
                             focus will be on coding activities that enhance the
                             performance and scalability of the Web site application.
                             The Model View Controller (MVC) architectural pattern will
                             be explained in this course. ASP.NET MVC will be
                             introduced and compared with Web Forms so that students
                             know when each should/could be used.**

ELEMENT	PERFORMANCE CRITERIA *Italicized* terms are elaborated in the Range of Variables
1. Employ basic project management tasks	1.1 Feature list is prepared in line with **client requirements** 1.2 Effective Excel Survey form is used to gather client information and requirements 1.3 Release plan is prepared and agreed with clients based on dependencies and business values 1.4 Visual presentation and supporting documents are presented during the client meeting 1.5 Tasks required to complete the feature are identified based on feature list and established design approach 1.6 Time estimates per task is assigned based on degree of complexity and size. 1.7 Analysis of **development resources** is performed 1.8 Tasks and To-do list are itemized using **OneNote** and **Team Sites** 1.9 **Lync Online** is used effectively for online internal meetings throughout the project development 1.10 Used test case scenarios are formulated and documented using **specified format** and in line with client requirements 1.11 Test cases are plotted in **Excel Worksheets** 1.12 Business process are documented using agreed **representation format**
2. Perform Object-oriented programming with C#	2.1 The architecture of .NET Framework applications is described and the features that Visual Studio 2013 or higher and programming language provided are used to support .NET Framework development. 2.2 The basic data types, operators, and expressions are used. 2.3 Standard programming language constructs is used. 2.4 Create and invoke methods, pass parameters to methods, and return values from methods are used. 2.5 Overloaded methods are created and optional & output parameters are used. 2.6 Exceptions are catch and handled and information are written to the event log. 2.7 Requirement for implementing logging, tracing, and profiling is explained when building large-scale applications. 2.8 Principles of object-oriented programming are described. 2.9 Custom classes are created and used.

	2.10	Custom interfaces are defined and implemented.
	2.11	Generics are used to implement type-safe collections.
	2.12	Abstract classes and inherit are defined from base classes to create a class hierarchy.
	2.13	Inherit from .NET Framework classes and extension methods are used to add custom functionality to the inherited class.
	2.14	Generic classes and methods are created.
3. Develop ASP.NET Web Applications using Model View Controller architectural pattern	3.1	The **Microsoft Web Technologies stack** is described.
	3.2	Appropriate technology is selected to use to develop any given application.
	3.3	**Architecture of web application** is designed to meet a set of functional requirements, user interface requirements, and address business models.
	3.4	The implementation of a web application that will meet a set of functional requirements, user interface requirements, and address business models is designed.
	3.5	MVC Models are **created**.
	3.6	Code that implements business logic within Model methods, properties, and events is written.
	3.7	**Controllers** to an MVC Application is added to manage user interaction
	3.8	Models are updated.
	3.9	Select and return Views.
	3.10	Razor syntax is used to **create Views** in an MVC application that display
	3.11	Data is edited.
	3.12	Interact with Models and Controllers.
4. Test and debug ASP.NET MVC Web Applications	4.1	Unit tests and debugging tools are ran against a web application in Visual Studio
	4.2	An application for troubleshooting is configured.
5. Create custom URLs and apply style to ASP.NET MVC Web applications	5.1	A web application that uses the ASP.NET routing engine to present friendly URLs is developed.
	5.2	A web application that uses the ASP.NET routing engine to a logical navigation hierarchy to users is developed.
	5.3	Template Views are used.
	5.4	A Consistent Look and Feel is applied to an MVC Application
	5.5	Web Pages for Different Browsers are adapted
6. Create responsive web pages and implementing security	6.1	Partial page updates and caching are used to reduce the network bandwidth used by an application
	6.2	**Accelerate responses** to user requests is performed.
	6.3	JavaScript code that runs on the client-side is written
	6.4	JQuery script library is utilized to **optimize the responsiveness** of an MVC web application.
	6.5	**Authentication** is implemented in an MVC web application using Membership Provider.
	6.6	**Authorization** is implemented in an MVC web application using Role Provider.
	6.7	A **secure MVC application** that resists **malicious attacks** is built.
	6.8	An MVC application that persist information about users and preferences is built.

7. Create RESTful API, modify URLs and deploy ASP.NET MVC applications	7.1 A Web API is described
	7.2 Why developers might add a Web API to an application is identified and explained.
	7.3 RESTful web API is built and consumed from other applications.
	7.4 Modifying the way browser requests are handled by an MVC application using **HTTP modules and HTTP handlers**.
	7.5 Request from browser using web sockets is intercepted.
	7.6 **Windows Azure web service** is written
	7.7 Code written to call the web service from and MVC application.
	7.8 An ASP.NET MVC 5 web application from a development computer is packaged to a web server **for staging or production**.
	7.9 An ASP.NET MVC 5 web application from a development computer is deployed to a web server for staging or production.

