Model Curriculum

QF Name: Web Technologies

QF Version: 1.0

NSQF Level: 4.5

Model Curriculum Version: 1.0

IT-ITeS Sector Skill Council || IT-ITeS Sector Skill Council, NASSCOM, Plot No - 7, 8, 9 & 10, 3rd Floor,
Sector 126, Noida
Uttar Pradesh – 201303

Table of Contents

Contents

Training Parameters

Sector	IT-ITeS
Sub-Sector	Future Skills
Occupation	Web and Mobile
Country	India
NSQF Level	4.5
Aligned to NCO/ISCO/ISIC Code	NCO-2015/ NIL
Minimum Educational Qualification and Experience	➢ Completed 1st year of 3-year/ 4-years UG OR ➢ Pursuing 1st year of 3-year/ 4-years UG and continuing education OR ➢ Previous relevant Qualification of NSQF Level 4 with 1.5 years of relevant experience
Pre-Requisite License or Training	--
Minimum Job Entry Age	19 Years
Last Reviewed On	TBD
Next Review Date	TBD
NSQC Approval Date	TBD
QP Version	1.0
Model Curriculum Creation Date	TBD
Model Curriculum Valid Up to Date	TBD
Model Curriculum Version	1.0
Minimum Duration of the Course	510 hours
Maximum Duration of the Course	510 hours

Program Overview

This section summarizes the end objectives of the program along with its duration.

Training Outcomes

At the end of the program, the learner should have acquired the listed knowledge and skills.

- Define business requirements to derive the user experienceneeds of the customer
- Apply various approaches to demonstrate 21st century employability skills communication skills and interpersonal skills
- Define the user experience design process by explaining the user experience research process, empathy mapping processand user behavior process
- Define user persona and behavior patterns by collecting user behavior data, on-boarding unbiased study groups, classifyingusers into groups and defining specific user persona
- Develop wireframes and illustrations of the product using standard tools and methodologies
- Construct secure front-end web applications that meet the functional, non-functional and user experience requirements of the application
- Define metrics, tools, frameworks, and APIs to be used for web development.
- Apply suitable frameworks and tools for web application architecture.
- Develop secure and scalable back-endtechnology stack for different web and mobile based applications
- Explain how to develop a style guide, a design system and userinterface specifications for developers and designers
- Develop a user experience map to define the user experiencejourney and user flow using standard practices and templates
- Identify, report and fix software bugs using best practices
- Implement the principles for continuous delivery, continuousintegration and continuous deployment in the software development process
- Empower team members by motivating them, demonstrating confidence in their abilities, and setting appropriate performance goals
- Plan and prioritize content operations

Compulsory Modules

The table lists the modules and their duration corresponding to the Compulsory NOS of the QP.

NOS and Module Details	Theory Duration (In Hours)	Practical Duration(In Hours)	On-the-Job Training Duration (Mandatory)	On-the-Job Training Duration (Recommended)	Total Duration (In Hours)
SSC/N8440 : Work Organization and Management for Web Technologies **NSQF Level 4.5** **NOS Version 1**	10:00	20:00	00:00	00:00	30:00

ww

NOS and Module Details	Theory Duration (In Hours)	Practical Duration(In Hours)	On-the-Job Training Duration (Mandatory)	On-the-Job Training Duration (Recommended)	Total Duration (In Hours)
SSC/N8441: Communication and interpersonal skills for Web technologies **NSQF Level 4.5** **NOS Version 1**	**10:00**	**20:00**	**00:00**	**00:00**	**30:00**
SSC/N8442: Design Implementation **NSQF Level 4.5** **NOS Version 1**	**60:00**	**90:00**	**00:00**	**00:00**	**150:00**
Module 1: Style guide, design systems and specifications	10:00	`10:00	00:00	00:00	20:00
Module 2: User experience map, journey and user flow	10:00	10:00	00:00	00:00	20:00
Module 3: User Experience Design Process	10:00	10:00	00:00	00:00	20:00
Module 4: User persona and behavior patterns	10:00	20:00	00:00	00:00	30:00
Module 5: Wireframes and illustrations	10:00	20:00	00:00	00:00	30:00
Module 6: Evaluation of tools, APIs and frameworks	10:00	20:00	00:00	00:00	30:00
SSC/N8443: Front End web Development **NSQF Level 4.5** **NOS Version 1**	**60:00**	**90:00**	**00:00**	**00:00**	**150:00**
Module 7: Front-end Web Development	60:00	90:00	00:00	00:00	150:00
SSC/N8444: Back End web Development **NSQF Level 4.5** **NOS Version 1**	**60:00**	**90:00**	**00:00**	**00:00**	**150:00**
Module 8: Back-end Engineering	30:00	30:00	00:00	00:00	60:00
Module 9: Bugs fixing and performance improvement	10:00	20:00	00:00	00:00	30:00
Module 10: Continuous Integration, Delivery and Deployment	20:00	40:00	00:00	00:00	60:00
Total Duration	**200.00**	**310.00**	**00.00**	**00.00**	**510.00**

