Al Hussein Technical University

جامعة الحسين التقنية

Graduation Project

Title: Al-Mosafer Website

Students:

Farah Mango

Ahmad AboShanab

Osama Zoubi

Supervisor: AbedalRaheem AlSaqqa

Acknowledgments

We would like to express our sincere gratitude and appreciation to Al Hussein technical university for providing us with the opportunity to complete the Quality Assurance course successfully. This achievement would not have been possible without the support and guidance of many individuals.

First and foremost, we are deeply grateful to our instructors who imparted their knowledge and expertise throughout the duration of the course. Their dedication, enthusiasm, and commitment to excellence have been instrumental in shaping our understanding of the subject matter and fostering our intellectual growth.

We would also like to extend our heartfelt thanks to the administrative staff and support personnel at Al Hussein technical university. Their tireless efforts in managing the logistics, organizing resources, and ensuring a conducive learning environment have significantly contributed to our positive learning experience.

In conclusion, this accomplishment marks a significant milestone in our life, and We are excited to apply the knowledge and skills We have acquired as We embark on new opportunities and endeavors.

Thank you once again for your unwavering support and belief in our abilities**.**

Abstract

The main objective of this document is to provide the readers with a brief about the Almosafer website and its objectives.

And it describes the test cases that we have done on this website to validate the website, the documentation also explores the tools and technologies used to solve the project and the methodology that we used.
We also made a test report, and performance testing on this website using JMeter.

It provides a comprehensive overview of the Quality Assurance Course, offering insights into the key topics, and skills that were covered throughout the training.

About Us

❖ Farah Mango

Phone: 0777684302
Email: farahmangoo@gmail.com
GitHub: farahmango
LinkedIn:

Figure 1: Farah's LinkedIn

I Graduated from the World Islamic Sciences and education university and majored in Software Engineering in 2021.
I entered the coding academy by Orange for seven months to learn more about programming languages, I have learned about HTML, CSS, JavaScript, PHP Laravel, and MySQL databases.
Then I was employed for one year as a PHP developer.

I have one year of experience in software development, I worked as a developer to increase my skills in coding and logical thinking before I entered quality assurance.
I took many courses in quality assurance like Automation Selenium using Java, and Quality Assurance manual ISTQB and then entered this quality assurance course at HTU to be ready for the job market.

I am skilled in using SDLC and STLC terms, performance testing using JMeter, and selenium using Java, I am also skilled in API testing using Postman.

Overall, I am passionate about quality assurance and take pride in my ability to ensure that all products and services are of the highest quality. I am excited to bring my skills and experience to a new role and make a valuable contribution to a dynamic team.

❖ Ahmad AbuShanab

Phone: 0795096405
Email: abushanab323@gmail.com
GitHub: Ahmmad Abu Shanab
LinkedIn:

Figure 2: Ahmad LinkedIn

I'm Ahmad Abushanab, I graduated from Hashemite University and majored in business information technology.

During my university years, I've learned a lot of skills, some of them are soft skills like teamwork and communication, and some of them are technical skills like OOP and data-structure database algorithms.
Also, I've participated in some competitions like the hult prize and Amazon stockholder.
Also, I have developed an application for my graduation project using Android (kotlin , firebase).
Also, I've trained in JEPCO at the Information Technology Infrastructure department, and I've completed an up-skilling program in automation quality assurance at HTU.

❖ Osama Zoubi

Phone: 0796780572
Email: osamayahia777@gmail.com
GitHub: Osama Alzoubi
LinkedIn:

Figure 3: Osama LinkedIn

Graduated from Yarmouk University and majored in Computer Information Systems In 2021.

During my years of education, I learned the basics of different programming languages such as C++, C#, Oracle PL, Python, PHP, and SQL. I also created several projects during my years of study like desktop applications using Oracle forms and reports, and a Windows application using c#, also I created a website for my graduation project using PHP, Bootstrap, HTML, CSS, and Apache server. After my graduation, I worked for 6 months in the Ministry of Digital Economy and Entrepreneurship on Digital Identity Project (Sanad Application) As a guide and activator of the digital identity at Sanad stations. After my work at the Ministry of Digital Economy and Entrepreneurship ended, I applied for a training course in Quality Assurance at HTU.

Table of Content

List of Figures

List of Abbreviations

• Software development life cycle	SDLC	IV,16
• Software testing life cycle	STLC	IV,16
• Object oriented programming	OOP	V
• Hypertext markup language	HTML	IV, V
• Cascading style sheet	CSS	IV, V
• Structured Query language	SQL	V
• Programming language	PL	V
• Hussein technical university	HTU	IV, V
• Jordan Electric Power Company	JEPCO	V
• Integrated development environment	IDE	5
• International software qualifications board	ISTQB	IV, 20 ,27
• Application programming interface	API	IV, 20
• Hypertext transfer protocol	HTTP	20
• Test Next Generation	TestNG	21
• Quality assurance	QA	16
• Quality control	QC	16
• The name of co-founder Michael Widenius's daughter My, and "SQL", the acronym for Structured Query Language.	MySQL	IV
• iPhone operating system	IOS	21
• Return on investment	ROI	21

Chapter 1

INTRODUCTION

Introduction

Al Mosafer is a leading travel website that offers a convenient and comprehensive platform for individuals to plan, book, and manage their travel experiences. With its user-friendly interface and extensive range of services, Al Mosafer aims to make travel planning and booking a seamless and enjoyable process for users worldwide.

At Al Mosafer, users can easily search for and compare flights, hotels, rental cars, and other travel services, allowing them to find the best options that suit their preferences and budget. The website provides detailed information about various destinations, attractions, and travel tips, empowering users to make informed decisions about their travel plans.

One of the key strengths of Al Mosafer is its commitment to customer satisfaction. The website ensures a seamless booking process, secure transactions, and excellent customer support, ensuring that users have a hassle-free experience from start to finish. Whether it's assistance with inquiries, resolving issues, or providing travel advice, Al Mosafer is dedicated to helping users at every step of their travel journey.

In summary, Al Mosafer is a trusted travel website that offers a comprehensive range of services, exceptional customer support, and a commitment to delivering an exceptional travel experience. Whether it's for business or leisure, Al Mosafer aims to be the go-to platform for individuals seeking convenient, reliable, and enjoyable travel planning and booking.

Figure 4: Almosafer Logo

Objectives

- Provide a Comprehensive Travel Platform: The primary objective of Al Mosafer is to offer a comprehensive online platform that caters to various aspects of travel planning and booking.
- Enhance User Experience: Al Mosafer strives to provide a user-friendly interface and intuitive navigation, ensuring a seamless and enjoyable experience for its users.
- Offer Competitive Pricing and Deals: Al Mosafer aims to provide competitive pricing for flights, hotels, and other travel services.
- Provide Destination Information and Travel Tips: Al Mosafer aims to be a reliable source of destination information and travel tips.
- Ensure Customer Satisfaction and Support: Al Mosafer places a strong emphasis on customer satisfaction and support.
- Expand Travel Services and Partnerships: Al Mosafer continually seeks to expand its range of travel services and establish partnerships with airlines, hotels, and other travel-related businesses.
- Embrace Technological Advancements: Al Mosafer aims to stay at the forefront of technological advancements in the travel industry.

