**Software Requirements**



**Specification**

**for**

**III-9 NITC Travel Together.**

**Version 1.0**

**Prepared by**

**Group Number:** ​**III-9**

**Tushar Gupta M180499CA**

**Kapil kumar Chhipa M180265CA**

**Abhilasha Sharma M180275CA**

**Soumya Parashar M180260CA**

**Project Owner:Dr. Vinod Pathari**​

**Course: CS4096D Software Engineering Laboratory**

**Date:17-01-2020**

**Contents**

1

[**1 Introduction** **3**](#_Toc18734)

[1.1 Document Purpose 3](#_Toc18735)

[1.2 Product Scope 3](#_Toc18736)

[1.3 Intended Audience and Document Overview 3](#_Toc18737)

[1.4 Definitions, Acronyms and Abbreviations 3](#_Toc18738)

[1.5 Document Conventions 3](#_Toc18739)

[1.6 References and Acknowledgments 4](#_Toc18740)

[**2 Overall Description** **5**](#_Toc18741)

[2.1 Product Overview 5](#_Toc18742)

[2.2 Product Functionality 5](#_Toc18743)

[2.3 Design and Implementation Constraints 5](#_Toc18744)

[2.4 Assumptions and Dependencies 5](#_Toc18745)

[**3 Specific Requirements** **6**](#_Toc18746)

[3.1 External Interface Requirements 6](#_Toc18747)

[3.1.1 User Interfaces 6](#_Toc18748)

[3.1.2 Hardware Interfaces 6](#_Toc18749)

[3.1.3 Software Interfaces 6](#_Toc18750)

[3.2 Functional Requirements 6](#_Toc18751)

[3.3 Use Case Model 7](#_Toc18752)

[3.3.1 Use Case #1 (use case name and unique identifier – e.g. U1) 7](#_Toc18753)

[3.3.2 Use Case #2 8](#_Toc18754)

[**4 Other Non-functional Requirements** **8**](#_Toc18755)

[4.1 Performance Requirements 8](#_Toc18756)

[4.2 Safety and Security Requirements 8](#_Toc18757)

[4.3 Software Quality Attributes 9](#_Toc18758)

[**5 Other Requirements** **9**](#_Toc18759)

[**Appendix A - Activity Log** **9**](#_Toc18760)

**Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| Draft Type and  Number | Tushar Gupta  Kapil Chhipa  Abhilasha Sharma  Soumya Parashar | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

# Introduction

This section of the SRS provides the introduction to what the user can expect from **NITC** Travel Together application, its scope, objective and goals, also giving us the various text specification like bold text, italic, font size and type.

# This project work proposes a NITC Travel Together android application for students so that they can travel together with the NITCmates. The main focus of the project are the students who are travelling to/from same location by sharing the expense of travelling and to get a known company. Also, if there is a group of users who are travelling to the same location or nearby location from the same place, they can simply book a cab, auto or any conveyance and travel together in a group by sharing the fare.

# 

## Document Purpose

This document contains the details of an application named as **NITCTT.**  The application provides **NITC** students a platform to choose whether they want to travel along with other **NITC** students, if they are travelling to a common destination and are within 1 KM radius distance.

## Product Scope

The **NITCTT** enables the students to travel along with other **NITC** students if they have a common destination and are within 1 KM radius. The student enters their destination and finds other students nearby who are travelling to the same destination, and requests for a ride, if the other one accepts their request, their phone number becomes accessible and they can further communicate. Upon exiting from the application, students can rate other students as users of the application.

The objective of the application is to connect **NITC** students so that they can share their rides to common destinations. This application benefits students to save money and have good company of **NITC** students by sharing rides*.*

## Intended Audience and Document Overview

This document describing the **NITCTT** can be useful for various users, including students of **NITC** who travel long distances and are in need to save money and have company of **NITC** students.

The students need to read the document to ensure that their personal information taken by the application will not be made available to other students without their approval.

The project manager reads this document in order to check whether all the specifications required by the user present in the **NITCTT.**

## Definitions, Acronyms and Abbreviations

API- Application Programming Interface.

**NITCTT**-National Institute of Technology Calicut Travel Together*.*

**NITC-** National Institute of Technology Calicut.

OS- Operating System.