SSC/N8440: Work Organization and Management for Web Technologies

Duration: *10:00(In Hours)*	**Duration:** *20:00(In Hours)*
Key Learning Outcomes- Theory	**Key Learning Outcomes- Practical**

• Troubleshoot common web design and development problems. • Work within specified time limitations and deadlines • Use a computer with a range of software packages. • Apply research techniques and skills to keep up to date with industry best practices. • Apply deployment optimization, such as page loading, with industry best practices. • Ensure the work is completed according to a given schedule. • Include linked images, fonts, native files, and production file format when archiving. • Use software version control systems such as git.	• Demonstrate management skills in a work organization.
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	
Labs equipped with the following: PCs/Laptops Internet with Wi-Fi (Min 2 Mbps Dedicated) Microphone / voice system for lecture and class activities Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools	

SSC/N8442: Communication and interpersonal skills for Web Technologies

Duration: *10:00(In Hours)*	**Duration:** *20:00(In Hours)*
Key Learning Outcomes- Theory	**Key Learning Outcomes- Practical**
• Read and understand specifications documents • Read and use provided source code of front-end and back-end • technologies • Deliver products that respond to client requirements and • specification • Gather, analyse, and evaluate information • Interpret standards and requirements • Match client requirements • Present concepts to meet business requirements	Demonstrate communication skills to meet business requirements.
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	

Labs equipped with the following:
PCs/Laptops
Internet with Wi-Fi (Min 2 Mbps Dedicated)
Microphone / voice system for lecture and class activities
Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools

Module Details

Module 1: Style guide, design systems and specifications

Mapped to NOS SSC/N8442, v1.0

Terminal Outcomes:

- Explain how to develop a style guide, a design system and userinterface specifications for developers and designers

<table>
<tr><td><b>Duration:</b> <i>10:00(In Hours)</i></td><td><b>Duration:</b> <i>10:00(In Hours)</i></td></tr>
<tr><td><b>Theory – Key Learning Outcomes</b></td><td><b>Practical – Key Learning Outcomes</b></td></tr>
<tr><td>• Explain how to assess the business of the client
• State the principles of typography
• Examine different types of design elements in a user interface (such as button, scroll bar, text box etc.)
• Discuss principles of spacing, padding and placement of design elements for an aesthetically pleasing user interface
• Discuss principles of color combination, shades and specification</td><td>• Develop typography for different text elements (such as titles, subtitle, heading, etc.)
• Create samples to showcase proposed typography, color palette and placement of design elements
• Design guidelines for developing different user interface elements (such as Icons, toolbars, dialog box etc.)
• Document the design guidelines and the rationale for the proposed design system</td></tr>
<tr><td colspan="2"><b>Classroom Aids:</b></td></tr>
<tr><td colspan="2">Whiteboard and Markers
Chart paper and sketch
pens
LCD Projector and Laptop for presentations</td></tr>
<tr><td colspan="2"><b>Tools, Equipment and Other Requirements:</b></td></tr>
<tr><td colspan="2">Labs equipped with the following:
PCs/Laptops
Internet with Wi-Fi (Min 2 Mbps Dedicated)
Microphone / voice system for lecture and class activities
Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools
Prototype Design tools: Sketch, Zeplin, Figma, FramerX, Mockflow, Balsamiq, Flinto, RapidUI etc.
Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.</td></tr>
</table>

