IT Applications Developer Senior

Purpose of Class: Responsible for all phases of development and maintenance on assigned computer applications; communicates with system users to determine systems needs; researches systems enhancements; performs design and analysis of computer systems; performs programming duties.

Distinguishing Characteristics

Level: Second in a series of three

Work Direction Received: Works under general/limited supervision

Direction of Others: Guides/Oversees; may lead in providing work direction

Scope/nature of Discretion: General discretion; performs duties with moderate opportunity to exercise independence within broadly-defined policies and procedures

Examples of Duties

General Duties
- Consults with clients on application issues
- Uses application development tools and utilities as appropriate
- Assists and mentors peers and subordinates
- Participates on cross-functional teams to represent IT issues and needs

Accomplishes the following tasks and develops quality products of typical scope and complexity with limited assistance:

Systems Analysis
- Business problem definition
- Information analysis and organization
- Information collection
- Economic evaluation
- Solution development and evaluation

Programming and Testing
- Develops, debugs, and assembles/compiles code
- Tests programs
- Documents programs

Logical Database Design (working in conjunction with IT Data/Database Analyst)
- Data analysis
- Logical views
- Data management
- Data structure
- Data definition
Examples of Duties (continued)

Independently accomplishes the following tasks of typical complexity with minimal assistance, in a competent manner:

System Design and Implementation
- Functional design
- Technical design
- System specifications
- Develop implementation plan
- Training
- Support Material
- On-going support

Integrates support software and data management interfaces

Communicates with peers, clients, and management

Maintains regular and reliable attendance.

Physical Database Design (working in conjunction with IT Data/Database Analyst)
- Physical design
- Access methods
- Sub-system facilities
- Relational databases

Minimum Qualifications

Bachelor’s degree in a related area such as computer science, computer engineering, or management information systems, plus two years of experience in programming. Any equivalent education and/or work experience may be substituted in order to meet the minimum qualifications of the position.

Optional: Some positions may require coursework, training or work experience in certain programming languages. Lesser degrees may be considered with increasing years of experience as follows:

<table>
<thead>
<tr>
<th>Degree</th>
<th>Years of Programming Work Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bachelor’s degree in an unrelated area</td>
<td>4-years</td>
</tr>
<tr>
<td>Associate’s degree in a related area</td>
<td>6-years</td>
</tr>
<tr>
<td>Associate’s degree in an unrelated area</td>
<td>8-years</td>
</tr>
<tr>
<td>High school diploma or equivalent</td>
<td>10-years</td>
</tr>
</tbody>
</table>

Knowledge and Abilities

Applicants will be screened through written, oral, performance and/or other evaluation methods for the following:

Knowledge
- Basic math
• Software development principles and methods
• Design techniques, principles, tools and instruments
• Principles, methods and procedures for designing, developing, optimizing and integrating new/reusable systems components
• Software system testing procedures, programming and documentation infrastructure requirements
• Systems engineering concepts and factors

Knowledge (continued)
• Principles and processes involved in business and organizational planning, coordination and execution
• Instructional methods and training techniques
• Principles and processes for providing customer service
• Infrastructure requirements; system analysis
• Logical/physical database design
• Database principles and methodologies

Abilities
• Use basic math to solve problems
• Write various computer programs
• Apply programming languages
• Read, listen, and understand written/oral communication from others
• Communicate, both orally and in writing, information/ideas to others
• Apply general rules to specific problems to arrive at logical conclusions; decide if an answer makes sense; identify discrepancies
• Order information; organize different pieces of information into a meaningful pattern
• Concentrate without distraction while performing tasks over a period of time
• Organize, plan and prioritize work
• Test, install, implement, document, and maintain software
• Maintain source code; modify and upgrade code as necessary
• Actively learn/work with new material/information to grasp its implications
• Write, debug and maintain code
• Integrate hardware/software components
• Determine output media/formats
• Design, code, test, and debug large and complex programs
• Develop creative ways to solve a problem
• Identify or detect a known pattern hidden in other distracting material
• Perceive similarities and differences in pieces of information
• Provide guidance to less experienced co-workers in solving programming problems
• Apply computer assisted software engineering tools to the design and development process
• Analyze needs and product requirements to create a design
• Troubleshoot
• Weigh relative costs and benefits of potential actions
• Platform selection
• Identification of key causes
• Visioning
• Idea evaluation