

Objective : Select any System, Such as ATM System and thoroughly examine its System specifications to identify and document the different Software defects present.

common Software defects in ATM System :

- (1) Validity of the card
- (2) Withdraw Transaction flow of ATM
- (3) Authentication of the user's
- (4) Dispense the cash from the account

Bug Report :

Bug Id : ATM-001

Bug Description : Invalid card

Steps to Reproduce : Keep valid card in ATM

Expected : Welcome Screen

Actual Result : Invalid card

Status : Pass/fail

Bug ID : ATM-002 Bug Description : Invalid Pin entered

Steps to Reproduce : 1. Keep valid card in ATM

2. Enter Authorized PIN

3. Menu Screen Should displayed

Expected Result : Menu Screen Should displayed

Actual Result : Invalid PIN Screen displayed

Status : Pass/Fail

Bug ID : ATM-003

Bug Description : Invalid account type Selected

Steps to Reproduce : Enter a Valid PIN number

Select the withdraw in menu

choose the correct type of
account (Savings/Current)

Expected Result : Enter amount Screen displayed

Actual Result : Invalid account type Screen displayed

Status : Pass/Fail

Bug Id : ATM-005

Bug Description : Insufficient balance

Steps to reproduce : 1. Menu Screen should be displayed

2. Select Withdraw option

3. Select correct type of account

4. Enter Sufficient amount to withdraw

Expected Result : Collect the amount Screen display

Actual Result : Insufficient balance in account

Status : Pass/Fail

Bug Id : ATM-005

Bug Description : Amount enter denominations

Steps to reproduce : keep a Valid ATM card

Enter authorized Pin

Enter the amount which should Multiplies 100

Cash Dispenser screen display

Expected Result : Collect the amount is display

Actual Result : Amount enter not in required denominations.

Status : Pass/Fail

case Study on ARMS
Portal - Positive Test Cases

Objective: Requirements analysis and Testing
for ARMS Portal

Item to be tested	Features	Approach	Test Environment	Training needed	Risk	Test cases	Pass/Fail
1 Home	Holidays Notification Attendance confirmation	Manual approach	Operating System Browser	Not Required	No	open the website arms. Saveetba.com Enter username and Password check holidays	Pass
2 My course	Progress course completed course External credits	Manual approach	Operating System Browser	Not required	No	open arms. Saveetba.com Enter user name Password click on my click on my course	Pass
3 Attendance	Report course code course name	Manual approach	Browser	Not required	No	click on button and go to attendence	

4	Assignment * Course Select * Completed	Manual approach	Browsers Not required	Click on = button and go to assignment Pass
5	Examination * No due Payment and history financial second Point Payment	Manual approach	Browsers not required	Click on = Button and go Examination and fee pop up the window Then go to fee financial record and check fee Payments.

⑥

Exp: 3

Case Study on ARMS Portal

Negative Test cases

Objective : Case study to identify and analyse the requirements of ARMS Portal of Negative cases.

Sno	Features	Approach	Data test	Test Environment	Training needed	Risk	Test cases	Expected output	Pass/Fail
1	Holidays notification attendance Home conformation	Manual user operating system approach	user name & Password	Browser	Not required	No	open arms portal Enter username and Password and entered wrong	username and Pass word invalid	Pass
2	In Progress My course graduation course status External credit status	user approach	user name & Password	Browser required	Not required	No	click on = button Go to my course wasn't check whether appeared	My course is there or not	Pass

		Manual user approach	Not required	No	click on = Button go to enrollment	Pass
3	Select slot A and B	user name & Password	Not required	No	click on = Button go to enrollment	Pass
4	completed assignment content from Exam	user name & Password	Not required	No	click on = button go on assignment Select Course opening assignment	Pass
5	No due request Exam Please in a Popup window enable	user name & Password	Not required	No	click on = button click on examination go to nodue request whatever it is there are not	Pass

(8)

Exp No: 4 Case Study on e-commerce

Applications - Positive Test cases

Objective : Case study to identify and analyze the requirements of flipkart positive Test case

