IT5090 Web Capstone Project Assessment Part2

Student Id-20220897

Name-Anuradha Mangalpalli

Solution Design and Test Planning

Planning Document

Scope of the Project

Brief Introduction

Maharaja Kitchen is an Indian Takeaway located in the Porirua suburb, north of Wellington. It operates from a small shop in the commercial hub, close to the North city Mall and the supermarket.

It is owned by 3 friends, all second-generation immigrants from Western India. Pre-covid they had a restaurant in the Food Court in North City Mall. Like many businesses they too made losses during the Lockdowns and had to close shop.

Post covid they decided to run a takeaway with their savings, from a small shop in the adjacent commercial hub. It's been in operation since a year and half now.

They have 2 chefs; 2 kitchen helps and one admin staff whom they have retained from their earlier business. The takeaway has a small number of loyal customers.

Current business processes

- The business is run in an old-fashioned way.
- Everything is run manually.
- The customers walk in, place the orders at the counter, wait in the reception area until the order is ready, pay and leave.
- Sometimes, customers place orders on the phone, and can pay only on arrival as no online payment systems exist.
- At present customer base is limited to local area and slowly spreading through word of mouth or accidental walk-ins.

Stakeholder requirements

Overarching goal is to expand business and thereby profits. They have realized that, taking their business online could bring great benefits and help achieve their goal. So they commissioned me and my team, to create an attractive, robust, responsive web application where customers can easily perform the following functions: -

- Login easily
- Navigate through different pages with information on products smoothly and easily.
- Web application should be responsive to all devices like Mobile, Tablet, Laptop etc.
- Select and place orders and make payments online.
- Links to social media like Facebook, Instagram, Snapchat to reach out to wider population.
- If possible, a built-in calculator for accounts.
- Contact form for customers to provide feedback.
- Provide information on special deals and discounts.

Non-functional requirements

In addition to the above functional requirements, they had some non-functional requirements too. Non-functional requirements are related to the software system's desired quality in various dimensions such as availability, performance, security, reliability.

These requirements define the overall qualities of the software designed. They also consider the legal requirements, rules and regulations, policy and procedures and state explicitly all the qualities the stakeholders want in the product.

Some of the Non-functional requirements are listed below: -

Performance

Measures response times, speed with which the page loads when the user clicks on the website or individual elements like buttons. If it takes more than 2 seconds for leading, then it's categorized as poor performance.

Scalability

The product's ability to handle increasing amount of data or users.

Capacity

The ability to increase processing while providing acceptable response times.

Availability

The system should be available at any time. User cannot access a web browser, only by the server downtime.

Reliability

Consistency in performance in accordance with specifications

Recoverability

The ability to restore back to a point of failure.

Interoperability

Ability for systems and devices to share and interpret data.

Usability

Easy to interact with the product.

Data integrity

Maintenance and the assurance of data accuracy and consistency over its entire lifecycle. Precisely, ensuring data is recorded exactly as intended.

Security

Ability to protect data from unauthorized partiers. System uses security plugins so that the data is secure in all transactions, including the user confidential information.

Legal and regulatory

Conforming to the existing laws like Data protection Act, Licensing and copyright act, social media laws and project certification procedures.

Portability

Ability for the product to be installed in and uninstalled in different environments.

Maintainability

Ability to implement changes without causing unexpected failures.

Excluded requirements.

