TIMEWISE

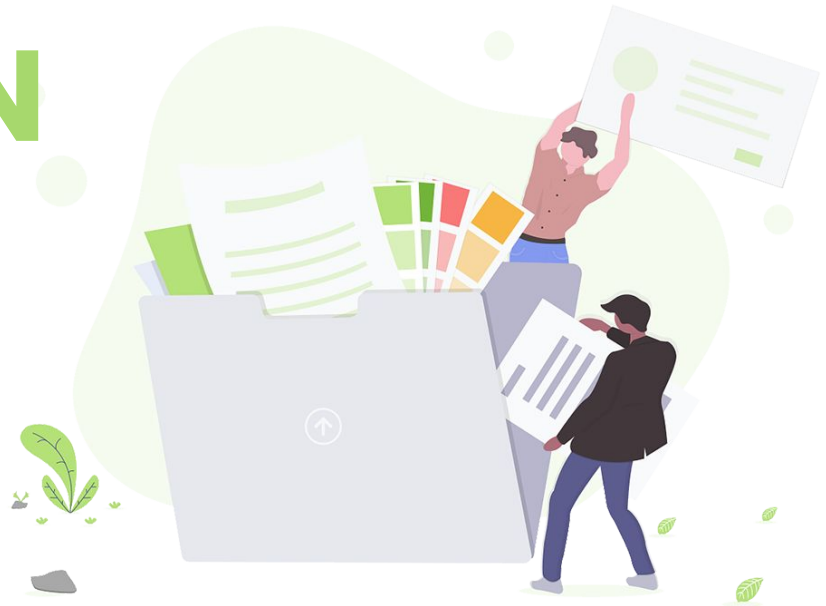
Mantri Raghav - PM (U1822309B)
Lek Jie Ling - QM (U1622502E)
Koh Hui Ling - QA (U1822228K)
Xue Xueting - QA (U1822318J)
Mittal Madhav - RM (U1822408H)

Dwivedee Lakshyajeet - DEV Lead (U1822289L)
Harding James - DEV BackEnd (N1902368K)
Alex Bernini - DEV FrontEnd (N1904075H)

CONTENTS

1. Product Introduction
2. Design for Maintainability
3. Software Quality Assurance
4. Project Management
5. Risk Management