Module 2: User experience map, journey and user flow

Mapped to NOS SSC/N8442, v1.0

Terminal Outcomes:

- Develop a user experience map to define the user experiencejourney and user flow using standard practices and templates

Duration: *10:00(In Hours)*	**Duration**: *10:00(In Hours)*
Theory – Key Learning Outcomes	**Practical – Key Learning Outcomes**
• Define the scope of the user experience map • Outline the requirements of different user persona for the software solution • Discuss standard practices and templates to map and sketch a user journey	• Demonstrate how to identify user touch points and stages of engagement • Exhibit ways to achieve multiple goals of a user while effectively capturing different complexities (such as multiple users, media etc.)
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	
Labs equipped with the following: PCs/Laptops Internet with Wi-Fi (Min 2 Mbps Dedicated) Microphone / voice system for lecture and class activities Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools Prototype Design tools: Sketch, Zeplin, Figma, FramerX, Mockflow, Balsamiq, Flinto, RapidUI etc. Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.	

Module 3: User Experience Design Process

Mapped to NOS SSC/N8442, v1.0

Terminal Outcomes:

- Define the user experience design process by explaining the user experience research process, empathy mapping processand user behavior process

<table>
<tr><td><b>Duration</b>: <i>10:00(In Hours)</i></td><td><b>Duration</b>: <i>10:00(In Hours)</i></td></tr>
<tr><td><b>Theory – Key Learning Outcomes</b></td><td><b>Practical – Key Learning Outcomes</b></td></tr>
<tr><td>• Discuss what is a user experience design process
• Examine different types of user experience research (such as qualitative research, quantitative research etc.) and their methodologies (such as interviews, surveys, questionnaires, observations etc.)
• Explain the empathy mapping process
• Work on standard processes to understand user behavior
• Define personas of users
• Explain the process of mapping user journey and user flow</td><td>• Demonstrate how to develop frameworks to support user experience design process
• Build prototypes through sketching, visualization etc.</td></tr>
<tr><td colspan="2"><b>Classroom Aids:</b></td></tr>
<tr><td colspan="2">Whiteboard and Markers
Chart paper and sketch pens
LCD Projector and Laptop for presentations</td></tr>
<tr><td colspan="2"><b>Tools, Equipment and Other Requirements:</b></td></tr>
<tr><td colspan="2">Labs equipped with the following:
PCs/Laptops
Internet with Wi-Fi (Min 2 Mbps Dedicated)
Microphone / voice system for lecture and class activities
Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools
Prototype Design tools: Sketch, Zeplin, Figma, FramerX, Mockflow, Balsamiq, Flinto, RapidUI</td></tr>
</table>

Module 4: User Persona and Behavior Patterns

Mapped to NOS SSC/N8442, v1.0

Terminal Outcomes:

- Define user persona and behavior patterns by collecting user behavior data, on-boarding unbiased study groups, classifyingusers into groups and defining specific user persona

<table>
<tr><td><b>Duration</b>: <i>10:00(In Hours)</i></td><td><b>Duration</b>: <i>20:00(In Hours)</i></td></tr>
<tr><td><b>Theory – Key Learning Outcomes</b></td><td><b>Practical – Key Learning Outcomes</b></td></tr>
<tr><td>• Collect data on user behavior
• Discuss how to identify and recruit an unbiased study group based on both demographic and psychographic factors
• Discuss how to ensure the integrity of the user data collected
• Evaluate interpret the gathered user behavior data
• Explain how to identify patterns and repeatable metrics to tag important insights and problems
• Discuss how to classify users based on the user behavior data collected
• Explain how to define a suitable identify, name and persona for a user based on different parameters (age, education, income, likes etc.)
• Discuss how to ensure that the user persona appropriately represent a larger set of users</td><td>• Demonstrate how to gather user data (such as questionnaires, face-to-face interviews, group discussions etc.)
• Demonstrate how the user persona can be suitably leveraged to develop a user experience solution
• demonstrate how to optimize function design by leveraging the developed user persona</td></tr>
<tr><td colspan="2"><b>Classroom Aids:</b></td></tr>
<tr><td colspan="2">Whiteboard and Markers
Chart paper and sketch
pens
LCD Projector and Laptop for presentations</td></tr>
<tr><td colspan="2"><b>Tools, Equipment and Other Requirements:</b></td></tr>
<tr><td colspan="2">Labs equipped with the following:
PCs/Laptops
Internet with Wi-Fi (Min 2 Mbps Dedicated)
Microphone / voice system for lecture and class activities
Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools
Prototype Design tools: Sketch, Zeplin, Figma, FramerX, Mockflow, Balsamiq, Flinto, RapidUI etc.
Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.</td></tr>
</table>

