GenC HR

SPE Java Standard (Delta + Stage 2) - Handbook

Why do we need this Academy Enablement Program?

Academy enablement program engages young talents with a comprehensive learning pathway, giving these millennials an opportunity to interact with Subject Matter Experts (SME) and understand the corporate environment and groom themselves even before they join us.

Cognizant emphasizes on Learner Autonomy where students take charge of their own learning, with the available tools and resources. More focus is on “learning” than “teaching”. Get ready to embark your own learning adventure!

Program at a glance

Learning consisting of 2 Stages and an Integrated Development Project:

- **Stage 1 Delta** - Foundation in Software Development
- **Stage 2** - Application Development and Maintenance Practices
- Integrated Development Project (IDP)

Program Highlights

- The entire learning journey is structured based on adult learning principles, emphasizing problem-solving and practical application of acquired skills over purely conceptual learning.
- Participants will receive mentorship from Subject Matter Experts (SMEs) from the Business Unit (BU), whose motivation and guidance will support and accelerate their learning journey.
- Learner autonomy is fostered through a Flipped Classroom approach, utilizing a world-class learning platform that provides comprehensive resources, freeing learners from the constraints of traditional instructor-led teaching.
- Through Project-Based Learning (PBL), GenCs will experience all phases of a project lifecycle while adopting Agile methodologies and practices.

Service Lines

Service lines can simply be defined as a modern organizational structure strategy for resource planning and allocation for any size of business. Typically, traditional organizational structure models are more vertically aligned -- think of an employee who has several bosses in the hierarchical ladder before being directly under the company's owner or president. Conversely, service lines follow a more horizontal continuum approach, where the company is strategically segmented into more manageable departments. The service line approach tends to focus more on the requirements of customers, which often results in noticeable increases in the customer satisfaction rate.

What is Application Development?

Application development goes through a process of planning, creating, testing, and deploying an information system, also known as the software development lifecycle. Applications are also often developed to automate some type of internal business process or processes, build a product to address common business challenges, or drive innovation.

What is Application Maintenance?

Application maintenance is the continuous updating, analyzing, modifying, and re-evaluating of your existing software applications. Application maintenance must be an ongoing task to ensure your applications are always running to the best of their abilities. Due to evolving customer expectations, the fight to survive in an existing market, and technological advancements, modifying and implementing new strategies is critical in maintaining sustainability and staying competitive. Every competitive business needs to constantly enhance and manage the IT solutions that have been developed in order to stay relevant and meet the wavering needs of users. This is where application maintenance and support come into the picture.

Contrary to popular belief, application maintenance is not just about fixing defects, but modifying a software product after delivery to correct faults, as well as to improve performance. Application maintenance and enhancement to existing applications begin with a thorough study of existing applications to identify areas of improvement.

Tips for Successfully Carrying Out Application Development and Maintenance

Great user experience to end customers through the development and maintenance of modern apps is a must-have. Today, applications (web or mobile) are the most cost-effective and powerful ways to reach out to a vast market and generate revenues. With millions of applications being rolled out every day, it's a good idea to keep in mind a few tips:

- Be as clear as possible as to what your requirements for your application are
- Thoroughly understand the services offered by application development companies and identify the right partner if you're using a partner
- Evaluate the various development platforms and choose the one that best fits the needs of your business
- Make sure to embed processes that focus on continuous improvements and iterations to add new features and/or fix bugs
- When developing your application, make security your top priority

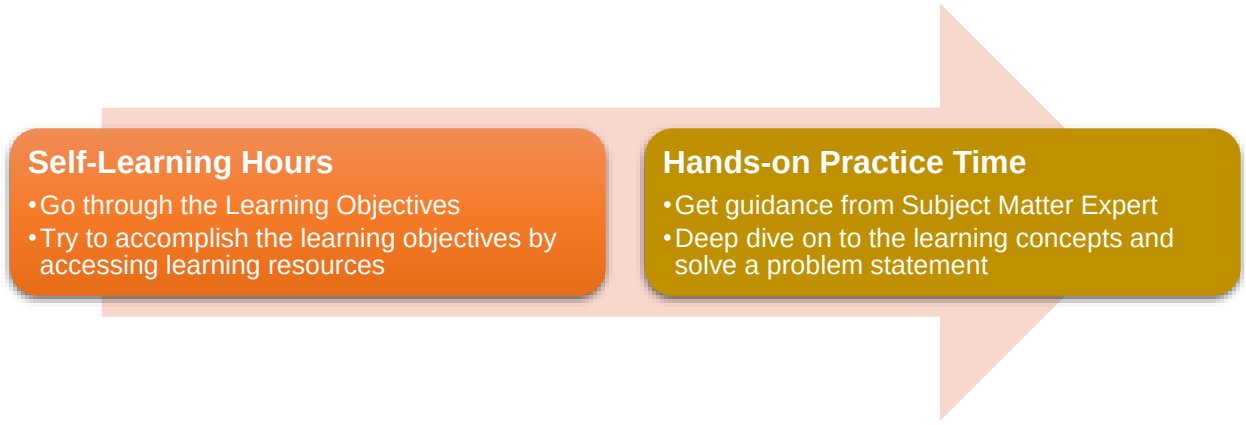
- Regularly update and test your application to deliver improved and better performance, high security, and a bug-free, seamless user experience

Learning Journey with Flipped Classroom

This program encourages you to be more autonomous learners during guided self-learning hours, completing the learning objectives on your own pace and style, and get ready for the hands-on practice time.

The complete learning path is set in the GEN C Learn Platform, which you can login with SSO.

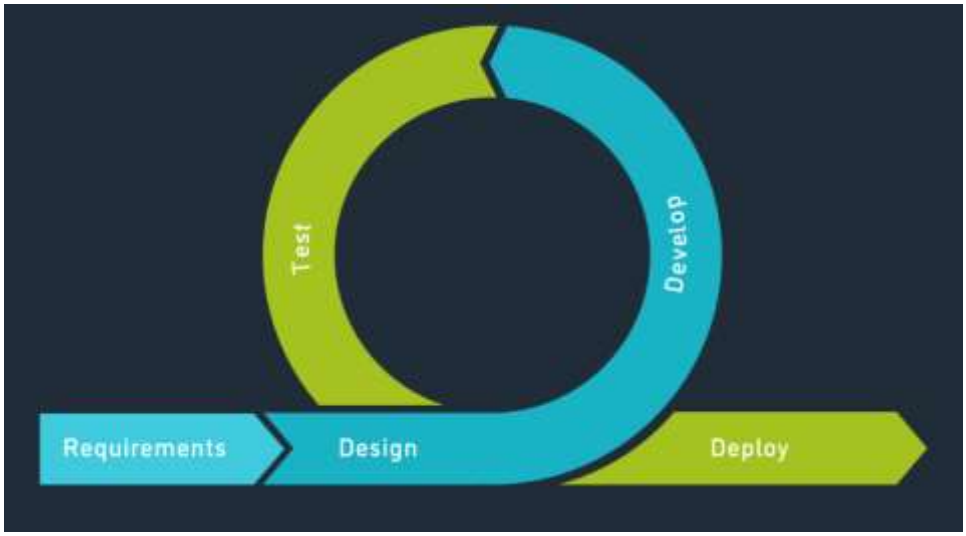
Flipped Classroom



What is Integrated Development Project (IDP)?

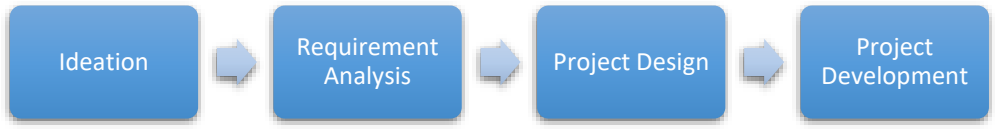
Integrated Development Project is an approach wherein the learner experiences the entire software development processes in an incremental fashion as part of the GenC Training. The IDP implementation is purely based on **Agile Software Development** methodologies and inspired from **PBL (Project-Based Learning)** which is learning while doing. It gives learners the opportunity to gain a deeper understanding of a topic through problem-solving using real-world examples and challenges.

Following is the Agile Development Methodology at high-level.



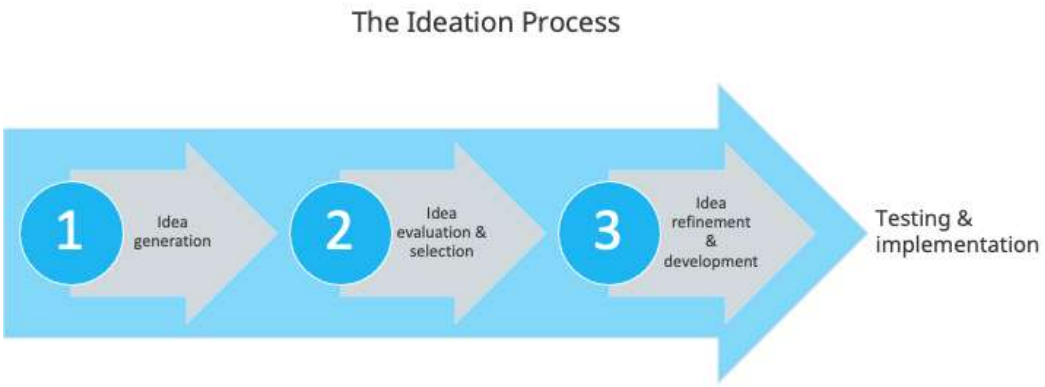
Stages of IDP

Following are the four seminal phases of IDP.



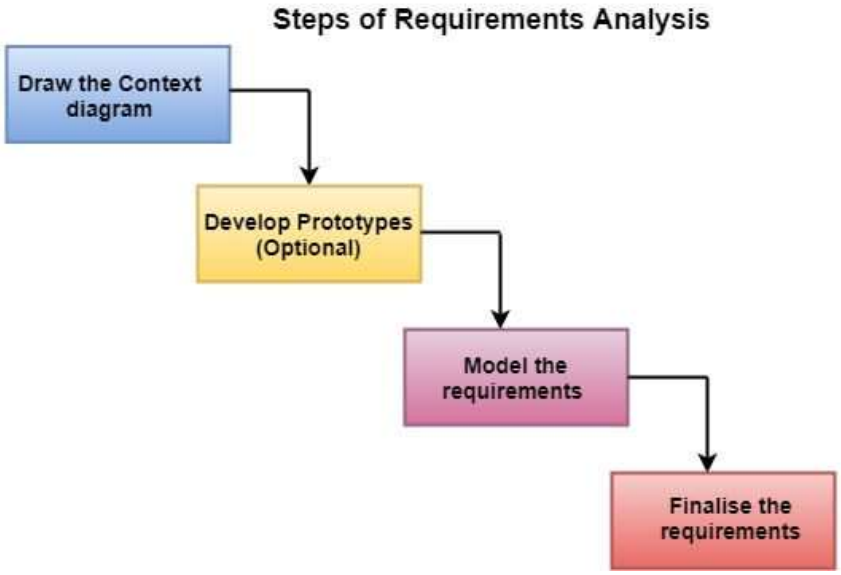
Phase 1: Ideation

Ideation is the creative process of generating, developing, and communicating ideas. It's important to note that these ideas don't have to be completely new. You can ideate to solve specific problems, look into new ways of implementing a solution, or even collect feedback and evaluate ideas.



