Deanship of Graduate Studies
Al-Quds University

Robotic Process Automation Framework for Web Applications Performance and Security Usability Testing

Aysar Satee Fayez Qasrawi

M.Sc. Thesis

Jerusalem-Palestine

1444 / 2022

Robotic Process Automation Framework for Web Applications Performance and Security Usability Testing

Prepared By:

Aysar Satee Fayez Qasrawi

Supervisor:

Dr. Radwan Qasrawi

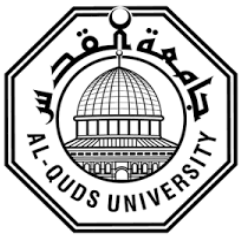
A Thesis Submitted in Partial Fulfillment of Requirements for The Degree of Master's in Computer Science / Department of Computer Science / Faculty of Graduate Students / Deanship of Graduate Studies / Al-Quds University.

1444 / 2022

Deanship of Graduate Studies

Al-Quds University

Computer Science



Thesis Approval

Robotic Process Automation Framework for Web Applications Performance and Security Usability Testing

Prepared By: Aysar Satee Fayez Qasrawi

Registration No: 21920177

Supervisor: Dr. Radwan Qasrawi

Master thesis submitted and accepted, Date: 1/9/2022

The names and signatures of the examining committee are as follow:

1- Head of Committee: Dr. Radwan Qasrawi Signature .....

2- Internal Examiner: Dr. Raed Zaghal Signature .....

3- External Examiner: Dr. Iyad Tumar Signature .....

Jerusalem – Palestine

1444 / 2022

Declaration

I certify that this thesis submitted for the degree of Master, is the result of my research, except where otherwise acknowledged, and that this study (or any part of the same) has not been submitted for a higher degree to any other university or institution.

Aysar Satee Fayez Qasrawi

Signature: 

Date: 01/09/2022

Dedication

إلى من لا يضاهيهما أحد في الكون، يا من أمرنا الله ببرّهما، إلى من بذلا الكثير، وقدّما ما لا يمكن أن يردّ، إلى أبي وأمي الغاليين

إلى جدي وجدتي وعائلتي الذين كانوا دائماً بالنسبة لي بمثابة العضد والسند

أهدي لكم جميعا هذا البحث، فقد كنتم خير داعم لي طوال مسيرتي الدراسية

Aysar Satee Fayez Qasrawi

Acknowledgment

I would like to express my deep gratitude to Dr. Radwan Qasrawi, my research supervisor, for his guidance, enthusiastic encouragement, and useful critiques of this research work.

I would also like to thank my thesis committee, Dr. Raid Zaghal, and Dr. Iyad Tumar for the insightful comments and feedback.

Finally, I must express my very profound gratitude to my parents and family for providing me with unfailing support and continuous encouragement. This accomplishment would not have been possible without them.

Abstract

User Experience and Usability analysis of websites are important aspects that every website should focus more on. It indicates how effectively and successfully a website will work with real users. Several tools, both traditional and automated, have been developed to assist developers in user experience testing and evaluation. Most of these tools have limited access due to the accessibility cost and time-consuming. The goal of this research is to facilitate the user experience testing process and enhance websites' performance, security, and accessibility by developing an integrated user experience testing framework based on Robotic Process Automation.

The framework is integrated with JMeter for performance testing, ImmuniWeb for Security testing, and WAVE for accessibility testing. Experiments were conducted on 16 local websites from 4 different categories which are High Educational Institutions, Governmental, News and eCommerce websites. The Performance tests were conducted using 3G network and 8 Mbps LAN network, and we found that the average response time (website loading time) of all websites in 8 Mbps LAN network was 6458 msec while it was 10604 msec in 3G mobile network. Moreover, we found that 50% (8/16) of the websites have returned errors to some users in the 8 Mbps experiment due to the high traffic, while the percentage was 31% (5/16) of the websites in the 3G network experiment.

The results of security testing indicated that there are major issues in 4 websites, where there are outdated third-party software or CMS components in these websites, and should be updated as soon as possible, such websites are vulnerable to publicly known vulnerabilities. Most websites have a good SSL test grade, with approximately 81% (13/16) of the websites receiving a grade higher than A-.

The accessibility testing results showed that all websites have critical accessibility issues, and none of the analyzed websites met the required compliance level A. Therefore, these websites cannot be used effectively or satisfactorily by all stakeholders, particularly disabled prospective users.

Robotic Process Automation bots showed a high capability in the testing of all websites, which will help in enhancing the quality and usability of websites based on international HCI standards and guidelines.

List of Tables

Table 3.1: Performance Testing Measurement Parameters 46

Table 3.2: Accessibility testing measurement parameters 49

Table 3.3: Traffic Details of the Selected Websites 56

Table 4.1: Web software security scanning results 57

Table 4.2: Web software security testing results by website 57

Table 4.3: GDPR testing results 58

Table 4.4: PCI DSS testing results 59

Table 4.5: Security testing results of HEI websites 59

Table 4.6: Security testing results of Governmental websites 61

Table 4.7: Security testing results of news websites 63

Table 4.8: Security testing results of eCommerce websites 64

Table 4.9: Failed Security Tests Statistics by Website's Category 65

Table 4.10: Number of failed websites in SSL testing by website’s category 68

Table 4.11: SSL testing results by website 69

Table 4.12: Failed SSL testing items by website's category 72

Table 4.13: Performance testing results using 3G network 73

Table 4.14: Performance testing results using 8 Mbps LAN network 77

Table 4.15: Accessibility testing results 81

List of Figures

Figure 2.1: Robotic Process Automation components 32

Figure 3.1: Accessibility testing parameters 49

Figure 3.2: Proposed framework's architecture 52

Figure 3.3: Developed database’s diagram 53

Figure 3.4: Lifecycle of the Security Testing Bot 54

Figure 3.5: Lifecycle of the Performance Testing Bot 55

Figure 3.6: Lifecycle of the Accessibility Testing Bot 55

Figure 4.1: Security testing results of HEI websites 61

Figure 4.2: Security Testing Results of Governmental Websites 62

Figure 4.3: Security testing results of News website 64

Figure 4.4: Most common security issues detected by Bot security model 68

Figure 4.5: SSL testing results of HEI websites 70

Figure 4.6: SSL testing results of News websites 71

Figure 4.7: SSL testing results of eCommerce websites 72

Figure 4.8: Average response time in msec by website using 3G network 75

Figure 4.9: Throughput by website using 3G network 76

Figure 4.10: Number of completed samples by website using 3G network 77

Figure 4.11: Average response time in msec by website using 8 Mbps LAN network 78

Figure 4.12: Throughput by website using 8 Mbps LAN network 79

Figure 4.13: Number of completed samples by website using 8 Mbps LAN network 79

Figure 4.14: Total number of errors, contrast errors, and alerts in HEI websites 82

Figure 4.15: Total number of errors, contrast errors, and alerts in Governmental websites 82

Figure 4.16: Total number of errors, contrast errors, and alerts in News websites 83

Figure 4.17: Total number of errors, contrast errors, and alerts in eCommerce websites 83

Table of Contents

Chapter 1: Introduction *1*

1.1 Introduction **1**

1.2 Problem Statement **2**

1.3 Objectives of the Study **3**

1.4 Motivation **3**

1.5 Thesis Organization **4**

Chapter 2: Background and Literature Review *5*

2.1 Background **5**

2.1.1 Human-Computer Interaction and Usability Testing **5**

2.1.2 Web Usability Testing **6**

2.1.3 HCI standards **7**

2.1.4 Performance Testing **9**

2.1.5 Security Testing **13**

2.1.6 Accessibility Testing **17**

2.1.7 Robotic Process Automation **18**

2.2 Literature Review **21**

2.1.1 Related Work **21**

2.1.2 Discussion **28**

Chapter 3: Methodology *30*

3.1 Study Design **30**

3.2 User Experience Security Testing Model **30**

3.2.1 Website Security and CMS Test Bot **31**

3.2.1 SSL Security Test Bot **33**

3.3 User Experience Performance Testing Model **34**

3.4 User Experience Accessibility Testing Model **36**

3.5 User Experience Framework Development **39**

3.6 Bots Development **41**

3.7 Selected Websites for Usability Testing **44**

Chapter 4: Experimental Results and Analysis *45*

4.1 Security Testing Results **45**

4.1.1 Web Security and CMS Testing Results **45**

4.1.2 SSL Testing Results **55**

4.2 Performance Testing Results **60**

4.3 Accessibility Testing Results **66**

Chapter 5: Discussion and Conclusion *70*

5.1 Research Limitations **70**
5.2 Discussion **70**
5.3 Conclusion and Future Work **73**
References ***75***
الملخص ***82***

Chapter 1: Introduction

1.1 Introduction

Websites and the internet have grown significantly in recent years. Studies indicate that approximately five billion people – roughly 63 percent of the world's population use the Internet [1]. Users should be able to navigate around an interface easily upon their first encounter, without relying on experts to achieve goals. User Experience Design clearly focuses on the user and seeks to anticipate an audience's needs and how they expect a website to function, look, and feel. These elements include all touch points a user encounter when visiting a website: navigation, content, imagery, calls to action, interaction, forms, etc.