Chapter 2

METHODOLOGY

Software Tools

- **Eclipse**

Figure 5: Eclipse

An integrated development environment (IDE) primarily used for developing Java applications.

- **Microsoft teams**

Figure 6: Microsoft Teams

We are accustomed to meeting here in case there are matters that must be discussed in an urgent manner and at any time.

- **Word**

Figure 7: Word

Word was used to write the documentation.

- **Canva**

Figure 8: Canva

To write the presentation.

- **Selenium**

Figure 9: Selenium

To validate web applications across different browsers and platforms.

Methodology process

- **Data collection methods**

- **Focus Group**

We are three people. We met at Microsoft Teams to talk about the project, study it, and reach the best reaults by suggesting the best solution. We used to meet for three hours at least.

- **Brainstorming**

One of the methods of collective creativity is where we try to find a solution to a problem by compiling a list of ideas and solutions.

Chapter 3

TESTCASES AND CODE

Testcases

The purpose of writing test cases is to systematically check and verify that an application or software system correctly meets its specified requirements and functions. Test cases serve as detailed instructions for executing tests and help ensure that all aspects of the program are thoroughly tested.

Our trainer gave us six test cases to validate the site, which were evenly distributed to us.
The first two test cases are done by Ahmad, the third and fourth are done by Farah, and the last two test cases are done by Osama.

```
1  import java.util.Random;
7
8  public class Parameters {
9
10     SoftAssert myAssert = new SoftAssert();
11     Random myRandom = new Random();
12     WebDriver driver = new EdgeDriver();
13     static String [] URLS ={"https://www.almosafer.com/ar","https://www.almosafer.com/en"};
14
15     static String [] citiesInEnglish ={"Riyadh","Dubai","Jeddah"};
16     static String [] citiesInArabic ={"عمان,الأردن" , "جدة"};
17     static String DefaultCurrency = "SAR";
18     static String ContactNumber = "+966554400000";
19 }
20
```

Figure 10: Parameters class

1. Go to the website and check the default language is English**.**

```
public class TestCase1 extends Parameters {

    @BeforeTest
    public  void beforeTest() {
        driver.manage().window().maximize();
        driver.get("https://www.almosafer.com/");
    }

    @Test
    public void  checkTheDefualtLanguage() {
        String actualURL = driver.getCurrentUrl();
        System.out.println(actualURL);
        String expectedURL ="https://www.almosafer.com/en";
        myAssert.assertEquals(actualURL,expectedURL );

        }

    @AfterTest
    public void afterTest() {
        myAssert.assertAll();
        driver.quit();

    }
}
```

Figure 11: Test Case Code1

2. Go to the website and check the default currency is **SAR.**

```
 9 public class TestCase2 extends Parameters{
10
11     @BeforeTest
12     public  void beforeTest() {
13         driver.manage().window().maximize();
14         driver.get(URLS[1]);
15     }
16
17     @Test
18     public void checkTheDefualtCurrency() throws InterruptedException {
19         Thread.sleep(5000);
20         WebElement sarCurrency = driver.findElement(By.xpath("/html[1]/body[1]/div[2]/header[1]/div[1]/
21         String actualvalue= sarCurrency.getText();
22         String expectedvalue= "SAR";
23         myAssert.assertEquals(actualvalue,expectedvalue);
24
25         }
26
27     @AfterTest
28     public void afterTest() {
29         myAssert.assertAll();
30         driver.quit();
31
32     }
33 }
```

Figure 12: Test Case Code2

3. Go and select the countries to be randomly from (Riyadh, Dubai, Jeddah), to Amman and select the date to be after 2 days from the time you execute this test, and the return date will be after 4 days from the date of travel.

```
    @Test()
    public void selectCountriesRandomly() throws InterruptedException{
        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));

        WebElement countryFrom = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/d
        int random = myRandom.nextInt(citiesInEnglish.length);
        countryFrom.sendKeys(citiesInEnglish[random]+Keys.ENTER  );
        Thread.sleep(3000);

        WebElement countryTo = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/div
        countryTo.sendKeys("Amman" +Keys.ENTER );
        Thread.sleep(3000);
        //get the actual country for assertion
        String actualCountry = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/div
                .getAttribute("value");
        int commaIndex = actualCountry.indexOf(",");
        String actualCountryFrom = actualCountry.substring(0, commaIndex);

        //enter the date
        WebElement dateFrom = driver.findElement(By.className("ghIalB"));
        dateFrom.click();
        Thread.sleep(3000);

        //Date
        LocalDate currentDate = LocalDate.now();
        // Add two day to the current date
        LocalDate updatedDateFrom = currentDate.plusDays(2);
        // Add six day to the current date
        LocalDate updatedDateTo = currentDate.plusDays(6);

        DateTimeFormatter formatter = DateTimeFormatter.ofPattern("MMM dd yyyy");
        // Format the LocalDate object as a string with the specified format
        String formattedDateFrom = updatedDateFrom.format(formatter);
        String formattedDateTo = updatedDateTo.format(formatter);

        List<WebElement> dateDiv = driver.findElements(By.className("DayPicker-Day"));
        //selected date
        for(int i =0 ;i < dateDiv.size();i++) {
            String disabledDate = dateDiv.get(i).getAttribute("aria-disabled");
            String dateValue = dateDiv.get(i).getAttribute("aria-label");
            System.out.println(dateDiv.get(i).getAttribute("aria-disabled"));
```

Figure 13: Test Case Code3-a

```
            if(disabledDate.contains("false") && dateValue.contains(formattedDateFrom)) {
                    System.out.println(disabledDate);
                    System.out.println(dateValue);
                    Thread.sleep(5000);
                    //to solve Stale Element Reference Exception
                     try{
                        dateDiv.get(i).click();
                        }
                      catch(Exception e){
                           System.out.println(e.getMessage());
                        }
                }

            if(disabledDate.contains("false") && dateValue.contains(formattedDateTo)) {
                System.out.println(disabledDate);
                System.out.println(dateValue);
                Thread.sleep(5000);
                //to solve Stale Element Reference Exception
                 try{
                    dateDiv.get(i).click();
                     break;
                    }
                  catch(Exception e){
                        System.out.println(e.getMessage());
                    }
            }
        }

    Thread.sleep(3000);
    WebElement searchButton = driver.findElement(By.xpath("/html[1]/body[1]/div[2]/section[2]/div[4]/div[1]/div[1]/div[1]/div[1]/div[1]/div[2]/div
    searchButton.click();
    Thread.sleep(3000);
//  make soft assertion
    String expectedCountryFrom = driver.findElement(By.xpath("/html/body/div[2]/div[2]/div[1]/div[2]/div[1]/div[1]/div/div/div[2]")).getText();
    myAssert.assertEquals(actualCountryFrom, expectedCountryFrom);

    Thread.sleep(3000);
    String actualCountryToo = "Amman";
    String expectedCountryToo = driver.findElement(By.xpath("/html/body/div[2]/div[2]/div[1]/div[2]/div[1]/div[2]/div/div/div[2]")).getText();
    myAssert.assertEquals(actualCountryToo, expectedCountryToo);
    }
```