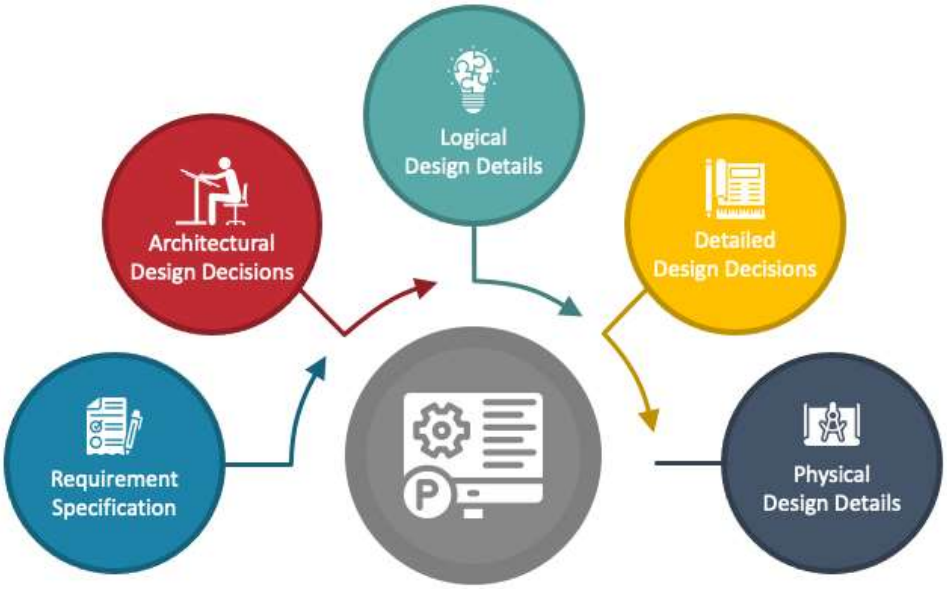
Phase 2: Requirement Analysis

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.



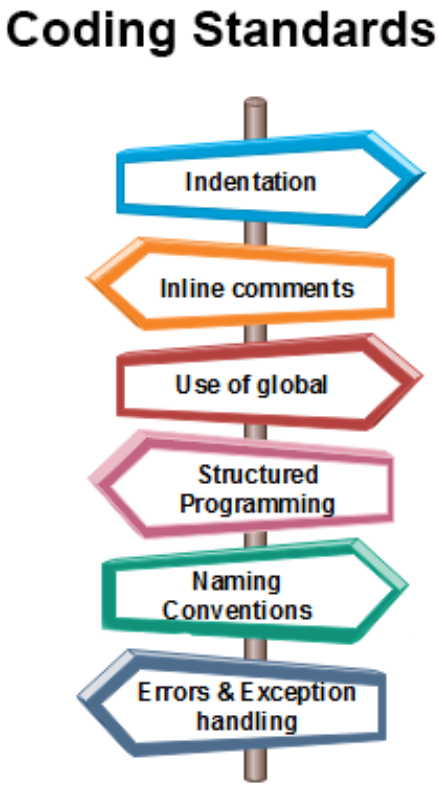
Phase 3: Project Design

Project design is a process to transform user requirements into some suitable form, which helps the programmer in software coding and implementation.



Phase 4: Project Development

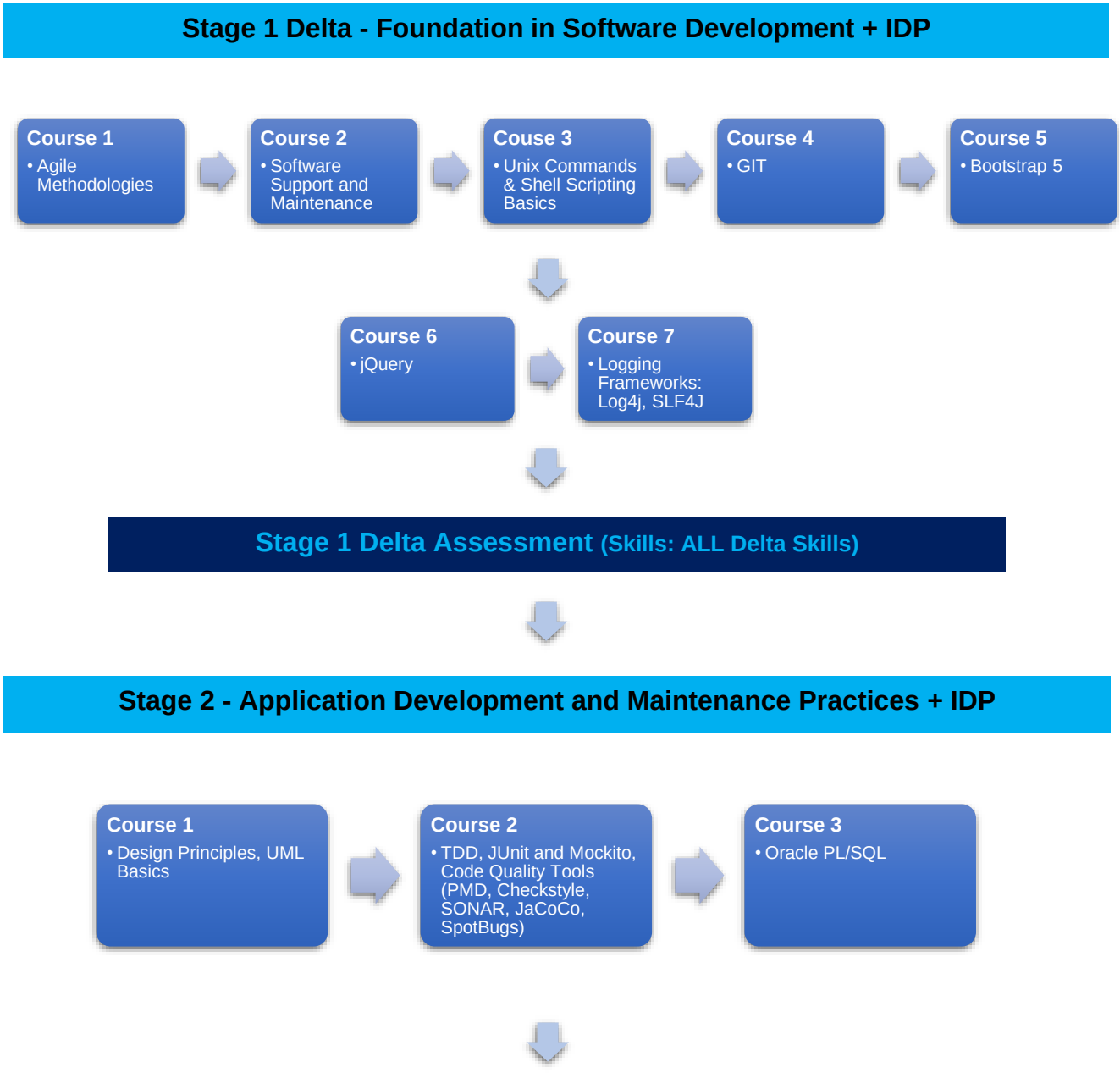
Once the system design phase is over, the next phase is development. In this phase, developers start build the entire system by writing code using the chosen programming language. In this phase, tasks are divided into units or modules and assigned to the various developers. It is the longest phase of the Software Development Life Cycle process.

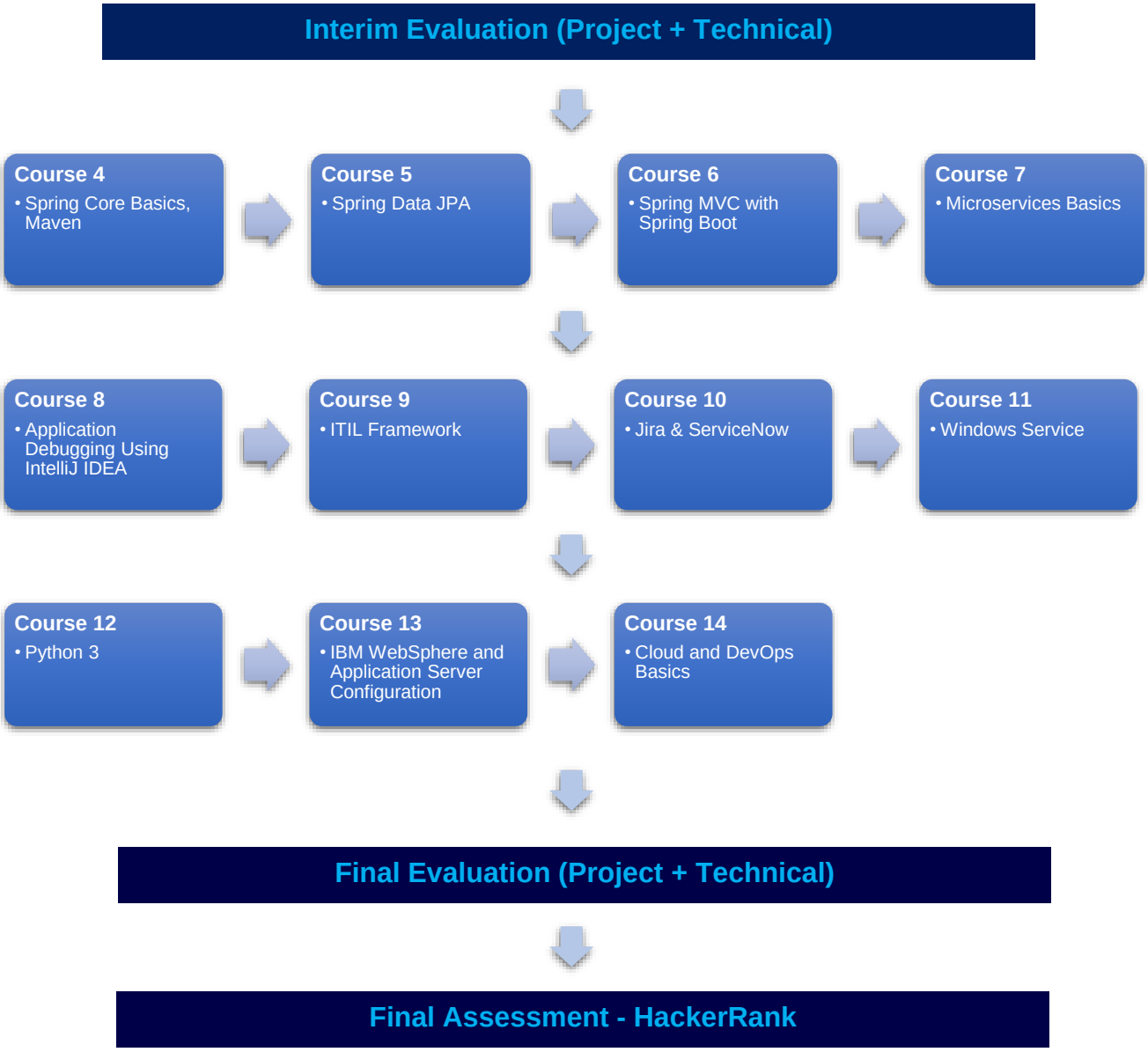


Recommended Program Sequence

The learning journey starts with **5 days of Icebreaker sessions**, followed by two stages of technical learning and a Project**.**

- **Stage 1 Delta** - Foundation in Software Development
- **Stage 2** - Application Development and Maintenance Practices
- Integrated Development Project (IDP) - Stage 1 & 2





Key Learning and Evaluation Components of the Program

Self-Learning via Udemy

Cognizant has collaborated with Udemy to provide world class learning videos for the evolving future of work. These Udemy programs are woven into a learning path, empowering you to plan and learn at your style.