Customers who find the website or mobile application confusing or boring will likely favor another competitor with better usability. According to Forrester Research [2], a "well-constructed user interface can increase your website's conversion rate by up to 200%, and a better UX design can achieve conversion rates of up to 400%." It is ill-advised for companies to improve the end-user experience.

Human-computer interaction, otherwise known as HCI, studies how people interact with computers and the extent of computers' development for successful interaction with humans [3]. Due to its rapid expansion around the globe throughout the past three decades, HCI attracts the interest of researchers and professionals from various fields. Thus, HCI has integrated a wide variety of concepts and approaches. Usability is a significant concept in HCI which refers to measuring how effectively, efficiently, and satisfactorily a specific user in a particular context can use a product or design to achieve a defined goal. To ensure maximum usability, designers and developers usually measure a design's usability throughout the development process. The usability assessment aims to identify and address any potential problems before the product is released. Users are introduced to the new design during the usability test to guarantee that it is easy to use and provides a satisfying user experience. In

addition to traditional usability testing methods, automated usability testing can provide many benefits to end users and businesses. The utilization of automation tools aids in resource-saving, cost reduction, performance, and security testing. In this research, we will develop one framework to evaluate 16 Palestinian local websites using Robotic Process Automation which is integrated with performance, usability, and accessibility automation tools.

1.2 Problem Statement

A Website Application's usability testing is one of the major steps in the software development cycle. It has a major impact on the application efficiency and effectiveness, furthermore, it plays an important role in customer satisfaction. Recently, due to the rapid technology development and the increase of internet accessibility, software development companies start paying more attention to website and mobile applications development. On the other hand, the simplicity of website development produced many websites that have been developed without considering the software engineering development cycle in terms of usability, performance, security, and accessibility issues.

There are several tools that have been developed for helping the developers in user experience testing and evaluation, either traditional or automated tools. Such tools provide specific user experience testing measure, and there are very limited frameworks that provide integrated user experience testing according to HCI standards and guidelines. Most of the available tools have limited access due to the accessibility cost and time-consuming, at the same time the developers need to subscribe and access many user experience testing tools to have complete testing and evaluation process.

Providing an integrated user experience framework based on Robotic Process Automation (Bot programs) will facilitate user experience testing process and enhance websites usability and accessibility. For this reason, our research aims at deploying the robotics process

automation bots in user experience testing and evaluation, and provide an integrated framework for performance, security, and accessibility websites testing. The research questions are:

- Does Robotic Process Automation RPA improve the usability testing in terms of websites performance testing?
- Can RPA be a good tool to check the security of the websites' applications in terms of SSL and Website security evaluation?
- Does Robotic Process Automation RPA help improve the usability testing in terms of websites accessibility testing?

1.3 Objectives of the Study

- Design and develop a framework for evaluating the usability of websites based on international HCI standards and guidelines, including three different types of testing: performance, security, and accessibility.
- Improve user experience testing process by deploying the Robotic Process Automation as an automated testing tool.
- Improve the software development process by providing an integrated usability testing framework.
- Provide the local software development industry with an efficient user experience framework for enhancing the quality and usability of local website applications.

1.4 Motivation

Designing and developing a framework to evaluate a website's performance, security, and accessibility while simultaneously employing Robotic Process Automation in the evaluation process that reduces the costs of usability evaluation. Furthermore, this proposed

framework also reduces the need to engage usability experts to conduct the assessment. Its usage will benefit the website engineers who lack such specialized knowledge and expand the evaluation's coverage of usability factors.

Many integrated tools within our framework use international standards as a reference to evaluate the security and accessibility of a website, which is challenging to accomplish when evaluated by humans. Moreover, using our proposed framework with RPA eliminates inconsistencies in detected usability problems, as well as misinterpretations and incorrect application of usability guidelines, which also overcomes the limitations of the human element mentioned by some researchers [4][5].

1.5 Thesis Organization

This thesis consists of five chapters, which are as follows:

- Chapter 1 provides a general introduction to the research topic, a discussion of the problem, the formulation of objectives, the definition of research questions, the declaration of the motivation, and the contribution of the research.
- Chapter 2 begins with a description of Human-Computer Interaction (HCI) and its related topics, including usability testing, performance, accessibility, and stress testing. In addition to a general background in Robotic Process Automation. Subsequently, it contains an analysis and review of the related works, the methodology, and the results for each study.
- The first section of Chapter 3 defines the study design and data description; the second section discusses the architecture, methodology, design, and workflow of the proposed methodology
- Chapter 4 presents the experiments and findings.
- Chapter 5 presents this work's overall findings and outcomes, summarizes the thesis contribution, and provides a potential future research area.

Chapter 2: Background and Literature Review

2.1 Background

The following sections include information on Human Computer Interaction, Usability Testing, Performance Testing, Security Testing, Accessibility Testing, and Robotic Process Automation.

2.1.1 Human-Computer Interaction and Usability Testing

HCI is essential as it will likely become a requirement for all types of products to be more successful, safe, practical, and functional. Additionally, it will improve the user's experience in the long term. As a result, it is critical to have someone with HCI skills involved in any product or system development phase. HCI is also required to keep products or projects from failing completely. HCI is essential when developing simple, intuitive systems that can be used by people with a variety of skills and knowledge and those who have not received formal training.

A significant concept within the field of HCI is usability, defined as how easy to perform a task. The idea of usability within human-computer interaction consists of three distinct aspects: effectiveness, efficiency, and satisfaction [6]. Effectiveness refers to the accuracy with which users accomplish goals; efficiency denotes the relationship between the users' accuracy and completeness of specific goals and the resources used to achieve them. On the other hand, satisfaction is defined as the user's comfort and positive attitude towards the system [7], [8]. Overall, companies must address a product's usability during the software development stage, as it may shift how users interact with the product. Usability testing allows design and development teams to identify issues before they are coded. The earlier the problems are identified and resolved, and the less expensive the fixes will be in terms of staff time and potential schedule impact.

2.1.2 Web Usability Testing

The three most important characteristics that distinguish a usable website, according to Nielsen and Loranger [8], are simplicity, naturalness, and ease of use. When good usability is combined with good web design characteristics, particularly visual attractiveness, the website positively impacts user behavior and the trust in the relationship between a user and the company [9]. This trust is also affected by the user's perceived usability, which is the user's perception of the usability of a website before they use it. However, this measure strongly correlates with webpage usability [9].

Researchers like Safavi [10] has demonstrated that users are more satisfied with websites with a user-friendly design and an easy-to-use interface. Montero et al. [11] have demonstrated how websites that violate usability guidelines confuse users and cost the businesses that created them money. According to research, a website’s usability depends on the market it is competing in and the level of competition. Thus, a website in the highly competitive e-commerce environment is more likely to lose a potential customer due to poor usability [11], then a website in a less competitive environment.

Usability testing is measured by users trying a website in person and observing their behavior. Users are presented with a scenario in which they must complete a task and answer some questions while observers watch and listen. This activity aims to evaluate users' satisfaction with the website's interface and determine whether users can use the interface to achieve their tasks. Then, collecting user feedback on how to improve the product and implement improvements to increase customer retention. There are numerous usability testing methods, such as hallway usability testing, in which the tester selects random people to test a website. Another method is guerilla usability testing, a quick type of usability testing in public places such as cafes and libraries. Again, individuals are chosen randomly and asked to

participate on the day, so they are not formally recruited ahead of time. Another standard methodology is remote usability testing, which allows participants to try the website whenever they want with no time or space constraints because observers are not required to be present during testing.

Although there are numerous benefits to usability testing, there are some drawbacks [12], like selecting a target audience. Usability testing is frequently conducted on a sizable audience, which can be tricky. It won't be easy to find target users in some age range to test the website, like building a website for 2- to 5-year-olds. It is also challenging to conduct the testing team usually puts lots of time and effort into testing and test data analysis. Traditional usability tests are costly, have a wide range, and take a lot of time to complete. A larger budget and consequently higher costs for the organization result from this. Moreover, the test outcome may be compromised because users are chosen randomly, and the testing is not entirely representative of real-life experiences. It is unrealistic to expect all participants to report reasonable and accurate inputs to obtain absolutely accurate results.

Automation usability testing tools can help the testing team to evaluate many major scenarios that cannot be assessed in the traditional ways. Heatmaps, Eye tracking, Accessibility testing tools, performance testing tools, and security scanning tools allow you to see how people can complete a given task on your prototype, website, or application

2.1.3 HCI standards

A heuristic evaluation is a usability inspection method for computer software that helps to identify usability problems in the user interface (UI) design. It specifically involves evaluators examining the interface and judging its compliance with recognized usability principles (the "heuristics") [12] A heuristic is a quick and practical method for solving issues or making decisions. Nielsen published a set of ten general principles in 1994 to help individual

evaluators make assessments to improve evaluation effectiveness. Professional evaluators in user experience (UX) design employ heuristic evaluation to systematically determine the usability of a design/product. They go through a checklist of criteria to identify issues that design teams missed. According to the Nielsen-lich heuristics, a system should:

1. Promptly and appropriately update users on its status.

2. Present information in the users' language and in ways that they can understand how the real-world works.

3. Give users control and easy access to undoing mistakes.

4. Maintain consistency to prevent user confusion between different words, icons, etc.

5. Error prevention. A system should either avoid conditions where errors may occur or warn users before they take risky actions (for example, by asking, "Are you sure you want to do this?").

