Software Requirements Specification

For

Transport4You

Version 1.0 approved

Prepared by Li Yi, Yang Hang and Wu Huanan

National University of Singapore

Created on 15/10/2010

Revision History

Date	Description	Author	Comments
15/10/2010	Version 0.9	Li Yi	First Draft
23/02/2011	Version 1.0	Li Yi	Final Draft

Document Approval

This Software Design Document has been accepted and approved by the following:

Signature	Name	Title	Date
	Yang Hang	Project Manager	28/02/2011

Table of Contents

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) identifies the system-level requirements for the Transport4You Intelligent Public Transportation Manager (IPTM) system. The objectives of this specification of the Transport4You IPTM system are to provide an overview of the system, including definition, goals, objectives, context, as well as major capabilities and specify its associated functional requirements, quality requirements and constraints.

1.2 Intended Audiences

The intended audiences of stakeholders for this specification of the IPTM include:

- Transport4You IPTM system development team:
  - Project Manager.
  - Architectural Designers, whose design must meet the requirements specified in this SRS.
  - Developers, whose implementations have to follow the requirements specified in this SRS.
  - Testers, who must validate the system according to the requirement specifications.
- Transport4You IPTM system project stakeholders (clients):
  - Mangers.
  - Customer Representatives, who must approve it.
- End users:
  - Administrators, who maintain and manage the central mainframe.
  - Registered passengers, who use the services provided by the IPTM system.
  - Unregistered passengers, who register to the IPTM system through the "Transport4You user portal".

1.3 Specification Overview

This specification is organized into the following sections:

- Section 1 (Introduction), which introduces the specification for the IPTM system to its readers.
- Section 2 (Overall Description), which provides a brief, high-level description of the IPTM system including its definition, backgrounds, context and capabilities.
- Section 3 (Functional Requirements), which specifies the functional system requirements in terms of a use case model.

- Section 4 (System Quality Requirements), which specifies the required system quality factors.
- Section 5 (System Constraints), which documents required architecture, design, and implementation constraints on the IPTM system.
- Section 6 (Appendices), which defines ancillary information including requirement group definitions and sorted requirement list.

2. Overall Description

This section provides a high level description of the Transport4You Intelligent Public Transportation Manager (IPTM) system, including its definition, backgrounds and system capabilities.

2.1 Definition

The Transport4You intelligent public transportation manager system is a specifically designed municipal transportation management solution which is able to simplify the fare collection process and provide customized services to each subscriber.

2.2 Backgrounds

Up until the 1970s and into the early 1980s, conductors were a common feature of many local bus services in larger towns and cities in the UK and Ireland(contributors, 2010). Conductors can still be found on buses in many countries even today. However, with the fast development of information and communication technologies, automated fare collection systems and many types of digital tickets have replaced bus conductors, which greatly enhances the efficiency and accuracy of the fare collection process. To further improve the service quality and effectiveness, many municipal transportation authorities are seeking new public transportation management solutions.

To be specific, a system that is able to provide customized trip information and timely responses to each subscriber is to be built to satisfy the increasing needs. In other words, the new system should not only play the role of a bus conductor, but also be a trip advisor who informs the users of changes in the lines and possibly suggests the optimized route for them. Moreover, to free passengers from troublesome boarding procedures, the system should not require users to do any additional actions, such as, tap a card or insert a ticket. The system should also be reliable and fault-tolerant in the sense that no major glitches should occur during the normal operating hours and no serious consequences should be caused to avoid any loss or inconvenience of users. In addition, the system must be easily extendable to larger scales to fit various situations in different cities.

2.3 Context

This subsection documents the context of the IPTM system and the externals that it interacts with. As illustrated in Figure 1 below, the IPTM system interacts with five external actors and systems:

- Administrators
- Registered Users
- Unregistered Users

- Mobile Detection Hardware (equipped on buses)
- Networks, including server-bus communication network and GSM network.