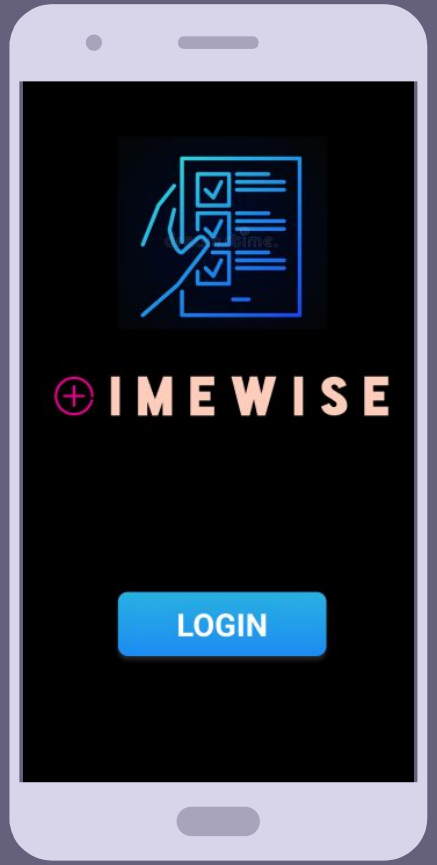
1.
PRODUCT INTRODUCTION

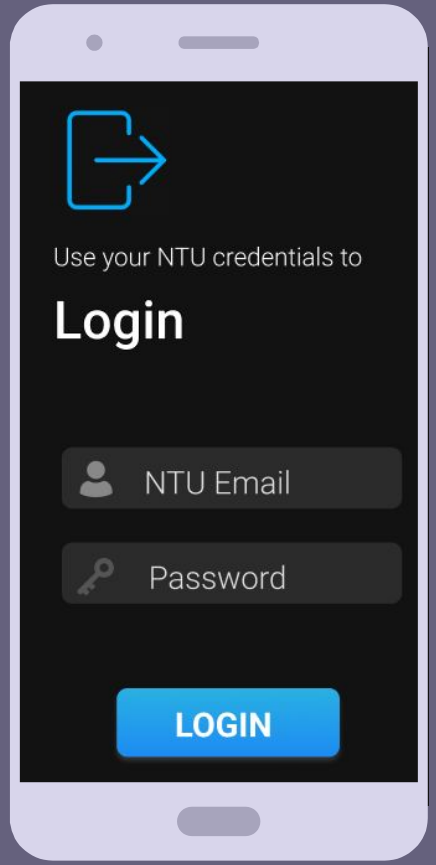


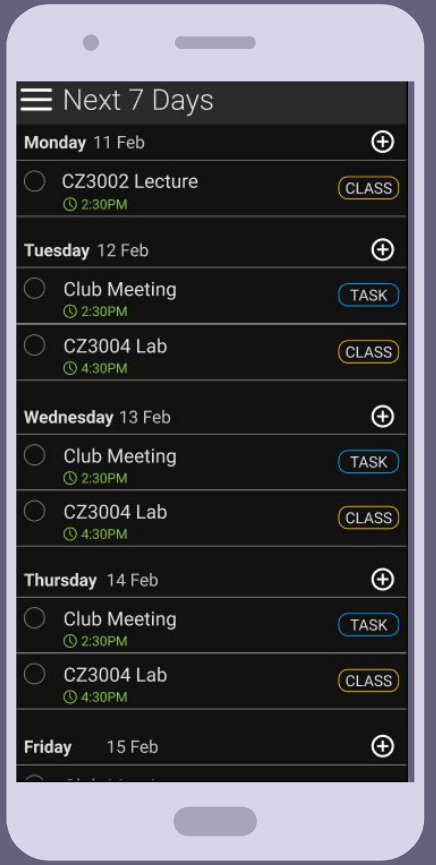
So, what is TIMEWISE?

- Task planning app
- Import time-table from STARS
- Add your own tasks too!