6. Allow users to recognize options, actions, etc., by providing visible information, instructions, etc.

7. Flexibility and efficiency of use, so that experienced users can find faster ways to achieve their goals.

8. Contain only the information needed to complete the tasks and are clutter-free.

9. Give simple explanations of errors and solutions.

10. Help and Documentation. List of clear steps for resolving issues in lean

Both newly developed and already existing products can benefit from heuristic evaluation. Instead of usability tests, which demand much more participation, heuristic evaluation can be conducted in the first scenario. It is better to use both methodologies to ensure the highest usability of a product. Heuristic evaluation can be helpful in UX audit in the second scenario. Jakob Nielsen updated the “10 Usability Heuristics for User Interface Design” in

2020, and he highlighted that the ten usability heuristics developed in 1994 are still relevant and probably will be used in the future.

2.1.4 Performance Testing

Performance testing is a software testing process for testing the speed, response time, stability, reliability, scalability, and resource usage of a software application under a particular workload [13].

Several studies have found that users prefer websites that load quickly and respond to their requests immediately. As a result, a website's overall performance directly impacts its popularity and desirability. Therefore, when an enterprise performs extensive performance testing as part of the software testing process, it can quickly make the website popular after its deployment. Furthermore, the enterprise can deploy web applications most efficiently while completely eliminating the possibility of unexpected system failure.

Web performance has a significant impact on traffic and user experience. Page load speed is an essential indicator of web performance. Google's Think With Google published a paper detailing their findings on page speed, load times and bounce rate, and they employed a deep neural network to accomplish this with a large set of bounce rate and conversions data. The neural net, which had a 90% prediction accuracy, found that as page load time goes from one second to seven seconds, the probability of a mobile site visitor bouncing increases 113%. Similarly, as the number of elements—text, titles, images—on a page goes from 400 to 6,000, the probability of conversion drops 95% [14].It is worth noting that their benchmark load time is one second. While Google tends to not talk specifics about load time other than sites should be fast, one second seems to be the speed to aim for.

Nobody wants to waste time navigating a slow and ineffective website; slowness never attracts a good audience or keeps them on the site for long. When people visit a website, they want to be able to interact with it in a way that is convenient and useful to them. They want to find what they're looking for quickly and easily. For example, if you run a blog, your visitors expect to get information quickly and read your posts without waiting long periods. If you run an e-commerce site, you already know that your online customers want to quickly find the items they're looking for. Web performance has a direct impact on web traffic and user engagement. So, regardless of the type of site, it must have good overall performance and a fast-loading speed. Without this, your visitors will abandon your website rather than continue browsing, resulting in a high bounce rate. As a result, you will not have many readers or sell many products through your e-commerce sites.

Performance testing is vital in Search Engine Visibility as well. Google and other major search engines have already begun ranking websites based on their performance and load speed. Testers perform specific tests during performance testing to determine the load time of the web application and identify the factors that cause the website to load slowly. Desirability is another major factor, demonstrating that performance testing affects the visibility of a website on primary search engine result pages (SERPs). It also makes finding the web application easier for users. As a proactive measure, a company should strive to avoid downtime and crashes to improve the usability and credibility of its website in the long run. When application performance issues are identified during the development process, developers can more easily repair all these performance issues without putting in extra time and effort [15]

For the website's stability, performance testing allows professionals to evaluate the application's performance under sustained user load and based on standard parameters. As a result, testers can quickly identify performance bottlenecks and architectural issues impacting

the application's user experience after deployment. Furthermore, the performance testing results will assist the company in improving the website's stability by deploying it with adequate memory, bandwidth, and other resources.

For the Accessibility factor, testers can use spike testing to see how the application behaves when the number of users increases suddenly and drastically. They can even use robust performance testing tools to evaluate the application's behavior under varying user loads based on critical parameters such as devices, networks, and memory. As a result, an enterprise can quickly deploy additional resources to make the website available to many users while providing an optimal user experience. Web application performance testing should not just focus on server-side speed but also on user/client performance. Understanding where bottlenecks occur in our web apps is critical for identifying areas for improvement and providing a better user experience.

Making sure that testers' test scenarios are set up to reflect the conditions our web application will encounter in the real world is one of the essential aspects of performance testing web applications. It's always a good idea to review normal and peak traffic conditions if the application has already been released into production, we should probably have a good idea of the traffic conditions our application will be exposed to. The ability to script and automate our tests is a crucial component of performance testing web applications, as the days of using actual people to conduct our tests are long gone. Instead, performance tests can be completed using scripting tools that can navigate our applications like a regular user. Many testing tools like LoadRunner, LoadUI, and JMeter can conduct performance testing.

A particular kind of web performance test is a scalability test, this test is crucial if you want more users and visitors to interact with the web system and having a scalable website that can be updated in real-time may be a huge advantage in today's digital world. A website's

potential weaknesses can be found, and the areas that need to be strengthened to accept updates and changes and be more scalable can be found by performing web performance testing, which is how a website test of this nature improves a website.

When it comes to advanced performance testing methods, it is essential to simulate the experience by creating scripts that emulate critical user flows and scenarios and then testing those scripts against high levels of concurrent and simultaneous users from multiple points worldwide. We discussed the various performance testing tools, and some offered more user-friendly scripting options. Some on-premises solutions and tools necessitate extensive knowledge of specific technologies. In contrast, others, such as LoadView and LoadNinja, provide a point-and-click scripting tool that requires no prior scripting experience.

When it comes to advanced performance testing techniques, having scripting experience is vital because it will make the process faster and more accurate. There are many advanced techniques like determining how to set up your load curves for your test. Some tools only allow you to set the maximum number of users over a specified time. Still, others, such as LoadView, provide multiple load curve options, including the ability to increase or decrease load levels during a test to see how your system responds in real-time.

All software and applications should be tested as part of the performance testing process to ensure that it satisfies performance and business requirements. Benchmark and baseline testing are a part of that procedure. Baseline testing is done to ensure a consistent product. To accomplish this, a team will test the software and assess various performance factors, including code, network, and hardware, and the test's results are noted and documented. If the program or application is updated, it will be tested similarly to compare the outcomes to the previous test. Benchmark tests compare an application's performance to other applications or an industry standard to ensure that it meets or exceeds quality standards. This is especially important for organizations that want to set quality standards or meet specific service-level agreements

(SLAs) for their applications, software users, and partners. Benchmarking is driven by the business and organization, which helps build trust with potential customers and positions your company as a leader in your industry.

2.1.5 Security Testing

Security testing is a type of Software Testing that uncovers vulnerabilities of the system and determines that the data and resources of the system are protected from possible intruders to ensure that the software system and application are free from any threats or risks that can cause a loss [16]. There are many types of Security Testing, including Vulnerability Scanning, Security Scanning, Penetration Testing, Risk Assessment, Security Auditing, Ethical Hacking, and Posture Assessment. Security testing is essential Application Security Testing is critical for all businesses and industries because a data breach leads key clients to lose trust and tarnishes a company's brand in the long term. So, it maintains the brand's image, prevents the disclosure of sensitive information, helps organizations ensure that customers' data are secure, and increases customer trust in the organization.

Data availability, integrity, and confidentiality make up a website's security. Most of the data is represented by the website's files and databases. A DDoS attack may frequently disrupt website availability and prevent authorized users from accessing it. Availability refers to unhindered and quick access to the website and its content. Integrity includes the security of the stored data; for instance, hackers must be unable to alter data or manipulate any information on the website. Keeping sensitive information protected adequately, such as website users' logins and passwords, so that only authorized people may access it is what confidentiality refers to. Complete verification of the website's availability, integrity, and secrecy must be performed. A website can effectively secure users' information by maintaining privacy and compliance with the relevant data protection laws and regulations.

The General Data Protection Regulation (EU GDPR) is a European law that aims to protect Personally Identifiable Information (PII) of residents of Europe by increasing transparency. This law increases transparency regarding data handling, giving the website's owner the power to manage their PII data and asking businesses and organizations to return and then delete any PII that relates to them. GDPR compliance refers to compliance with all the EU GDPR. The law was enacted in response to the alarming increase in data breaches, leaks, and the shady use of PII for business or even illegal purposes without individuals' consent.

Another security standard is The Payment Card Industry Data Security Standard (PCI DSS), a set of requirements to ensure that all companies that process, store, or transmit credit card information maintain a secure environment [18]. First, define the tested organization's Cardholder Data Environment (CDE) to verify PCI DSS compliance. The CDE scope defines the corporate network and cloud storage segments where credit card data is stored or processed. It is critical to define our CDE scope properly; otherwise, we may overprotect or overspend on PCI DSS compliance, resulting in fines or significant financial losses.