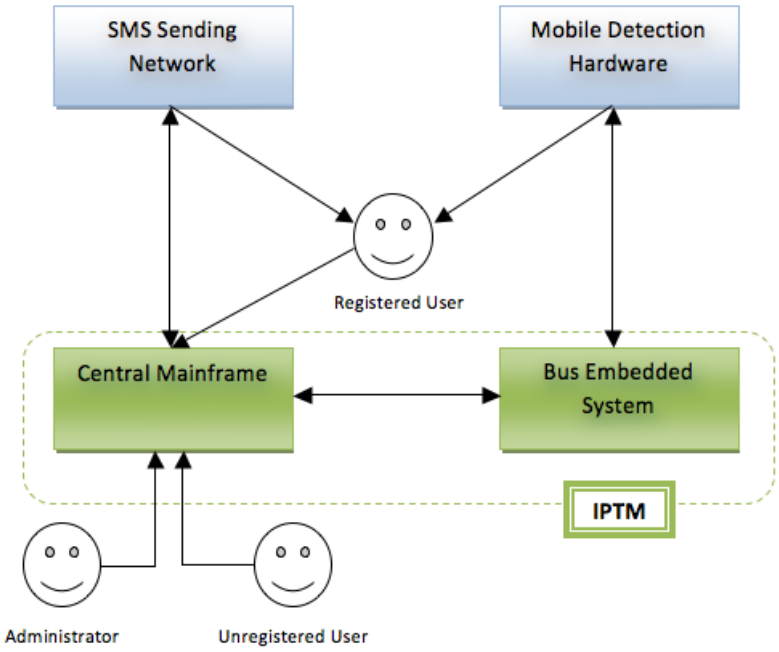

Figure 1 Intelligent Public Transportation Manager Context Diagram

2.3.1 External Hardware

The IPTM system interacts directly or indirectly with the following external hardware:

- Mobile Detection Hardware: the subsystem, bus embedded system retrieves the identification information of the current-boarding passengers by constantly communicating with the mobile detection hardware, which is installed on every bus.
- Network:
  - Internet, which is used by users to communicate with the central mainframe.
  - GSM Network, which is used to send SMS to registered users.

2.3.2 External Roles

The IPTM system interacts directly or indirectly with the following external roles:

- Administrators, who are in charge of maintaining and managing the central mainframe.
- Registered Users, who are able to view and update their account information by indirectly accessing the central server via a user portal.
- Unregistered Users, who are able to register to the IPTM system by indirectly accessing the central server via a user portal.

2.4 System Capabilities

This subsection provides a high-level overview of major capabilities of the IPTM system. Note that this subsection provides useful information for understanding the following functional requirements, but does not contain specific testable requirements.

2.4.1 Administrator Capabilities

The IPTM system provides following capabilities to administrators:

- User Behavior Analysis
- Road Condition Update
- User Trip Information View
- User Account Information View

2.4.2 Registered User Capabilities

The IPTM system enables registered users to:

- Receive System Notifications (via SMS)
  - Fare deduction and e-ticket purchase notification
  - Inadequate balance notification
  - Road condition update notification
- Manage Account, including:
  - Add value
  - Personal particulars modification
  - Account settings modification
- Automatically purchase e-ticket when boarding

2.4.3 Unregistered User Capabilities

The IPTM system enables unregistered users to:

- Create account

3. Functional Requirements

This section specifies the functional requirements of the IPTM system in terms of use cases and their associated use case paths. The use case model is primarily organized in terms of the external roles that benefit from the use cases.

3.1 Summary Use Case Diagram

The following use case diagram (Figure 2) summarizes the functional requirements of the IPTM system.