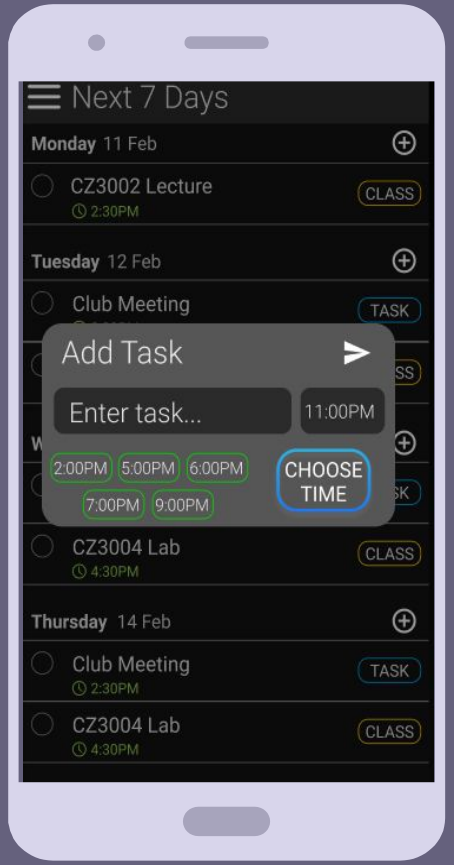
Introductory, Login and Home Page

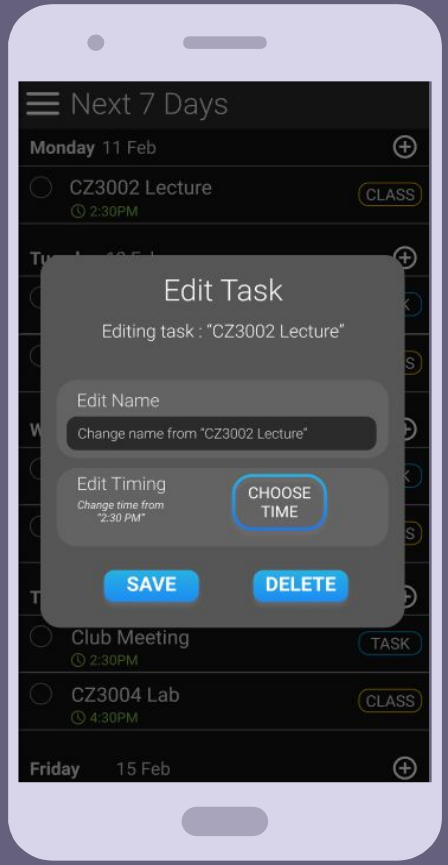






Add and Edit Tasks

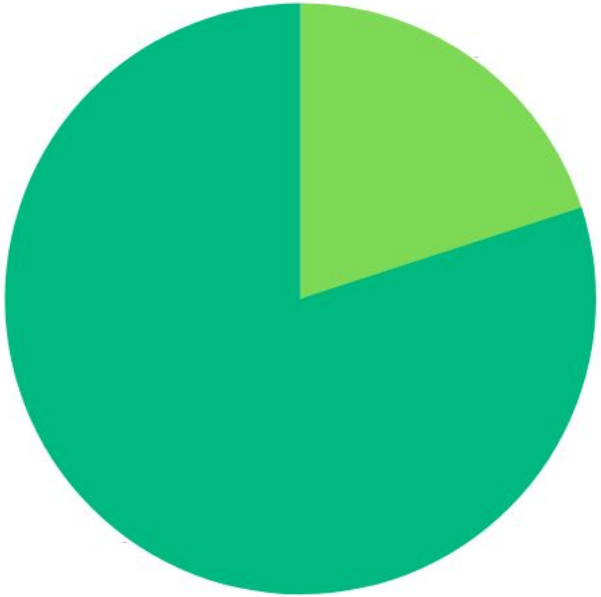




2.
DESIGN FOR MAINTAINABILITY

COST

- 80% - Maintenance
- 20% - Development of new software



Technical Consideration

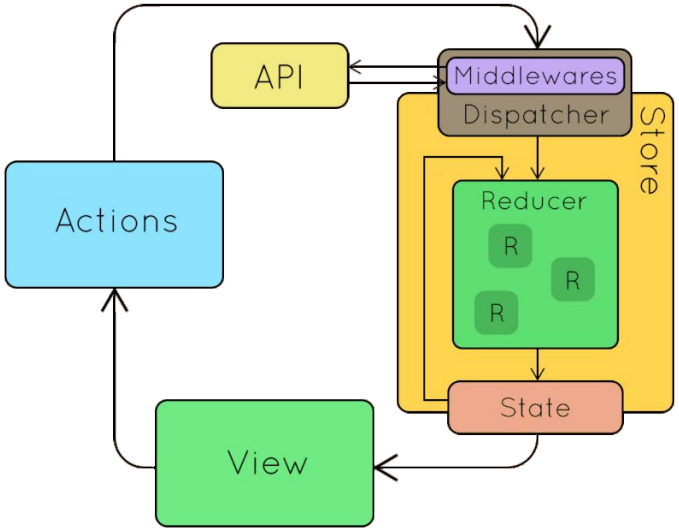
- FrontEnd : React Native
- BackEnd : Express
- Database - MongoDB
- Server - Heroku
- FrontEnd Testing - React Native Testing Library
- BackEnd Testing : Mocha/Chai

✔️ Cross-platform

✔️ Easy development (single language)

✔️ Cheap (pay per use)

Design Pattern



Redux

Central state

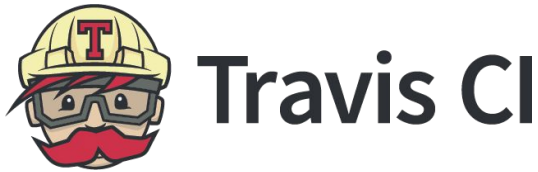
View dispatches Actions

Actions call Reducers

Reducers update State

- Separation of concerns
- Loose coupling
- Modularity

Software Management



Software Maintenance

Corrective Maintenance - Used NPM to update packages when library changed

Preventive Maintenance - React Native allows easy change of components

Adaptive Maintenance - Heroku for easy adaptation to changing users

Perfective Maintenance - Custom task tags in the future made easy by Redux

3.
SOFTWARE QUALITY ASSURANCE

Technical Review

Walkthrough

- Informal walkthrough will be conducted whenever a milestone is reached during development.
- The reviewer will also raise any issues found

Inspection

- A formal inspection will be done and the five inspection roles are: Reader, Author, Inspectors, Recorder and Moderator.
- With inspection, all issues raised will be documented and future project can learn from it.

Management Review

By having a management review, we can ensure that all the goals and constraints are met and also to improve points that are needed.

It also ensures our solution and application meets all requirements, even of the budget