HTTP Headers scanning is another major verification in security testing. In order to pass additional information about the sent content, its format or structure, or to specify specific security or privacy measures like setting the Do Not Track (DNT) directive, HTTP headers which are a component of an HTTP request sent by a web browser to a web server, or vice versa. Some HTTP headers could be unique to a given browser or web server. Due to the requirement to maintain adequate website functionality, some security headers, such as Content Security Policy (CSP), are somewhat difficult to configure. However, if properly implemented, CSP may mitigate a variety of XSS (Cross Site Scripting) and other attacks by preventing untrusted or insecure content from running in the user's web browser. HTTP headers may be necessary to meet the needs of a website owner or web browser user. Some security-related

headers (on the web server side), for example, X-XSS-Protection or even more robust Content Security Policy (CSP), are recommended to improve web application and web server security by mitigating some XSS and associated attacks. Other server-side headers, such as X-Powered-By or Server, on the other hand, may disclose internal or sensitive information and must be removed. The DNT (Do Not Track) header is becoming increasingly prevalent regarding client-side headers.

Many security standards and compliance requirements, such as PCI DSS, explicitly mandate the presence of a Web Application Firewall (WAF). Even though the websites, web services, and APIs are vulnerable to SQL injection or other typical security problems, WAF can protect them. Attackers cannot exploit a vulnerability in our website's source code if a correctly configured WAF secures it, whether on-premises or in the cloud. Modern WAF also minimizes the number of malicious bots, speeds up the website performance, and bans IP addresses known to be infested with malware or involved in DDoS attacks.

Security of a web Content Management System (CMS) typically refers to the safety of the web software used to run a website; two examples of web CMSs are WordPress and Drupal. The security of a CMS is ensured by its developers, who put security controls and defenses in place to fend off known attacks like SQL injections and XSS. Website owners must maintain security by the timely installation of security patches, using different, strong passwords, and ensuring that the website hosting is secure. No matter how strong the CMS security is, if the web server security is compromised, the website will be immediately in the hands of the attackers. For example, if FTP access or the admin password to the server is compromised. The most critical factor in determining WordPress security is whether our installation of WordPress CMS, its plugins, and themes are all up to date. First, make a complete list of all WordPress components and plugins and make sure they are all up to date before we test the security of WordPress. Next, check vulnerability databases to find if any WordPress plugins or extensions

have security vulnerabilities that are known but not yet fixed. If they do, deactivate the corresponding parts.

Verifying that the Drupal CMS and all plugins used in our Drupal installation and all other third-party software are up-to-date is the first step in the Drupal security check. Then, check various vulnerability databases for details on flaws or vulnerabilities that have been publicly disclosed but not patched. In case we found such a component, website developers should disable, or quickly deactivate it until the vendor releases a patch. Another verification is to ensure that all privileged users have strong, one-of-a-kind passwords. Additionally, Drupal must be installed from a safe web environment, with limited access to configuration files, and we have a workable method for regularly updating Drupal's security.

SSL testing is one of the leading security testing types. It is a set of network protocols called Secure Sockets Layer (SSL), which aims to encrypt data transfer through other, more advanced protocols that carry web content, email, or other information. Technically, an SSL certificate is a file that is kept on the server. In practice, an SSL certificate is used to encrypt and decrypt information transmitted or received by a web, email, or another server that supports SSL/TLS encryption. Furthermore, specific SSL certificates may authenticate the website owner's identity, assuring visitors that they are dealing with a legitimate website they can trust. An SSL certificate is essential to enable HTTPS data encryption between our website and visitors. Google, Mozilla, and many other organizations may provide warnings or block access to a website lacking HTTPS encryption.

Security of web browsers is another crucial area to pay attention to. Information can be saved in a web browser for user’s convenience, but it may eventually be accessed by others. Therefore, it offers a significant surface area for email accounts, usernames, various passwords, and private or business information to be exposed. Attackers often target the web browser to hijack or sniff on the web traffic from it, they could also use it to gain access to the device's

files or the device itself. The most vulnerable parts of a web browser are Connections to DNS servers, websites, and other online resources, Browser plugins and Browser-specific vulnerabilities [[17]. Users can improve the Web Browser Security by using the latest web browser version, restricting user access, and using of custom security settings such as disabling cookies, JavaScript, Plugins/Add-ons, and Pop-up windows and enable them only if a trusted site needs them.

2.1.6 Accessibility Testing

The goal of web accessibility is to create and maintain more user-friendly websites for all users, including anyone with disabilities, and it means that more accessible content can be offered by websites that have implemented accessibility principles [18]. Usability and accessibility share overlapping principles. Although it may appear that they are both working toward the same objective, there is a difference between the two technological aspects.

People without disabilities can also benefit from web accessibility, including the elderly with aging-related changes in their abilities, people with "situational limitations" like being in bright sunlight or in a place where they can't hear audio, and people using mobile devices like smartwatches, etc., and people with slow Internet connections.

Web Content Accessibility Guidelines (WCAG) standards suggest improving your website's content usability and accessibility for those with disabilities. Additionally, adhering to WCAG standards frequently results in web content that is more enjoyable and interactive. WCAG standards have 12-13 guidelines. The guidelines are organized under four principles: perceivable, operable, understandable, and robust. Each principle has testable success criteria at three levels: A, AA, and AAA [19]. There are many major test cases for all end-users that should be executed to verify the accessibility of a website such as ensuring that color contrast ratio is maintained, verifying that audio/video content is properly audible/visible, and meaningful multimedia caption is provided, ensuring that instructions are clearly given, content

is clear, concise, and understandable. There are other major test cases that related to the website's language, when a website uses multiple languages, the most used text-processing language is used as the default. According to the WCAG 2.1 – 3.1.2 – Language of Parts (Level AA) Guideline, screen readers need to be notified when the language on a page change [19].

It is crucial that your website is easily accessible to make it more acceptable and user-friendly. There are various accessibility testing tools available to check the website's accessibility like Wave, TAW and aDesigner.

2.1.7 Robotic Process Automation

Robotic Process Automation is the technology that allows anyone today to configure computer software or a "robot" to emulate and integrate the actions of a human interacting within digital systems to execute a business process [20]. RPA enables software developers to build robots, or "bots" that can learn, imitate, and conduct business processes that follow established rules. By analyzing human digital behavior, RPA automation enables users to build bots. Robotic Process Automation software bots can interact with any application or device in the same way humans do - except that RPA bots can run 24 hours a day, nonstop, much quicker, and with high precision and accuracy [21].

Robotic Process Automation bots possess the same level of digital proficiency as humans. Therefore, RPA bots can be considered a form of Digital Workforce capable of interacting with any device or application. Bots can perform tasks such as copying and pasting, scraping web data, performing calculations, opening, and moving files, parsing emails, logging into programs, connecting to APIs, and extracting unstructured data. Additionally, since bots can adapt to any interface or workflow, automating does not require changing business structures, software, or existing processes.

RPA bots are simple to deploy, manage, and share. If the user is familiar with performing a simple video recording on their computing device, they can configure RPA bots. It's as natural as pressing the record, play, and stop buttons and dragging and dropping files around at work. In addition, RPA bots can be scheduled, cloned, customized, and exchanged through an enterprise to automate business processes.

Robotic process automation (RPA) is mainly intended to assist with office-related functions that often involve the ability to perform several tasks in a particular order. It develops and deploys a virtual robot capable of initiating and operating other software. In many ways, the basic idea is similar to conventional manufacturing automation, which focuses on automating a particular portion of a workflow — or even a single task — by developing a robot that performs that task. Production processes always involve the same type of repetitive operation. Since the data is being manipulated through multiple systems and tools, a physical robot is not needed, but rather a software bot.

RPA reduces processing costs while increasing per-employee output by shortening processing times and reducing costly errors. The typical savings from these efficiency improvements range between 25% and 50%. Software robots can increase a team's productivity by 35 to 50 percent. They can also work more quickly, reducing the time it takes to process data by 30% to 50%. You can easily accomplish more at the same time when low-level and back-office work responsibilities are divided between humans and robots [21].

RPA architecture is made up of three primary components, which are configured to enable ARP software to function as intended. The components include Control Room, Bot Creator, and Bot Runner [22]. Figure 2.1 shows an overview of the RPA components.