RANGE OF VARIABLES

VARIABLE	RANGE
1. Client requirements	Client requirements may include: • Application features • Technical requirements
2. development resources	Development resources may include: • personnel • time • client specifications
3. OneNote	Office OneNote may include: • Creating notebook • Creating sections • Record meetings
4. Team Sites	Team Sites may include: • Team tasks/to do • Team calendar • Team files • Team OneNote
5. Lync Online	Lync Online may include: • Audio and Video Conference • Desktop and PowerPoint Sharing • OneNote and File Attachment Sharing
6. Specified format	Specified format may be: • UML –Use Cases • Step-by-step procedure • Product backlog list • Sprint backlog list • Burn down chart
7. Excel Worksheet	Excel Worksheet may include but not limited: • Tables or PivotTables • Spark Lines • Graphs or Charts
8. Representation format	Representation format may be: • UML Activity diagram • Diagram for the domain model • Flow chart • Written step-by-step procedure • Table (ie. Taxation table) • Lists of company policies • Sample reports and forms
9. Microsoft web technologies stack	ASP.NET web technologies stack may include : • Personalization • Membership • Roles • Data Sources • Site Map • Resource • Security

10. Architecture of web applications	This may include the following: • Planning Models • Planning Controllers • Planning Views • Architecting *an MVC Application*
11. Create MVC Models	Creating MVC Models may include: • Creating an MVC Project and Adding a Model • Creating a New SQL Azure Database in Visual Studio • Adding Properties and Methods to MVC Models • Using Display *and Edit Annotations in MVC Models*
12. Add Controllers	This may include the following: • Adding an MVC Controller and Writing the Actions • Writing the *Action Filters in a Controller*
13. Create Views	This may include the following: • Adding a View for Web Page Display • Creating and Using a Partial View • Testing the Views
14. Accelerate Responses	This may include the following: • Choose a state management mechanism (in-process and out of process state management) • Plan for scalability; use cookies or local storage to maintain state • Apply configuration settings in web.config file • Implement session less state (for example, QueryString) • Implement page output caching (performance oriented) • Implement data caching; implement HTTP caching • Read and write string and binary data asynchronously (long-running data transfers) • Choose a connection loss strategy • Decide a strategy for *when to use Web Sockets*
15. Optimize the responsiveness	This may include the following: • Implement client validation with JQuery • Use JavaScript and the DOM to control application behavior • Extend objects by using prototypal inheritance Use AJAX to make partial page updates • Implement the UI *by using JQuery*
16. Authentication	This may include the following: • Configuring Authentication and Membership Provider • Authenticate users • Enforce authentication settings • Choose between Windows, Forms, and custom authentication • Manage user session by using cookies • Configure membership providers • Create custom membership *providers* • Building the Logon and Register Views • Building a Password *Reset View*

17. Authorization	This may include the following: • Configuring Authorization and Roles Provider • Authorizing Access to Resources • Create roles • Authorize roles by using configuration • Authorize roles programmatically • Create custom role providers; implement WCF service authorization
18. Secure MVC Web Application	This may include the following: • Secure communication by applying SSL certificates • Salt and hash passwords for storage • Use HTML encoding to prevent cross-site scripting attacks (ANTI-XSS Library) • Implement deferred validation and handle unvalidated requests, for example, form, querystring, and URL • Prevent SQL injection attacks by parameterizing queries • Prevent cross-site *request forgeries (XSRF)*
19. Malicious Attacks	This may include the following: • Cross-site scripting • SQL injection
20. HTTP Modules and HTTP Handlers	This may include the following: • Implement synchronous and asynchronous modules and handlers • Choose between modules and handlers in IIS
21. Windows Azure Web Service	This may include the following: • Creating and Coding a service on Windows Azure Cloud Services • Consuming Data from a Windows Azure Service
22. Staging or Production	Staging or production server may include: • On-premise web server • Windows Azure Cloud Service, Websites, or Virtual Machine with Internet information Services (IIS) installed

EVIDENCE GUIDE

1. Critical aspects of Competency	Assessment requires evidence that the candidate:
	1.1. Employed basic project management tasks
	1.2. Performed Object-oriented programming with C#
	1.3. Developed ASP.NET Web Applications using Model View Controller or architectural patterns
	1.4. Tested and debugged ASP.NET MVC Web Applications
	1.5. Created custom URLs and apply style to ASP.NET MVC Web applications
	1.6. Created responsive web pages and implementing security
	1.7. Created RESTful API, modify URLs and deploy ASP.NET MVC applications
2. Underpinning Knowledge	2.1. Value of an end-to-end view of Application Lifecycle Management (ALM) tools and practices. ALM practice may include agile methodologies, such as, SCRUM and KANBAN
	2.2. Benefits of fast feedback
	2.3. Implementation strategies to reduce end-to-end cycle time
	2.4. Implementation strategies to improve software quality
	2.5. Implementation strategies to reduce waste
	2.6. Create a process improvement plan
	2.7. Role of different ALM processes
	2.8. Implementation of a Scrum/Agile process
	2.9. Define a Scrum/Agile process for a team
	2.10. Implementation of Microsoft Solution Framework for CMMI process improvement
	2.11. Plan a release
	2.12. Definition of a project-tracking process
	2.13. Scoping of a project
	2.14. Elicit requirements
	2.15. Estimation of requirements
	2.16. Document requirements
	2.17. Prioritization of requirements
	2.18. Define code quality indicators
	2.19. Incorporation of unit testing
	2.20. Code Development
	2.21. Quality Validation
	2.22. Team project customization
	2.23. Implementation of an automated deployment process
	2.24. Implementation of pre-production environment
	2.25. Manage feedback between development and operations
	2.26. Troubleshoot production issues
	2.27. Planning the application layers
	2.28. Designing a distributed application
	2.29. Designing and implementing the Windows Azure life cycle
	2.30. Configure state management
	2.31. Designing a caching strategy
	2.32. Designing and implementing a Web Socket strategy
	2.33. Designing HTTP modules and handlers
	2.34. Apply the user interface design for a web application
	2.35. Designing and implementing UI behavior
	2.36. Compose the UI layout of an application