Figure 14: Test Case Code3-b

4. Let the website go randomly either to Arabic or English website, if the website is in English then it will enter from (Riyadh to Dubai), if it is in Arabic it will enter from (Amman to Jeddah).

```
21●    @Test()
22     public void randomWebSiteLanguage() throws InterruptedException {
23         driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
24         int random = myRandom.nextInt(URLS.length);
25         driver.get(URLS[random]);
26         String url = driver.getCurrentUrl();
27
28         if(url.equals(URLS[1])) {
29             System.out.println("english website => from riyadh to dubai");
30
31                     WebElement oneWay = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[1]/div[1]/div[1]"));
32                     oneWay.click();
33
34                     WebElement countryFrom = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/div[2]/div
35                     Thread.sleep(2000);
36                     countryFrom.sendKeys(citiesInEnglish[0]+Keys.ENTER);
37                     Thread.sleep(5000);
38
39                     WebElement countryTo = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/div[2]/div[3
40                     Thread.sleep(2000);
41                     countryTo.sendKeys(citiesInEnglish[1]+Keys.ENTER);
42
43                     Thread.sleep(3000);
44                     WebElement searchButton = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[2]/div/div[2]/di
45                     searchButton.click();
46
47                     String actualCountryFrom = "Riyadh";
48                     String actualCountryTo = "Dubai";
49                     Thread.sleep(5000);
50                     String expectedCountryFrom = driver.findElement(By.xpath("/html/body/div[2]/div[2]/div[1]/div[2]/div[1]/div[1]/div/div/div[2]")).getText();
51                     String expectedCountryTo = driver.findElement(By.xpath("/html/body/div[2]/div[2]/div[1]/div[2]/div[1]/div[2]/div/div/div[2]")).getText();
52                     myAssert.assertEquals(actualCountryFrom, expectedCountryFrom);
53                     myAssert.assertEquals(actualCountryTo, expectedCountryTo);
54
55         }else if(url.equals(URLS[0])) {
56             System.out.println("arabic website => from amman to jeddah");
57             WebElement oneWay = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[1]/div[1]/div[1]"));
58             oneWay.click();
59
60             WebElement countryFrom = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/div[2]/div[1]/div/
61             Thread.sleep(3000);
```

Figure 15: Test Case Code4-a

```
60             WebElement countryFrom = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/div[2]/div[1]/
61             Thread.sleep(3000);
62             countryFrom.sendKeys(citiesInArabic[0]+Keys.ENTER );
63             Thread.sleep(5000);
64
65             WebElement countryTo = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[1]/div/div[2]/div[3]/di
66             Thread.sleep(3000);
67             countryTo.sendKeys(citiesInArabic[1]+Keys.ENTER);
68
69             Thread.sleep(3000);
70             WebElement searchButton = driver.findElement(By.xpath("/html/body/div[2]/section[2]/div[4]/div/div/div/div[1]/div/div[2]/div[2]/div/div[2]/div/bu
71             searchButton.click();
72
73             String actualCountryFrom = "عمان";
74             String actualCountryTo = "جدة";
75             Thread.sleep(5000);
76             String expectedCountryFrom = driver.findElement(By.xpath("/html/body/div[2]/div[2]/div[1]/div[2]/div[1]/div[1]/div/div/div[2]")).getText();
77             String expectedCountryTo = driver.findElement(By.xpath("/html/body/div[2]/div[2]/div[1]/div[2]/div[1]/div[2]/div/div/div[2]")).getText();
78             myAssert.assertEquals(actualCountryFrom, expectedCountryFrom);
79             myAssert.assertEquals(actualCountryTo, expectedCountryTo);
80         }else{
81             myAssert.assertEquals(
82                     url.equals(URLS[0]) || url.equals(URLS[1]),
83                     true, "this is to test the website url ");
84
85         }
86     }
87
88●    @AfterTest
89     public void afterTest(){
90         myAssert.assertAll();
91         driver.quit();
```

Figure 16: Test Case Code4-b

5. Check the title for Arabic and English websites.

```java
public class TestCase5 extends Parameters {

    @BeforeTest
    public  void beforeTest() {
        driver.manage().window().maximize();
    }

    @Test(priority = 1)
    public void englishWebsiteTitleTest() {

            driver.get(URLS[1]);
            String ActualTitle = driver.getTitle();
            String ExpectedTitle = "Almosafer: Flights, Hotels, Activities & Airlines Ticket Booking";
            myAssert.assertEquals(ActualTitle, ExpectedTitle);

    }
    @Test(priority = 2)
    public void arabicWebsiteTitleTest() {

        driver.get(URLS[0]);
        String ActualTitle = driver.getTitle();
        String ExpectedTitle = "المسافر: رحلات، فنادق، أنشطة ممتعة و تذاكر طيران";
        myAssert.assertEquals(ActualTitle, ExpectedTitle);
    }
    @AfterTest
    public void afterTest() {
        myAssert.assertAll();
        driver.quit();
    }
}
```

Figure 17: Test Case Code5

6. Go to the hotel tab and complete one complete process.

```java
public class TestCase6 extends Parameters {


    @BeforeTest
    public  void beforeTest() {

        driver.manage().window().maximize();
        driver.get(URLS[1]);
    }

    @Test()
    public void hotelsBookTest() throws InterruptedException {
        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
        WebElement Hotels_tab = driver.findElement(By.id("uncontrolled-tab-example-tab-hotels"));
        Hotels_tab.click();
        Thread.sleep(2000);
        WebElement Hotels_input = driver.findElement(By.className("AutoComplete__Input"));
        Select SelectRoomsButton = new Select(driver.findElement(By.className("tln3e3-1")));
        WebElement Searchbutton = driver.findElement(By.className("js-HotelSearchBox__SearchButton"));

        // Hotel info steps
        Hotels_input.sendKeys("Amman" + Keys.ARROW_DOWN + Keys.ENTER);
        SelectRoomsButton.selectByVisibleText("1 Room, 1 Adult, 0 Children");
        Searchbutton.click();
        // Select hotel
        JavascriptExecutor jse = (JavascriptExecutor) driver;
        jse.executeScript("window.scrollBy(0,50)");
        WebElement SeeRoomsButton = driver.findElement(By.xpath("//*[@id=\"hotelCard-1447944\"]/div[3]/div/div[2]/button"));
        SeeRoomsButton.click();

        String currentTab = driver.getWindowHandle();
        for (String tab : driver.getWindowHandles()) {
            if (!tab.equals(currentTab)) {
                driver.switchTo().window(tab);
                // Select a room
                WebElement RoomChoicesButton = driver.findElement(By.xpath("//*[@id=\"__next\"]/div[2]/div[1]/div/div/a[2]"));
                RoomChoicesButton.click();
                WebElement FreeCancellation = driver.findElement(By.xpath("//*[@id=\"__next\"]/div[2]/div[3]/section[2]/section/div/div/div/div[2]/labe
                FreeCancellation.click();
                WebElement BookNowButton = driver.findElement(By.xpath("//*[@id=\"__next\"]/div[2]/div[3]/section[2]/section/section/div[1]/div[1]/div[
                BookNowButton.click();
                // Checkout Page
```