SRS-Software Requirement Specification.

S/W- Software

## Document Conventions

* The description area of this document is written in 11 Arial, and corresponding heading in 12 Arial bold.
* Comments have been added in italics.
* Abbreviations in bold in the text area correspond to a noun which could be the application itself or the institute it is made for.
* The subsections are declared by a subsection no. following the section no. separated by a ‘.’ Next to the subsection name.

## References and Acknowledgments

References:

<https://www.guru99.com/functional-requirement-specification-example.html>

<http://staruml.io/>

Acknowledgements:

1. Project Owner and Subject Mentor:

Dr. Vinod Pathari

Associate Professor, CSE Department.

# Overall Description

## Product Overview

*<*​*Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface*​*. In this section it is crucial that you will be creative and provide as much information as possible.*

*TO DO: Provide at least one paragraph describing product perspective. Provide a general diagram that will illustrate how your product interacts with the rest of the environment and in what context it is being used. This is not a formal diagram, but rather something that is used to illustrate the product at a high level. You may draw this diagram using any online tool>*

## Product Functionality

* Register to the application.
* Login to the application.
* Enter travel destination.
* Modify travel destination.
* View nearby students with common destination.
* Offer lift to others.
* Request lift from others.
* Cancel the offer.
* Cancel request.
* Accept request to travel.
* Making phone number available
* Reject request to travel.
* Rate other user.
* Update rating of other user.*<*​*Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary is needed here. These can be at the level given in the project description.> TO DO:*

*Provide a bulleted list of all the major functions of the system. No need to explain them.*

## Design and Implementation Constraints

* Location of the user should be available at all times.
* This application will be using Java as the backend language and xml in the frontend.
* Android studio will be used which minimum 3GB RAM and 8GB needs RAM recommended

Plus 1GB for Android Emulator. Also, minimum 2GB space and 4GB recommended.*<*​*Describe any items or issues that will limit the options available to the developers. These might include: hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software). You can be creative here to some degree. As you know, we will be using Android Studio for development and will follow SDLC. You may mention them here.*​*>*

## Assumptions and Dependencies

* We assume that the user has turned his location at all times.

*<*​*List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project. If you are using a location based project, for example, you may decide to depend on google maps. This should come here.*

*TO DO: Provide a short list of some major assumptions that might significantly affect your design.>*

# Specific Requirements

## External Interface Requirements

### User Interfaces

*<*​*Describe the logical characteristics of each interface between the software product and the users.*

*.*

*TO DO: Provide the GUI of the app and state how users are expected to interact with it>*

### Hardware Interfaces

No specific H/W interfaces required.*<*​*Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware. You are not required to specify what protocols you will be using to communicate with the hardware, but it will be usually included in this part as well.*

*TO DO: Since we do not intend to use sensors or other typical hardware, we may not need this. But if the mess management project people would like to have an interface with the finger print scanners that needs to be mentioned here.>*

### Software Interfaces

* Hosted database from which the application can get the data, is needed.
* To show the current location of the users the application uses Google API.*<*​*Describe the connections between this product and other specific software components. If yours is a standalone app there is nothing to include here* ​*>*