The program also connects you with Subject Matter Experts (SMEs) to get the professional guidance on your queries in the learning journey.

RAG as PHS (Performance Health Status)

The program continuously evaluates if you are able to apply those self-learnt skills to solve a real-time business problem. Depicted below are the two key evaluation components, which are distributed across the learning journey for the purpose of continuous evaluation.

Interim Evaluation:

During the interim evaluation, the GenC will undergo a video interview on the learning platform. This interview will be conducted by a Tech SME from the BU. The purpose of this evaluation is to assess the GenC's knowledge and understanding of the skills covered in the training program up to the halfway point. It also encompasses an evaluation of the GenC's progress in their Integrated Development Project (IDP). The evaluation will involve a technical discussion as well as an assessment of the IDP progression to gauge the GenC's proficiency in the skills learned thus far.

Final Evaluation:

For the final evaluation, the GenC will participate in a video interview conducted by a Tech SME from the BU. This evaluation aims to assess the GenC's knowledge and expertise in all the skills covered throughout the entire training program. Similar to the interim evaluation, this assessment will involve a technical discussion via a video interview on the learning platform, along with a project evaluation to assess the GenC's capabilities and their IDP's progress. It serves as a comprehensive evaluation of the GenC's skills and capabilities acquired during the training.

Icebreaker Sessions

Icebreaker session will be conducted for a duration of initial **5 days**. During the session, various topics related to Corporate Induction, Talent Management, Cognizant Agenda on Core Values, Leader Talks, Alumni, BU Mentor connects will be covered. Followed by icebreaker, technical training will kick start.

Following sessions will be covered during the 5 days of icebreaker

- Corporate Induction
- Talent Manager Connect
- Cognizant Agenda Sessions on Core Values
- Leader Talks (Academy) and many more…

Stage 1 Delta - Foundation in Software Development

Overview

Stage 1 Delta deals with foundational technology skills that help GenCs to get start with their software engineering career. We provide unique learning experience to learners by including diversified learning content and learning methodologies that are based on adult learning principles. At the end of this stage, there will be a **Stage 1 Delta Assessment** which covers the delta skills learnt during the stage 1 training.

As part of Stage 1 Delta of your training, the following skills will be covered.

- Agile Methodology
- Software Support and Maintenance
- Unix Commands & Shell Scripting Basics
- GIT
- Bootstrap 5
- jQuery
- Logging Frameworks: Log4j, SLF4J

How and From Where to Learn?

Udemy courses are recommended for learning, and hands-on exercises are organized within a learning path on the **Tekstac** platform for practice. Additionally, you can utilize other sources mentioned in this handbook for learning.

Stage 1 -> Course 1 -> Agile Methodology

Course Overview

In the **Course 1** of the **Stage 1**, learners will be introduced to the basics of **Agile methodology**. Agile is an approach to project management and software development that emphasizes flexibility, collaboration, and customer satisfaction. It involves adaptive planning, iterative development, early delivery, and continuous improvement. Agile methodologies, like Scrum and Kanban, focus on delivering value to the customer and responding to change effectively.

Learning Objectives

After completing this course, GenCs will be able to:

- Understand the principles and values of Agile methodology.
- Describe the benefits of using Agile in software development.
- Explain the differences between Agile and traditional project management approaches.
- Identify the key roles and responsibilities in Agile teams.
- Describe the iterative and incremental nature of Agile development.
- Explain the importance of customer collaboration and feedback in Agile.
- Describe common Agile practices, such as user stories, sprints, and retrospectives.

- Identify common Agile frameworks, such as Scrum, Kanban, and Extreme Programming (XP).
- Explain how Agile principles can be applied in different project environments.

Day 1

Agile Methodology

Key Topics: Introduction to Agile, Agile Manifesto, Scrum Framework, Agile Estimation and Planning, Agile User Stories, Agile Metrics and Reporting

Continuous Learning: Technical Enablement

Agile Crash Course: Agile Project Management; Agile Delivery

- Learn All Sections in this Udemy course.

Stage 1 -> Course 2 -> Software Support and Maintenance

Course Overview

The **Course 2** of the **Stage 1** provides a comprehensive understanding of software development and maintenance processes, focusing on the Software Development Life Cycle (SDLC) phases and Agile methodology. It explores the critical need for software maintenance, challenges encountered during the maintenance phase, and the various categories of maintenance activities. Additionally, learners will gain insights into software reverse engineering and its significance, along with best practices for providing effective software support. The course is designed to equip learners with the necessary skills to ensure the efficiency and longevity of software systems in dynamic environments.

Learning Objectives

After completing this course, GenCs will be able to:

- Explain the stages of the Software Development Life Cycle and their role in delivering robust software solutions.
- Demonstrate knowledge of Agile principles and practices to enhance flexibility and collaboration in software development and maintenance.
- Articulate the importance of software maintenance in ensuring system performance and alignment with evolving requirements.
- Analyze common challenges in software maintenance and propose strategies to address them effectively.
- Differentiate between various types of software maintenance, including corrective, adaptive, perfective, and preventive maintenance.
- Describe the purpose and process of software reverse engineering in understanding legacy systems and facilitating updates.
- Develop strategies for providing efficient and proactive software support to meet user and business needs.

Day 2

Software Support and Maintenance

Key Topics: SDLC vs Agile, Software maintenance and support - overview

Continuous Learning: Technical Enablement

Software Engineering 101: Plan and Execute Better Software

- Learn the sections listed below in this Udemy course and take up the Quizzes in each section in order to check your understanding about the subject.
    - **Section 2:** Software Lifecycle
    - **Section 3:** Requirements and Specifications
    - **Section 4:** Design: Architecture
    - **Section 5:** Design: Modularity
    - **Section 6:** Implementation and Deployment
    - **Section 7:** Testing
    - **Section 8:** Software Development Models

Additional Learning

Learn about Software Maintenance from the following:

- Overview of Software Maintenance

Stage 1 -> Course 3 -> Unix Commands & Shell Scripting Basics

Course Overview

The **Course 3** of the **Stage 1** is designed to provide learners with a comprehensive introduction to Unix and Unix-based systems. It covers essential skills required for effective file and directory management, mastery of file permissions, and the use of basic utilities. Learners will explore powerful tools like pipes, filters, and text processing commands to manipulate data efficiently. The course also delves into process management and network communication utilities, laying a strong foundation for Unix system usage. Additionally, participants will gain hands-on experience with Unix shell scripting, from writing their first script to understanding scripting fundamentals and incorporating error-handling techniques. This course equips learners with the skills needed to automate tasks, streamline workflows, and enhance their productivity in a Unix environment.

Learning Objectives

After completing this course, GenCs will be able to:

- Understand the Unix operating system and its core functionalities.
- Perform effective file and directory management using Unix commands.
- Configure and modify file permissions and access modes for secure operations.

- Utilize basic utilities, pipes, and filters to process and manipulate data efficiently.
- Apply text processing commands to manage and analyze text-based data.
- Manage processes and monitor system performance using Unix process management tools.
- Utilize network communication utilities for system interaction and data transfer.
- Develop a foundational understanding of Unix shell scripting, including scripting syntax and structure.
- Write and execute basic shell scripts to automate routine tasks.
- Implement error-handling techniques in shell scripts to ensure robust script execution.

Unix Commands

Key Topics: Introduction, File Management, Directory Management, File Permission / Access Modes, Basic Utilities, Pipes and Filters, Process Management, Network Communication Utilities

Continuous Learning: Technical Enablement

The Linux Command Line Bootcamp: Beginner To Power User

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 2:** Introduction
  - **Section 3:** Command Basics
  - **Section 4:** Getting Help
  - **Section 5:** Navigation
  - **Section 6:** Creating Files & Folders
  - **Section 8:** Deleting, Copying, & Moving
  - **Section 10:** Working With Files
  - **Section 11:** Redirection
  - **Section 12:** Piping
  - **Section 14:** Finding Things
  - **Section 15:** Grep
  - **Section 16:** Permissions Basics
  - **Section 17:** Altering Permissions
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Display the days of the year 2016
- Calculator
- Display the hidden files or directories
- Copy a File using relative path
- Move a File to mammals using relative path
- Change File Permission adding write permission to group
- Change File permission by removing write permission as specified
- Display the lines of the file that do not contain the string

- List all the file-names starting with "t" or "s"
- Count the Files in a directory

Additional Learning

Learn about Process Management and Network Communication Utilities from the following

- Linux/Unix Process Management
- Network Communication Utilities

Technical Quiz

Attempt the following technical quiz in the Learning Path at Tekstac for checking your knowledge level on Unix and Shell Scripting.

- Introduction to Unix
  - Pre-Quiz
  - Post-Quiz
- File System
  - Pre-Quiz
  - Post-Quiz
- Filters
  - Pre-Quiz
  - Post-Quiz
- Test Your Understanding - Introduction to Unix
- Test Your Understanding - File System
- Test Your Understanding – Filters

Day 3 - Forenoon

Shell Scripting Basics

Key Topics: Introduction to Shell Scripting, Basic Shell Scripting Concepts, Control Structures, Command-Line Arguments, Functions, Text Processing, Error Handling and Debugging

Continuous Learning: Technical Enablement

Shell Scripting: Discover How to Automate Command Line Tasks

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 2:** Shell Scripting, Succinctly
  - **Section 5:** Shell Script Checklist and Template
  - **Section 7:** Case Statements
  - **Section 9:** While Loops
  - **Section 3:** Exit Statuses and Return Codes
  - **Section 4:** Functions
  - **Section 8:** Logging
  - **Section 10:** Debugging
  - **Section 11:** Data Manipulation and Text Transformation with Sed
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within

each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Permission Change
- Factorial of a number

Technical Quiz

Attempt the following technical quiz in the Learning Path at Tekstac for checking your knowledge level on Unix and Shell Scripting.

- Vi Editor
  - Pre-Quiz
  - Post-Quiz
- Bourne Shell
  - Pre-Quiz
  - Post-Quiz
- Test Your Understanding - Vi Editor
- Test Your Understanding - Bourne Shell

Stage 1 -> Course 4 -> GIT

Course Overview

The **Course 4** of the **Stage 1** provides a comprehensive understanding of Git, a powerful version control system widely used in software development. Participants will learn the foundational concepts and advanced techniques necessary to effectively manage and collaborate on projects using Git. The course is structured to guide learners from working locally with Git to mastering collaborative workflows, including branching, merging, and rebasing. By the end of this course, participants will be well-equipped to handle real-world version control challenges and streamline their development processes.

Learning Objectives

After completing this course, GenCs will be able to:

- Grasp fundamental Git concepts, including repositories, versioning, and the role of Git in software development workflows.
- Set up and initialize local Git repositories.
- Track changes and manage versions using essential Git commands like add, commit, and log.