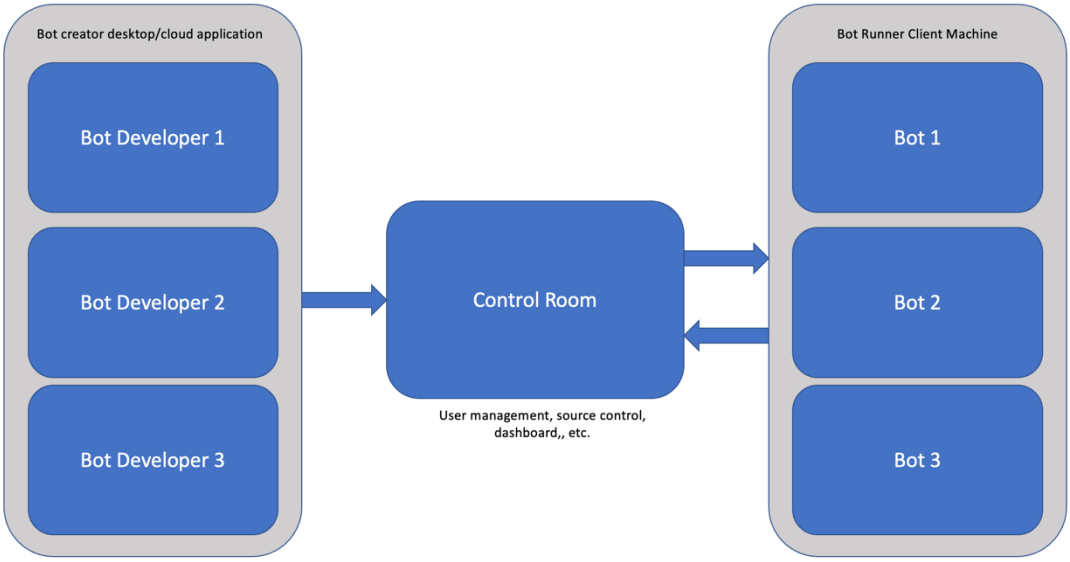


Figure 2.1: Robotic Process Automation components

a. Control room

The control room is a web-based interface that controls the automation of the usability tests. In other words, it's the Server that controls Automation Anywhere bots. Moreover, it is designed with a source control feature in which the code for the bots is managed so that it becomes easy to distribute the code across various systems.

b. Bot Creator

This component will be used to create the bots for automated usability tests – desktop applications are used to accomplish this. The usability test codes to be used by the bots are stored in the control room. Different developers can create individual tasks/bots. These bots are being combined and will run at the same time.

c. Bot Runner

The Bot Runner is the central component on which the created usability bots are executed. It is possible to run multiple bots at the same time. The bots report to the monitoring or the control room with the execution logs and pass/fail status of the usability tests created.

2.2 Literature Review

The following section reviews many previous studies and experiments in the field of HCI and Usability Testing.

2.1.1 Related Work

Human-computer interaction research includes several contributions to usability techniques for web and mobile applications. There are different goals of these studies such as ensuring that web applications are fast, reliable, secure, accessible, and up to date because they are ubiquitous and can deal with a huge number of users.

In [23], researchers propose a multi-criteria evaluation framework for assessing and improving the usability of services provided by websites related to the Hajj and Umrah system, such as the Ministry of Hajj and Umrah (MHU) and the General Presidency of Haramain (GPH). According to the researcher's findings, the MHU website scored higher on all evaluation criteria, with an average score of 3.2 out of 5, compared to a GPH with a score of 2.8. The results also revealed a significant weakness in both websites regarding the "coverage" criterion, with an average score of 0.7 out of 5. Furthermore, the user satisfaction questionnaire results revealed that both websites achieve high levels of satisfaction, with the MHU website scoring higher at 4.3 and the GPH website scoring lower at 3.7. Finally, the heuristic evaluation method revealed 17 usability errors for both websites, with seven found on the MHU website and the remaining ten on the GPH website.

The study [24] aims to improve the user experience and security of users' personal information on social networking websites. Two IT specialists evaluated the Facebook website, revealing significant privacy and security problems. In addition, researchers highlighted that Facebook's design has evolved, so users should constantly examine their privacy settings. Recommend that Facebook ask users to evaluate their privacy settings annually, and support

that by publishing a video on each user's wall that explains the stages in depth and demonstrates how they should be updated based on the user's needs.

In [25], they use the trunk test to evaluate the website navigability of recognized private educational institutes in Pakistan to determine how the websites are designed to be comfortably navigable and usable for the user. Using the trunk test approach, more than a hundred private sector educational websites in Pakistan are tested and evaluated. This research shows that poorly designed interfaces and websites dissatisfy users and force them to abandon the website in the middle due to poor user experience and navigational issues. Furthermore, provincial results show that educational websites in Giglit need to be improved. However, more than half of educational websites in Pakistan adhere to general web navigation and usability standards. Therefore, modifying a website that successfully passes the trunk test for navigation can increase user satisfaction and make it more likely to be used for its intended purpose.

In the study [26], they used three automated methods to analyze the educational websites of KPK, Pakistan, including Qualidator, which rated the usability of websites by analyzing accessibility, and search engine optimization, and usability. Website Grader was the second tool used, and it assessed websites based on their mobile readiness, performance, SEO, and security. Website Analyzer is the third tool, and it considers the website for accessibility, content, design, mobile readiness, page analysis, performance, SEO, security, and usability. According to the results of the University of Peshawar's Qualidator website, Peshawar is better than other websites. According to the results of the University of Peshawar's Qualidator website, Peshawar is better than other websites. Bannu is better than other websites, according to the USTB Website Grader website results. At the same time, Kohat is better than other websites, according to the KUST Website Analyzer website results. Researchers' findings also reveal the weak spots of the chosen websites; if concerned authorities work on these, their websites will undoubtedly become more usable.

An interpretive (qualitative) approach was used [27]. The primary goal of this study was to create a website design technique and evaluate whether or not it will assist designers in meeting the needs of customers, including determining whether participation by both end-users and client-customers will benefit the website development process, considering how designers can address issues of usability, HCI, iteration, and use of actual interaction data, and Determine whether or not this new methodology will meet the needs of the Western Australian website industry. Most industry participants in the interviews and questionnaires felt that Usability and HCI aspects would benefit the firm by increasing sales, lowering website development and maintenance expenses, and making clients happy with the website's outcomes.

Using the expectancy disconfirmation theory, the study [28] investigated Hong Kong passengers expected and experienced levels of performance in terms of website usability in the context of Hong Kong-based travel websites. They found that there is a significant gap between groups that were expected and those that were experienced, as they indicated that the perceived performance and expected performance differed significantly. Moreover, researchers concluded that travel agencies should verify that the internal links work consistently with other issues besides layout, graphics, text colors, and color contrast.

In [29], researchers present a new usability model that measures website usability as a calculated number based on specified indicators and factors that may be assessed using Nibbler, Gtmetrix, Checkmycolours, and PowerMapper tools. The proposed model contains 24 measurable criteria, each with a specific metric, distributed and replicated among nine factors hierarchically to achieve the weight concept. Measures are aggregated in the usability equation to determine the approximation value of website usability. This study also includes a case study of Jordan University's website, which has an estimated usability value of 72.18.

The study [30] assessed the accessibility, usability, and security of prospective students' university websites from Europe, North America, and Oceania. Researchers concluded that

there is a relationship between website presence and usability, safety, and accessibility as higher-ranking university websites in terms of web presence have fewer accessibility issues. In addition, more than half of the websites tested for page performance control have performance numbers below the threshold that is considered acceptable. And according to the testing of SSL encryption findings, more than 90% of websites offer safe connections.

In [31] the researcher assesses the accessibility, usability, quality performance, and readability of all Turkish forms and personal university websites. The findings show that Turkish institutions should invest more resources to make their websites accessible, useable, high-quality performance, and legible for all potential consumers. Only ten state university websites and four private university websites achieved conformance Level A out of 110 state university websites and 69 personal university websites. AChecker was the tool used for accessibility testing. In addition, website Pulse, Pingdom, and Page Speed Google were used to measure download time, response time, and mobile loading speeds. And 19-dichotomous web usability was applied to the homepage of each website to evaluate usability testing.

Many studies evaluated the usability of government websites such as [32] where accessibility, usability, and security evaluation were assessed. The findings revealed that none of the 25 websites of the evaluated Hungarian public sector organizations could totally meet the Web Content Accessibility Guidelines (WCAG) standards. Furthermore, that half of the websites had just the lowest level of compliance in usability testing. Moreover, security scanning results revealed critical issues in many websites where around half of all websites have out-of-date server versions and programming languages. The tools used are WAVE for Accessibility checking, GTmetrix checker for usability tests, and the Security test verified using the Sucuri checker tool.

Researchers in [33] concluded that usability is a very low priority in Tanzania e-government websites. Researchers assessed the usability, accessibility, and web security of 79

e-government websites in Tanzania using automated assessment technologies such as Pingdom, Google Speed Insight, Wave, W3C Checker, and Acunetix. The findings revealed a high number of usability issues, where all websites have broken links, and all have accessibility issues and violate w3c Web Content Accessibility Guidelines (WCAG) 1.0. Additionally, about two-thirds of the sites have loading time issues, and around half of the assessed websites have one or more high-severity vulnerabilities.

The study [34] introduces usability measures for government portal websites in Zhejiang Province as samples. Then, it uses them in an experiment to put them into practice. Twenty-four students participated in this study, all of whom had had at least two years of experience using the internet. Still, none of them have utilized government websites in Zhejiang Province.