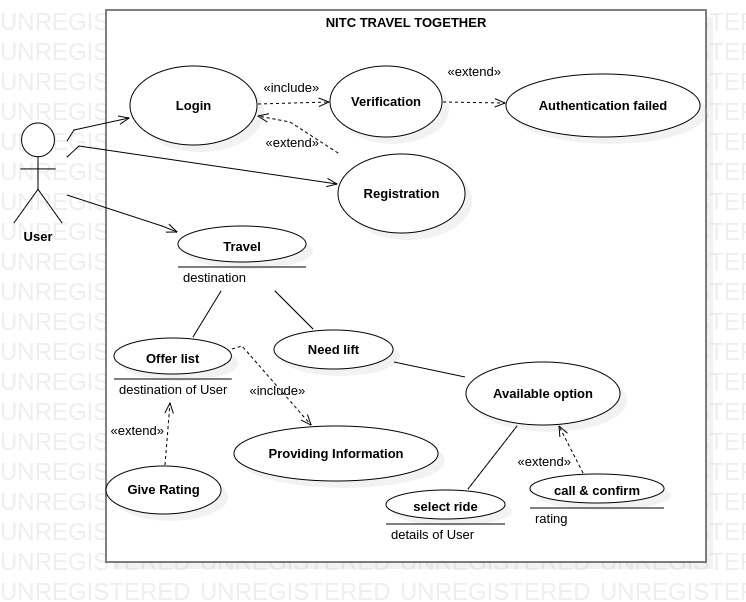
## Functional Requirements

*<* ​*Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. This section is the direct continuation of section 2.2 where you have specified the general functional requirements. Here, you should list in detail the different product functions. It is normal to have 10-15 requirements for each of your projects.*

F1: The system shall … **<**​ **Functional Requirement or Feature #**1​ **>**​F2: The system shall … **<**​ **Functional Requirement or Feature #**2​ **>**​

…

## Use Case Model



*TO DO: Provide a use case diagram that will encapsulate the entire system and all actors. Please follow UML conventions strictly. You need to study about Use Case Modeling before jumping to draw one. Depending on your team number, you will be using either StartUML or PlantUML*

### Use Case 1 (Login U1)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** XXXXXXX

**Purpose**​ - Before using the application user will login so that It can be used by NITC student only.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – Input profile exists in data base. User password matches profile.

**Post conditions**​ – User should logged in into the application and can use it now.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the Actor.

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

* Basic Flow – User logged in into the Application.
* Alternative Flow – Username or password can be wrong. User will be redirected to the same login page.
* Exceptions – Server is not responding.

**Includes**​ (U2)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 2(verification U2)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXX

**Purpose**​ – to check whether User registered or not.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - HIgh

**Preconditions**​ – User must provide username and password for verification.

**Post conditions**​ – User will be redirected to the main application.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor .

**Extends –** ​ authentication fails U3.

**Flow of Events**

1 Basic Flow – Entered username and password will be verified and user will be redirected to main application.

1. Alternative Flow – username or password is incorrect. User will be redirected to login page again.
2. Exceptions – server is not responding or network is not available.

**Includes**​ (other use case IDs)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 3(authentication fails U3)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – if authentication fails user has to try again for login.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – High.

**Preconditions**​ – User must enter username and password for authentication.

**Post conditions**​ – User profile will be redirected to again login page after authentication fails.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

* + - 1. Basic Flow – After authentication failed user will see login page again.
      2. Alternative Flow –After authentication failed user might click on forgot password.
      3. Exceptions – Server is not responding.

**Includes**​ (other use case IDs)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 4(Registration U4)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – User has to registered so that they can login using username and password.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – High.

**Preconditions**​ – User must enter NITC email id, name, mobile number and gender for registration.

**Post conditions**​ – After registration User can login using username and password.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –** Login U1.

**Flow of Events**

* + - 1. Basic Flow – User will enter all the required information and registered.
      2. Alternative Flow –Entered NITC email id is not available.
      3. Exceptions – Server is not responding.

**Includes**​ (other use case IDs)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 5(Travel U5)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – if user wants to travel he/she can use this functionality.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – User will see list of all the people who also wants to travel.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –**

**Flow of Events**

* + - 1. Basic Flow – All the user who also wants to travel will be listed.
      2. Alternative Flow –no one wants to travel at that time.
      3. Exceptions – Server is not responding.

**Includes**​ (other use case IDs)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 6(Offer lift U6)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – Any User can offer lift to other users so that they can share fare.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – User can give rating to the other user.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –** give rating UX.

**Flow of Events**

* + - 1. Basic Flow – User will offer lift and provide his/her information.
      2. Alternative Flow – user can change his/her desire to offer a lift.
      3. Exceptions – Server is not responding.

**Includes**​ (provide information UX)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 7(Give raging U7)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – User can give rating to the other users . it will be helpful for other users.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – rating will displayed on user’s profile.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –** .

**Flow of Events**

* + - 1. Basic Flow – User will give rating to the other user.
      2. Alternative Flow – User can change previous rating of a particular user.
      3. Exceptions – Server is not responding.