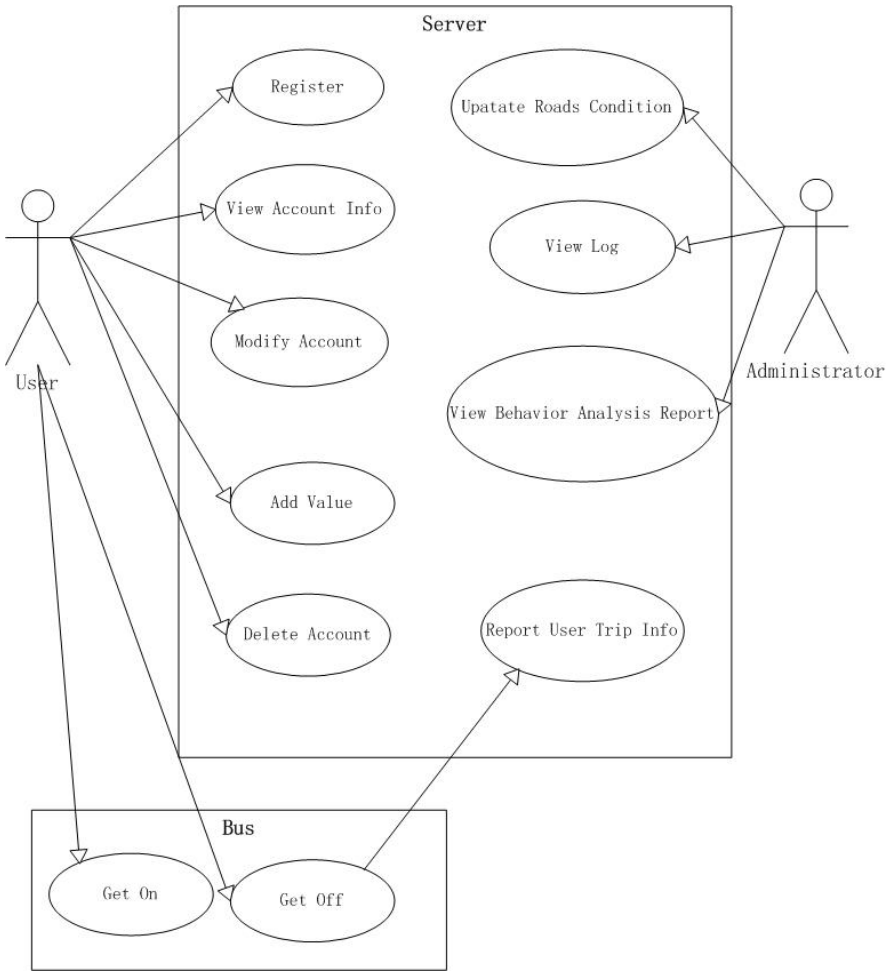

Figure 2 Summary Use Case Diagram

There are mainly two actors directly interacting with the system, namely user and administrator. The system also contains two sub-systems, located on server and bus respectively. Users interact with both the server system and the bus system, while administrator only manages and maintains the server system.

3.2 External Roles

This subsection describes and specifies external roles, the associated responsibilities, and all use cases primarily driven by these externals.

3.2.1 Administrator

The subsection specifies the functional requirements primarily associated with administrators.

Definition

An *administrator* is the role played by a municipal transportation authority employee who is in charge of the central server maintenance.

Use Cases

- Update Road Condition
- System Logs and Reports Monitor

3.2.1.1 Use Case: Update Road Condition

Level: User-goal

Primary Actor: Administrator

Stakeholders and Interests

- Administrator: Wants to update conditions of lines. Wants to get correct response on whether his operation succeeds.
- Transportation Authority: Wants to know the conditions of bus lines accurately. Wants to alert related users of changes in lines.
- User: Wants to be notified of changes in lines.

Preconditions: System is working normally.

Post Conditions: Changes in lines are accepted. Related users are notified.

Main Success Scenario

1. Conditions of some lines have changed.
2. Administrator logs in.
3. Administrator inputs the line-change information including line number, stops affected, duration, and further details.
4. Server accepts the changes and informs the administrator that his or her input is accepted.
5. Server finds out the users related to the lines affected.
6. Server sends SMS alert to all the related users.
7. Server logs the event.

Extensions

3a. The input is illegal.

1. Server discards the input.
2. Server shows an error message to the administrator.

Special requirements

- System should be fault-tolerant. System is able to handle unexpected information or messages and remain working stably.

