

# HOW THE NEEDS OF BUDGETWISE SOLUTIONS

**The main points in the presentation  
will include:**

1. STAKEHOLDERS.
2. FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS.
3. DIFFICULTIES ENCOUNTERED.
4. PLATFORM AND FEATURES.
5. RISK ANALYSIS.



# StakeHolder Presentation in Project

## INVESTORS WANTS

- ENSURE PROFITS FROM THE PROJECT.
- REGULAR REPORTS ON PROGRESS AND COSTS.
- WANT TO SEE CLEAR ROI (RETURN ON INVESTMENT) INDICATORS.

## CUSTOMERS WANTS

- PRODUCTS THAT MEET THEIR NEEDS AND EXPECTATIONS.
- HIGH-QUALITY PRODUCTS WITH NO DEFECTS.
- QUICK AND EFFECTIVE CUSTOMER SUPPORT SERVICES.

## PROJECT MANAGERS WANTS

- CONTROL PROJECT PROGRESS TO ENSURE TIMELY COMPLETION.
- MANAGE THE BUDGET EFFECTIVELY, WITHOUT EXCEEDING PROJECTED COSTS.
- ORGANIZE REGULAR MEETINGS TO UPDATE PROJECT STATUS.



# StakeHolder Presentation in Project

## PROPOSED BUDGET (12000\$)

- ALLOCATE BUDGET FOR EACH PHASE OF THE PROJECT.
- ANTICIPATE ADDITIONAL EXPENSES (PERSONNEL, TECHNOLOGY, MARKETING).
  - PERSONNEL COSTS
  - TECHNOLOGY COSTS
  - OTHER COSTS
- PREPARE CONTINGENCY PLANS FOR FINANCIAL RISKS.
  - CONTINGENCY: ADD ABOUT 10-15% OF YOUR TOTAL BUDGET TO COVER UNFORESEEN EXPENSES.