	2.37. Enhance application behavior and style based on browser feature detection
	2.38. Plan an adaptive UI layout
	2.39. Plan for search engine optimization and accessibility
	2.40. Plan and implement globalization and localization
	2.41. Design and implement MVC controllers and actions
	2.42. Design and implement routes
	2.43. Control application behavior by using MVC controllers and actions
	2.44. Design and implement routes
	2.45. Control application behavior by using MVC extensibility points
	2.46. Reduce network bandwidth
	2.47. Prevent and troubleshoot runtime issues
	2.48. Design an exception handling strategy
	2.49. Debug a Windows Azure application
	2.50. Configure authentication
	2.51. Configure and apply authorization
	2.52. Design and implement claims-based authentication across federated identity stores
	2.53. Manage data integrity
	2.54. Implement a secure site with ASP.NET
	2.55. Implement multithreading and asynchronous processing
	2.56. Manage multithreading
	2.57. Implement program flow
	2.58. Create and implement events and callbacks
	2.59. Implement exception handling
	2.60. Create types
	2.61. Consume types
	2.62. Enforce encapsulation
	2.63. Create and implement a class hierarchy
	2.64. Find, execute & create types at runtime by using reflection
	2.65. Manage the object life cycle
	2.66. Manipulate strings
	2.67. Validate application input
	2.68. Perform symmetric and asymmetric encryption
	2.69. Manage assemblies
	2.70. Debug an application
	2.71. Implement diagnostics in an application
	2.72. Perform I/O operations
	2.73. Consume data
	2.74. Query and manipulate data and objects by using LINQ
	2.75. Serialize and deserialize data
	2.76. Store data in and retrieve data from collections
3. Underpinning Skills	3.1. Interaction diagram modeling.
	3.2. Detailed/ step-by-step process modeling
	3.3. Documentation
	3.4. Using IDE
	3.5. Testing and debugging ASP.NET MVC Web Applications
	3.6. Tracing
	3.7. Developing ASP.NET web applications
	3.8. Creating an installation package
	3.9. Deploying and configuring installation package

4. Resource Implications	The following resources **MUST** be provided:
	4.1. Access to PC work station and requisite software
	4.2. Access to .NET development tools and hardware
	4.3. Office 365 account
	4.4. Materials relevant to the proposed activity or tasks
	4.5. Internet connection (for setting up and familiarization with software sources)
5. Methods of Assessment	Competency may be assessed through: 5.1. Demonstration with questioning 5.2. Observation with questioning 5.3. Third party report 5.4. Portfolio 5.5. Deployment of "Developing ASP.NET MVC5 (or higher) Web Applications" certification exam
6. Context for Assessment	6.1. Competency may be assessed in workplace or in a simulated workplace setting 6.2. Assessment shall be observed while task are being undertaken whether individually or in group

Reference: Contents of Core Competencies are derived from the contents of the Certification Exams from Microsoft Learning < http://www.microsoft.com/learning>.

SECTION 3   TRAINING STANDARDS

These guidelines are set to provide the Technical and Vocational Education and Training (TVET) providers with information and other important requirements to consider when designing training programs for **Programming (.NET Technology) NC III**.

3.1   CURRICULUM DESIGN

Course Title: Programming (.NET Technology)
NC Level: NC III

Nominal Training Duration: 64 hrs. – Basic Competencies
16 hrs. – Common Competencies
120 hrs. – Core Competencies (*6hours x 20 workdays*)
352 hrs. – S.I.T. (*Supervised-Industry Training*)
----
552 hrs. – Total Training Duration

Course Description:

This course is designed to develop & enhance the knowledge, skills, & attitudes of a programmer in accordance with industry standards. It covers the basic, common & core competencies on programming language, specifically .NET technology. The nominal duration of **552** hours covers the required units at Programming (.NET Technology) NC III. TVET providers can however, offer a longer, ladderized course covering the Programming (.NET Technology) NC III basic, common and core competency unit/s.

BASIC COMPETENCIES
64 hrs

Unit of Competency	Learning Outcomes	Methodology	Assessment Approach
1. Lead workplace communication	1.1 Communicate information about workplace processes 1.2 Lead workplace discussions 1.3 Identify & communicate issues arising in the workplace	• Group discussion • Role Play • Brainstorming	• Observation • Interviews
2. Lead small teams	2.1 Provide team leadership 2.2 Assign responsibilities among members 2.3 Set performance expectation for team members 2.4 Supervise team performance	• Lecture • Demonstration • Self-paced (modular)	• Demonstration • Case studies

Unit of Competency	Learning Outcomes	Methodology	Assessment Approach
3. Develop and practice negotiation skills	3.1 Identify relevant information in planning negotiations 3.2 Participate in negotiations 3.3 Document areas for agreement	• Direct observation • Simulation/role playing • Case studies	• Written test • Practical/ performance test
4. Solve workplace problem related to work activities	4.1 Explain the analytical techniques. 4.2 Identify the problem. 4.3 Determine the possible cause/s of the problem.	• Direct observation • Simulation/role playing • Case studies	• Written test • Practical/ performance test
5. Use mathematical concepts and techniques	5.1 Identify mathematical tools and techniques to solve problem 5.2 Apply mathematical procedures/solution 5.3 Analyze results	• Direct observation • Simulation/role playing • Case studies	• Written test • Practical/ performance test
6. Use relevant technologies	6.1 Identify appropriate technology 6.2 Apply relevant technology 6.3 Maintain/enhance relevant technology	• Direct observation • Simulation/role playing • Case studies	• Written test • Practical/ performance test

COMMON COMPETENCIES
16 hrs.