- Navigate through commit history and revert changes when necessary.
- Connect local repositories to remote services like GitHub or GitLab.
- Push and pull changes to and from remote repositories.
- Understand and resolve common conflicts during collaborative workflows.
- Create and manage branches for feature development and bug fixes.
- Merge changes effectively to maintain a stable codebase.
- Rebase commits to keep a clean and linear project history.

Day 3 - Afternoon, Day 4 - Forenoon

GIT

Key Topics: Introduction, Working Locally with Git, Working Remotely with Git, Branching, Merging, and Rebasing with Git

Continuous Learning: Technical Enablement

Git Complete: The definitive, step-by-step guide to Git

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 1:** Introduction
  - **Section 2:** Installation
  - **Section 3:** Git Quick Start
  - **Section 6:** Basic Git Commands
  - **Section 8:** Comparisons
  - **Section 9:** Branching and Merging
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Git Config
- Clone Repo
- Add, Commit And Push
- Pull And Merge
- Merge - Resolve Conflict
- Git Tags

Course Overview

The **Course 5** of the **Stage 1** provides a comprehensive introduction to Bootstrap 5, the world's most popular front-end framework for building responsive, mobile-first websites. GenCs will explore the fundamental concepts and tools that Bootstrap 5 offers, empowering them to design and develop modern, dynamic web interfaces efficiently. From understanding the layout and grid system to leveraging essential components, utilities, and helpers, this course will equip them with the skills to create visually appealing and user-friendly web pages. Additionally, GenCs will learn how to customize and extend Bootstrap to meet specific project requirements while mastering responsive design principles.

Learning Objectives

After completing this course, GenCs will be able to:

- Describe the purpose and features of Bootstrap 5 as a front-end framework.
- Design responsive web layouts using the Bootstrap grid system.
- Implement advanced layout techniques for complex web pages.
- Incorporate common UI elements like navigation bars, modals, buttons, and cards to enhance interactivity and usability.
- Use Bootstrap utilities and helper classes to streamline the styling and functionality of web pages.
- Develop web pages that adapt seamlessly to different devices and screen sizes.
- Leverage responsive breakpoints effectively for design adjustments.
- Modify default themes and styles to align with specific design requirements.
- Integrate Bootstrap with custom CSS and JavaScript to extend its functionality.

Day 4 - Afternoon

Bootstrap 5

Key Topics: Introduction to Bootstrap 5, Layout and Grid System

Continuous Learning: Technical Enablement

The Ultimate Bootstrap Guide - Bootstrap 5 from Scratch

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 1:** Getting Started with Bootstrap 5
    - **Section 2:** Layouts in Bootstrap 5
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Ohm's Law

Day 5

Bootstrap 5

Key Topics: Essential Components, Utilities and Helpers

Continuous Learning: Technical Enablement

The Ultimate Bootstrap Guide - Bootstrap 5 from Scratch

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 5:** Components in Bootstrap 5
    - **Section 6:** Helpers in Bootstrap 5
    - **Section 7:** Utilities in Bootstrap 5
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Sticky Notes
- Portfolio
- Library Home Page
- Hi-Tech Digital world

Day 6 - Forenoon

Bootstrap 5

Key Topics: Responsive Design, Customization and Extensions

Continuous Learning: Technical Enablement

- Responsive Web Design - Media Queries
- Responsive Web Design - Images
- Bootstrap 5 Navbar Responsive behaviors Toggler
- How to theme, customize, and extend Bootstrap with Sass

Course Overview

The **Course 6** of the **Stage 1** is designed to provide a comprehensive introduction to jQuery, a powerful JavaScript library that simplifies web development by streamlining tasks such as DOM manipulation, event handling, and AJAX integration. Through practical examples, GenCs will learn how to leverage jQuery's robust features to enhance user interfaces and create dynamic, interactive web applications.

Learning Objectives

After completing this course, GenCs will be able to:

- Explain the purpose and benefits of using jQuery in web development.
- Identify the core features and components of the jQuery library.
- Select and manipulate HTML elements efficiently using jQuery selectors and methods.
- Apply various jQuery methods to alter content, attributes, and styles dynamically.
- Perform advanced DOM manipulation, including traversing and filtering elements.
- Attach and manage event handlers to create interactive web experiences.
- Use jQuery's AJAX capabilities to send and receive data asynchronously.

Day 6 - Afternoon, Day 7 - Forenoon

jQuery

Key Topics: jQuery and its features, Basic components, DOM manipulation & events, Basic AJAX with jQuery

Continuous Learning: Technical Enablement

The Complete jQuery Course: From Beginner To Advanced!

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 1:** Introduction
    - **Section 3:** Element Selectors
    - **Section 4:** Manipulating the DOM I – Inserting, Replacing and Removing Elements
    - **Section 5:** Manipulating the DOM II – Changing Element Data and CSS
    - **Section 6:** Events I – Handling Mouse Events & Keyboard Events
    - **Section 7:** Events II – Forms
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Day 7 - Afternoon, Day 8 - Forenoon

IDP

- High-Level Design Demo

Stage 1 -> Course 7 -> Logging Frameworks: Log4j, SLF4J

Course Overview

The **Course 7** of the **Stage 1** is designed to provide a comprehensive understanding of logging in Java applications. Logging plays a crucial role in monitoring, debugging, and maintaining software systems. In this course, GenCs will explore why logging is important, its common use cases, and different logging levels, along with hands-on experience using popular Java logging frameworks such as java.util.logging, Log4j, and SLF4J. Participants will gain the skills to configure, use, and optimize these frameworks to enhance application observability and troubleshooting.

By the end of this course, GenCs will have a strong foundation in integrating robust logging mechanisms into their Java applications, enabling better debugging, improved system performance, and effective issue tracking.

Learning Objectives

After completing this course, GenCs will be able to:

- Articulate why logging is essential in software development.
- Identify common use cases of logging in application monitoring and debugging.
- Distinguish between various logging levels, including INFO, WARNING, ERROR, and DEBUG.
- Utilize the java.util.logging package for basic logging in Java applications.
- Explain the purpose and benefits of Log4j as a logging framework.
- Configure Log4j using configuration files such as log4j.properties and log4j.xml.
- Use logging levels, loggers, and appenders effectively.
- Write and format log statements to include detailed exception stack traces.
- Implement various appenders, such as Console, File, Rolling File, SMTP, and Database appenders.
- Understand and manage Logger Hierarchy, including logger names, hierarchies, and properties.
- Understand the purpose and advantages of SLF4J over Log4j.
- Use the SLF4J API with components such as LoggerFactory and Logger interface.
- Write structured log statements using SLF4J's logging methods and markers.
- Configure SLF4J with default and custom settings and bind it with various logging frameworks.
- Design efficient logging strategies tailored to application requirements.
- Optimize logging configurations for performance and maintainability.
- Demonstrate the ability to troubleshoot and debug applications using advanced logging techniques.

Day 8 - Afternoon, 9

Logging Frameworks: Log4j, SLF4J

Key Topics: Introduction to Logging, Java Logging Frameworks

Continuous Learning: Technical Enablement

Java Best Practices for Efficient, Scalable, and Secure Code

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 23: Logging in Java**
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Day 10, 11, 12 - Forenoon

Project High Level Design Demo, Project Artifacts Review

- High-Level Design Demo
- Project Artifacts Review

Stage 1 Delta Assessment

Day 12 - Afternoon

Stage 1 Delta Assessment

- This day will be dedicated for the Stage 1 Delta Assessment.

Assessment Overview

- **Assessment Type:** Knowledge-Based Assessment (KBA)
- **Duration:** 90 minutes
- **Number of Questions:** 40
- **Modules Covered:**
  - Software Support and Maintenance
  - Unix Commands & Shell Scripting Basics
  - GIT
  - Bootstrap 5
  - jQuery

- Log4j
- SLF4J

- **Maximum Attempts:** 3

Stage 2 - Application Development and Maintenance Practices

Overview

Stage 2 focuses on Application Development and Maintenance Practices essential for developing and maintaining diverse software applications. We offer a unique learning experience by providing diversified content and methodologies based on adult learning principles

As part of Stage 2 of your training, the following skills will be covered.

- Design Principles, UML Basics
- TDD, JUnit and Mockito, Code Quality Tools (PMD, Checkstyle, SONAR, JaCoCo, SpotBugs)
- Oracle PL/SQL
- Spring Core Basics, Maven
- Spring Data JPA
- Spring MVC with Spring Boot
- Microservices Basics
- Application Debugging Using IntelliJ IDEA
- ITIL Framework
- Jira & ServiceNow
- Windows Service
- Python 3
- IBM WebSphere and Application Server Configuration
- Cloud and DevOps Basics

How and From Where to Learn?

- Udemy learnings are recommended in the Platform to understand the fundamental concepts. In addition to this, you can also learn from any other sources as they are mentioned in this handbook.

Stage 2 -> Course 1 -> Design Principles, UML Basics

Course Overview

The **Course 1** of the **Stage 2** provides a comprehensive understanding of the principles and practices of software design and modeling, focusing on the **SOLID principles** and the DRY (Don't Repeat Yourself) approach to software development. Students will gain practical knowledge of the **Unified Modeling Language (UML)**, its purpose, benefits, and various diagram types used in the software development lifecycle.

By mastering UML's structural, behavioral, and interaction diagrams, participants will develop the ability to create detailed and effective designs for complex systems. Topics include the use of UML symbols and notations, understanding classes, objects, and interfaces, and exploring relationships such as association, aggregation, composition, and inheritance. Additionally, the course delves into

advanced UML concepts such as multiplicity, navigation, attributes and methods, packages and subsystems, stereotypes, and profiles.

Learning Objectives

After completing this course, GenCs will be able to:

- Understand and apply SOLID principles and the DRY approach to improve code maintainability and reusability.
- Explain the purpose and benefits of UML in the software development process.
- Identify and differentiate between UML structural diagrams, behavioral diagrams, and interaction diagrams, including their specific uses and components.
- Develop comprehensive UML models using appropriate symbols, notations, and diagramming techniques.
- Describe and illustrate the relationships between classes, objects, and interfaces, including associations, aggregations, compositions, and inheritance.
- Apply concepts of multiplicity, navigation, attributes and methods, packages, subsystems, stereotypes, and profiles to create detailed system models.
- Employ UML diagrams to document and communicate system requirements and designs effectively in agile development environments.

Day 12 - Afternoon

Design Principles, UML Basics

Key Topics: Different Types of Software Design Principles

Continuous Learning: Technical Enablement

SOLID Principles: Introducing Software Architecture & Design

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **All** Sections
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Day 13, 14 - Forenoon

Design Patterns and Principles, UML Basics

Key Topics: Introduction to UML, UML Diagram Types, UML Diagram Elements

Continuous Learning: Technical Enablement

The Complete UML Course (2024): Learn to Design UML Diagrams

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.

- **All** Sections
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Stage 2 -> Course 2 -> TDD, JUnit and Mockito, Code Quality Tools (PMD, CheckStyle, SONAR, JaCoCo, SpotBugs)

Course Overview

The **Course 2** of the **Stage 2** provides a comprehensive introduction to **Test-Driven Development (TDD)** and its principles, alongside essential tools and frameworks used for unit testing, code quality, and coverage analysis in Java. GenCs will learn how to write clean, maintainable, and reliable code through a hands-on approach, leveraging TDD cycles, unit testing frameworks like **JUnit and Mockito**, and static code analysis tools such as **PMD**, **CheckStyle**, **SonarQube**, **SpotBugs**, and **JaCoCo**. The course emphasizes the importance of adopting best practices for ensuring software quality, improving productivity, and building confidence in the development process.