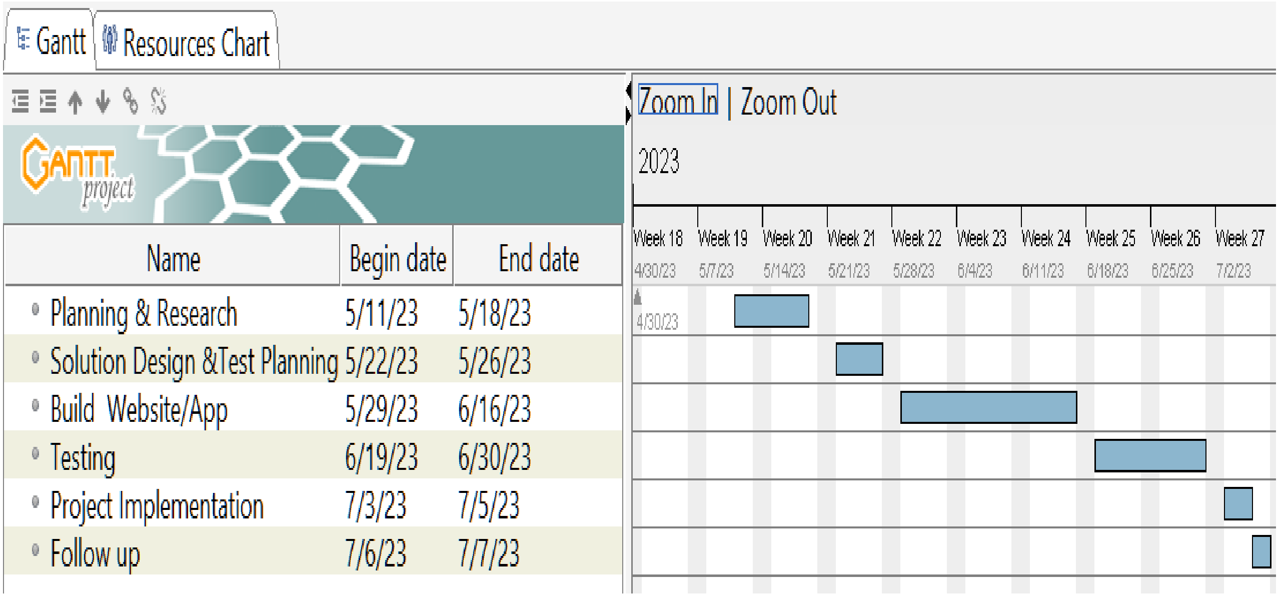
Calculator

Responsiveness’ to Smart TV and big screens

Time Schedule and Development steps

The sequence of tasks and timelines to execute them are given below in the Gantt chart.

Table 1



Name	Begin date	End date
Planning & Research	5/11/23	5/18/23
Solution Design &Test Planning	5/22/23	5/26/23
Build Website/App	5/29/23	6/16/23
Testing	6/19/23	6/30/23
Project Implementation	7/3/23	7/5/23
Follow up	7/6/23	7/7/23

Estimated project costs.

Buglet allocated for the project is $50,000.

The Tables below shows the budget layout:

Table 2

Product Budget				
Product Design Product Tasks				
Planning & Research	Labour Hours 40 hrs	Labour costs $1200	Travel costs $100	Total $1300
Solution design	48 hrs	$1,440	$120	$1560
Test planning	24 hrs	$720	$60	$780
		Total costs	$3,640	

Table3

Product Development				
Tasks	**Labour hours**	**Labour costs**	**Travel costs**	**Total**
Web app development	**150 hrs**	**$6000**	**$400**	**$6400**

Testing	40hrs	$1600	$100	$1700
Deployment	20hrs	$800	$60	$860
Follow up	20hrs	$800	$60	$860
Equipment costs				$20,000 $20,000
Domain hosting costs				$3000
SSL certificate (Security)cost				$1000
Miscellaneous costs				$10,000
Total costs $47,460				

Release scope

Success of a product also depends on small chunks of software built and tested based on user stories.

Agile software development has the customer or the end user at its core. The user stories are informal, general explanations of software features based on end user requirements.

In Agile projects, the project scope is formed of a "product backlog" of user stories, from which a small number of user stories are selected to form an individual release's scope (Canvas)

A survey of customers and target audience in Wellington revealed that: -

- Almost all of them wanted an easy to navigate website with all the information about the menu items and prices, deals & discounts.
- A particular feature everyone seemed to want was an easy login/log out option.
- Another important feature in demand was, the provision to order food items and be able to make payments, online.
- Information on the days and hours the takeaway services are available. For e.g., Monday to Friday/Weekends 5pm to 9pm
- Home delivery of food on order is welcome.