Figure 18: Test Case Code6-a

```java
                // Checkout Page
                WebElement FirstName = driver.findElement(By.name("contact.firstName"));
                WebElement LastName = driver.findElement(By.name("contact.lastName"));
                WebElement Email = driver.findElement(By.name("contact.email"));
                WebElement PhoneNumber = driver.findElement(By.name("contact.phoneNumber"));
                WebElement ContinueToPayment = driver.findElement(By.xpath("//*[@id=\"__next\"]/div[2]/div/div[3]/div/div[2]/div/div[1]/div[5]/
                // Enter guest details Steps
                FirstName.sendKeys("Osama");
                LastName.sendKeys("Alzoubi");
                Email.sendKeys("osamayahia878@gmail.com");
                PhoneNumber.sendKeys("7485259");
                ContinueToPayment.click();
                // Payment Page
                WebElement PayLater = driver.findElement(By.xpath("//*[@id=\"__next\"]/div[2]/div/div[3]/div/div[1]/form/div/div/div[2]/div/div
                PayLater.click();
            }
        }

    }

    @AfterTest
    public void afterTest() {
        myAssert.assertAll();
        driver.quit();
    }
}
```

Figure 19: Test Case Code6-b

Chapter 4

COURSE OVERVIEW

Introduction

This course provides a comprehensive understanding of quality assurance principles and practices in various industries. It explores the key concepts, methodologies, and tools used to ensure the delivery of high-quality products and services. Throughout the course, we gained insights into the importance of quality assurance in meeting customer expectations, enhancing organizational performance, and achieving business success.

It also contained soft skills sessions with a number of experts in this field, from whom we benefited a lot.

Course Duration

The course began on the 14 of March to 2 of May of 2023, consisting of a number of modules that cover a range of topics related to quality assurance.

The key takeaways and knowledge we have gained

- ✓ **Quality assurance and Quality control**

  Quality Assurance aims to prevent quality issues by establishing processes, standards, and policies, while Quality Control focuses on detecting quality issues through inspections, testing, and measurements. Both QA and QC are essential components of a comprehensive quality management system and work together to ensure the delivery of high-quality products or services.

- ✓ **SDLC and STLC**

  SDLC provides a framework for managing the entire software development process, from concept to deployment and maintenance, while STLC focuses specifically on the testing aspects within the development life cycle. Both SDLC and STLC are crucial in ensuring the delivery of high-quality software applications by following a structured approach.

- ✓ **Environment types**

  Each type of environment plays a crucial role in the software development process, supporting activities like coding, testing, deployment, and maintenance. By utilizing these different environments, software development teams can efficiently develop, validate, and deliver high-quality software applications while minimizing risks and ensuring the stability and reliability of the final product.

✓ **Testing levels**

Testing levels refer to the different stages or layers of testing that are performed during the software testing process. Each testing level focuses on specific aspects of the software and helps ensure its quality and functionality.

Figure 20: Testing levels

✓ **Testing types**

Testing types refer to the different approaches or techniques used to evaluate software applications during the testing process. Each testing type focuses on specific aspects of the software and helps uncover different types of defects or issues.

Figure 21: Testing types

✓ **Testing techniques**

Testing techniques are specific methods or approaches used during the software testing process to design and execute test cases and uncover defects. These techniques help ensure effective test coverage and improve the efficiency and effectiveness of testing efforts.

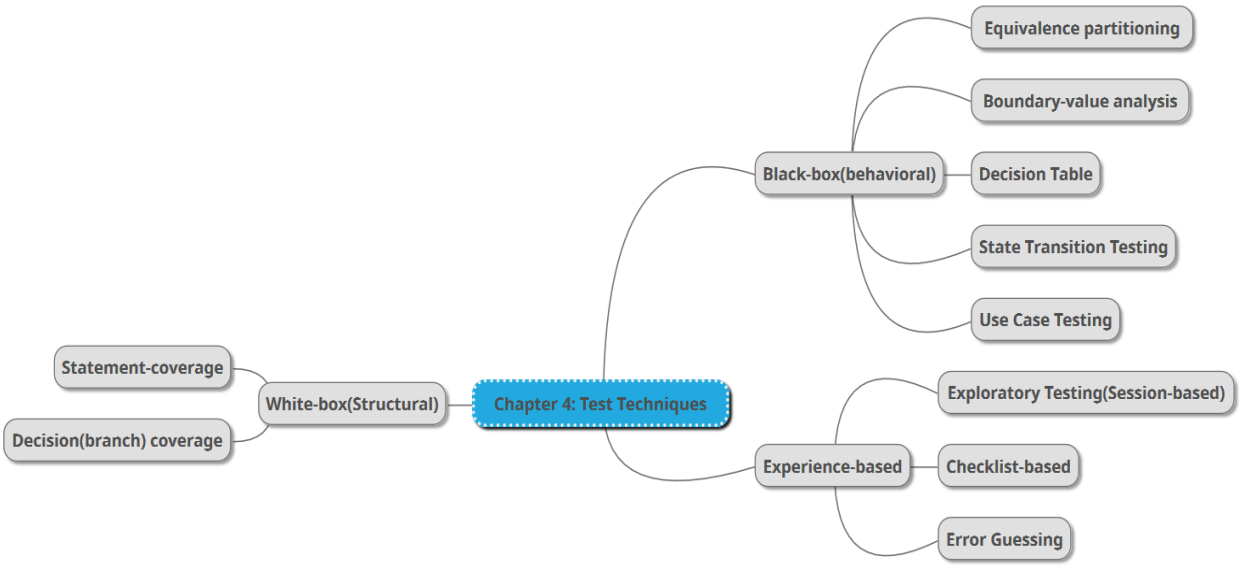


Figure 22: Testing techniques

✓ **Test case and bug report**

A test case is a set of instructions that outlines the steps to be followed to execute a specific test scenario, whereas a bug report is a document that identifies and describes a defect or issue found during testing.