Through interactive sessions, practical exercises, and real-world examples, GenCs will gain proficiency in writing test-first code, refactoring confidently, and utilizing tools to analyze and improve code quality. The course also highlights strategies for maintaining test clarity and organization, making it ideal for developers seeking to enhance their testing and development capabilities.

Learning Objectives

After completing this course, GenCs will be able to:

- Explain the principles and the Red-Green-Refactor cycle of TDD.
- Design and write test cases before implementing code.
- Refactor code with confidence, ensuring functionality remains intact.
- Define unit testing and explain its purpose and benefits.
- Recognize the importance of unit testing in improving code quality and detecting bugs early.
- Write and execute test cases using JUnit and understand the structure of test classes.
- Use assertions effectively to validate code behavior.
- Leverage annotations like @Test for simplifying test case management.
- Understand the purpose and benefits of Mockito for mocking dependencies.
- Create and use mock objects to simulate behavior in unit tests.
- Verify method calls and parameters for thorough testing.
- Set up and use tools like PMD, Checkstyle, SonarQube, and SpotBugs to analyze and improve code quality.
- Understand and customize rules to fit project-specific needs.
- Generate and interpret detailed reports for identifying and addressing issues.
- Set up and run JaCoCo to measure code coverage.
- Generate and analyze coverage reports to ensure robust testing.
- Write tests with clarity and focus on readability and organization.
- Avoid overusing mocks and prioritize behavior-driven testing.

Day 14 - Afternoon

TDD, JUnit and Mockito

Key Topics: Test-Driven Development (TDD), Unit Testing Basics, Introduction to JUnit, Introduction to Mockito

Continuous Learning: Technical Enablement

Instant Test Driven Development with Java, JUnit and Mockito

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 1:** Introduction
  - **Section 2:** The Basics
  - **Section 3:** Basic Testing with Mocks using Mockito
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- EB Connection
- CarcoExpress
- Employee Details
- Employee Appraisal

Day 15

TDD, JUnit and Mockito

Key Topics: JUnit Test Structure, Stubbing and Verification, Mockito Best Practices

Continuous Learning: Technical Enablement

Instant Test Driven Development with Java, JUnit and Mockito

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 4:** JUnit in more Depth
  - **Section 5:** More Assertions
  - **Section 6:** Parameterized Tests
  - **Section 7:** Mockito in more Depth

- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Employee Address Update
- Computation Service
- Employee Details
- Crisp Placement Training

Code Challenges (For Practice Only)

Attempt the following Code Challenge through the Learning Path at Tekstac to assess your skill level in Unit Testing concepts. You need to score 70% or higher to clear this challenge. You will have a maximum of three attempts to complete the practice test.

Do not copy paste the code. Write the code yourself.

- Assess-Type-1: Code Challenge - TDD, Junits

Day 16, 17 - Forenoon

Code Quality Tools (PMD, Checkstyle, SONAR, JaCoCo, SpotBugs)

Key Topics: PMD, Checkstyle, SonarQube, JaCoCo, SpotBugs

Continuous Learning: Technical Enablement

Tool Name	Topics
PMD	Introduction to PMD
	PMD Rules
	PMD Reports
	Rule Customization and Suppression
Checkstyle	Introduction to Checkstyle
	Checkstyle Rules
	Checkstyle Reports
	IDE Integrations
SonarQube	Introduction to SonarQube
	SonarQube Metrics and Rules
	SonarQube Reports
	IDE Integrations
SpotBugs	Introduction to SpotBugs
	SpotBugs Analysis

	SpotBugs Reports
	IDE Integrations
JaCoCo	Introduction to JaCoCo
	JaCoCo Reports

Hands-On

- Try out the example codes and steps covered in the above links.

Code Challenges (For Practice Only)

Attempt the following Code Challenge through the Learning Path at Tekstac to assess your skill level in Code Quality concepts. You need to score 70% or higher to clear this challenge. You will have a maximum of three attempts to complete the practice test.

Do not copy paste the code. Write the code yourself.

- Assess-Type-1: Code Challenge - Code Quality

Day 17 - Afternoon, 18 - Forenoon

- The above days have been included to adjust the behavioral training.

Stage 2 -> Course 3 -> Oracle PL/SQL

Course Overview

The **Course 3** of the **Stage 2** offers a comprehensive exploration of **PL/SQL (Procedural Language/Structured Query Language)**, equipping GenCs with the knowledge and skills necessary to leverage PL/SQL in database programming and management. Designed for individuals seeking to deepen their understanding of database development, the course emphasizes practical problem-solving through hands-on coding, structured learning, and real-world applications. Key topics include the foundational elements of PL/SQL, its advantages over SQL, and advanced features like cursors, triggers, packages, dynamic SQL, error handling, and bulk operations. By the end of the course, GenCs will have a strong grasp of PL/SQL's capabilities and its integration with Oracle databases.

Learning Objectives

After completing this course, GenCs will be able to:

- Define PL/SQL and explain its purpose.
- Compare and contrast PL/SQL with SQL.
- Recognize the advantages of using PL/SQL in database applications.
- Identify the components of a PL/SQL block.
- Declare and initialize variables and constants.
- Understand the scope and lifetime of variables.
- Develop anonymous blocks, stored procedures, functions, and packages.
- Create triggers to automate database processes.
- Differentiate between scalar, composite, and LOB data types.

- Utilize %TYPE and %ROWTYPE attributes for dynamic typing.
- Implement conditional statements like IF-THEN-ELSE and CASE.
- Utilize looping constructs such as FOR, WHILE, and simple loops.
- Identify and manage different types of exceptions.
- Create and propagate custom exceptions.
- Employ implicit and explicit cursors effectively.
- Utilize cursor attributes for better control and optimization.
- Write and execute dynamic SQL with bind variables and error handling.
- Develop and manage packages, distinguishing between public and private members.
- Use autonomous transactions and logging mechanisms for debugging.
- Implement bulk operations with BULK COLLECT and FORALL statements.

Day 18 - Afternoon

Oracle PL/SQL

Key Topics: Introduction to PL/SQL

Continuous Learning: Technical Enablement

The Complete PL/SQL Bootcamp : "Beginner to Advanced PL/SQL"

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 1:** What is PL/SQL?
  - **Section 2:** Software Installation
  - **Section 3:** Let's Start Coding
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Update Salary
- Delete Employees

Day 19

Oracle PL/SQL

Key Topics: PL/SQL Variables and Data Types, Control Structures in PL/SQL.

Continuous Learning: Technical Enablement

The Complete PL/SQL Bootcamp : "Beginner to Advanced PL/SQL"

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 4:** PL/SQL Variables
  - **Section 5:** Control Structures
  - **Section 6:** Using SQL in PL/SQL
  - **Section 7:** Composite Data Types
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Day 20

Oracle PL/SQL

Key Topics: Exception Handling, Cursors, Procedures and Functions

Continuous Learning: Technical Enablement

The Complete PL/SQL Bootcamp : "Beginner to Advanced PL/SQL"

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 9:** PL/SQL Exceptions
  - **Section 8:** PL/SQL Cursors
  - **Section 10:** PL/SQL Functions & Procedures
  - **Section 12:** PL/SQL Triggers
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Department Name -Procedure
- Rating Info- Procedure Using Cursor
- Alpha Count-Function
- Promotion- Function
- Address Validation- Exception Handling
- Delete a Record - Trigger

Day 21 - Forenoon

Oracle PL/SQL

Key Topics: Packages, Dynamic SQL in PL/SQL, Error Handling and Logging, Bulk Processing

Continuous Learning: Technical Enablement

 The Complete PL/SQL Bootcamp : "Beginner to Advanced PL/SQL"

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 11:** PL/SQL Packages
  - **Section 13:** PL/SQL Debugging: Debugging Your Codes & Subprograms
  - **Section 14:** Using Dynamic SQL&PL/SQL in PL/SQL!
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

 Do not copy paste the code. Write the code yourself.

- Employee Information - Package

Code Challenge (For Practice Only)

Attempt the following Code Challenge through the Learning Path at Tekstac to check your skill level in PL/SQL. You need to score 70% or higher to clear this challenge.

 Do not copy paste the code. Write the code yourself.

- Assess-Type-1: Code Challenge - Advanced SQL

IDP - Project Activities

Day 21 - Afternoon, 22, 23, 24

Sprint 1 Development, Demo

- These three days will be spent on Sprint 1 Development, Review & Re-work.

Day 25 - Forenoon

- This has been included to adjust Behavioral Training and Cohort Mentor Sessions.

Interim Evaluation

Day 25 - Afternoon, 26, 27 - Forenoon

Interim Evaluation (Project + Technical)

- Interim Evaluation will be conducted on these days.

Stage 2 -> Course 4 -> Spring Core Basics, Maven

Course Overview

The **Course 4** of the **Stage 2** provides a comprehensive introduction to **Spring Framework and Maven**, empowering GenCs with the skills to develop robust, scalable, and maintainable Java applications. Beginning with an understanding of the Spring Framework's core principles and modules, the course delves into essential concepts like Dependency Injection, Inversion of Control, and resource management. It also covers practical approaches to integrating Spring with Maven for efficient build and deployment processes. By the end of this course, students will have a strong foundation to design and implement applications using Spring and Maven effectively.

Learning Objectives

After completing this course, GenCs will be able to:

- Define the Spring Framework and describe its advantages.
- Explore the structure and functionality of key Spring modules.
- Explain the role of the Spring Container in managing application components.
- Implement Setter and Constructor Injection.
- Utilize autowiring to simplify bean configurations.
- Manage complex data structures with Injecting Collections and Inner Beans.
- Understand and apply inheritance, interfaces, and bean scopes.
- Utilize PropertyPlaceholderConfigurer and ResourceBundleMessageSource for dynamic property and message management.
- Manage database transactions and execute queries effectively.
- Set up Maven and manage project dependencies.
- Use repositories effectively and configure the pom.xml file for various project needs.
- Integrate Maven with IDEs like Eclipse for seamless project management.
- Understand Maven’s build lifecycle and use environment variables.
- Configure and use Maven plugins for custom builds.
- Create, build, and deploy Maven projects, including running builds for a Spring application.

