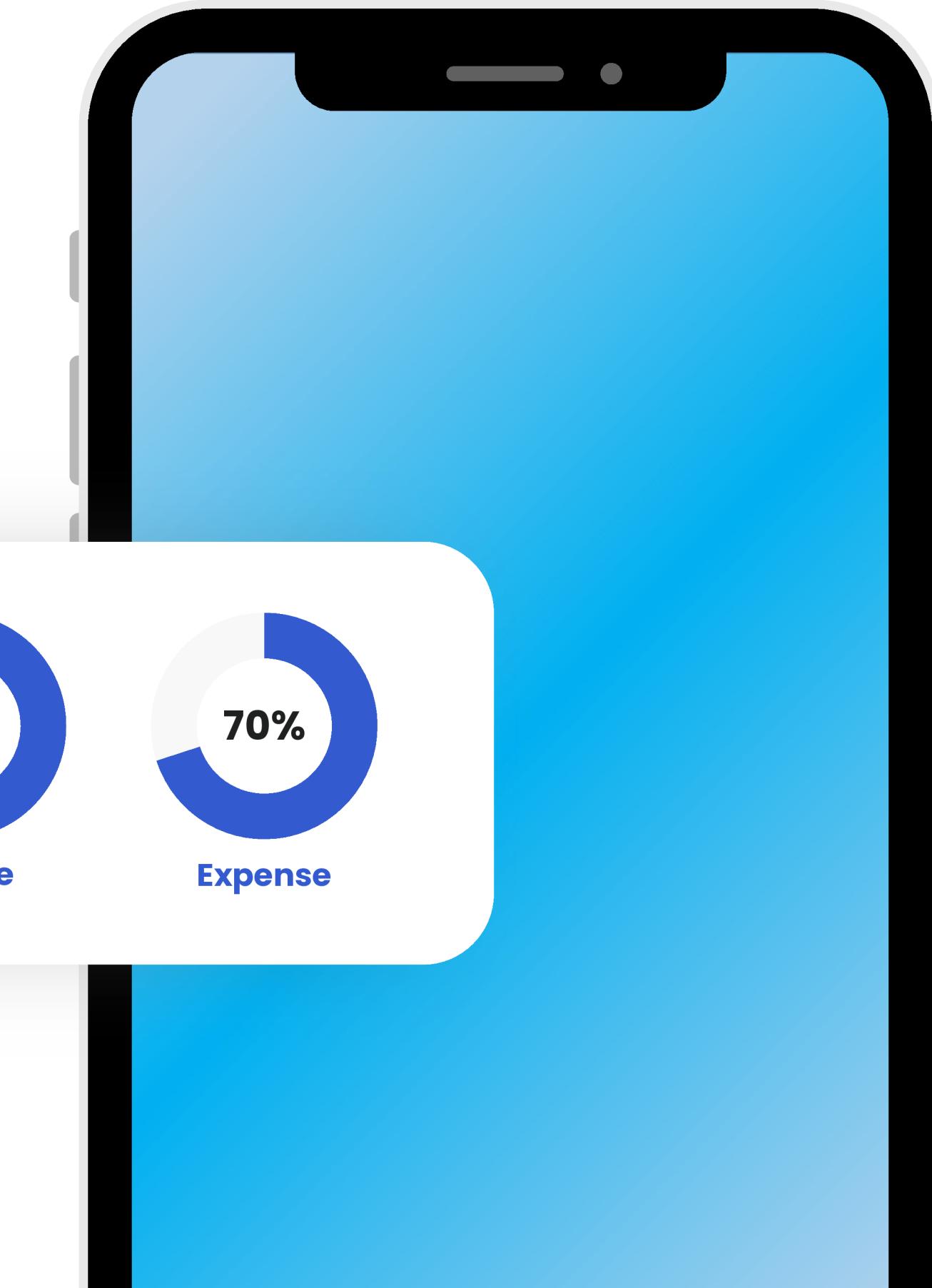
