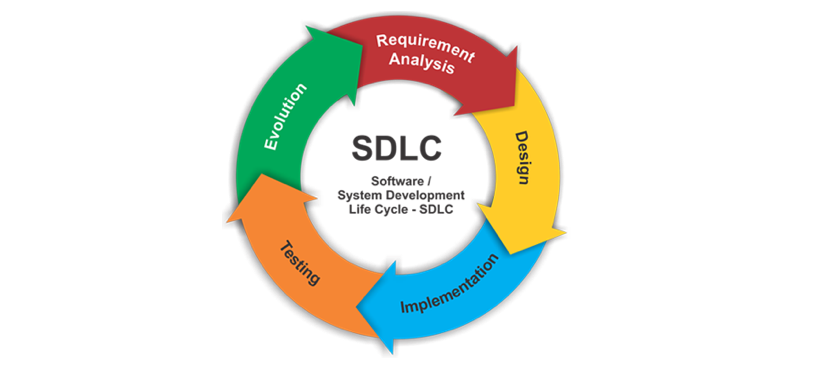
***Software development life cycle: Day-1\_Assignments Name : Ch.Vinay Kiran***

***Assignment 1: SDLC Overview - Create a one-page info-graphic that outlines the SDLC phases (Requirements, Design, Implementation, Testing, Deployment), highlighting the importance of each phase and how they interconnect.***

* *The Software Development Life Cycle: This Software development life cycle (SDLC) is a process, which is used to design, develop, testing, deploying and maintenance of a software projects by the software development teams. The life span of the software project is within the software development life cycle, as all the implementations, developments, designing, testing, deployment and maintenance are performed in the software development life cycle.*
* *The software development life cycle consists of several phases, each phase have its own set of activities, deliverable and objectives.*
* *The main phases of the software development life cycle are: Requirements, Design, Implementation, Testing, Deployment.*

**

* *Requirements Phase: In the requirement phase the gathering and analyze the clients requirements to understand the use, objectives and functionality of the software application are carried out.*
* *The requirements can be accessed by some activities like Interview stakeholders, conducting surveys, conducting workshops, verifying the documentations, and other use cases.*
* *In the software development life cycle the requirement phase is very vital and building block or foundation for all subsequent phases of the software development life cycle. They guide the design, development, testing, and deployment of the software project.*
* *Any errors or faults in the requirement phase can cause the whole software project failure or fault software project.*
* *Design Phase: The Second phase in the software development is the design phase, The design phase is used to create a high-level and detailed design of the software application based on the requirements gathered in the previous phase.*
* *The design phase defines the system architecture, data models, user interface designs, and software components.*
* *In the design phase, creating prototypes, diagrams to visualize the software design is carried out.*
* *The requirements are translated into a software application's blueprint during the design process. It provides information to developers about the organization, communication, and data flow of the program.*
* *Any faults in the design phase will affect the subsequent phases but not the requirement phase.*
* *Development Phase: In the development phase the execution and implementation of the software program is performed according to the requirements and design criteria.*
* *Code writing, software component development, integrating third-party libraries or APIs, and code reviews are among the tasks involved.*
* *By converting design specifications into executable code, development expands on the design phase. To make sure that the developed features fulfill user expectations and the intended functionality, developers consult design papers and requirements.*
* *Any faults in the development phase will affect the subsequent phases but not the requirement phase and design phase.*
* *Testing Phase: The testing phase ensure that the software program satisfies user expectations, functional requirements, and quality standards by validating and verifying it.*
* *Activities include creating and carrying out test cases as well as unit, integration, system, and acceptability testing. Determine and disclose any flaws or problems.*
* *Testing verifies if the software generated satisfies the prerequisites outlined in the preceding stages. Requirements and design specifications are the source from which test cases are created in order to guarantee thorough coverage of all application functionality.*
* *Deployment Phase: The deployment phase will release the software program so that end users can access and utilize it in staging or production environments.It create deployment packages, set up servers, carry out deployment duties, keep an eye on the deployment procedure, and carry out testing after deployment. The move from development to production settings is signified by deployment. To guarantee a successful release to end users, it entails confirming that the deployed application complies with the specifications, design, and testing results.*
* *There is a constant feedback loop and inter-connectivity between phases in the SDLC. Design decisions are influenced by requirements, which in turn influence development chores. Development generates code for testing, which verifies that the program satisfies the required specifications and design standards. The process comes to a close with deployment, which guarantees that the software is released into production environments after extensive testing and validation.*

***Assignment 2: Requirements Gathering - Conduct a 30-minute mock interview to gather requirements for a fictional app that helps organize community events. Summarize the requirements and how you would document and trace them in a one-page brief***.

