

Software Development Course

Distance Learning



G-CITI CAMPUS

Changing Lives Through Technology



What is Software Development?

Software development refers to a set of computer science activities dedicated to the process of creating, designing, deploying and supporting software. Software itself is the set of instructions or programs that tell a computer what to do. Also defined as a systematic approach used by the software industry to design, develop, and test high-quality software. The main goal behind SDLC (Software Development Life Cycle) is to produce high-quality software that meets or exceeds customer expectations, reaches completion within time and cost estimates.

Why study Software Development?

Software developers, or software engineers, are now working in practically every industry. As the world becomes increasingly digitized, there is more need for people who can create specific software applications. If you're wondering why you should become a software developer, it will be helpful to know what other benefits this career path offers

Software developers are in high demand. Practically every industry needs its own type of software and people who can create it to their specifications. As the world becomes increasingly digital and operated online, the need for software developers is likely to rise even further. If you're looking for a career that has a good long-term outlook, software development is an appealing option.

Becoming a Software Developer will give you the ability to work in a wide range of industries. For instance, you could build vital healthcare software, a content management system or video games. If you ever want to change work environments or the type of software you produce, you can sometimes switch over to another field using the same skills.

Lastly, there is typically room for advancement within the software development sector. After starting as an entry-level programmer, software developers can advance into senior programming positions. Programmers can often then move into management positions where they oversee the development process rather than spending their time coding

ITS (Information Technology Specialist) Software Development standards are global/International. That means your skills and certifications are recognized anywhere in the world your career takes you. ITS has worked its way up through global corporations in places all over the world.

Why study ITS Software Development at G-CITI Campus?

G-CITI Campus is a South African Accredited and Global / International IT College, that offers both qualifications and certifications in IT Training. With over 10 000 students completing a wide variety of Global / International Certifications, G-CITI Campus has become the leading IT College to obtain credentials in preparation for the 4th Industrial Revolution and the Digital Economy.

IT SPECIALIST EXAM OBJECTIVES

Our innovative approach to learning and skills development will help students go beyond what they have signed up for! Providing a holistic approach to learning and Job Readiness, we ensure a quality level of the services we offer. With its course offerings, G-CITI Campus offers a range of partners, such as CompTIA, Microsoft, Adobe, ITS, Cert Nexus and Cisco, G-CITI Campus offers both online learning subscriptions, exam vouchers and simulation labs to enable students to learn online, conduct digital assessments and conduct vendor certifications online.

Boosting your technical skills by understanding the world of Software Development and how it applies to business today, or the career of your dreams, is one of the many reasons to upskill in this field. All of the modules on the course speak directly to the skills that employers are actively seeking within the Python Development sector.

Studying ITS (Information Technology Specialist) Software Development with G-CITI Campus through Distance Learning, will enable candidates who complete this course and successfully complete the online exam to demonstrate core software development skills, including object-oriented programming, web applications, and databases. Candidates are expected to have some experience with C# and ANSI SQL.

Also, we have a range of ITS Certificates where the networking is part of a family of certifications. See more certifications on our website: www.gciticampus.academy under **Distance Learning**

Potential Career Opportunities in Software Development

- Software tester
- Front-end developer
- Back-end developer
- Software engineer
- Full stack developer
- Development operations engineer

Use the banking Details below for any payment

Bank Name: FNB
Account Name: Genesis CITI
Account Nr: 62563325755
Account Type: Cheque
Branch: Epping
Branch Code: 200810
Deposit Reference: ID Number

COST OF THE COURSE

WAS R 4 990.00 | NOW R 2 950.00

Deposit: R 2000.00

Balance: R 950.00 (To be paid before undertaking final exam)

What is included in the course fees

Please, be aware that all course materials are only offered online. In other words, you will not be receiving any hard copies of the textbook, and you will need to access all the required content through your online classroom, where you'll be able to find the following course content:

- A digital textbook (PDF) focused on all the content you'll need to pass the exam successfully
- Videos and Learning material to ensure you are covered to understand practical concepts
- Practice files OR quizzes to accompany the step-by-step exercises in your textbook
- A PDF summary of everything you've covered in the textbook
- A study guide and exercise files to help you with your exam prep
- A set of mock exams to be covered before undertaking global exams.

