STE Manual

Comuter engineering (Bharati Vidyapeeth's College of Engineering)

Vidyalankar Polytechnic

DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	01
Title of Experiment	Design test cases for purchase order management based on system specification.

Test case id	Test case name	Actual input	Expected output	Actual output	Status
TC_01	Username	Enter Username = " abcde123"	It should accept the username	It accepted the username	pass
TC_02	Username	Enter Username = "abcde"	It shouldn’t accept the username	It didn’t accept the username	pass
TC_03	Username	Enter Username = "abcdef1234567 89 "	It should accept the username	It accepted the username	pass
TC_04	Username	Enter Username = "abcdefg12345678 9"	It shouldn’t accept the username	It didn’t accept the username	pass
TC_05	Username	Enter Username = " abcde@123"	It should accept the username	It accepted the username	pass
TC_06	Username	Enter username ="abcde123"	It should accept the username	It accepted the username	pass
TC_07	Username	Enter username ="123abcde"	It shouldn’t accept the username	It didn’t accept the username	pass
TC_08	Username	Enter username ="abcde 123"	It shouldn’t accept the username	It didn’t accept the username	pass

TC_09	Password	Enter password ="accept12	It should accept the password	It accepted the password	pass
TC_10	Password	Enter password = "accept"	It shouldn’t accept the password	It didn’t accept the password	pass
TC_11	Password	Enter password="accept @1"	It should accept the password	It accepted the password	pass
TC_12	Password	Enter password="copy paste"	It shouldn’t accept the password	It didn’t accept the password	pass
TC_13	Password	Enter password="acc1 2"	It shouldn’t accept the password	It didn’t accept the password	pass
TC_14	Submit	1) Username="ab cde 123". Password="ac cept 123". Click on Submit button	It should proceed to next page	It proceeded to next page	pass
TC_15	Submit	Username="abcd e123". Password="acc ept ". Click on Submit button.	It shouldn’t proceed to next page	It didn’t proceed to next page	pass
TC_16	Submit	Username="ab cde ". Password="afv k uh". Click on Submit button.	It shouldn’t proceed to next page	It didn’t proceed to next page	pass
TC_17	Submit	Username= " nvgg". Password="hvs gu gs". Click on Submit button.	It shouldn’t proceed to next page	It didn’t proceed to next page	pass
TC_18	Clear	Username="ab cde123". Password="acc ept123". Click on Clear button.	It should clear the username and password	It cleared username and password	pass
TC_19	Clear	Username="". Password="". Click on Clear button.	It should clear the username and password	It cleared username and password	pass

2. Test cases for purchase order system.

Test case ID	Test case Name	Actual input	Expected result	Actual Output	Status
TC_01	Name of product	Name of product = "Camera", Click on search button.	It should display the availability of the product.	It displays the availability of the product.	pass
TC_02	Name of product	Name of product = "Bikes", Click on search button.	It should display the availability of the product.	It displays the availability of the product.	pass
TC_03	Name of product	Name of product = "Toolkit", Click on search button.	It should display the unavailability of the product.	It displays the unavailability of the product.	pass
TC_04	Product (Drop Down)	Product=Enter Initial letter of "C"	It should display all product starting with letter "C"	It displays all product with initial letter "C"	pass
TC_05	Product (Drop Down)	Product=Enter Initial letter of "A"	It should display all product starting with letter "A"	It displays all product with initial letter "A"	pass
TC_06	Product (Drop Down)	Product = Enter any number	It should prompt for invalid message.	It prompts an invalid message	pass
TC_07	Quantity	Enter a number	Itshouldcheckthe availability, and inform user	It checks the availability and inform user.	pass
TC_08	Quantity	Enter an alphabet	It should prompt an invalid message	It prompts an invalid message	pass
TC_09	addthe product	1) Select product. 2)Click on add product.	It should add the product	Itisaddingtheproduct	pass
TC_10	add the product	1) Click on add product.	It should prompt the message of “First select product"	It is promptingthe messageof“First select product"	pass

TC_11	remove the product	1) Select the product from the purchase list click on remove product button.	It should remove the product	It is removingthe product	pass
TC_12	remove the product	1) Click on remove product button.	It should prompt the message of “First select product"	It is prompting the message	pass
TC_13	review product	1) Select the product.2) Click on review product button.	It should review the product	It is reviewing the product	pass
TC_14	review product	1)click on review product button	It should prompt the message of “First select product"	It is prompting the message	pass
TC_15	Submit	1)Select the product. 2)Enter the quantity. 3) Add the product. 4) Click on submit button.	It should generate order id	It is generating the order id	pass
TC_16	Submit	1)click on submit button	It should prompt the message of “Fill the details".	it is prompting the message	pass

Practical Related Question

1) What are the objectives of software testing?

Answer:

The objectives of software testing are as follows:

1. Finding Errors
   - Testing is process of executing a program with an intention of finding an error.
2. Creating good test cases
   - A good test case is one that has a high probability of finding yet undiscovered error.
3. Quality Improvement
   - Defects are fixed by developer, so quality is improved
4. Satisfying customer requirements:
   - Testing demonstrates customer that software works properly as per specification.

2) How can we design test cases from requirements?

Anwer: For designing test cases, a tester who is performing test cases needs a software requirement specification to design test cases.

3) Static vs dynamic testing

Static Testing	Dynamic Testing
Testing is done without executing the program	Testing is done by executing the program
This testing does verification process	Dynamic testing does validation process
Static testing gives assessment of code and documentation	Dynamic testing finds bugs in the software application
Static testing involves checklist and process to be followed.	Dynamic testing involves test cases for execution.
It is performed in the early stage of the software development.	It is performed at the later stage of the software development.
Cost of fixing bug is low	Cost of fixing is high
Example : Inspection, walkthrough	Example : Boundary value analysis ,Equivalence Partitioning

Exercise:

1.Identify the situation when to start and stop software testing.

When to start ?

1. Testing starts from requirement phase and continues till the end of SDLC(software development lifecycle)

2. SDLC testing can be started from requirement gathering phase and last till the development of the software.

3. however it also depends on the development model that is being used for ex. in waterfall model testing is conducted in testing is performance at the end of every increment

4. testing is done in difficult form at every phase of SDLC like during requirement gathering phase, the analysis and verification of requirement are also considered as testing.

When to stop?

1. Unlike when to start testing , it is difficult to determine when to stop. testing is never ending process, and no one can say that any software is 100% tested.
following other aspect which should be considered to stop the testing

A. Testing deadline

B. Completion of test case execution , all requirement coverage or code

Coverage has been achieved

C. Bug rate falls below a certain level and no high priority bugs are found

D. Management Decision

3.In white box testing identify the parameters to verify

Ans. In White box testing the parameters to verify are:

a. statement/line coverage

b. branch coverage

c. condition coverage

d. function coverage.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	02
Title of Experiment	Design test cases for purchase order management based on system specification.

Q. Design Test Case for Inventory Management System based on System Specification.

Test Case ID	Test Case Name	Actual Input	Expected Output	Actual Output	Status
TC_01	Mobile Number	Mobile Number = "abcdefghij"	It should not accept the data.	It is not accepting the data.	pass
TC_02	Mobile Number	Mobile Number ="1234567890"	It should accept thedata.	It is accepting the data.	pass
TC_03	Email Address	Email Address = "1_abcd@example.com"	It should not accept the data.	It is not accepting the data.	pass
TC_04	Email Address	Email Address ="abcd_1@example.com"	It should accept thedata.	It is accepting the data.	pass

TC_05	Quantity	Quantity = "ab"	It should not accept the data.	It is not accepting the data.	pass
TC_06	Quantity	Quantity = "05'"	It should accept the data.	It is accepting the data.	pass
TC_07	Email Address	Email Address ="abc@pqrs"	It should not accept the data.	It is not accepting the data.	pass
TC_08	Mobile Number	Mobile Number = "123456780"	It should not accept the data.	It is not accepting the data.	pass
TC_09	Quantity	Quantity = "123"	It should not accept the data.	It is not accepting the data.	pass
TC_10	Quantity	Quantity = "0"	It should not accept the data.	It is not accepting the data.	pass

Practical Related Question

1) What are major system specifications of Inventory Management System?

Answer: The major inventory management system specifications involve providing a way to store, organize, manage and analyze inventory data. Systems requirements include:

- An easy-to-use interface that doesn't require advanced training, support or documentation.
- Automation for eliminating manual processes of business functions related to inventorxy management.
- A reliable, secure database that provides accurate, real-time data.
- Performance that enables fast, actionable inventory monitoring and control.
- The ability for administrators to easily add software modules with minimal configuration so that the system is scalable.
- Software integrations and automated features that minimize manual inventory updates or inputs.

2) What are the functions of Inventory Management System?

Answer: Following are the functions of inventory management system:

Answer:

- Improved Productivity and Efficiency
- Avoid Stock-outs and Over-stock
- Quality Management
- Improved Profitability
- Forecasting and planning
- Balanced Supply and Demand
- Inventory Reports
- Inventory Tracking

3) Give the significance of Inventory Management System as per Business perspective.

Answer:

- Inventory management can make or break a business. Inventory is often the largest item in the current assets category on a balance sheet. Issues with inventory can contribute to business losses, even failures.

- Proper management of the supply chain, on the other hand, can allow a business to thrive.
- Good inventory management strikes a balance between the amount of inventory coming in and going out. It controls the timing and costs of non-capitalized assets and stock items, allowing a business to reach optimal profitability

Vidyalankar Polytechnic

DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	03
Title of Experiment	Design test cases for a Calculator to verify its function.

Q. Design test cases for a Calculator to verify its function.

Test Case ID	Test Case Name	Actual Input	Expected Output	Actual Output	Status
TC_01	Number Input	"123467891234567 "	Take input Successfully	Takes input Successfully	Pass
TC_02	Number Input	"123467891234567891234"	Do not take input	Does not take input	Pass
TC_03	Arithmetic operation (Addition)	Add " 50 + 50 "	" = 100 "	" = 100 "	Pass
TC_04	Arithmetic operation (Addition)	Add " 0 + 0 "	" = 0 "	" = 0 "	Pass
TC_05	Arithmetic operation (Addition)	Add " 1234567891234567 + 1234567891234567 "	" = 1,23,45,67,89,12 ,34,567 "	" = 1,23,45,67,89,12 ,34,567 "	Pass
TC_06	Arithmetic operation (Subtraction)	Sub " 50 - 50 "	" = 0 "	" = 0 "	Pass
TC_07	Arithmetic operation(Multiplication)	Mul " 50 * 50 "	" = 2500 "	" = 2500 "	Pass

TC_08	Arithmetic operation(Multiplication)	Mul " 50 * 0"	" = 0 "	" = 0 "	Pass
TC_09	Arithmetic operation (Division)	Div " 50 / 50 "	"1"	" 1 "	Pass
TC_10	Arithmetic operation (Division)	Div " 0 /5 0 "	"0"	"0"	Pass
TC_11	Arithmetic operation (Division)	Div " 50 / 0 "	Display "Cannot Divide By Zero"	Displays "Cannot Divide By Zero"	Pass
TC_12	Scientific operation (Sin)	" sin 90 "	" 1 "	" 1 "	Pass
TC_13	Scientific operation (Cos)	" cos 0 "	" 1 "	" 1 "	Pass
TC_14	Scientific operation (Tan)	" tan 45 "	" 1 "	" 1 "	Pass
TC_15	Scientific operation (π)	" π "	" 3.14159265359 "	" 3.14159265359 "	Pass
TC_16	Negative and Positive	" 2 + 2 + (-2) +(-2) "	" 0 "	" 0 "	Pass
TC_17	Negative and Positive	" 2 - 2 - (-2) "	" 2 "	" 2 "	Pass
TC_18	Negative and Negative	" - 2 - 2 "	" -4 "	" -4 "	Pass
TC_19	Negative and Positive	" 2 - 2 - (-2) "	" 2 "	" 2 "	Pass
TC_20	Logarithm	" Log2 8 "	" 3 "	" 3 "	Pass
TC_21	Trigonometry	" sin 0 + cos 90 "	" 1 "	" 1 "	Pass
TC_22	Trigonometry	" sin 90 + cos 0"	" 1 "	" 1 "	Pass
TC_23	Trigonometry	" sin 0 + tan 0 "	" 0 "	" 0 "	Pass
TC_25	Trigonometry	" cos 0 + tan 0 "	" 1 "	" 1 "	Pass

Practical Related Questions

1. State key factors to be tested in black box testing

Answer: The key features to be tested in Black Box Testing are as follows:

- Requirements
- Boundary Value Analysis
- User Documentation
- Decision table
- Equivalence Partitioning.

2. What are the sources of knowledge for Black box testing?

Answer: The sources of knowledge for Black Box Testing are as follows:

- User Requirements
- System Requirement Specification
- Design Documentation

3. State advantages and disadvantages of Black Box testing.

Answer: Advantages of Black Box Testing:

- ➢ Well suited and efficient for large code segments.
- ➢ Code access is not required.
- ➢ Clearly separates user's perspective from the developer's perspective through visibly defined roles
- ➢ Large numbers of moderately skilled testers can test the application with no knowledge of implementation, programming language, or operating systems.

Disadvantages of Black Box Testing:

- Limited coverage, since only a selected number of test scenarios is actually performed.
- Inefficient testing, due to the fact that the tester only has limited knowledge about an application.
- Blind coverage, since the tester cannot target specific code segments or errorprone areas
- Blind coverage, since the tester cannot target specific code segments or error prone areas

Vidyalankar Polytechnic

DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing (STE)	Subject Code: 22518
Semester: 5	Course: CO51-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	04
Title of Experiment	Test various functionality of railway reservation system.