## *Software development life cycle (Requirement & Analysis)*

*Requirement & Analysis - Documentation for Online Shopping application (Amazon)*

*Introduction*

*This document outlines the Software Development Life Cycle (SDLC) - Requirement & Analysis Phase for developing the Online Shopping application - Amazon, detailing the Requirement & Analysis Phase, activities, and deliverable involved in the gathering the requirements. In the requirement phase the gathering and analyze the clients requirements to understand the use, objectives and functionality of the software application are carried out, ensuring that projects are planned, executed, and delivered effectively. In the software development life cycle the requirement phase is very vital and building block or foundation for all subsequent phases of the software development life cycle. They guide the design, development, testing, and deployment of the software project.*

*Overview*

*Project Scope*

*Software development life cycle (Requirement & Analysis) our project aims to gather the requirements needed to design, develop and maintain the Online Shopping application - Amazon.*

*Requirement & Analysis*

*1. Identify Stakeholders:*

* *Customers: Regular shoppers who use Amazon for purchasing a wide range of products.*
* *Sellers: Individuals or businesses who list their products on Amazon's platform.*
* *Internal Teams: Teams within Amazon responsible for product management, marketing, customer service, logistics, and technology development.*

*2. Conduct Interviews and Surveys:*

* *Interview customers to understand their shopping habits, preferences, pain points, and expectations from an online marketplace.*
* *Survey sellers to gather feedback on their experience with listing products, managing orders, and interacting with customers on Amazon's platform.*
* *Engage with internal teams to gather insights into business goals, technical capabilities, and operational requirements.*
* *3. Analyze Data:*
* *Analyze website and app usage data to identify popular features, user behaviors, and areas for improvement.*
* *Use analytic tools to track customer interactions, such as product searches, purchases, and reviews, to understand user preferences and trends.*

*4. User Personas:*

* *Create user personas representing typical Amazon customers and sellers to better understand their demographics, preferences, and needs.*
* *Use these personas as a reference point when defining requirements to ensure they meet the needs of different user segments.*

*5. Feedback Channels:*

* *Provide channels for customers and sellers to submit feedback and suggestions directly through the Amazon website, app, or customer service channels.*
* *Monitor social media, forums, and review platforms to gather additional insights and feedback from users.*

*6. Prototype and User Testing:*

* *Develop prototypes or mock-ups of new features or enhancements based on the gathered requirements.*
* *Conduct user testing sessions with real customers and sellers to gather feedback on the prototypes and validate that they meet user needs.*

*7. Iterative Approach:*

* *Take an regularly review and update requirements documentation to ensure alignment with evolving stakeholder needs and market trends.*

*Conclusion*

*The Requirement & Analysis document gives the design, development, testing, deployment & maintenance phases a clear road-map and their requirements for managing the software development process, ensuring that projects are planned, executed, and delivered effectively. And this are the different approaches to gather the requirements and the analysis for the design, develop and maintain of the online shopping application - Amazon.*

***Assignment 3****:* ***Agile Principles Application - Write a two-paragraph reflection on how the Agile values of individuals and interactions, working solutions, and customer collaboration apply to the development of the community event app.***

