

Group No: 10

Testing Document



E-commerce website for community car-rental service

Table of Contents

<u>Introduction</u>	<u>4</u>
<u>Objectives & Tasks</u>	<u>4</u>
<u>Objectives</u>	<u>4</u>
<u>Tasks</u>	<u>4</u>
<u>Testing Strategy</u>	<u>4</u>
<u>Test Process</u>	<u>4</u>
<u>Requirements Understanding</u>	<u>4</u>
<u>Preparing Test cases And Test Matrix</u>	<u>4</u>
<u>Reviewing Test cases and Matrix</u>	<u>5</u>
<u>Executing Test Cases</u>	<u>5</u>
<u>Regression Testing and Retesting</u>	<u>5</u>
<u>Deployment</u>	<u>5</u>
<u>Testing Methods</u>	<u>5</u>
<u>TestingUnit</u>	<u>5</u>
<u>Integration Testing</u>	<u>6</u>
<u>System Testing</u>	<u>6</u>
<u>Regression Testing</u>	<u>6</u>
<u>GUI Testing</u>	<u>7</u>
<u>Acceptance Testing</u>	<u>7</u>
<u>Features to be tested</u>	<u>8</u>
<u>Sign-Up</u>	<u>8</u>
<u>Login</u>	<u>8</u>
<u>Rent A Car</u>	<u>8</u>
<u>Lend A Car</u>	<u>8</u>
<u>Quality Objectives</u>	<u>8</u>
<u>Primary Objectives</u>	<u>8</u>
<u>Secondary Objectives</u>	<u>9</u>
<u>Entry and exit criteria</u>	<u>9</u>
<u>Entry Criteria</u>	<u>9</u>

Exit Criteria	9
Resources and Environment needs	10
Testing Tools	10
Test Environment	10
Pass/Fail criteria	10
Suspension Criteria	10
Resumption Criteria	11
Approval Criteria	11
Dependencies, Assumptions and Risks	11
Dependencies	11
Assumptions	12
Risks	12
Limitations	12
Functional test review - findings	13
Testing (Backend)	13
Testing (Frontend)	17
Integration, System and Acceptance testing	19
Non-functional test review	20
Usability	20
Security	21
Meeting details	22
Conclusion	23

1. Introduction

With the cities expanding and endless roadlines, transportation is becoming an issue for the people who do not have their own personal transport and in this information age, where anything and everything is facilitated through online means, we plan to build a Online Car Rental Website. If someone needs a car for a short period of time, they can find one closest to their choice and range on here. Even if one's on a vacation in an unfamiliar place, this site will provide a trusted and convenient source car rental. And if someone has a car lying around that they don't use, they could profit off it by lending it. So this software will be for the users who would like to book cars as well as for users who'd like to lend cars, all within a few clicks. The navigation and the purpose of the website will be simple and accessible across multiple devices. The users will also be able to look up for the cars they need by company, price etc.

The purpose of this software test plan is give a brief idea about the scope and the objectives of the project, strategies to test the components, testing of different features, set of items, the types of testing used, requirements needed to test, roles and responsibilities of team members and risks associated with the plan.

2. Objectives & Tasks

a. Objectives

- To assess and analyse the working of different components used in the project
- To get an idea for the improvements needed for the components which failed the testing criteria
- After testing different components, to get a fair picture on the overall working of the project

b. Tasks

- Make a testing strategy
- Test all the components and the features using the decided strategy.