Chunks of software with these specific features will be designed, released, and tested in parts and feedback sought from the users and stakeholders.

- A minimum viable product in this case, buying food online can be released and tested for its usability and convenience.

System Design document

Feasibility

Deliverables

- It is well within their allocated budget to build a product to incorporate most of their requirements (listed in the requirements section).
- A multipage website can be built, with separate pages dedicated to menu items and their prices, contact information and information about their company itself.
- The website will be responsive to different devices like Mobile, Tablet and Laptop.
- All pages will have header with anchor tags, depicting multiple pages navigable from one page to another, with ease.
- Links/buttons will be provided for adding items to purchased/ordered to the cart.
- Online payment systems for ordered items will be built in.
- Features with links to various social media platforms like Facebook, Instagram, Snapchat, and Twitter are possible.
- As per their requirements, a responsive web app can be built for different devices but cannot extend this to Television or smart watches.

Non-deliverables

- However, it is made clear to them that, the product will need to be upgraded and modified from time to time and might need periodic training of their staff. The present budget will not cover these future costs.
- Also given the limited capacity for data storage, only a few important customer details can be recorded and maintained.
- As per their requirements, a responsive web app can be built for different devices but cannot extend this to Television or smart watches.
- Building a calculator and integrating into the software may not be possible in the timeline. Can be considered towards the end and might cost a bit more money and time.
-

Technology Feasibility

There are several CMS (content management systems) like WordPress, Joomla, Drupal, Magento, PrestaShop, Concrete5 one could build the software, but since they come with their own problems like security and cost issues, it was decided to build the software in-house.

The technologies and frameworks in brief (already discussed in Planning Part 1 Document) are as follows:

HTML5, CSS3, Advanced JavaScript, Bootstrap, Media queries and React.js., Visual Studio Code & PowerShell. For testing Jest and Python automation tools.

Solution Design

According to the requirements analysis report solution design should have.

Navbar - where we can have the required needs as written on the analysis report.

There will be the restaurant logo, different anchor tags like Home, Menu, contact us and about us displayed on it.

Footer Where Help, FAQs and Terms & Policies and Social media links are given**.**

1. **Home Page** –

   Home page will contain some menu items, deals and discounts.

   It will also contain a search bar so person can search for information.

2. **Starters page**

   The page will display different menu items(starters) and their prices. Also, an add to cart button below the menu items. On clicking the items, adding to cart, and completing the order, the customer will be directed to payment page.

3. **Main menu page**

   The page will display different menu items (Main dishes) and their prices. Also, an add to cart button below the menu items. On clicking the items, adding to cart, and completing the order, the customer will be directed to payment page.

4. **Your order**

   Payment methods and details will be displayed.