Test Case ID	Test Case Name	Actual Input	Expected Output	Actual Output	Status
TC_01	Username	" Username = tanmay6626 "	It should accept the data	It accepted the data	Pass
TC_02	Password	" Password =Tanmay6626 "	It should accept the data	It accepted the data	Pass
TC_03	Confirm Password	" Confirm Password = Tanmay6626 "	It should accept the data	It accepted the data	Pass
TC_04	First Name	" First Name = Tanmay "	It should accept the data	It accepted the data	Pass
TC_05	Middle Name	" Middle Name =Rajesh "	It should accept the data	It accepted the data	Pass
TC_06	Last Name	" Last Name = Thakur "	It should accept the data	It accepted the data	Pass
TC_07	Date of Birth	" Date of Birth = 11/12/2000 "	It should accept the data	It accepted the data	Pass
TC_08	Gender	Click on Male	It should accept the data	It accepted the data	Pass
TC_09	Mobile Number	" Mobile Number = 9821006626 "	It should accept the data	It accepted the data	Pass
TC_10	Email ID	" Email ID = tanmayt124@gmail.com "	It should accept the data	It accepted the data	Pass
TC_11	Next Button	Click on Next Button and fill all details	It should go to next page of Registration and Send OTP on Mail ID and Phone Number	It is going to next page of Registration and sends OTP on Mail ID and Phone Number	Pass
TC_12	Username	" Username =	It should accept	It accepted the	Pass

		tanmay6626 "	the data	data	
TC_13	Password	" Password =Tanmay6626 "	It should accept the data	It accepted the data	Pass
TC_14	Login Button	Click on Login Button	It should go to next page and ask for OTP	It is going to next page and asking for OTP	Pass
TC_15	Mobile OTP	" Mobile OTP = 269591"	It should accept the data	It accepted the data	Pass
TC_16	Email OTP	" Email OTP = 956438 "	It should accept the data	It accepted the data	Pass
TC_17	Verify User	Click on Verify User	It should prompt Congratulation’s dialog box and log in	It is prompting Congratulation’s dialog box logged in	Pass
TC_18	Plan my Journey	Click on Plan my Journey	It should go to next page	It is going to next page	Pass
TC_19	From Button	Click on From Button	It should go to next page and show Search Station	It is going to next page and Showing search station	Pass
TC_20	Search Station	" Search Station = Mumbai Central (MMCT) "	It should accept the data and go to (Search Station) section	It accepted the data and going to (Search Station) section	Pass
TC_21	Search Station	" Search Station = Delhi (DLI) "	It should accept the data and go to Plan my Journey page	It accepted the data and going to Plan my Journey page	Pass
TC_22	Date	Click on First date section and click on select Date	It should show a calendar	It is showing the calendar	Pass
TC_23	Date	Click on second date section and click on select date	It should show a calendar	It is showing the calendar	Pass
TC_24	Search Train Button	Click on Search Train Button	It should go to next page and show all available train	It is going to next page and showing all available train	Pass
TC_25	Train selection	Select any available train and select Sleeper coach	It should availability of seats and book now option	It is showing availability of seats and book now option	Pass
TC_26	Class selection	Select any suitable class and Click Book now	It should go to next page and get confirmation	It is going to next page and getting	Pass

			of booking from user	conformation from user	
TC_27	Add Passenger Button	Click on Add Passenger Button	It should go to next page	It is going to next page	Pass
TC_28	Name	" Name = Tanmay "	It should accept the data	It accepted the data	Pass
TC_29	Age Between 05 to 125	" Age Between 05 to 125 = 18 "	It should accept the data	It accepted the data	Pass
TC_30	Gender	Click on Male	It should accept the data	It accepted the data	Pass
TC_31	Select Birth Preference	" Select Birth Preference = SIDE UPPER "	It should accept the data	It accepted the data	Pass
TC_32	Select Concession Type	" Select Concession Type = General "	It should accept the data	It accepted the data	Pass
TC_33	Done	Click on Done	It should go to confirmation page	It is going to confirmation page	Pass
TC_34	Edit Passenger	Click on passenger	It should go to edit page	It is going to edit page	Pass
TC_35	Age Between 05 to 125	" Age Between 05 to 125 = 65 "	It should accept the data and show Option for Senior Citizen Concession	It is showing Option for Senior Citizen Concession	Pass
TC_36	Option for Senior Citizen Concession	Select any Concession type	It should accept the data	It is accepting the data	Pass

Practical Related Questions

1. Give significance of testing in railway reservation system.

   The significance of testing in railway reservation are as follows:

   a) Testing determines the quality of software after a programmer develops it. Quality needs to be maintained throughout the day.
   b) It is also essential to make sure about customer's reliability & satisfaction.
   c) Railway Reservation System is going to be used by many people, so quality and integrity both matters.
   d) To maintain both quality and integrity, testing plays an important role.
   e) Testing is also done to fix bugs and errors occurred during development phase.

2. What are different test methodologies that shall be applied while testing railway reservation system? Justify.

   Testing methodologies are divided into two parts:
   a) Functional Testing: It involves testing the applications against the business requirements. Functional testing involves following:
      1. Unit testing
      2. Integration testing
      3. System Testing
      4. Acceptance Testing
   b) Non-Functional Testing: It involves testing methods incorporate all test types of methods that focus on operational aspects of a piece of software. Non-functional testing includes following methods of testing:
      1. Performance
      2. Security
      3. Usability
      4. Compatibility

   Both types of testing ensure the quality of software.

3. Give the significance of system testing in railway reservation system.

   a) System testing is important part in software testing.
   b) System testing means testing the software as whole.
   c) System testing is done in railway reservations system to check whether it meets system / specified business requirements or not.
   d) System testing helps to minimize troubleshooting and support calls after deployments.

VP
Vidyalankar
Polytechnic

DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	05
Title of Experiment	Validate login procedure for E-Commerce application

Test Case	Test Case Name	Actual Input	Expected Output	Acutal Output	Status
TC_01	Mobile Number Registration	Mobile Number = "abcd"	Invalid Mobile Number	Invalid Mobile Number	Pass
TC_02	Mobile Number Registration	Mobile Number = "123"	Invalid Mobile Number	Invalid Mobile Number	Pass
TC_03	Mobile Number Registration	Mobile Number = "' .';"/ "	Invalid Mobile Number	Invalid Mobile Number	Pass
TC_04	Mobile Number Registration	Mobile Number = "9372973"	Invalid Mobile Number	Invalid Mobile Number	Pass
TC_05	Mobile Number Registration	Mobile Number = "9372973077"	Valid Mobile Number	Valid Mobile Number	Pass
TC_05	OTP Verification	1	Invalid OTP	Invalid OTP	Pass
TC_06	OTP Verification	123	Invalid OTP	Invalid OTP	Pass
TC_07	OTP Verification	'.';"/	Invalid OTP	Invalid OTP	Pass
TC_08	OTP Verification	123456789	Invalid OTP	Invalid OTP	Pass
TC_09	OTP Verification	abcd	Invalid OTP	Invalid OTP	Pass

TC_10	OTP Verification	3720	Valid OTP	Valid OTP	Pass
TC_10	Search Bar	Search Bar = "12345678"	Could not find any matches	Could not find any matches	Pass
TC_11	Search Bar	Search Bar = "@,./;'[]"	Could not find any matches	Could not find any matches	Pass
TC_12	Search Bar	Search Bar = "Blank Space"	Should not search anything	Does not search anything	Pass
TC_13	Search Bar	Search Bar = "a"	Display all the items containing letter 'a'	Displays all the items containing letter 'a'	Pass
TC_14	Search Bar	Search Bar = "Shirt"	Display all the shirts	Display all the shirts	Pass
TC_14	Filter	Filter = "Men"	Display all the men relaed items	Displays all the men relaed items	Pass
TC_15	Filter	Filter = "Categories = Tshirts"	Display all the Tshirts	Displays all the Tshirts	Pass
TC_16	Filter	Filter = "Categories = 126354"	Display no output	Displays no output	Pass
TC_17	Filter	Filter = "Categories = ';.[@"	Display no output	Displays no output	Pass

Practical Related Questions

1. What are the security threats for E-Commerce Systems?

Answer: The various security threats for an E-Commerce System are as follows:

- **Electronic payments system:** In e-cash, we stored financial information on the computer, electronic device or on the internet which is vulnerable to the hackers
- **The Risk of Tax Evasion:** The problem with electronic systems is that they don't provide cleanly into this paradigm. It makes the process of tax collection very frustrating for the Internal Revenue Service. It is at the business's choice to disclose payments received or made via electronic payment systems. The IRS has no way to know that it is telling the truth or not that makes it easy to evade taxation.
- **Denial of service attacks:** A denial-of-service attack (DoS attack) is a security attack in which the attacker takes action that prevents the legitimate (correct) users from accessing the electronic devices.
- **The Risk of Payment Conflicts:** In electronic payment systems, the payments are handled by an automated electronic system, not by humans. The system is prone to errors when it handles large amounts of payments on a frequent basis with more than one recipients involved.

2. List various authentication protocols that can be used in providing security for E-Commerce System

Answer: various authentication protocols that can be used in providing security for E-Commerce System are as follows:

- **Password Authentication Protocol (PAP):** It is essentially a routine log in process that requires a username and password combination to access a given system, which validates the provided credentials.

- **Challenge Handshake Authentication Protocol (CHAP):** CHAP is an identity verification protocol that verifies a user to a given network with a higher standard of encryption using a three-way exchange of a “secret.”
- **Extensible Authentication Protocol (EAP):** This protocol supports many types of authentication, from one-time passwords to smart cards. When used for wireless communications, EAP is the highest level of security as it allows a given access point and remote device to perform mutual authentication with built-in encryption

3. Describe various encryption techniques that can be used to provide storing login credentials.

Answer: Various encryption techniques that can be used to provide storing login credentials are as follows:

- **Triple DES – Replaces Data encryption standard(DES)** algorithm, uses 3 individual keys with 56 bit.TripleDES is an advanced DES form that applies the DES cipher algorithm thrice to all the data blocks. They are used to encrypt ATM PINs, etc.
- **RSA**: Public encryption algorithm to protect the data over the internet. It is an asymmetric key encryption algorithm which uses public and private keys. RSA is an algorithm based on the factorization of the product of two prime numbers. If the receiver knows these numbers only then, he/she can decrypt the message. RSA finds its applications in digital signatures but is often slow when a large volume of data is to be encrypted.
- **Blowfish:** It splits the message into 64 bits and encrypts them, is used in certain payment gateways. It is fast, effective and flexible. Blowfish finds its application in embedded systems and has been deemed as reasonably secure.
- **Twofish:** Keys in this algorithm are 256 bits in length and it is a symmetric key encryption technique. Twofish is still in use by many file and folder encryption software solutions. It is a license-free technique to encrypt 128

bits of a data block, it also always encrypts data in rounds of 16, which makes it slower.

- **AES:** Advanced encryption standard, trusted by many standard organizations. It can encrypt 128 bit, 192 bit as well as 256-bit. AES is a symmetric encryption algorithm that is mostly in use today. AES is used for both rest data as well as at transit.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	06
Title of Experiment	Testing Functionality of Web page

Practical Related Questions

1. Give significance of performance and stress testing for web applications.

Answer:

Significance of Performance testing in web application:

- Developers do performance testing for web application projects to assess the overall speed, stability, and scalability of the project.
- Project teams implement performance testing to ensure the final build meets the expectations of an end user and helps a business achieve its objectives.
- As a result of a testing session, companies might end up with a list of mismatches between expected and actual performance.
- Having a deeper insight into the weaker point of an app helps during fine-tuning and brings more substance to the decision-making process.

Significance of Stress testing in web application:

- Stress testing helps teams define issues that only become visible under peak load conditions.
- During this type of system validation, QA specialists simulate loads that exceed reasonable estimates to test the performance of web application.
- Stress testing prepares the maintenance team for extreme situations and helps establish triggers of system shutdown for proactive management.
- Stress tests can also help you uncover the following:

- Synchronization and timing bugs
- Interlock problems
- Priority problems
- Resource loss bugs
- Memory leaks
- Data loss & corruption

2. What is the necessity of configuration testing? Describe various configurations that are required to be considered while performing website testing.

Answer: The objective of configuration testing is:

- To determine whether the software application fulfils the configurability requirements.
- To identify the defects that were not efficiently found during different testing processes.
- To determine an optimal configuration of the application under test.
- To do analyse of the performance of software application by changing the hardware and software resources.
- To do analyse of the system efficiency based on the prioritization.
- To verify the degree of ease to how the bugs are reproducible irrespective of the configuration changes.

Various Configurations that are required to be considered while performing website testing:

- **Operating System Configuration:**
  Win XP, Win 7 32/64 bit, Win 8 32/64 bit, Win 10 etc.
- **Database Configuration:**
  Oracle, DB2, MySql, MSSQL Server, Sybase etc.
- **Browser Configuration:**
  IE 8, IE 9, FF 16.0, Chrome, Microsoft Edge etc.

3. Describe various security testing to be performed for web application. Give its importance

Answer:

- **Vulnerability Scanning**: This is done through automated software to scan a system against known vulnerability signatures.
- **Security Scanning:** It involves identifying network and system weaknesses, and later provides solutions for reducing these risks. This scanning can be performed for both Manual and Automated scanning.
- **Penetration testing**: This kind of testing simulates an attack from a malicious hacker. This testing involves analysis of a particular system to check for potential vulnerabilities to an external hacking attempt.
- **Risk Assessment:** This testing involves analysis of security risks observed in the organization. Risks are classified as Low, Medium and High. This testing recommends controls and measures to reduce the risk.
- **Security Auditing:** This is an internal inspection of Applications and Operating systems for security flaws. An audit can also be done via line by line inspection of code
- **Ethical hacking:** It's hacking an Organization Software systems. Unlike malicious hackers, who steal for their own gains, the intent is to expose security flaws in the system.
- **Posture Assessment:** This combines Security scanning, Ethical Hacking and Risk Assessments to show an overall security posture of an organization.

Exercise:

1. Prepare the test case to test your college website for any 5 links.
2. Execute the above test case created in question 1 by performing appropriate operations and verify result.

Test Case	**Test Case Name**	**Actual Input**	**Expected Output**	**Acutal Output**	**Status**
TC_01	Home > Academics	Click on Computer Engineering	It should redirect to Computer Engineering page	It redirects to Compute r Engineering page Pas	Pass
TC_02	Home > Computer Engineering > First Year Syllabus	Click on First Year Syllabus	It should redirect to First Year Syllabus page	It redirects to First Year Syllabus page	Pass
TC_03	Home > Computer Engineering > Second Year Syllabus	Click on Second Year Syllabus	It should redirect to Second Year Syllabus page	It redirects to Second Year Syllabus page	Pass
TC_04	Home > Computer Engineering > Third Year Syllabus	Click on Third Year Syllabus	It should redirect to Third Year Syllabus page	It should redirect to Third Year Syllabus page	Pass
TC_05	Home > Academics	Click on Electrical Engineering	It should redirect to Electrical Engineering page	It redirects to Electrical Engineering page	Pass
TC_06	Home > Electrical Engineering > First Year Syllabus	Click on First Year Syllabus	It should redirect to First Year Syllabus page	It redirects to First Year Syllabus page	Pass
TC_07	Home > Electrical Engineering > Second Year Syllabus	Click on Second Year Syllabus	It should redirect to Second Year Syllabus page	It redirects to Second Year Syllabus page	Pass

TC_07	Home > Electrical Engineering > Third Year Syllabus	Click on Third Year Syllabus	It should redirect to Third Year Syllabus page	It should redirect to Third Year Syllabus page	Pass
TC_08	Home > Academics	Click on Information Technology	It should redirect to Information Technology page	It redirects to Information Technology page	Pass
TC_09	Home > Information Technology > First Year Syllabus	Click on First Year Syllabus	It should redirect to First Year Syllabus page	It redirects to First Year Syllabus page	Pass
TC_10	Home > Information Technology > Second Year Syllabus	Click on Second Year Syllabus	It should redirect to Second Year Syllabus page	It redirects to Second Year Syllabus page	Pass
TC_11	Home > Information Technology > Third Year Syllabus	Click on Third Year Syllabus	It should redirect to Third Year Syllabus page	It should redirect to Third Year Syllabus page	Pass
TC_12	Home > Student Life	Click on Student Life	It should redirect to Student Life page	It redirects to Student Life page	Pass
TC_13	Home > Placement page	Click on Placement	It should redirect to Placement page	It redirects to Placement Page	Pass
TC_14	Home > Online Payment page	Click on Online Payment	It should redirect to Online Payment page	It redirects to Online Payment Page	Pass
TC_15	Home > Latest Updates page	Click on Latest Updates	It should redirect to Latest Updates page	It redirects to Latest Updates Page	Pass
TC_16	Home >Project Outline page	Click on Project Outline	It should redirect to Project Outline page	It redirects to Project Outline Page	Pass
TC_17	Home > Who are we page	Click on Who are we	It should redirect to Who are we page	It redirects to Who are we Page	Pass

TC_18	Home > Alumni page	Click on Alumni	It should redirect to Alumni page	It redirects to Alumni Page	Pass
TC_19	Home > Contact Us page	Click on Contact Us	It should redirect to Contact Us page	It redirects to Contact Us Page	Pass
TC_20	Home > Notice Board page	Click on Notice Board	It should redirect to Notice Board page	It redirects to Notice Board Page	Pass

3. Prepare test case for any website which sends OTP on your email address/ mobile number.

   Test case for any GitHub Sign up

Test Case	**Test Case Name**	**Actual Input**	**Expected Output**	**Acutal Output**	**Status**
TC_01	Email	anuragDubey	Should not allow Email	Does not allow Email	Pass
TC_02	Email	451224	Should not allow Email	Does not allow Email	Pass
TC_03	Email	Anuragd393.3@gmail.com	Should allow Email	Does allow Email	Pass
TC_04	Username	anruag	Display ‘Not available’	Displays ‘Not available’	Pass
TC_05	Username	1224	Display ‘Username should include alphabets, numbers’	Displays ‘Username should include alphabets, numbers’	Pass
TC_06	Username	anruag=3933	Should allow username	Does allow username	Pass
TC_07	Password	1234	Display ‘Enter strong password’	Displays ‘Enter strong password’	Pass
TC_07	Password	password@123	Should allow password	Does allow password	Pass
TC_08	Continue	Click on Continue	Should accept all the data and redirect to email verification page	Does accept all the data and redirect to email	Pass

				verification page	
TC_09	OTP	0000	Display ‘Invalid OTP’	Displays ‘Invalid OTP’	Pass
TC_10	OTP	5461	Should redirect to home page of github	Does redirect to home page of github	Pass

VP Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	07
Title of Experiment	Design test case for control and decision making statements.

