Ground Transport System Requirements Specification

CS441, Group 3

Version 1.0

Fall 2013

**By Jack Petraitis, Ibraham Al Thomali, Satabdi Aditya,
Swaraj Gunda, Phani Vempalli, Kevin Richner, Marc Moylan**

Table of Contents

1. Executive Summary

1.1   *Project Overview*

Some companies have the responsibility of transportation of employees. Such employers need a way to calculate the cost of providing this service to all or some employees. Some employees may live in the same area as other employees and hence they can travel together to save the company some money. This system provides the cost of this service depending on the type of vehicle used, number of vehicles needed, and the distance between the company and the area where a set of employees need to go.

The distributed Ground Transportation System provides the ability for companies to offer transportation of employees using one of three transportation types of vehicles. The types are a motorcycle, cab, or a bus. Each type has a different capacity with a standard rate per distance. The distance is based on zones. A zone is a location identifier that is assigned to the physical addresses (address of company as well as address of employees). The price is based on the distance i.e. zone to zone and type of vehicle.

1.2   *Purpose and Scope of this Specification*

The purpose of this specification is to outline the features of our distributed system to companies who would like to implement it with their current logistics.

This documentation is meant to describe the context of the problem being solved, and the different features that will be required in order to have an acceptable solution to this problem. This document is aimed at potential customers or reviewers of our product.

The scope of this document includes recommendations for important features and a loose description as to how our program will function to meet these requirements. Items outside the scope of this document are design implementation.

1.3   *Goals and Objectives*

Most companies have transportation services for employees. The main objective of this software is to minimize the cost of transportation for the employees. For example: The employees living in the same area would be able to share a vehicle to work. The factors that have been used for the minimization of cost using this software are: the type of vehicle used, the number of vehicles needed and the distance between the company and the area where a set of employees needs to go.

The goal of this project is to provide companies the ability to calculate the cost of transportation of employees. Any company can register their employees and receive a price quote for a specific mode of transportation. One of the goals is to have the company request pricing for different modes of transportation for a set of employees. The system must provide the number of vehicles needed to transport a specific set of employees. Hence, a company can decide on who is provided with a specific mode of transportation and manage the cost of such services. Another goal from the vehicle service provider side of the system is to accommodate the lowest number of vehicles needed to transport a set of employees traveling to the same destination zone. Finally, the goal of the transportation management system is to manage all requests from companies and to insure weather conditions are ideal for vehicle operation.

2. Product/Service Description

2.1   *Product Context*

This product is a conglomeration of other products out on the market today. There are no comprehensive human logistic solutions available on our market. Our distributed system takes the

major features of publicly available transportation information and combines them with some privately contracted transportation options.

Our program is supposed to act as an interface to every mode of ground transportation available to get an employee to work. The different modes are Cab Services, Motorcycle Services, and Bus Services. Each of these services will be briefly touched on here, but more thoroughly explored in Section 3.6.1.

The Cab Services interface will work with a cab server that we program to calculate how many cabs will be needed to transport employees from their home zone to the work zone. This will also calculate the total expected cost for the trips.

The Motorcycle Services interface will work with a motorcycle server that we will populate with our own data. This part of the program will find out how many motorcycles would be required to take all the employees from a certain zone and transport them to the work zone. An interesting note about this service is that it cannot operate in the rain. This will be discussed further in Section 3.6.1.

The Bus Services interface will connect to a bus server. The objects retrieved from this information database will help us calculate how many buses are needed to transport employees in one bus service zone to the work zone.

The Train Services interface will connect to a train server. The objects retrieved from this information database will help us calculate how many trains are needed to transport employees in one train service zone to the work zone.

2.2   User Characteristics

The typical users of this program will be the following:

- Human Logistics Engineer
- Transportation Manager
- Financial Reviewer

2.3   Assumptions

We are assuming that the operating systems that run the client systems have a java compiler and are capable of running jar files generated and designated as the client. We are also assuming that the server programs developed are installed onto Linux machines.