Technology and Data Variations List

5a. AI planning techniques

3.2.1.2 Use Case: System Logs and Reports Monitor

Level: User-goal
Primary Actor: Administrator
Stakeholders and Interests
- Administrator: Wants to monitor reports and logs in the system.
Preconditions: Administrator logs into the system.
Post Conditions: N/A
Main Success Scenario
1. Administrator logs into the system
2. Administrator chooses which information he or she wants to monitor
3. System displays the information to Administrator
Extensions
1-2a. Network fails (the connection is unavailable).
1. Administrator tries to log in again

3.2.2 Registered User

The subsection specifies the functional requirements primarily associated with registered users.
Definition
A *registered user* is a passenger who has already registered to the IPTM system.
Use Cases

- Get on Bus
- Get off Bus
- Add Value
- Delete Account
- User Account Management

3.2.2.1 Use Case: Get on Bus

Level: User goal
Primary Actor: Registered User
Stakeholders and Interests
- Passenger (registered user): Wants to get on bus without complicated procedures. Wants SMS notification about trip information after getting on bus. Wants to be informed if his or her account balance is too low.
- Transportation Authority: Wants to manage the passenger's account. Wants to inform passenger the trip information via SMS. Wants to notify passenger if his or her account balance is inadequate. Wants to record the transactions consistently.
- Administrator: Wants to monitor network status. Wants to fix network errors efficiently. Wants to check users' status and account information at anytime.

Preconditions: Bus is equipped with the embedded user detection system. The passenger is registered.
Post conditions: The server saves Passenger's get-on information. Payment is authenticated and recorded. Confirmation message is sent to the user. Notification is sent to the user if his or her account balance is inadequate.
Main Success Scenario
1. A registered user gets on the bus.
2. Bus detects the user's cell phone.
3. Bus records get-on information, including user identity, bus stop, line number, and boarding time.
4. Bus transfers the information generated in step(3) to the server.
5. Server receives the information from the bus and records it.
6. Server handles the payment.
7. Server sends a confirming SMS containing the transaction information to the user, including bus stop, line number, boarding time, payment method, and the payment amount.
8. User receives a SMS from the server.
Extensions
4-5a. Network fails (connection is unavailable):
1. Bus delays the transmission.
4-5b. Network communication times out.
1. Server discards the transmission.
2. Bus delays the transmission.
6a. Server checks the user's status
1a. User holds a valid ticket (He or she has purchased a ticket and it is still valid).
1. Server does not charge the user.
1b. The user doesn't hold a valid ticket.
1. Server Checks the payment method
1a. Pay by pre-pay account
1. The server checks the user's pre-pay account.
1a. The account has more than T_COST+UNPAID.
1. Server updates the passenger's account by subtracting the T_COST+UNPAID.
2. Server sets the UNPAID zero when UNPAID is not equal to zero.
1b. The account has less than T_COST+UNPAID.
1. Server notifies the user through a SMS.
2. Server updates the UNPAID by adding the bus fare.
2. The server records the payment transaction.
1b. Pay by credit card.
1. Server charges T_COST+UNPAID on the user's credit card.

2. Server sets the UNPAID zero when UNPAID is not equal to zero.
3. Server generates a ticket for the user stamped with the start time.
4. The payment is authenticated and recorded.

Special requirements

- System should be fault tolerant. System is able to handle unexpected information or messages and remain working stably.
- System should be able to provide services to concurrent requests.

Technology and Data Variations List

- Wi-Fi or Bluetooth is used to detect cell phones.
- The cellular devices are detected only after the door is closed and the bus starts moving to the next stop.

3.2.2.2 Use Case: Get off Bus

Level: User goal

Primary Actor: Registered User

Stakeholders and Interests

- Passenger (registered user): Wants to get off bus without complicated procedures.
- Transportation Authority: Wants to match and record user's get-off information consistently with his or her get-on information.
- Administrator: Wants to monitor network status. Wants to fix network errors efficiently. Wants to check users' status and account information at anytime.

Preconditions: Bus is equipped with the embedded user detection system. The passenger has registered. The passenger is already detected on the bus.

Post conditions: The server records Passenger's complete journey information.

Main Success Scenario

1. User gets off the bus.
2. Bus detects the user is getting off.
3. Bus records the get-off information, including user identity, bus stop, line number, and get off time.
4. Bus sends the information generated in step(3) to the server.
5. Server receives the user's get-off information from the bus.
6. Server records the user's get-off information together with user's get-on information.

Extensions

4-5a. Network fails (the connection is unavailable).
1. Bus delays the transmission.

4-5b. Network communication times out.
1. Server discards the transmission.
2. Bus delays the transmission.