3. Testing Strategy

a. Test Process

i. Requirements Understanding

- After identifying the stakeholders, noting all the requirements from them.
- Assessing and analysing the requirements and how they will be included in the project.

ii. Preparing Test cases And Test Matrix

- Preparing test cases by exploring different scenarios and user stories.
- Making a test matrix which maps test cases to respective requirements of the project.
- The test matrix will make sure to cover all the requirements.

iii. Reviewing Test cases and Matrix

- Conducting peer reviews for test cases and matrices.
- Suggestions given by the reviewers will be implemented in further test cases and matrices.
- Thereafter review for the improved test cases and matrix will also be conducted.

iv. Executing Test Cases

- Test cases will be performed for various scenarios keeping in mind the requirements and the user stories.
- Test case status (Pass/Fail) of execution for different test cases will be updated in the document.

v. Regression Testing and Retesting

- Regression testing will be performed if there is some problem with the database, a new functionality is added to the existing build or when the website is not responding. Other than that retesting for the fixed bugs will be conducted and once it is resolved, it will be updated in the document.

vi. Deployment

- Report will be issued after the final testing is done with no further bugs.
- The application is deployed in the suitable environment and is further tested for its performance and functionality.
- After obtaining successful test results, it becomes suitable for new production.

4. Testing Methods

We are using Extreme Programming (XP) methodology in Agile development. In it, we do testing at every stage of development. We also make use of user stories to write acceptance tests that are done even before the product code is created. The acceptance criteria will help in deriving the definition of done i.e. if a user story passes the acceptance criteria, it is considered to be done. We start with unit tests using user stories.

a. Unit Testing

- This testing is the first level of testing where individual modules like programs, functions, procedures etc. are tested. It is concerned with the functional correctness of individual modules.
- It is conducted by performing white box testing method. It identifies the security holes, poorly structured paths in the coding processes, flow of specific inputs and outputs generated and testing of each unit on an individual basis.
- We do it manually for front-end and automated tests are created using jest for back-end.
- We will start with sign-up of the users and will move forward to the other features. In each feature, the functionalities are broken down into units and tested individually.
- Eg. In the login form, the function which validates the password should output an error message when the input password is correct.

b. Integration Testing

- This testing is the second level of testing where several units which are interconnected are tested to make sure that the transition from one unit to another is tested.
- It is done by using black box testing method and the integration testing is done

using a bottom-up approach.

- It is performed by other team members barring the members who developed the module.
- The interfaces and functionalities for the car browsing page and car details page were built separately and were then integrated to form a part of the rent a car feature.

c. System Testing

- This testing is the third level of testing where it validates the fully functional software product. End-to-end system specifications will be tested here.
- It is performed by other members barring the members who developed the software.
- The validity of the system is done by checking that it meets the pre-defined requirements mentioned during the development phase of the product.
- After integrating all the individual models like lending a car, renting a car, login/signup, the whole system is tested.

d. Regression Testing

- This testing is done keeping in mind that a change in the code or program has not propagated unintended consequences. Here, we re-execute the test cases to make sure the existing functionalities work fine.
- It is performed by the members of the development team and testers among them.
- Eg. When the payment for renting a car is not successful, the car which is already selected should not be discarded and the user need not go through the whole process.

e. GUI Testing

- This test is based on testing the graphical user interface where the users interact with the system.
- It mostly focuses on the design of the screen and its responsiveness and the user interaction with the elements on the screen.
- An individual who is not part of the development team conducts a test.
- Eg: Testing of pages, its transition, menus, pop-ups, buttons, icons, etc.

f. Acceptance Testing

- This test is based on the user stories. It checks whether the system is able to satisfy the acceptance criteria given for each user story.
- This is the last phase of testing so it is performed before the system is ready for the actual use.
- This testing primarily focuses on specific user requirements therefore it's an end-user testing.
- Eg: The user who has a spare car can profit by lending their car on this website, therefore the user should be able to successfully add a car and should be visible on the site so that other users can rent it.