Day 27 - Afternoon

Spring Core Basics, Maven

Key Topics: Introduction to Spring Framework, Dependency Injection

Continuous Learning: Technical Enablement

Spring Framework in Easy Steps

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 1:** Introduction
    - **Section 2:** Software Setup
    - **Section 3:** Spring Core Concepts
    - **Section 4:** Setter Injection
    - **Section 5:** Life Cycle Methods
    - **Section 6:** Dependency Check, Inner beans and scopes
    - **Section 7:** Constructor Injection
    - **Section 9:** Autowiring
    - **Section 10:** Standalone Collections
    - **Section 11:** Stereotype Annotations
    - **Section 13:** Injecting Interfaces
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Display Staff Details
- Display Book Details – Autowiring
- Gold Rate Calculation – Collections
- Shipment-Item-Scope
- DBConfig-SetterBasedInjection
- Account-Loan
- Azure Horizon Resort
- Hospital Management
- Loan Pro

Day 28

Spring Core Basics, Maven

Key Topics: Reading values from Property Files, Spring JDBC and Transactions

Continuous Learning: Technical Enablement

Spring Framework in Easy Steps

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 8:** Using Properties
  - **Section 14:** Spring JDBC
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- PatientManagement
- Billing Software Application
- EBill

Day 29 - Forenoon

Spring Core Basics, Maven

Key Topics: Maven Basics, Build Lifecycle, Plugins, Running Maven Builds

Continuous Learning: Technical Enablement

Maven Crash Course

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 1:** Introduction
  - **Section 2:** Simple Software Setup
  - **Section 3:** Maven Project Creation and Key Concepts
  - **Section 4:** Maven in Eclipse
  - **Section 5:** Maven Web Application
  - **Section 6:** Multi Module Project Creation
  - **Section 7:** Organizing the multi module project
  - **Section 8:** Scopes

    - **Section 9:** Dependency Management
    - **Section 10:** Profiles
    - **Section 15:** Plugins
    - **Section 16:** Properties
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Build Web Application
- Compiling Executing Java using POM
- Junit With Maven
- Maven Directory Structure
- Maven Shade Plugin
- Maven App to access External Service
- Capital Service

Code Challenge (For Practice Only)

Attempt the following Code Challenge through the Learning Path at Tekstac to check your skill level in Spring Core. You need to score 70% or higher to clear this challenge.

Do not copy paste the code. Write the code yourself.

- Assess-Type-1: Code Challenge - Spring Framework

Stage 2 -> Course 5 -> Spring Data JPA

Course Overview

The **Course 5** of the **Stage 2** introduces GenCs to Java Persistence API (JPA) and Spring Data JPA, two essential frameworks for simplifying data access in Java applications. It covers the fundamental concepts of JPA, its advantages, and why Spring Data JPA is a preferred choice for database interactions. Through hands-on exercises and examples, learners will understand how to configure, implement, and optimize JPA repositories for various use cases, including querying, sorting, filtering, and working with relationships between entities.

The course also dives into Hibernate configurations and explores advanced topics such as relationship mappings and fetch strategies, equipping learners with the skills to efficiently manage database interactions in real-world projects.

Learning Objectives

After completing this course, GenCs will be able to:

- Explain the purpose of JPA and its advantages.
- Describe the architecture of Spring Data JPA and its differences from Hibernate.
- Configure Hibernate logging and ddl-auto settings.
- Implement entity mappings using annotations like @ManyToOne, @OneToMany, @ManyToMany, and define fetch strategies using FetchType.EAGER and FetchType.LAZY.
- Use key JpaRepository methods such as findById(), save(), and deleteById().
- Define query methods to perform complex searches, including filtering by text, paging and sorting, and fetching records based on date ranges or numeric conditions.
- Establish and manage relationships between entities using annotations like @JoinColumn and @JoinTable.
- Handle bidirectional relationships with the mappedBy attribute.
- Leverage advanced query methods to retrieve data efficiently.
- Compare and contrast different fetch strategies and their impact on application performance.
- Build scalable and maintainable data access layers using Spring Data JPA.
- Implement best practices for repository and entity design.

Day 29 - Afternoon

Spring Data JPA

Key Topics: ORM Basics

Continuous Learning: Technical Enablement

Spring Data JPA using Hibernate

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 4:** ORM Basics
  - **Section 5:** Simple CRUD operations
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Additional Learning

Go through the below topic to learn Spring Data JPA features and Repository.

- Spring Data JPA Introduction

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- School Strength
- Loan Lenders

Day 30, 31, 32 - Forenoon

Spring Data JPA

Key Topics: Mapping, Persisting objects into database, Detached Objects

Continuous Learning: Technical Enablement

Spring Data JPA using Hibernate

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 6:** Generators
    - **Section 7:** Spring Data Finder Methods
    - **Section 8:** Paging and Sorting
    - **Section 9:** JPQL
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Additional Learning

Go through the below topic to learn Spring Data JPA Relationships.

- Spring Data JPA Relationships

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Account Transaction
- Sales Statistics
- Account with ATM Card With Logger

Code Challenge (For Practice Only)

Attempt the following Code Challenge through the Learning Path at Tekstac to check your skill level in Spring Data JPA. You need to score 70% or higher to clear this challenge.

Do not copy paste the code. Write the code yourself.

- Assess-Type-1: Code Challenge - Spring Data JPA with Spring Boot

Stage 2 -> Course 6 -> Spring MVC with Spring Boot

Course Overview

The **Course 6** of the **Stage 2** offers a comprehensive introduction to building modern, scalable, and secure web applications using Spring Boot and related technologies. Designed for aspiring developers, the curriculum covers the foundational aspects of Java-based web development, including installing Java and setting up the development environment with Eclipse. GenCs will explore key features of Spring Boot, its seamless integration with Spring MVC, and advanced concepts like reactive programming using Spring WebFlux. With a focus on hands-on learning, the course emphasizes practical application through the creation of RESTful services, form handling, database connectivity, and security configuration. By the end of the course, learners will have gained a strong foundation in developing robust, performant, and secure web applications.

Learning Objectives

After completing this course, GenCs will be able to:

- Add and configure Spring Boot dependencies.
- Explain the core features and benefits of Spring Boot.
- Configure Spring Boot applications using properties and starter POMs.
- Develop and run a simple Spring Boot application.
- Understand the MVC architecture and request-response cycle.
- Create Spring MVC projects with controllers and JSP views.
- Implement best practices for request mapping and parameter handling.
- Perform data binding and form handling using Spring MVC.
- Validate form data and display error messages effectively.
- Differentiate between JDBC, JPA, and Hibernate.
- Set up and configure a database connection.
- Create and map entity classes and initialize database schemas.
- Create dynamic HTML templates using Thymeleaf syntax.
- Configure view resolvers and integrate views into Spring MVC applications.
- Explain the importance of authentication and authorization.
- Configure Spring Security for web applications.
- Define and implement security rules and authentication providers.
- Differentiate between Spring MVC and Spring WebFlux.
- Set up reactive web applications using Spring WebFlux.

- Develop reactive controllers, services, and RESTful APIs using Mono<T> and Flux<T>.
- Understand the scalability and performance benefits of Spring WebFlux.
- Process asynchronous data effectively.
- Integrate with reactive databases using R2DBC.

Day 32 - Afternoon

Spring MVC with Spring Boot

Key Topics: Setting Up Development Environment, Spring Boot Fundamentals

Continuous Learning: Technical Enablement

[NEW] Spring Boot 3, Spring 6 & Hibernate for Beginners

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 1:** NEW - Spring Boot 3 Quick Start
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- SwiftCartPortal

Day 33

Spring MVC with Spring Boot

Key Topics: Spring MVC Basics, Spring Boot and Spring MVC Integration, Handling Form Submissions

Continuous Learning: Technical Enablement

[NEW] Spring Boot 3, Spring 6 & Hibernate for Beginners

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 6:** NEW - Spring MVC
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Sports Academy
- Top Colleges List

Day 34

Spring MVC with Spring Boot

Key Topics: Data Access with Spring Boot

Continuous Learning: Technical Enablement

[NEW] Spring Boot 3, Spring 6 & Hibernate for Beginners

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 7:** NEW - Spring MVC CRUD
    - **Section 3:** NEW - Hibernate/JPA CRUD
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Additional Learning

Walkthrough the below articles to understand the concepts about app server and web server.

- Introduction to Tomcat
- Tomcat servlet interactions
- Application Server Vs Web Server
- MVC Frameworks Overview
- MVC Architecture

Hands-On

Complete the following set of hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- SignUp Form
- Dexler Preschool
- PulsePursuit Tournament

Day 35, 36 - Forenoon

Spring MVC with Spring Boot

Key Topics: Spring Security, Integrating Reactive Programming with Spring WebFlux

Continuous Learning: Technical Enablement

[NEW] Spring Boot 3, Spring 6 & Hibernate for Beginners

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 8:** NEW - Spring MVC Security
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Reactive Programming with Spring Framework 5

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 2:** Introduction to Reactive Programming
    - **Section 3:** Netflux Reactive Example
    - **Section 4:** Spring Web Client
    - **Section 5:** Spring WebFlux
    - **Section 6:** R2DBC
    - **Section 7:** Functional Endpoints with Spring WebFlux
    - **Section 8:** Reactive Streams
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Technical Quiz

Attempt the technical quiz in the Learning Path at Tekstac to check your knowledge level of Spring MVC with Spring Boot.

- Spring Boot, Spring MVC

Code Challenge (For Practice Only)

Attempt the following Code Challenge through the Learning Path at Tekstac to check your skill level in Spring MVC with Spring Boot. You need to score 70% or higher to clear this challenge.

Do not copy paste the code. Write the code yourself.

- Assess-Type-1: Code Challenge - Spring MVC with Spring Boot

Course Overview

The **Course 7** of the **Stage 2** provides a comprehensive introduction to Microservices, a modern approach to software architecture that breaks down applications into independent, modular components. GenCs will explore the foundational principles, benefits, and practical applications of Microservices, equipping them to design, build, and manage scalable, resilient systems. The course delves into key architectural patterns, communication mechanisms, and security considerations, alongside insights into anti-patterns to avoid.

Learning Objectives

After completing this course, GenCs will be able to:

- Identify what Microservices are, why they are important, and the principles that guide their design and implementation.
- Evaluate the advantages of adopting a Microservices architecture while recognizing common anti-patterns to avoid.
- Apply Microservices principles to design RESTful APIs and enable effective client-to-microservice communication through API gateways and event-driven systems.
- Understand best practices for securing Microservices, including authentication and data protection.
- Utilize advanced patterns like CQRS, Event Sourcing, and the Saga Pattern to manage data consistency and integrity in distributed systems.
- Learn to deploy and manage Microservices using containerization technologies, including tools like Docker and Kubernetes.
- Gain familiarity with key tools and frameworks used in Microservices development and deployment.

Day 36 - Afternoon

Microservices

Key Topics: Introduction to Microservices, Building Blocks, Data Management, Deployment and Infrastructure

Continuous Learning: Technical Enablement

Microservices Architecture - The Complete Guide

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 2:** History of Microservices
    - **Section 3:** Problems with Monolith & SOA
    - **Section 4:** Microservices Architecture
    - **Section 5:** Problems Solved by Microservices
    - **Section 6:** Designing Microservices Architecture
    - **Section 7:** Deploying Microservices

- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Stage 2 -> Course 8 -> Application Debugging Using IntelliJ IDEA

Course Overview

The **Course 8** of the **Stage 2** provides a comprehensive introduction to debugging, an essential skill for every software developer. You will explore the importance of debugging, learn how to identify and resolve errors in your code efficiently, and become familiar with the tools and techniques that streamline the debugging process. With a focus on IntelliJ IDEA, the course covers setting up a debugging environment, using breakpoints, inspecting variables, and leveraging advanced debugging features such as exception breakpoints, thread debugging, and memory profiling. By the end of the course, you will have the skills to debug complex applications and improve software performance effectively.