* *Agile Methodology: Agile methodology is an adaptable and iterative approach to software development that prioritizes regular delivery of small, incremental releases over a single, substantial release at the conclusion of the development cycle. Collaboration, flexibility, and client satisfaction are prioritized at every stage of the development process.*
* *The popular Agile frameworks are scrum, kanban, each contains its own set of principles, rituals and practices. Scrum, for example, employs roles like Product Owner, Scrum Master, and Development Team, along with specific ceremonies such as Sprint Planning, Daily Stand-ups, Sprint Review, and Sprint Retrospective to facilitate the Agile process.*
* *The Agile Principles: The Agile principles serve as the main role for Agile methodologies like kanban, Scrum. By the help of the Agile principles we will achieve customer satisfaction through early and continuous delivery of software applications. As the agile prioritizes delivering working software at early and frequently to provide value to the customers and gather feedback for iterative improvements.*
* *The agile accepts change as a competitive advantage and understands that needs can alter over time. It promotes adaptability and flexibility to meet shifting needs.*
* *Deliver working software frequently with a preference for the shortest timescale. Agile emphasizes collaboration between business stakeholders and development teams to ensure alignment, clarify requirements, and make timely decisions.*
* *Building projects around motivated individuals is the crucial principle in the agile as it helps the development team as they need, and trust them to get job done.*
* *Agile values motivated and empowered teams, it enables the teams members to form a strong trust and form a supportive environment.*
* *Create Projects centered on Motivated People. Trust them to do the job and provide them with the environment and assistance they require: Agile emphasizes empowered and driven teams. It creates an atmosphere of support where people are trusted, given freedom, and given the tools they need to be successful.*
* *Face-to-Face Conversations are the Most Efficient and Effective Way to Communicate with and Within a Development Team: Agile places a strong emphasis on direct communication and faces-to-face interactions as a means of reducing misunderstandings, fostering teamwork, and fostering rapport.*
* *Utilizing Software is the Main Way to Assess Advancement: Agile places a strong emphasis on producing usable software increments as the main indicator of success, allowing stakeholders to evaluate value delivery and make defensible decisions.*
* *Scrum Methodologies Encourage Sustainable Growth. It should be possible for the developers, sponsors, and users to keep up a steady pace indefinitely. Sustainable development methods that put the team's welfare first and support a consistent work tempo throughout time are encouraged by agile.*
* *Constant Focus on Technical Excellence and Good Design Promotes Agility: Agile places a strong emphasis on the value of these elements in order to preserve flexibility, reduce technical debt, and enable future improvements.*
* *The art of maximizing the amount of work not done, or simplicity, is crucial. Agile promotes simplicity in both design and implementation, motivating teams to provide the most important features and reduce superfluous complexity.*
* *The Agile values of individuals and interactions, working solutions, and customer collaboration to the development of a community event app:*
* *Interacting with the people among the community:*
* *Within the community event app development setting, this Agile ideal highlights how crucial it is to cultivate good teamwork and communication. This can be accomplished by promoting transparent channels of communication among the development team to enable the exchange of thoughts, criticism, and worries.   
  encouraging a climate of mutual respect and trust in which team members are free to voice their concerns and thoughts.*
* *Setting up regular meetings, like weekly sync-ups or daily stand-ups, can help to guarantee that team members are in agreement regarding the objectives, status, and difficulties of the project.*
* *Whenever possible, maintain a priority on in-person or virtual encounters to foster trust and improve collaboration.*
* *Working Solutions over Detailed Documentation: Put more effort into developing a working app that satisfies community demands right away rather than wasting time on producing extensive documentation up front. This entails segmenting the development process into more manageable jobs and releasing over time functional features. Present the app to stakeholders on a regular basis and get their input so you can refine and enhance it.*
* *Collaboration with Customers Over Contract Negotiation: During the apps development, interact with the community and potential consumers. Through surveys, user interviews, and usability testing sessions, get their feedback on functionality, usability, and design preferences. You may make sure that the app closely satisfies users' wants and expectations by including them in the development process.*

***Assignment 4: Scrum Framework Overview - Prepare a one-page cheat sheet on the Scrum framework that includes roles, responsibilities, artifacts, and ceremonies. Provide a brief example of a Sprint task list for the earlier mentioned app project.***