The experiment identifies specific usability issues with the offered websites, and the study concludes with numerous practical recommendations for improving government portal websites. This paper does not employ technical indicators, widely utilized and subjective evaluation methods like TAM. Instead, it uses usability as a criterion for evaluating government websites. The study examines existing research and employs an experiment to conduct a usability assessment, after which it offers usability-related website design principles. The report investigates and reviews 11 local government websites in Zhejiang Province. Finally, the study provides recommendations for improving government websites as the researchers found that some websites need further improvement.

The study [35] aims to help identify various vulnerabilities. Researchers collected data using multiple tools and methods to examine the security gaps. Using Acunetix, Zed Attack Proxy (ZAP), Wafw00f, and Wappalyzer, Researchers selected websites from the government and banking sectors to conduct vulnerability evaluations. Researchers identified the top five significant vulnerabilities by analyzing the security-related records. They also demonstrated

mitigation solutions for these vulnerabilities, as the Bangladesh government might concentrate on those problems and be more concerned about future development.

Researchers in [36] conducted a security audit of 16 government websites in this report. First, researchers gathered and verified information about websites. They then discovered data regarding webpage environments. Following that, they began the vulnerability assessment by doing vulnerability scanning and SSL encryption evaluations. They also performed a manual content analysis to ensure that website had a published security and privacy policy. A safety evaluation methodology that combines vulnerability assessment, content analysis, and SSL evaluation was created as an outcome.

Another study used open-source tools for security evaluation. Researchers in [37] reported their findings from assessing the security testing of 150 Saudi websites with open-source tools like W3AF, Skipfis., Wapiti, Andiparos, and Powerfuzzer. Many vulnerabilities with varying degrees of impact were discovered in the selected websites. Many critical issues were reported on many websites. Researchers found that governmental websites are safer than eCommerce websites, and most of the detected vulnerabilities can be addressed quickly and cheaply

Initiation work toward a security-threat model for HCISec security-usability analysis was presented in [38]. Researchers have used use scenarios and threat scenarios to identify system and external aspects dangerous to a system's usability, security, or both in the proposed security-usability threat model for performing security-usability evaluations. For example, threat scenarios are used to find vulnerabilities that might allow non-malicious users to compromise a system's security. In contrast, use scenarios are used to find problems that could make a system less usable. Researchers found that the users are more likely to conduct the threat scenarios of a system than the usage scenarios when the threat scenarios are more functional. This is because external factors may also do actions they may not usually take.

In [39], 11 models that have previously been used from the year 2000 to 2018 to assess the usability and security of e-commerce websites were reviewed.

Some studies have developed models for analyzing the features of e-commerce websites. Still, these models offer limited insight and don't consider all usability and security factors. This study found that there isn't a single comprehensive model that can measure the security and usability components.

SEOptimer, Website grader, and Qualidator automated tools were employed in [40] to evaluate a website's performance using automated evaluation tools. The case study used the website cosmeticsotop.com which was assessed in terms of Performance, Usability, Search Engine Optimization: SEO, Social, and Security. The highest-level items, Usability, Mobile-friendliness, and Security, all received a perfect score. Furthermore, the effect calculated each aspect; it was discovered that the highest-level things were Usability, SEO, and Security, each with a score of 100%. Furthermore, a calculated result revealed that the most elevated level item - Accessibility - has a rating of 75.1%, Usability has a rating of 70.5%, and SEO has a rating of 67.1%. In addition, several suggestions to improve the website have been made, such as combining files and minimizing caching to speed up the loading of frequently used content and remove any unneeded JavaScript or CSS

Researchers in [41] briefly present the methods used for finding usability issues in a given web application, to deduce the technical reasons, and improve the entire website’s performance by concentrating on the problems identified, like response time capabilities when increasing the usage workload

The researchers in the study used JMeter to measure the application's behavior as the number of users gradually increased, and the reported error % was calculated for each case scenario. The study findings revealed that the system's usability was improved by using the proposed methodology to find and resolve the reported issues.

Identifying usability issues in Mobile Web Applications is the primary goal in [42] based on Usability Guidelines for Responsive Web Design in Mobile Web Applications and to propose an improved version of Usability Guidelines, particularly in terms of performance by analyzing, identifying, offering, implementing, and measuring new performance attributes in Usability Guidelines. The researchers measured the versions of two case studies using First Contentful Paint (FCP), Speed Index (SI), Time to Interactive (TtI), First Meaningful Paint (FMP), First CPU Idle (FCI), and Estimated Input Latency (EIL), and the result shows a better score at 90-100 (fast-GREEN) with the proposed performance attributes compared to another website without it, which averages at 50-89 (average-ORANGE).

2.1.2 Discussion

An overview of research studies regarding usability evaluation, performance testing, accessibility testing, and security testing has been outlined and presented. The numerous and diverse studies in this field in various sectors are unequivocal proof of the significance of HCI and usability testing for businesses and users. It is worth noting that there is a wide range of evaluation methods in the studies presented in the previous section, including manual and online automation tools. Various online automation tools improve traditional manual usability evaluation methodologies by conducting various non-functional tests that manual testers cannot verify. All previous automated methods proposed used online tools from different sources to complete the usability, performance, security, and accessibility tests. Gtmetrix was the most utilized performance tool. However, other web tools such as Website Pulse, Pingdom, and Google Speed Insight were also utilized in other studies. The primary benefit of using Apache JMeter in our research is the ability to run performance and functional testing with a single tool on any essential online application, including web services, databases, FTP servers, and web servers. JMeter can be used to test local and cloud-based apps as well. Moreover,

JMeter can be easily integrated with RPA, and DevOps and test management tools such as JIRA and Jenkins.

Sucuri and Acunetix were the most utilized security testing tools, such tools are necessary for developers to identify security vulnerabilities before releasing a product to end users. In our proposed framework, we preferred to utilize ImmuniWeb because it validates the website against many international laws and standards, like PCI DSS and EU GDPR. Moreover, it has an open-source version that can be easily customized and integrated with RPA bots.

The researchers in [39] highlighted that designing one comprehensive model that can evaluate all the usability dimensions along with security components is necessary where this will improve a website's usability and security. Furthermore, designing and developing one framework integrated with automation tools to verify all these tests in one place with a detailed report based on HCI guidelines is helpful for businesses and test engineers. Therefore, in this study, we will develop and enhance a framework by employing Robotic Process Automation bots. RPA Bots can be integrated with different automation tools and API services to evaluate the usability, performance, security, and accessibility. Moreover, it can be used to connect with different databases, store and read data from a database and evaluate the test results based on HCI guidelines and standards which will help businesses in evaluating their products more effectively, save time, money, and effort.

Chapter 3: Methodology

3.1 Study Design

The study used the usability testing framework based on robotic process automation for performance, security, and accessibility testing of website applications. The framework was designed and implemented as an automated user experience testing model that allows developers and system users to run testing bots and visualize the results of human-computer interaction usability standards and guidelines. The framework comprises three main user experience models (Performance, Security, and Accessibility), which are used to test the user experience of educational, governmental, commercial, and media website applications.

3.2 User Experience Security Testing Model

The user experience security testing model was designed in reference to the ImmuniWeb security testing software, a machine learning, and Artificial Intelligence (AI) security testing tool that assesses website security and vulnerabilities [43]. The model used the General Data Protection Regulation and the Payment Card Industry Data Security Standard (GDPR & PCI DSS) security testing standards in designing the BOT testing model. The Bot was designed to read, and analyze the CSP & HTTP Headers Check, Website content management system (CMS) Security Test, and in addition, the supports of Secure Socket Layer (SSL) testing, which allows users to verify Web Server SSL, Email Server SSL, SSL Certificate Test, and PCI DSS, HIPAA, and NIST Test. Furthermore, the bot supports the reading and analysis of the Dark Web Exposure Test, including Dark Web Exposure Monitoring, Phishing Detection, Monitoring, Domain Squatting Monitoring, and Trademark Infringement Monitoring.

SSL tests provided by the ImmuniWeb are designed to identify and highlight configurational, implementational, and cryptographic issues with SSL/TLS protocols and supporting software. Furthermore, it provides a free SSL test to discover any known security and cryptography vulnerabilities in our SSL/TLS-enabled services (e.g., HTTPS or SMTPS servers) and to determine whether our SSL/TLS-enabled services are secure. When using HTTPS encryption, SSL/TLS security testing assures that our clients and other website users are effectively secured from Man-In-The-Middle (MITM) attacks and different types of data interception.

For the security testing, the model requires a pre-identification of the type of security testing (Website security test and CMS, or SSL testing).

3.2.1 Website Security and CMS Test Bot

This bot verifies the website security testing based on the GDPR and PCI DSS compliance, CMS, CSP security, HTTP Headers Check, and WordPress & Drupal Scanning. Website security test category contains of the following tests:

1- **PCI DSS Compliance**: The website's application should fall into the CDE scope (Cardholder Data Environment), which includes the following requirements:

- **Requirement 6.2:** Installing applicable vendor-supplied security patches and all other critical security patches within one month of release to ensure that all website components are protected from known vulnerabilities.