Practical Related Questions

1. State different ways to test the decision and control statements.

   Answer: The different ways to test the decision and control statements are

   - Branch coverage -It attempts to cover all paths (True and false) in the Software code.
   - Condition coverage - It attempts to cover branches with Boolean expression in software code

2. For any decision statement what will be the possible outcomes while writing the test cases.

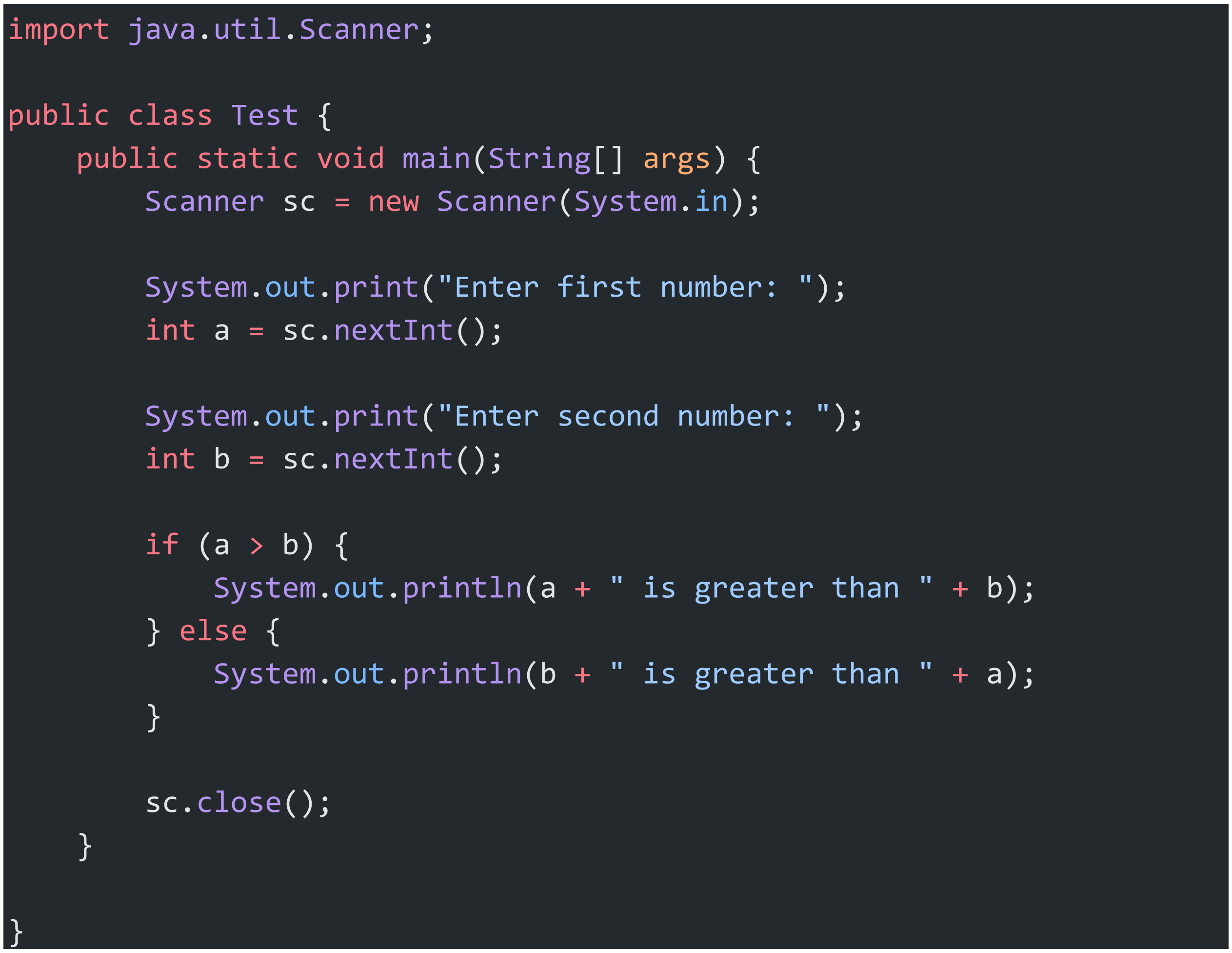
   Answer: Whenever there are two or more possible exits from the statement like an IF statement, a DO-WHILE or a CASE statement it is known as decision because in all these statements there are two outcomes, either TRUE or FALSE. Alternatively, you can say that control statement IF has been evaluated both to TRUE and FALSE

3. Can we test the relational operator? Validate your answer with justification.

   Answer: Yes, we can test the relational operators.

   Example:

   Code:



```java
import java.util.Scanner;

public class Test {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter first number: ");
        int a = sc.nextInt();

        System.out.print("Enter second number: ");
        int b = sc.nextInt();

        if (a > b) {
            System.out.println(a + " is greater than " + b);
        } else {
            System.out.println(b + " is greater than " + a);
        }

        sc.close();
    }

}
```

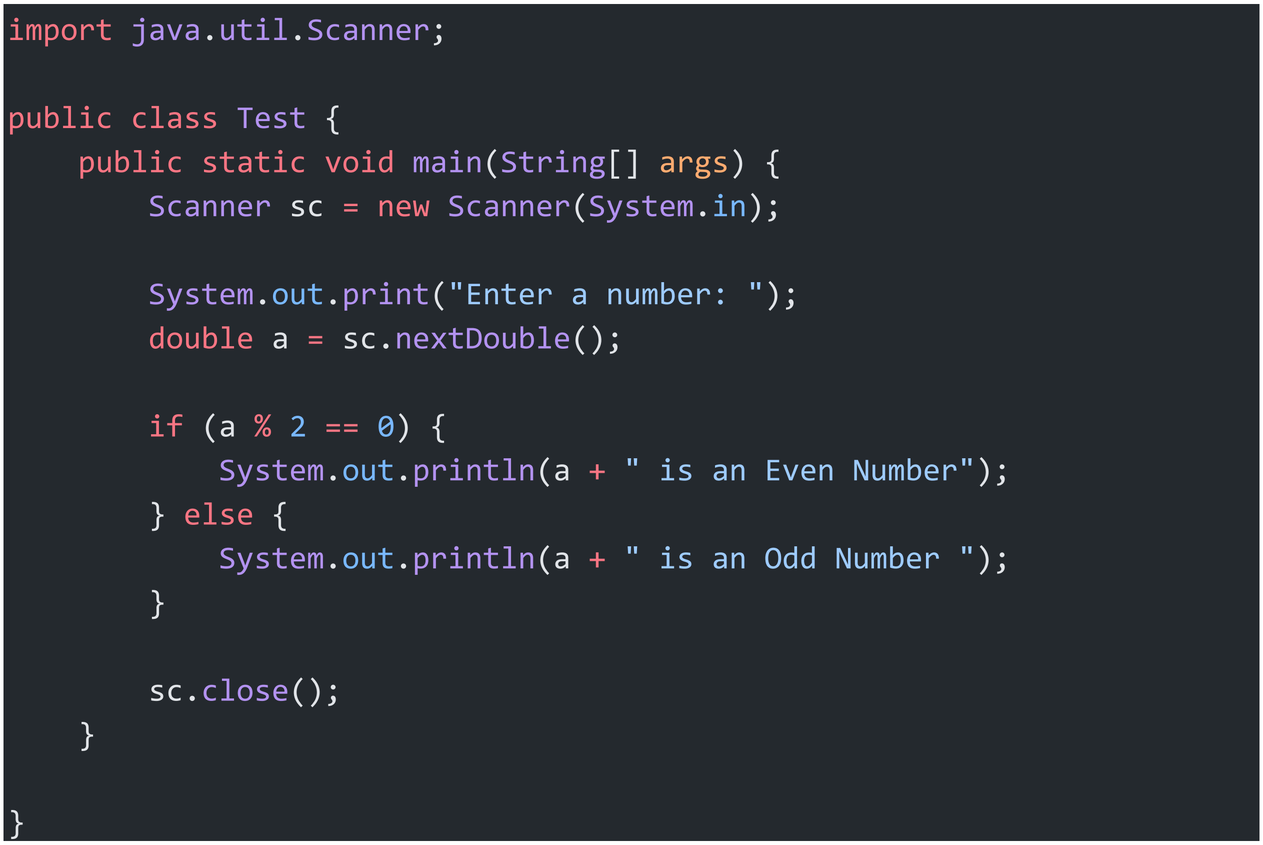
Test case:

Test Case	Test Case Name	Actual Input	Expected Output	Actual Output	Status
TC_01	Greater than	a = 10 b = 20	20 is greater than 10	20 is greater than 10	Pass
TC_02	Greater than	a = 30 b = 20	30 is greater than 20	30 is greater than 20	Pass

Exercise:

1. Generate the test case to check the program written for Even and Odd numbers.

Code:



```java
import java.util.Scanner;

public class Test {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a number: ");
        double a = sc.nextDouble();

        if (a % 2 == 0) {
            System.out.println(a + " is an Even Number");
        } else {
            System.out.println(a + " is an Odd Number ");
        }

        sc.close();
    }

}
```

Test case:

Test Case	**Test Case Name**	**Actual Input**	**Expected Output**	**Actual Output**	**Status**
TC_01	Even or Odd	Enter a number:12	12 is an Even Number	12 is an Even Number	Pass
TC_02	Even or Odd	Enter a number: 13	13 is an Odd Number	13 is an Odd Number	Pass

2. Execute above test case in Question 1 by entering following inputs and verify results.
   Input – 4, 7, 2.5, 8.1

Test case:

Test Case	Test Case Name	Actual Input	Expected Output	Actual Output	Status
TC_01	Even or Odd	Enter a number:4	4 is an Even Number	4 is an Even Number	Pass
TC_02	Even or Odd	Enter a number: 7	7 is an Odd Number	7 is an Odd Number	Pass
TC_03	Even or Odd	Enter a number: 2.5	2.5 is an Odd Number	2.5 is an Odd Number	Pass
TC_04	Even or Odd	Enter a number: 8	8 is an Even Number	8 is an Even Number	Pass
TC_05	Even or Odd	Enter a number: 1	1 is an Odd Number	1 is an Odd Number	Pass

3. Generate the test case to check the program written for printing the day of week.

   Code:

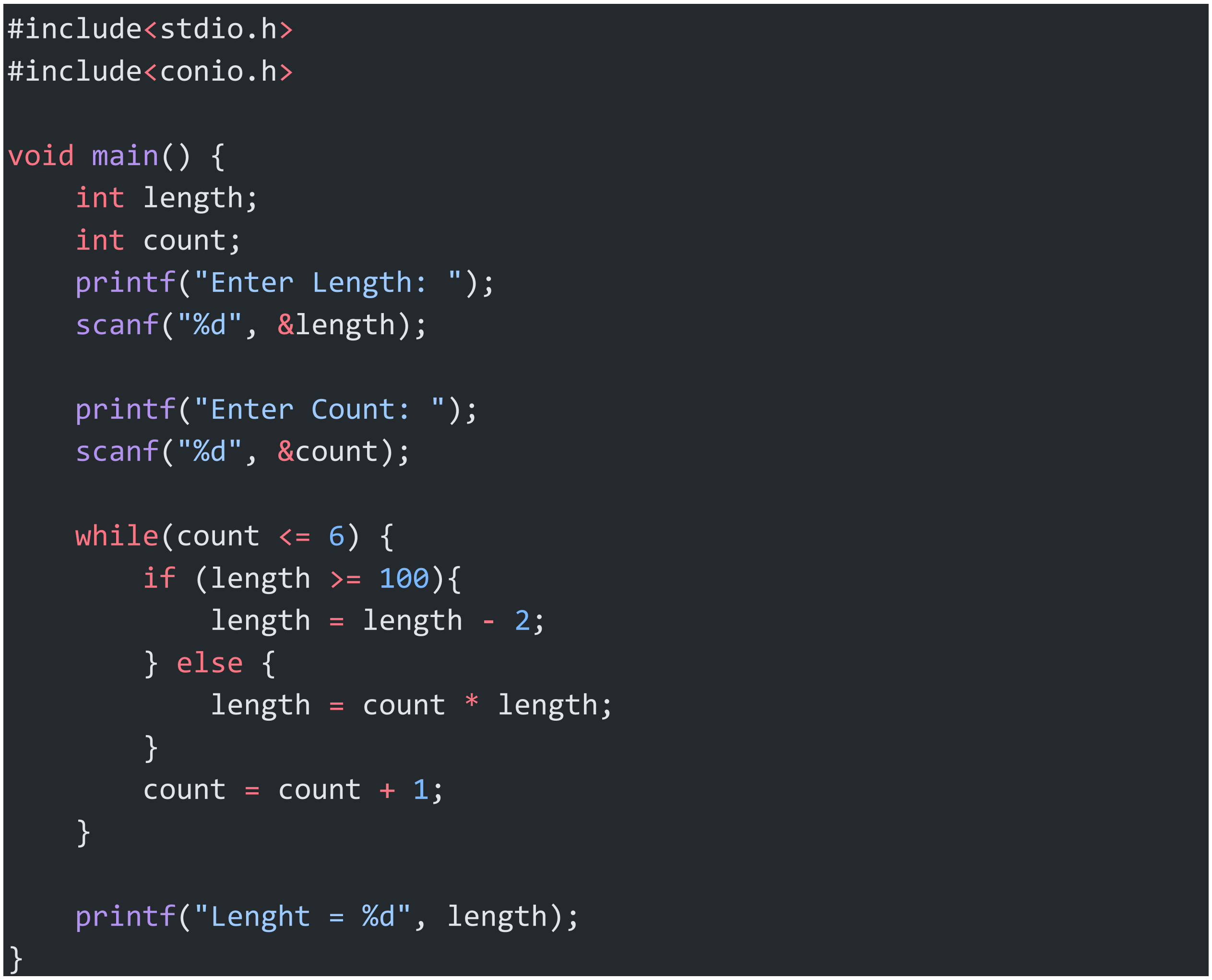
```
import java.util.Scanner;

public class Test {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int weekday = sc.nextInt();

        switch (weekday) {
        case 1:
            System.out.println("Monday");
            break;
        case 2:
            System.out.println("Tuesday");
            break;
        case 3:
            System.out.println("Wednesday");
            break;
        case 4:
            System.out.println("Thursday");
            break;
        case 5:
            System.out.println("Friday");
            break;
        case 6:
            System.out.println("Saturday");
            break;
        case 7:
            System.out.println("Sunday");
            break;
        default:
            System.out.println("Please enter weekday number between 1-
7.");
        }

        sc.close();
    }

}
```

Test case:

Test Case	**Test Case Name**	**Actual Input**	**Expected Output**	**Actual Output**	**Status**
TC_01	Day of week	Enter a number: 1	Monday	Monday	Pass
TC_02	Day of week	Enter a number: 2	Tuesday	Tuesday	Pass
TC_03	Day of week	Enter a number: 3	Wednesday	Wednesday	Pass
TC_04	Day of week	Enter a number: 4	Thursday	Thursday	Pass
TC_05	Day of week	Enter a number: 5	Friday	Friday	Pass
TC_06	Day of week	Enter a number: 6	Saturday	Saturday	Pass
TC_07	Day of week	Enter a number: 7	Sunday	Sunday	Pass
TC_08	Day of week	Enter a number: 0	Please enter weekday number between 1-7.	Please enter weekday number between 1-7.	Pass
TC_09	Day of week	Enter a number: b	Please enter weekday number between 1-7.	Please enter weekday number between 1-7.	Pass
TC_010	Day of week	Enter a number: #	Please enter weekday number between 1-7.	Please enter weekday number between 1-7.	Pass

4. Create the test cases for following algorithm and write the ‘Expected Outcome’ and ‘Actual Outcome’ in following table by executing the code.

