Rupal Manwatkar

SDLC ASSIGNMENT

**ASSIGNMENT 1**

1. SDLC overview creates a one-page infographic that outlines the SDLC phases(requirements, design, implementation, testing, deployment) highlighting the importance of each phase and how they interconnect. the application is Gpay

**SDLC Overview Of GPay**

**Requirements**

Description:

* + - Gather and analyze user needs
    - Define market demands
    - Create user stories and objectives
  + Interconnection: Foundation for design and development

**Design**

Description:

* + - Create system architecture
    - Develop wireframes and mockups
    - Design data models
  + Interconnection: Blueprint for implementation

**Implementation**

Description:

* + - Write and integrate code
    - Set up databases
    - Initial configurations
  + Interconnection: Functional system ready for testing

**Testing**

Description:

* + - Conduct unit, integration, and system tests
    - Perform user acceptance testing
    - Identify and resolve defects
  + Interconnection: Ensure reliability and security

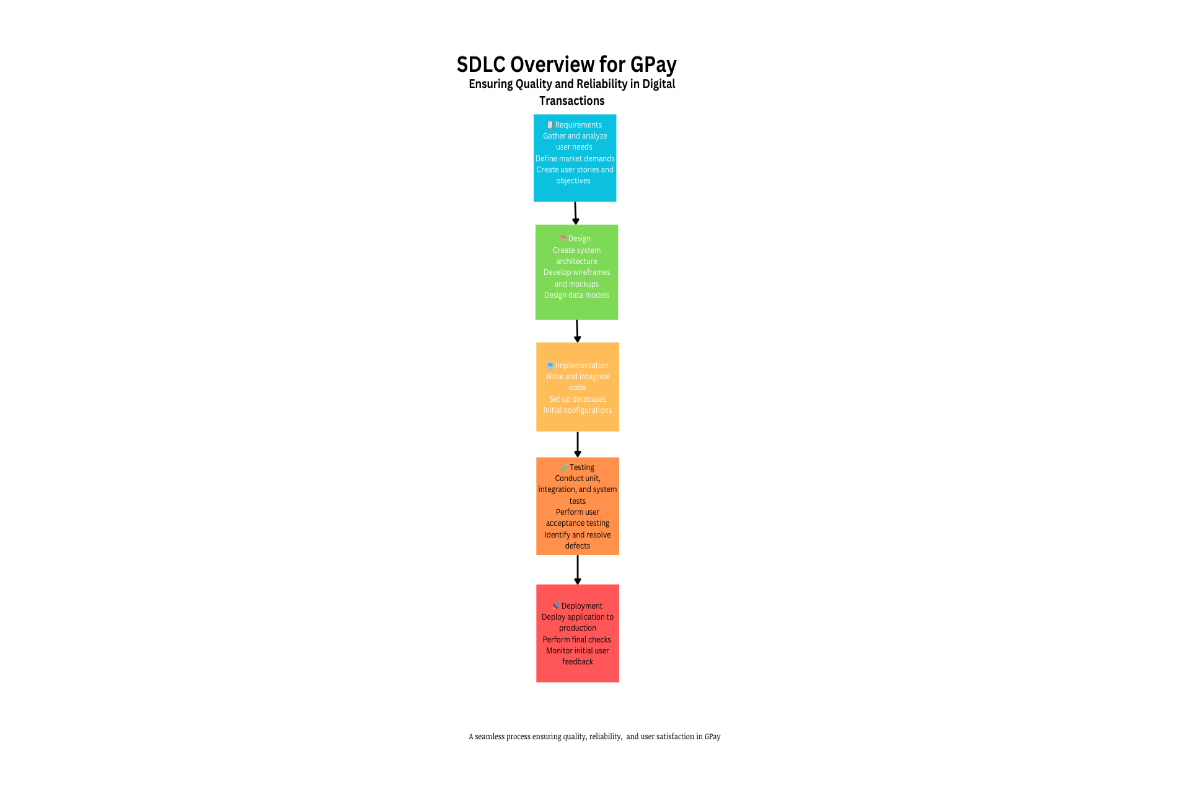
**Deployment**

Description:

* + - Deploy application to production
    - Perform final checks
    - Monitor initial user feedback
  + Interconnection: Complete the cycle, feed into future requirements

**Footer**

* Callout: A seamless process ensuring quality, reliability, and user satisfaction in GPay.



