



Software Development Project Management
Department of Computer Science
American International University-Bangladesh (AIUB)

Project topic: Developing a Software Development Project Management Plan for
Dhaka Subway Systems Automated Ticket Issuing System

Subject: Software Development Project Management

Section: B

PREPARED BY

No	Name	ID
1.	Rafa, Humayara Chowdhury	17-35413-3

GUIDED BY

S.M. Abdur Bhuiyan Rouf

Department of Computer Science

American International University-Bangladesh

Table of Contents

1. Title page.....	1
2. Revision history page	3
3. Introduction	3
4. Process Model	
a. Software model	3
b. Reason for chose this model... ..	3
c. Software life cycle flow chart	4
5. Quality gate for each phase of software development.....	4
6. List of tasks.....	5
7. Estimation for each task	
a. Use estimation model COCOMO'81	5
8. Schedule the tasks	7
9. Prepare list of milestones	8
10. Staffing Plan	
a. Assign Engineers Requirements	9
b. Backup Assign Engineers Requirements	10
11. Monitoring and Controlling Mechanism	
a. Schedule Tracking Process.....	10
b. Team leader meetings.....	11
c. Project of Contact and Monitoring.....	11
12. Risk Management	11
13. List of Deliverables	
a. Deliverable work Products	13
14. Defect Tracking Process	14
15. Metrics	15
16. Postmortem	
Plan a postmortem	16

Revision History Page:

Date	Author	Revision	Description
September 10,2020	Hasan, S M Rafa, Humayara Ismail, Isrita Hossain, Md. Mridul Karmokar, Mrinmoy	Version 0.0.1	First Version

Introduction:

This document discusses the criteria for a Dhaka Subway Systems Automated Ticket Issuing system. The target audience for this document are project managers. It is the document guiding this project. It describes the application technologies and administrative methods used to produce the product. As such, the necessary documents are in the Requirements Analysis Report. Changes to one of these may represent changes to the other text. It includes all the technical and administrative tasks required for turning the results. This includes scheduling, identification of tasks, and factors that may impact the project and planning.

Process Model:

Software Model

Agile development iterative approach that incrementally builds software, rather than delivering the full product near the end of the timeline. This method is streamlined and versatile, enabling to make changes whenever necessary.

Reason for chose this model

- Speeds up the application deployment
- Reduced wastage by resource minimization
- Increased versatility and changeable adaptability
- Expanded success from more concentrated activities
- Turnaround times faster
- Recognition of bugs and weaknesses more easily
- Optimized systems for production
- Optimum management of project
- Focusing constantly on individual consumer demands

Software life cycle flow chart:



Figure: 1.0

Quality Gates for Each Phase of SW Development:

Improved productivity is a prime objective of the agile approach, but the term has a distinct meaning from that commonly associated with custom application development. Because it's even more intuitively, the degree to which an application meets specifications and the prevention of defects after delivery of the application are also common in manufacturing. According to this, quality is defined as both the degree to which a delivered application satisfies user needs, and the degree to which a delivered system has low maintenance costs. It tries to deliver quality through the strong involvement of users in the analysis and particularly in the design phases.

Metric Performance	Trigger
Open defects vs. closed defects over time	Open defect rate increase $> 0.5 \times$ rate of increase in closed faults in the past
Line Of Code (LOC)	100,000
Source code comment percentage	$< 25\%$
Defects per KLOC	<p>$> 11\%$ Defined Standard for Implementation stage, followed by:</p> <ul style="list-style-type: none"> • Coding: 13 • Complication: 7 • Black-box test: 6 • Integration: 4

List of Tasks

- Requirements Elicitation
- Project Planning
- Requirements Analysis
- System Design
- Object Design
- Implementation & Unit Testing
- System Integration & System Testing

Estimation for each task:

Usually the number of KLOC to complete a Software System is 100.
So, person-months need for the project (Labor working hour)

$$\begin{aligned}\text{Effort} = \text{PM} &= \text{Coefficient}_{\langle \text{Effort Factor} \rangle} * (\text{SLOC}/1000)^P \\ &= 3 * (100,000/1000)^{1.12} \\ &= 521 \text{ hours}\end{aligned}$$

Here,

$\text{Coefficient}_{\langle \text{Effort Factor} \rangle} = 3$ (as the project is a Semi-detached project)

SLOC (Source Line of Codes) = 100,000

$P = 1.12$ (as the project is a Semi-detached project)