Code:



```
#include<stdio.h>
#include<conio.h>

void main() {
    int length;
    int count;
    printf("Enter Length: ");
    scanf("%d", &length);

    printf("Enter Count: ");
    scanf("%d", &count);

    while(count <= 6) {
        if (length >= 100){
            length = length - 2;
        } else {
            length = count * length;
        }
        count = count + 1;
    }

    printf("Lenght = %d", length);
}
```

Test case:

Test Case	Count	Length	Expected Output	Actual Output
1	5	101	594	594
2	5	99	493	493
3	7	99	99	99
4	0	0	0	0

	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	08
Title of Experiment	Prepare Test plan for an identified Mobile Application

Practical Related Questions

1. On which basis the Test Approach is decided for any application?

   Answer: A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approach has two techniques:

   - **Proactive -** An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
   - **Reactive -** An approach in which the testing is not started until after design and coding are completed.

2. Define the Test Management

   Answer:

   - Test Management is a process of managing the testing activities in order to ensure high quality and high-end testing of the software application.
   - The method consists of organizing, controlling, ensuring traceability and visibility of the testing process in order to deliver the high quality software application.

3. State the use of Test Case Specification. Enlist the Test Case Specification Identifiers.

Answer:

Use of Test Case Specification:

It specifies the purpose of a specific test, identifies the required inputs and expected results, provides step-by-step procedures for executing the test, and outlines the pass/fail criteria for determining acceptance.

Test Case Specification Identifiers:

- **Test Case Objectives:** Purpose of the test
- **Test Items:** Items (e.g., requirement specifications, design specifications, code, etc.) required to run a particular test case. This should be provided in "Notes" or "Attachment" feature. It describes the features and conditions required for testing.
- **Input Specifications:** Description of what is required (step-by-step) to execute the test case (e.g., input files, values that must be entered into a field, etc.). This should be provided in "Action" field.
- **Output Specifications:** Description of what the system should look like after the test case is run. This should be provided in the "Expected Results" field.
- Environmental Needs: Description of any special environmental needs. This includes system architectures, Hardware & Software tools, records or files, interfaces, etc

4. Enlist the parameters that should be considered while preparing Test Summary Report.

   Answer:

   - Test summary report is a document which contains summary of test activities and final test results. After the testing cycle it is very important that you communicate the test results and findings to the project stakeholders.
   - So that decisions can be made for the software release i.e. .If further testing is required and we need to delay the release.
   - Test summary report will be different for different kind of testing.
   - It should include relevant metrics and details regarding the software testing process, such as test case adequacy, cost of finding defects, test case effectiveness, test efficiency, rework & review efforts ratio, etc.
   - In addition to test coverage and unresolved defects test summary reports should also contain test strategy, test objectives and overall result of test effort

Exercise:

1. Prepare test plan along with the test case to check the any chatting application.
2. Prepare the test summary report for the application used in Quest

Test Plan
Swiggy Version 3.77.1

Version: 1.0

Created: 19/11/2021
Last Updated: 19/11/2021

[illegible]

Document History

Version	Date	Author	Description of Change
1	1/11/2021	Anurag Dubey	Draft
2	1/11/2021	Vinay Savla	Draft
3	12/11/2021	Sahil Makhijani	Draft - Reviewed
4	12/11/2021	Kuber Dhure	Draft - Reviewed

Approvers List

Name	Role	Approver / Reviewer	Approval / Review Date
Vijay Patil	Project Manager		
Supriya Kadam	Test Lead		
Meenakshi Khamkar	Test Lead		
Anurag Dubey	Technical Lead		

Reference Documents -

Version	Date	Document Name
1.0		WHATSAPP **Version 2.21.23.17**

Table of Contents

1. INTRODUCTION

1.1. Purpose

This test plan describes the testing approach and overall framework that will drive the testing of the WHATSAPP application. The document introduces:

- Test Strategy: rules the test will be based on, including the givens of the project (e.g.: start / end dates, objectives, assumptions); description of the process to set up a valid test (e.g.: entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
- Execution Strategy: describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
- Test Management: process to handle the logistics of the test and all the events that come up during execution (e.g.: communications, escalation procedures, risk and mitigation, team roster)

1.2. Project Overview

WhatsApp Messenger is a cross-platform instant messaging application that allows iPhone, BlackBerry, Android, Windows Phone and Nokia smartphone users to exchange text, image, video and audio messages for free. WhatsApp is especially popular with end users who do not have unlimited text messaging.

Audience

- Project team members perform tasks specified in this document, and provide input and recommendations on this document.
- Project Manager Plans for the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the task herein specified, approves the document and is accountable for the results.
- The stakeholders' representatives and participants (individuals as identified by the PM or Leads) may take part in the UAT test to ensure the business is aligned with the results of the test.
- Technical Team ensures that the test plan and deliverables are in line with the design, provides the environment for testing and follows the procedures related to the fixes of defects.
- Business analysts will provide their inputs on functional changes.

2. TEST STRATEGY

2.1. Test Objectives

The objective of the test is to verify that the functionality of WHATSAPP works according to the specifications.

The test will execute and verify the test scripts, identify, fix and retest all high and medium severity defects per the entrance criteria, prioritize lower severity defects for future fixing.

The final product of the test is twofold:

- A production-ready software;
- A set of stable test scripts that can be reused for Functional and UAT test execution.

2.2. Test Assumptions

Key Assumptions

- Production like data required and be available in the system prior to start of Functional Testing

General

- Exploratory Testing would be carried out once the build is ready for testing
- Performance testing is not considered for this estimation.
- All the defects would come along with a snapshot JPEG format
- The Test Team will be provided with access to Test environment via VPN connectivity
- The Test Team assumes all necessary inputs required during Test design and execution will be supported by Development/BUSINESS ANALYSTs appropriately.
- Test case design activities will be performed by QC Group
- Test environment and preparation activities will be owned by Dev Team
- Dev team will provide Defect fix plans based on the Defect meetings during each cycle to plan. The same will be informed to Test team prior to start of Defect fix cycles
- BUSINESS ANALYST will review and sign-off all Test cases prepared by Test Team prior to start of Test execution
- The defects will be tracked through BUGZILLA only. Any defect fixes planned will be shared with Test Team prior to applying the fixes on the Test environment
- Project Manager/BUSINESS ANALYST will review and sign-off all test deliverables

- The project will provide test planning, test design and test execution support
- Test team will manage the testing effort with close coordination with Project PM/BUSINESS ANALYST
- Project team has the knowledge and experience necessary, or has received adequate training in the system, the project and the testing processes.
- There is no environment downtime during test due to outages or defect fixes.
- The system will be treated as a black box; if the information shows correctly online and in the reports, it will be assumed that the database is working properly.

Functional Testing

- During Functional testing, testing team will use preloaded data which is available on the system at the time of execution
- The Test Team will be perform Functional testing only on WHATSAPP

UAT

- UAT test execution will be performed by end users (L1, L2 and L3) and QC Group will provide their support on creating UAT script.

2.3. Test Principles

- Testing will be focused on meeting the business objectives, cost efficiency, and quality.
- There will be common, consistent procedures for all teams supporting testing activities.
- Testing processes will be well defined, yet flexible, with the ability to change as needed.
- Testing activities will build upon previous stages to avoid redundancy or duplication of effort.
- Testing environment and data will emulate a production environment as much as possible.
- Testing will be a repeatable, quantifiable, and measurable activity.
- Testing will be divided into distinct phases, each with clearly defined objectives and goals.
- There will be entrance and exit criteria.

2.4. Data Approach

- In functional testing, WHATSAPP will contain pre-loaded test data and which is used for testing activities.

2.5. Scope and Levels of Testing

2.5.1. Exploratory

PURPOSE: the purpose of this test is to make sure critical defects are removed before the next levels of testing can start.

SCOPE: First level navigation, dealer and admin modules

TESTERS: Testing team.

METHOD: this exploratory testing is carried out in the application without any test scripts and documentation

TIMING: at the beginning of each cycle.

2.5.2. Functional Test

PURPOSE: Functional testing will be performed to check the functions of application. The functional testing is carried out by feeding the input and validates the output from the application.

Scope: The below excel sheet details about the scope of Functional test. Note: The scope is high level due to changes in the requirement.

TESTERS: Testing Team.

METHOD: The test will be performed according to Functional scripts, which are stored in Selenium.

TIMING: After Exploratory test is completed.

TEST ACCEPTANCE CRITERIA

1. Approved Functional Specification document, Use case documents must be available prior to start of Test design phase.
2. Test cases approved and signed-off prior to start of Test execution
3. Development completed, unit tested with pass status and results shared to Testing team to avoid duplicate defects
4. Test environment with application installed, configured and ready to use state

Sign-off

- Approved Functional Specification Document
- Approved Use cases
- Approved Test cases

Readiness

- Development completed & unit tested
- Application deployed and system ready for testing on Test environment
- Production like data is available to test all functionalities.
- Defect fixes planned based on Defect triage (Unit Testing) and evaluation criteria

TEST DELIVERABLES

Sr.No.	Deliverable Name	Author	Reviewer
1.	Test Plan	Test Lead	Project Manager/ Business Analyst's
2.	Functional Test Cases	Test Team	Business Analyst's Sign off
3.	Logging Defects in BUGZILLA	Test Team	Test Lead/ Programming Lead
4.	Daily/weekly status report	Test Team/ Test Lead	Test Lead/ Project Manager
5.	Test Closure report	Test Lead	Project Manager

MILESTONE LIST

The milestone list is tentative and may change due to below reasons

a) Any issues in the System environment readiness
b) Any change in scope/addition in scope
c) Any other dependency that impacts efforts and timelines

2.5.3. User Acceptance Test (UAT)

PURPOSE: this test focuses on validating the business logic. It allows the end users to complete one final review of the system prior to deployment.

TESTERS: the UAT is performed by the end users (L1, L2 and L3).

METHOD: Since the business users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts. Test team write the UAT test cases based on the inputs from End user (L1, L2 and L3 users) and Business Analyst's.

TIMING: After all other levels of testing (Exploratory and Functional) are done. Only after this test is completed the product can be released to production.

TEST DELIVERABLES

Sr.No.	Deliverable Name	Author	Reviewer
1.	UAT Test Cases	Test Team	Business Analyst's Sign off

2.6. Test Effort Estimate

This document lists out all the activities that have to be performed by the QC team and estimates how many man-hours each activity is going to take.

New_Detailed DRFT
Test estimate v1.xlsx

3. EXECUTION STRATEGY

3.1. Entry and Exit Criteria

- The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.
- The exit criteria are the desirable conditions that need to be met in order proceed with the implementation.
- Entry and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final "go-no go" decision.
- Entry criteria to start the execution phase of the test: the activities listed in the Test Planning section of the schedule are 100% completed.
- Entry criteria to start each cycle: the activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

Exit Criteria	Test Team	Technical Team	Notes
100% Test Scripts executed			
95% pass rate of Test Scripts			
No open Critical and High severity defects			
95% of Medium severity defects have been closed			
All remaining defects are either cancelled or documented as Change Requests for a future release			
All expected and actual results are captured and documented with the test script			
All test metrics collected based on reports from BUGZILLA			
All defects logged in BUGZILLA			

Test Closure Memo completed and signed off		▨	
Test environment cleanup completed and a new back up of the environment	▨		

3.2. Test Cycles

  - There will be two cycles for functional testing. Each cycle will execute all the scripts.
  - The objective of the first cycle is to identify any blocking, critical defects, and most of the high defects. It is expected to use some work-around in order to get to all the scripts.
  - The objective of the second cycle is to identify remaining high and medium defects, remove the work-around from the first cycle, correct gaps in the scripts and obtain performance results.
- UAT test will consist of one cycle.

3.3. Validation and Defect Management

- It is expected that the testers execute all the scripts in each of the cycles described above. However it is recognized that the testers could also do additional testing if they identify a possible gap in the scripts. This is especially relevant in the second cycle, when the Business analyst's join the TCOE in the execution of the test, since the BUSINESS ANALYSTs have a deeper knowledge of the business processes. If a gap is identified, the scripts and traceability matrix will be updated and then a defect logged against the scripts.
- The defects will be tracked through BUGZILLA only. The technical team will gather information on a daily basis from BUGZILLA, and request additional details from the Defect Coordinator. The technical team will work on fixes.
- It is the responsibility of the tester to open the defects, link them to the corresponding script, assign an initial severity and status, retest and close the defect; it is the responsibility of the Defect Manager to review the severity of the defects and facilitate with the technical team the fix and its implementation, communicate with testers when the test can continue or should be halt, request the tester to retest, and modify status as the defect progresses through the cycle; it is the responsibility of the technical team to review BUGZILLA on a daily basis, ask for details if necessary, fix the defect, communicate to the Defect Manager the fix is done, implement the solution per the Defect Manager request.

Defects found during the Testing will be categorized according to the bug-reporting tool "BUGZILLA" and the categories are:

Severity	Impact
1 (Critical)	▪ This bug is critical enough to crash the system, cause file corruption, or cause potential data loss ▪ It causes an abnormal return to the operating system (crash or a system failure message appears). ▪ It causes the application to hang and requires re-booting the system.
2 (High)	▪ It causes a lack of vital program functionality with workaround.
3 (Medium)	▪ This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality - for example through another screen. ▪ This bug prevents other areas of the product from being tested. However other areas can be independently tested.
4 (Low)	▪ There is an insufficient or unclear error message, which has minimum impact on product use.
5(Cosmetic)	▪ There is an insufficient or unclear error message that has no impact on product use.