# Stages of budget use

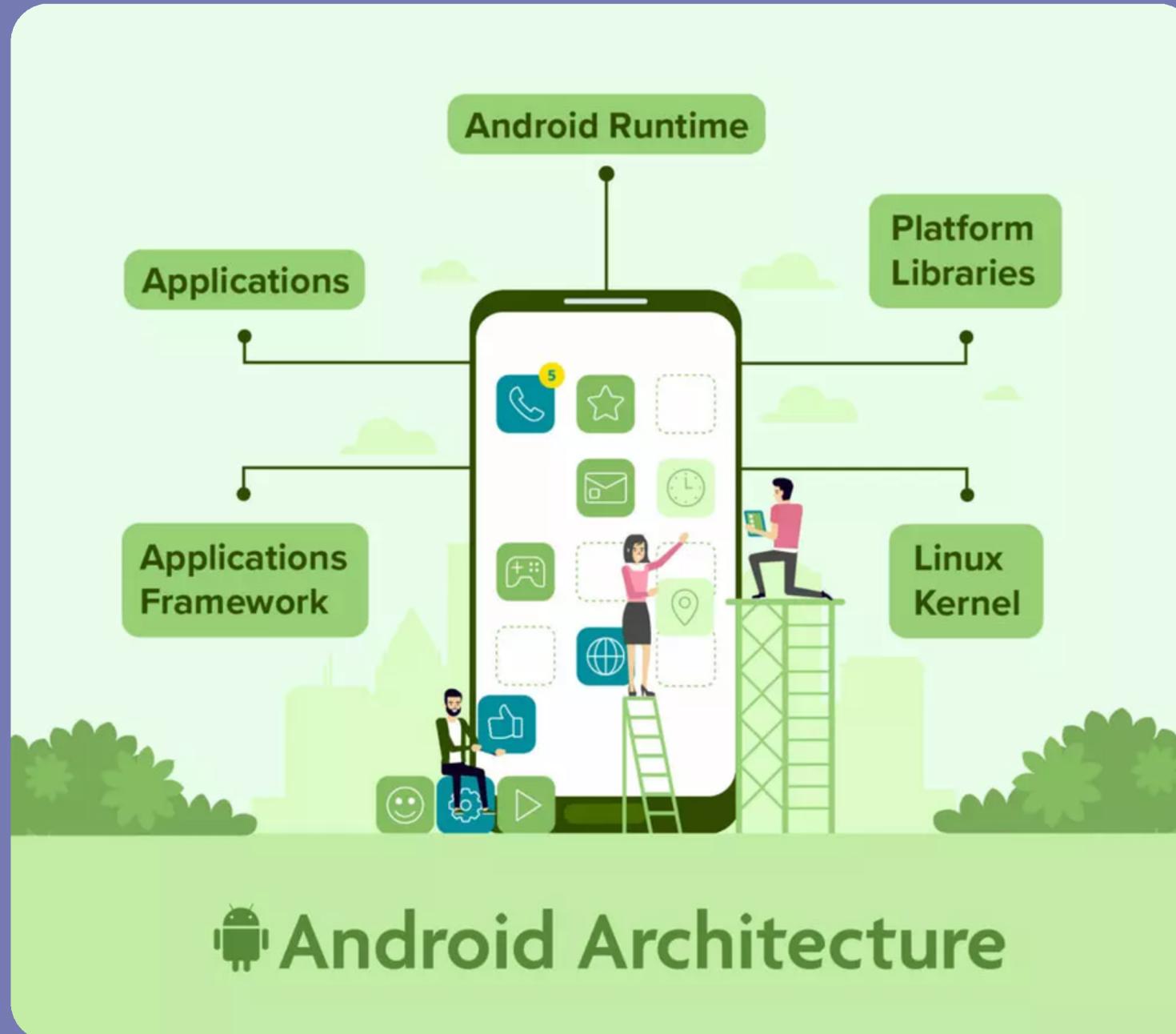


## Phase 1: Research and Requirement Analysis (10-15% of total budget)

- Requirement (500\$):
  - Identify user needs
  - Market analysis
  - Analyze competitors
  - Define the app's core functions for optimal user experience.
- Personnel Costs (700\$): Primarily for hiring business analysts and market researchers, often on short-term or hourly contracts.
- Technology Costs (300\$): If needed, for tools or software to conduct keyword and competitor analysis.
- Total Cost: Approximately 10-15% of the total budget.



# Stages of budget use



## Phase 2: Interface Design and System Architecture (15–20% of total budget)

- Objective: Design a user-friendly interface (UI/UX) and establish a solid technical architecture.
- Personnel Costs:
  - UI/UX Designer (700\$): For designing an intuitive user interface and experience, working closely with developers to ensure functionality and ease of use.
  - System Architect (1000\$): To structure data and software architecture for efficient, scalable, and secure app operations.

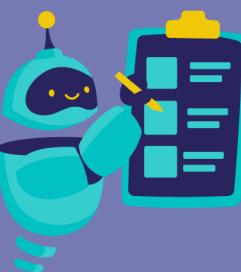
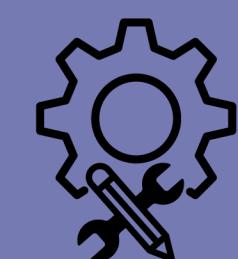


# Stages of budget use



## Phase 3: Development and Coding (35-40% of total budget)

- Objective: Build the app's codebase, including both front-end and back-end.
- Personnel Costs:
  - Back-End Developer (1500\$): Develop core functionalities and database, ensuring secure user data handling.
  - Front-End Developer (1000\$): Implement the UI and ensure smooth user experience, collaborating closely with the UI/UX designer.
- Technology Costs:
  - Servers and Hosting (500\$): For back-end infrastructure if the app requires its own server and database service.
  - API and Third-Party Services (500\$): Costs for integrating external APIs, such as payment or location services.
- Total Cost: Approximately 35-40% of the total budget.



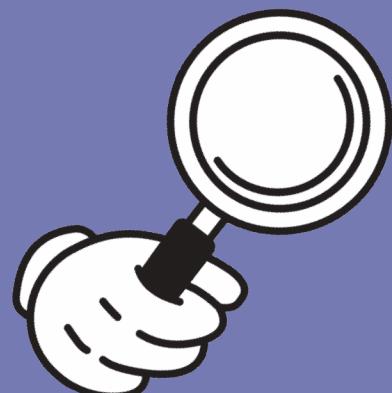


# Stages of budget use



## Phase 4: Testing and Deployment (15-20% of total budget)

- Objective: Ensure app stability, minimal bugs, and smooth user experience before the official release.
- Personnel Costs:
  - Quality Assurance (QA) Engineer (900\$): To conduct functional, performance, security, and usability testing, identifying and resolving any issues.
  - Tester (800\$): Will be responsible for thoroughly checking the application for errors before putting it into operation.
- Technology Costs:
  - Testing Tools (300\$): For automated testing using tools like Selenium or Appium to test on different devices.
- Total Cost: Approximately 15-20% of the total budget.