Item tested	Features Approach	Data Test Environment	Training needed	Risk	Test Case	Pass / Fail
→ To me ton Shop, notification, Account, Cart	Search by Manual approach	laptop, Android	No	No	open the flip cart click on email, Password click on home Page	Pass
→ Search for Product brands and shirts	Search by Manual approach	laptop, Android	No	No	Open the flipcart and click on the home page the search button appear click on the Search button and find the product you want to Search	Pass

3) Shops Trending Stores	Offerzone Manual	No	daptop Android	No	No	open the flipkart and click on home pages and then click on shops.
4) Notifications offers approach	All	Manual	No	daptop Android	No	click on home page and click on notifications we can find the any offers or getting to buy free product
5) cart	My cart, Address, approach	Manual	No	daptop Android	No	click on flipkart and on home page and then cart click the product and place the order in my cart.
	Product, Order Place					

⑤ Capstone Case Study on e-commerce

Applications Negative Test cases

Objective : To write a case study to identify and analyse the e-commerce Applications on negative Test cases.

Items	Feature	Apporach	Data Test	Test Environment needed	Risk	Test Case	Expected Output	Pass
1) Home	Search Button, shops, notifications	Manual	daptop, Android	No	No	open the flipkart click on email, Password	clear tab open Incorrect name and Password	Pass
2)	Search Products	Search for Manual products approach	daptop	No	No	Search for products	Wrong product appear	Pass
3)	Shops	Search for Manual mobiles approach	daptop	No	No	open the flipkart click on the mobiles	do not appear	Pass

4) Send ing for Student stores and wedding store	Search Manual approach	No Android	No No	open tree shops In-heat open trending stores But it will open another store	Not open student's club
5) Noti fici ati ons	Order list Manual list/ approach	No Android	No No	open tree notification placed tree product Doesn't appear notifications	-Help center doesn't appear in tree screen to visible

⑫

Installation of Eclipse Software

The steps to be followed to install the Eclipse Software

(1) Download Eclipse :

- * Go to the official Eclipse download Page
- * choose the Eclipse IDE for Java Developers Package
- * Download the installer for your operating System

(2) Run the Installer :

- * Execute the downloaded installer
- * Follow the on-screen instructions to set up

(3) Select IDE Packages :

- * During installation, choose the package that includes features for Java development and testing such as Eclipse IDE for Java Developers

(4) Configure Installation :

- * Specify the installation directory and any other configuration setting as needed.

(5) Install Eclipse:

- * Allow the installer to download and install Eclipse based on your selected configuration.

(6) Launch Eclipse:

- * Once the installation is complete launch Eclipse.

(7) Set Workspace:

- * Choose a workspace where your projects will be stored.

(8) Install Plugins for Testing:

- * If your testing involves specific tools or frameworks, install relevant plugins.

(9) Create or Import Projects:

- * Create a new project or import existing projects into Eclipse.

(10) Start Testing:

- * With Eclipse configured and projects set up, you can start creating and running tests.

Junit Execution Procedure

It involves following steps:

(1) Writing Test cases:

* Create test classes containing Methods annotated with "@Test"

(2) Compile code and Test:

* Compile your Source code and test classes.

(3) Run Tests:

* Execute tests using a Junit runner, such as "JUnitCore" or IDE's built-in test runner.

(4) Stepup and Teardown:

* Utilize "@Before" and "@After" annotations for each Stepup and Teardown.

(5) Assertions:

* Use various assertions methods like "assert Equals", "assertTrue".

(6) Annotations:

* Leverage annotations like "@Ignore" to skip specific tests like "@Runnlite".

(7) Test Suites :

- * Group related test classes into test Suites using "@RunWith(Suite.class)"

(8) Parameterized Tests :

- * Employ "@Parameterized" for running the same test with multiple inputs

(9) Reporting :

- * JUnit generates test reports indicating Passed or failed tests

(10) Integration with Build Tools :

- * Integrate JUnit into build tools like Maven or Gradle for seamless execution as part of the build process.