As we have selected agile model to complete the project and by estimating effort we find that we will need 65 working days to complete the project. As the model allows 60-90 (around) working days to complete a project our estimated time is quite viable and by basis of RUP (Rational Unified Process) we are dividing the time phase for each task as it shows in below:

Task of phase	Days	Hours
Requirements Elicitation	08	64
Project Planning	11	88
Requirements Analysis	08	64
System Design	12	96
Object Design	10	80
Implementation & Unit Test	08	64
System Integration & System Testing	09	54

Note: Each engineer works for 8 hours in a day and 5 days of a week. Total project duration is 66 working days (excluding national holidays).

And, Development time

$$\mathbf{DM = 2.50*(PM) ^T}$$

$$= 2.50*521^{0.35}$$

$$= 22 \text{ hours}$$

Here,

PM = Person-months need for the project (Labor working hour) = 521 hours

T = 0, 35 (as the project is a Semi-detached project)

And, Required Number of People

$$\mathbf{ST = PM/DM}$$

$$= 521/22$$

$$= 23 \text{ peoples}$$

Here,

PM = Person-months need for the project (Labor working hour) = 521 hours

DM = Development time = 22 hours

So, our total estimation for the project is

- We need 521hours to complete the project.
- For Development we need 22 hours.
- To complete the project, we need 23 engineers.