* *Scrum: The framework known as Scrum is mostly applied to agile software development. It offers a framework for groups to collaborate and work iteratively to create complicated products. Scrum was initially created to manage software development projects, but it has now found use in a variety of domains outside of software, such as marketing, research, and cutting-edge technology.*
* *Roles: The Product Owner, the Scrum Master, and the Development Team are the three main roles that Scrum defines. In addition to representing the stakeholders, the Product Owner is in charge of optimizing the product's value. The Scrum Master removes obstacles and helps the Scrum process along. The delivery of the product increment is the responsibility of the development team.*
* *Responsibilities: The scrum framework contains three roles, product owner, the scrum master and the development team each role has a specific responsibilities.*
* *The product owner responsibility is to define and prioritize the features of the product, maintain and regularly refine the product backlog, communicate the vision goals of the product to the development team, should maintain and make decisions regarding the features and implementation to be included in each sprint.*
* *The Scrum master responsible to ensure the scrum framework is understood and followed by the team. Train the team to become self-organized and cross-functional.*
* *The Development team responsible for self-organize to accomplish the work in progress in the sprint backlog. Collaborate with the product owner to understand the requirements of the product backlog items. Join in the daily scrum meetings and provide feedback on the process and the development of the project.*
* *Artifacts: Scrum employs three primary artifacts, namely the Product Backlog, Sprint Backlog, and Increment. A prioritized list of the product's features, modifications, improvements, and problem fixes is called the product backlog. A portion of the Product Backlog items chosen for the current Sprint make up the Sprint Backlog. At the conclusion of a Sprint, the total number of finished Product Backlog items is known as the Increment.*
* *Ceremonies: Scrum outlines a number of events to help the team communicate and work together. Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective are a few of these. The team arranges the tasks to be completed in the next Sprint during Sprint Planning. Team members coordinate their work and make plans for the following day during the brief daily meeting known as the Daily Scrum. At the conclusion of the sprint, the Increment is inspected, and if necessary, the Product Backlog is adjusted, at the Sprint Review. Following the Sprint Review, a meeting known as the Sprint Retrospective is held to evaluate the previous Sprint and pinpoint areas for improvement.*
* *The example of the sprint task list for the earlier mentioned app project in the scrum framework overview:*
* *Sprint Goal: Implement user authentication and basic user profile functionality*
* *Sprint Duration: 4 Weeks*
* *Sprint Tasks Design and Develop: Review and improve user stories pertaining to profile functionality and user authentication. Divide up user stories into manageable tasks. Construct an Authentication System: Look into and choose a framework or library for authentication. Implement the features for user registration, login, and logout. Handle edge cases and authentication problems.*
* *Create a database schema for user profiles in order to implement user profiles.   
  Establish API endpoints to get and modify user profiles. Include required data in the user profile, including name, email, and avatar.*
* *Create and implement user interface elements for the profile, login, and sign-up pages. Make sure the user interface is accessible and responsive.*
* *Sprint Task Testing and maintenance: Create unit tests to verify profile functionality and authentication.   
  To guarantee seamless communication between the front-end and back-end, conduct integration testing.   
  Verify user login and profile functionality by conducting user acceptance testing.*
* *Add information on authentication and profile capabilities to the project's documentation. Give developers guidance on how to operate the authentication mechanism. Re-factoring and bug fixes: Take care of any problems or defects found during testing. Re-factor code to make it more scalable and maintainable.   
  Review and Demo: To guarantee code quality and conformity to coding standards, conduct internal code reviews.   
  Present the completed features to interested parties and get their input.*
* *The team should be able to move steadily closer to reaching the Sprint Goal if each work is manageable and can be finished in a day or two.*
* *The team works closely together during the sprint, meets daily to monitor progress, and makes necessary adjustments to offer the best product increment by the conclusion of the sprint.*

***Software development life cycle: Day-2\_Assignments Name: Ch.Vinay Kiran***

***Assignment 1: Agile Project Planning - Create a one-page project plan for a new software feature using Agile planning techniques. Include backlog items with estimated story points and a prioritized list of user stories.***