Unit of Competency	Learning Outcomes	Methodology	Assessment Approach
1. Apply Quality Standards	1.1 Asses quality of received materials 1.2 Assess own work 1.3 Engage in quality improvement	▪ Field trip ▪ Symposium ▪ Film showing ▪ Simulation ▪ On the job training	▪ Demonstration & questioning ▪ Observation & questioning ▪ Third party report
2. Operate a Personal Computer	2.1 Plan and prepare for task to be undertaken 2.2 Input data into computer 2.3 Access information using computer 2.4 Produce output/data using computer system 2.5 Use basic functions of a web browser to locate information 2.6 Maintain computer equipment and systems	• Modular • Film showing • Computer based training (e-learning) • Project method • On the job training	• Demonstration & questioning • Observation & questioning • Third party report • Assessment of output product • Portfolio • Computer-based assessment

CORE COMPETENCIES
472 hrs. (120 hrs. in-school + 352 hrs. S.I.T.*)

Unit of Competency	Learning Outcome	Methodology	Assessment Approach
1. Perform Programming in HTML5 with JavaScript and CSS3	1.1. Develop basic HTML document using HTMP5 and CSS 3 syntax 1.2. Create HTML5 document using advanced techniques with JavaScript and CSS3 1.3. Communicate with Remote Data Source and Create Objects and Methods using JavaScript 1.4. Create interactive and responsive user interface 1.5. Add offline support to Web Applications 1.6. Implement an Adaptive User Interface 1.7. Use CSS3 to create scalable graphics and animated user interface 1.8. Implement real-time communications by using web-sockets and web worker process 1.9. Design and implement Windows Store applications using MVVM design pattern	• Lecture/ Discussion • Hands on • Exercises • Demonstration	• Written exam • Practical exam • Observation in workplace • Interviews/ questioning
2. Develop ASP.NET MVC5 Web Applications	2.1. Employ basic project management tasks 2.2. Perform Object-oriented programming with C# 2.3. Develop ASP.NET Web Applications using Model View Controller or architectural patterns 2.4. Test and debug ASP.NET MVC Web Applications 2.5. Create custom URLs and apply style to ASP.NET MVC Web applications 2.6. Create responsive web pages and implementing security 2.7. Create RESTful API, modify URLs and deploy ASP.NET MVC applications	• Lecture/ Discussion • Hands on • Exercises • Demonstration	• Written exam • Practical exam • Observation in workplace • Interviews/ questioning

*Note: Supervised-Industry Training (SIT) in actual work environment is included in the training duration.

3.1. TRAINING DELIVERY

The delivery of training should adhere to the design of the curriculum. Delivery should be guided by the 10 basic principles of the competency-based TVET.

- The training is based on curriculum developed from the competency standards;

- Learning is modular in its structure;

- Training delivery is individualized and self-paced;

- Training is based on work that must be performed;

- Training materials are directly related to the competency standards and the curriculum modules;

- Assessment is based in the collection of evidence of the performance of work to the industry required standard;

- Training is based both on and off-the-job components;

- Allows for recognition of prior learning (RPL) or current competencies;

- Training allows for multiple entry and exit; and

- Approved training programs are nationally accredited.

The competency-based TVET system recognizes various types of delivery modes, both on and off-the-job as long as the learning is driven by the competency standards specified by the industry. The following training modalities may be adopted when designing training programs:

- The dualized mode of training delivery is preferred and recommended. Thus programs would contain both in-school and in-industry training or fieldwork components. Details can be referred to the Dual Training System (DTS) Implementing Rules and Regulations.

- Modular/self-paced learning is a competency-based training modality wherein the trainee is allowed to progress at his own pace. The trainer facilities the training delivery.

- Peer teaching/mentoring is a training modality wherein fast learners are given the opportunity to assist the slow learners.

- Supervised industry training or on-the-job training is an approach in training designed to enhance the knowledge and skills of the trainee through actual experience in the workplace to acquire a specific competencies prescribed in the training regulations.

- Distance learning is a formal education process in which majority of the instruction occurs when the students and instructors are not in the same place. Distance learning may employ correspondence study, or audio, video or computer technologies.

3.2. TRAINEE ENTRY REQUIREMENTS

Trainees or students should possess the following requirements:
- Must have completed at least 10 yrs. basic education or an ALS grade 10 certificate of rating holder
- Can communicate either oral or written;
- With basic computer skills

This list does not include specific institutional requirements such as educational attainment, appropriate work experience, and others that may be required of the trainees by the school or training center delivering the TVET program.

3.3. LIST OF TOOLS, EQUIPMENT AND MATERIALS

Recommended list of tools, equipment and materials for the conduct of training in **Programming (.NET Technology) NC III**:
(Computed/estimated based on a class size of 15)

\ TOOLS		\ EQUIPMENT		MATERIALS/RESOURCES	
Qty	Description	Qty	Description	Qty	Description
As required	Computer Software *e.g. Visual Studio Express 2012 or higher*	15	Network Computer (on Windows 8 or higher) with peripherals	1 free account	www.microsoftvirtual academy.com
		1	Network Printer		Practice materials
1	Internet access	1	White board		Hand-outs
As required	MVC5 or higher	1	LCD projector and screen		Reference books
1	Application servers *e.g.-SQL Server Express 2012 or higher* *- IIS Express*	15	Ergonomic chairs and tables		Learning materials/guide

The quantity of tools and equipment to be used for the conduct of training for this qualification shall depend on the number of students, size of the class, and/or modality of training. The most important consideration is to make sure that tools and equipment are adequately provided to all trainees when needed. The actual list of tools, equipment, machines, supplies and other materials to be used shall be identified and detailed in the Competency Based Curriculum (CBC) to be submitted by the TVET provider when registering a course or training program with TESDA.