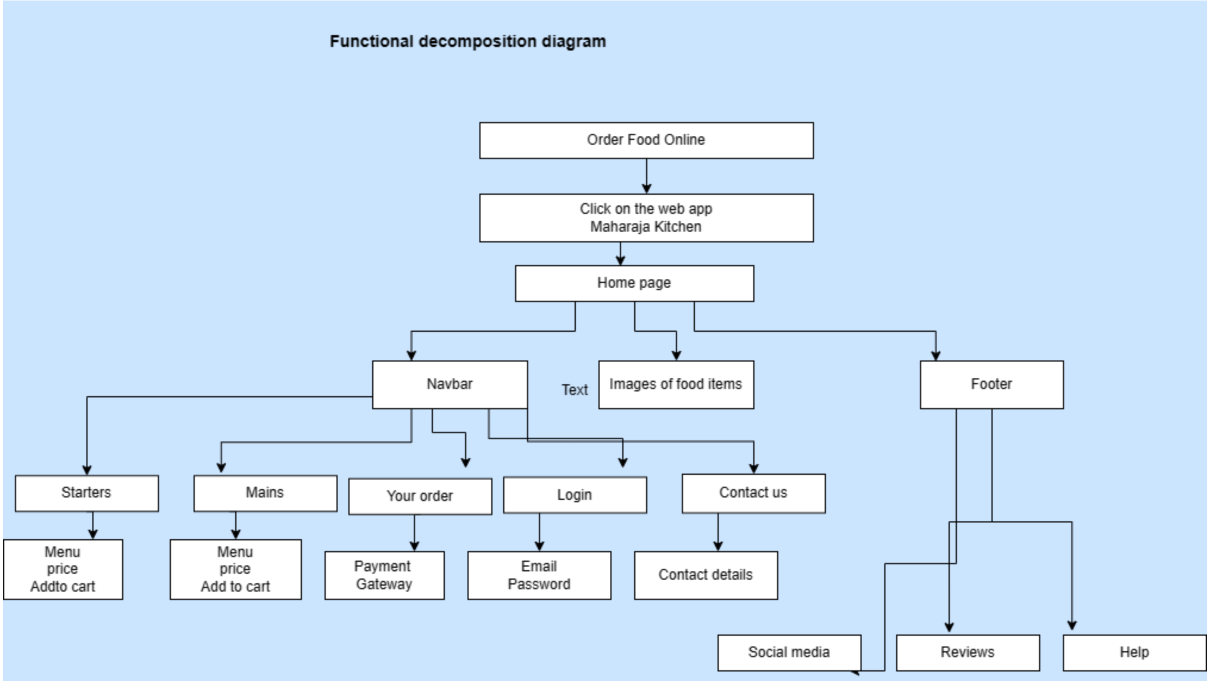
5. **Login page**

   Customer will be able to Login.

6. **Contact page**

   It will have the contact details with email, Phone number and location address.

Functional Decomposition Diagram is given below:

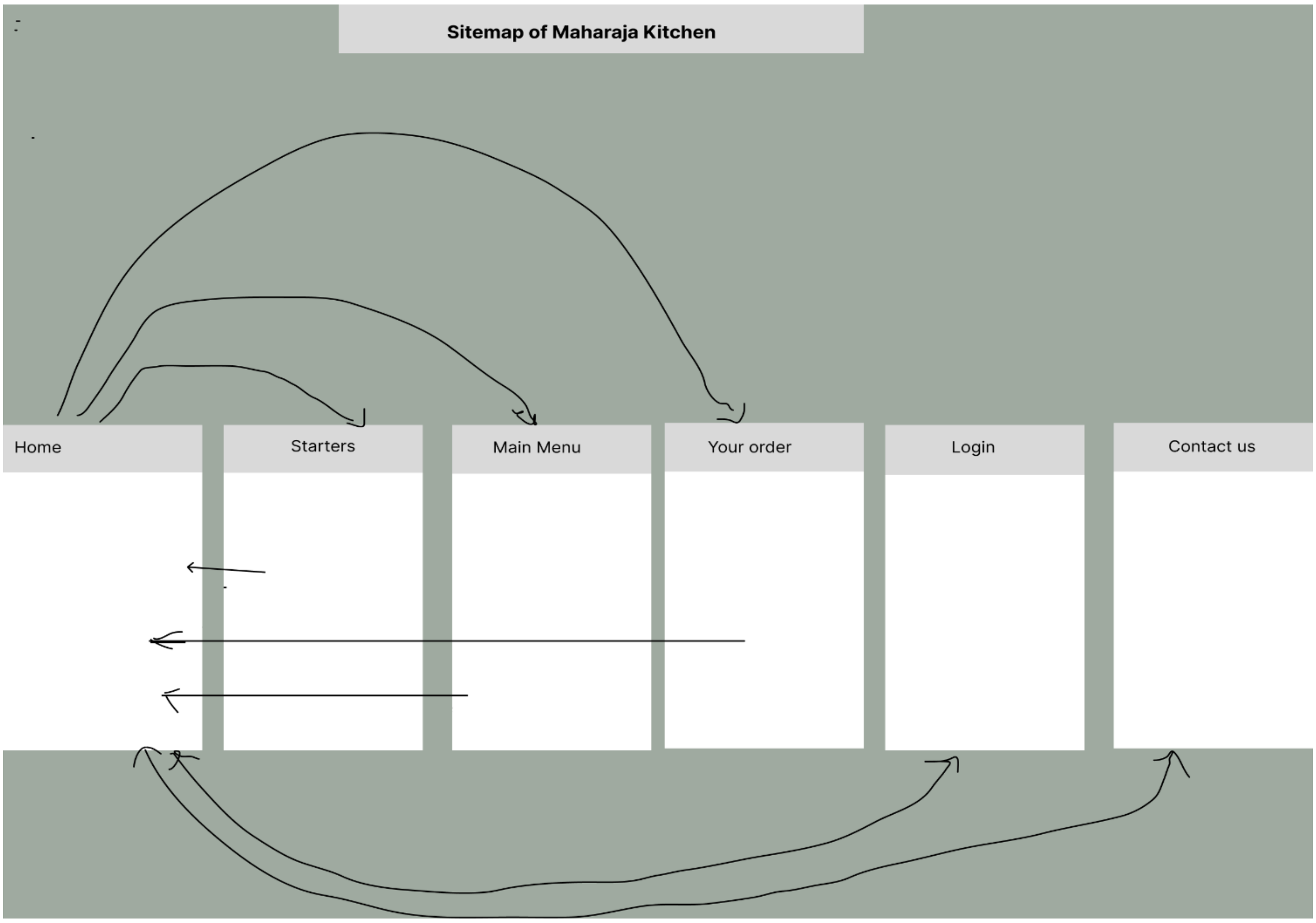


Non-functional Design requirements

Product will incorporate most of the important non-functional requirements desired by the clients listed in the above sections plus specifically the following:

- A backup of the product
- Security through applying for SSL certificate.
- Reusable and scalable components
- Enhanced usability through well designed and placed UI elements.

Sitemap of Maharaja Kitchen



User Documentation

User documentation refers to the documentation for a product or service provided to the end users. It is designed to assist end users to use the product or service. This is often referred to as user assistance and is a part of overall product delivered to the client/customer (Wikipedia).

According to Dartmouth Computer science department (Wiki) a user document should include the following:

- Minimum hardware and software requirements

- Installation guide
- How to start the system
- How to use different features of the system
- Screenshots explaining main features of the system.
- Example inputs and outputs
- Explanations of error messages and troubleshooting guides
- Information to contact the developer of the system if an undocumented question arises.

It provides step by step instructions as to how to use your product. Also it brings down the customer support costs in terms of money and time.

Our client Maharaja Kitchen will be given a document detailing all the above-mentioned steps to be able to use the product. This will be useful to all the stakeholders, employees involved to understand the product and its functioning. The document will also provide information on troubleshooting and information on where to seek help.

User Document for End-Users

Documentation for end-users is crucial as it offers many benefits to both end-users and business. Some of them are as follows:

Reduced support costs

It helps users find answers to their questions online, instead of making calls to customer care or writing emails. The management too will save time and money in not having to have personnel for answering the calls and replying to the emails.

Also, users will be able to learn how to use a product without assistance and avoid mistakes.

Customer satisfaction

User experience with the product is central to the customer satisfaction. It increases dramatically if the user interfaces are well designed and easy to use.

Provide edge over competition.

Can help customers choose or prefer products over the ones that do not have documentation. Like for instance, my client has to compete with the other existing takeaways in the town. If we make it easy for our customers to use our website, they might ours over others which are not so easy.

Important End-User documentation features

- Must be written in simple language, clear and concise.
- Must be easy to follow flows logically.
- Step by step instructions must be provided for using the product.
- Up-to-date information must be provided. Most of the times websites are not updated and it leaves the customers frustrated and upset.