Special requirements

- System should be fault-tolerant. System is able to handle unexpected information or messages and remain working stably.
- System should be able to provide services to concurrent requests.

Technology and Data Variations List

- Wi-Fi or Bluetooth is used to detect cell phones.
- The cellular devices are detected only after the door is closed and the bus starts moving to the next stop.

3.2.2.3 Use Case: Add Value

Level: User goal
Primary Actor: Registered User
Stakeholders and Interests
- Passenger (registered user): Wants to add value into his or her prepay account balance.
- Transportation Authority: Wants to make sure the account balance is always correct.

Preconditions: The passenger has registered.
Post conditions: User gets notification from Server that whether his or her operation is successful.
Main Success Scenario
1. User logs into the system.
2. User chooses add-value function.
3. User indicates his or her bank account and amount of value wanted to add.
4. System deducts value from user bank account.
5. Server sends successful notification to user.

Extensions
1-2a. Network fails (the connection is unavailable).
   1. User tries to log in again.

3-4a. Network communication times out.
   1. Server discards the trade.

3.2.2.4 Use Case: Delete Account

Level: User goal
Primary Actor: Registered User
Stakeholders and Interests
- Passenger (registered user): Wants to delete his or her account.

Preconditions: The passenger has registered.
Post conditions: User gets notification from Server that whether his or her operation is successful.
Main Success Scenario
1. User logs into the system.
2. User chooses delete account function.
3. System deletes users account.

4. System returns pre-account value into user's bank account.
5. Server sends successful notification to user.

Extensions

1-2a. Network fails (the connection is unavailable).
1. User tries to log in again.

3.2.2.5 Use Case: User Account Management

Level: User goal
Primary Actor: Registered User
Stakeholders and Interests
- Passenger (registered user): Wants to add value into his or her pre-paid account balance.
- Transportation Authority: Wants to make sure user's account information is always valid.

Preconditions: Passenger has registered.
Post conditions: User gets notification from Server that whether his or her operation is successful.
Main Success Scenario
1. User logs into the system.
2. User chooses maintain account function.
3. User modifies account information.
4. System checks the account information after modification and judge it is still valid.
5. Server sends successful notification to user.

Extensions

1-2a. Network fails (the connection is unavailable).
1. User tries to log in again.

4a. Account information is invalid.
1. Server discards the trade.

5a. Network communication times out.
1. Server discards the trade.

3.2.3 Unregistered User

The subsection specifies the functional requirements primarily associated with unregistered users.
Definition
An *unregistered user* is the role played by a passenger who does not own an IPTM account.
Use Cases
- User Registration

3.2.3.1 Use Case: User Registration

Level: User-goal
Primary Actor: Unregistered User

Stakeholders and Interests

- Unregistered User: Wants his information accurately stored. Wants to get correct and timely response on whether his registration is successful.

- Transportation Authority: Wants to store the information of new user accurately. Wants to check whether the information provided is valid and legal (the data format is correct and no conflict with existing accounts).

Preconditions: The system is working normally.

Post Conditions: User's information is validated. User account has been created and all information is stored accurately. The user is informed by a system message.

Main Success Scenario

1. New user accesses the web interface and chooses the registration function.
2. System provides a HTML form for user to fill in.
3. New user fills in the form with personal information including username, password, name, address, contact number, mobile identifier (either Bluetooth address or Wi-Fi Mac address), and then clicks "Submit".
4. System validates the information provided by the user and checks it against existing accounts.
5. System creates a new account with the information.
6. System informs the new user that his or her registration is approved.

Extensions

3-6a. Network communication times out:

1. System disregards the pending application even if partial information is received.
2. Web interface terminates the communication process and displays network error message to the user.

4-6a. The information filled in by the user is illegal:

1. Server disregards the information.
2. Server informs the user what the errors are.

4. System Quality Requirements

This section specifies the required system quality factors that are not related to the specific functional requirements documented in the use case model.

4.1 Fault Correction Requirements

In practice, the Bluetooth and Wi-Fi signals are beyond the physical boundaries of a bus. Thus, a detector on bus can detect the users outside the bus and even users on another bus that is nearby. Therefore, when the central server receives a bus detection report, it should try its best to correct the erroneous information that might occur during the detection. For instance, there may be multiple buses report a same subscriber is on board at the same time.