Due to the fast-changing nature of the Information and Communications Technology (ICT) sector, TVET providers are reminded to use and provide their trainees with the latest technology tools, equipment and materials where appropriate and applicable.

In cases where there are specialized tools, equipment and facilities that are not generally considered standard requirements or not absolute requisites for training, the industry working group or TESDA may provide guidelines or specific advice on such matters.

3.4. TRAINING FACILITIES

The Programming workshop must be of concrete structure. Based on class size of 25 student/trainees the space requirements for the teaching/learning and circulation areas are as follows:

TEACHING/LEARNING AREAS	SIZE IN METERS	AREA IN SQ. METERS	QTY	TOTAL AREA IN SQ. METERS
Lecture Area	5 x 8	40	1	40
Computer laboratory	6 x 8	48	1	48
Learning Resource Area	4 x 5	20	1	20
Wash,Toilet & Locker Room	2 x 5	10	2	20
Total				128
Facilities / Equipment / Circulation**				38
Total Area				**166**

*** Area requirement is equivalent to 30% of the total teaching/learning areas*

3.5. TRAINERS QUALIFICATION

Programming (.NET Technology) NC III

- Must be a holder of National TVET Trainer's Certificate Level 1(NTTC)
  - Must be a holder of Programming NC IV or equivalent
- Must be a holder of a current Microsoft Developer Certificate
- Must have at least 2-years relevant industry experience*

 * *Optional: Only when required by the hiring institution.*

3.6. INSTITUTIONAL ASSESSMENT

Institutional Assessment is undertaken by trainees to determine their achievement of units of competency. A certificate of achievement is issued for each unit of competency. The institutional assessment is administered by the trainer/assessor.

The result of the institutional assessment may be considered as an evidence for national assessment.

SECTION 4   NATIONAL ASSESSMENT AND CERTIFICATION ARRANGEMENTS

4.1   To attain the National Qualification of **Programming (.NET Technology) NC III**, the candidate must demonstrate competency in all the units listed in Section 1. Successful candidates shall be awarded a **National Certificate level III** signed by the TESDA Director General.

4.2   The qualification **Programming (.NET Technology) NC III** may be attained through accumulation of Microsoft Examinations, specifically relevant to the following employable competencies:

      4.2.1 Perform Programming in HTML5 with JavaScript and CSS3

      4.2.2 Develop ASP.NET MVC5 (or higher) Web Applications

      Upon accumulation and submission of all Microsoft certificates acquired for all the relevant units of competency comprising this qualification, an individual shall be issued the corresponding **National Certificate**.

4.3   Assessment shall focus on the core units of competency. The basic and common units shall be integrated or assessed concurrently with the core units.

4.4   The following are qualified to apply for assessment and certification:

    4.4.1.   Graduate of formal, non-formal, and informal, including enterprise-based, training programs.

    4.4.2.   Experienced workers (wage employed or self-employed)

4.5   The guidelines on assessment and certification are discussed in detail in the "Procedures Manual on Assessment and Certification" and "Guidelines on the Implementation of the Philippine TVET Qualification and Certification System (PTQCS)".

COMPETENCY MAP – INFORMATION AND COMMUNICATIONS TECHNOLOGY (ICT) SECTOR

BASIC COMPETENCIES

Receive and Respond to workplace communication	Work with Others	Demonstrate Work Values	Participate in Workplace Communication	Work in a Team Environment	Practice Career Professionalism	Practice occupational Health and Safety Procedures
Practice Housekeeping Procedures (5S)	**Lead Workplace Communication**	**Lead Small Team**	**Develop and Practice Negotiation Skills**	**Solve Problems Related to Work Activities**	**Use Mathematical Concepts and Techniques**	**Use Relevant Technologies**
Utilize Specialized Communication skills	Develop Team and Individual	Apply Problem Solving Techniques in the Workplace	Collect, Analyze and Organize Information	Plan and Organize Work	Promote Environmental Protection	

COMMON COMPETENCIES

Perform Computer Operation	**Apply Quality Standards**

Programming (.Net Technology) NC III

CORE COMPETENCIES

Communicate Effectively using the English Language	Deliver Quality Customer Service	Communicate Effectively in a Customer Contact Center	Utilize Enterprise/ Company Technology	Conduct Contact Center Campaign	Provide Specialized Support and Assistance to Customers	Manage the Activities of a Contact Center Work Team
Demonstrate Understanding and Knowledge of the American Culture and Geography	Perform Basic Computer Operation and Internet Navigation	Lead a Contact Center Work Team	Use Business Technology	Use Medical Terminology to Carry Out Task	Produce Text from Audio Transcription	Produce Cleaned-up and In-betweened Drawings
Review / Edit Documents	Manage the Activities of a Work Team	Lead a Team in Delivering Quality Service	Produce Over-all Designs for Animation	Create 2D Models and Images	Produce 2D Colored Animation	Produce Key Drawings for Animation
Produce Background Designs	Composit and Edit Animation Sequence	Produce Storyboard for Animation	Create 3D Models and Images	Coordinate the Production of Animation	Use email and search the web using browsers	Install Computer Systems and Networks
Configure Computer Systems and Networks	Diagnose and Troubleshoot Computer Systems and Networks	Maintain Computer Systems and Networks	Operate a word-processing application	Operate a spreadsheet application	Operate a presentation package	Install and maintain a server
Design program logic	Apply program development approach	Apply object-oriented program language skills	Apply programming skills in a second language	Install network hardware to a network	Install software to networked computers	Install and configure a network
Determine and confirm client business expectations and needs	Create a simple mark-up language document to specification	Design a website to meet technical requirements	Transfer content to a website using commercial packages	Determine and apply appropriate development methodologies	Ensure website content meets technical protocols and standards	Build a database
Use structured query language to create database structures & manipulate data	Develop detailed technical design	Act on and complete change requests	Determine and act on client computing problems	Provide one -to - one instruction	Provide first-level remote help desk support	Design pages using a page layout application
Develop design studies	Create vector graphics using a graphics application	Create raster graphics using a graphics application	Develop designs for print media	Develop designs for electronic media	Develop designs for product packaging	Design booth & product/window display
Formulate program logic	Plan development of application features	Process/Manage data in a relational database management system using SQL	Apply .NET programming skills	Develop Windows form applications	Develop web applications	Develop service oriented applications
Develop command-line/ console and desktop applications using Java technology	Develop enterprise/web applications using Java technology	Develop applications using COBOL or similar language	Develop applications using Oracle forms	Develop applications using Oracle reports	**Perform Programming in HTML5 with JavaScript and CSS3**	**Develop ASP.NET MVC 5 (or higher) Web Applications**