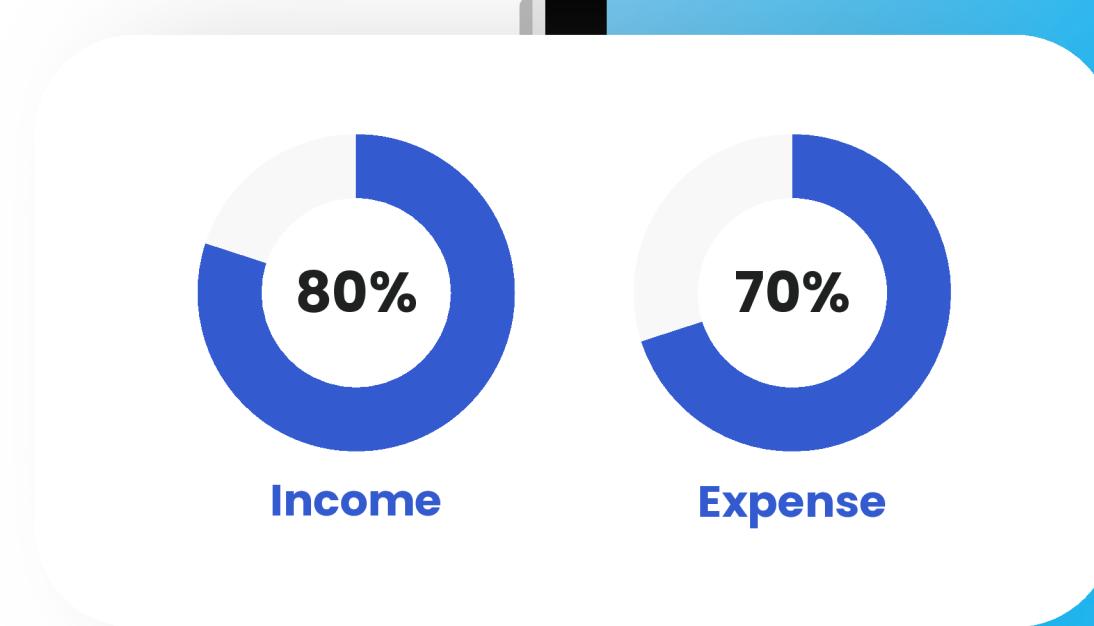