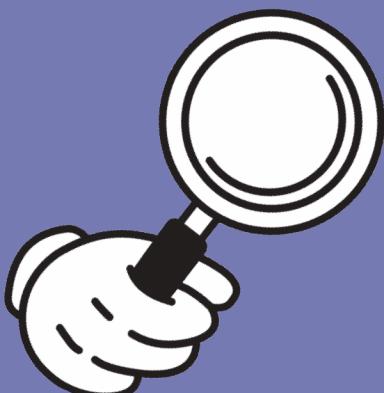


# Additional contingency costs for the project



## Phase 5: Additional project budget (15–20% of total budget)

- Cost of handling unexpected change requests.
- Covering costs arising from risks (such as software errors, time extension).
- Supporting maintenance or upgrade costs after the implementation phase.



# Functional Requirement and

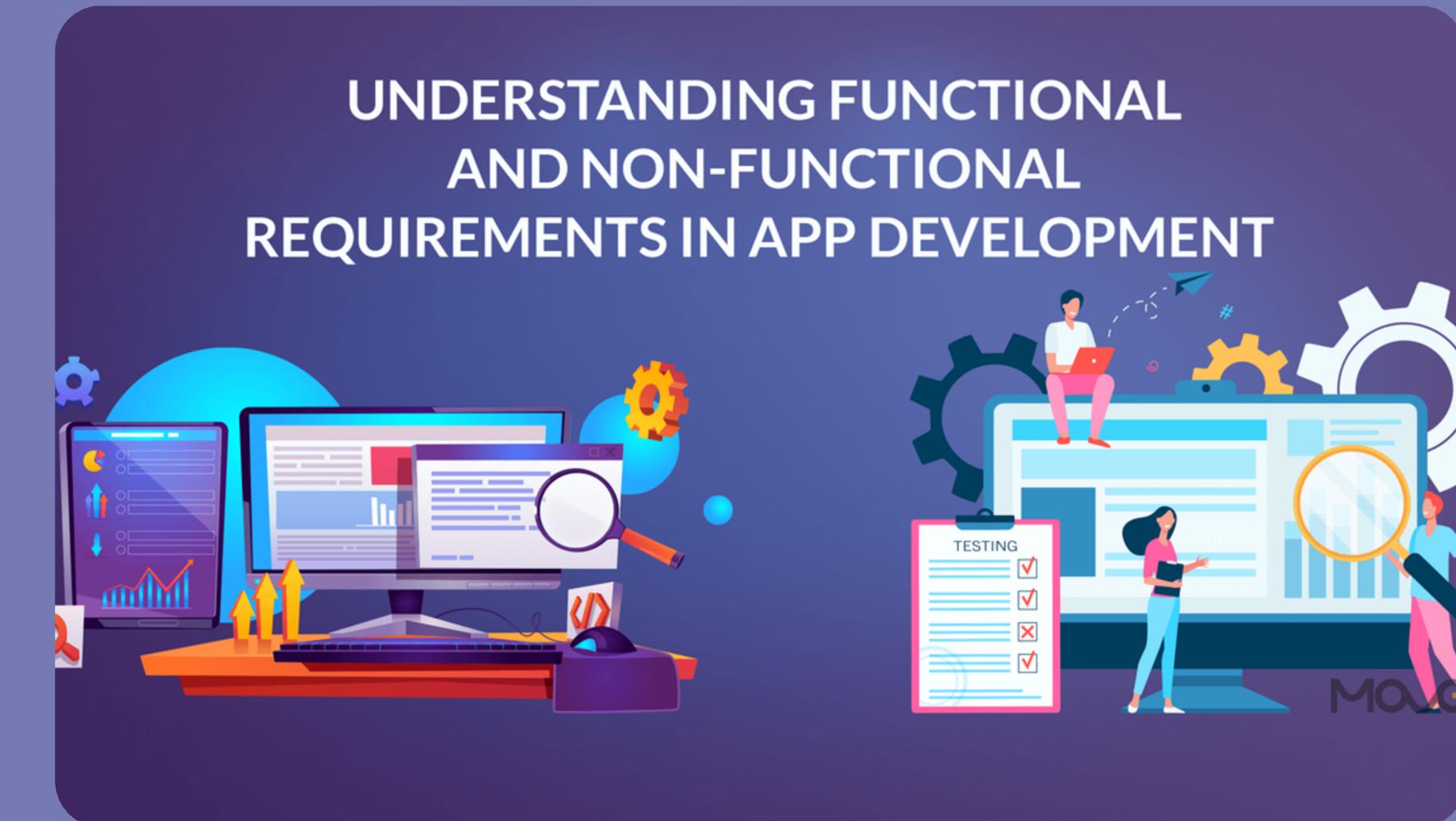
## Non Functional Requirements

### FUNCTIONAL REQUIREMENT

- **USER LOGIN/LOGOUT/FORGOTPASSWORD AND ACCOUNT MANAGEMENT.**
- **DATA MANAGEMENT (ADD, EDIT, DELETE).**
- **GENERATE REPORTS AND ANALYZE DATA.**
- **NOTIFICATION AND REMINDER FEATURES.**

### NON – FUNCTIONAL REQUIREMENT

- **RELIABILITY: THE SYSTEM MUST OPERATE CONTINUOUSLY WITHOUT INTERRUPTION.**
- **SCALABILITY: EASILY ADD NEW FEATURES IN THE FUTURE.**
- **PERFORMANCE: RESPONSE TIME UNDER 2 SECONDS FOR BASIC OPERATIONS.**
- **SECURITY: ENCRYPT SENSITIVE DATA AND IMPLEMENT STRONG USER AUTHENTICATION.**



# Functional Requirement



Login

Don't have an account? [Sign up](#)

Terms of Use | Privacy Policy



Register

Already have an account? [Sign in](#)



# Difficulties encountered

## DIFFICULTIES ENCOUNTERED IN THIS PROJECT



- INEXPERIENCED STAFF: BECAUSE THEY ARE STILL STUDENTS AND DO NOT HAVE MUCH EXPERIENCE, THEY ARE STILL BEHIND SCHEDULE AND MAKE SMALL MISTAKES.
- POOR BUDGET CALCULATION: STILL NOT OPTIMALLY CONTROLLING HUMAN AND FINANCIAL COSTS
- THE TEAM IS WORKING TOGETHER FOR THE FIRST TIME, SO THERE ARE STILL MANY THINGS THAT ARE NOT GOOD AT TEAMWORK



# Solutions to the Problem of Inexperienced Staff

Group Number