Module 5: Wireframes and illustrations

Mapped to NOS SSC/N8442, v1.0

Terminal Outcomes:

- Develop wireframes and illustrations of the product usingstandard tools and methodologies

<table>
<tr><td><b>Duration</b>: <i>10:00(In Hours)</i></td><td><b>Duration</b>: <i>20:00(In Hours)</i></td></tr>
<tr><td><b>Theory – Key Learning Outcomes</b></td><td><b>Practical – Key Learning Outcomes</b></td></tr>
<tr><td>• Explain what are wireframes and why are they required
• Discuss the methodology to develop clear and logical wireframes for any software
• solution
• Describe elements illustrated in a wireframe (such as user flow, interactions, screen size
• etc.)
• Explain how to identify the intentions of the customer
• Discuss how to evaluate goals of the customer against a smoother user experience
• Evaluate a wireframe prototype to demonstrate a clean and smooth user experience journey
• Discuss how to continuously incorporate feedback from the customer</td><td>• Demonstrate ways to gather user feedback
• Demonstrate the methodologies for developing a wireframe prototype
• Demonstrate a usability test
• Leverage analytics and other standard tools to refine customer journey.</td></tr>
<tr><td colspan="2"><b>Classroom Aids:</b></td></tr>
<tr><td colspan="2">Whiteboard and Markers
Chart paper and sketch pens
LCD Projector and Laptop for presentations</td></tr>
<tr><td colspan="2"><b>Tools, Equipment and Other Requirements:</b></td></tr>
<tr><td colspan="2">Labs equipped with the following:
Microphone / voice system for lecture and class activities
Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser, Outlook / Any other Email Client, and chat tools
Prototype Design tools: Sketch, Zeplin, Figma, FramerX, Mockflow, Balsamiq, Flinto, RapidUI etc.
Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.</td></tr>
</table>

Module 6: Evaluation of tools, APIs and frameworks

Mapped to NOS SSC/N8442, v1.0

Terminal Outcomes:

- Define metrics, tools, frameworks and APIs to be used for web development.

Duration: *10:00(In Hours)*	**Duration**: *20:00(In Hours)*
Theory – Key Learning Outcomes	**Practical – Key Learning Outcomes**
• Explain how to define KPIs and evaluate tools, frameworks and APIs for web development • Define what business continuity is and how is it affect by third-party tools, frameworks and APIs	• Demonstrate how to measure long-term dependencies on third-party tools, APIs and frameworks • Demonstrate how to evaluate how secure third- party tools, APIs and frameworks are • Assess the performance and SLA compliance against business goals
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	
Labs equipped with the following: Microphone / voice system for lecture and class activities Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools Architecture design tools: Cloudkraft, Gliffy, Microsoft Visio, SmartDraw etc. Application monitoring tools: AppDynamics, Retrace etc. Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.	