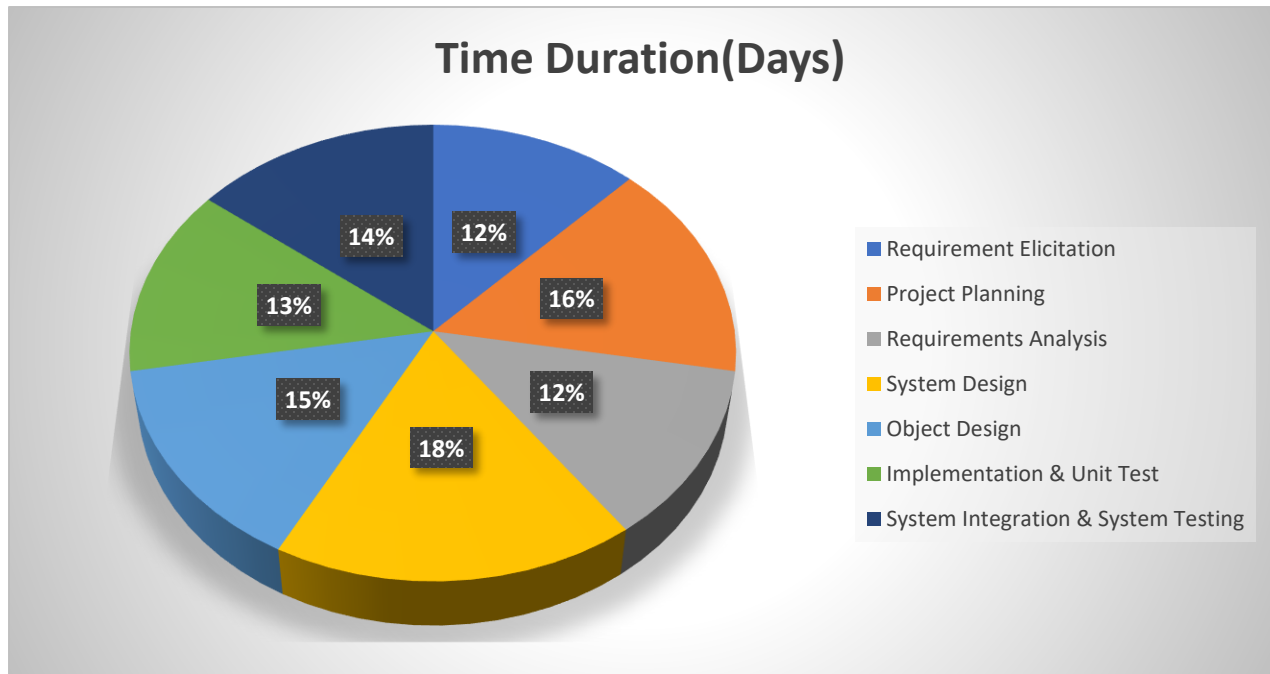
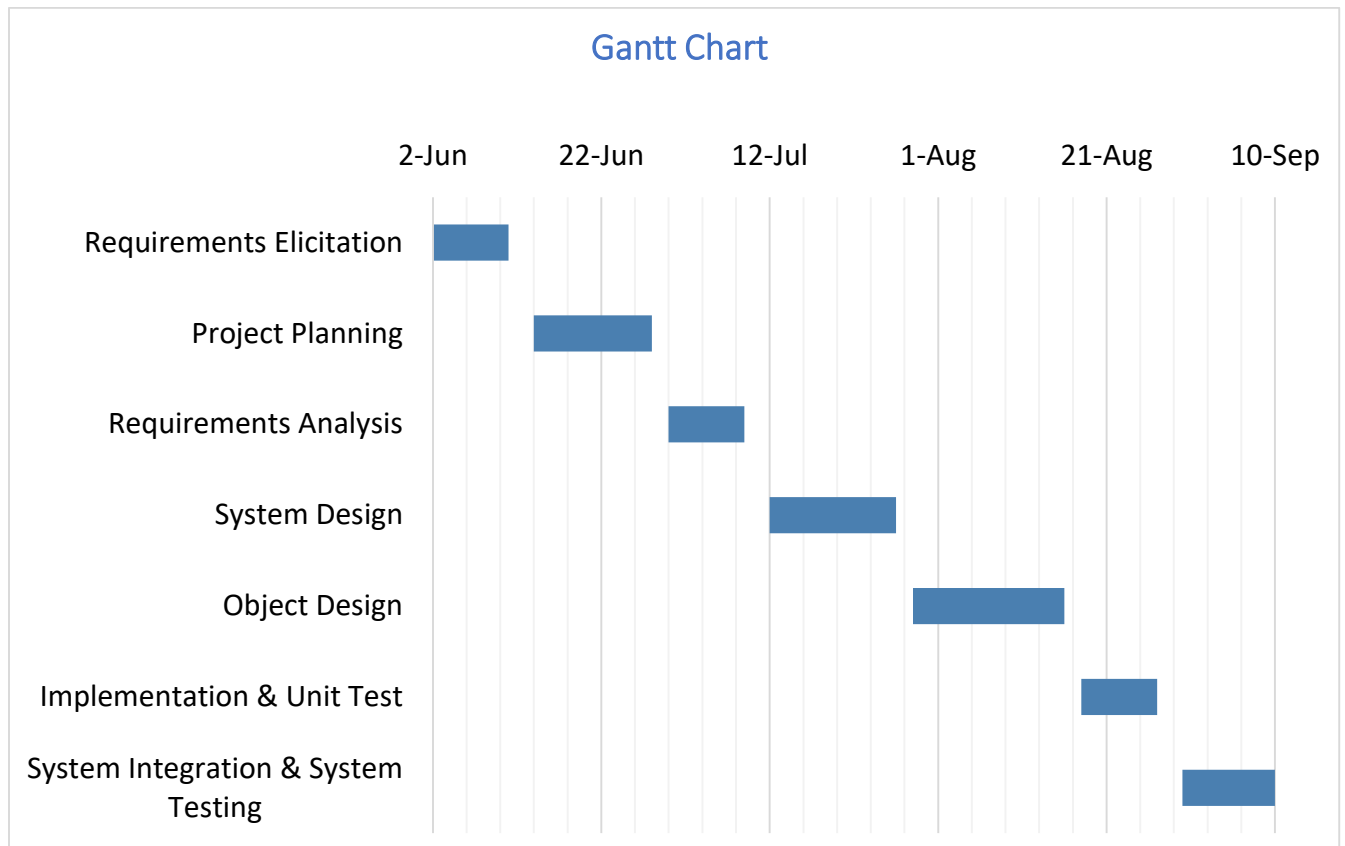


Figure: 1.1

Schedule the tasks:

Project Phase	Date
Requirements Elicitation	June 2 – Jun 11
Project Planning	June 14 – June 28
Requirements Analysis	June 30 – July 9
System Design	July 12 – July 27
Object Design	July 29 – August 16
Implementation & Unit Test	August 18 – August 27
System Integration & System Testing	August 30 – September 10

**Figure: 1.2****Prepare List of Milestone:**

Date	Project Milestones
June 12	Project Presentation
July 11- 13	Analysis Review
August 19	Project Review
September 1	Object Design Review
August 26	Project Agreement
September 2	Internal Project Review (functional prototype)
September 10	Project Acceptance

Staffing Plan:

The purpose staffing plan is to make certain the project has sufficient with the right skills and experience to ensure a successful project completion.