2.4   Constraints

The constraints on our project include:

- Databases must be between 300MB and 2GB.
- Must have 20 database tables with 5 average attributes
- Each table must have primary keys and some foreign keys
- Data must come from a publically available source using another API

2.5   Dependencies

The program will depend on the data availability from a weather service, and the uptime of our databases.

3. Requirements

3.1   Functional Requirements

In the table below, the requirement numbering has a scheme - BR_0## (BR for Business Requirement). This scheme was designed with having the goal of making the traceability matrix easier to create. SME stands for "Subject Matter Expert". TMS = "Transportation Management System".

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_01	Upon client startup connect to weather services via WeatherAPI	Used for motorcycle service	1	9/26/13	Ibrahim
BR_02	Upon client startup connect to Motorcycle service	Requires successful Weather connection	1	9/26/13	Teja
BR_03	Upon client startup connect to Bus Service		1	9/26/13	Jack
BR_04	Upon client startup connect to Cab Service		1	9/26/13	Swaraj
BR_05	Upon client startup connect to Train Service		1	9/26/13	Satabdi
BR_06	Client has the ability to generate random database of employees	This should be a command or a button	2	9/26/13	Marc
BR_07	The client connects to a TMS server		2	9/26/13	Kevin
BR_08	The client can choose a type of transportation and the TMS object calculates the number of employees who can take this mode of transportation		2	9/26/13	Swaraj
BR_09	The client can have the TMS object calculate the financial burden of transporting employees		2	9/26/13	Ibrahim
BR_10	There can be multiple clients performing operations using the TMS object		2	9/26/13	Satabdi
BR_11	A transportation manager has the ability to take down a server instance		3	9/26/13	Jack

3.2 User Interface Requirements

All of the commands that a customer can run with his/her client must be accessible from the start page of the app. Each button should have text describing what it does. The user should not have to navigate away from the page to see the results of his/her operations.

3.3 Usability

There should be a help context menu that is available by entering the help command into a command line interface.

3.4 Performance

Static Numerical Requirements

The server application will allow only one terminal from each company to be opened at any given time. This limitation will prevent data inconsistencies between client and database and help to lower the chance of deadlocks within the system. It will also lead to a higher efficiency in that it will

eliminate the processing of the same request multiple times. In order to provide a greater level of functionality, the system will be able to provide services to at least 100% of the registered users simultaneously. As the number of users grows, more hardware will have to be added to maintain this level of service. The system will be able to process only the data specified in the data requirements. Any erroneous data will simply be discarded at the time of process.

Dynamic Numerical Requirements

Although the system will have the ability to service all registered users simultaneously, this represents only an extreme case. On average the system will expect to see a 50-75% load at peak times and no more than a 25% load at off peak times. During these times the amount of data transfer and processing will vary. During peak times the system will be expected to handle a maximum of 10Gbps data transfer and 2Gbps during off peak times. While the system is under these limits, users can expect to see response times between 1 and 10ms dependent on network conditions.

3.4.1 Capacity

The maximum users we are expecting with one server instance is 5. We are expecting this number to scale as the managers spin up new server instances. We will be experimenting with provisioning of resources using a virtual network to better define this capacity.

3.4.2 Availability

The program is expected to be available:

- Twenty-four hours a day
- In the Chicago area
- Downtime shouldn't affect users because servers will be updated sequentially, balancing the load across other servers as each one is taken down to be updated

3.4.3 Latency

The maximum acceptable latency time is 20ms.

3.5 Manageability/Maintainability

3.5.1 Monitoring

We will rely on the hosting company's hard drive and system health monitors already in place. S.M.A.R.T. hard disk logs will be gathered periodically.

3.5.2 Operations

Normal operations available to the user include:

- Creating, Updating, and Deleting employees
- Randomly generating and populating employee databases
- Taking a server instance down (Manager only)

3.6 System Interface/Integration

3.6.1 Systems Interfaces

Systems interface requirements:

A. Client to TMS System Interface

The client will request a remote object to be created on the TMS Server application. This object will handle calls to further extrapolated systems and return necessary information to the client.