## Short-term training



Organize technical training sessions, provide self-study materials or instructional videos.

**Android Java Masterclass - Become an App Developer**  
Improve your career options by learning **Android** app Development. Master **Android S**  
build your first app today  
Tim Buchalka, Jean-Paul Roberts, Tim Buchalka's Learn Programming Academy  
**4.6 ★★★★★** (10,632)  
60.5 total hours • 324 lectures • All Levels

**The Complete Android 15 Course [Part 1]-Master Java & Kotlin**  
Learn **Android** App Development in both **Java** & **Kotlin** Languages. You'll master **Andro**  
ZERO to HERO  
Abbass Masri - Doc. Ali Alaeddine  
**4.4 ★★★★★** (5,672)  
76 total hours • 777 lectures • All Levels

**Jump right in** (1 lecture about "android studio java")  
**▶ Running Java Programs in Android Studio**

Provide self-study materials or video tutorials. (take courses on Udemy)

# Solution to these difficulties

Group Number

## Reasonable division of labor



Assign simpler tasks to new people, leaving the more experienced members to handle the complex parts.

## Build a supportive environment



Establish a mentor-mentee process, where experienced members guide newcomers.

# Solutions to the Problem of Poor Budget Calculation



- Use budget management tools: Use tools like Excel, Trello, or financial software to track and allocate your budget effectively.
- Better cost contingency: Allocate at least 10-15% of your budget for unexpected or risky expenses.
- Regular monitoring: Schedule weekly budget checks to make adjustments as soon as there are signs of overspending.