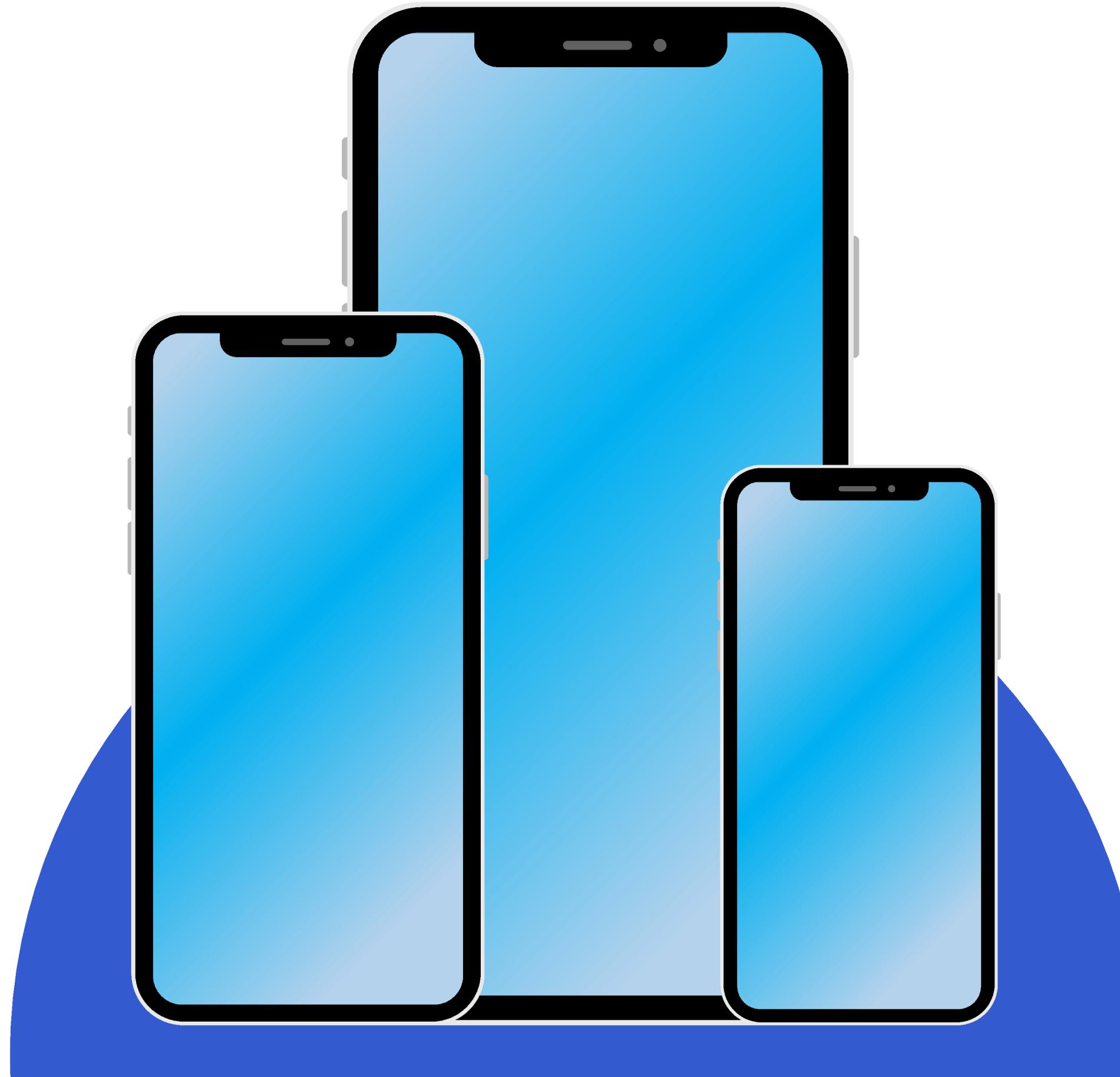