User-document evaluations

- User_feedback-collected through surveys, interviews, and user-testing
- Documentation views and Time spent on the page-recorded through user-documentation KPIs, provide valuable insights.
- Bounce rate-Shows percentage of users leaving document after one page indicating it’s not serving their purpose exactly.

All the above-mentioned features and requirements will be kept in mind while creating a En-User documentation for my client. They will be provided with steps to use the product and links to look for when experiencing problems.

Prototype End-User Document for Maharaja Kitchen

Start-

1. Click on the website. It will take you to the home page. -

- Home page will display links on the top, to different pages like starters - menu page, Mains-Menu, Sign-up page/Login, Check-out and payment page and Contact Us page.
- Home page will have some images of different food items you can choose and order.
- Links to social media and Help and Reviews will be provided at the bottom of the page.
- Reviews will clarify the quality of our products and services.
- Help will be able to assist you with problems using the app.

2. Clicking on starters menu- you will be directed to page with the details about the starters you can choose and order. Every item will have price and add to cart button displayed underneath it.

3. Clicking on Mains menu- you will be directed to page with the details about the Main dishes you can choose and order. Every item will have price and add to cart button displayed underneath it.

4. Clicking on add to cart button -will lead to adding all items you wish to order

5. Clicking on Login page- you will be directed to page where you can Login to the system.

6 Clicking on Check-out and payment page page- you will be directed to page where you can make your payments.

7. Clicking on Contact us page- you will be directed to page where you can find information about our contact numbers, Email address and location address.

Note- please leave your review of the products and services, so that we can improve and serve you best.

Test Planning

Introduction

Testing is an important part of software development life cycle. Tarlinder, A. (2016) defines Testing as "An activity performed to ensure correctness and quality of software". Bach & Bolton define it as "Testing is the process of evaluating a product by learning about it through exploration and experimentation, which includes to some degree: questioning, study, modelling, observation, inference, etc".

It contributes to the improvement of the quality of the software which in turn consists of factors like functionality, reliability, efficiency, maintainability, and portability.

Developers sometimes are blind to the bugs in their code and unable to detect problems. So it is recommended to have independent testers or a testing team to detect the defects.

Two fundamental approaches to testing are the critiquing (the waterfall method) and supporting (agile testing method).

The Waterfall method or traditional method-This is testing the software once the development has finished and requires evaluation. It checks to see if the software meets the specification, and all functionality has been included.

Agile testing -tests are run throughout the development of the software product. After a developer writes some code, they can then test it to make sure that the code behaves as expected.

Advantages are that defects are reported soon after the code has been written unlike the former where mistakes found at the end of the development might prove costly in terms time, effort, and costs.

Levels of testing

Software systems are tested at various levels of software development lifecycle, in the order depicted below:

Unit test

Involves testing of individual units of source code-sets of one or more program modules together with associated control data, usage, procedures, and operating procedures, to determine whether they are fit for use.

Integration testing

Involves testing in which different units, modules or components of a software application are tested as a combined entity.

System test

Involves testing and validation of fully integrated software i.e., evaluating the system's compliance with its specified requirements.

Acceptance test

It includes 4 types of testing which are as follows:

- User acceptance testing (UAT)
- Operational acceptance testing (OAT)
- Contractual and regulatory acceptance testing
- Alpha and Beta testing

User acceptance testing refers to the end user software testing prior to the release of the product. It is done to ensure that the product delivers what it intends to do and meets all the business requirements.

Fundamental test activities

The process of Testing involves 5 fundamental activities. Schaefer et al (2014) (Canvas) identified them as follows:

1. Planning and Control
2. Analysis and Design
3. Implementation and Execution
4. Evaluation of Exit Criteria and Reporting
5. Test closure activities

Given the significant role Testing plays in SDLC, we decided to create a detailed Test plan for our product as part of our Solution Design planning phase.