3.4. Test Metrics

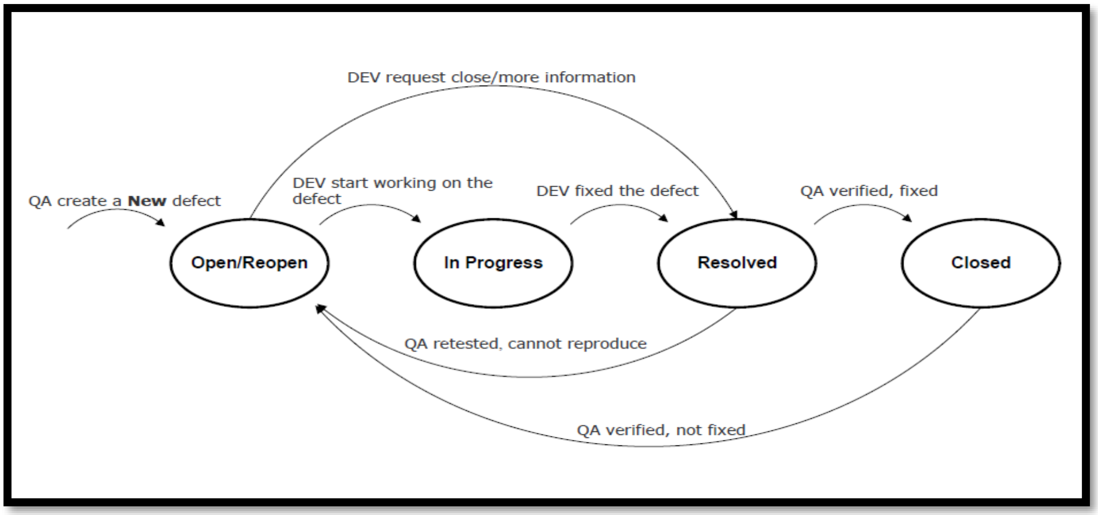
Test metrics to measure the progress and level of success of the test will be developed and shared with the project manager for approval. The below are some of the metrics

Report	Description	Frequency
Test preparation & Execution Status	To report on % complete, % Pass, % Fail Defects severity wise Status – Open, closed, any other Status	Weekly / Daily (optional)
Daily execution status	To report on Pass, Fail, Total defects, highlight Showstopper/ Critical defects	Daily

Project Weekly Status report	Project driven reporting (As requested by PM)	Weekly – If project team needs weekly update apart from daily and there is template available with project team to use.

3.5. Defect tracking & Reporting -

Following flowchart depicts Defect Tracking Process:


DEV request close/more information
QA create a **New** defect
DEV start working on the defect
DEV fixed the defect
QA verified, fixed
Open/Reopen
In Progress
Resolved
Closed
QA retested, cannot reproduce
QA verified, not fixed

4. TEST MANAGEMENT PROCESS

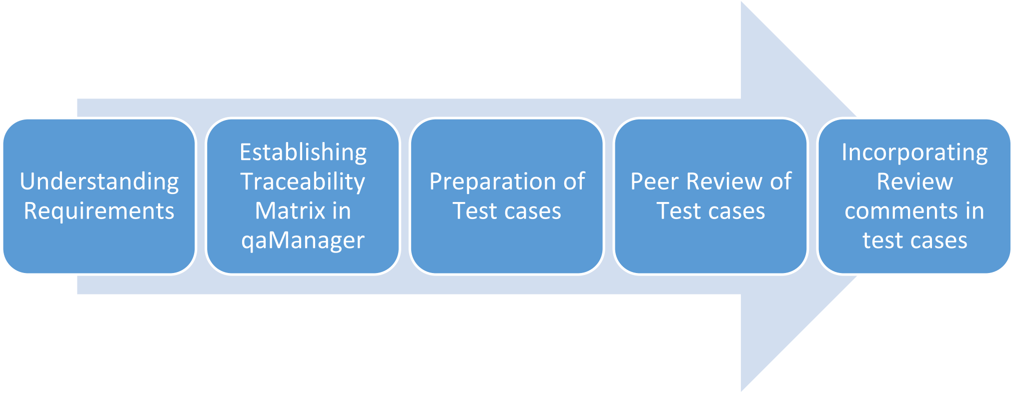
4.1. Test Management Tool

QA Manager Lifecycle Management is the tool used for Test Management. All testing artifacts such as Test cases, test results are updated in the qaManager Lifecycle Management.

- Project specific folder structure will be created in qaManager to manage the status of this project.

- Each resource in the Testing team will be provided with Read/Write access to add/modify Test cases in qaManager.
- During the Test Design phase, all test cases are written directly into qaManager. Any change to the test case will be directly updated in the qaManager.
- Each Tester will directly access their respective assigned test cases and update the status of each executed step in qaManager directly.
- Any defect encountered will be raised in BUGZILLA linking to the particular Test case/test step.
- During Defect fix testing, defects are re-assigned back to the tester to verify the defect fix. The tester verifies the defect fix and updates the status directly in BUGZILLA.
- Various reports can be generated from BUGZILLA to provide status of Test execution. For example, Status report of Test cases executed, Passed, Failed, No. of open defects, Severity wise defects etc.

4.2. Test Design Process

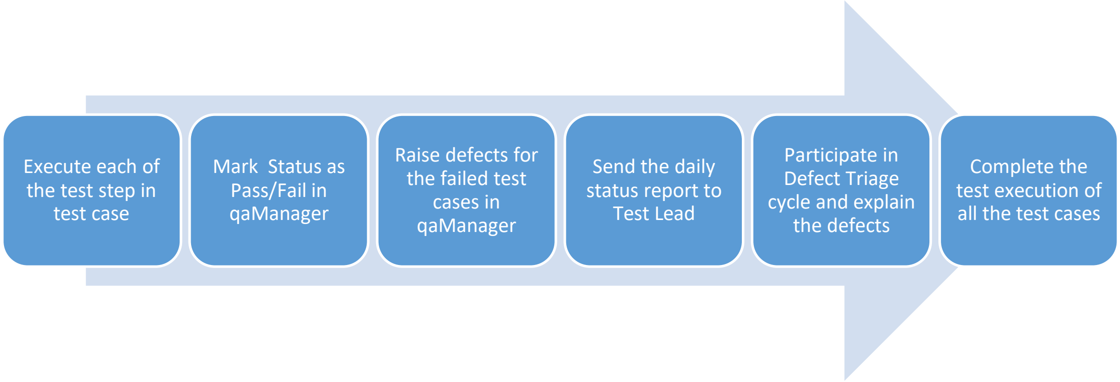

Understanding Requirements
Establishing Traceability Matrix in qaManager
Preparation of Test cases
Peer Review of Test cases
Incorporating Review comments in test cases

- The tester will understand each requirement and prepare corresponding test case to ensure all requirements are covered.
- Each Test case will be mapped to Use cases to Requirements as part of Traceability matrix.
- Each of the Test cases will undergo review by the BUSINESS ANALYST and the review defects are captured and shared to the Test team. The testers will rework on the review defects and finally obtain approval and sign-off.
- During the preparation phase, tester will use the prototype, use case and functional specification to write step by step test cases.
- Testers will maintain a clarification Tracker sheet and same will be shared periodically with the Requirements team and accordingly the test case will

be updated. The clarifications may sometimes lead to Change Requests or not in scope or detailing implicit requirements.

- Sign-off for the test cases would be communicates through mail by Business Analyst's.
- Any subsequent changes to the test case if any will be directly updated in qaManager.

4.3. Test Execution Process


Execute each of the test step in test case
Mark Status as Pass/Fail in qaManager
Raise defects for the failed test cases in qaManager
Send the daily status report to Test Lead
Participate in Defect Triage cycle and explain the defects
Complete the test execution of all the test cases

- Once all Test cases are approved and the test environment is ready for testing, tester will start an exploratory test of the application to ensure the application is stable for testing.
- Each Tester is assigned Test cases directly in qaManager.
- Testers to ensure necessary access to the testing environment, BUGZILLA for updating test status and raise defects. If any issues, will be escalated to the Test Lead and in turn to the Project Manager as escalation.
- If any showstopper during exploratory testing will be escalated to the respective development team for fixes.
- Each tester performs step by step execution and updates the executions status. The tester enters Pass or Fail Status for each of the step directly in BUGZILLA.
- Tester will prepare a Run chart with day-wise execution details
- If any failures, defect will be raised as per severity guidelines in BUGZILLA tool detailing steps to simulate along with screenshots if appropriate.
- Daily Test execution status as well as Defect status will be reported to all stakeholders.
- Testing team will participate in defect triage meetings in order to ensure all test cases are executed with either pass/fail category.
- If there are any defects that are not part of steps but could be outside the test steps, such defects need to be captured in BUGZILLA and map it

Risk	Prob.	Impact	Mitigation Plan
SCHEDULE Testing schedule is tight. If the start of the testing is delayed due to design tasks, the test cannot be extended beyond the UAT scheduled start date.	High	High	• The testing team can control the preparation tasks (in advance) and the early communication with involved parties. • Some buffer has been added to the schedule for contingencies, although not as much as best practices advise.
RESOURCES Not enough resources, resources on boarding too late (process takes around 15 days.	Medium	High	Holidays and vacation have been estimated and built into the schedule; deviations from the estimation could derive in delays in the testing.
DEFECTS Defects are found at a late stage of the cycle or at a late cycle; defects discovered late are most likely be due to unclear specifications and are time consuming to resolve.	Medium	High	Defect management plan is in place to ensure prompt communication and fixing of issues.
SCOPE Scope completely defined	Medium	Medium	Scope is well defined but the changes are in the functionality are not yet finalized or keep on changing.
Natural disasters	Low	Medium	Teams and responsibilities have been spread to two different geographic areas. In a catastrophic event in one of the areas, there will resources in the other areas needed to continue (although at a slower pace) the testing activities.
Non-availability of Independent Test environment and accessibility	Medium	High	Due to non-availability of the environment, the schedule gets impacted and will lead to delayed start of Test execution.
Delayed Testing Due To new Issues	Medium	High	During testing, there is a good chance that some "new" defects may be identified and may become an issue that will take time to resolve. There are defects that can be raised during testing because of unclear document specification. These defects

			can yield to an issue that will need time to be resolved. If these issues become showstoppers, it will greatly impact on the overall project schedule. If new defects are discovered, the defect management and issue management procedures are in place to immediately provide a resolution.

- against the test case level or at the specific step that issue was encountered after confirming with Test Lead.
- This process is repeated until all test cases are executed fully with Pass/Fail status.
- During the subsequent cycle, any defects fixed applied will be tested and results will be updated in BUGZILLA during the cycle.

As per Process, final sign-off or project completion process will be followed

4.4. Test Risks and Mitigation Factors

4.1. Communications Plan and Team Roster

4.2. Role Expectations

The following list defines in general terms the expectations related to the roles directly involved in the management, planning or execution of the test for the project.

	Roles	Name
1.	Project Manager	Vijay Patil
2.	Test Lead	Vinay Savla
3.	Business Analyst	Arfia Shaikh
4.	Development Lead	Sahil Makhijani
5.	Testing Team	Kuber Dhure
6.	Development Team	Devesh Vengurlakar
7.	Technical Lead	Anurag Dubey

4.2.1. Project Management

- Project Manager: reviews the content of the Test Plan, Test Strategy and Test Estimates signs off on it.

4.2.2. Test Planning (Test Lead)

- Ensure entrance criteria are used as input before start the execution.
- Develop test plan and the guidelines to create test conditions, test cases, expected results and execution scripts.
- Provide guidelines on how to manage defects.
- Attend status meetings in person or via the conference call line.
- Communicate to the test team any changes that need to be made to the test deliverables or application and when they will be completed.
- Provide on premise or telecommute support.
- Provide functional (Business Analysts) and technical team to test team personnel (if needed).

4.2.3. Test Team

- Develop test conditions, test cases, expected results, and execution scripts.
- Perform execution and validation.
- Identify, document and prioritize defects according to the guidance provided by the Test lead.
- Re-test after software modifications have been made according to the schedule.
- Prepare testing metrics and provide regular status.

4.2.4. Test Lead

- Acknowledge the completion of a section within a cycle.
- Give the OK to start next level of testing.
- Facilitate defect communications between testing team and technical / development team.

4.2.5. Development Team

- Review testing deliverables (test plan, cases, scripts, expected results, etc.) and provide timely feedback.
- Assist in the validation of results (if requested).
- Support the development and testing processes being used to support the project.

- Certify correct components have been delivered to the test environment at the points specified in the testing schedule.
- Keep project team and leadership informed of potential software delivery date slips based on the current schedule.
- Define processes/tools to facilitate the initial and ongoing migration of components.
- Conduct first line investigation into execution discrepancies and assist test executors in creation of accurate defects.
- Implement fixes to defects according to schedule.

5. TEST ENVIRONMENT

The following mobile operating systems are supported:

- iOS 9.0 to iOS 11.0
- Android 4.3 to 9.0

Hardware Requirements for Mobile Testing

The following hardware is required to support mobile testing:

CPU

- Intel Core i7 2.4 GHz or AMD equivalent

Memory

- Cloud or Remote Mobile Tests: 4GB
- Local Simulators: 8GB

6. APPROVALS

The Names and Titles of all persons who must approve this plan.

Signature:

Name: Vijay Patil
Role: Project Manager
Date: 14/11/21

Signature:
Name: Supriya Kadam
Role: Test Lead
Date: 14/11/19

3. Prepare test plan for any food application with test cases.

Test Plan
Swiggy Version 3.77.1

SWIGGY

FOOD DELIVERY APP

Version: 1.0
Created: 14/11/2021
Last Updated: 14/11/2021

Revision and Signoff Sheet

Document History

Version	Date	Author	Description of Change
1	1/11/2021	Anurag Dubey	Draft
2	1/11/2021	Vinay Savla	Draft
3	12/11/2021	Sahil Makhijani	Draft - Reviewed
4	12/11/2021	Kuber Dhure	Draft - Reviewed

Approvers List

Name	Role	Approver / Reviewer	Approval / Review Date
Vijay Patil	Project Manager		
Supriya Kadam	Test Lead		
Meenakshi Khamkar	Test Lead		
Anurag Dubety	Technical Lead		

Reference Documents -

Version	Date	Document Name
1.0	14/11/21	SWIGGY FOOD DELIVERY APP VERSION 3.77.1

Table of Contents

7. INTRODUCTION

7.1. Purpose

This test plan describes the testing approach and overall framework that will drive the testing of the SWIGGY application. The document introduces:

- Test Strategy: rules the test will be based on, including the givens of the project (e.g.: start / end dates, objectives, assumptions); description of the process to set up a valid test (e.g.: entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
- Execution Strategy: describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
- Test Management: process to handle the logistics of the test and all the events that come up during execution (e.g.: communications, escalation procedures, risk and mitigation, team roster)

7.2. Project Overview

Swiggy Online food delivery application a cross-platform instant food ordering application that allows iPhone, BlackBerry, Android, Windows Phone and Nokia smartphone users to order food from any restaurants online. Swiggy is especially popular with end users for ordering of food and also tacks the food making and delivery process online.

Audience

- Project team members perform tasks specified in this document, and provide input and recommendations on this document.
- Project Manager Plans for the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the task herein specified, approves the document and is accountable for the results.
- The stakeholders' representatives and participants (individuals as identified by the PMO Leads) may take part in the UAT test to ensure the business is aligned with the results of the test.
- Technical Team ensures that the test plan and deliverables are in line with the design, provides the environment for testing and follows the procedures related to the fixes of defects.
- Business analysts will provide their inputs on functional changes.

8. TEST STRATEGY

8.1. Test Objectives

The objective of the test is to verify that the functionality of SWIGGY works according to the specifications.

The test will execute and verify the test scripts, identify, fix and retest all high and medium severity defects per the entrance criteria, prioritize lower severity defects for future fixing.

The final product of the test is twofold:

- A production-ready software;
- A set of stable test scripts that can be reused for Functional and UAT test execution.

8.2. Test Assumptions

Key Assumptions

- Production like data required and be available in the system prior to start of Functional Testing

General

- Exploratory Testing would be carried out once the build is ready for testing
- Performance testing is not considered for this estimation.
- All the defects would come along with a snapshot JPEG format
- The Test Team will be provided with access to Test environment via VPN connectivity
- The Test Team assumes all necessary inputs required during Test design and execution will be supported by Development/BUSINESS ANALYSTs appropriately.
- Test case design activities will be performed by QA Group
- Test environment and preparation activities will be owned by Dev Team
- Dev team will provide Defect fix plans based on the Defect meetings during each cycle to plan. The same will be informed to Test team prior to start of Defect fix cycles
- BUSINESS ANALYST will review and sign-off all Test cases prepared by Test Team prior to start of Test execution
- The defects will be tracked through BUGZILLA only. Any defect fixes planned will be shared with Test Team prior to applying the fixes on the Test environment
- Project Manager/BUSINESS ANALYST will review and sign-off all test deliverables

- The project will provide test planning, test design and test execution support
- Test team will manage the testing effort with close coordination with Project PM/BUSINESS ANALYST
- Project team has the knowledge and experience necessary, or has received adequate training in the system, the project and the testing processes.
- There is no environment downtime during test due to outages or defect fixes.
- The system will be treated as a black box; if the information shows correctly online and in the reports, it will be assumed that the database is working properly.