4.2 User Behavioral Analysis Requirements

In general, a user tends to have his or her own traveling patterns, which means he or she takes some buses more often than the others. The system should analyze trip logs to determine what the users' regular routes are. This not only could help service providers re-arrange bus lines to meet the changing needs of the public, but also is useful when related users need to be notified because of occasional bus line interruptions.

4.3 Usability

In both design and implementation, it should be kept in mind that the operations of users should be kept to a minimum. Additionally, it should be very easy to learn how to use the system. Hence, the user interface of the system should be intuitive and concise.

As a measurement of the software usability, the application shall enable at least 90% of a statistically valid sample of representative novice users to:

- Register as a subscriber within 5 minutes.
- Modify his or her personal particulars within 2 minutes.
- Modify his or her account settings within 2 minutes.
- Add value to his or her account within 3 minutes.

The application shall enable at least 90% of a statistically valid sample of representative experienced users to:

- Register as a subscriber within 2.5 minutes.
- Modify his or her personal particulars within 1 minute.
- Modify his or her account settings within 1 minute.
- Add value to his or her account within 1.5 minutes.

4.4 Avoid over charge

In any case, including system failures and undetermined situations, the subscribers should not be over charged by the system. The benefits of subscribers should be valued the most.

5. System Constraints

The section documents the major architecture, design, and implementation constraints on the system.

5.1 Scale Constraints

In the case of this project, it is impractical to implement a real distributed system of such a scale. The testing and demonstration of the system will be difficult if a large number of bus lines and passengers are involved. Therefore, a simulator is to be built for the demonstration purpose. The simulator simulates the activities of buses and passengers, as well as the interactions among them.

5.2 Hardware Constraints

The subsection documents all required constraints associated with minimum or actual hardware.

5.2.1 Limited Bandwidth

Every bus features a GPRS (or higher quality protocol) connection that allows the bus embedded system to interact with the central mainframe. Due to the cost of this connection and its limited bandwidth, its usage should be minimized.

5.2.2 Detection Imprecision

As a drawback to be considered during design and implementation, the Bluetooth/Wi-Fi signal is strong enough to detect cell phones in the immediate vicinity of the bus, but outside it.

6. Appendices

This section documents the following appendices:

- Requirement Groups
- List of Requirements

6.1 Requirement Groups

We categorize the requirements into five requirement groups as shown below:

Table 1 Requirement Groups

Identification	Requirement Group
CORE	Main Application Core
UIM	User Information Management
SIM	Simulator UI and Animation
FC	Fault Correction
UBA	User Behavior Analysis

6.2 List of Requirements

The following table lists all the requirements sorted by their requirement groups for reference:

Table 2 List of Requirements

Identity	Description	Host
	Main Application Core	
CORE-1	Detecting Users	Bus
CORE-2	User boarding/alighting report	Bus
CORE-3	E-ticket Maintains	Server
CORE-4	T_COST and V_VAL setting	Client/ Server
CORE-5	User Account Balance Operation (add value, charging)	Client/ Server
CORE-6	User Notification (via SMS)	Server
	User Information Management	
UIM-1	New User Registration	Client/Server
UIM-2	User setting and personal information modification	Client/Server
UIM-3	User Unregistration	Client/Server
	Simulator UI and Animation	
SIM-1	Animation of Simulator	Simulator
SIM-2	Simulator UI	Simulator
SIM-3	Server basic functions and Communication between server and simulator	Simulator/Server
	Fault Correction	
FC-1	Level-1 Correction [Section 6.1.1]	Bus
FC-2	Level-2 Correction [Section 6.1.2]	Server
	User Behavior Analysis	
UBA-1	User Trips Logging	Server

UBA-2	User Trips Time Distribution Statistics	Server
UBA-3	User Behavior Analysis (with bottom line setting) [Section 6.2]	Server

Works Cited

contributors, W. (2010, December 21). *Conductor (transportation).* Retrieved January 2, 2011, from Wikipedia: http://en.wikipedia.org/w/index.php?title=Conductor_(transportation)&oldid=403472179