5. Features to be tested

a. Sign-Up

- Create an account option for a new renter or lender.
- Enter details for the account.
- Verification of the account by receiving an otp on the registered email address.

b. Login

- Login option if the user already has an account.
- View user profile.
- Change password option if the user has forgotten the password.
- Edit details option.

c. Rent A Car

- Selecting date and location.
- Option for browsing the available cars.
- Filtering car options based on color, model etc.
- Book a car and view its details.
- View the terms and conditions and accept them.
- Option for viewing history.
- Option for updating user profile.

d. Lend A Car

- Lending a new car.
- Enter details of the car.
- Authentication of the details.
- Adding the car.
- Updating the details of the car.
- Option for viewing history.
- Option for updating user profile.

6. Quality Objectives

a. Primary Objectives

The primary objective for conducting testing is to ensure that all the requirements - functional as well as non functional have been met by the project. It also has to be checked whether the quality metric for each individual requirement has been satisfied or not. All the requirements and acceptance criteria mentioned by the user stories should be implemented. The ultimate aim at the end of the project development cycle is to ensure that the user identifies all of the requirements mentioned initially have been provided with much quality and are up to the expectations. Furthermore, any updates made to the requirements, functions or the design of the project will be appropriately documented and tested to provide the deliverables of highest quality within the available time frame.

b. Secondary Objectives

The secondary set of objectives will be to recognise any errors/bugs associated with the working of the current system and communicate about the same to the team members. After the system has been examined completely and the issues identified, a proper methodology should be employed to resolve the same in the given time frame. In all, a complete analysis of the system will be done followed by fixing the identified bugs.

7. Entry and Exit criteria

a. Entry Criteria

- Hardware and Software platforms required for testing must be installed. Testing environments must function properly.
- In order to allow the testers to test the system properly, requirements understanding documentation as well as design documents should be ready.
- Test data scenarios must be well known to the testing team.
- All the group members should have a good understanding of functionality and requirements.
- Proper review and proofreading of test cases and requirements should be done beforehand.

b. Exit Criteria

- To see whether rudimentary levels of requirements are satisfied.
- No bugs/defects marked as high or medium priority are left to be resolved.
- The testing plan has been followed.

8. Resources and Environment needs

a. Testing Tools

Process	Tools
Test Case Creation	Microsoft Excel
Test Case Tracking	Microsoft Excel
Test Case Execution	Jest(Backend) , Manual(Frontend)
Test Case Management	Microsoft Excel
Test Reporting	PDF

b. Test Environment

- Hardware - Computer
- Software - Javascript editor, Mongodb (for database), Google Chrome / Mozilla Firefox / Safari
- Language - Javascript, HTML, CSS
- Backend Framework - Mongodb
- Test Framework - Jest (For Backend)

9. Pass/Fail criteria

This criteria is to determine whether the execution of a particular test functionality passes or fails. If the guidelines are followed appropriately by the user and no mistake is done while executing the test functionality then the test is considered 'Pass' else it is considered 'Fail'. Normally, there are three situations while executing the test cases - normal, suspension, resumption.

a. Suspension Criteria

- These are the criteria to suspend all execution until a specific requirement of the functionality is met.
- It can occur in case of unavailability of external dependent systems during execution, when a defect is introduced which cannot allow further testing, significant change in business requirements suggested by the client or software/hardware problems.
- For example, the system will go to a suspended state when the user enters incorrect username , password or type.

b. Resumption Criteria

- These are the criteria to resume the suspended execution after the specific requirement of the functionality is met.
- It can occur when the dependencies become available, a solution is introduced for the caused defect , when the client agrees on extending the time for delivery or the issues related to software/hardware are resolved.
- For instance, the system will ask for a correct username, password or type when

the user enters an incorrect one.

c. Approval Criteria

- These are the criteria which when fulfilled will result in the approval of the test cases.
- When all the requirements of a particular test case is fulfilled, it is tested appropriately by the testers and the check for quality is done with a positive outcome then the approval criteria is said to be satisfied.
- For example, the system user has to enter correct login credentials which will then be approved by the system in order to access the respective functionalities.