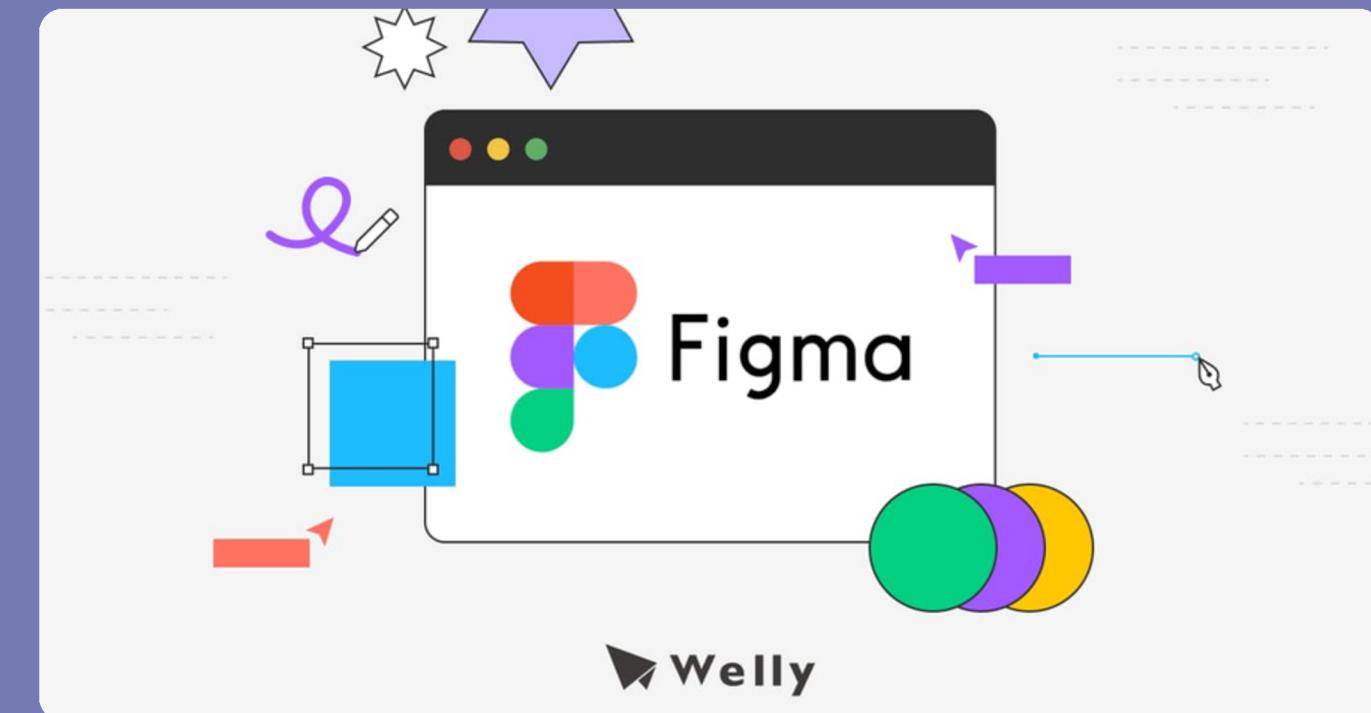
# Solutions to the Problem of Teamwork Challenges



- Establish clear processes: Divide tasks clearly and consistently, and always check and monitor progress reports.
- Communicate effectively: Hold short meetings every day or every week to discuss progress and solve problems.
- Team building activities: Organize off-project activities to increase cohesion among members.



# Technologies and tools used



# Required features

Group Number

- List of features:
  - Sign In/ Sign Up/ ForgotPass
  - Add 1 transaction (spending) - expense money
  - Add 1 source of money (Budget) - revenue remoney
  - Categorize spending categories
  - Set budget limits (For example, setting a monthly limit of 3 million will give a warning when spending exceeds the monthly limit)
  - Statistics of spending by time (day - to day) or by month, by week
  - Statistics of spending by category
  - Draw a chart showing comparison between time points by week, month

# Risk Analysis

## Group Number

### Human Risks

- Cause:
  - Lack of experience and knowledge.
  - Ineffective teamwork.
- Consequences:
  - Delays in expected progress.
  - Project quality is not really good.

