



IT SOFTWARE AND APPLICATION DEVELOPMENT

IT Software and Application Development youth apprentices gain code writing, troubleshooting, and system security skills related to the development of software and applications.

Length of Apprenticeship: One year

COMPETENCIES

Youth apprentices must complete a **total of 6** competencies. **Five** must be from the list below. If necessary, employers can substitute up to **1** competency with another occupationally appropriate skill. That skill should be added to the competency list for assessment. Note that where necessary, skills can be simulated.

Competencies
<ol style="list-style-type: none">1. Troubleshoot application issues2. Create a database3. Test application programming changes4. Gather application software requirements5. Write application code6. Maintain information and system security

REGISTERED APPRENTICESHIP BRIDGING OPPORTUNITIES

Some of the related instruction courses can bridge into the following registered apprenticeship:

- IT Service Desk
- IT Software Developer

POST-SECONDARY PATHWAY OPPORTUNITIES

There are several post-secondary pathway opportunities in this area. Following is partial list.

- Software Development Specialist Technical Diploma
- Software Project Coordinator Technical Diploma
- IT Help Desk Support Specialist Technical Diploma



IT Software and Application Development

Youth Apprenticeship
ON-THE-JOB LEARNING PERFORMANCE STANDARDS GUIDE

YOUTH APPRENTICE INFORMATION

Youth Apprentice Name	
YA Coordinator	YA Consortium
School District	High School Graduation Date

REQUIREMENTS

Requirements

Youth apprentices must complete ALL the items listed below. Check completed areas.

- Competency checklist
- Employability Skills checklist (in this job book) or the DPI Employability Skills Certificate
- Related instruction equal to 1 high school credit or at least 3 college credits
- Minimum of 450 work hours

HOURS

Record the hours the youth apprentice worked.

Total Hours Employed	Company Name	Telephone Number

COMPETENCIES

Youth apprentices must complete a **total of 6** competencies. **Five** must be from the list below. If necessary, employers can substitute up to **1** competency with another occupationally appropriate skill. That skill should be added to the competency list for assessment. Note that where necessary, skills can be simulated.

Rating Scale

3: Exceeds entry level criteria | Requires minimal supervision | Consistently displays this behavior

2: Meets entry level criteria | Requires some supervision | Often displays this behavior

1: Needs improvement | Requires much assistance and supervision | Rarely displays behavior

Competency and Rating Criteria	Minimum Rating of 2 for EACH Check Rating		
	1	2	3
1. Troubleshoot application issues <ul style="list-style-type: none"> • detect application issue • identify criticality of issue • use available diagnostic tools to identify root cause • identify possible issue solutions • test solution to confirm resolution • document troubleshooting results and solutions 	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. Create a database <ul style="list-style-type: none"> • create a database from model specifications • define data architecture • document data architecture • use CRUD (Create Read Update Delete) • populate the database created with test data 	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. Test application programming changes <ul style="list-style-type: none"> • follow defined test procedures • conduct testing based on business and design requirements • conduct unit testing, as required • conduct integration testing, as required • conduct regression testing, as required • conduct user-acceptance testing • document testing results 	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Competency and Rating Criteria	Minimum Rating of 2 for EACH Check Rating		
	1	2	3
<p>4. Gather application software requirements</p> <ul style="list-style-type: none"> • define business problem to be solved by the application • perform workflow analysis • divide design specifications into logical process blocks • identify constraints • identify key functions and subsystems of the application • create flowcharts • assess appropriateness of software solution • create technical requirements documentation 	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p>5. Write application code</p> <ul style="list-style-type: none"> • define logic using a program flowchart • define logic using at least one alternative to flowcharting such as pseudo-coding • review design (e.g., peer and/or user walk-through) • build code • follow secure code practices • execute code • debug code • prepare code documentation • prepare unit testing plan • conduct unit testing and bug fixes • verify code meets project requirements 	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p>6. Maintain information and system security</p> <ul style="list-style-type: none"> • identify the impact of sensitive data exposure • use virus and malware protection tools • use strong passwords • monitor security of company and personal data 	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p>Competency Substitute (if you replaced a competency above, note the competency and rating)</p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p>Comments:</p>			