* *Agile Methodology: Agile methodology is an adaptable and iterative approach to software development that prioritizes regular delivery of small, incremental releases over a single, substantial release at the conclusion of the development cycle. Collaboration, flexibility, and client satisfaction are prioritized at every stage of the development process.*
* *The project plan for implementing a new software feature or an add-on (In gaming sector) using agile planning techniques. In this project we are going to implement a new software feature “Player Customization” feature to an existing game, where in place of default character now players can create, edit and customize there existing character with various outfits, hairstyles and accessories.*
* *Project Name: Player character customization, add-on and features.*
* *Project Vision: By the help of player customization it will enhance the user experience and engagement by allowing players to personalize their in-game characters.*
* *The Agile Planning: As the agile planning is a flexible and iterative approach to project management, so it is applicable not only to the software development but also in different industries.*
* *Sprint Planning: The first task in the agile methodology is the sprint planning in the sprint planning we implement our goals, focusing on specific user stories, and in the sprint planning there is certain fixed time duration.*
* *Duration : 2 weeks per sprint task.*
* *Sprint Goal: Implement basic player customization functionally.*
* *Backlog Refinement: The backlog refinement is an ongoing process in agile project in agile project management where the product backlog is reviewed, revised and prioritized to ensure that it contains the most valuable and well defined items for upcoming sprints.*
* *Duration: Ongoing, scheduled before each sprint planning meeting.*
* *Refinement Activities: Prioritization of backlog items, breaking down epics into user stories, estimating story points.*
* *Backlog items with estimated story points:*
* *Epic 1: Player Customization*
* *Story 1: As a player, I want to choose different outfits for my character.*
* *Story Points: 5*
* *Story 2: As a player, I want to select various hairstyles for my character.*
* *Story Points: 3*
* *Story 3: As a player, I want to pick accessories (hats, glasses, etc.) for my character.*
* *Story Points: 3*
* *Epic 2: UI/UX Enhancements*
* *Story 1: As a player, I want the customization interface to be intuitive and user-friendly.*
* *Story Points: 8*
* *Story 2: As a player, I want to preview my character with different customization before confirming.*
* *Story Points: 5*
* *Epic 3: Persistence*
* *Story 1: As a player, I want my chosen customization to be saved and applied every time I log in.*
* *Story Points: 5*
* *Prioritized List of User Stories: We've prioritized the backlog based on user needs and story points.*
* *I want to dress my character differently as a gamer.(5 SP)*
* *As a player, I want the customization interface to be simple and user-friendly. (8 SP)*
* *As a player, I want to select various hairstyles for my character.(3 SP)*
* *As a player, I want to pick accessories.(3 SP)*
* *As a player, I want to preview my character with different customization before confirming.(5 SP)*
* *As a player, I want my chosen customization to be saved and applied every time I log in.(5 SP)*
* *Sprint Planning (Sprint 1): We'll work in two-week sprints, focusing on specific user stories. Sprint 1 will focus on implementing basic customization features.*
* *Goal: Implement basic player customization functionality*
* *Selected User Stories:*
* *Story 1: As a player, I want to choose different outfits for my character.*
* *Story 2: As a player, I want to select various hairstyles for my character.*
* *Story 4: As a player, I want to pick accessories (hats, glasses, etc.) for my character.*
* *Estimated Capacity: The team will estimate their capacity based on historical velocity and team availability.*
* *Sprint Review and Retrospective: Each sprint will conclude with an assessment of our accomplishments and an identification of areas for improvement.*
* *Duration: End of each sprint*
* *Review: Demonstrate completed user stories to stakeholders.*
* *Retrospective: Identify what went well, what could be improved, and action items for the next sprint.*
* *Sprint Backlog: List of tasks derived from selected user stories, assigned to team members for implementation during the sprint.*
* *This plan makes sure the team focuses on providing value to the users in an incremental and iterative manner by providing an organized strategy to integrating the new software feature in gaming through Agile planning approaches*.

***Assignment 2: Daily Stand-up Simulation - Write a script for a Daily Stand-up meeting for a development team working on the software feature from Assignment 1. Address a common challenge and incorporate a solution into the communication flow***.

* *Daily Stand-up meeting:*
* *The daily stand-up simulation for the player character customization feature project (Assignment - 1) is the meeting conducted by the scrum master with the project development team and the stakeholders.*
* *In the daily stand-up meet would be a brief, daily check-in where development team members provide updates and the challenges they face on their progress, discuss any challenges and faults they’re facing, and plan their tasks for the day.*
* *Daily Stand-up simulation - Round of updates: The project owner and the scrum master will conduct the daily stand up meeting for the player character customization feature project. The purpose of the meeting is to sync up on the progress, identify any roadblocks, and ensure the development team aligned towards the project goals.*
* *The Project owner/Scrum master will conduct the daily stand-up meet for each individual team member about there challenges and the progress of the tasks assigned from them for a single day.*
* *Each development team member will be addressed on there progress like the tasks for the project player’s character customization feature like implementing the back-end logic for saving players customization choices. And integrating that logic with the front-end user interface.*
* *Another member from the development team will address there day tasks like designing the UI for selecting outfits, and encountered challenge with ensuring responsiveness across the different screen sizes, plan to refine the UI and coordinate with the front-end team to address the challenge.*
* *Daily Stand-up simulation - Addressing the challenge: After the round of updates, if there is any challenge occurred the update round will be converted to the address the challenge and the daily stand-up meet will address the challenges faced by the individual team members or the whole team, and the challenge will be addressed by scheduling a meeting to sync up with the front-end team and brainstorm solutions together.*
* *After the challenges are cleared or noted the daily stand-up will continue the update round with the rest of the development team if any challenge is faced by any member then the addressing of the challenge will take place.*
* *And some team members will work on the fixing of the bug related to character customization not persisting across sessions. And other visual bugs will be fixed.*
* *Daily Stand-up simulation - wrap-up: The daily stand-up meet come to a conclusion after address all the development team members progress and challenges.*
* *The daily stand-up ensures that everyone in the development is on the same page and sync, promotes transparency and fosters collaboration within the team environment. It’s a key ritual in agile methodology to keep projects moving forward efficiently.*