	A	B	C
1	**Category**	**Label**	**Value**
2	Bug ID	ID number	#123
3		Name	CART - Unable to add new item to my cart
4		Reporter	Mike A
5		Submit Date	03/04/16
6	Bug overview	Summary	When my cart contains one item, I am unable to add a second item via the add to cart button on a product page
7		URL	www.example.com/product/abc
8		Screenshot	www.example.com/screenshot123
9	Environment	Platform	Macintosh
10		Operating System	OS X 10.12.0
11		Browser	Chrome 53
12	Bug details	Steps to reproduce	add one item to cart > go to product abc via the search bar > add new item to cart via "add to cart" button (see screenshot) > go to cart
13		Expected result	The cart should contain 2 items
14		Actual result	The cart contains only 1 item
15		Description	/
16	Bug tracking	Severity	Major
17		Assigned to	/
18		Priority	High
19	Notes	Notes	/

Figure 23: Bug report

	A	B	C	D	E	F	G
1	**Id**	**Description**	**Test Data**	**Expected Result**	**Actual Result**	**Pass/Fail**	**Notes**
2	1	Sign Up	First Name:Ahmad, Last Name:abushanab, Email:abushanab323@gmail.com, Password:Ahmad*21, confirm Password:Ahmad*21	Create Account With my gmail	As Expected	Pass	
3	2	Sign In	Email:abushanab323@gmail.com / Password:Ahmad*21	Login to my Accouont	As Expected	Pass	
4	3	Searching for item	Jacket	show all the jackets that available in the store	As Expected	Pass	
5	4	Adding 3 items of RadiantTee	RadiantTee	Adding 3 items of Radianttee to my cart	As Expected	Pass	
6	5	Adding 2 items of breathe-easytank	breathe-easytank	Adding 2 items of breathe-easytank to my cart	As Expected	Pass	
7	6	Adding 4 items of argusallweathertank	argusallweathertank	Adding 4 items of argusallweathertank to my cart	As Expected	Pass	
8	7	Adding 1 item of herohoodie	herohoodie	Adding 1 item of herohoodie to my cart	As Expected	Pass	
9	8	Adding 5 items of Fusion Backpack	Fusion Backpack	Adding 5 items of Fusion Backpack to my cart	The requested quantity is not available	Fail	

Figure 24: Test Case

✓ **The seven principles of testing**

The Seven Principles of Testing are a set of guidelines that help testers design and execute effective testing strategies. These principles, defined by ISTQB (International Software Testing Qualifications Board), are widely accepted and followed in the field of software testing.

By following these principles, testers can establish a structured and effective testing approach, increase the likelihood of identifying defects, and contribute to delivering high-quality software.

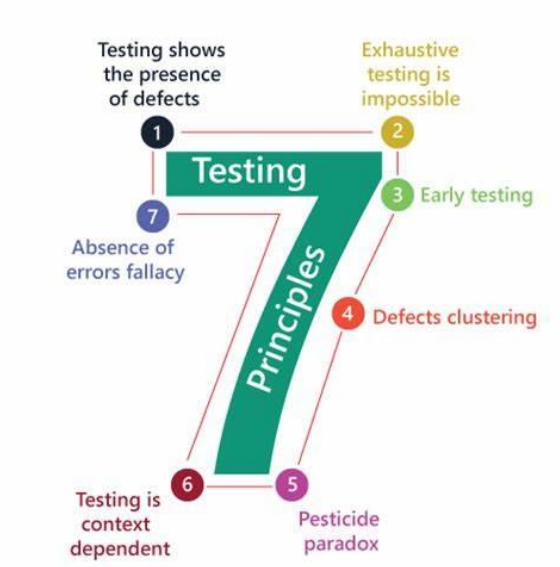


Figure 25: The seven principles of testing

✓ **Selenium using java**

Selenium is an open-source automation testing framework widely used for web application testing. It provides a robust set of tools and APIs to interact with web browsers and automate browser-based tasks. Selenium supports various programming languages, including Java, making it a popular choice for Java developers in test automation.

✓ **API testing using postman**

Postman is a widely used tool for API testing and development. It provides a user-friendly interface that allows testers and developers to send HTTP requests, analyze responses, and validate API functionality.

Using Postman for API testing provides a convenient and efficient way to validate and interact with APIs.

✓ **Mobile testing using Appium**

Appium is an open-source automation framework that is widely used for mobile testing. It allows testers to automate the testing of mobile applications on various platforms, including Android and iOS.

Appium enables testers to ensure the quality and reliability of mobile applications across different platforms and devices.

✓ **TestNG Framework**

TestNG is a widely used testing framework for Java applications that provides a powerful and flexible platform for writing and executing tests. It offers various features and functionalities that enhance the testing process and enables efficient test management.

✓ **Generate testing reports**

The goal of generating a testing report is to provide stakeholders with a comprehensive overview of the testing process, outcomes, and risks associated with the software being tested.

Reports helps you to identify the status of the test case (Pass/Fail/Skip), Using reports we calculate time taken by each test case that helps to calculate ROI(Return on Investment), You can share automation reports with your team and clients as well to share the status of testing progress etc.

This is our test report on the website, we have six classes with seven methods.