This Course Fee includes your Certiport exam voucher.



Software Development

Candidates for this exam are seeking to prove core software development skills, including object-oriented programming, web applications, and databases. Candidates are expected to have some experience with C# and ANSI SQL.

1. Core Programming Concepts

1.1 Describe computer storage and data types

- How a computer stores programs and instructions in computer memory, memory stacks and heaps, memory size requirements for various data storage types, numeric data and textual data, garbage collection

1.2 Construct and analyze algorithms and flowcharts to solve programming problems

- Decision structures used in all computer programming languages; if decision structures; multiple decision structures, such as if...else and switch; reading and constructing flowcharts; decision tables; evaluating expressions; for loops, while loops, do...while loops; recursion

1.3 Incorporate error handling into applications or modules

- Structured exception handling (try-catch-finally), unit testing, throwing exceptions, reading the stack, defensive coding, understand scoping in exception handling

1.4 Construct and analyze code based on functional programming patterns

- Event, delegate, promises, synchronous vs. asynchronous programming (AJAX, XHR), immutability

2. Software Development Principles

2.1 Describe software development lifecycle (SDLC) management

- Requirement analysis, planning and design, implementation, testing, deployment, maintenance; Agile concepts

2.2 Interpret application specifications

- Reading application specifications and translating them into prototypes and code, selecting appropriate application type and components

2.3 Construct and analyze code that uses algorithms and data structures

- Arrays, stacks, queues, linked lists, dictionaries (key-value pairs), sorting algorithms (selection sort, bubble sort, quick sort, merge sort), searching algorithms (linear search, binary search), performance implications of various data structures, choosing the right data structure, FIFO, LIFO

2.4 Describe the purpose of version control systems

- GitHub, check-in, check-out, merge, branch, rollback, clone, resolving conflicts

2.5 Describe secure coding concepts

- Encryption, hashing, and digital signatures; public, private, and shared keys; mitigating cross-site request forgery (csrf); SQL injection; risks of using iframes

IT SPECIALIST EXAM OBJECTIVES

3. Object-Oriented Programming

3.1 Construct, analyze, and use classes

- Properties, methods, events, fields, and constructors; how to create classes; how to use classes in code; access modifiers; instantiation; static vs. instance; encapsulation; composition

3.2 Construct and analyze code that uses inheritance

- Inherit the functionality of a base class into a derived class, generic classes, abstract classes

3.3 Construct and analyze code that uses polymorphism

- Extending the functionality of a class after inheriting from a base class, overriding methods in the derived class, interfaces, overloading

4. Web Applications

4.1 Construct and analyze web applications

- HTML5, CSS3, and JavaScript ES6; browser developer tools; HTTP request or response; state management; cookies, local, and session storage; page lifecycle; event model; client-side vs. server-side programming

4.2 Describe and configure web hosting

- Creating virtual directories and websites, publishing web applications, role of the web server

4.3 Describe and configure web services

- Web services that are consumed by client applications, accessing web services from client applications, JSON, REST API, OAuth, XML

4.4 Describe and identify architectural patterns

- Model-view-controller (MVC), model-view-viewmodel (MVVM), single page application (SPA)

5. Databases

5.1 Design and normalize a database

- Characteristics and capabilities of database products, database design, Entity Relationship Diagrams (ERDs), normalization concepts (to 3NF), indexes, constraints, primary key, foreign keys, relationships

5.2 Construct, analyze, and optimize ANSI SQL queries

- Creating and accessing stored procedures, updating and selecting data, DML vs. DDL, functions, triggers, cursors, joins, indexes

5.3 Manage transactions

- Commit, rollback, save, concurrency, isolation levels, lock

5.4 Describe database access methods

- Entity Framework (Code-first, Database-first), connection pools, LINQ

5.5 Describe types of NoSQL databases

- Document databases, key-value databases

