

# Software Engineering Day1 Assignment

## Part 1

### Introduction to Software Engineering

1 software engineering is a complex and multifaceted discipline that involves the analysis, design, implementation, testing, and maintenance of software systems

Software engineering plays a crucial role in the technology industry as it encompasses the design, development, and maintenance of software systems that power various technologies we use today. As technology continues to advance and evolve at a rapid pace, software engineers are essential in creating innovative solutions to meet the ever-changing needs of businesses and consumers

2 Milestones include the development of programming languages (e.g., Fortran, C), the establishment of software engineering as a discipline in the 1960s, the advent of structured programming in the 1970s, and the rise of agile methodologies in the 2000s

3 Requirements: Gathering and documenting user needs and system requirements.

- Design: Creating high-level and detailed designs of the software architecture and user interface.

- Implementation: Writing code and building the software according to the design specifications.
- Testing: Conducting various tests to ensure the software meets quality standards and functional requirements.
- Deployment: Releasing the software to users or customers.
- Maintenance: Providing ongoing support, updates, and enhancements to the software after deployment.

4 The key difference between Agile vs. Waterfall is that Waterfall breaks down software development into isolated phases that flow into each other, while Agile advocates iterative development cycles in which multiple lifecycle phases can run in parallel.

For example, an electronic device manufacturer may wait until the assemblers finish their work before preparing the project for packaging.

5 Software Developer: Responsible for writing code and implementing software solutions.

Quality Assurance Engineer: Ensures software quality by designing and executing test plans.

- Project Manager: Oversees the planning, execution, and delivery of software projects.

6 Integrated Development Environments (IDES): Software suites that provide comprehensive tools for writing, debugging, and testing code (e.g., Visual Studio, Eclipse, IntelliJ IDEA).

Version Control Systems (VCS): Software tools for tracking changes to source code and coordinating work among team members (e.g., Git, Subversion).

7 Software engineers encounter various challenges throughout the development process, including:

- Changing Requirements: Requirements may change during the development cycle, leading to scope creep and project delays.
- Tight Deadlines: Pressure to deliver software products on schedule can result in rushed

development and compromised quality. - Technical Debt: Accrued from shortcuts or suboptimal solutions, technical debt can impede future development efforts and increase maintenance costs.

Strategies for Overcoming Challenges: Strategies for overcoming challenges include effective communication, agile methodologies, prioritization of tasks, and regular reassessment of project goals and timelines.

8 Importance of Testing: Testing is a critical aspect of QA and involves various types of testing, including:

Unit Testing: Testing individual components or modules of software.

Integration Testing: Testing interactions between different components or subsystems.

- System Testing: Testing the entire software system as a whole.

- Acceptance Testing: Testing the software against user requirements to ensure it meets user needs.

Importance of Quality Control: Quality control measures such as code reviews, automated testing, and continuous integration help identify and fix defects early in the development process, leading to higher-quality software products.

## Part 2

1 Prompt engineering is all about crafting questions or statements to get the best possible responses from AI models.

Prompt engineering helps mitigate bias that may be present from existing human bias in the large language models' training data. Further, it enhances the user-AI interaction so the AI understands the user's intention even with minimal input.

2 Vague Prompts: Phrases like “Tell me everything about...” and in full its tell me everything about Mandela, leave the AI application guessing about your specific needs and it can be improved more effective by using clear and direct wording. Avoiding ambiguity, colourful language, metaphors and slang, all of which can produce unexpected and undesirable results.