10. Dependencies, Assumptions and Risks

a. Dependencies

When the tasks or the modules of a project are dependent on certain requirements then those requirements are called dependencies. Some of the dependencies are listed below :

- Database is required to store and fetch data for the functioning.
- Internet based servers are required for the working of the website.
- The system is dependent on Google Authentication for verification for signup functionality.

b. Assumptions

- Proper internet connectivity will be available with users in order to accomplish a request.
- Every user who rents a car is a licensed driver.
- Every car added to the system has a valid Car registration number.
- The images of the car provided by the lender justifies the original conditions of the car.
- The payment is conducted securely offline.

c. Risks

Risk identification and management are the main concerns for a software project. Risks are identified, classified and managed before the actual execution of the project.

Risk Type	Reason for Risk	Mitigating action
Schedule Risk	Wrong estimation of time because some resources may not be tracked properly.	Gather all the requirements and track all the resources beforehand
Programmatic Risk	Some flaw is generated in the database, an incorrect logic is written in code.	After writing code, an overview of it should be done by different team peers in order to recognise any fallacy if it exists.
Technical Risk	Advanced technology is not available or the integration of project modules is difficult.	Prepare alternatives for the existing needs; a system modular approach will help ease the process of integration.
Budget Risk	Cost overruns due to expansion of project scope.	A buffer should be kept for the cost in case the requirement changes.

11. Limitations

- Redundancy of passwords is not checked for whenever a new account is created.
- Payment is done offline. No online payment functionality is available.
- Security options for the car are not available. e.g. if the renter has an accident with the car then there is no precaution option like insurance.
- User's driving licence is not verified.
- Car registration number is not verified.
- Car verification based on the images provided is not done. e.g. a lender uploads an image of Creta while his car is Swift, there is no option for its verification.

12. Functional test review - findings

a. Testing (Backend)

Testcase_ID	Testing Type	Testcase Description	Test Step	Pre-Condition	Test Data	Post Condition	Expected Output	Test Status
01	Unit Testing	Successful addition of a new car	Enter car details like pictures, registration number, rent, rating, company, model, category, fuel type, no. of passengers, color, features, plan, city and lender's email id	Valid car details	<pre>pictures : ["test.png"], registration_no : "test1234", Rent:500, rating: 8, company: "test", modl: "testing car", category: ["test"], fuel_type: "diesel", no_of_passengers : 5, color : "black", features : ["Testing</pre>	User should be able to save the data	Successful addition of the car(details entered in the database)	Passed

					the car function"] , plan : 60, city: "ahmedab ad", lender_em ail: "test111@ com"			
02	Unit Testing	Showing details of car given car id	Car id will be sent to the database	Valid CarID	CarID:"C1"	User should see the car details	A page showing all the details of the car	Passed
03	Unit Testing	Showing available cars given location and date	Enter location and date	Valid location and valid date	date:{date: "2021-04-30"}, city:{city: "ahmedabad"}	User should see the available cars	A new page showing the available cars	Passed
04	Unit Testing	Update user profile(lender, renter)	Enter updated details	Valid details	City:{city: "Rajkot"}	User should be able to save the updated details	The page should show the updated user profile	Passed
05	Unit testing	User Sign-up	Enter email ,password ,name,phone. no,address,pincode,city	Valid Details	name: "sam", email: "sam22@ed.info",	User Should get OTP And enter	Successful Sign-up	Passed

					password: "sam1234" , phone_no: "1134567890", address : "atrangi complex,n avrangpur a" pincode:" 380009", city:"Ahm edabad"	Correc t OTP		
06	Unit testing	User Sign-up	Enter email ,password ,name,phone. no,address,pi ncode,city	Valid Detail s	name: "sam", email: "sam22@e d.info", password: "sam1234" , phone_no: "1134567890", address : "atrangi complex,n avrangpur a" pincode:" 380009",	User Should get OTP And enter incorr ect OTP	Unsuccess ful Sign-up	Passed