Functional Testing

- During Functional testing, testing team will use preloaded data which is available on the system at the time of execution
- The Test Team will be perform Functional testing only on SWIGGY

UAT

- UAT test execution will be performed by end users (L1, L2 and L3) and QA Group will provide their support on creating UAT script.

8.3. Test Principles

- Testing will be focused on meeting the business objectives, cost efficiency, and quality.
- There will be common, consistent procedures for all teams supporting testing activities.
- Testing processes will be well defined, yet flexible, with the ability to change as needed.
- Testing activities will build upon previous stages to avoid redundancy or duplication of effort.
- Testing environment and data will emulate a production environment as much as possible.
- Testing will be a repeatable, quantifiable, and measurable activity.
- Testing will be divided into distinct phases, each with clearly defined objectives and goals.
- There will be entrance and exit criteria.

8.4. Data Approach

- In functional testing, SWIGGY will contain pre-loaded test data and which is used for testing activities.

8.5. Scope and Levels of Testing

8.5.1. Exploratory

PURPOSE: the purpose of this test is to make sure critical defects are removed before the next levels of testing can start.

SCOPE: First level navigation, dealer and admin modules

TESTERS: Testing team.

METHOD: this exploratory testing is carried out in the application without any test scripts and documentation

TIMING: at the beginning of each cycle.

8.5.2. Functional Test

PURPOSE: Functional testing will be performed to check the functions of application. The functional testing is carried out by feeding the input and validates the output from the application.

Scope: The below excel sheet details about the scope of Functional test. Note: The scope is high level due to changes in the requirement.

TESTERS: Testing Team.

METHOD: The test will be performed according to Functional scripts, which are stored in Selenium.

TIMING: After Exploratory test is completed.

TEST ACCEPTANCE CRITERIA

5. Approved Functional Specification document, Use case documents must be available prior to start of Test design phase.
6. Test cases approved and signed-off prior to start of Test execution
7. Development completed, unit tested with pass status and results shared to Testing team to avoid duplicate defects
8. Test environment with application installed, configured and ready to use state

Sign-off

- Approved Functional Specification Document
- Approved Use cases
- Approved Test cases

Readiness

- Development completed & unit tested
- Application deployed and system ready for testing on Test environment
- Production like data is available to test all functionalities.
- Defect fixes planned based on Defect triage (Unit Testing) and evaluation criteria

<u>TEST DELIVERABLES</u>

S.No.	Deliverable Name	Author	Reviewer
1.	Test Plan	Test Lead	Project Manager/ Business Analyst's
2.	Functional Test Cases	Test Team	Business Analyst's Sign off
3.	Logging Defects in BUGZILLA	Test Team	Test Lead/ Programming Lead
(4.	Daily/weekly status report	Test Team/ Test Lead	Test Lead/ Project Manager
5.	Test Closure report	Test Lead	Project Manager

<u>MILESTONE LIST</u>

The milestone list is tentative and may change due to below reasons

d) Any issues in the System environment readiness
e) Any change in scope/addition in scope
f) Any other dependency that impacts efforts and timelines

8.5.3. User Acceptance Test (UAT)

<u>PURPOSE</u>: this test focuses on validating the business logic. It allows the end users to complete one final review of the system prior to deployment.

TESTERS: the UAT is performed by the end users (L1, L2 and L3).

METHOD: Since the business users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts. Test team write the UAT test cases based on the inputs from End user (L1,L2 and L3 users) and Business Analyst's.

TIMING: After all other levels of testing (Exploratory and Functional) are done. Only after this test is completed the product can be released to production.

TEST DELIVERABLES

S.No.	Deliverable Name	Author	Reviewer
1.	UAT Test Cases	Test Team	Business Analyst's Sign off

8.6. Test Effort Estimate

This document lists out all the activities that have to be performed by the QA team and estimates how many man-hours each activity is going to take.

New_Detailed DRFT
Test estimate v1.xlsx

9. EXECUTION STRATEGY

9.1. Entry and Exit Criteria

- The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.
- The exit criteria are the desirable conditions that need to be met in order proceed with the implementation.

- Entry and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final “go-no go” decision.
- Entry criteria to start the execution phase of the test: the activities listed in the Test Planning section of the schedule are 100% completed.
- Entry criteria to start each cycle: the activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

Exit Criteria	Test Team	Technical Team	Notes
100% Test Scripts executed			
95% pass rate of Test Scripts			
No open Critical and High severity defects			
95% of Medium severity defects have been closed			
All remaining defects are either cancelled or documented as Change Requests for a future release			
All expected and actual results are captured and documented with the test script			
All test metrics collected based on reports from BUGZILLA			
All defects logged in BUGZILLA			
Test Closure Memo completed and signed off			
Test environment cleanup completed and a new back up of the environment			

9.2. Test Cycles

- There will be two cycles for functional testing. Each cycle will execute all the scripts .
- The objective of the first cycle is to identify any blocking, critical defects, and most of the high defects. It is expected to use some work-around in order to get to all the scripts.
- The objective of the second cycle is to identify remaining high and medium defects, remove the work-around from the first cycle, correct gaps in the scripts and obtain performance results.

- UAT test will consist of one cycle.

9.3. Validation and Defect Management

- It is expected that the testers execute all the scripts in each of the cycles described above. However it is recognized that the testers could also do additional testing if they identify a possible gap in the scripts. This is especially relevant in the second cycle, when the Business analyst's join the TCOE in the execution of the test, since the BUSINESS ANALYSTs have a deeper knowledge of the business processes. If a gap is identified, the scripts and traceability matrix will be updated and then a defect logged against the scripts.
- The defects will be tracked through BUGZILLA only. The technical team will gather information on a daily basis from BUGZILLA, and request additional details from the Defect Coordinator. The technical team will work on fixes.
- It is the responsibility of the tester to open the defects, link them to the corresponding script, assign an initial severity and status, retest and close the defect; it is the responsibility of the Defect Manager to review the severity of the defects and facilitate with the technical team the fix and its implementation, communicate with testers when the test can continue or should be halt, request the tester to retest, and modify status as the defect progresses through the cycle; it is the responsibility of the technical team to review BUGZILLA on a daily basis, ask for details if necessary, fix the defect, communicate to the Defect Manager the fix is done, implement the solution per the Defect Manager request.

Defects found during the Testing will be categorized according to the bug-reporting tool "BUGZILLA" and the categories are:

Severity	Impact
1 (Critical)	▪ This bug is critical enough to crash the system, cause file corruption, or cause potential data loss ▪ It causes an abnormal return to the operating system (crash or a system failure message appears).

	▪ It causes the application to hang and requires re-booting the system.
2 (High)	▪ It causes a lack of vital program functionality with workaround.
3 (Medium)	▪ This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality - for example through another screen. ▪ This bug prevents other areas of the product from being tested. However other areas can be independently tested.
4 (Low)	▪ There is an insufficient or unclear error message, which has minimum impact on product use.
5(Cosmetic)	▪ There is an insufficient or unclear error message that has no impact on product use.

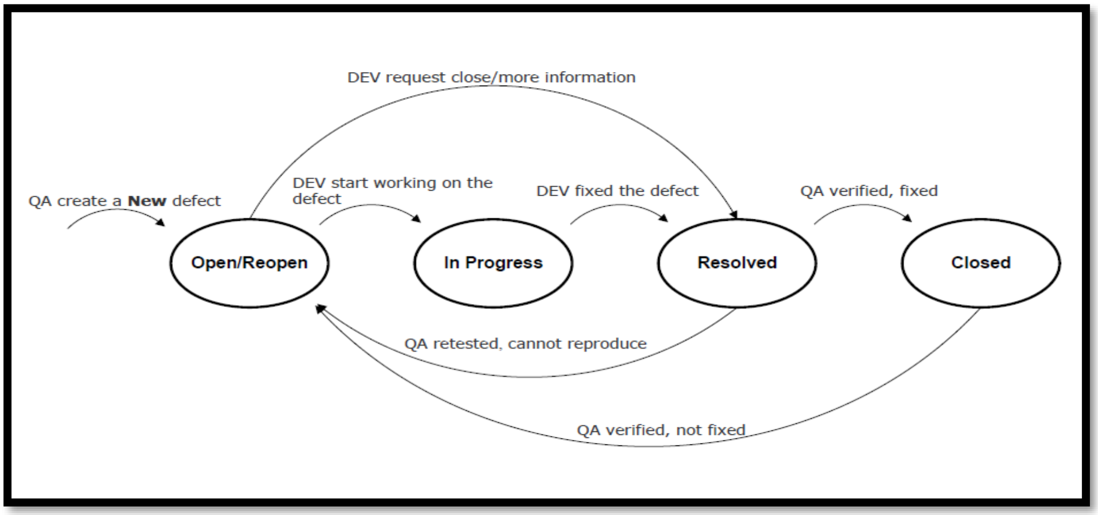
9.4. Test Metrics

Test metrics to measure the progress and level of success of the test will be developed and shared with the project manager for approval. The below are some of the metrics

Report	Description	Frequency
Test preparation & Execution Status	To report on % complete, % Pass, % Fail Defects severity wise Status – Open, closed, any other Status	Weekly / Daily (optional)
Daily execution status	To report on Pass, Fail, Total defects, highlight Showstopper/ Critical defects	Daily
Project Weekly Status report	Project driven reporting (As requested by PM)	Weekly – If project team needs weekly update apart from daily and there is template available with project team to use.

9.5. Defect tracking & Reporting -

Following flowchart depicts Defect Tracking Process:



10. TEST MANAGEMENT PROCESS

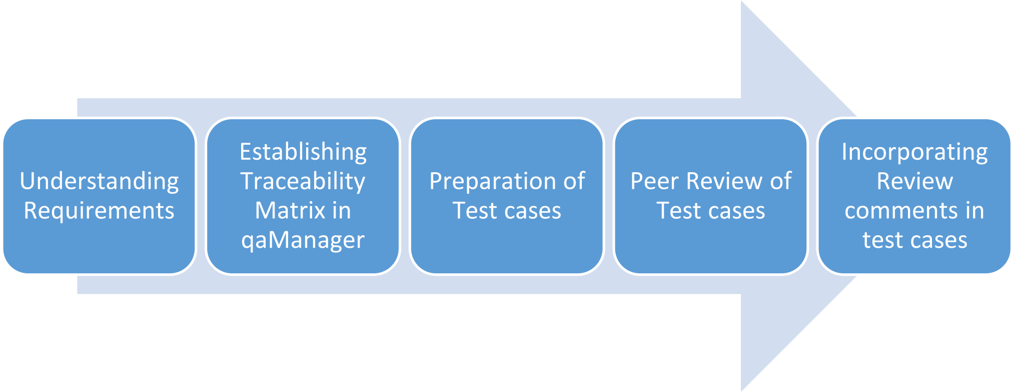
10.1. Test Management Tool

qaManager Lifecycle Management is the tool used for Test Management. All testing artifacts such as Test cases, test results are updated in the qaManager Lifecycle Management.

- Project specific folder structure will be created in qaManager to manage the status of this project.
- Each resource in the Testing team will be provided with Read/Write access to add/modify Test cases in qaManager.
- During the Test Design phase, all test cases are written directly into qaManager. Any change to the test case will be directly updated in the qaManager.
- Each Tester will directly access their respective assigned test cases and update the status of each executed step in qaManager directly.
- Any defect encountered will be raised in BUGZILLA linking to the particular Test case/test step.

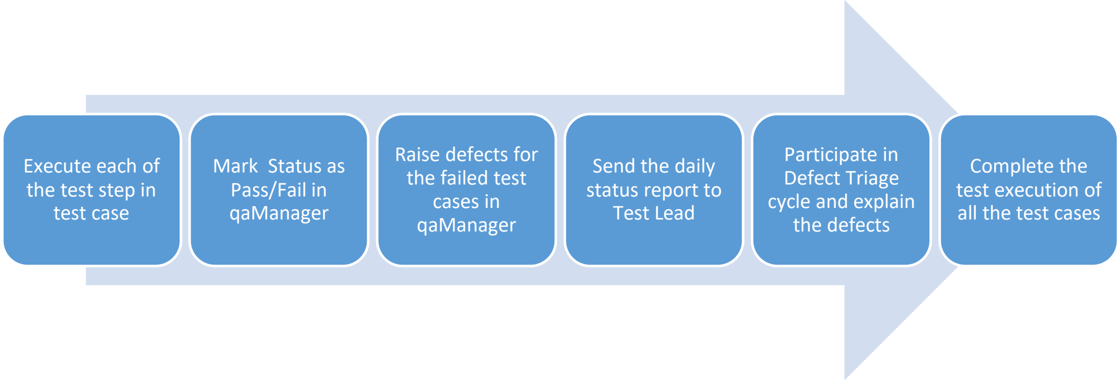
- During Defect fix testing, defects are re-assigned back to the tester to verify the defect fix. The tester verifies the defect fix and updates the status directly in BUGZILLA.
- Various reports can be generated from BUGZILLA to provide status of Test execution. For example, Status report of Test cases executed, Passed, Failed, No. of open defects, Severity wise defects etc.

10.2. Test Design Process


Understanding Requirements
Establishing Traceability Matrix in qaManager
Preparation of Test cases
Peer Review of Test cases
Incorporating Review comments in test cases

- The tester will understand each requirement and prepare corresponding test case to ensure all requirements are covered.
- Each Test case will be mapped to Use cases to Requirements as part of Traceability matrix.
- Each of the Test cases will undergo review by the BUSINESS ANALYST and the review defects are captured and shared to the Test team. The testers will rework on the review defects and finally obtain approval and sign-off.
- During the preparation phase, tester will use the prototype, use case and functional specification to write step by step test cases.
- Testers will maintain a clarification Tracker sheet and same will be shared periodically with the Requirements team and accordingly the test case will be updated. The clarifications may sometimes lead to Change Requests or not in scope or detailing implicit requirements.
- Sign-off for the test cases would be communicates through mail by Business Analyst's.
- Any subsequent changes to the test case if any will be directly updated in qaManager.

10.3. Test Execution Process


Execute each of the test step in test case
Mark Status as Pass/Fail in qaManager
Raise defects for the failed test cases in qaManager
Send the daily status report to Test Lead
Participate in Defect Triage cycle and explain the defects
Complete the test execution of all the test cases

- Once all Test cases are approved and the test environment is ready for testing, tester will start a exploratory test of the application to ensure the application is stable for testing.
- Each Tester is assigned Test cases directly in qaManager.
- Testers to ensure necessary access to the testing environment, BUGZILLA for updating test status and raise defects. If any issues, will be escalated to the Test Lead and in turn to the Project Manager as escalation.
- If any showstopper during exploratory testing will be escalated to the respective development team for fixes.
- Each tester performs step by step execution and updates the executions status. The tester enters Pass or Fail Status for each of the step directly in BUGZILLA.
- Tester will prepare a Run chart with day-wise execution details
- If any failures, defect will be raised as per severity guidelines in BUGZILLA tool detailing steps to simulate along with screenshots if appropriate.
- Daily Test execution status as well as Defect status will be reported to all stakeholders.
- Testing team will participate in defect triage meetings in order to ensure all test cases are executed with either pass/fail category.
- If there are any defects that are not part of steps but could be outside the test steps, such defects need to be captured in BUGZILLA and map it against the test case level or at the specific step that issue was encountered after confirming with Test Lead.
- This process is repeated until all test cases are executed fully with Pass/Fail status.
- During the subsequent cycle, any defects fixed applied will be tested and results will be updated in BUGZILLA during the cycle.