Module 7: Front End Web development

Mapped to NOS SSC/N8443, v1.0

Terminal Outcomes:

- Construct secure front-end web applications that meet the functional, non-functional and user experience requirements of the application

Duration: *60:00(In Hours)*	**Duration:** *90:00(In Hours)*
Theory – Key Learning Outcomes	**Practical – Key Learning Outcomes**
• Comprehend the scope of any application • Discuss what are functional and non-functional requirements • Discuss what are user- experience requirements • Explain the front-end components of any web application - HTML, CSS and JavaScript • Discuss common security controls implemented to secure a web-site	• Exhibit essential components of a web page (such as header, menu, footer etc.) • Demonstrate how to build static web pages using HTML and CSS • Demonstrate how to construct interactive web pages using JavaScript • Demonstrate how to develop responsive web- sites that can adjust to any screen size • Demonstrate how to create single page websites using standard web frameworks (such as Angular, Ember etc.) • Demonstrate how to build re-usable web UI components • Build prototypes using standards web builder tools • Build test cases to check the web application for bugs before launch • Run unit tests on different modules of the web site • Demonstrate how to automate testing using standard tools (such as selenium, Appium etc.)
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	
Labs equipped with the following: Microphone / voice system for lecture and class activities Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools Integrated Development Environments: Eclipse, Netbeans, Visual Studio, Atom etc. Application monitoring tools: Amazon Cloudwatch, Microsoft cloud monitoring, AppDynamics, Retrace etc. CI/CD tools: Jenkins, TravisCI, GitLab etc. Configuration management tools: Puppet, Chef, Ansible, CFEngine, JUJU, Bamboo Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.	

Module 8: Back-end Engineering

Mapped to NOS SSC/N8444, v1.0

Terminal Outcomes:

- Develop secure and scalable back-endtechnology stack for different web and mobile based applications.

Duration: *30:00(In Hours)*	**Duration:** *30:00(In Hours)*
Theory – Key Learning Outcomes	**Practical – Key Learning Outcomes**
• Explain what back-end engineering is and how to demarcate between font-endand back-end responsibilities • Discuss how to identify thescope of back-end operations and functionalities • Explain what is caching anddiscuss popular caching solutions • Explain what message queues are and why are they used • Document the functionalities of backend APIs • Explain how to build scalable and reliable back-end systems • Discuss how to scale applications horizontally using auto-scaling and load balancing solutions • Explain how to manage security configurations of back-end applications • Discuss the concepts of Identity and Access Management (IAM) • Examine how to test back- end functionalist using scripts	• Design and develop server end-points to connect back-end servers with the client • Develop stubs for upstreamand backstream • List the different types of back-end application dependencies (such as Databases, Caching, Messaging Queues, Webservices, HTTP APIs etc.) • Demonstrate how to mapapplication dependencies • Create databases usingdifferent Database Management Systems (DBMS) • Demonstrate how to integrate database management systems withdifferent sub-systems • Demonstrate how to create micro-services • Demonstrate how to create reusable RESTful and secure APIs • Demonstrate how to deploy back-end systems on cloud platforms (such as AWS, Azure etc.) • Demonstrate how to encrypt data in transit and data at rest •
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	
Labs equipped with the following: Microphone / voice system for lecture and class activities Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools Integrated Development Environments: Eclipse, Netbeans, Visual Studio, Atom etc. Application monitoring tools: Amazon Cloudwatch, Microsoft cloud monitoring, AppDynamics, Retrace etc. CI/CD tools: Jenkins, TravisCI, GitLab etc. Configuration management tools: Puppet, Chef, Ansible, CFEngine, JUJU, Bamboo Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.	

Module 9: Bugs fixing and performance improvement

Mapped to NOS SSC/N8444, v1.0

Terminal Outcomes:

- Identify, report and fix software bugs using best practices

Duration: *10:00(In Hours)*	**Duration:** *20:00(In Hours)*
Theory – Key Learning Outcomes	**Practical – Key Learning Outcomes**
• Explain how to identify and record software bugs • Explain different types of bugs (such as unexpected,null, bad input etc.) • Discuss how to continuously iterate and develop software code freeof any bugs • Log all activities of theapplication	• Demonstrate best practicesfor logging bugs in a case tracking system • Demonstrate how to identifyuser behavior prior to bug identification • Demonstrate how to analyze and isolate portionof source code where the bug occurs • Demonstrate how build unittest cases to identify and isolate software bugs • Analyze abnormal systembehavior using applicationlog
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	
Labs equipped with the following: Microphone / voice system for lecture and class activities Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools Integrated Development Environments: Eclipse, Netbeans, Visual Studio, Atom etc.	