**Includes**​ -

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 8(Provide information U8)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – to offer lift to other users one has to provide his/her information. So that they contact each other.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – information will be visible to other user.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –** .

**Flow of Events**

* + - 1. Basic Flow – user will provide his/her information. And other user can see it.
      2. Alternative Flow – user can change his/her desire to offer a lift. And don’t provide information.
      3. Exceptions – Server is not responding.

**Includes**​ -

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 9(Need left U9)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – if user want lift he/she can see how many other users are offering lift.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – User will see available option for lift.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –** .

**Flow of Events**

1. Basic Flow – user will see available option for lift and can select anyone.
2. Alternative Flow – user can change his/her desire to take a lift.
3. Exceptions – Server is not responding.

**Includes**​ -

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 10(Available option U10)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – user can see all the available option for the lift.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – User will see available option for lift.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –** .(call and confirm)

**Flow of Events**

1. Basic Flow – user will see available option for lift and can select anyone.
2. Alternative Flow – user can change his/her desire to take a lift.
3. Exceptions – Server is not responding.

**Includes**​ -

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 11(Select ride U11)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – user can select his/her ride according to his/her convenience.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – After selecting ride he/she has to enter his details.

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –**  NO

**Flow of Events**

1. Basic Flow – User will select the ride and provode his/her details.
2. Alternative Flow – – user can change his/her desire to take a lift.
3. Exceptions – Server is not responding.

**Includes**​ -

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 12(Call and confirm U12)

*TO DO: Provide a specification for each use case diagram. Please refer to similar documents before proceeding to fill this up.*

**Author –** ​XXXXXXXXX

**Purpose**​ – users can call to each other to confirm the ride.

**Requirements Traceability –**Identify all requirements traced to this use case - the F​*n* ​numbers

from Section 3.2 above

**Priority**​ – medium.

**Preconditions**​ – User must logged in.

**Post conditions**​ – users will talk to each other and confirm the ride. They can also give rating .

<Please be careful while filling up Pre and Post conditions>

**Actors**​ – User will be the actor.

**Extends –**  NO

**Flow of Events**

1. Basic Flow – user can conform the ride.
2. Alternative Flow – user refused the ride.
3. Exceptions – Server is not responding.

**Includes**​ -

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

…

# Other Non-functional Requirements

## Performance Requirements

* Username and password verification at the time of login.
* The number of concurrent users may vary.
* Proper and fast internet connection is required for Google API. *<*​*If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.*

*TODO: Provide performance requirements based on the information your idea. You may also take opinion of your project owner for this>*

## Safety and Security Requirements

* The password at the time of login is concealed
* Private user data like phone no. should not be available to other users.*<*​*Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied*​*.* ​*Specify any requirements regarding security or privacy issues surrounding the use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. TODO:*

*● Provide safety/security requirements based on your idea of the App. The owner may help.*

## Software Quality Attributes

* **Availability** : The application will only be available to the students of **NITC,** an individual cannot access the account of another.
* **Robustness:** The application provides complete security of the user. Personal data like phone number is only available to the other users when the request is approved.
* **Maintainability:** With the help of Android studio the application can be maintained. *<*​*Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.*

*TODO: Use subsections (e.g., 4.3.1 Reliability, 4.3.2 Adaptability, etc…) provide requirements related to the different software quality attributes. Make sure that you do not just write “This software shall be maintainable…” Indicate how you plan to achieve it, & etc…. Please note that you need to include* ​***at least*** *2 quality attributes. You can Google for examples that may pertain to your system. Please do not define what reliability, portability etc. in your document. How these attributes are specified for your App is what is required.>*

# Appendix A - Activity Log

|  |  |
| --- | --- |
| Meeting 1 |  |
| Date-14-01-2020 |  |
| Duration-30 minutes |  |
| Team discussed the introduction and idea behind the project. |  |
| Meeting 2 | |
| Date-15-01-2020 | |
| Duration-60 minutes | |
| Team worked on the use cases and user interfaces. | |

*<Provide details of group meetings - when you met and for how long - including the meeting details with the owner. You must also state what was the contribution (the sections mainly, then diagrams) of each of the team members. Team Lead will have complete responsibility and freedom to complete the Activity Log.>*