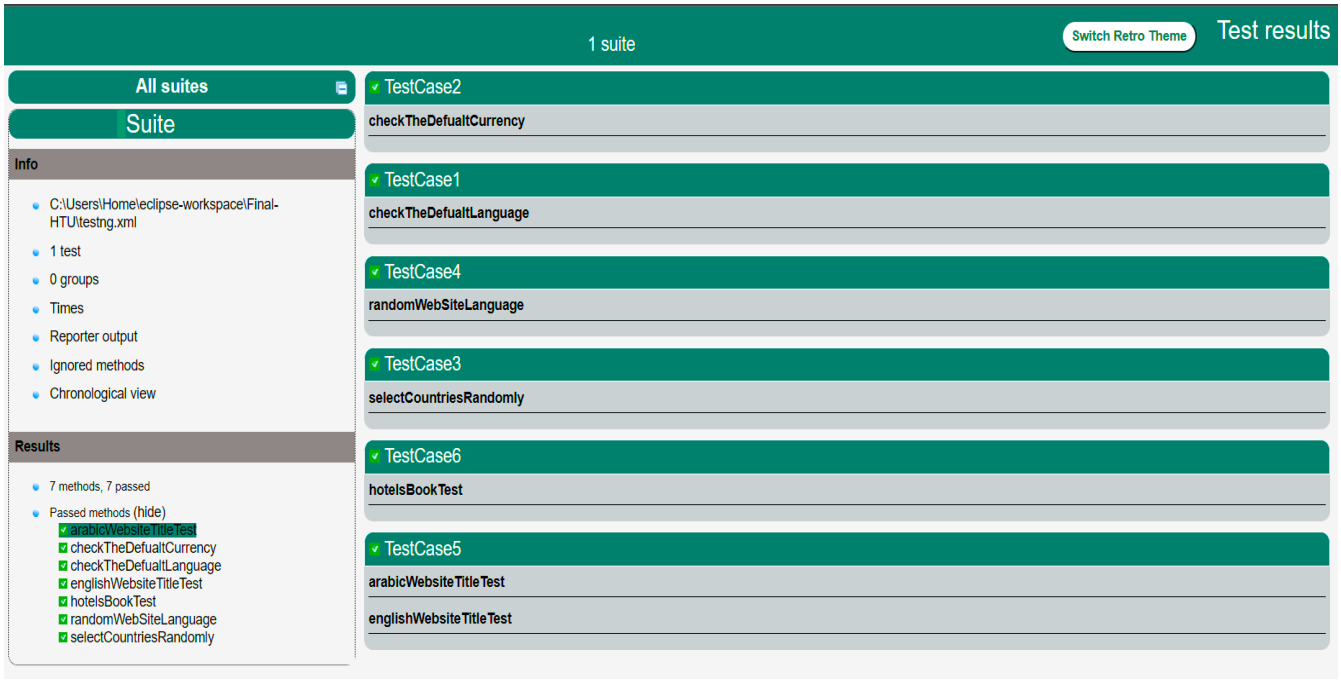


Figure 26: Testing Report(index.html)

Test	# Passed	# Skipped	# Retried	# Failed	Time (ms)	Included Groups	Excluded Groups
Suite							
Test	7	0	0	0	342,349		

Class	Method	Start	Time (ms)
Suite			
Test — failed (configuration methods)			
TestCase1	afterTest	1685879600143	1
Test — skipped (configuration methods)			
TestCase2	afterTest	1685879600150	-1685879600150
TestCase3	afterTest	1685879600153	-1685879600153
TestCase4	afterTest	1685879600156	-1685879600156
TestCase5	afterTest	1685879600159	-1685879600159
TestCase6	afterTest	1685879600161	-1685879600161
Test — passed			
TestCase1	checkTheDefualtLanguage	1685879275700	49
TestCase2	checkTheDefualtCurrency	1685879275758	5154
TestCase3	selectCountriesRandomly	1685879280916	47598
TestCase4	randomWebSiteLanguage	1685879328517	82213
TestCase5	arabicWebsiteTitleTest	1685879414040	1455
	englishWebsiteTitleTest	1685879410733	3303
TestCase6	hotelsBookTest	1685879415496	184644

Figure 27: Testing Report(emailable-report.html)

✓ **JMeter performance testing**

Apache JMeter is a popular open-source tool used for performance testing, load testing, and stress testing of web applications. It provides a comprehensive set of features and capabilities to simulate heavy loads and measure the performance of a system.

We have done performance testing on Almosafer website using JMeter and we viewed the result in two ways: graph and table.

We put 1000 users per 1000 seconds and the number of loop count is 10.

