



Appendix A

WORK PROCESS SCHEDULES AND RELATED INSTRUCTION OUTLINE

IONnovate, LLC

**SOFTWARE DEVELOPER
(Existing Title: APPLICATION DEVELOPER)**

O*NET-SOC CODE: 15-1252.00 RAPIDS CODE: 1129CB

**APPROVED BY
THE NEVADA LABOR COMMISSIONER AND THE NEVADA STATE APPRENTICESHIP COUNCIL**

Richard J. Williams, Nevada State Apprenticeship Director

REGISTRATION DATE: _____

RAPIDS PROGRAM ID NUMBER: _____

**DEVELOPED IN COOPERATION WITH THE
THE NEVADA LABOR COMMISSIONER, THE NEVADA STATE APPRENTICESHIP COUNCIL AND
THE U.S. DEPARTMENT OF LABOR**

Appendix A

WORK PROCESS SCHEDULE

This schedule is attached to and a part of these Standards for the above identified occupation.

1. TYPE OF OCCUPATION

☐ Time-based ☒ Competency-based ☐ Hybrid

2. TERM OF APPRENTICESHIP

The term of the occupation shall be defined by the attainment of all competencies of the position. 1) If the program uses a time-based approach, requires the completion of not less than 2,000 hours of [work experience,] on-the-job learning, consistent with training requirements as established by practice in the trade; (2) If the program uses a competency-based approach, specifies the skills that must be demonstrated by an apprentice and addresses how on-the-job learning will be integrated into the program; or (3) If the program uses a hybrid approach, specifies the skills that must be acquired and the minimum number of hours of on-the-job learning that must be completed by an apprentice.

This competency-based approach would be expected to occur within approximately 1250 hours of OJL, supplemented by a minimum of 500 hours of related apprenticeship instruction per year

3. RATIO OF APPRENTICES TO JOURNEYWORKERS

The apprentice to journey worker/fully trained worker ratio is: 5 apprentice(s) to 1 journey worker/fully trained worker(s).

4. APPRENTICE WAGE SCHEDULE

An apprentice minimum starting wage will be at least \$16 per hour. Apprentices shall be paid a progressively increasing schedule of wages based on either a percentage or a dollar amount of the current hourly journey worker/fully trained worker wage. A journey worker/fully trained worker minimum wage will be at least \$25.

Wages by Period:

1st = \$16

2nd = \$20

3rd = \$25

Periodic review and evaluation of the apprentice's on-the-job learning and related technical instruction will be conducted in alignment with the wage schedule established.

5. WORK PROCESS SCHEDULE (See attached Work Process Schedule)

The sponsor may modify the work processes to meet local needs prior to submitting these Standards to the appropriate Registration Agency for approval.

6. RELATED INSTRUCTION OUTLINE (See attached Related Instruction Outline)

The sponsor may modify the related instruction to meet local needs prior to submitting these Standards to the appropriate Registration Agency for approval.

Appendix A

WORK PROCESS SCHEDULE

The term of the occupation shall be defined by the attainment of all competencies, both technical and behavioral, of the position, which would be expected and approximated to occur within 1250 hours of OJL supplemented by a minimum of 500 hours of related instruction per year of apprenticeship.

Apprenticeship Competencies – Technical

Item	Work Processes	Approx. Hours
A	Participates in and supports the creation of product, platform, and/or software development life cycles by assisting Principal Developer and team with initial determination of applicable specifications, requirements, systems and concepts to produce the desired output.	250
B	Supports the Principal Developer and team with mapping out requirement specifications, communicates with other key team members.	250
C	Participates in and supports designing software or platform with the appropriate team.	500
D	Supports the development and assembly of the software, platform, or product	750
E	Supports testing and debugging; participates in integration and deployment.	750
F		
G		
H		
I		
J		
K		
L		
M		
N		
O		
P		
Q		
	Total hours (approximate)	2,500

The above on-the-job-learning (OJL) work process competencies are intended as a guide. It need not be followed in any sequence, and it is understood that some adjustments may be necessary in the hours allotted for different work experience. In all cases, the apprentice is to receive sufficient experience to make them fully competent and use good workmanship in all work processes, which are a part of the industry. In addition, the apprentice shall be fully instructed in safety and OSHA requirements.

Apprenticeship Competencies – Behavioral

In addition to mastering all the essential technical competencies, an apprentice must consistently demonstrate at an acceptable level the following behavioral competencies, to complete the apprenticeship.