Module 10: Continuous Integration, Delivery and Deployment

Mapped to NOS SSC/N8444, v1.0

Terminal Outcomes:

- Implement the principles for continuous delivery, continuousintegration and continuous deployment in the software development process

Duration: *20:00(In Hours)*	**Duration:** *40:00(In Hours)*
Theory – Key Learning Outcomes	**Practical – Key Learning Outcomes**
• Discuss the principles of continuous integration, continuous delivery and continuous deployment • Explain what version control is • Discuss how to secure the source code repository • Examine and evaluate standard practices for code tagging, branching, merger and integration • Explain different types of application environment variables and how to manage the configurations of target environments • Explain how to continuously integrate bugs fixes in the application builds • Discuss best practices for application deployment	• Demonstrate how to manage changes to source code using standard version control tools • Demonstrate how to integrate version control systems with the deployed project management tools • Demonstrate how to automate application testing using standard tools and scripts • Demonstrate how to write test cases to indent failure • Demonstrate how to push applications to their appropriate services (such as web servers, API services, database services etc.) • Demonstrate how to automate the CI/CD (continuous integration/continuous delivery) pipeline using standard automation tools (such as Chef, Bamboo etc.)
Classroom Aids:	
Whiteboard and Markers Chart paper and sketch pens LCD Projector and Laptop for presentations	
Tools, Equipment and Other Requirements:	
Labs equipped with the following: Microphone / voice system for lecture and class activities Computer Lab with 1:1 PC: trainee ratio and having internet connection, MS Office / Open office, Browser,Outlook / Any other Email Client, and chat tools Integrated Development Environments: Eclipse, Netbeans, Visual Studio, Atom etc. Application monitoring tools: Amazon Cloudwatch, Microsoft cloud monitoring, AppDynamics, Retrace etc. CI/CD tools: Jenkins, TravisCI, GitLab etc. Configuration management tools: Puppet, Chef, Ansible, CFEngine, JUJU, Bamboo Workflow management tools: Evernote, Jira, VersionOne, Workzone, Scrum Mate etc.	

Annexure

Trainer Requirements

1.	**Trainer's Qualification and experience in the relevant sector (in years)** *(as per NCVET guidelines)*	Graduate in Engineering/Technology/ Statistics/ Mathematics/Computer Science/Physical Sciences with Minimum 5 years of relevant experience and 2 years of full-time training experience in programing languages such as HTML, CSS, JavaScript etc.
2.	**Master Trainer's Qualification and experience in the relevant sector (in years)** *(as per NCVET guidelines)*	Graduate in Engineering/Technology/ Statistics/ Mathematics/Computer Science/Physical Sciences with Minimum 5 years of relevant experience and 2 years of full-time training experience in relevant field programing languages such as HTML, CSS, JavaScript etc.
3.	**Tools and Equipment Required for the Training**	☒Yes ☐No *(If "Yes", details to be provided in Annexure)*
4.	**In Case of Revised Qualification, details of Any Upskilling Required for Trainer**	NA

Assessor Requirements

1.	**Assessor's Qualification and experience in relevant sector (in years)** *(as per NCVET guidelines)*	Graduate in Engineering/Technology/ Statistics/ Mathematics/Computer Science/Physical Sciences with Minimum 5 years of relevant experience and 2 years of full-time training experience in programing languages such as HTML, CSS, JavaScript etc.
2.	**Proctor's Qualification and experience in relevant sector (in years)** *(as per NCVET guidelines), (wherever applicable)*	Graduate in Engineering/Technology/ Statistics/ Mathematics/Computer Science/Physical Sciences with Minimum 5 years of relevant experience and 2 years of full-time training experience in relevant field programing languages such as HTML, CSS, JavaScript etc.
3.	**Lead Assessor's/Proctor's Qualification and experience in relevant sector (in years)** *(as per NCVET guidelines)*	Graduate in Engineering/Technology/ Statistics/ Mathematics/Computer Science/Physical Sciences with a Minimum 5 years of relevant experience and 2 years of full-time training experience in programming languages such as HTML, CSS, JavaScript etc.
4.	**Assessment Mode** *(Specify the assessment mode)*	Can be either in the classroom or online
5.	**Tools and Equipment Required for Assessment**	☒ Same as for training ☐ Yes ☐ No *(details to be provided in Annexure-if it is different for Assessment)*

Assessment Strategy

This section includes the processes involved in identifying, gathering, and interpreting information to evaluate the learner on the required competencies of the program.