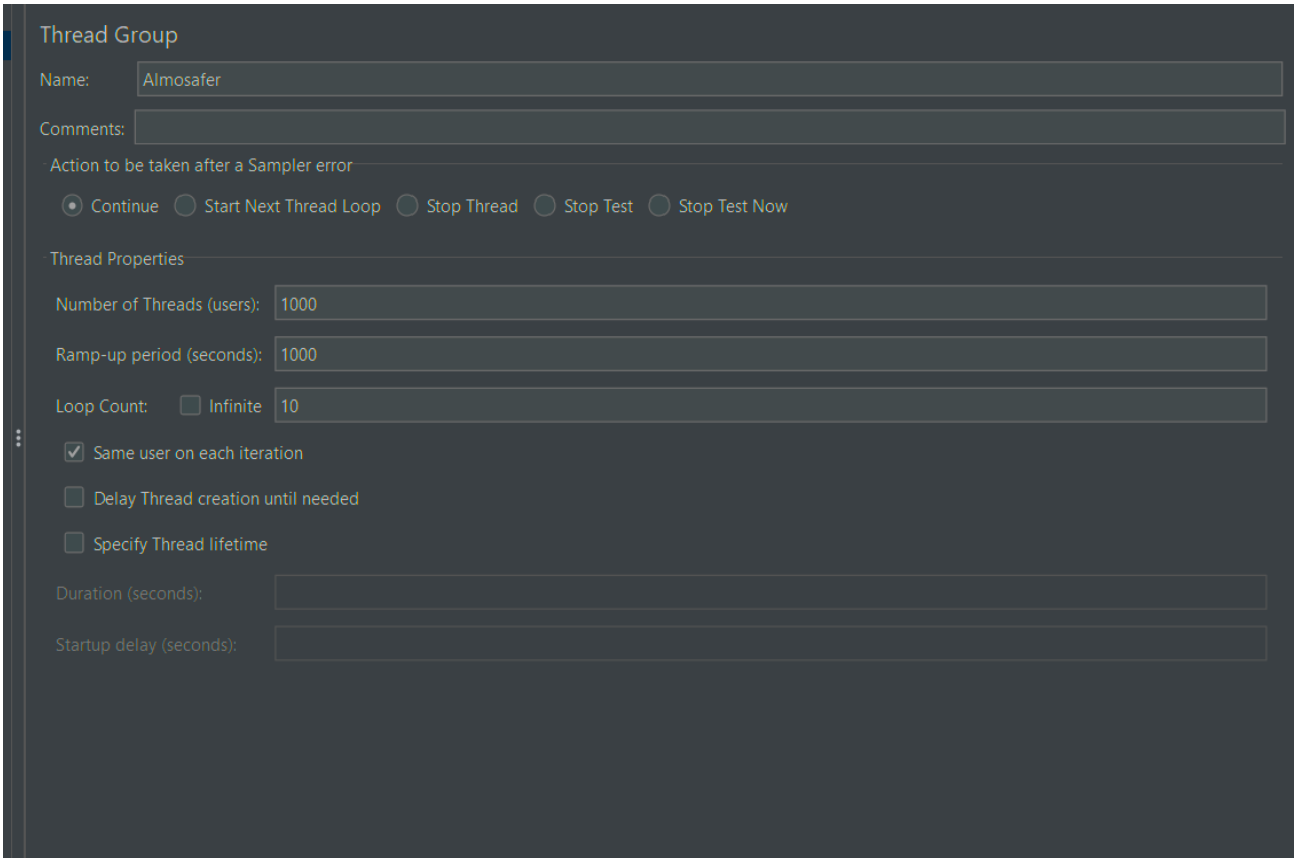
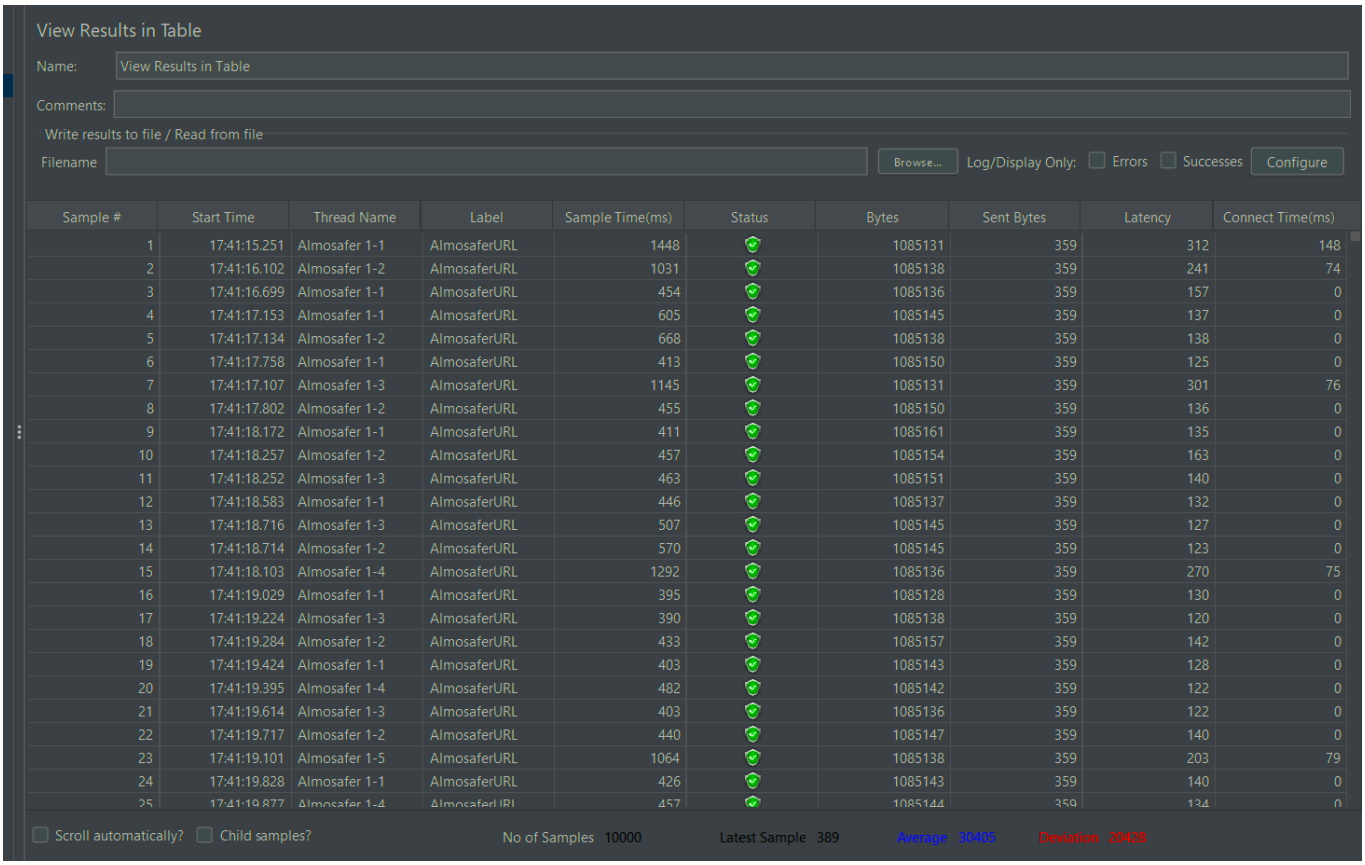
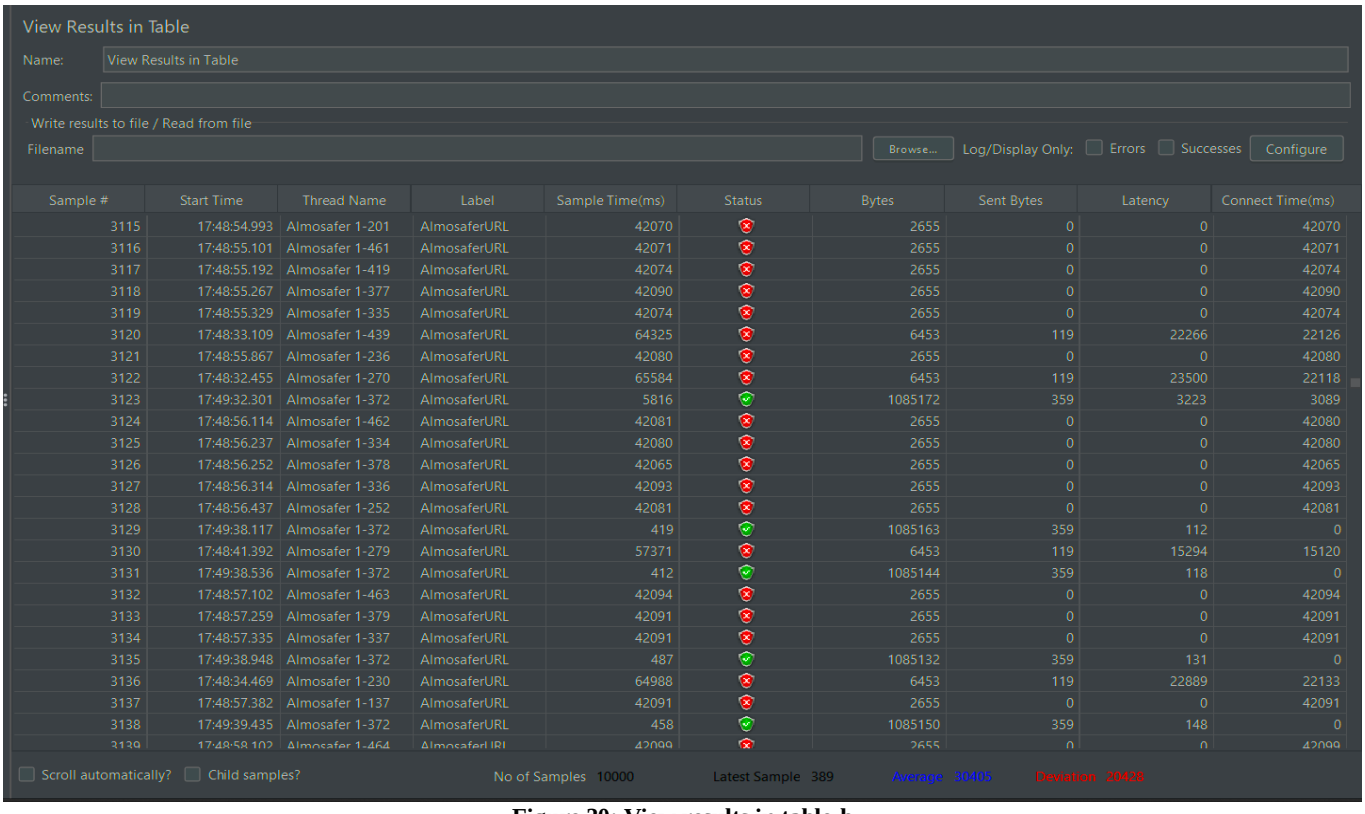