Software Quality Planning

Functionality

Simplicity

Portability

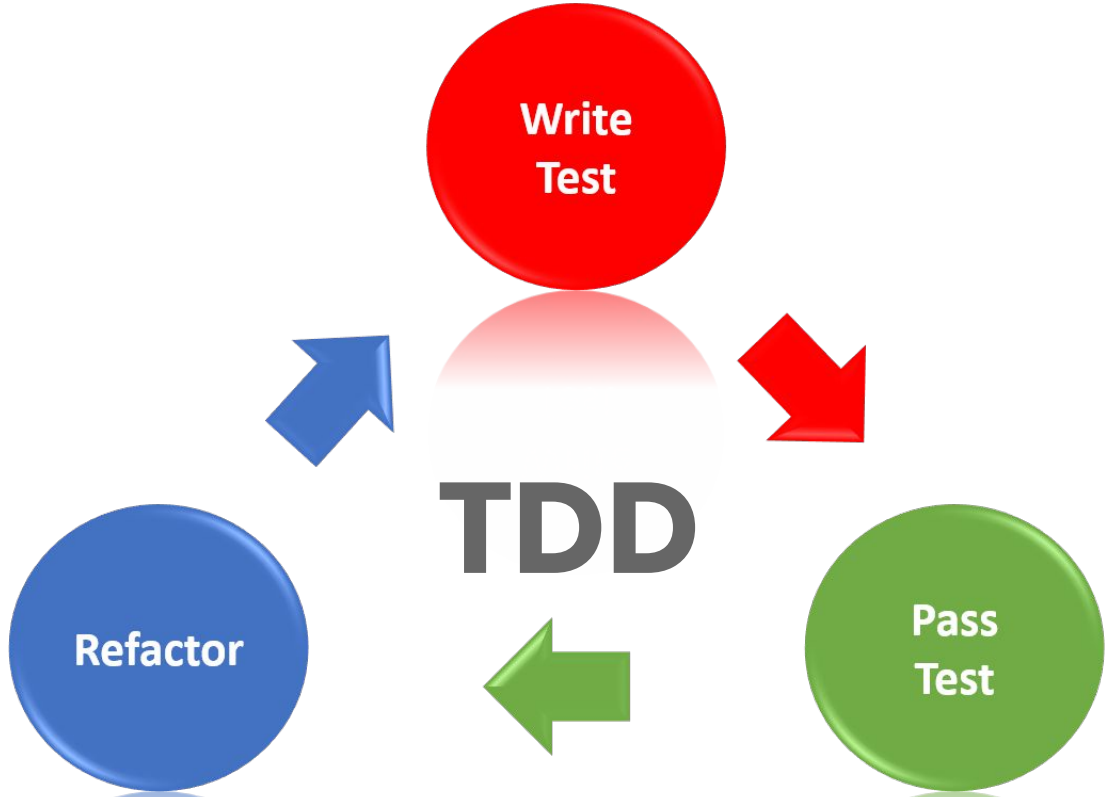
Reliability

Software Reviews

GitHub

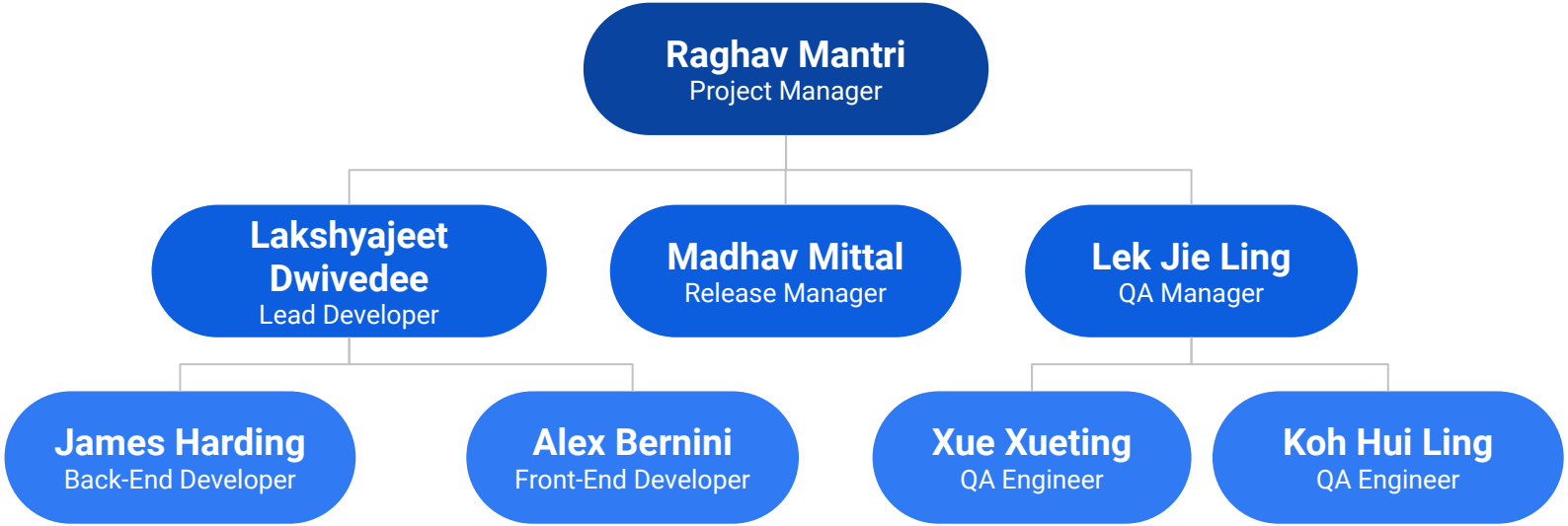
UAT Tests

Testing


Write
Test
TDD
Pass
Test
Refactor

4.
PROJECT MANAGEMENT

Project Organization



Project Estimation

Function Points

Element	Complexity	Detail
Inputs	Low	Login
	Low	University Timetable
	High	Task Creation
Logical Files	High	Tasks
	Medium	Student Account
Outputs	High	Display University Timetable
	High	Display User Tasks
Inquiries	High	Selecting the tasks
	Low	Selecting settings
Interfaces	Medium	Tasks, Schedule