CampusExpense Manager

Mobile Application

Group 01





Stack Holder

1. University Students:

- Primary users of the app.
- Expectations: An intuitive and secure platform to manage finances, track expenses, and set budgets effortlessly.

2. BudgetWise Solutions Team:

- Responsible for app development and maintenance.
- Priorities: Ensure functionality, usability, and regulatory compliance.

3. University Administrators:

- Aim to promote financial responsibility among students.
- Expect the app to align with institutional values and mission.

Software Requirements

FUNCTIONAL REQUIREMENTS

- Secure user registration and login system.
- Expense tracking with customizable categories (e.g., rent, food).
- Budget-setting features with notifications for spending limits.
- Data visualization using charts and graphs.
- Offline mode with cloud synchronization for seamless data access.



Software Requirements

NON-FUNCTIONAL REQUIREMENTS

- **User Interface:** Intuitive and accessible, complying with usability standards.
- **Performance:** Ensure expense lists load within 2 seconds.
- **Security:** Implement AES-256 encryption and daily data backups.
- **Scalability:** Support up to 1 million users without performance degradation.
- **Reliability:** Maintain 99.9% uptime for user satisfaction.

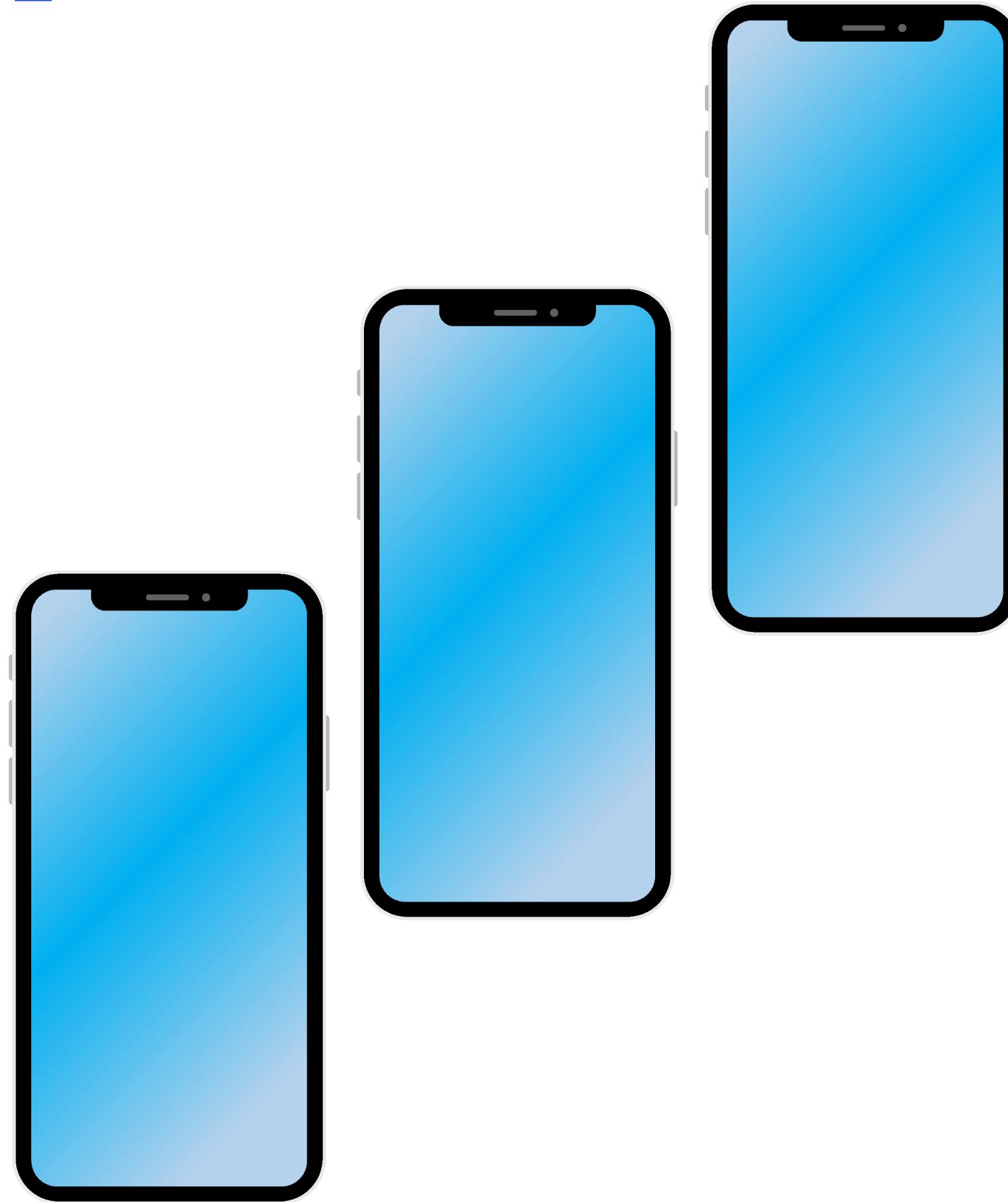




Anticipated Challenges

Challenges:

- 1. Inexperienced Team Members:**
 - Limited experience in mobile app development could cause delays.
- 2. Budget Constraints:**
 - Tight budget may restrict resources for advanced features.
- 3. Team Collaboration:**
 - New team members will require time to establish effective workflows.
- 4. Data Security Risks:**
 - Ensuring robust protection for sensitive financial data is critical.