Figure 28: Thread group(users)



Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	17:41:15.251	Almosafer 1-1	AlmosaferURL	1448		1085131	359	312	148
2	17:41:16.102	Almosafer 1-2	AlmosaferURL	1031		1085138	359	241	74
3	17:41:16.699	Almosafer 1-1	AlmosaferURL	454		1085136	359	157	0
4	17:41:17.153	Almosafer 1-1	AlmosaferURL	605		1085145	359	137	0
5	17:41:17.134	Almosafer 1-2	AlmosaferURL	668		1085138	359	138	0
6	17:41:17.758	Almosafer 1-1	AlmosaferURL	413		1085150	359	125	0
7	17:41:17.107	Almosafer 1-3	AlmosaferURL	1145		1085131	359	301	76
8	17:41:17.802	Almosafer 1-2	AlmosaferURL	455		1085150	359	136	0
9	17:41:18.172	Almosafer 1-1	AlmosaferURL	411		1085161	359	135	0
10	17:41:18.257	Almosafer 1-2	AlmosaferURL	457		1085154	359	163	0
11	17:41:18.252	Almosafer 1-3	AlmosaferURL	463		1085151	359	140	0
12	17:41:18.583	Almosafer 1-1	AlmosaferURL	446		1085137	359	132	0
13	17:41:18.716	Almosafer 1-3	AlmosaferURL	507		1085145	359	127	0
14	17:41:18.714	Almosafer 1-2	AlmosaferURL	570		1085145	359	123	0
15	17:41:18.103	Almosafer 1-4	AlmosaferURL	1292		1085136	359	270	75
16	17:41:19.029	Almosafer 1-1	AlmosaferURL	395		1085128	359	130	0
17	17:41:19.224	Almosafer 1-3	AlmosaferURL	390		1085138	359	120	0
18	17:41:19.284	Almosafer 1-2	AlmosaferURL	433		1085157	359	142	0
19	17:41:19.424	Almosafer 1-1	AlmosaferURL	403		1085143	359	128	0
20	17:41:19.395	Almosafer 1-4	AlmosaferURL	482		1085142	359	122	0
21	17:41:19.614	Almosafer 1-3	AlmosaferURL	403		1085136	359	122	0
22	17:41:19.717	Almosafer 1-2	AlmosaferURL	440		1085147	359	140	0
23	17:41:19.101	Almosafer 1-5	AlmosaferURL	1064		1085138	359	203	79
24	17:41:19.828	Almosafer 1-1	AlmosaferURL	426		1085143	359	140	0
25	17:41:19.877	Almosafer 1-4	AlmosaferURL	457		1085144	359	134	0

Figure 29: View results in table-a



Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
3115	17:48:54.993	Almosafer 1-201	AlmosaferURL	42070		2655	0	0	42070
3116	17:48:55.101	Almosafer 1-461	AlmosaferURL	42071		2655	0	0	42071
3117	17:48:55.192	Almosafer 1-419	AlmosaferURL	42074		2655	0	0	42074
3118	17:48:55.267	Almosafer 1-377	AlmosaferURL	42090		2655	0	0	42090
3119	17:48:55.329	Almosafer 1-335	AlmosaferURL	42074		2655	0	0	42074
3120	17:48:33.109	Almosafer 1-439	AlmosaferURL	64325		6453	119	22266	22126
3121	17:48:55.867	Almosafer 1-236	AlmosaferURL	42080		2655	0	0	42080
3122	17:48:32.455	Almosafer 1-270	AlmosaferURL	65584		6453	119	23500	22118
3123	17:49:32.301	Almosafer 1-372	AlmosaferURL	5816		1085172	359	3223	3089
3124	17:48:56.114	Almosafer 1-462	AlmosaferURL	42081		2655	0	0	42080
3125	17:48:56.237	Almosafer 1-334	AlmosaferURL	42080		2655	0	0	42080
3126	17:48:56.252	Almosafer 1-378	AlmosaferURL	42065		2655	0	0	42065
3127	17:48:56.314	Almosafer 1-336	AlmosaferURL	42093		2655	0	0	42093
3128	17:48:56.437	Almosafer 1-252	AlmosaferURL	42081		2655	0	0	42081
3129	17:49:38.117	Almosafer 1-372	AlmosaferURL	419		1085163	359	112	0
3130	17:48:41.392	Almosafer 1-279	AlmosaferURL	57371		6453	119	15294	15120
3131	17:49:38.536	Almosafer 1-372	AlmosaferURL	412		1085144	359	118	0
3132	17:48:57.102	Almosafer 1-463	AlmosaferURL	42094		2655	0	0	42094
3133	17:48:57.259	Almosafer 1-379	AlmosaferURL	42091		2655	0	0	42091
3134	17:48:57.335	Almosafer 1-337	AlmosaferURL	42091		2655	0	0	42091
3135	17:49:38.948	Almosafer 1-372	AlmosaferURL	487		1085132	359	131	0
3136	17:48:34.469	Almosafer 1-230	AlmosaferURL	64988		6453	119	22889	22133
3137	17:48:57.382	Almosafer 1-137	AlmosaferURL	42091		2655	0	0	42091
3138	17:49:39.435	Almosafer 1-372	AlmosaferURL	458		1085150	359	148	0
3139	17:48:58.102	Almosafer 1-464	AlmosaferURL	42099		2655	0	0	42099

Figure 30: View results in table-b

Figure 31: Graph results

The Throughput is the most important parameter in graph result. It represents the ability of the server to handle a heavy load. The higher the Throughput is, the better is the server performance.

In this test, the throughput of Almosafer server is 475.213/minute. It means Almosafer server can handle 475.213 requests per minute. This value is quite high so we can conclude that Almosafer server has good performance.

The deviation is shown in red is 20428– it indicates the deviation from the average. The smaller the better.

As we see in figure 30, in some samples the status is failed because the user cannot access the server and some of them with high latency.

Chapter 5

CONCLUSION

Conclusion

In conclusion, this Quality Assurance course was a transformative experience, as it enabled us to use the tools, techniques, and mindset needed to excel in the field of Quality Assurance. We are passionate about applying our newly acquired knowledge and skills to drive continuous improvement in the quality of programs and to contribute to the success of future projects.

We would like also to thank Al Hussein technical university, our instructors, and all those responsible for this work and for giving us the opportunity to participate in it.

Future Work

Our goal is to become a skilled and influential quality assurance professional who contributes to the development of high-quality software products. We envision ourselves taking on challenging projects and leading testing efforts to ensure that software meets the highest standards of quality, reliability, and user satisfaction.

We aim to stay up-to-date with industry trends and best practices, embracing automation, and agile methodologies, and to take the ISTQB Certificate foundation level.

<<The End>>