Item #	Behavioral Competencies
1.	Participates in the implementation of development life cycles and specifications, including supporting of mapping out work plans, under supervision.
2.	Supports the creation of tools and languages needed per output in the work plan, under supervision.
3.	Participates in supporting Principal Developer to establish overall project goals with senior and other key team members primarily, as well as any external project members, as applicable.
4.	Supports project team members with team consultations that determine end product's feasibility in economic, operational, and technical areas.
5.	Assists with offering and applying technical and cost-effective approaches for mitigating risks at the end result, under direction from Principal Developer; and communicates any potential questions or concerns based on preliminary assessments.
6.	Coordinates with the Project Manager to communicate desired requirements and objectives clearly to other team members such as the User Experience (UX) team, Quality Assurance (QA) testers, etc.
7.	Ensures proper use of desired Software Requirement Specification (SRS), and clearly defines and documents the product requirements, under supervision.
8.	Supports the team/ QA testers in preliminary quality assurance requirements assessments & potential risk assessments, under supervision.
9.	Supports the UX team, or appropriate team members, with software design and structure of the software as it relates to implementation, its data models, interfaces between system components, and if applicable, the algorithms used, under supervision.
10.	Participates in the identification and development of the best prototype suited for the project, if any; supports identification of appropriate languages, operating systems, and monitoring methods applicable for the final program.
11.	Applies best practices to the company-specific source code management processes.
12.	Continues to support identifying program and project changes or newfound needs as the software or system is formed, apply such project changes or needs effectively under direction from Principal Developer or applicable team members.
13.	Participates in building the program using the appropriate languages

	and/or applicable development methods.
14.	Participates in recognizing concepts to determine Continuous Integration (CI)/Continuous Delivery (CD) configuration, supports building and applying CI/CD integrations for manual and/or automated functionalities.
15.	Supports application of prototyping methods, if applicable.
16.	Supports close monitoring to identify issues and reports them in a clear and concise way to senior team members.
17.	After issue is reported, clearly tracks and works with team to fix, and re-test until quality standards are met.
18.	Makes extensive, detailed notes when any changes are made and/or to clarify why a function must remain the same in a clear way for other team members.
19.	Participates in curating implementation preparation documents and plans.
20.	Supports integration and test phase, properly notes progress relevant to project success.
21.	Participates in ongoing monitoring of platform, software, or application under project requirements, and supports with providing maintenance, troubleshooting assistance, and applies problem solving capabilities as applicable, under supervision.

RELATED INSTRUCTION OUTLINE

The related instruction has been developed in cooperation with employer-partners as part of the apprenticeship. The following is a set of courses to be delivered by subject matter experts.

Related Technical Instruction (RTI) - This instruction shall include, but not be limited to, at least 500 hours per year for each year of the apprenticeship. The related theoretical education listed below is tightly integrated with real work product. The curriculum is defined as a variety of classes, around which the exams and projects are based. By defining the RTI this way, all competencies required of the students are met, through project work.

COURSE TOPICS	HOURS
A. IT Foundations	104
B. IT Applications	104
C. Data Structures and Algorithms	104
D. Data Management – Applications	104
E. Advanced Data Management	78
F. Organizational Behavior and Leadership	78
G. Web Development Applications	156
H. Software Engineering	104
I. Software Quality Assurance	78
J. Mobile Application Development	78
K. User Experience Design	78
L. User Interface Design	104
M. Software I	156
N. Software II	156
O. Software Development Capstone	104
TOTAL HOURS	1,586

COURSE TOPIC DESCRIPTIONS

- A. IT Foundations is the first course in a two-part series preparatory for the CompTIA A+ exam, Part I. Students will gain an understanding of personal computer components and their functions in a desktop system; computer data storage and retrieval; classifying, installing, configuring, optimizing, upgrading, and troubleshooting printers, laptops, portable devices, operating systems, networks, and system security; recommending appropriate tools, diagnostic procedures, preventative maintenance and troubleshooting techniques for personal computer components in a desktop system; strategies for identifying, preventing, and reporting safety hazards and environmental/human accidents in a technological environment; and effective communication with colleagues and clients as well as job-related professional behavior.