As per Process, final sign-off or project completion process will be followed

10.4. Test Risks and Mitigation Factors

Risk	Prob.	Impact	Mitigation Plan
SCHEDULE Testing schedule is tight. If the start of the testing is delayed due to design tasks, the test cannot be extended beyond the UAT scheduled start date.	High	High	• The testing team can control the preparation tasks (in advance) and the early communication with involved parties. • Some buffer has been added to the schedule for contingencies, although not as much as best practices advise.
RESOURCES Not enough resources, resources on boarding too late (process takes around 15 days.	Mediu m	High	Holidays and vacation have been estimated and built into the schedule; deviations from the estimation could derive in delays in the testing.
DEFECTS Defects are found at a late stage of the cycle or at a late cycle; defects discovered late are most likely be due to unclear specifications and are time consuming to resolve.	Mediu m	High	Defect management plan is in place to ensure prompt communication and fixing of issues.
SCOPE Scope completely defined	Mediu m	Medium	Scope is well defined but the changes are in the functionality are not yet finalized or keep on changing.
Natural disasters	Low	Medium	Teams and responsibilities have been spread to two different geographic areas. In a catastrophic event in one of the areas, there will resources in the other areas needed to continue (although at a slower pace) the testing activities.

Risk	Prob.	Impact	Mitigation Plan
Non-availability of Independent Test environment and accessibility	Medium	High	Due to non availability of the environment, the schedule gets impacted and will lead to delayed start of Test execution.
Delayed Testing Due To new Issues	Medium	High	During testing, there is a good chance that some "new" defects may be identified and may become an issue that will take time to resolve. There are defects that can be raised during testing because of unclear document specification. These defects can yield to an issue that will need time to be resolved. If these issues become showstoppers, it will greatly impact on the overall project schedule. If new defects are discovered, the defect management and issue management procedures are in place to immediately provide a resolution.

10.5. Communications Plan and Team Roster

10.6. Role Expectations

The following list defines in general terms the expectations related to the roles directly involved in the management, planning or execution of the test for the project.

	Roles	Name
1.	Project Manager	Vijay Patil
2.	Test Lead	Vinay Savla
3.	Business Analyst	Arfia Shaikh
4.	Development Lead	Sahil Makhijani

	Roles	Name
5.	Testing Team	Vinay Savla
6.	Development Team	Devesh Vengurlekar
7.	Technical Lead	Anurag Dubey

10.6.1. Project Management

- Project Manager: reviews the content of the Test Plan, Test Strategy and Test Estimates signs off on it.

10.6.2. Test Planning (Test Lead)

- Ensure entrance criteria are used as input before start the execution.
- Develop test plan and the guidelines to create test conditions, test cases, expected results and execution scripts.
- Provide guidelines on how to manage defects.
- Attend status meetings in person or via the conference call line.
- Communicate to the test team any changes that need to be made to the test deliverables or application and when they will be completed.
- Provide on premise or telecommute support.
- Provide functional (Business Analysts) and technical team to test team personnel (if needed).

10.6.3. Test Team

- Develop test conditions, test cases, expected results, and execution scripts.
- Perform execution and validation.
- Identify, document and prioritize defects according to the guidance provided by the Test lead.
- Re-test after software modifications have been made according to the schedule.
- Prepare testing metrics and provide regular status.

10.6.4. Test Lead

- Acknowledge the completion of a section within a cycle.
- Give the OK to start next level of testing.

- Facilitate defect communications between testing team and technical / development team.

10.6.5. Development Team

- Review testing deliverables (test plan, cases, scripts, expected results, etc.) and provide timely feedback.
- Assist in the validation of results (if requested).
- Support the development and testing processes being used to support the project.
- Certify correct components have been delivered to the test environment at the points specified in the testing schedule.
- Keep project team and leadership informed of potential software delivery date slips based on the current schedule.
- Define processes/tools to facilitate the initial and ongoing migration of components.
- Conduct first line investigation into execution discrepancies and assist test executors in creation of accurate defects.
- Implement fixes to defects according to schedule.

11. TEST ENVIRONMENT

The following mobile operating systems are supported:

- iOS 9.0 to iOS 11.0
- Android 4.3 to 9.0

Hardware Requirements for Mobile Testing

The following hardware is required to support mobile testing:

CPU

- Intel Core i7 2.4 GHz or AMD equivalent

Memory

- Cloud or Remote Mobile Tests: 4GB
- Local Simulators: 8GB

12. APPROVALS

The Names and Titles of all persons who must approve this plan.

Signature:
Name: Vijay Patil
Role: Project Manager
Date: 14/11/21

Signature:
Name: Supriya Kadam
Role: Test Lead
Date: 14/11/21

DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	10
Title of Experiment	Generate Defect report for Library Management System.

Practical Related Questions

1. Compare Defect, Error and failure.

   Answer:

Comparison basis	Defect	Error	Failure
Definition	The **Defect** is the difference between the actual outcomes and expected outputs.	An **Error** is a mistake made in the code; that's why we cannot execute or compile code.	If the software has lots of defects, it leads to failure or causes failure.
Raised by	The **Testers** identif y the defect. And it was also solved by the developer in the development phase or stage.	The **Developers and automation test engineers** raise the error.	The failure finds by the manual test engineer through the **development cycle**.

Different types	Different type of Defects are as follows: Based on **priority**: ○ High ○ Medium ○ Low And based on the severity: ○ Critical ○ Major ○ Minor ○ Trivial	Different type of Error is as below: ○ Syntactic Error ○ User interface error ○ Flow control error ○ Error handling error ○ Calculation error ○ Hardware error ○ Testing Error	-----
Reasons behind	The below reason leads to the **defects**: Giving incorrect and wrong inputs. Dilemmas and errors in the outside behavior and inside structure and design. An error in coding or logic affects the	The reasons for having an **error** are as follows: Errors in the code. The Mistake of some values. If a developer is unable **to compile or run a program successfully.**	Following are some of the most important reasons behind the **failure:** Environmental condition System usage Users Human error

	software and causes it to breakdown or the failure.	Confusions and issues in programming. Invalid login, loop, and syntax. Inconsistency between actual and expected outcomes. Blunders in design or requirement actions. Misperception in understanding the requirements of the application.	
Way to prevent the reasons	With the help of the following, we can prevent the **Defects**: Implementing several innovative programming methods. Use of primary and correct software development techniques. Peer review	Below are ways to prevent the **Errors**: Enhance the software quality with system review and programming. Detect the issues and prepare a suitable mitigation plan. Validate the	The way to prevent **failure** are as follows: Confirm re-testing. Review the requirements and revisit the specifications. Implement current protective techniques. Categorize and

	It is executing consistent code reviews to evaluate its quality and correctness.	fixes and verify their quality and precision.	evaluate errors and issues.

2. Describe different status of Defects.

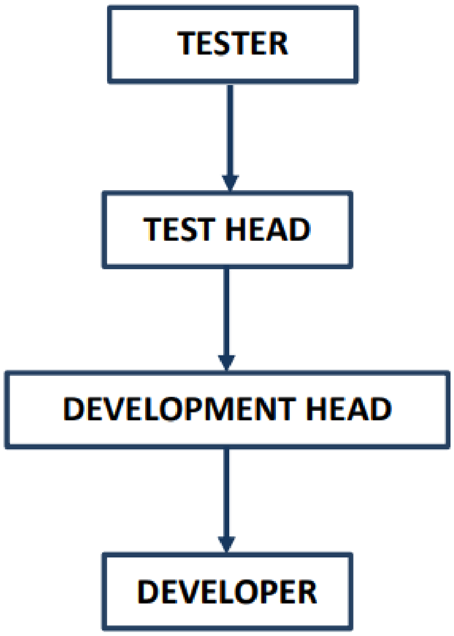
Answer:

- **New:** When a new defect is logged and posted for the first time. It is assigned a status as NEW.
- **Assigned:** Once the bug is posted by the tester, the lead of the tester approves the bug and assigns the bug to the developer team
- **Open:** The developer starts analyzing and works on the defect fix
- **Fixed:** When a developer makes a necessary code change and verifies the change, he or she can make bug status as "Fixed."
- **Pending retest:** Once the defect is fixed the developer gives a particular code for retesting the code to the tester. Since the software testing remains pending from the testers end, the status assigned is "pending retest."
- **Verified:** The tester re-tests the bug after it got fixed by the developer. If there is no bug detected in the software, then the bug is fixed and the status assigned is "verified."
- **Reopen:** If the bug persists even after the developer has fixed the bug, the tester changes the status to "reopened". Once again the bug goes through the life cycle.
- **Closed:** If the bug is no longer exists then tester assigns the status "Closed."
- **Duplicate:** If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to "duplicate."
- **Rejected:** If the developer feels the defect is not a genuine defect then it changes the defect to "rejected."

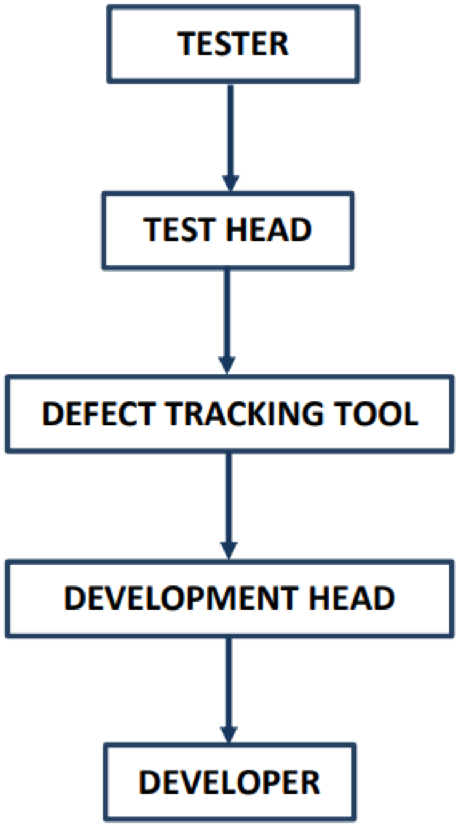
- **Deferred:** If the present bug is not of a prime priority and if it is expected to get fixed in the next release, then status "Deferred" is assigned to such bugs
- **Not a bug:** If it does not affect the functionality of the application then the status assigned to a bug is "Not a bug".

3. Describe various defects reporting process with example.

Answer: Defect reporting process in Medium Scale Organization is:



Defect reporting process in Large Scale Organization is:


TESTER
TEST HEAD
DEFECT TRACKING TOOL
DEVELOPMENT HEAD
DEVELOPER

Exercise

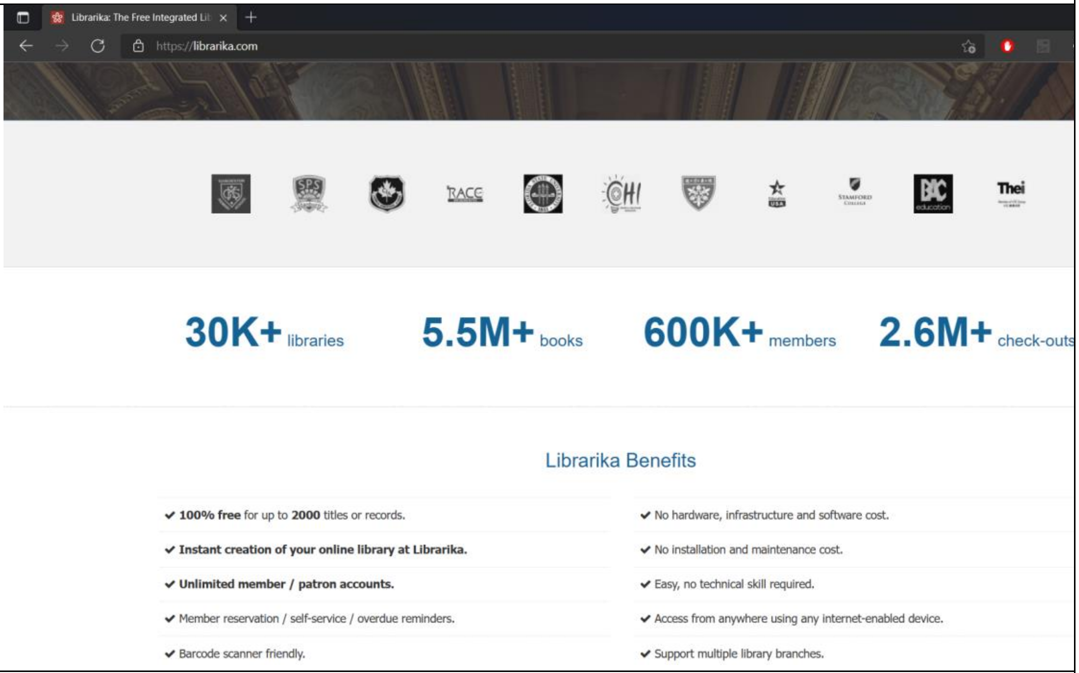
1. Generate online library management system with the help of following website. Perform at least 4 tests for same. Prepare defect report for library management system
   https://librarika.com/

Test Case	Test Case Name	Actual Input	Expected Output	Actual Output	Status
TC_01	Home > Features	Click on Features	It should redirect to features page	It redirects to features page	Pass
TC_02	Home > Plans	Click on Plans	It should redirect to Plans page	t redirects to Plans page	Pass
TC_03	Home > Upgrade	Click on Upgrade	It should redirect to Upgrade page	It redirects to Upgrade page	Pass
TC_04	Home > Articles	lick on Articles	It should redirect to Articles page	It redirects to Articles page	Pass
TC_05	Home > Discover	Click on Discover	It should redirect to Discover page	It redirects to Discover pag	Pass
TC_06	Home > FAQ's	Click on FAQ's	It should redirect to FAQ's page	It redirects to FAQ's page	Pass
TC_07	Home > Create Free Library	Click on Create Free Library	It should redirect to Create New Library page	It redirects to Create New Library page	Pass
TC_08	Home > Create Free Library	Library Nickname = "VPLib"	It should accept the data	It accepts the data	Pass
TC_09	Home > Create Free Library	Library Name = "Vidyalankar Polytechnic Library"	It should accept the data	It accepts the data	Pass

TC_10	Home > Create Free Library	Library Type = "College library"	It should accept the data	It accepts the data	Pass
TC_11	Home > Create Free Library	Library Library Email = "vplib056@gmail.com"	It should accept the data	It accepts the data	Pass
TC_12	Home > Create Free Library	Library Password = "vplib@03"	It should accept the data	It accepts the data	Pass
TC_13	Home > Create Free Library	Confirm Password = "vplib@03"	It should accept the data	It accepts the data	Pass
TC_14	Home > Create Free Library	Country = "India"	It should accept the data	It accepts the data	Pass
TC_15	Home > Create Free Library	Click on I agree, please create my library	It should create a Library	It creates a Librar	Pass

Defect Report:

ID	Def_01
Project	Librarika
Product	https://librarika.com/
Release Version	v1.0
Module	Home Page>Image in div
Detected Build Version	V1.1
Summary	First image not found
Description	The First image in the sequence is not found.
Steps to Replicate	1) Open the website 2) Scroll down till the parallax ends
Actual Results	It does not display the logo of a particular college.