Test plan Document

It will be based on guidelines set by IEEE 829, for documenting software testing. These guidelines provide a framework for creating many different types of documents like test plan, specifications of test design, test cases, test procedures, test logs and test incident reports. Adherence to these guidelines ensures that all relevant information is recorded during testing, enabling a quality software product.

Introduction:

This testing plan is for the Web application “Maharaja Kitchen (Indian Takeaway)”, version 1.0.

Objectives: -

Ensure that the web application meets the business requirements.

Eliminate defects if any.

Test Items:

- Web Application: ‘Maharaja Kitchen ‘web app, version 1.0
- Build Number: 101

Features to be tested:

- ▪ Navigation from page-to-page functionality
- ▪ User login/Signup
- ▪ Add to cart and checkout functionality.
- ▪ Payment gateway incorporated.
- ▪ Quality of images (observation)

Test approach:

- Manual testing
- Automated testing e.g., Jest for browser testing
- Combination of both if necessary
- White-box testing

Note- Usually unit and integration tests are automated and are created using API - bases tools. At the system and acceptance levels, the automated tests are created using GUI-based tools.

Test Environment:

- Operating System: Windows 10, MacOS
- Browser: Google Chrome, Firefox, Safari
- Hardware: Intel i5 processor, 8GB RAM
- Server: AWS
- Visual studio code, PowerShell

Test Schedule:

- Test Planning: from May 19, 2023, to May 26, 2023
- Test Case Development: May 27, 2023, to June 16, 2023.
- Test Execution: Right through development -from May 27, 2023, onwards until completion.
- Test Closure: June 30, 2023

Test Deliverables:

- Test cases
- Test scripts
- Test reports
- Defect reports
- Performance test report
- Responsiveness test report
- Cross-browser test report

Test Responsibilities:

- Test engineer: independent tester (will be responsible for developing test cases and scripts and executing test)s.
- Developer: Unit tests will be done by developer too
- Test Lead: Responsible for overall test planning and execution (Developer and test engineer).

- Server administrator: Responsible for maintaining the test environment.

Test Approach:

- Manual testing will be used to test all the functionalities of the web application.
- Automated testing will be used to test the performance and load of the web application.
- Responsive testing will be done to ensure the web application is compatible with different devices and screen sizes.
- Cross-browser testing to see its performance on different browsers.

Exit Criteria:

- All the defects identified and corrected and verified.
- All the test cases are executed and passed.
- Sufficient coverage of required functionalities.
- All the test deliverables completed and submitted.
- Performance test should pass the threshold limit.

Incident report

Incident reporting too, will be based on IEEE guidelines.

1.Incident identification: Basic information of an incident such as the date and time of its occurrence, the location of the defect and the severity of the problem will be recorded.
2. Incident description: Detailed description of the incident, as to what it is and how it happened and what the expected behaviour was, will be recorded.

3. Incident impact: Is all about the effect the incidence has on the functionality of the product and its overall performance.

4. Incident priority: The urgency of the issue is noted and the order in which it should be addressed will be prioritized.

5.Incident resolution: Involves detailing of steps needed to be taken to resolve the incident.

6. Incident closure: Recording of details as to how the incidence was closed, whether it is fully resolved and necessary tracking activities if any.

7. Incident documentation: Complete recording of test cases, test logs and screen shots (capturing all relevant information related to the incident).

Summary of test report

Test Summary report identity will be MK-101.

Summary

The objective of the testing process is to verify that the software developed met all the business requirements. The scope of the test plan was to test the' Maharaja Kitchen' web app, for its usability, compatibility with various operating systems, security, performance, and functionality.

Variances

If any variances like slow loading on the web browser is encountered, will be reported under variances.

Comprehensive assessment

_A complete picture of all defects, small and big affecting the functionality of the product will be recorded. A low reliability and poor performance signal that the product is not ready for deployment.

Summary of test results

A summary of total number of tests conducted, tests passed, and tests failed, and their severity will be recorded.

Evaluation

Implications of test results and the remedial steps will be detailed here.