Assign Engineers Requirements:

The following is a detailed breakdown of the roles required to execute the project. It includes: the project developers name, their role, the project responsibility of the role, skills required, no of staff required fulfilling the role, the estimated start date and the expected duration the staff resource will be needed on the project.

Name	Role	Project Responsibility	Skills Required	No. of Staff Required	Estimated Start Date	Duration Required
1.Hasan 2.Rafa 3.Isrita	Java Developer	Java Development	Java Server pages (JSP) and servlets, Web Framework, Python.	3	June 2,2020	July 28, 2020
1.Mridul 2.Mrinmoy 3.A	Web Developer	Web Development	HTML/CSS, JavaScript, Testing and Debugging.	3	June 2,2020	July 28, 2020
1.A 2.B 3.C 4.D 5.E 6.F	Database Management	Full control in Database	Oracle10g, SQL, MySQL	6	June 2,2020	July 28, 2020
1.Mahmudul 2.Ismail 3.Humayara	Software Engineer	Software Testing and Quality Assurance	Test Planning and Documentation, Testing tools and Techniques, knowledge of SDLC.	3	June 2,2020	July 28, 2020
1. G 2. H	Project Manager	Project Leader	Look after all over project	2	June 2,2020	July 28, 2020

Backup Assign Engineers Requirements:

In the project that any of the allocated engineers will be unavailable for some reason these replacement engineers work for the project as the expected start date to the necessary period date.

Name	Role	Project Responsibility	No. of Backup Staff Required	Estimated Start Date	Duration Required
Aa	Java Developer	Developer	1	June 2,2020	July 28, 2020
Bb	Web Developer	Developer	1	June 2,2020	July 28, 2020
Cc Dd	Database Management	Full control in Database	2	June 2,2020	July 28, 2020
Ee	Software Engineer	Software Testing and Quality Assurance	1	June 2,2020	July 28, 2020
Ff	Project Manager	Project Leader	1	June 2,2020	July 28, 2020

Monitoring and Controlling Mechanism:

Establish reporting systems, data types, monitoring and assessment structures, and other instruments and methods for recording and controlling adherence to the SPMP. Project management will take place at work kit stage. Including procedures for monitoring and maintaining the support functions for the project (Quality Assurance, Configuration Control, Reporting and Training).

Schedule Tracking Process:

The work performed by the members has to be conducted every week. Each team member must fill out their hours in a web-based journal. Developers must fill in on this file. A week starts on Sunday, and ends on Tuesday. The PM sends an e-mail to the SM each week, which includes the hours spent on the various job package and the hours spent on the following categories: Linked project, Non-project connect, Documentation, Specification, Architecture, Source Code, Checking, Inspection, Restructuring, and Rework. Additionally, an average of available hours is applied for each work-package.

Team leader meetings:

He / She should schedule a team meeting anytime the Project Manager (PM) considers it necessary. Team leader meetings are very casual and rare, and the activity needed for one of the meeting purposes. Flow Matrix for reporting and information:

Action	Weekly	Monthly
Status meetings	Project Team Meeting, Project Manager & Department Director	Project Manager and Directors
Reports	Complete Tasks, Details on Schedules & Tasks for Next Week	Summary of Standard Reports

Project of Contact and Monitoring:

The project monitoring and contact plan can be illustrated with a chart. The correspondence table will show the daily notifications and contact scheduled for the project, such as weekly progress notes, routine feedback, or as-needed contact. The basic ways of communication vary from group to group, so identifying the expected means is helpful at project start.

Additional details Submitted	From	To	Period of Time
Report of Status	Project Team	Project Manager	Weekly
Report of Status	Project Manger	Project Team & Software Manager	Weekly
Project Review	Project Team	Software Manager	Monthly

Risk Management:

A number of potential project risks are listed in this section. Measures or acts are also described for hazard prevention or reduction.

Four risk categories are defined as follows:

- Risks about the job to be performed.
- Risks about the management.
- Risks related to the capital.
- Risks in relation to client.