B. TMS System to Weather System Interface

The TMS System will now act as a client and request weather information from the Weather server. This Weather server will be part of a publically-available system that can support API calls. This weather data is necessary for the next system to function.

C. TMS System to Motorcycle System Interface

The TMS System will now act as a client and request a motorcycle management object. Based on the weather data gathered in the previous statement the system will decide if motorcycles are capable of transporting that day and how much it will cost.

D. TMS System to Bus System Interface

The TMS System will act as a client and request a Bus Management object. This will then calculate the cost of transportation and how many buses are needed for a zone.

E. TMS System to Cab System Interface

The TMS System will act as a client and request a Cab Management object. This will then calculate the cost of transportation and how many cabs are needed for a zone.

F. TMS System to Train System Interface

The TMS System will act as a client and request a Train Management object. This will then calculate the cost of transportation and how many trains are needed for a zone.

3.7 Security

3.7.1 Protection

The factors that will protect the system from malicious or accidental access, modification, disclosure, destruction, or misuse are SQL user accounts.

- Only managers can take down server instances or delete data
- All other users can create, edit, and randomly generate employee data

3.8 Data Management

Information that is to be placed into a database, includes

- Randomly generated employee data, including addresses and zones
- Workplace zones that the employees need to get to
- Company data for each company that uses the service
- Cab data
- Train data
- Bus data
- Motorcycle data
- Weather data

4. User Scenarios/Use Cases

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David starts up Ground Transport System and sees the web user interface. He goes to generate database and clicks on the button. He is sent a confirmation notice once the employees are in the database.

Use Case Name:	Register Employees
Actors:	Customer
Preconditions:	Database should be running.
Description:	1. The company wants to provide transportation for all the employees. 2. The company registers details like name, age, grade, gender of all the employees into the database. a. The employee details are populated automatically.
Postconditions:	All company employees are in the database.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to select which of his employees need transportation. He uses the web interface to designate Zone Logan as the zone that needs transportation.

Use Case Name:	Select Employees
Actors:	Customer
Preconditions:	The database should be up and running. It should have details of all the employees.

Description:	1. The system shows the criteria available for choosing employees who need transportation.
	
	2. The customer chooses one criteria from the available options. The criteria can be from below
	
	    a. Age of the employee
	
	    b. Gender of the employee
	
	    c. Grade of the employee

Postconditions:	We have the criteria to be used for selecting employees who need transportation.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to select what mode of transportation he wants to calculate limits and prices for. He clicks on the vehicle button to select his choice.

Use Case Name:	Select Mode
Actors:	Customer
Preconditions:	The database should be up and running. The different modes of transportation should be available in the database.

Description:	1. The company wants a specific type of vehicle for transportation of all the employees.
	
	2. The company can choose one vehicle type from the following
	
	     a. Bus
	
	     b. Cab
	
	     c. Motorcycle

Postconditions:	Vehicle type for transportation is chosen.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to select what mode of transportation he wants to calculate limits and prices for. He clicks on the "Bus" button to see how many busses he will need to hire and how much it will cost.

Use Case Name:	Request Bus
Actors:	Customer
Preconditions:	Selected vehicle type should be 'Bus'
Description:	1. The company would like to use bus for transportation. 2. The request is sent to Transport Management System with the specified employee criteria.
Postconditions:	The company gets the total price and number of busses needed for

	transportation.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to select what mode of transportation he wants to calculate limits and prices for. He clicks on the "Cab" button to see how many cabs he will need to hire and how much it will cost.

Use Case Name:	Request Cab
Actors:	Customer
Preconditions:	Selected vehicle type should be 'Cab'
Description:	1. The company would like to use cab for transportation. 2. The request is sent to Transport Management System with the specified employee criteria.
Postconditions:	The company gets the total price and number of cabs needed for transportation.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to select what mode of transportation he wants to calculate limits and prices for. He clicks on the "Motorcycle" button to see how many motorcycles he will need to hire and how much it will cost.

