Design Principles

Design principles are fundamental guidelines that help create designs that are user-friendly, aesthetically pleasing, and effective.

1. **Contrast:** Contrast refers to the use of differences in visual properties to make elements stand out from each other.
Application: Use contrasting colors, sizes, shapes, and fonts to create visual interest and guide the user's attention.

2. **Repetition:** Repetition involves using the same elements or styles throughout the design to create a sense of unity and consistency.
Application: Repeat visual elements like colors, shapes, and fonts to create a cohesive look. For instance, using the same header style on all pages of a website.

3. **Alignment:** Alignment refers to arranging elements in a way that their edges line up along common rows or columns.
Application: Align text, images, and other elements to create a clean, organized layout. This makes the design look more professional and readable.

4. **Proximity:** Proximity involves grouping related elements together to indicate their relationship.
Application: Place related items close to each other to create a clear visual hierarchy.

5. **Balance:** Balance refers to the distribution of visual weight in a design.
Application: Achieve balance by distributing elements evenly across the layout. This can be symmetrical (mirror image) or asymmetrical (balanced by contrast in size, color, etc.).

6. **Hierarchy:** Hierarchy is the arrangement of elements to signify their importance.
Application: Use size, color, and placement to highlight the most important elements. For example, larger, bolder headlines indicate main topics, while smaller text indicates secondary information.

7. **Whitespace:** Whitespace is the empty space around and between elements in a design.
Application: Use whitespace to prevent clutter and improve readability. It helps to focus attention on the content and creates a clean, open design.

8. **Unity and Harmony:** Unity and harmony refer to the coherence of a design, where all elements work together to create a whole.
Application: Ensure that all elements of the design complement each other and fit within a cohesive style. This can be achieved through the consistent use of colors, fonts, and styles.

9. **Emphasis:** Emphasis involves making certain elements stand out to draw attention.
Application: Use contrasting colors, sizes, or shapes to emphasize key elements like call-to-action buttons or important information**.**

10. **Movement:** Movement refers to guiding the viewer's eye through the design in a deliberate way.
Application: Use lines, shapes, and positioning to lead the viewer's eye from one element to

another, creating a visual flow. For example, using arrows or leading lines to direct attention to a focal point.

11. **Simplicity:** Simplicity involves minimizing unnecessary elements to focus on what's essential.
Application: Remove clutter and keep the design straightforward and focused on the core message or functionality. This often involves using a minimalist approach.

12. **Functionality:** Functionality refers to the design's practical usability and effectiveness.
Application: Ensure that the design serves its intended purpose well, whether it's conveying information, enabling interaction, or facilitating navigation. Focus on user needs and usability.

13. **Consistency:** Consistency ensures that similar elements are presented in the same way throughout the design.
Application: Maintain uniformity in typography, color schemes, button styles, and other design elements across the project. This helps users predict how things work and reduces the learning curve.

14. **Accessibility:** Accessibility ensures that the design is usable by people with varying abilities.
Application: Use high contrast, readable fonts, and alternative text for images. Design for keyboard navigation and screen readers to ensure inclusivity.

Design System

1. Overview

The UPI app's design system is a thorough framework that guarantees a uniform and smooth user experience on all platforms. It outlines the visual, interaction, and branding components that shape the app's identity and user-friendliness.

2. Components

- **Color Palette:**
  - **Primary Color:** Teal (#008080) - Used for main buttons, headers, and active states.
  - **Secondary Color:** Light Teal (#66CCCC) - Used for success messages, links, and highlights.
  - **Neutral Colors:** Grayscale (#212121 to #E0E0E0) - Used for backgrounds, borders, and secondary text.
- **Typography:**
  - **Primary Font:** Roboto - Regular, Medium, Bold (16px, 14px, 12px)
  - **Secondary Font:** Montserrat - Used for headings and special sections (24px, 18px)
- **Icons and Imagery:**
  - **Icons:** Simple, line-based icons adhering to Material Design guidelines.

    - **Imagery:** High-resolution, relevant to the UPI ecosystem (e.g., payment, banking).
- **Layout and Spacing:**
    - **Grid-based layout with 8px spacing units.**
    - **Consistent margin and padding across all screens.**
- **Buttons and Controls:**
    - **Primary Buttons:** Full-width, rounded corners, with clear call-to-action labels.
    - **Secondary Buttons:** Outlined, used for secondary actions.
    - **Inputs:** Underlined text fields with clear labels and helper text.

Functional Requirements

These are the functionalities provided by the system in order to fulfil the user requirements.

1. User Registration:
A user shall be able to get registered with his mobile number and bank account.
Registration should be done through secure authentication mechanisms, like OTP verification.

2. Bank Account Linking:
The user will be able to link multiple bank accounts to the UPI app.
An app that can fetch bank account details securely from a mobile number.

3. Fund Transfer:
A user shall be able to send and receive money using UPI IDs, mobile numbers, and QR codes. The system shall support different kinds of transactions: peer-to-peer, peer-to-merchant, and merchant-to-merchant.

4. Balance Check:
The user shall be in a position to check the balance of their linked bank accounts.

5. UPI PIN Management:
Users should have the ability to set, reset, and change their UPI PIN securely.

6. Transaction History:
The ability to view the transaction history, including successful, pending, and failed transactions.

7. QR Code Generation/Scanning:
The application shall support the generation of Quick Response Codes for receiving payments. Supports scanning Quick Response Codes to make payments.

8. Notifications:
For transactions—successful payment, failed payment, or pending request—users should get real-time notifications.

9. Multi-Language Support:
It should support multiple languages so that all kinds of users can be catered to.

10. Recurring Payment Support:
In-system settings of recurring payments related to subscriptions, utility bills, and other service bills.

11. Customer Support:
Customer support is available in-app via chat or through helpdesk integration to help users with any issues or inquiries they might come up with.

12. Security and Fraud Detection:
Implementation of security features to detect and prevent fraud.

Non-Functional Requirements

They define how a system should operate and interact with users.

1. Scalability:
The platform needs to handle multiple users and their transactions simultaneously. It especially needs to do so during peak traffic.

2. Performance:
The transaction performance time of the application should be less; it should have low latency and ideally take only a few seconds.
Downtime should be low and availability high; 99.9% uptime should be targeted.

3. Security:
* Data in transit and at rest should be protected using end-to-end encryption.
* The system shall develop and maintain compliance with various security standards imposed on the industry for the processing of payment information, like PCI-DSS.

4. Reliability:
* The system shall be reliable and recover quickly in case of failure or outage.
* Data consistency shall be maintained throughout every transaction.

5. Usability:
User-friendly and intuitive for both tech-savvy and non-tech-savvy users
The most common features are directly on display or easily reached: send money or check balance.

6. Compatibility:
Support several operating systems and devices: iOS, Android, and web browsers.
Backward compatibility with the earlier versions of the OS

7. Localization:
The application should be able to support regional languages, currencies, and time formats for different regional users.

8. Compliance:
The platform shall adhere to local regulatory requirements, including data protection laws like GDPR in Europe and PDPB in India.

9. Maintainability:
The system should be designed for easy updates and maintenance; patches and upgrades shouldn't lead to a lot of downtime.

10. Auditing and Logging:
It should keep detailed logs of all the transactions and activities against users for purposes of audit and security.
The platform shall detect, monitor, and report all suspected activities.

11. Data Backup and Recovery:
Backups of all critical data shall be regularly done with fast recovery in case of data loss.

12. Energy Efficiency:
The app has to be optimized regarding its usage of mobile device batteries and shall not drain the batteries so quickly.