DEFINITION OF TERMS

GENERAL

1) **Basic Competencies -** are the skills and knowledge that everyone needs for work

2) **Certification -** is the process of verifying and validating the competencies of a person through assessment

3) **Certificate of Competency (COC)** – is a certification issued to individuals who pass the assessment for a single unit or cluster of units of competency

4) **Common Competencies** - are the skills and knowledge needed by all people working in a particular industry

5) **Competency** - is the possession and application of knowledge, skills and attitudes to perform work activities to the standard expected in the workplace

6) **Competency Assessment -** is the process of collecting evidence and making judgments on whether competency has been achieved

7) **Competency Standard (CS)** - is the industry-determined specification of competencies required for effective work performance

8) **Context of Assessment** - refers to the place where assessment is to be conducted or carried out

9) **Core Competencies - are** the specific skills and knowledge needed in a particular area of work - industry sector/occupation/job role

10) **Critical aspects of competency -** refers to the evidence that is essential for successful performance of the unit of competency

11) **Elective Competencies -** are the additional skills and knowledge required by the individual or enterprise for work

12) **Elements** - are the building blocks of a unit of competency. They describe in outcome terms the functions that a person perform in the workplace

13) **Evidence Guide** - is a component of the unit of competency that defines or identifies the evidences required to determine the competence of the individual. It provides information on critical aspects of competency, underpinning knowledge, underpinning skills, resource implications, assessment method and context of assessment

14) **Level -** refers to the category of skills and knowledge required to do a job

15) **Method of Assessment** - refers to the ways of collecting evidence and when evidence should be collected

16) **National Certificate (NC)** – is a certification issued to individuals who achieve all the required units of competency for a national qualification defined under the Training Regulations. NCs are aligned to specific levels within the PTQF

17) **Performance Criteria** - are evaluative statements that specify what is to be assessed and the required level of performance

18) **Qualification** - is a cluster of units of competencies that meets job roles and is significant in the workplace. It is also a certification awarded to a person on successful completion of a course in recognition of having demonstrated competencies in an industry sector

19) **Range of Variables** - describes the circumstances or context in which the work is to be performed

20) **Recognition of Prior Learning (RPL)** – is the acknowledgement of an individual's skills, knowledge and attitudes gained from life and work experiences outside registered training programs

21) **Resource Implications -** refers to the resources needed for the successful performance of the work activity described in the unit of competency. It includes work environment and conditions, materials, tools and equipment

22) **Training Regulations (TR)** – refers to the document promulgated and issued by TESDA consisting of competency standards, national qualifications and training guidelines for specific sectors/occupations. The TR serves as basis for establishment of qualification and certification under the PTQF. It also serves as guide for development of competency-based curricula and instructional materials including registration of TVET programs offered by TVET providers

23) **Underpinning Knowledge -** refers to the competency that involves in applying knowledge to perform work activities. It includes specific knowledge that is essential to the performance of the competency

24) **Underpinning Skills** - refers to the list of the skills needed to achieve the elements and performance criteria in the unit of competency. It includes generic and industry specific skills

25) **Unit of Competency** – is a component of the competency standards stating a specific key function or role in a particular job or occupation; it is the smallest component of achievement that can be assessed and certified under the PTQF

SECTOR SPECIFIC

1. **ADO.NET** - is a set of computer software components that programmers can use to access data and data services. It is a part of the base class library that is included with the Microsoft .NET Framework. It is commonly used by programmers to access and modify data stored in relational database systems, though it can also access data in non-relational sources. ADO.NET is sometimes considered an evolution of ActiveX Data Objects (ADO) technology, but was changed so extensively that it can be considered an entirely new product.

2. **ASP.NET** - is a web application framework developed and marketed by Microsoft to allow programmers to build dynamic web sites, web applications and web services. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language.

3. **Algorithm** - is a type of effective method in which a list of well-defined instructions for completing a task will, when given an initial state, proceed through a well-defined series of successive states, eventually terminating in an end-state. The transition from one state to the next is not necessarily deterministic; some algorithms, known as probabilistic algorithms, incorporate randomness.

4. **Artificial intelligence programmer** - develops the logic the game uses to carry out a large number of actions. An AI programmer may program pathfinding, strategy and enemy tactic systems. This is one of the most challenging aspects of game programming and its sophistication is developing rapidly.

5. **Browser** – a software package that provides the user interface for accessing Internet, intranet and extranet Web sites.