- **Requirement 6.5**: Protection against all injection flaws for all applications, all buffers overrun vulnerabilities, all insufficiently secure cryptographic key storage, sufficiently secure communications and traffic, all improper error handling behaviors by users, and all threats rated as “high risk” per Requirement 6.1, cross-site scripting (XSS) risks, improper access control measures across,

cross-site request forgery (CSRF) and authentication management flaws across web apps [44].

- **Requirement 6.6**: The requirement for reviewing applications or installing web application firewalls is intended to reduce the number of compromises on public-facing web applications due to poor coding or application management practices.

2- Content Security Policy (CSP) Test

a. The Content-Security-Policy defines allowed sources for each type of content (e.g., text, images), helping to defend against XSS attacks. It also controls the browser's settings, from sandbox enforcement to the value of the HTTP Referer header [45].

b. Content-Security-Policy-Report-Only: Allows developers to experiment with policies by monitoring without enforcing their effects.

4. HTTP Headers Check

The HTTP headers check includes the missing required HTTP headers, which might weaken the website, such as Strict-Transport-Security, Content-Security-Policy, and X-Frame-Options—furthermore, missing other optional titles, such as Access-Control-Allow-Origin, Public-Key-Pins, Expect-CT, Server Header, X-Powered-By that is commonly used to display web server's software or its components (e.g., programming language or CMS), X-Frame-Options that prevent a well-known vulnerability called Clickjacking, the X-Content-Type-Options that is important to protect against MIME sniffing vulnerabilities, Permissions-Policy that allows developers to selectively enable and disable the use of various browser features and APIs (e.g., camera, location, etc.).

5. Web Software Security Test, WordPress & Drupal Scanning

Web Software Security Test, WordPress & Drupal Scanning testing model includes Fingerprinted CMS & Vulnerabilities that are used to ensure that all components of CMS like JS libraries were fingerprinted using non-intrusive techniques are not guaranteed to be accurate.

3.2.1 SSL Security Test Bot

The SSL test includes Web Server SSL Test, Email Server SSL Test, SSL Certificate Test and PCI DSS, HIPAA & NIST Test:

- **PCI DSS Compliance Test (PCI DSS 3.2.1, Requirements 2.3 and 4.1):** To verify PCI DSS Compliance and make sure that all the certificates provided by the server are trusted. And make sure that the list of all cipher suites supported by the server (e.g., TLSv1.2, TLSv1.1, etc.) are configured correctly. Supported Protocols should also be verified, including all SSL/TLS protocols supported by the server like TLSv1.1 and TLSv1,2, and consider dropping for all non-compliant protocols with PCI DSS requirements like TLSV1.0.

- **HIPAA and NIST Compliance Test**

    i. SSL X.509 Certificates is a critical public infrastructure (PKI) standard for secure Internet communications. Many common cryptographic protocols, such as TLS/SSL, ISAKMP/IKE, and IKEv2, use it to offer strong cryptographic entity authentication.

    ii. The server should be configured to support Online Certificate Status Protocol (OCSP) stapling: The OCSP is an internet protocol that examines a certificate's validity status in real-time. It's a replacement for Certificate Revocation Lists (CRLs). By merging two requests into one, OCSP Stapling enhances the connection speed of the SSL handshake. This reduces the time it takes for an encrypted webpage to load. In

addition, the end user's anonymity is protected with OCSP Stapling because no connection to the CRL is made for the OCSP request. The CA will only view OCSP requests from the website, not from its users, rather than seeing which websites a user has visited.

iii. Supported Ciphers, supported protocols, and Supported Elliptic Curves, EC_POINT_FORMAT Extension, and Diffie-Hellman Parameter Size

- **Industry Best Practices Test**: This category contains extra necessary validations from ImmuniWeb, including DNS Certification Authority Authorization, Server has Cipher preference, Always-On SSL (AOSSL) verification, supporting of client-initiated secure renegotiation, verifying that the server HTTP site does not support TLS compression supports redirect to the HTTPS version and verifying those server providers HTTP Strict Transport Security (HSTS).

The Bot was able to access, run and analyze the security testing through an API integration program with ImmuniWeb and Automation Anywhere Platforms. In addition, the output data was stored in a framework database system in human-readable format through the analysis and visualization tool.

3.3 User Experience Performance Testing Model

The user experience performance testing was designed to test the website's performance in reference to human-computer interaction standards and guidelines. The Bot performance testing provides the stakeholders with the data and information related to website speed, response time, reliability, scalability, stability, and resources usage under various loads. The following parameters were included in the BOT programing for performance testing:

Table 3.1: Performance Testing Measurement Parameters

Number	Parameter	Description
1	Average Response Time	The estimated time from the user request to the system response
2	Throughput	The total amount of completed transactions during the test period. Throughput = number of requests divided by time
3	Number of Samples	The total number of samples sent during the run time
4	Error Rate	Percentage of errors in the samples
5	Network (KB/sec):	The metric KB/sec is simply the throughput measured in bytes. As a result, KB/sec = (Throughput*Average bytes)/1024. In this case, Average bytes refers to the average value of the sample response in bytes, and 1024 is used to convert the value (Throughput*Average bytes) to kilobytes.
6	Min Response Time	The sample takes the minimum response time in milliseconds, representing the server's fastest response during the runtime.
7	Max Response Time	The maximum response time in milliseconds taken by the sample represents the server's slowest response during the runtime.

The Bot analysis identified the overall website performance, including four leading performance indicators: long load time, Poor response time, Poor scalability, and Bottlenecking

(throughput decrease due to coding error or hardware issue). The Bot program was integrated with the JMeter platform, a Java opensource software designed to test website applications' functional behaviors and performance. The JMeter supports protocols like SOAP, FTP, HTTP, LDAP, JDBC, and JMS. The open source can be used to test a wide range of applications, including Web applications, web services, databases, shell scripts, etc. [46]. One of the significant advantages of JMeter is that it is customizable software, where developers can customize its functions and develop their tools to meet specific performance testing requirements. Furthermore, it supports distributed load testing capabilities that allow us to set up a master-slave configuration for running load tests across multiple machines. Another advantage of JMeter is the recording feature, enabling users to record HTTP/HTTPS traffic to create test plans. In addition to the use of Proxy Server, which allows JMeter to monitor and record your actions as you navigate through your web application with your regular browser [47]

The RPA Bot was integrated with the JMeter through the non-GUI mode by using command lines API, which is useful for developers and test engineers to reduce the consumption of resources or memory, especially for heavy load testing.

3.4 User Experience Accessibility Testing Model

Websites accessibility means that people with different abilities can access, interact, understand, and navigate websites smoothly. However, according to human-computer interaction standards and guidelines, the technically usable website might be inaccessible. Therefore, website accessibility was considered in our Bot user experience testing framework. The Bot accessibility testing involves measuring the ease with which users can perform and complete website tasks without difficulties. Our Bot accessibility testing includes visual,

hearing, and physical impairment measures. The following parameters were considered through the Bot design and implementation:

1- **Visual Impairment**

   a. Ability to use site using a screen reader.

   b. Ability to adjust color contrast.

   c. Ability to adjust font and page sizes.

   d. Disabling flashing, rotating, or moving functions.

2- **Hearing Impairment:** Ability to see audio content—closed captioning, etc.

3- **Physical impairment:** Ability to navigate websites without using a mouse.

Figure 3.1 shows the main accessibility testing parameters

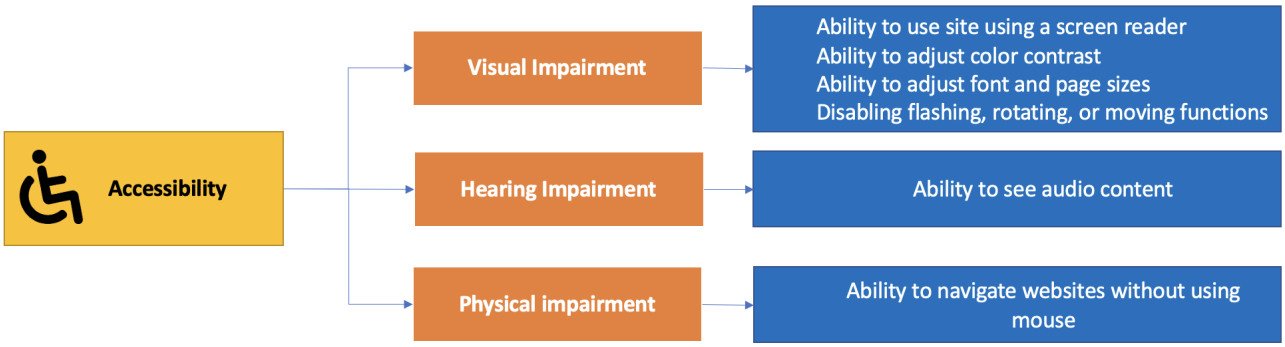


Figure 3.1: Accessibility testing parameters

The accessibility Bot program was integrated with the WAVE API accessibility testing platform. The accessibility testing provides a website evaluation and identifies the vulnerable accessibility issues according to accessibility guidelines WCAG 2.1. The Bot is designed to activate the WAVE testing tools, read, and analyze the accessibility parameters, and provides an overall accessibility evaluation. Bot results are stored in JSON format compatible with the Bot user experience framework database.