Risks	Probability	Impact	Rating	RMMM
Project Manager Availability	50%	3	Medium	R-1
Schedule slips	70%	1	High	R-2
System goes hour	60%	3	Medium	R-3
Project canceled	30%	4	Low	R-4
False feature rich	40%	2	Low	R-5
Programmers doesn't have good experience	50%	3	Medium	R-6
Late delivery	50%	3	Medium	R-7
Customer Participation in Beta Testing	30%	4	Low	R-8

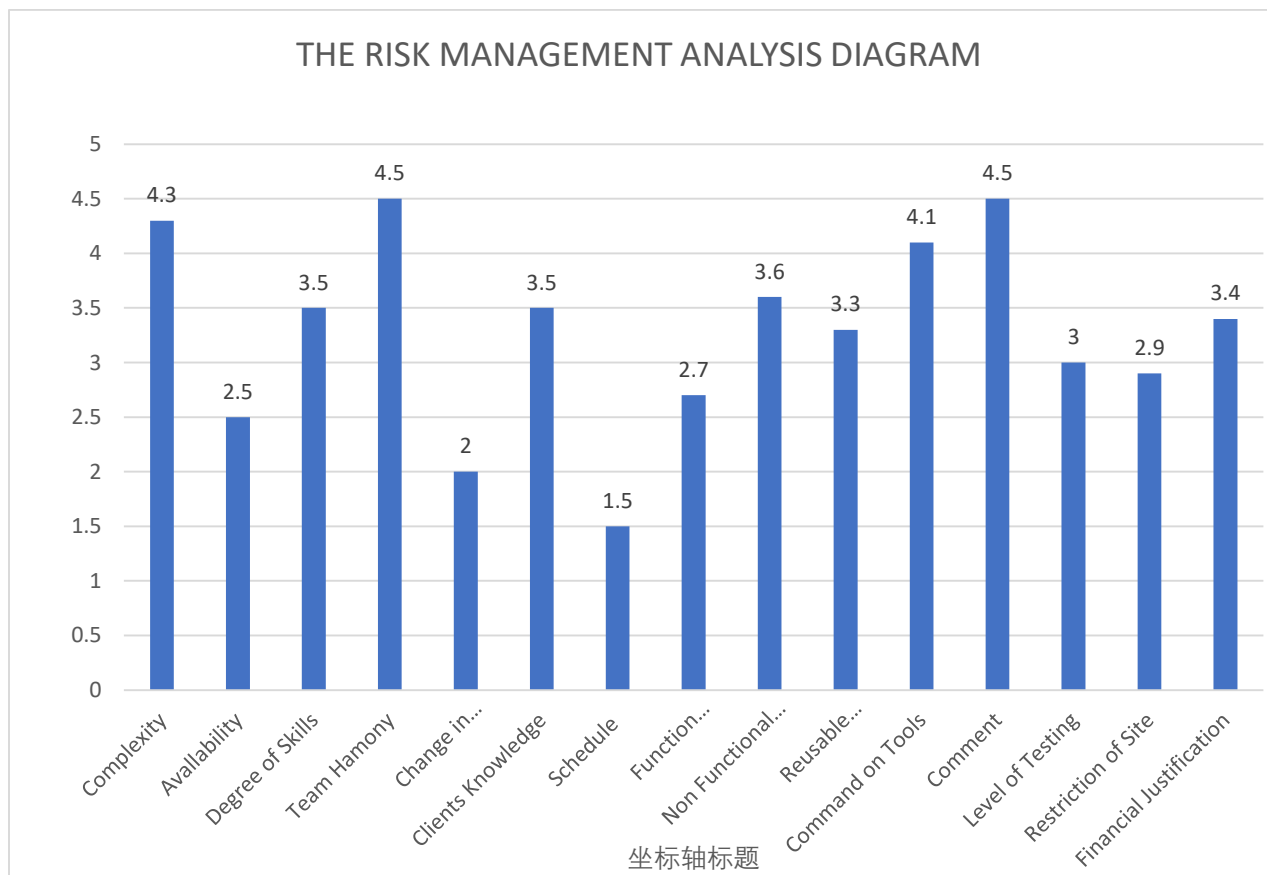


Figure: 1.3

Below is a list of the threats for each group. A definition, the likelihood of occurrence, for every risk the associated action and the effect of the risk shall be given. It's clear that there are going to be issues during the project. The following rules are for avoiding problems.