Learning Objectives

After completing this course, GenCs will be able to:

- Define debugging and explain its significance in software development.
- Identify common debugging challenges and strategies to overcome them.
- Configure IntelliJ IDEA for effective debugging.
- Navigate the debugging tools and settings within IntelliJ IDEA.
- Set and manage breakpoints to control program execution.
- Run the debugger to step through code and identify errors.
- Inspect variables and expressions to analyze program behavior.
- Use watches to monitor specific values during debugging.
- Implement exception breakpoints to handle runtime errors.
- Perform thread debugging to resolve issues in multithreaded applications.
- Conduct memory and performance profiling to optimize application efficiency.
- Apply debugging techniques to real-world scenarios.
- Use profiling tools to diagnose and resolve memory and performance bottlenecks.

Day 37 - Forenoon

Application Debugging Using IntelliJ IDEA

Key Topics: Introduction to Debugging, Getting Started with IntelliJ IDEA Debugger, Basic Debugging Techniques, Advanced Debugging Techniques

Continuous Learning: Technical Enablement

Java Debugging With IntelliJ IDEA

- Learn All Sections in the following Udemy course.

- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

IDP – Project Activities

Day 37 - Afternoon, 38, 39 - Forenoon

Sprint 2 Development, Integration

- These two days will be spent on Sprint 2 Development and Integration.

Stage 2 -> Course 9 -> ITIL Framework

Course Overview

The **Course 9** of the **Stage 2** provides a comprehensive understanding of the ITIL 4 framework, its evolution from ITIL v3, and its application in modern IT service management. The program is designed to equip GenCs with the foundational knowledge of service management principles, the Service Value System (SVS), and the Service Value Chain (SVC). Participants will explore the roles of organizations, people, technology, and partners in creating and delivering value through service management practices.

Additionally, this course delves into the guiding principles of ITIL 4, offering practical insights into their application. GenCs will gain expertise in service management practices, continual improvement, and the integration of ITIL's best practices into organizational processes.

Learning Objectives

After completing this course, GenCs will be able to:

- Explain the key concepts and principles of ITIL 4.
- Describe the evolution from ITIL v3 to ITIL 4 and its relevance to modern service management.
- Understand the definition, scope, and principles of service management.
- Describe the components and significance of the ITIL 4 Service Value System (SVS).
- Identify the roles of organizations, people, information, technology, partners, and suppliers in the SVS.
- Illustrate value streams and processes within the service management lifecycle.
- Explain the activities and interconnections within the SVC, including planning, improving, engaging, designing, obtaining/building, and delivering/supporting services.
- Understand and apply general management practices such as service level management, service catalog management, availability management, and more.
- Implement ITIL practices like monitoring, deployment, release, and service request management.
- Explain the guiding principles of ITIL 4 and their role in service management.

- Apply principles such as focusing on value, iterative progression, collaboration, simplicity, and optimization in service delivery.
- Understand the continual improvement model and its importance in ITIL 4.
- Apply measurement and reporting techniques to drive improvement initiatives.

Day 39 - Afternoon, 40

ITIL Framework

Key Topics: Introduction to ITIL 4, Service Management, Four Dimensions of Service Management, ITIL 4 Service Value System (SVS), ITIL 4 Service Value Chain (SVC), ITIL 4 Practices, ITIL 4 Guiding Principles, ITIL 4 Continual Improvement

Continuous Learning: Technical Enablement

Introduction to Service Management with ITIL 4

- Go through ALL sections of this course to understand ITIL 4.
- Take up the Practice Exam given as part of this course to check your understanding about ITIL 4.

Quiz - Mandatory

Take up the following quiz to assess your knowledge on the ITIL Framework.

- ITIL Quiz

Stage 2 -> Course 10 -> Jira & ServiceNow

Course Overview

The **Course 10** of the **Stage 2** This course provides a comprehensive introduction to **JIRA and ServiceNow**, two of the most widely used tools for issue tracking, project management, and IT service management (ITSM). GenCs will explore the essential features and capabilities of these platforms, gaining hands-on experience in creating and managing projects, configuring workflows, and utilizing advanced functionalities to streamline processes. The course also covers key concepts of integration, customization, and user management, equipping learners with practical knowledge to apply these tools effectively in software development, IT operations, and beyond.

Learning Objectives

After completing this course, GenCs will be able to:

- Understand the fundamental concepts and benefits of JIRA as an issue tracking and project management tool.
- Create a JIRA account, set up new projects, and navigate the user interface confidently.
- Master JIRA terminology, including issues, projects, and workflows, to effectively manage tasks.
- Create and manage various types of issues such as tasks, bugs, and stories, including assigning, prioritizing, and linking them.

- Search, filter, and organize issues to support efficient project management.
- Understand the concept of workflows and customize them to align with project requirements, including configuring transitions, statuses, and resolutions.
- Manage users, groups, permissions, and security schemes within JIRA.
- Configure settings and customize JIRA to meet organizational needs.
- Gain an overview of ServiceNow as an enterprise cloud platform and understand its evolution and benefits for ITSM and other domains.
- Explore the high-level architecture of the ServiceNow platform and its key components, such as the database, user interface, and business rules.
- Understand ServiceNow's integration capabilities with other systems and tools.
- Familiarize with various modules and applications available in ServiceNow, including ITSM, IT operations management, HR service delivery, and customer service management.
- Identify the features, functionalities, and use cases for different ServiceNow modules and applications.
- Manage user and access permissions, and configure ServiceNow instances to meet organizational requirements.
- Administer ITSM processes such as incident management, problem management, and change management using ServiceNow.
- Utilize the Configuration Management Database (CMDB) and workflows for automating ITSM processes effectively.

Day 41

JIRA, ServiceNow

Key Topics: Introduction to JIRA, Getting Started with JIRA, Working with Issues in JIRA, JIRA Workflows, JIRA Administration

Continuous Learning: Technical Enablement

Jira for Beginners - Detailed Course to Get Started in Jira

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 2:** Concepts You Need to Know in Jira
  - **Section 3:** Using Team-Managed Projects
  - **Section 4:** Using Company-Managed Projects
  - **Section 5:** Jira's Administrative Back End
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Day 42 - Forenoon

JIRA, ServiceNow

Key Topics: Introduction to ServiceNow, ServiceNow Architecture, ServiceNow Modules and Applications, ServiceNow Administration, ServiceNow IT Service Management (ITSM)

Continuous Learning: Technical Enablement

The Complete ServiceNow System Administrator Course

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 2:** ServiceNow Overview
  - **Section 3:** Working With Lists & Forms
  - **Section 5:** Tables & Fields
  - **Section 6:** User Administration
  - **Section 7:** Core Applications
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Stage 2 -> Course 11 -> Windows Service

Course Overview

The **Course 11** of the **Stage 2** provides a comprehensive understanding of **Windows Services** and their integration with Java applications to automate tasks and manage workflows. GenCs will explore the foundational concepts of Windows Services, delve into the principles of job scheduling, and gain hands-on experience in creating, configuring, and managing Java applications as Windows Services. Through practical exercises and real-world examples, GenCs will develop the skills necessary to establish seamless communication between services, handle errors effectively, and implement robust logging mechanisms for monitoring and troubleshooting.

Learning Objectives

After completing this course, GenCs will be able to:

- Explain the role and functionality of Windows Services within the Windows operating system.
- Describe the importance of job scheduling and its applications in task automation.
- Develop and configure a Java application to function as a Windows Service.
- Identify and utilize key job scheduling libraries in Java, such as Quartz and Spring Scheduler.
- Configure job scheduling in Java applications to manage workflows effectively.
- Establish communication channels between Windows Services and Java applications.
- Pass parameters and exchange data between Windows Services and Java applications securely and efficiently.
- Implement error handling strategies in both Windows Services and Java applications to ensure system stability.
- Design and maintain logging mechanisms for event monitoring and troubleshooting purposes.

Day 42 - Afternoon

Windows Service

Key Topics: Introduction to Windows Service and Job Scheduling, Java Integration with Windows Service, Scheduling Jobs in Java, Communication Between Windows Service and Java Application, Error Handling and Logging

Continuous Learning: Technical Enablement

Learn about Windows Service and Job Scheduling from the following:

- Watching Directory
- Watch folder/Directory in Java
- Java Application as Windows Service
- Java Mail Scheduler
- Run Automated Task – fixed interval
- Quartz API

Hands-On

Try out the code snippets given in the above articles.

Do not copy paste the code. Write the code yourself.

IDP - Project Activities

Day 43

Sprint 2 Review & Demo

- This day will be spent for Sprint 2 Review and Demo.

Stage 2 -> Course 12 -> Python 3

Course Overview

The **Course 12** of the **Stage 2** This course provides a comprehensive introduction to **Python**, a versatile and widely-used programming language. Designed for beginners and intermediate learners, the curriculum covers essential programming concepts, Python-specific features, and advanced techniques. GenCs will explore Python's foundational constructs, control flow, data structures, object-oriented programming (OOP), file handling, exception management, and modular programming.

Through practical exercises and hands-on coding, GenCs will develop the skills needed to write efficient, readable, and scalable Python programs.

Learning Objectives

After completing this course, GenCs will be able to:

- Trace Python's version history and identify key features of Python 3.
- Set up and configure the Python environment, including environment variables.
- Execute Python programs via the command line and work with .py files.
- Define and use variables, numbers, strings, and casting in Python.
- Employ string methods to manipulate and process text data.
- Differentiate between Python data types and their use cases.
- Utilize mathematical, assignment, comparison, logical, membership, identity, and bitwise operators effectively.
- Understand operator precedence to write accurate and optimized expressions.
- Implement if, elif, and else conditions with logical and compound expressions.
- Write and manage loops (while, for) with control statements (break, continue).
- Leverage advanced looping techniques like looping through strings and range-based iterations.
- Define and call functions with various types of arguments, including anonymous functions.
- Differentiate between global and local variables and their scopes.
- Employ the return statement to retrieve function outputs.
- Perform operations on lists, tuples, and dictionaries, including indexing, slicing, and built-in methods.
- Understand the properties and behaviors of these data types in Python programs.
- Grasp the principles and characteristics of OOP.
- Create Python classes with properties and methods, using the __init__ function and self argument.
- Implement inheritance and method overriding to build reusable and scalable code.
- Work with file objects, read/write text files, and handle directory structures.
- Utilize file handling methods to perform advanced operations, including renaming and deleting files.
- Identify Python's exception handling features and standard exceptions.
- Implement robust exception handling mechanisms using try, except, finally, and assertions.
- Design custom user-defined exceptions to meet specific program needs.
- Import and utilize modules and packages effectively.
- Understand the role of namespaces, the PYTHONPATH variable, and scoping.
- Leverage built-in functions like dir(), globals(), and reload() for advanced module operations.