Use Case Name:	Request Motorcycle
Actors:	Customer
Preconditions:	Selected vehicle type should be 'Motorcycle'
Description:	1. The company would like to use motorcycle for transportation. 2. The request is sent to Transport Management System with the specified employee criteria.
Exception:	Required vehicle type is not available because of bad weather. The customer is informed of this situation and must provide another mode of transportation.
Postconditions:	The company gets the total price and number of motorcycles needed for transportation.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to see if he should bother calculating the motorcycle vehicle option. Even though the server would reply with an error anyways, he wants to know what the weather will be like for today. So he clicks on the "Check Weather" button.

Use Case Name:	Weather check
Actors:	Customer
Preconditions:	Selected vehicle type should be 'Motorcycle'. Weather public API is available.
Description:	1. The information about weather is required for motorcycle to operate. 2. Connect to weather API and get the weather information. 3. The motorcycle transportation is possible when the weather is good.
Exception:	Required vehicle type is not available because of bad weather. The customer is informed of this situation and must provide another mode of transportation.
Postconditions:	The information about the weather is obtained.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to select where the employees will be coming from and going to. He uses the web interface dropdown menus to select his source and destination zones.

Use Case Name:	Group employees by zone
Actors:	Customer
Preconditions:	The source and destination zones of employee should be available. The weather should be favorable for transportation.
Description:	1. The system must determine number of employees in each zone.
Postconditions:	The number of employees in each destination zone is determined.

Persona: David Davidson is a transportation engineer at ChemWorks, LLC in Chicago, IL. His company just purchased our program and he is using it to get workers from Logan Square to Goose Island.

Use Scenario: David needs to select what mode of transportation he wants to calculate limits and prices for. He clicks on the vehicle button to make his selection. He sets his source and destination zones. He clicks calculate and the system returns the price and number of vehicles required.

Use Case Name:	Compute Price
Actors:	Customer
Preconditions:	The number of employees and vehicle type are available.
Description:	1. The system computes the number of vehicles needed. 2. The system will compute the price for transportation of selected employees. 3. The system will use defined price for this computation.
Postconditions:	The price and number of vehicles required for transportation are computed.

5. Restrictions, limitations, and constraints

5.1 Restrictions

The project must be completed by December 9th, 2013. The RPC implementation must be JAVA RMI. Several deliverable deadlines must be met, including databases, app skeleton, and more documentation.

5.2 Limitations

System specific:
The system limits the functionality only to the companies that are already registered with the TMS. To compute the vehicle count and price number of employees is required and it is available only for the companies which are registered with TMS. The system limits vehicle types only to be bus, cab and motorcycle. A specific constraint exists when the vehicle type is motorcycle. Transportation cannot be possible in motorcycle if the climate is not favorable. The system accuracy is limited by the consistency between the relational and non-relational databases. A coordination of changes must be made between the two in order to provide the best and most accurate results.

Hardware:
The hardware limitations of this is system is the same limitation that are applicable to the Java Virtual Machine. The system should be able to run on any hardware that has a java runtime environment installed on it.

5.3 Constraints

The constraints in this system are formed from the requirements of the project. The following is a list of constraints that the system should adhere to:

- RMI functionality to meet a distributed environment constraint.
- Must implement two types of databases (Relational and Non-Relational)
- Must accommodate unlimited number of clients.
- Implemented using Java as the development language.
- Databases must be between 300MB and 2GB.
- Must have 20 database tables with 5 average attributes
- Each table must have primary keys and some foreign keys
- Data must come from a publically available source using another API

6. Requirements Confirmation/Stakeholder sign-off

Meeting Date	Attendees (name and role)	Comments
9/26/13	Jack Petraitis, Ibraham Al Thomali, Satabdi Aditya, Swaraj Gunda, Phani Vempalli, Kevin Richner, Marc Moylan	Confirmed