**ASSIGNMENT 2**

1. 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

Requirements Gathering for Community Events Organization App

Mock Interview Summary:

**Interviewer:** Could you describe the main purpose of the app?

**interviewee:** The app should help users organize community events efficiently by providing tools for event planning, management, and communication.

**Interviewer:** What specific features do you envision for this app?

**interviewee:**

1. Event Creation and Management:

* Users should be able to create events with details like title, date, time, location, and description.
* Options for recurring events.

1. RSVP and Attendance Tracking:

* Users should RSVP to events.
* Organizers should be able to track attendance and send reminders.

1. Notifications and Reminders:

* Push notifications for upcoming events, changes, or cancellations.
* Email and SMS reminders.

1. Communication Tools**:**

* In-app messaging for organizers and participants.
* Discussion boards for events.

1. Resource Management**:**

* Booking and managing resources like venues, equipment, and volunteers.

1. Integration with Social Media**:**

* Share events on platforms like Facebook, Twitter, and Instagram.

1. User Profiles and Roles**:**

* Different user roles (organizers, participants).
* User profiles with basic information and past event history.

1. Feedback and Reviews**:**

* Participants can leave feedback and rate events.

1. Analytics and Reporting**:**

* Event analytics (attendance rates, engagement levels).
* Exportable reports.

**Interviewer:** Are there any security or privacy considerations we should keep in mind?

**interviewee:**

* Secure login and authentication.
* Data encryption for sensitive information.
* Compliance with privacy laws (e.g., GDPR).

**Interviewer:** Who are the primary users of the app?

**interviewee:**

* Community organizers.
* Event participants (community members).
* Volunteers.

**Documentation and Tracing Requirements**

**1. Requirement Documentation:**

* **Functional Requirements:**
  + Create, manage, and categorize events.
  + RSVP functionality and attendance tracking.
  + Notification and reminder systems.
  + In-app communication and discussion boards.
  + User profiles with roles.
  + Feedback and review system.
  + Event analytics and reporting.
* **Non-Functional Requirements:**
  + Security (secure login, data encryption).
  + Privacy compliance.
* **User Requirements:**
  + User-friendly interface for both organizers and participants.
  + Accessible on multiple devices (web, mobile).
  + Reliable notification system.

**2. Tracing Requirements:**

* **Traceability Matrix:**
  + Create a traceability matrix to map each requirement to its corresponding design, development task, test case, and user story.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID** | **Description** | **Design** | **Development** | **Testing** |
| 01 | Event Creation & Management | D-01 | Dev-01 | Test-01 |
| 02 | RSVP & Attendance tracking | D-02 | Dev-02 | Test-02 |
| 03 | Notification & Remainders | D-03 | Dev-03 | Test-03 |
| 04 | Communication tools | D-04 | Dev-04 | Test-04 |
| 05 | Resource Management | D-05 | Dev-05 | Test-05 |
| 06 | Social Media | D-06 | Dev-06 | Test-06 |
| 07 | User Profiles & roles | D-07 | Dev-07 | Test-07 |
| 08 | Feedback & Reviews | D-08 | Dev-08 | Test-08 |
| 09 | Analytics & Reporting | D-09 | Dev-09 | Test-09 |
| 10 | Secure Login | D-10 | Dev-10 | Test-10 |
| 11 | Data encryption | D-11 | Dev-11 | Test-11 |

**Version Control and Stakeholder Review**

* Maintain a version-controlled document to track changes.

**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

* The Agile values of individuals and interactions are central to the development of the community event app, as they prioritize effective communication and teamwork over rigid processes. In the context of this project, fostering a collaborative environment allows team members to leverage their diverse skills and perspectives, leading to more innovative solutions. Regular interactions, such as daily stand-ups and sprint reviews, ensure that everyone is aligned with the project's goals and can quickly address any challenges that arise. This focus on people over processes helps create a dynamic and adaptable development process, crucial for responding to the evolving needs of the community and the event organizers.
* Working solutions and customer collaboration are equally vital in this endeavor. By prioritizing functional software and continuous feedback from users, the development team can ensure that the app meets the actual needs of its end-users. Engaging with community members and event organizers throughout the development process enables the team to gather valuable insights and iterate on the app's features in real time. This ongoing dialogue not only ensures that the app remains relevant and user-friendly but also fosters a sense of ownership and satisfaction among stakeholders. Ultimately, the Agile values guide the team towards creating a practical, user-centered app that effectively serves the community's needs