

GROUP MEMBERS (GROUP 2)

<i>NAME</i>	<i>REG. NO</i>	<i>SIGNATURE</i>
MIKE MAINA	SC150/0122/2022
BETH NJERI	SC150/0791/2022
BRIAN MURUNGA	SC150/0520/2021
DEBORAH WANGARI	SC150/0102/2022
JOSEPH MWANGI	SC150/0187/2020

GROUP 2

Software Development Life Cycle Software – a collection of executable programming code, associated libraries and documentations.

Development – this is the process of creating or designing the software development life cycle is a process used in software engineering to design, develop and deploy software.

The main aim of the SDLC is to produce high quality software that meets or exceeds customer specifications reach completion within time and cost estimates.

There are several software development life cycle models, each with its own approach to managing the stages of software development. Here are some popular ones:

1. **Waterfall Model:** A linear and sequential approach with distinct phases like requirements, design, implementation, testing, deployment, and maintenance.

2. **Agile Model:** Emphasizes iterative development, collaboration, and customer feedback. Scrum and Kanban are popular Agile methodologies.

3. **Iterative Model:** Similar to Agile, but with fixed iterations. Each iteration produces a potentially shippable product increment.

4. **Spiral Model:** Combines elements of both Waterfall and iterative models. It focuses on risk analysis and incremental development.

5. **V-Model (Validation and Verification Model):** Links testing phases with development phases, ensuring each development stage has a corresponding testing phase.

6. **RAD (Rapid Application Development):** Focuses on rapid prototyping and quick development cycles.

7. **DevOps:** Not a traditional model but a cultural and technical approach that emphasizes collaboration between development and operations teams for faster, continuous delivery.

8. **Incremental Model:** Divides the project into smaller parts or increments, each delivering a portion of the complete functionality.

9. **Big Bang Model:** Development starts with minimal planning and evolves as the project progresses.

10. **Lean Software Development:** Emphasizes reducing waste, improving efficiency, and delivering value to customers.

Each model has its advantages and is suitable for different project types and contexts. The choice of the model depends on factors like project requirements, timelines, and team dynamics.

Question 2:

The SDLC is a process followed for a software project within a software organization. The life cycle defines a methodology for improving the quality of software and overall development process. The SDLC is a structured process that enables the production of high-quality, low-cost software, in the shortest possible production time. The software development cycle involves the following stages:

- I. Planning
- II. Feasibility Study
- III. Requirement analysis
- IV. System design
- V. Implementation (Coding)
- VI. Testing
- VII. Deployment
- VIII. Maintenance and Support
- IX. Documentation
- X. Training
- XI. Change Management
- XII. Quality Assurance

XIII. Risk Management

XIV. Communication and Collaboration

1. ***Planning:** Define the project scope, objectives, and requirements. Create a project plan, allocate resources, and establish timelines and budgets.
2. ***Feasibility Study:** Evaluate the technical, economic, and operational feasibility of the project. Determine if it's worth pursuing.
3. ***Requirements Analysis:** Gather and document detailed requirements from stakeholders. Define what the software must do and its constraints.
4. ***System Design:** Create a high-level system architecture and detailed design specifications. Decide on technologies, databases, and overall system structure.
5. ***Implementation (Coding):** Write the actual code for the software based on the design specifications. This is the development phase where developers write, test, and debug the code.
6. ***Testing:** Verify that the software meets the specified requirements. This involves unit testing, integration testing, system testing, and user acceptance testing (UAT).
7. ***Deployment:** Release the software to the production environment or distribute it to end-users. This includes installation, configuration, and data migration if necessary.
8. ***Maintenance and Support:** After deployment, the software requires ongoing maintenance, bug fixes, updates, and support to ensure it continues to function correctly.
9. ***Documentation:** Create user manuals, technical documentation, and any necessary guides for end-users and developers.

10. ***Training:** Train end-users and support staff on how to use and maintain the software.
11. ***Change Management:** Manage changes and updates to the software, ensuring they are properly tested and implemented.
12. ***Quality Assurance:** Continuously monitor and improve the quality of the software and development processes.
13. ***Risk Management:** Identify and mitigate risks throughout the SDLC to ensure the project stays on track.
14. ***Communication and Collaboration:** Effective communication and collaboration among team members, stakeholders, and customers are essential throughout the SDLC.
15. ***Feedback and Iteration:** Gather feedback from users and stakeholders to make necessary improvements and enhancements to the software.
16. ***Closure:** Evaluate the project's success, document lessons learned, and officially close the project.

These activities can vary in sequence and emphasis depending on the specific SDLC model being used (e.g., Waterfall, Agile). The goal is to produce a high-quality software product that meets user needs and business objectives while managing risks and resources effectively.

NAME:.....

SIGNATURE:.....