					city:"Ahmedabad"			
07	Unit testing	User Login	Enter a valid username & valid password	Valid Test data	Username :sam22@ed.info Password: sam1234	User should be able to see the home page	Successful Login	Passed
08	Unit testing	User Login	Enter a valid username & an invalid password	Valid Test data	Username :sam22@ed.info Password: samm1234	Error message "invalid username/password"	A pop up message box to show an Error "invalid username/password"	Passed
09	Unit testing	User Login	Enter an invalid username & valid password	Valid Test data	Username :sam222@ed.info Password: sam1234	Error message "invalid username/password"	A pop up message box to show an Error "invalid username/password"	Passed
10	Unit testing	User Login	Enter an invalid username & invalid password	Valid Test data	Username :sam222@ed.info Password: samm1234	Error message "invalid username/password"	A pop up message box to show an Error "invalid username/password"	Passed
11	Unit Testing	Token_auth	Test the auth token for logged in					Failed

			user, test refresh token and verification of token					
--	--	--	--	--	--	--	--	--

b. Testing (Frontend)

Testcase_ID	Testing Type	Testcase Description	Test Step	Pre-Condition	Test Data	Post Condition	Expected Output	Test Status
12	Unit Testing	Validate password constraint	Enters a password	Valid password	Password: a1	Error Message: Password should be of at least 8 characters and 1 number.	An error message on top of the pswd field saying "Password should be of at least 8 characters and 1 number"	Passed
13	Unit Testing	Validate rented and lended cars bifurcation	Cars info will be sent from database	User should be a renter/lender	Views the profile page for info of lended/rented cars	Cars should be bifurcated	The page should show rented/lended cars	Passed
14	Unit Testing	Validate user's email constraint during sign-up	User signs up	User is a new customer and gives correct email-id	name: "sam", email: "test22@ed.info",	User should be able to signup	User should be able to see the home page	Passed

					password: "sam1234", phone_no: "1134567890", address : "atrangi complex, navrangpura" pincode: "380009" , city:"Ahmedabad"			
15	Unit Testing	Validate user's email constraint during login	User logs in	User is an old customer and gives correct email-id	Username: test22@ed.info Password: sam1234	User should be able to login	User should be able to see the home page	Passed
16	Unit Testing	Validate non-unique email ID	User signs up with already an existing email id	Valid Email ID	name: "Tom", email: "test22@ed.info", password: "Tom1234",	User should not be able to sign up	User should not be able to sign up and the page should say "Emailid already registered"	Passed

					phone_no: "1134567890", address : "atrangi complex, navrangpura" pincode: "380005" , city:"Ahmedabad"		d.”	
17	GUI Testing	Validate only logged-in user access to 'Request A Car' button	Clicks on a 'Request Car' button	User should be logged in	Unlogged user clicks on the button	Should open login page	Opens login page	Passed
18	Unit Testing	Validate 'Accept' status confirmation	-	-				Failed
19	GUI Testing	Validate rental rate, no. of seats, fuel type visibility in car information	Clicks on a car catalogue box in 'Rent A Car' page	Shows information under the image	-	Not changeable	Information shown under the image and is non changeable	Passed

20	GUI Testing	Validate 'Lend A Car' button	Click on the 'Lend A Car' button	Enabled button	-	Clicked	Opens login/User profile page	Passed
21	GUI Testing	Validate 'Request Car' Button (T&Cs not checked)	Click on the 'Request Car' Button	Grayed out button	-	Grayed out button	Not-clickable	Passed
22	GUI Testing	Validate 'Request Car' Button (T&Cs checked)	Click on the 'Request Car' button	Enabled button	-	Clicked	Opens login/User profile page	Passed
23	Regression Testing	A new feature of user profile was integrated	User lends a car	User has entered all valid details	All details like registration number, model, color etc	User should be able to lend the car	The lent car should be seen on the website	Passed