Development Platform and Critical Features

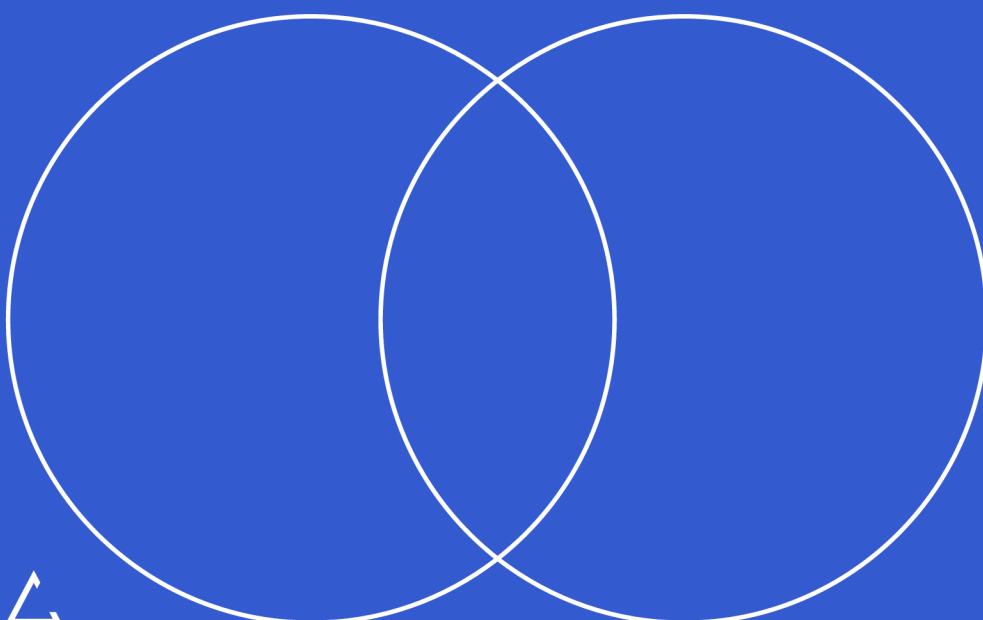
PLATFORM CHOICE

- **Android Studio:**
 - Comprehensive tools for development, including emulators and performance profilers.
 - Compatibility with Firebase for authentication and backend integration.
- **Draw.io:**
 - Tool for creating system diagrams to ensure a clear architectural design

CRITICAL FEATURES

- Secure user authentication.
- Categorized expense tracking and budgeting.
- Customizable notifications and alerts.
- Data analytics with visual graphs for insights.
- Offline functionality with automatic data synchronization.

Risk Analysis and Management Strategies



Identified Risks:

1. **Data Breaches:**
 - Risk of unauthorized access to sensitive financial information.
2. **Delays:**
 - Potential delays due to team's unfamiliarity with tools like Firebase.
3. **Integration Issues:**
 - Challenges in integrating frontend and backend systems.
4. **Performance Bottlenecks:**
 - Problems with handling large data volumes efficiently.

Mitigation Strategies:

- Implement AES-256 encryption and SSL for secure data transmission.
- Regular code reviews and automated testing to ensure quality.
- Conduct training sessions to familiarize the team with new tools.
- Perform stress testing to identify and resolve performance issues.
- Create contingency plans to reallocate resources for addressing delays.

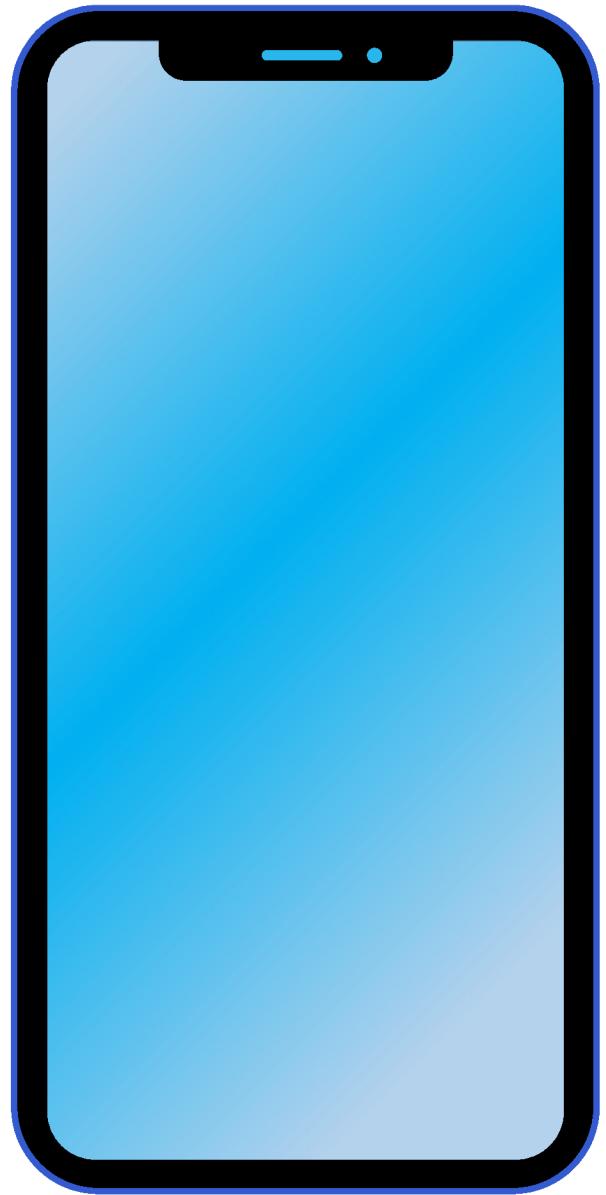
Project Timeline

Phases:

- 1. Planning: Weeks 1-2**
- 2. Risk Assessment: Week 3**
- 3. Development: Weeks 4-10**
- 4. Testing: Weeks 11-12**

Milestones:

- Initial Prototype: Completed by Week 5**
- Final Product Delivery: Completed by Week 12**



Team Responsibilities Team Members and Roles:

Team Members and Roles:

1. Nguyen Hoang Dung:
 - Role: Project Leader
 - Oversees project planning, execution, and delivery.
2. Tran Nhu Minh:
 - Role: IT Support and Security Specialist
 - Ensures secure data handling and system support.
3. Nguyen Thi Tuyet Nhung:
 - Role: UI/UX Designer and Researcher
 - Designs an intuitive and user-friendly interface.
4. Ta Van Toan:
 - Role: Quality Assurance and Testing Lead
 - Conducts rigorous testing to ensure app reliability.
5. Dong Manh Cuong:
 - Role: Software Developer and Documentation Manager
 - Develops core functionality and maintains project documentation.

Tools and Technologies

Development Tools:

- Android Studio: For app development.
- Microsoft Project: For project management.
- Draw.io: For system architecture diagrams.

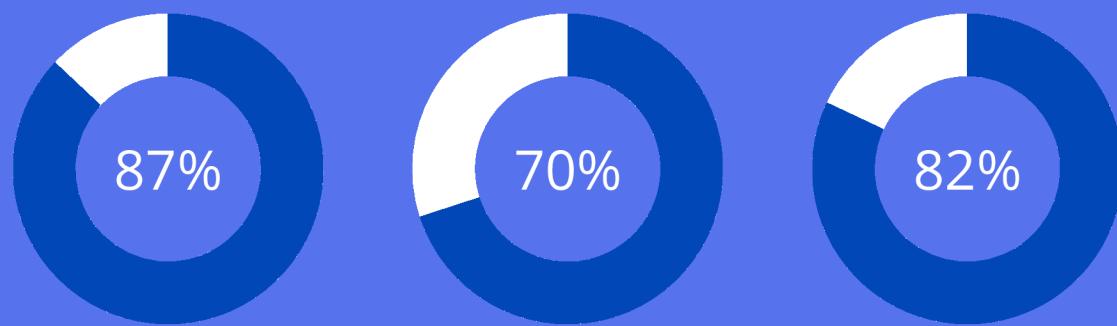
Key Technologies:

- SQLite: For data storage.
- Firebase: For authentication and cloud integration.
- Java/Kotlin: Programming languages for app development.

Design Tools:

Figma: For creating wireframes and prototypes.

Budget Breakdown



Categories:

- 1. Development Tools and Licenses:**
 - Includes software and platform licenses.
- 2. Team Salaries and Training:**
 - Allocated for developer compensation and skill development.
- 3. Testing and Quality Assurance:**
 - Covers tools and resources for thorough testing.
- 4. Contingency Fund:**
 - Reserved for unexpected costs or delays.



Security Measures



Implemented Security:

1. Encryption:

- **AES-256 for secure data storage and transmission.**

2. Authentication:

- **Multi-factor authentication for enhanced login security.**

3. Backup:

- **Automated daily backups to prevent data loss.**



Feedback Collection:

Focus Groups:

Regularly engage users to gather insights on usability.

Usability Tests:

Assess app features and user experience.

Iteration Process:

Implement changes based on user feedback.

Continuously refine features to improve user satisfaction.



Key Risks and Mitigation Plans

Risks

Risks:

1. **Budget Overruns:**
 - Overspending due to unforeseen challenges.
2. **Feature Delays:**
 - Delays in implementing advanced features.

Mitigation Strategies:

- Regular budget reviews to ensure cost control.
- Prioritize critical features and defer non-essential ones for later updates.



Thank You