6. **COBOL** - is one of the oldest programming languages. Its name is an acronym for **CO**mmon **B**usiness-**O**riented **L**anguage, defining its primary domain in business, finance, and administrative systems for companies and governments.

7. **Compiler** - is a computer program (or set of programs) that transforms source code written in a programming language (the source language) into another computer language (the target language, often having a binary form known as object code). The most common reason for wanting to transform source code is to create an executable program.

8. **Computer** – a device that has the ability to accept data; internally store and execute a program of instructions; perform mathematical, logical, and manipulative operations on data; and report the results.

9. **Computer program** - also a software program, or just a program, is a sequence of instructions written to perform a specified task for a computer. A computer requires programs to function, typically executing the program's instructions in a central processor. The program has an executable form that the computer can use directly to execute the instructions. The same program in its human-readable source code form, from which executable programs are derived (e.g., compiled), enables a programmer to study and develop its algorithms.

10. **Computer programming** - is the iterative process of writing or editing source code. Editing source code involves testing, analyzing, and refining, and sometimes coordinating with other programmers on a jointly developed program. A person who practices this skill is referred to as a computer programmer, software developer or coder. The sometimes lengthy process of computer programming is usually referred to as software development.

11. **Computer Terminal** – any input/output device connected by telecommunications links to a computer.

12. **C language** - is an imperative (procedural) systems implementation language. It was designed to be compiled using a relatively straightforward compiler, to provide low-level access to memory, to provide language constructs that map efficiently to machine instructions, and to require minimal run-time support. C was therefore useful for many applications that had formerly been coded in assembly language.

13. **C++ language** - is a statically typed, free-form, multi-paradigm, compiled, general-purpose programming language. It comprises a combination of both high-level and low-level language features. Some of its application domains include systems software, application software, device drivers, embedded software, high-performance server and client applications, and entertainment software such as video games.

14. **Data** - objective measurements of the attributes (characteristics) of entities such as people, places, things, and events.

15. **Data access** - typically refers to software and activities related to storing, retrieving, or acting on data housed in a database or other repository. Historically, different methods and languages were required for every repository, including each different database, file system, etc., and many of these repositories stored their content in different and incompatible formats.

16. **Decompiler** - is the name given to a computer program that performs the reverse operation to that of a compiler. That is, it translates a file containing information at a relatively low level of abstraction (usually designed to be computer readable rather than human readable) into a form having a higher level of abstraction (usually designed to be human readable)

17. **Documentation** – a collection of documents or information.

18. **Edit** – to modify the form or format of data

19. **End user –** anyone who uses an information system or the information it produces.

20. **Ergonomics** - the science and technology emphasizing the safety, comfort, and ease of use of human-operated machines. The goal of ergonomics is to produce systems that are user-friendly: safe, comfortable and easy to use.

21. **HTML**, which stands for HyperText Markup Language - is the predominant markup language for web pages. It is written in the form of HTML elements consisting of "tags" surrounded by angle brackets within the web page content. It is the building blocks of all basic websites.

22. **Information** – data placed in a meaningful and useful context for an end user.

23. **Information and Communication Technology (ICT) -** refers to technologies associated with the transmission and exchange of data in the form of sound, text, visual images, signals or any combination of those forms through the use of digital technology. It encompasses such services as telecommunications, posts, multimedia, electronic commerce, broadcasting, and information technology.

24. **Integrated development environment (IDE)** - is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of a source code editor, a compiler and/or interpreter, build automation tools, and (usually) a debugger. Typically an IDE is dedicated to a specific programming language, so as to provide a feature set which most closely matches the programming paradigms of the language. However, some multiple-language IDEs are in use, such as Eclipse, ActiveState Komodo, recent versions of NetBeans, and Microsoft Visual Studio.

25. **Java** - is a general-purpose, concurrent, class-based, object-oriented language that is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere". Java is currently one of the most popular programming languages in use, and is widely used from application software to web applications

26. **Java applications** - are typically compiled to bytecode (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture.

27. **Local Area Network (LAN) –** a communications network that typically connects computers, terminals, and other computerized devices within a limited physical area such as an office, building, manufacturing plant and other work sites.

28. **Microsoft .NET Framework** - is a software framework that can be installed on computers running Microsoft Windows operating systems. It includes a large library of coded solutions to common programming problems and a common language infrastructure that manages the execution of programs written specifically for the framework. The .NET Framework supports multiple programming languages in a manner that allows language interoperability, whereby each language can utilize code written in other languages; in particular, the .NET library is available to all the programming languages that .NET encompasses.

29. **Microsoft SQL Server** - is a relational model database server produced by Microsoft. Its primary query languages are T-SQL and ANSI SQL

30. **Object code** - or an object file, is the representation of code that a compiler or assembler generates by processing a source code file. Object files contain compact code, often called "binaries". A linker is typically used to generate an executable or library by linking object files together. The only essential element in an object file is machine code (code directly executed by a computer's CPU). Object files for embedded systems might contain nothing but machine code. However, object files often also contain data for use by the code at runtime, relocation information, program symbols (names of variables and functions) for linking and/or debugging purposes, and other debugging information.

31. **Oracle Forms** - is a software product for creating screens that interact with an Oracle database. It has a typical IDE including an object navigator, property sheet and code editor that uses PL/SQL. It was originally developed to run server-side in character mode terminal sessions. It was ported to other platforms, including Windows, to function in a client–server environment. Later versions were ported to Java where it runs in a Java EE container and can integrate with Java and web services. The primary focus of Forms is to create data entry systems that access an Oracle database.