Assessment System Overview

A uniform assessment of job candidates as per industry standards facilitates progress of the industry by filtering employable individuals while simultaneously providing candidates with an analysis of personal strengths and weaknesses.

Assessment Criteria

Criteria for assessment for each Qualification Pack will be created by the Sector Skill Council. Each Performance Criteria (PC) will be assigned marks proportional to its importance in NOS. SSC will also lay down the proportion of marks for Theory and Skills Practical for each PC.

The assessment for the theory part will be based on a knowledge bank of questions created by the SSC. Assessment will be conducted for all compulsory NOS, and where applicable, on the selected elective/option NOS/set of NOS.

Guidelines for Assessment			
Testing Environment	**Tasks and Functions**	**Productivity**	**Teamwork**
• Carry out assessments under realistic work pressures that are found in the normal industry workplace (or simulated workplace). • Ensure that the range of materials, equipment, and tools that learners use are current and of the type routinely found in the normal industry workplace (or simulated workplace) environments.	• Assess that all tasks and functions are completed in a way, and to a timescale, that is acceptable in the normal industry workplace. • Assign workplace (or simulated workplace) responsibilities that enable learners to meet the requirements of the NOS.	• Productivity levels must be checked to ensure that it reflects those that are found in the work situation being replicated.	• Provide situations that allow learners to interact with the range of personnel and contractors found in the normal industry workplace (or simulated workplace).

Assessment Quality Assurance framework

NASSCOM provides two assessment frameworks NAC and NAC-Tech.

NAC (NASSCOM Assessment of Competence)

NAC follows a test matrix to assess Speaking & Listening, Analytical, Quantitative, Writing, and Keyboard skills of candidates appearing for assessment.

NAC-Tech

NAC-Tech test matrix includes assessment of Communication, Reading, Analytical, Logical Reasoning, Work Management, Computer Fundamentals, Operating Systems, RDBMS, SDLC, Algorithms & Programming Fundamentals, and System Architecture skills.

Methods of Validation

To pass a QF, a trainee should score a minimum aggregate of 70% across qualification. In case of unsuccessful completion, the trainee may seek reassessment on the Qualification Pack

Method of assessment documentation and access

The assessment agency will upload the result of assessment in the portal. The data will not be accessible for change by the assessment agency after the upload. The assessment data will be validated by SSC assessment team. After upload, only SSC can access this data.

References

Glossary

Term	Description
Key Learning Outcome	Key learning outcome is the statement of what a learner needs to know, understand and be able to do to achieve the terminal outcomes. A set of key learning outcomes will make up the training outcomes. Training outcome is specified in terms of knowledge, understanding (theory) and skills (practical application).
Training Outcome	Training outcome is a statement of what a learner will know, understand and be able to do **upon the completion of the training**.
Terminal Outcome	Terminal outcome is a statement of what a learner will know, understand and be able to do **upon the completion of a module.** A set of terminal outcomes help to achieve the training outcome.
National Occupational Standard	National Occupational Standard specify the standard of performance an individual must achieve when carrying out a function in the workplace
Persons With Disability	Persons with Disability are those who have long-term physical, mental, intellectual, or sensory impairments which in interaction with various barriers may hinder their full and effective participation in society on an equal basis with others.
Integrated Development Environment	An integrated development environment is a software application that provides comprehensive facilities to computer programmers for software development.

Acronyms and Abbreviations

Term	Description
QP	Qualification Pack
NSQF	National Skills Qualification Framework
NSQC	National Skills Qualification Committee
NOS	National Occupational Standards
SSC	Skill Sectors Councils
NASSCOM	National Association of Software & Service Companies
PwD	Persons with Disability
IDE	Integrated Development Environment