Summary of activities

All the testing activities like usability testing, performance testing, functionality testing, security testing and compatibility testing will be mentioned here. Challenges encountered during the testing process and tools and technologies involved in testing will be summarized here.

Approvals

The report will be sent for approvals to all stakeholders. Their names and titles and date of approval will be recorded.

Test case Samples

IT 5001 Assignment

Web application visited - Movie Information | Tickets | Showtimes | Reading Cinemas

Test Case

Id-101

Date-12/05/2023

Name of the Tester-Anuradha Mangalpalli

Functionality Test Case

Test Case-A simple test for Login procedure

Test condition-If a valid username is entered and correct password given, the user is logged into the system.

Precondition-User is not already logged in. In my case I had already set up an account.

Input data-Valid username and correct password. When given a correct password it enabled login immediately. When the incorrect password was fed,' *Email, password do not match '*message popped up.

Expected results-Able to log in successfully.

Post condition-Was logged into the system until the job got done.

Test result-passed

Test Case

Id-102

Date-12/05/2023

Name of the Tester-Anuradha Mangalpalli

Functionality Test case -purchase movie ticket online

Test Steps: -

1. Click on the movie website.
2. Choose a movie and the show timings and click on it.
3. Navigates you to the page displaying the seats available.
4. Click on the number of seats desired.
5. Displays the ticket price and button saying Next.
6. Clicking on the Next button leads you to the Café page.
7. If you choose to skip buying eatables/drinks leads to Login and pay page.
8. Clicking Login and pay leads to the next page displaying Card section to be filled with credit card details.
9. Once entered the credit/debit card details and clicking on the button accepting their terms and conditions the transaction is completed and tickets issued.

Input Data in a nutshell -To choose seats, credit card details and accepting terms and conditions.

Expected output- Tickets issued on payment.

Post condition-Tickets in hand

Test results-passed (User succeeded in purchasing tickets)

Test Case

Id-103

Date-12/05/2023

Name of the Tester-Anuradha Mangalpalli

Performance Test Case

Input- clicks on the relevant buttons, giving login details, credit/debit card details for payment.

Output-lead to relevant page and desired outcomes.

Speed-Pages load efficiently and quickly. Time taken for the whole procedure-approximately 2 minutes.

Scalability-It was easy to add or delete the number of tickets one wanted. You could navigate through pages easily.

Stability-The app is robust and gives the same result over and over through iterations.

Test results-passed

Test Case

Id-104

Date-12/05/2023

Name of the Tester-Anuradha Mangalpalli

Security test Case

The above movie app seemed to have secured their website sufficiently.

Input -Verification code button had to be ticked to prove that I am not a robot.

Output-Transaction takes place immediately after proving that I am not a robot.

Test results-passed

Test Case

Id-105

Date-12/05/2023

Name of the Tester-Anuradha Mangalpalli

Integration test Case

Different component functions seemed to be integrated quite efficiently. For instance, the Café page options and if user decides to purchase items and add to ticket prices .it seems to do it effortlessly.

Test results-passed

Test Case

Id-106

Date-12/05/2023

Name of the Tester-Anuradha Mangalpalli

User Interface test

Here the test case was to check the overall functioning of the app. To check if different aspects of the application are performing correctly and user friendly.

The layout design could have been better with more appealing background colors. Also, including a trailer or two would have been very attractive for customers.

The layout on different pages was easy enough to get the information on movies, Café, availability of the seats and purchasing of tickets.

Test results-passed

Test Case

Id-107

Date-12/05/2023

Name of the Tester-Anuradha Mangalpalli

Test Case for Responsiveness

Test condition-Should be able to resize and display appropriately on different devices.

Input-open the application on different devices and view

Output-The website was responsive and opened and displayed efficiently on different devices.

Post condition-Stayed in the mode if the viewer wanted to view it.

Test results-passed

Negative tests

Invalid inputs give erroneous results.

Example- feeding negative numbers as income for tax calculation.

Salary or income cannot be negative