Unadjusted Function Points

Characteristic	**Low**		**Medium**		**High**	
Inputs	2	× 3	0	× 4	1	× 6
Outputs	0	× 4	0	× 5	2	× 7
Inquiries	1	× 3	0	× 4	1	× 6
Logical Files	0	× 7	1	× 10	1	× 15
Interfaces	0	× 5	1	× 7	0	× 10
Unadjusted FP	9		17		41	
Total=L+M+H	67					

Adjusted Function Points

- Influence Factors - **14**
- Total Influence Score - **27**
- Influence Multiplier - **0.92**
- Total Adjusted Function Points - **61.64**
- Total Lines of Code (47LOC/FP) - **2897 LOC**

Distribution of Effort

1990's Industry Data	Work Package	Distribution	Estimates
Preliminary Design 18 %	Project Plan	9%	53.28
	Requirement Specification	9%	53.28
Detailed Design 25 %	User Interface	7%	41.44
	Technical Architecture	11%	65.12
	Data Modeling	7%	41.44
Code & Unit Testing 26 %	Code & Unit testing	21%	124.32
	Online Documentation	5%	29.6
Integration & Test 31 %	Integration & Quality Assurance	31%	183.52
	Extrapolated total effort		592
	2% for project management		11.84
	3% for contingency		17.76
	Total effort		621.6

Cost Estimates

Hardware:

8 Dell Computers	Total $8,000.00
Pentium IV 1.4GHz single processor	
256 MB RAM	
20.8GB IDE drive	

Software:

Server - Heroku	$2,800.00
Database - MongoDB	$15,000.00

Licensing:

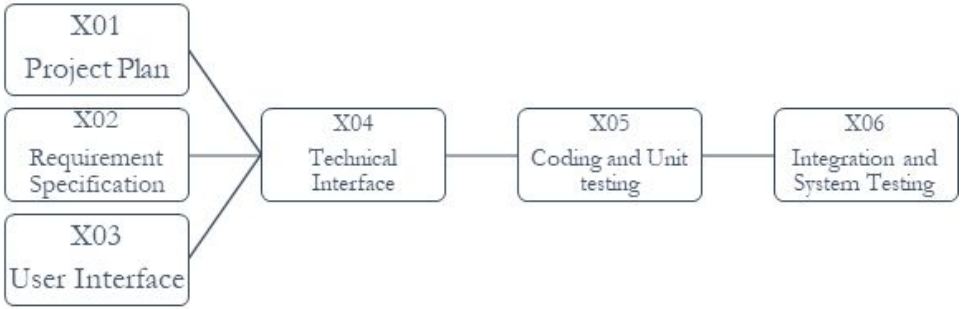
Google	$100.00
Apple	$100.00
NTU	$6,000.00

Staff:

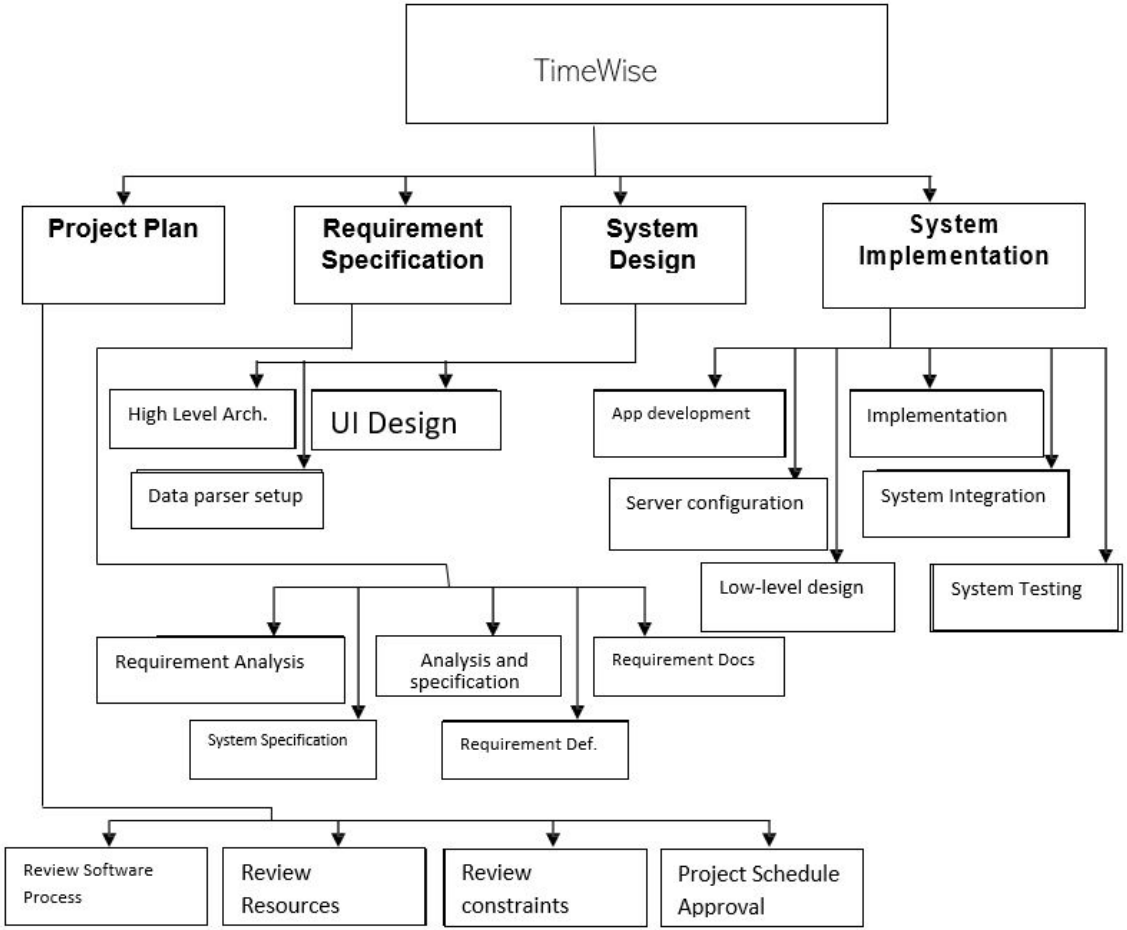
Project Manager	$30,000.00
7 Project Team Members	$21,000.00