Expected Results	It should display the image of some college logo.
Attachments	 30K+ libraries 5.5M+ books 600K+ members 2.6M+ check-outs Librarika Benefits ✓ **100% free** for up to **2000** titles or records. ✓ **Instant creation of your online library at Librarika.** ✓ **Unlimited member / patron accounts.** ✓ Member reservation / self-service / overdue reminders. ✓ Barcode scanner friendly. ✓ No hardware, infrastructure and software cost. ✓ No installation and maintenance cost. ✓ Easy, no technical skill required. ✓ Access from anywhere using any internet-enabled device. ✓ Support multiple library branches.
Remarks	Causes inconvenience to the user.
Defect Severity	Low
Defect Priority	Low
Reported By	Anurag Dubey
Assigned To	Supriya Angne
Status	Assigned

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	11
Title of Experiment	Validating Defect report for ATM Machine.

Practical Related Questions

1. Define Defect Deduction Percentage.

   Answer: Defect Deduction Percentage: The number of defects found by a test phase, divided by the number found by that test phase and any other means afterwards.

2. Describe Defect Reporting Process.

Answer:

- Defect Reporting in software testing is a process in which test managers prepare and send the defect report to the management team for feedback on defect management process and defects' status.
- Then the management team checks the defect report and sends feedback or provides further support if needed.
- Defect reporting helps to better communicate, track and explain defects in detail.
- The management board has right to know the defect status.
- They must understand the defect management process to support you in this project.
- Therefore, you must report them the current defect situation to get feedback from them.

3. Describe Defect Report Template.

Answer:

ID	Unique identifier given to the defect. (Usually Automated)
Project	Project name.
Product	Product name.
Release Version	Release version of the product. (e.g. 1.2.3)
Module	Specific module of the product where the defect was detected.
Detected Build Version	Build version of the product where the defect was detected (e.g. 1.2.3.5)
Summary	Summary of the defect. Keep this clear and concise
Description	Detailed description of the defect. Describe as much as possible but without repeating anything or using complex words. Keep it simple but comprehensive.
Steps to Replicate	Step by step description of the way to reproduce the defect. Number the steps.
Actual Result	The actual result you received when you followed the steps.
Expected Results	The expected results.
Attachments	Attach any additional information like screenshots and logs.
Remarks	Any additional comments on the defect.

ID	Unique identifier given to the defect. (Usually Automated)
Defect Severity	Severity of the Defect.
Defect Priority	Priority of the Defect.
Reported By	The name of the person who reported the defect.
Assigned To	The name of the person that is assigned to analyze/fix the defect.
Status	The
Fixed Build Version	Build version of the product where the defect was fixed (e.g. 1.2.3.9)

Exercise

1. Use ATM machine system simulator with the help of following website. Perform at given task and generate test case. Prepare defect report for ATM machine system.

   Task

   i. Enter Pin number of 4 digits.
   ii. Select Amount to be withdrawn.
   iii. In other functions, Check for Balance on Screen
   iv. In other functions, Print Mini Statement
   v. Verify functionality of Change Pin number.
   vi. Verify withdrawal of amount excess of 5000.

   https://www.motc.gov.qa/en/ditoolkit/migrant-workers/cash-machinesimulator-atm

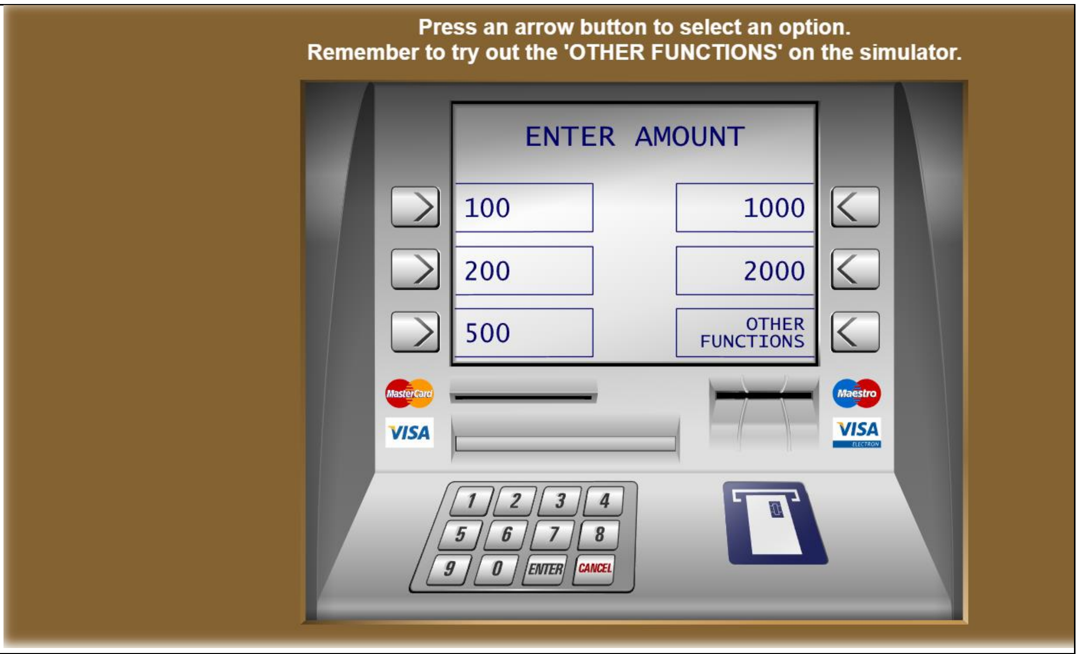
Test Case:

Test Case	Test Case Name	Actual Input	Expected Output	Actual Output	Status
TC_01	Pin Number	Enter 4 digit pin-1234	It should accept the pin and proceed to the main page.	It accepts the pin and proceeds to the main page	Pass
TC_02	Withdraw	Select amount - 2000	It should withdraw the money	It is withdrawing the money	Pass
TC_03	Balance on screen	Select Balance on screen in other functions screen	It should display the balance on screen balance on screen	It is displaying the balance on screen	Pass
TC_04	Mini Statement	Select mini statement in Other functions	It should print the mini statement	It is printing the mini statement	Pass

TC_05	Change Pin	Select Change Pin in other functions	It should ask for the new pin and change it with the old one.	It is asking for the new pin and changed It with the old one.	Pass
TC_06	Withdrawal	Select Cash withdrawal option	It should allow the user to select denomination of 5000.	It is not displaying the option to select denomination of 5000.	Pass

Defect Report

ID	Def_01
Project	Cash Simulator Cash (ATM)
Product	https://www.motc.gov.qa/en/ditoolkit/migrant-workers/cash-machine-simulator-atm
Release Version	v1.0
Module	Home Page> Our Programs > Digital Inclusion tools
Detected Build Version	V1.1
Summary	Limited denomination options in cash withdrawal function, restricting cash withdrawal only till 3000.
Description	No option of withdrawing of amount excess of 3000
Steps to Replicate	1) Open the website 2) Select our programs 3) Proceed to Digital Inclusion tools and select cash machine simulator (ATM) 4) Select language and skip to simulator 5) Enter the card 6) Select the account type 7) Go to Other functions and select cash withdrawal
Actual Results	It has displaying limited options of denominations in cash withdrawal option.
Expected Results	It should add more options in denominations in withdrawal function or it should take amount input from the user.

Attachments	 Press an arrow button to select an option. Remember to try out the 'OTHER FUNCTIONS' on the simulator. ENTER AMOUNT 100 1000 200 2000 500 OTHER FUNCTIONS MasterCard VISA Maestro VISA ELECTRON 1 2 3 4 5 6 7 8 9 0 ENTER CANCEL
Remarks	Causes inconvenience to the user in terms of limited cash withdrawal options.
Defect Severity	High
Defect Priority	High
Reported By	Anurag Dubey
Assigned To	Supriya Angne
Status	Assigned

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	12
Title of Experiment	Executing Test Case to Generate Defect Report for Login.

Practical Related Questions

1. What is the significance of encryption while saving password in database?

   Answer:

   - Encryption scrambles your password so it's unreadable and/or unusable by hackers.
   - That simple step protects your password while it's sitting in a database, and it offers more protection as your password zooms across the internet.

2. Give the significance of POST method while performing operations on web based

   Answer:

   - POST is a request method supported by HTTP used by the World Wide Web.
   - By design, the POST request method requests that a web server accept the data enclosed in the body of the request message, most likely for storing it.
   - It is often used when uploading a file or when submitting a completed web form.

3. Define step to write a defect report for given application

   Answer: `

   1. State the platform version.
   2. Login and examine data.
   3. List any non-standard configuration setting with values.
   4. State the workflow direction through application.
   5. Research to narrow down the root cause and get beyond the defect symptom:
      - Define who, what, where, when and how you got found the defect.
      - Scan error logs.
      - Execute database queries.
   6. Add support documentation:
      - Screenshots.
      - Stop recording files.
      - Video.
      - Database query results.
   7. Format all texts for reliability of easy understanding. Stick to known familiar format.

Exercise:

1. Use Login system simulator with the help of following website. Perform given task and generate test case. Prepare defect report for Login system.
   - Task
     1. Enter invalid user name.
     2. Enter invalid password.
     3. Enter password with only 3 characters.
     4. Enter user name as "invitado" and password as "hgm2015".
     5. https://codepen.io/opensoorce/pen/KQmvd

Test Case	Test Case Name	Actual Input	Expected Output	Acutal Output	Status
TC_01	To enter invalid username	Enter username as “abcdefg”	It should not accept invalid username.	It does not accept username and throws a message saying “Please enter correct username and password”	Pass
TC_02	To enter invalid password	Enter password as “pass@45”	It should not accept invalid password.	It does not accept the password and throws message saying “Please enter correct username and password”.	Pass
TC_03	To enter password with three characters only	Enter password with three characters as “abc”	It should not accept three character invalid password.	It does not accept password and throws message saying “Please enter correct username and password”.	Pass
TC_04	To test whether it accepts valid	Enter username as “invitado” and password	It should accept valid usernameand password	It accepts valid username and password and throws message	Pass

	username and password	as "hgm2015"		saying "Valid please wait"	

2. Consider any web base system which provides login procedure. Perform following tests for same. Prepare defect report for Login System.
   - Task
     1. Verify forgot password link.
     2. Test user name as "STEPR" and password as "STEPR".
   - Verify captcha for given system

Test case for any GitHub Sign up

Test Case	Test Case Name	Actual Input	Expected Output	Acutal Output	Status
TC_01	To test forgot password link	Clink on forgot password link and enter registered email.	It should send forgot password link to registered email address which will allow you to create a new password	Forgot password link is received and on clicking the link leads you to a site where you enter your new password and on submitting password is changed.	Pass
TC_02	Functionality of close button	Checks functionality of close button.	It should check functionality of close button.	It is checking functionality of close button.	Pass
TC_03	To test username as "STEPR" and password as "STEPR"	In username enter"STEPR" and password as"STEPR"	It should not accept username and password.	It shows an message saying "password length should	Pass

				be 6 to 126 characters”	
TC_04	To verfiy the captcha for the given system	Displayed captcha is “5Bn35” and entered is “5Bn30”	It should not accept wrong captcha	It does not accept the captcha and it shows the message as “invalid captcha” and a new captcha is now placed	Pass
TC_05	To verfiy the captcha for the given system	Displayed captcha is “2A031” and entered is “2A031”	It should not accept wrong captcha	It accepts captcha and it proceeds further to next details.	Pass

Vidyalankar Polytechnic

DEPARTMENT OF COMPUTER ENGINEERING

Subject: Software Testing	Subject Code: 22518
Semester: 5	Course: CO5I-A
Laboratory No:	Name of Subject Teacher: Supriya Angne
Name of Student: Anurag Dubey	Roll Id: 19203A0013

Experiment No:	14
Title of Experiment	Design and run test cases for MS Word application using an automation tool

Practical Related Questions

1. Compare any two software testing tools used in automation testing.

   Answer:

Features	**Katalon Studio**	**Selenium**	**TestComplete**	**UFT (QTP)**
Test development platform	Cross-platform	Cross-platform	Windows	Windows
Application under test	Web and mobile apps	Web apps	Windows desktop, web, mobile apps	Windows desktop, web, mobile apps
Scripting languages	Java/Groovy	Java, C#, Perl, Python, JavaScript, Ruby, PHP	JavaScript, Python, VBScript, JScript, Delphi, C++, and C#	VBScript
Programming skills	Not required.	Advanced skills needed	Not required. Recommended	Not required. Recommended

	Recommended for advanced test scripts	to integrate various tools	for advanced test scripts	for advanced test scripts	
Learning curves	Medium	High	Medium	Medium	
Ease of installation and use	Easy to set up and run	Require installing and integrating various tools	Easy to setup and run	Easy to setup and run	
Script creation time	Quick	Slow	Quick	Quick	
Object storage and maintenance	Built-in object repository, XPath, object re-identification	XPath, UI Maps	Built-in object repository, detecting common objects	Built-in object repository, smart object detection and correction	
Image-based testing	Built-in support	Require installing additional libraries	Built-in support	Built-in support, image-based object recognition	
Continuous integrations	Popular CI tools (e.g. Jenkins, Teamcity)	Various CI tools (e.g. Jenkins, Cruise Control)	Various CI tools (e.g. Jenkins, HP Quality Center)	Various CI tools (e.g. Jenkins, HP Quality Center)	

Product support	Ticketing support, community	Open source community	Dedicated staff, community	Dedicated staff, community	
License type	Freeware	Open source (Apache 2.0)	Proprietary	Proprietary	
Cost	Free	Free	License and maintenance fees	License and maintenance fees	

2. List some advantages and disadvantages of Automation testing.

Answer:

Advantages of Automated Testing:

1. Automated testing improves the coverage of testing as automated execution of test cases is faster than manual execution.
2. Automated testing reduces the dependability of testing on the availability of the test engineers.
3. Automated testing provides round the clock coverage as automated tests can be run all time in 24*7 environment.
4. Automated testing takes far less resources in execution as compared to manual testing.
5. It helps to train the test engineers to increase their knowledge by producing a repository of different tests.

Disadvantages of Automated Testing:

1. Automated testing is very much expensive than the manual testing.
2. It also becomes inconvenient and burdensome as to decide who would automate and who would train.
3. It has limited to some organisations as many organisations not prefer test automation.
4. Automated testing would also require additionally trained and skilled people.

5. Automated testing only removes the mechanical execution of testing process, but creation of test cases still required testing professionals.

3. Enlist the different factors which are determining the effectiveness of Automation testing.

   Answer:

   1. Build a dedicated team

   Test automation is a dedicated and focused activity to consider, and cannot be mixed. So, it is important to bring a dedicated team on board for the same.

   2. Selecting the tool

   It is important to select a tool that is compatible with the organization and the people who would be involved in the process.

   3. Finding the right tool is just the beginning

   The tool cannot be implemented everywhere, as it might not cover every scenario. But if blended with the right strategy and skill sets, the tool will provide the expected results.

   4. Blend every aspect together for desired results

   It is advisable to begin the automation process by creating the test case in a manual format - collect all the requirements and testing data to build the automation plan.

   5. Know your software/application

   This will help implement the tool effectively and enable the right third-party integrations. This will ensure effective automation even in the future. Moreover, it will help identify any possible defects, memory leaks, performance issues, scalability issues, and more.

   6. Automation cannot be done for everything

   You will have to loop in and recruit the right processes that can meet the desired goals.

4. State some good coding practices while automation.

   Answer: `

   1. Always use test design patterns and principles
   2. Never use Thread.sleep() unless there are specific test requirements
   3. Do not run ALL tests across ALL target browsers
   4. Separate your tests from your test automation framework
   5. Make your test automation framework portable
   6. Name your tests wisely
   7. Use soft assertions if you need to make a list of related checks on the same web page
   8. Take screenshots for failure investigation
   9. Make tests simpler instead of adding comments
   10. Follow the “green tests run” policy
   11. Use data-driven instead of repeated tests
   12. All tests should be independent
   13. Setup detailed automation tests reporting