- B. IT Applications is a continuation of the IT Foundations course preparatory for the CompTIA A+ exam, Part II. Students will gain an understanding of personal computer components and their functions in a desktop system. Also covered is computer data storage and retrieval including classifying, installing, configuring, optimizing, upgrading, and troubleshooting printers, laptops, portable devices, operating systems, networks, and system security. Other areas include recommending appropriate tools, diagnostic procedures, preventative maintenance, and troubleshooting techniques for personal computer components in a desktop system. The course then finishes with strategies for identifying, preventing, and reporting safety hazards and environmental/human accidents in a technological environment, and effective communication with colleagues and clients as well as job-related professional behavior. This course builds on the understanding of hardware from IT Foundations and is designed to build the skills to support 4 core components: Operating Systems, Security, Software Troubleshooting, and Operational Procedures. These are core competencies for IT professionals from cloud engineers to data analysts, and will empower you with a better understanding of the tools used during your career.
- C. Data Structures and Algorithms covers the fundamentals of dynamic data structures, such as bags, lists, stacks, queues, trees, hash tables, and their associated algorithms. With Python software as the basis, the course discusses object-oriented design and abstract data types as a design paradigm. The course emphasizes problem solving and techniques for designing efficient, maintainable software applications. Students will implement simple applications using the techniques learned.
- D. This course covers conceptual data modeling and provides an introduction to MySQL. Students will learn how to create simple to complex SELECT queries including subqueries and joins, and students will also learn how to use SQL to update and delete data. Topics covered in this course include exposure to MySQL; developing physical schemas; creating and modifying databases, tables, views, foreign keys/primary keys (FKs/PKs), and indexes; populating tables; and developing simple Select-From-Where (SFW) queries to complex 3+ table join queries.
- E. Advanced Data Management enables organizations to extract and analyze raw data. Skillful data management allows organizations to discover and explore data in ways that uncover trends, issues, and their root causes. In turn, businesses are better equipped to capitalize on opportunities and more accurately plan for the future. As organizations continue to extract larger and more detailed volumes of data, the need is rapidly growing for IT professionals possessing data management skills. These skills include performing advanced relational data modeling as well as designing data marts, lakes, and warehouses. This course will empower software developers with the skills to build business logic at the database layer to employ more stability and higher data-processing speeds. Data analysts will gain the ability to automate common tasks to summarize and integrate data as they prepare it for analysis.
- F. Organizational Behavior and Leadership explores how to lead and manage effectively in diverse business environments. The course requires students to demonstrate the ability to apply organizational leadership theories and management strategies in a series of scenario-based problems.
- G. This course builds upon a student's manual coding skills by teaching how to develop web documents and pages using the web development trifecta: Hypertext Markup Language version 5 (HTML5), Cascading Style Sheets version 3 (CSS3), and JavaScript. Students will utilize the skills learned in this course to create web documents and pages that easily adapt

to display on both traditional and mobile devices. In addition, students will learn techniques for code validation and testing, form creation, inline form field validation, and mobile design for browsers and apps, including Responsive Web Design (RWD).

- H. This course introduces the concepts of software engineering to students who have completed the core courses in programming and project management. The principles build on previously acquired concepts, switching the emphasis from programming simple routines to engineering robust and scalable software solutions. This course does not cover programming, but provides an overview of software engineering processes and their challenging nature, focusing on the need for a disciplined approach to software engineering. A generic process framework provides the groundwork for formal process models. Prescriptive process models such as the waterfall model and agile development are included. An introduction to the elements and phases of software engineering is included, which explores requirements for engineering, design concepts, and software quality.
- I. Software Quality Assurance applies a QA focus to every phase of the software development life cycle. This course investigates best practices for quality analysis, quality planning, and testing strategies as they pertain to the everyday practice of software development. Students will come to understand how their work fits into the bigger picture: how QA, testing, and code-writing practices interact within specific process models; the potential impact of new code on existing code or on other applications; the importance of usability, and the influence users have on the ultimate success of an application. Students will explore test plans, test cases, unit tests, integration tests, regression tests, usability tests, and test and review tools.
- J. This course introduces students to programming for mobile devices using a software development kit (SDK). Students with previous knowledge of programming will learn how to install and utilize a SDK, build a basic mobile application, build a mobile application using a graphical user interface (GUI), adapt applications to different mobile devices, save data, execute and debug mobile applications using emulators, and deploy a mobile application.
- K. User Experience Design explores multiple tools and techniques used in user experience design. Students are presented with an in-depth view of activities involved in the design of user experience and have the opportunity to create several deliverables, including persona profiles, information architectures, and prototypes of different levels of fidelity. In addition, the course also covers usability testing and the evaluation of quantitative and qualitative data derived from these and other experiments.
- L. This course covers tools and techniques employed in user interface design, including web and mobile applications. Concepts of clarity, usability, and detectability are included in this course, as well as other design elements such as color schemes, typography, and layout. Techniques like wireframing, usability testing, and SEO optimization are also covered.
- M. Software I builds object-oriented programming expertise and introduces powerful new tools for Java application development. You will learn about and put into action class design, exception handling, and other object-oriented principles and constructs to develop software that meets business requirements. This course requires foundational knowledge of object-oriented programming and the Java language.
- N. Software II - Advanced Java Concepts refines object-oriented programming expertise and builds database and file server application development skills. You will learn about and put into action lambda expressions, collections, input/output, advanced error handling, and the newest features of Java 8 to develop software that meets business requirements. This course requires intermediate expertise in object-oriented programming and the Java language.

-
- O. The capstone assessment challenges students to demonstrate mastery of all the BSITSW program outcomes. Students will develop a software application to solve a problem of their choice constrained only by the technology requirements provided in the assessment DRF.
-

SECTION 27 - OFFICIAL ADOPTION OF APPRENTICESHIP STANDARDS

IONnovate, LLC hereby adopts these standards of apprenticeship.

Sponsor(s) designate the appropriate person(s) to sign the standards on their behalf.

Joshua Leavitt **Date:** 9/10/2021
Signature of Sponsor (*designee*)

Joshua Leavitt CEO
Type Name & Title