Total - **$83,000**

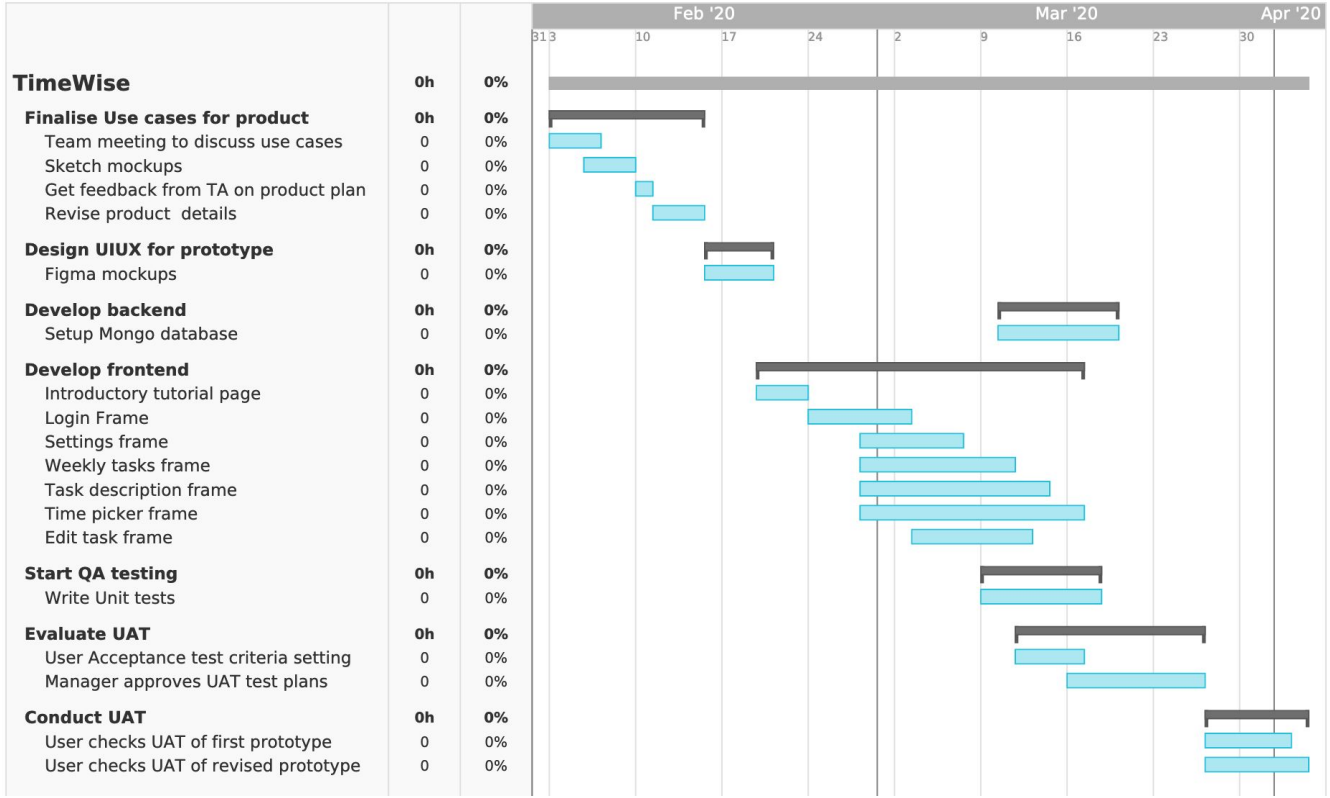
Activity Dependencies



Work Package #	Work Package Description	Duration	Dependencies
X01	Project Plan	7 days	--
X02	Requirement Specification	5 days	--
X03	User Interface	12 days	--
X04	Technical Architecture	5 days	X01,X02,X03
X05	Coding & Unit Testing	14 days	X04
X06	Integration and Quality assurance	10 days	X05


TimeWise
Project Plan
Requirement Specification
System Design
System Implementation
High Level Arch.
UI Design
Data parser setup
App development
Server configuration
Implementation
System Integration
Low-level design
System Testing
Requirement Analysis
Analysis and specification
Requirement Docs
System Specification
Requirement Def.
Review Software Process
Review Resources
Review constraints
Project Schedule Approval

Project Timeline

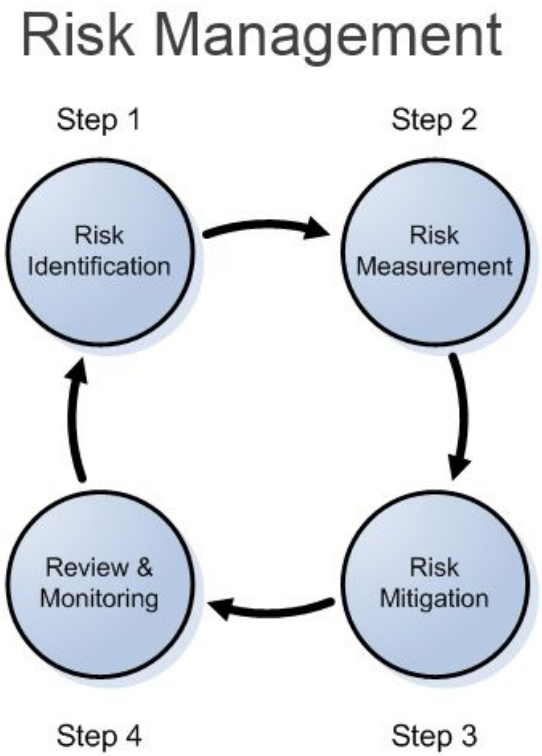


5.
RISK MANAGEMENT

OBJECTIVES OF RISK MANAGEMENT

- Plan ahead so that the team would be able to adapt to unplanned incidents, minimizing any negative impacts on the project itself.
- Ensure that the whole team is on the same page when having to deal with risks, avoiding any clashes.
- For recording purposes, to ensure that there exists a document that keeps tracks and logs the team's discussions with respect to the project risk.

Approach

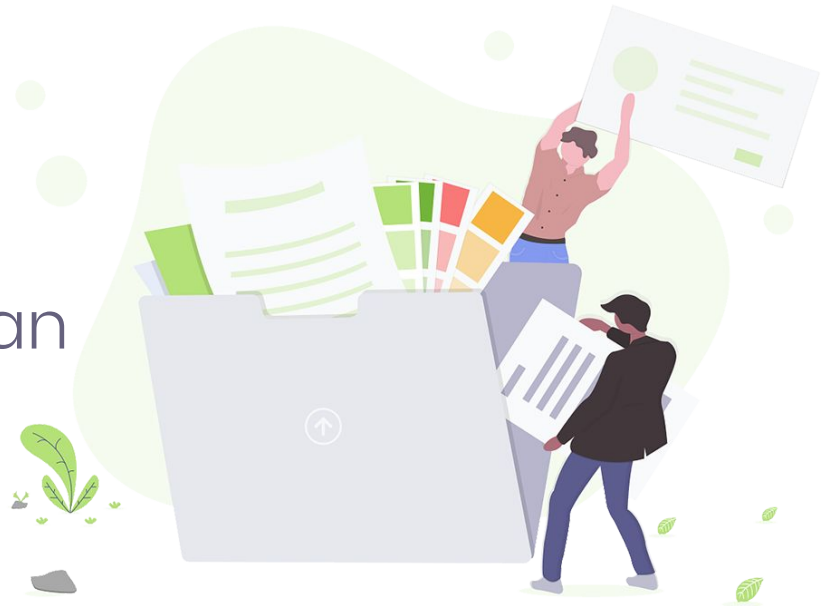

Risk Management
Step 1
Step 2
Risk Identification
Risk Measurement
Review & Monitoring
Risk Mitigation
Step 4
Step 3

