

---

# Test Document

for

## CAB SHARING

Version 1.0

Prepared by

Group 18:

Group Name: Code Closed

AAKASH LAWLA  
ABHISHEK GURJAR  
AKASH KUMAR BHOI  
RISHABH MUKATI  
PRINCE KUMAR AHIRWAR  
HARIOM SHAKYAWAL  
GOPAL AGGARWAL  
UJJAWAL GOYAL  
SOURABH MINA  
SOURAV ANAND

190006  
190037  
190082  
190704  
190646  
190354  
200390  
201058  
200996  
200997

aaklawla@iitk.ac.in  
gurjara96@gmail.com  
akashbhoi525@gmail.com  
mukatirishabh02@gmail.com  
princeprinceahirwar@gmail.com  
hariomkoli306@gmail.com  
gopalagggarwal5858@gmail.com  
ujjawalgo247@gmail.com  
sourabh2002wow@gmail.com  
souravanand1982@gmail.com

Course: CS253

Mentor TA: Pinaki Chakraborty

Date: 03/04/22



<b>CONTENTS</b>	<b>II</b>
<b>REVISIONS</b>	<b>II</b>
<b>1 INTRODUCTION</b>	<b>1</b>
<b>2 UNIT TESTING</b>	<b>2</b>
<b>3 INTEGRATION TESTING</b>	<b>3</b>
<b>4 SYSTEM TESTING</b>	<b>4</b>
<b>5 CONCLUSION</b>	<b>5</b>
<b>APPENDIX A - GROUP LOG</b>	<b>6</b>

## Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type and Number	Full Name	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

# 1 Introduction

## *Mention the test strategy*

We are using a feasible combination of manual and automated testing (via postman). The main goal was to determine whether the unit/module/system is performing its designated work, so decided to work manually to decrease the discrepancy.

*(Manual testing? Test automation? ...)*

## *When was the testing conducted?*

The test was performed over a period of 6 days (22/3/22– 29/3/22), i.e alongside the implementation, because our software was still updating during the process.

*(in parallel with the implementation? After the implementation was completed? ...)*

## *Who were the testers?*

The tester of this particular software is the developers themselves.

*(the developers? Independent testers? ...)*

## *What coverage criteria were used?*

## *Have you used any tool for testing?*

We have used the testing automation tool provided by the **postman**. And, we have preferred manual testing in most cases.

## 2 Unit Testing

### 1. LOG IN: -

The authentication system of the website was tested using Basic Authorisation from the **postman** unit test tool.

We provided it with the username and password, and it tested the validity of the user.

The login system passed the test giving a status of 200.

**Unit Details:** [Login](#)

**Test Owner:** [Sourabh Mina](#)

**Test Date:** [23/3/22-26/3/22](#)

**Test Results:** The authentication system of the website was tested using Basic Authorisation from the **postman** unit test tool.

We provided it with the username and password, and it tested the validity of the user.

The login system passed the test giving a status of 200.

**Structural Coverage:** Decision coverage

### 2. SIGN UP: -

The signup feature includes new user registration and uses their email and password to register them on our MongoDB database.

**Unit Details:** Router.post

**Test Owner:** Ujjawal Goyal

**Test Date:** [23/3/22- 27/3/22](#)

**Test Results:** The test was via the postman tool, we provided an already registered email. And since it already existed, the postman gave the Failed prompt to our already registered email.

While entering a new email (with password) the status showed 200, i.e. the test passed.

**Structural Coverage:** Branch coverage

### 3. Travelling form:-

We tested the submit function of the form, whether the form got submitted and was reflected in our MongoDB Database.

**Unit Details:** [Router.js](#)

**Test Owner:** Sourabh Mina

**Test Date:** 25/3/22 - 28/3/22

**Test Results:** The test performed was to confirm the updation of our travel database, and it was tested manually, by filling the form and checking the updated form in our database section.

The manual test was a success, the test passed.

**Structural Coverage:** Branch coverage

### 4. My Journey

The My Journey section shows the user's upcoming journey. We tested the page, by manually booking a journey and testing whether it reflected upon the page or not, alongside our database.

**Unit Details:** [My Journey](#)

**Test Owner:** Ujjawal Goyal

**Test Date:** [28/3/22 - 01/4/22](#)

**Test Results:** The My Journey Section is tested manually initially if there was no cab booking by the user, The My Journey page shows nothing. But If a user shares a journey his/her Journey's details are displayed there.

**Structural Coverage:** Decision coverage

## 5. Logout

The Logout button ensures successfully Logging out of the current user, And after logging out their detail can't be visible anymore.

**Unit Details:** [Logout.js](#)

**Test Owner:** [Ujjawal Goyal](#)

**Test Date:** [26/3/22 - 29/3/22](#)

**Test Results:** We tested Logout functionality using Postman, and it shows status 200 (success).

**Structural Coverage:** Decision/condition coverage

### 3 Integration Testing

#### 1. [Request](#)

The test was conducted, by sending a manual request to a user, to verify if the function was working correctly.

**Module Details:** Request.js

**Test Owner:** Sourabh Mina

**Test Date:** 25/3/22-27/3/22

**Test Results:** The test started with the postman automation to give the status of the page while performing every test. Then, we manually conducted the request test by filling a travelling form from a user's id, and searching the travelling form

**Additional Comments:** [\[Enter any summary comments\]](#)

#### 2. [Profile \(Driver\)](#)

The test was to verify the updation system of the profile section, i.e., profile picture (the driver has to upload a picture from the local storage) and personal details. The unit test was performed manually.

