**Software Development Life cycle**



**State all the SDLC phases: explain each phase, its role in software development and its input & output**

**Analysis**:

Explanation: in this phase, start collect and understand the project requirements (stories) and its conditions from clients, whose will work in this phase should have high understood of business terms.

Role: in this phase the team meat with client and gathering requirements and its conditions and try to put those requirements in the form which can understood by developers.

Input: meeting with the clints, who will start explaining their requirements

Output: list of stories (requirements), its steps and conditions. (product backlog)

**Design:**

Explanation: in this phase we translate the stories as graphics which the user can see, and plan how they can take action on.

Role: architects and designers create the system's architecture, data models, user interface (UI) design, and overall structure.

Input: list of stories (requirements) and its steps and conditions.

Output: UI/ UX Designers, system architecture

**Development:**

Explanation: in this phase, developers start code the requirements and its conditions, rather front end and backend develop.

Role: developers write code based on the design specifications.

Input: UX/UI design of project

Output: codes of whole project of all features and stories

**Testing:**

Explanation: in this phase, test the performance of site and ensures that the software works as expected and meets the requirements.

Role: testers execute various tests to identify issues or problems in the software.

Input: codes of site

Output: issues or errors in software

**Deployment:**

Explanation: in this phase, distribute the site to use in the real world, by select the domain of the site and hosting.

Role: select domain and hosting the software.

Input: code files of sites

Output: link of site

**Maintenance:**

Explanation: in this phase, make improvements to the site based on reports from user or tester after the site is published on the internet.

Role: defines user feedback, adds new features, and ensures the software remains secure and efficient.

Input: site issues, new features

Output: new version of site with improvements or repots to help user to improve the user experience

**State all the roles / titles that you know in the SDLC, their responsibilities and tools that they used if any**

**Product Owner:**

Responsibility: translate the requirements in the way like steps the developers can understand with priority and their estimation.

Tool: Jira, Trello

**Project Manager:**

Responsibility: who coordinate and manage the activities, processes, and resources needed to complete a project, such as time development of project, financial development and so on.

Tool: Microsoft Project, Asana, Gantt charts

**Business Analyst**:

Responsibility: who stay with clients and collect the requirements and then analyst those requirements.

Tool: diagramming tools (Visio)

**CTO:**

Responsibility: who oriented of project and he more advisory than executive and responsible of the technology of the project and also manage all the team.

Tool: Management project

**UX/UI designer**

Responsibility: design the graphics of site and also the actions.

Tool: Design tools (Figma, Adobe XD, Sketch), prototyping tools

**System Architect**

Responsibility: who design the code such as design structure, components, modules, and relationships. And also design the database, such as how the data will store. sometime this role done by the senior developer.

Tool: Architecture modeling tools (UML tools), cloud platform tools (AWS, Azure).

**Front end developer**

Responsibility: Develops the client-side of the software, who implements the UI design using HTML, CSS, and JavaScript.

Tool: Code editors (VS Code), front-end frameworks (React, Angular), browser developer tools.

**Backend developer**

Responsibility: Develops the server-side of the software. Who handles data processing, logic, and database interactions.

Tool: Code editors, back-end frameworks (.NET), database management tools (MySQL).

**Solutions Architect**

Responsibility: test and implements solutions of code structure which meet specific business needs. Thus, focuses on integrating different systems and technologies related of project.

Tool: Architecture modeling tools, integration platforms.

**DevOps**

Responsibility: Manages infrastructure, deployment, and monitoring.

Tool: GitLab, Docker

**Data administrator**

Responsibility: Storage and disrepute the data and the condition to access those data

Tool: MySQL, PostgreSQL and MongoDB.

**Tester**

Responsibility: test the site, he explores some of problems in the site.

Tool: Jira

**Support managers**

Responsibility: receive the problems from users and give to the developer's team.

Tool: Slack, AnyDesk and teams