Day 44

Python 3

Key Topics: Basics, Variables and Types, Program Flow, Functions

Continuous Learning: Technical Enablement

Complete Python Programming Masterclass Beginner to Advanced

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 2:** Getting Setup with Python

    - **Section 3:** Variables and Types
    - **Section 8:** Python Program Flow
    - **Section 13:** Python Functions
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Welcome Message
- Virtual Assistant
- Hospital Billing
- Stationary Shop
- Theme Park Pricing
- Cash Exchange
- House Rent Calculator
- Airline Reservation System
- Voting System
- Fitness Tracker Analyzer

Day 45

Python 3

Key Topics: Python Operators, Collections, Object Oriented Programming (OOP)

Continuous Learning: Technical Enablement

Complete Python Programming Masterclass Beginner to Advanced

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 4:** Python Operators
    - **Section 13:** Python Collections
    - **Section 14:** Python Object Oriented Programming (OOP)
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Non-Working Doctors
- Product Sales

- Participant Registration and Preference Analysis
- Student Electives
- Sort the Student Details Dictionary
- Generate Customer Id
- Online Shopping Cart System
- Pattern Generator
- Vehicle Premium
- Classic Curve Members
- Area calculation - Method Overrding
- Tournament - Method overloading
- Complex Addition

Day 46, 47 - Forenoon

Python 3

Key Topics: File I/O, Exception Handling, Modules

Continuous Learning: Technical Enablement

Complete Python Programming Masterclass Beginner to Advanced

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 10:** Working With Files
  - **Section 15:** Handling Errors in Python
  - **Section 7:** Python Modules
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Hands-On

Complete the following hands-on given in the Learning Path at Tekstac.

Do not copy paste the code. Write the code yourself.

- Basket Ball Player Analysis
- Percentage of marks - Lamda Functions
- Employee Appraisal
- Product Information
- Filter the Countries Sachin has Played Against - CSV file
- Filter Customers - JSON File
- Data Encryption
- Password Update
- Lucky Number – Generator
- Project Allocation – Iterator
- Summer Camp - Decorator

Code Challenge (For Practice Only)

Attempt the following Code Challenge through the Learning Path at Tekstac to check your skill level in Python 3. You need to score 70% or higher to clear this challenge.

Do not copy paste the code. Write the code yourself.

- Assess-Type-1: Code Challenge - Python

Stage 2 -> Course 13 -> IBM WebSphere and Application Server Configuration

Course Overview

The **Course 13** of the **Stage 2** provides an in-depth understanding of **IBM WebSphere**, a leading enterprise application server platform widely used in the industry for hosting, deploying, and managing Java-based applications. GenCs will explore WebSphere's key components, editions, and use cases, and learn how to configure, deploy, and manage applications effectively. The course combines theoretical concepts with hands-on practice to ensure learners gain both foundational knowledge and practical skills.
The curriculum also emphasizes critical aspects of WebSphere, such as clustering, load balancing, security configurations, and performance optimization, enabling participants to manage and troubleshoot WebSphere environments with confidence.

Learning Objectives

After completing this course, GenCs will be able to:

- Define WebSphere and its role in enterprise application deployment.
- Identify key use cases and distinguish between WebSphere editions, including Base, Network Deployment (ND), and Liberty Profile.
- Compare WebSphere with alternative platforms like Tomcat and WebLogic.
- Explain the basic components: Cell, Node, Server, and Deployment Manager.
- Understand the roles of Node Agents and Application Servers.
- Describe clustering and load balancing and their importance in enterprise applications.
- Perform various types of installations: GUI, command-line, and silent.
- Create and manage WebSphere profiles.
- Start and stop WebSphere servers and utilize the WebSphere Integrated Solutions Console effectively.
- Deploy WAR and EAR files and manage hot deployments and auto-recovery.
- Configure JDBC and JMS resources to enable database and messaging services.
- Perform key application management tasks, including starting, stopping, and uninstalling applications.
- Configure authentication and authorization settings in WebSphere.
- Set up LDAP directories and SSL/TLS for secure communications.
- Troubleshoot issues using WebSphere logs (e.g., SystemOut.log, SystemErr.log) and optimize performance with tools like the Tivoli Performance Viewer.

IBM WebSphere and Application Server Configuration

Key Topics: Introduction to IBM WebSphere Application Server (WAS), WebSphere Architecture Overview, WebSphere Server Installation and Configuration Basics, Application Deployment and Management, Security and Troubleshooting Basics

Continuous Learning: Technical Enablement

Mastering IBM WebSphere 9.x Administration

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
    - **Section 10:** Working With Files
    - **Section 15:** Handling Errors in Python
    - **Section 7:** Python Modules
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Stage 2 -> Course 14 -> Cloud and DevOps Basics

Course Overview

The **Course 14** of the **Stage 2** provides a comprehensive introduction to Cloud Computing and DevOps, two transformative technologies reshaping the IT landscape. GenCs will explore foundational concepts, modern cloud architectures, deployment models, and the essential services offered by leading cloud providers, such as AWS, Azure, and GCP. Additionally, the course delves into DevOps principles and tools, emphasizing collaboration, automation, and continuous integration/continuous deployment (CI/CD) practices for enhanced software delivery. Through practical examples and exercises, learners will gain the knowledge and skills required to leverage cloud platforms and implement DevOps workflows effectively.

Learning Objectives

After completing this course, GenCs will be able to:

- Define Cloud Computing and its key characteristics.
- Explain the architecture and components of Cloud Computing.
- Differentiate between Cloud Computing and Grid Computing.
- Describe how Cloud Computing works and identify real-world applications.
- Compare Public, Private, Hybrid, and Community Clouds, including their advantages and disadvantages.
- Assess use cases and scenarios for each deployment model.
- Explain Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).
- Identify practical applications of each service model.
- Understand the core services offered by AWS, Azure, and GCP.
- Discuss the comparative strengths of these platforms in different business contexts.
- Identify potential security risks associated with Cloud Computing.

- Discuss strategies and best practices to mitigate these risks.
- Define DevOps and its importance in modern software development.
- Understand the role of collaboration between Development and Operations teams.
- Explain Continuous Integration (CI) and Continuous Deployment (CD) processes and their role in the software lifecycle.
- Describe the key features and functionality of Jenkins as a CI/CD tool.
- Create and execute a simple Jenkins job for building and testing a Java project.
- Host Jenkins on cloud platforms like AWS, Azure, or GCP to automate deployments.
- Design and execute a complete CI/CD workflow: Code → Build → Test → Deploy.
- Automate deployments to cloud environments using Jenkins and other DevOps tools.

Day 47 - Afternoon

Cloud and DevOps Basics

Key Topics: Introduction to Cloud Computing, Types of Cloud, Cloud Service Models, Cloud Service Providers

Continuous Learning: Technical Enablement

Introduction to Cloud Computing on AWS for Beginners [2023]

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 3:** IT Fundamentals
  - **Section 4:** Cloud Computing Concepts
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Day 48, 49 - Forenoon

Cloud and DevOps Basics

Key Topics: CI/CD and DevOps Basics with Jenkins

Continuous Learning: Technical Enablement

Jenkins: Jobs, Pipelines, CI/CD and DevOps for Beginners

- Walkthrough the following Udemy course sections and focus on the corresponding topics within our training curriculum's technical scope.
  - **Section 1:** Introduction
  - **Section 2:** Implementing Continuous Integration (CI) with Jenkins
  - **Section 3:** Implementing Continuous Deployment (CD) with Jenkins
  - **Section 5:** Continuous Deployment to Amazon Web Services (AWS)
- Ensure that you learn these topics through self-learning and practice alongside the course instructor. It is NOT necessary to cover every topic comprehensively within each section.

Integrated Capability Test (ICT) (For Practice Only)

Attempt the following Integrated Capability Test (ICT) through the Learning Path at Tekstac to assess your skill level in Stage 2, focusing on Web App Programming skills. You must achieve a minimum

score of 70% to pass this practice test. You will have a maximum of three attempts to complete the practice test.

Do not copy paste the code. Write the code yourself.

- Java Assess-Type-2: Integrated Capability Test (ICT)

Day 49 - Afternoon

- This part of the day has been included to adjust the Behavioral Training duration.

Final Evaluation (Project + Technical)

Day 50, 51, 52, 53 - Forenoon

Final Evaluation (Project + Technical)

- Final evaluation will be conducted on these days.

Final Assessment - HackerRank

Day 53 - Afternoon, 54 - Forenoon

HackerRank Assessment (Core Java, SQL)

- The final assessment will be conducted on this day.

How to learn each day?

Each day has a set of learning objectives. These learning objectives can be met by going through the Udemy courses and by completing the hands-on exercises mentioned in the daily plan.

The below strategies will help you decide the learning approach.

Learning Strategy & Approach

Find below few imaginary profiles. For each of these profiles we have defined a recommended learning approach. This is not an exhaustive list. The approaches below might help invent a new way of learning.

Profile #1

Harry Reacher

Engineering Discipline: Electronics

Skills: Python, Ruby on Rails, nginx

Project: Mining Crime Data to get Route Cause Insights

Learning Approach to Programming Languages: I do not want to waste my time learning. I am more practice oriented. I want to work on the problem immediately

What will work for me?

- Directly complete hands on exercises
- Refer Internet or Udemy Courses
- If hands on are implemented early, clarify your friends questions and troubleshoot their issues

Profile #2

Olivia Richards

Engineering Discipline: Computer Science

Skills: Java, C, C++

Project: Library Management System

Learning Approach to Programming Languages: I have interest, but I don't know where to start.

What will work for me?

- Go through the recommended Udemy Course
- Try completing the hands on exercises
- Get your clarifications solved with help from Tech SME
- Get help from other learners in your batch whom had already completed

Profile #3

Greg Anderson

Engineering Discipline: Civil

Skills: C

Project: Fiber reinforced concrete

Learning Approach to Programming Languages: I am scared of programming languages. I haven't got my hands dirty with coding

What will work for me?

- Go through the recommended Udemy Course
- Implement the coding along with the author of the Udemy Course
- Try completing the hands on exercises
- Clarify queries with SME
- Troubleshoot programming issues with help from SME or learner from your classroom whom had already completed

FAQs

1. **Who can participate in this program?**

   Ans: Students who have enrolled for Full Internship Program (or) the Cognizant on-boarded GenCs can participate in this program.

2. **Is there any pre-learning I should do?**

   Ans: No. This program is open to all students from any academic discipline.

3. **What is Code Challenge?**

   Ans: A problem statement will be provided to you and you need to solve it using a single skill.

4. **What is Integrated Capability Test (ICT)?**

   Ans: A case study problem statement will be provided to you that you may need to solve using the combination of skills learned in the given stage.

5. **How many attempts are provided for the Coding challenge and ICTs? Is it open all the time for practice?**

   Ans: The coding challenges and ICTs are open from day 1, and a maximum of 3 attempts will be provided.

6. **What are the entry criteria for Stage 1 Delta Assessment?**

   Ans: The eligibility criterion for the Stage 1 Delta Assessment is 100% hands-on completion, along with an attempt and submission in both CC and ICT.

7. What skills are covered in the Stage 1 Delta Assessment?

Ans: The delta skills of Stage 1 are covered in this assessment. Maximum THREE attempts will be provided to pass with a minimum score of 70%.

8. What if I fail in the Interim evaluation?

Ans: Your coach will notify your performance in the Interim evaluation. However, you can continue with the learning.

9. How many chances will I get in the Final evaluation?

Ans: You'll get 2 chances in the Final evaluation which covers ALL the skills in the learning journey.

10. Will we be provided with Projects to work on?

Ans: Yes, we will provide the requirement specifications. You will need to implement them by forming a PoD (Pod of Developers). The implementation will be reviewed during both the interim evaluation and the final evaluation.

11. Whom do I reach out in case of any queries?

Ans: Batch Owner is your point of contact.