### Financial Risks

- Cause:
  - Inaccurate budget estimates.
  - Unplanned costs.
  - Lack of capital when the project takes longer than expected.
- Consequences:
  - Not achieving goals due to cost cutting.
  - Project is behind schedule due to additional costs.

### Project Management Risks

- Causes:
  - Plans are unclear or unfeasible.
  - Progress is not closely monitored.
  - Lack of coordination among team members.
- Consequences:
  - Project is behind schedule.
  - Key objectives are not completed.

### Technical Risks

- Cause:
  - Unstable technology or tools used.
  - Not enough error checking in the product before putting it into operation.
- Consequences:
  - Product does not meet technical requirements or does not operate properly.
  - Increased cost of repairing technical errors.





## Human Risks

- Solution:
  - Training: Provide short-term training and coaching sessions to improve members' capacity.
  - Communicate regularly: Increase communication to solve project problems.

## Technical Risks

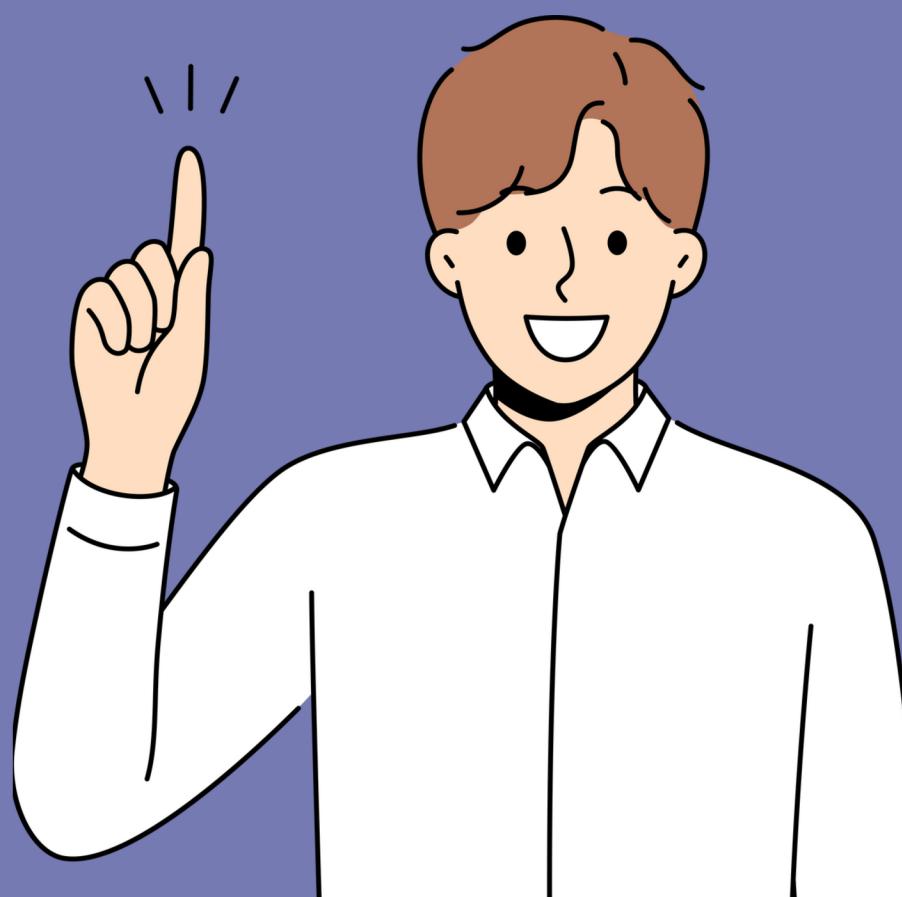
- Solution:
  - Choose the right technology: Research and evaluate technologies before applying.
  - Perform early testing: Carefully test the project when deployed to avoid errors.

## Financial Risks

- Solution:
  - Budgeting: Allocate the budget in stages and always set aside at least 10-15% as a contingency.
  - Monitor expenditures regularly: Prepare weekly or monthly financial reports.
  - Mobilize additional funding: Seek investment or funding sources to ensure the project is not interrupted.

## Project Management Risks

- Solution:
  - Make a detailed plan: Have a detailed plan and ask people with better experience to comment on whether the plan is good or not.
  - Check and report: Regularly check progress and make reports to monitor.
  - Regular meetings: Organize progress update meetings to promptly resolve arising issues.



Thank You  
for Listening