Risk Identification

Project Team + Stakeholders

Evaluate:

- ➔ Environmental Factors
- ➔ Organizational Culture
- ➔ Project Management Plan



Risk Analysis

Identified risks need to be assessed and prioritized:

Qualitative

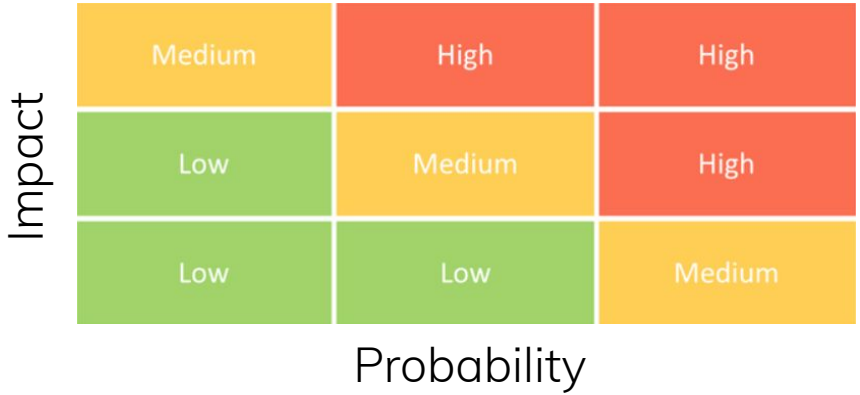
Quantitative

Qualitative Risk Analysis

Defined using

- Probability of occurrence of risk
  - High : P > 70%
  - Medium : P = 30 - 70%
  - Low : P < 30%

- Impact of risk on project
  - High : greatly impacts project
  - Medium : slightly impacts project
  - Low : relatively little impact on project



Impact	High	Data theft (2) Quarantined team members (7)	Scraper failure (1) Requirement changes (4) Sick Developer (6)	Server crash (3)
	Medium			
	Low	Overworked developers (5)	Underestimated deadlines (8)	
		Low	Medium	High
	Probability			

Risk that falls within the RED and YELLOW zones will have risk response planning, which may include both a risk mitigation and a risk contingency plan.

Quantitative Risk Analysis

Risks found are now prioritized by estimating their effect on project activities , and a numerical rating (ranging from 1-3) is applied.

Risk	Severity	Likelihood	Level of control	Significance
People	2	2	3	7
Technology	3	1	1	5
Requirement changes	3	3	1	7

- **Severity** : effect that will cause this project
- **Likelihood** : probability that the event will happen.
- **Level of control** : control we have for this particular risk type.
- **Significance** : sum of Severity, Likelihood and Level of control. Marked out of 10.

Risk Planning

Avoid

Mitigate

Accept

Transfer

Risk Monitoring

Constantly identify, analyze, and plan for new risks:

- Weekly scrum meetings
- Regular test runs

Monitoring previously identified risks:

- Re-prioritize risks depending on probability and complexity

RISK LOG

RISK #1 – PEOPLE

- **Risk type** : People
- **Possible Risks** :
  - Overworked Developers
  - Sick Developers
- **Probability (1-3)** : 2
- **Severity (1-3)** : 2
- **Effects** : Additional backlog due to lack of members / fatigue
- **Approach** : Mitigate
- **Plan** :
  - Find temporary replacements
  - Send regular updates to developers to keep them updated

RISK #2 – TECHNOLOGY

- **Risk type** : Technology
- **Possible Risks** :
  - Changes in NTU website leading to drastic changes in our backend
  - Data Theft
  - Server Crashes and Loss of data
- **Probability (1-3)** : 1
- **Severity (1-3)** : 3
- **Effects** :
  - An obsolete backend
  - Loss of client trust
  - Loss of client data
- **Approach** : Mitigate
- **Plan** :
  - Spend additional resources to update backend
  - Regain client trust with the help of professionals
  - Invest in a higher security database to prevent further data breaches

RISK #3 – REQUIREMENT CHANGES

- **Risk type** : Requirement Changes
- **Possible Risks** :
  - Complaints from users
  - New competitors with better UI/ functionalities
- **Probability (1-3)** : 3
- **Severity (1-3)** : 3
- **Effects** : Having to spend extra resources for perfective maintenance
- **Approach** : Accept
- **Plan** : Listening to user complaints, and adding new features to keep up with the competitors

Live Demo

Thank you!

Any questions?