32. **Oracle Reports** - is a tool for developing reports against data stored in an Oracle database. Oracle Reports consists of Oracle Reports Developer (a component of the Oracle Developer Suite) and Oracle Application Server Reports Services (a component of the Oracle Application Server).

33. **Outsourcing –** turning over all or part of an organization's information systems operation to outside contractors, known as systems integrators or facilities management companies.

34. **Programming language** - is an artificial language designed to express computations that can be performed by a machine, particularly a computer. Programming languages can be used to create programs that control the behavior of a machine, to express algorithms precisely, or as a mode of human communication.

35. **Program Evaluation and Review Technique (PERT)** – a statistical tool used in project management, that is designed to analyze and represent the tasks involved in completing a given project.

36. **Quality Assurance –** methods for ensuring that information systems are free from errors and fraud and provide information products of high quality.

37. **Relational database management system (RDBMS)** - is a database management system (DBMS) that is based on the relational model as introduced by E. F. Codd. Most popular commercial and open source databases currently in use are based on the relational database model. A short definition of an RDBMS may be a DBMS in which data is stored in the form of tables and the relationship among the data is also stored in the form of tables.

38. **Service-oriented programming (SOP)** - is a programming paradigm that uses "services" as the unit of computer work, to design and implement integrated business applications and mission critical software programs. Services can represent steps of business processes and thus one of the main applications of this paradigm is the cost-effective delivery of standalone or composite business applications that can "integrate from the inside-out."

39. **Software –** computer programs and procedures concerned with the operation of an information system.

40. **Source code** - is any collection of statements or declarations written in some human-readable computer programming language. Source code is the means most often used by programmers to specify the actions to be performed by a computer.

41. **SQL,** often referred to as **Structured Query Language** - is a database computer language designed for managing data in relational database management systems (RDBMS), and originally based upon relational algebra. Its scope includes data insert, query, update and delete, schema creation and modification, and data access control.

42. **Standards –** measures of performance developed to evaluate the progress of a system toward its objectives

43. **System –** an assembly of methods, procedures, or techniques unified by regulated interaction to form an organized whole

44. **User- friendly** – a characteristic of human-operated equipment and systems that makes them safe, comfortable, and easy to use.

45. **User interface** - is the system by which people (users) interact with a machine. The user interface includes hardware (physical) and software (logical) components. User interfaces exist for various systems, and provide a means of: 1) Input, allowing the users to manipulate a system, and/or 2) Output, allowing the system to indicate the effects of the users' manipulation.

46. **VB.NET** – is a redesigned, object-oriented dialect of Visual Basic.

47. **Web application** - is an application that is accessed over a network such as the Internet or an intranet. The term may also mean a computer software application that is hosted in a browser-controlled environment (e.g. a Java applet) or coded in a browser-supported language (such as JavaScript, combined with a browser-rendered markup language like HTML) and reliant on a common web browser to render the application executable.

48. **Windows Forms** - is the name given to the graphical application programming interface (API) included as a part of Microsoft's .NET Framework, providing access to the native Microsoft Windows interface elements by wrapping the existing Windows API in managed code. While it is seen as a replacement for the earlier and more complex C++ based Microsoft Foundation Class Library, it does not offer a paradigm comparable to model–view–controller.

49. **Windows Forms application** - is an event-driven application supported by Microsoft's .NET Framework. Unlike a batch program, it spends most of its time simply waiting for the user to do something, such as fill in a text box or click a button.

50. **Work Breakdown Structure (WBS)** - is a deliverable oriented decomposition of a project into smaller components.

ACKNOWLEDGEMENTS

The Technical Education and Skills Development Authority (TESDA) wishes to extend thanks and appreciation to the many representatives of business, industry, academe and government agencies who donated their time and expertise to the development and validation of these Training Regulations.

- **THE TECHNICAL EXPERTS PANEL (TEP)**

Mr. Christopher Lim	- Accenture
Mr. Sotero Saberon, Jr.	- Blast Asia, Incorporated
Mr. Allan Spartacus Mangune	- Taughtworks, Inc.
Mr. Edison Tan	- Hubnob TV
Mr. Timothy Tuason	- Green Light Training Consulting & Performance Development
Mr. Tommy C. Lim, Jr.	- Orange & Bronze Software Laboratories
Mr. Calen Legaspi	- Orange & Bronze Software Laboratories
Mr. Kristofferson Arao	- SQL Wizard
Ms. Grace Miralles	- Computer Professionals, Inc.
Ms. Joie Villanueva	- Philippine Software Industries Association of the Philippines

- **THE PARTICIPANTS IN THE NATIONAL VALIDATION OF THIS TRAINING REGULATION**

  - Philippine Software Industries Association of the Philippines Members
  - Philippine Society of IT Educators (PSITE)
    - John Ruero (John.Ruero@macquarie.com)
  - HR-IT
    - Penny Bongato (penny.bongato@logica.com)

- **THE TESDA BOARD - STANDARDS SETTING AND SYSTEMS DEVELOPMENT COMMITTEE**

- **THE MANAGEMENT AND STAFF OF THE TESDA SECRETARIAT**

  - **Qualifications and Standards Office (QSO)**
    - Competency Standards Division

Mr. Zoilo C. Galang	- CSD-QSO-TESDA
Mr. Samuel E. Calado Jr.	- CSD-QSO-TESDA
Mr. Stephen I. Cezar	- CSD-QSO-TESDA
Mr. Venzel Y. Concoles	- CSD-QSO-TESDA

    - Curriculum and Training Aids Division

Mr. Arsenio A. Mateo, Jr.	- CTAD-QSO-TESDA