All Team members will follow:

- Pay attention to detail to make sure the same thing is understood to everybody.
- To better coordinate and maintain quality continuity, obey the recommendations set out in (SQAP) and (SCMP).
- Focus on decided usage specifications reflecting the desires of the consumer
- Reduce people's tension by shared cooperation and encouragement.
- Attempt to signal concerns as quickly as possible and send them to the PM so that steps can be taken.

List of Deliverables:

The operating concepts will be defined in the framework documents. The execution consists of a device introduction, a review of the operating system and the satisfactory passage of the approval test. The acceptance test to be demonstrated successfully remotely over the Internet September 10, 2020. All deliverables of work will be on September 10, 2020.

Deliverable work Products

Name	Standard	Preparer	Reviewer	Date	Distribution List
Software Design Specification (SDS)	IEEE 1016-1998	IEEE 1016-1998	Project Manager	August 19, 2020	Document repository, Preparer, Reviewer, PM
End-user Documentation	IEEE 1063-2001	Technical Writer 1	Project Manager	September 10, 2020	Document expository, Preparer, Reviewer, PM
Software	IEEE 830-1998	Requirements	Requirements	September 10, 2020	Document
Software Project Management Plan (SPMP)	IEEE 1058-1998, IEEE 1074-1997	Project Manager	Dhaka Subway Project Manager	July 27, 2020	Document repository, Preparer, Reviewer, PM

Software Test Plan (STP)	Software Test Plan template (adapted from IEEE 829-1998)	Verification Engineer 1, Verification Engineer 2	Project Manager	August 17, 2020	Document repository, Preparer, Reviewer PM, Programmer 1
Software Quality Assurance Plan (SQAP)	IEEE 730-2002	Quality Analyst 1	Project Manager	August 18, 2020	Document repository, Preparer, Reviewer PM
Software Verification	IEEE1012-1998, IEEE 1012a-1998	Verification Engineer1,	Project Manager	September 10, 2020	Document repository

Defect Tracking Process:

This document is descriptions of the mechanism that accompanies the error or flaw. This also goes into explanation about how a. In the error monitoring log is inserted error /defect.

Lifecycle of Defect:

- Enable Bug / Defect with bug design and error monitoring tools
- Assigned to developer or person in charge of repairing
- App Bug Reports
- Builder acknowledges error or rejects error for various motives
- If the author discovers a error, otherwise a patch is in progress
 - Designer patches bugs and passes them back to those who installed them
 - QA confirms fix is finished in proper construction
 - QA Fixes the bug

When the developer rejects the bug, the developer must allocate the bug to the person who opened it with justification declining:

- QA either understands cause for closure or may reopen bug with more information or facts.
- Developer can provide explanations for: not a bug, by design, not reproducible, bug repeat.

Metrics:

Metrics	Description
Schedule	Milestones
Support efficiency	Charts of hours of person used every month Both predicted and real
Estimation	Average consumption figure over time They both projected to be real
No. of Requirements	Full specifications figure Identified over time, per module
Defects of Requirements No.	Number of defects found per graph system with time
Object No.	Charts of number of known objects then
Coding Performance	Amount of coded objects
Size of coding	Measured regular lines of code
Testing progress	The triggers of the unit assessments passed over time Amount of Test integration
Defect Tracking	Number of defects in code Number of software code defects Gone over time

Postmortem

The project was launched to solve some issues that now a days the passengers are facing while they book a ticket from a ticket counter. The problems are:

- A passenger have to maintain a long queue for booking a single ticket.
- Sometimes they have been told that there is no seats available for booking whether there is available seats to be booked.
- Customers cannot cancel a ticket.

The earlier ticket issuing system was manual that means it need to be managed by a team of humans. As a result the above mentioned problems were faced by the passengers. Also some peoples were doing black marketing with tickets.

The result of our project:

- Passengers don't have to stand in a long line.
- They can see exact available seats.
- They can book a ticket by themselves.
- We can prevent black marketing.

As our project's motive was to issuing ticket for a passenger by inserting his credit card and validating the pin we finalized our project by estimation, testing, scheduling, management, controlling and we think the passengers now can enjoy the results of our project.

For the next step or, next development we would like to work on ticket cancelation which was not been a part of our project goal. So, we can say it is a bad side of our project that the passenger still cannot cancel their booked ticket. Some other bad sides of our project is:

- A passenger may not have a credit card.
- A passenger may not take off to his destined location intentionally (if there is no TT-Traveler Ticket Examiner).

So, we would love to work on this aspects on our next development attempt.