**Test Owner:** [Ujjawal Goyal](#)

**Test Date:** 25/3/22-26/3/22

**Test Results:** *The profile page had a status of 200 (postman).*

The profile picture is updated when the driver uploads their picture for the local storage and refreshes the page.

While performing the test, we tested with 4-5 pictures with different extensions, jpeg, jpg, png, etc. The profile picture was updated every time.

The test performed was to confirm the foundation of our driver database, and it was tested manually, by filling the profile section and checking the updated driver details in our database section.

The test was passed.



### 3. Search

The search test was conducted, which shows the number of passengers travelling to a specific location as specified by the user on the particular date.

**Test Owner:** *Sourabh Mina*

**Test Date:** *27/3/22-29/3/22*

**Test Results:** The manual test conducted, goes as follows, we tried searching for a destination without having any prior form submitted for that destination. The search result showed blank, thus, the function was performing smoothly.

Then, we manually filled a form for a particular location (in this case Z-square) on a particular date (29/3/22), and submitted our travelling form. And, tried to search the form from different users' pov.

The result was a success since the form was visible to the user.

Moreover, the search page had a status of 200 while performing the tests. (Postman)

### 4. Profile

The test was to verify the updation system of the profile section, i.e., profile picture (the user has to upload a picture from the local storage) and personal details. The unit test was performed manually.

**Test Owner:** *Ujjawal Goyal*

**Test Date:** *25/3/22-26/3/22*

**Test Results:** *The profile page had a status 200 (postman).*

The profile picture is updated when the user uploads their picture for the local storage and refreshes the page.

While performing the test, we tested with 4-5 pictures with different extensions, jpeg, jpg, png, etc. The profile picture was updated every time.

The test performed was to confirm the foundation of our user database, and it was tested manually, by filling the profile section and checking the updated user details in our database section. The test was passed.

**Additional Comments:** It was performed similarly to the driver's profile case.

## 4 System Testing

### 1. Requirement: Registration

The signup feature includes new user registration and uses their email and password to register them on our MongoDB database.

**Test Owner:** Sourabh Mina

**Test Date:** 25/3/22-27/3/22

**Test Results:** The test was via the postman tool, we provided an already registered email. And since it already existed, the postman gave the Failed prompt to our already registered email.

While entering a new email (with password) the status showed 200, i.e. the test passed.

### 2. Requirement: Update Profile (Both User and Driver)

The test was to verify the updation system of the profile section, i.e., profile picture (the driver has to upload a picture from the local storage) and personal details. The unit test was performed manually.

**Test Owner:** Sourabh Mina

**Test Date:** 25/3/22- 29/3/22

**Test Results:** The profile picture is updated when the user uploads their picture for the local storage and refreshes the page.

While performing the test, we tested with 4-5 pictures with different extensions, jpeg, jpg, png, etc. The profile picture was updated every time.

The test performed was to confirm the foundation of our user database, and it was tested manually, by filling the profile section and checking the updated user details in our database section. The test was passed.

**Additional Comments:** Updation of profile of both user and driver is tested thoroughly.

### 3. Requirement: Travel detail and Match

For testing the requirement of travel detail, we tested the submit function of the form, whether the form got submitted and was reflected in our MongoDB Database.

And the testing for match requirement, The search test was conducted, which shows the number of passengers travelling to a specific location as specified by the user on the particular date.

**Test Owner:** Sourabh Mina

**Test Date:** 25/3/22-27/3/22

**Test Results:** The test was via the postman tool, we provided an already registered email. And since it already existed,

The manual test conducted, goes as follows, we tried searching for a destination without having any prior form submitted for that destination. The search result showed blank, thus, the function was performing smoothly.

Then, we manually filled a form for a particular location (in this case Z-square) on a particular date (29/3/22), and submitted our travelling form. And, tried to search the form from different users' pov.

The result was a success since the form was visible to the user.

Moreover, the search page had a status of 200 while performing the tests. (Postman)

The travelling form test performed was to confirm the updation of our travel database, and it was tested manually, by filling the form and checking the updated form in our database section.

The manual test was a success, the test passed.

## Conclusion

### *How Effective and exhaustive was the testing?*

The testing seemed quite exhaustive and lengthy, but the effectiveness of our testing was high, due to the use of automated tools, it made the quality of testing far better than doing without them.

### *Which components have not been tested adequately?*

Most of the components were tested thoroughly, but we feel that the interface after booking a cab could have done with more testing and verifying if the user is facing any problems or discomfort.

### *What difficulties have you faced during testing?*

The most difficult part for our team was to get familiarised with the postman platform, which was a little milestone for our team to complete.

### *How could the testing process be improved?*

The testing may have been better if it was given to the general public rather than the developers of the software to test the software.



## Appendix A - Group Log

<Please include here all the minutes from your group meetings, your group activities, and any other relevant information that will assist in determining the effort put forth to produce this document>

DATE	TIME	Discussion
1st April	4.00 pm - 4.20 pm	Tell every member to read the Design document and decide to meet on 11th Feb at 4.00 pm.
2nd April	6.00 pm - 6.40 pm	In zoom meet, discuss Design among us, clear some doubts, and discuss unclear doubts with TA on discord. Also divide work between us.
3rd April	1.00 pm - 1.45 pm	Evaluate work among us and help each other where anybody is stuck.
3rd April	4.00 pm - 5.10 pm	Discussion on zoom meetings and getting feedback from each other and discussing class diagrams.
3rd April	1.00 pm - 1.30 pm	Evaluate work among us and help each other where anybody is stuck.
04th April	1.00 pm - 1.45 pm	Finalise document and make log entry, fix minor error
04th April	8.00 pm -8.30 pm	Document reviewed by TA and make changes as he suggests