c. Integration , System and Acceptance testing

Testcase_ID	Testing Type	Testcase Description	Test Step	Pre-Condition	Test Data	Post Condition	Expected Output	Test Status
01	Integration	Check	User	Enter all	Enter all	Should	A page	Passed

	on Testing	integrati on of modules	rents a car	valid details	valid details	be able to rent the preferred car	showing the car he/she rented. (The lender receives the request)	
02	Integrati on Testing	Check integrati on of modules	User lends a car	Enter all valid details	Enter all valid details	Should be able to lend the preferred car	A page showing the car he/she lended	Passed
03	System Testing	Check working of whole system	User rents a car	Enter all valid details	Enter all valid details	Should be able to rent the preferred car	User receives the car for use	Passed
04	System Testing	Check working of whole system	User lends a car	Enter all valid details	Enter all valid details	Should be able to lend the preferred car	User gives the car for use	Passed
05	Accepta nce Testing	Check fulfilmen t of user requirem ents						Passed

13. Non-functional test review

a. Usability

Number	Test case	Test Case description	Checklist/Status of the testcase
1	Check if the layout is consistent with the design criteria	Layout of the pages should be done to facilitate and display all the details corresponding to the particular functionality in a clear manner	✓
2	Check for the Fonts, Colors, Sizes, etc	Appropriate font size and colors should be set which makes the site easy to read and attractive	✓
3	Check for alignment	The details on a page should be aligned by grouping similar content	✓
4	Mandatory fields need to be highlighted with an asterisk symbol	The fields required for signing up for an account should be highlighted with an asterisk when left empty	✓

b. Security

Number	Test case	Test Case description	Checklist/Status of the testcase
1	Negative testing- Password is not visible	When a password is entered, it should be shown in the form of a bullet.	✓

2	Check if the password is saved in clear or encrypted?	To check the form in which the password is stored in the database.	✓
3	Verify the application with valid userId and invalid userIds	Check the userId field for correctness while logging in to the system.	✓
4	Verify the application with valid password and various invalid passwords	Check the password field for correctness while logging in to the system.	✓
5	Validate only logged-in user access to 'Request A Car' button	To check whether the user is logged in or not and only allow further if logged in	✓
6	Validate only logged-in user access to 'Lend A Car' button	To check whether the user is logged in or not and only allow further if logged in	✓

14. Meeting details

Date	Agenda	Description
15/04/21	Review different documents like requirement doc, design doc, etc.	The requirements gathered from different stakeholders, the user interface, etc. were reviewed

16/04/21	Test planning	Planning the approach that we would follow during test phase (automated testing using jest for back-end & manual testing for front-end)
17/04/21	Learn Tools	Learns tools like jest for testing
20/04/21	Develop test cases	Creating test cases for jest
21/04/21	Develop test cases	Creating test cases manually for front-end
22/04/21	Create test case matrix	Made a test case matrix which is mapped to different requirements
22/04/21	Review test cases and matrix	Check if the test cases cover all requirements
04/05/21	Prepare to test and run tests	Test data using different methods
05/05/21	Review test results	Check status pass/fail and make changes to the code accordingly
05/05/21	Regression Testing	To check previous builds still run well after making changes to it
06/05/21	System Testing	To test a complete and integrated software to check its compliance with the requirements
06/05/21	Final test run	To check the status of previously reviewed errors after correction

15. Conclusion

The document gives a detailed report of the objectives, tasks, schedule, strategy, resources used. Testing is very important as it measures the quality of the application. Therefore, a test plan is an important part of it as it serves as an outline to conduct different software testing activities monitored by the test manager. Because of its structure, people outside the test team such as customers, business managers, developers etc. can easily understand testing. It also helps us understand important aspects like estimation, scope and strategy are documented in the test plan. It is reviewed by the management team and can also be re-used in other projects. Also, testing methods, entry/exit criteria, resources and environment etc. are all covered in this.