Table 3.2 shows the accessibility measures that the Bot program used in validating the website's accessibility issues:

Table 3.2: Accessibility testing measurement parameters

Number	Feature	Description
1.1	Missing alternative text for the linked image (Level A).	Any images that make up the entire link must have alternative descriptive text. A screen reader cannot inform the user of the purpose of the link if an image is contained within an association that has no text and that image does not offer an alternative text.
1.2	A form control lacks a corresponding label (Level A, Level AA).	A form control's function or purpose might not be clear to screen reader users if it lacks a properly matched text label. Additionally, form labels offer clear descriptions and larger clickable targets for form controls.
1.3	A heading has no content (Level A, Level AA).	Because keyboard and screen reader users frequently browse by heading components, addressing this type of error is crucial.
1.4	A button has no value text or is empty (Level A).	Users of screen readers must be given descriptive text explaining the purpose of each button when they go to them.
1.5	A link has no text (Level A).	The user will not be informed of its purpose or function if a link is text-free. For users of keyboards and screen readers, this may cause confusion.
2	Contrast Errors: Very low contrast	Text contrast must be sufficient for all users, but it is vital for those with low vision.

	between text and background colors (Level AA).	
3	Features	This attribute indicates the number of elements that, if implemented correctly, will enhance accessibility.
4	Structural Elements	It includes the number of elements that have been provided. For example, it might make it easier for screen readers or other assistive technology users to navigate your page, such as <nav> element, heading level 2, heading level 3, page header, page footer, etc.
5	Accessible Rich Internet Applications (ARIA),	Which enhances the semantics and accessibility of web content in ways that plain HTML cannot. Increased ARIA usage on pages correlated to higher detected errors. The more ARIA attributes present, the more accessibility errors are expected [48].

3.5 User Experience Framework Development

The Robotic Process Automation (Bots) user experience framework was developed using the ASP.NET software development environment and the Automation Anywhere Robotic Process Automation cloud platform. The Framework used the JMeter, ImmuniWeb, and WAVE open-source tools to evaluate the target website in terms of Security, Performance, and Accessibility. The framework includes a script API for integrating the open-source user experience testing tools, the script was designed as a scalable and customizable tool so that we can use it for all websites and call the hand using Command Line Interface (CLI), considering the diversity of technology used in various websites. The method of extracting reports and

displaying the results has been modified in the JMeter and ImmuniWeb Tools source code to be compatible with RPA Bot models. Figure 3.2 shows the architecture of the proposed framework.

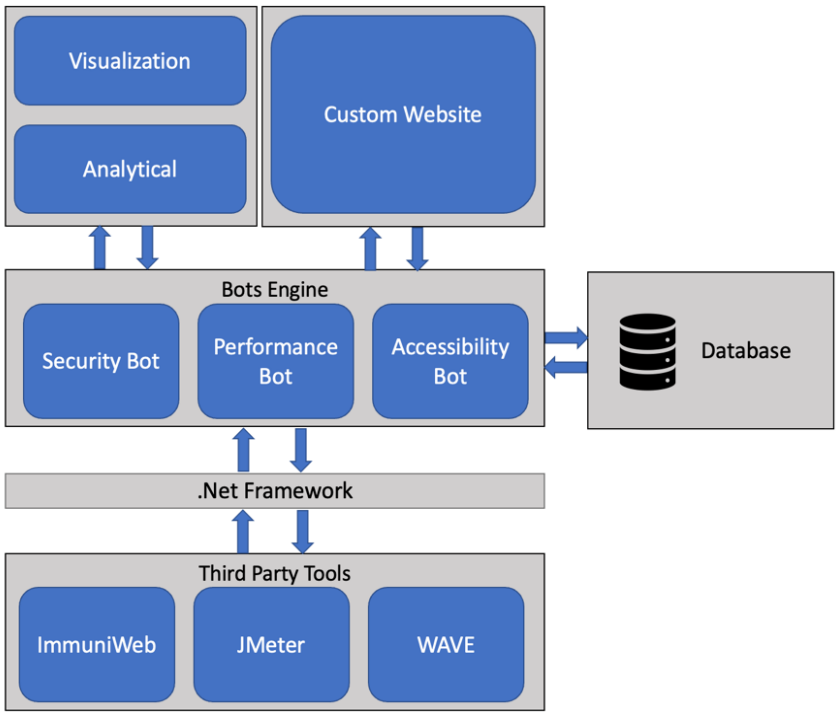


Figure 3.2: Proposed framework's architecture

RPA Bots can connect with a database, store, and read data from a database and evaluate the test results based on HCI guidelines and standards which will help businesses in evaluating their products more effectively, save time, money, and effort.

To get the website details by RPA bots, analyzing the results and saving the data for future assessments, we built a database that can be used by the RPA bots. The Website Security and SSL Testing bots save security scanning results into the SecurityResults table, the Performance Testing bot saves performance results into the PerformanceResults table, and the Accessibility Testing bot saves accessibility results into the AccessibilityResults table. Figure 3.3 shows the proposed database diagram.

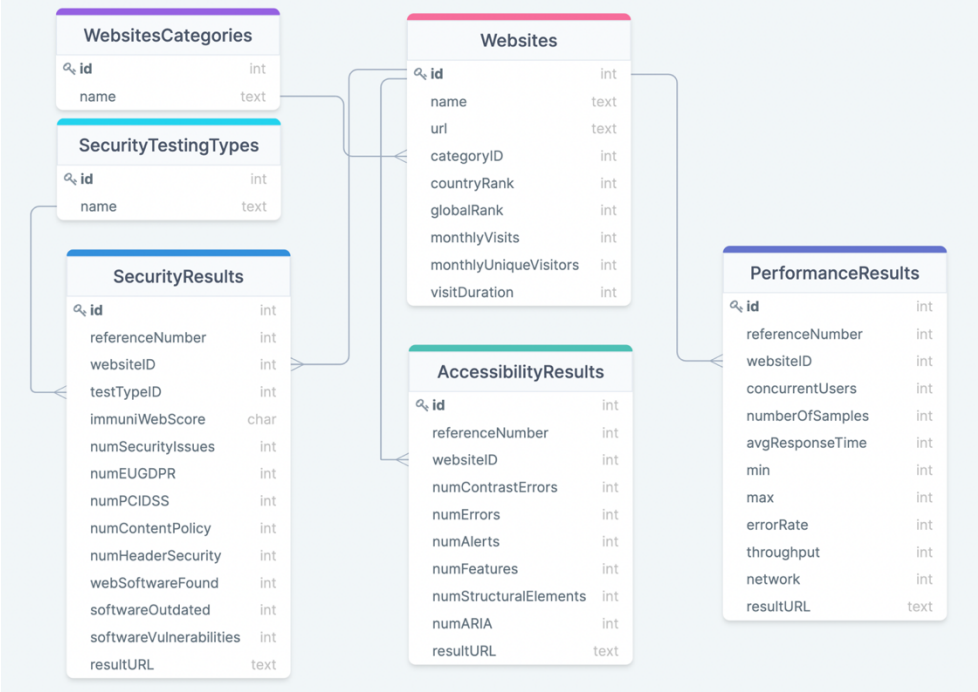


Figure 3.3: Developed database's diagram

3.6 Bots Development

In our proposed methodology, we aim to run the RPA bots and find the Performance, Security, and Accessibility issues for several local Palestinian websites to measure the accessibility, efficiency, effectiveness, Identity, security, and privacy of these websites. Therefore, we built three RPA bots as the following:

1. **Security Testing Bot**: Security Testing Bot is the key bot for security scanning tests to find and measure potential vulnerabilities in a website. This bot, which simulates end-user workflow on any website, begins by connecting to the database, retrieving the website's information, and then entering the website's URL, which is required for security screening. The next step is choosing a security scanning type: SSL or Web Scanning. Then, the bot runs the ImmuniWeb security testing script that by the developed framework and starting the security scanning of the target website. When the test is finished, the Bot uses the previously established database connection to insert the results into the SecurityResults table. After that, The Bot

disconnects and returns the scanning and evaluation results to the user. Figure 3.4 shows the lifecycle of the Security Scanning Bot

Figure 3.4: Lifecycle of the Security Testing Bot

2. **Performance Testing Bot**: The Performance Testing Bot is the key bot for Performance tests. The Bot launches and connects to the proposed database to retrieve information about the target website. The bot then enters the website's URL, Duration and Number of Concurrent Users. We configured the test duration in the script to be 20 minutes for all websites. The performance testing bot estimated and configured the number of concurrent users for each website based on the formula provided in [49] to determine the number of concurrent users where the Number of Concurrent Users is equal to Peak Hourly Visits X Average Visit Duration (in minutes)/60.

The third step is executing the JMeter Performance Testing script and begins testing the website. When the scan is completed, the bot inserts the results into the PerformanceResults table. Then, the bot closes the connection and displays the scanning results to the user. Figure 3.5 shows a lifecycle of the Performance Testing Bot.

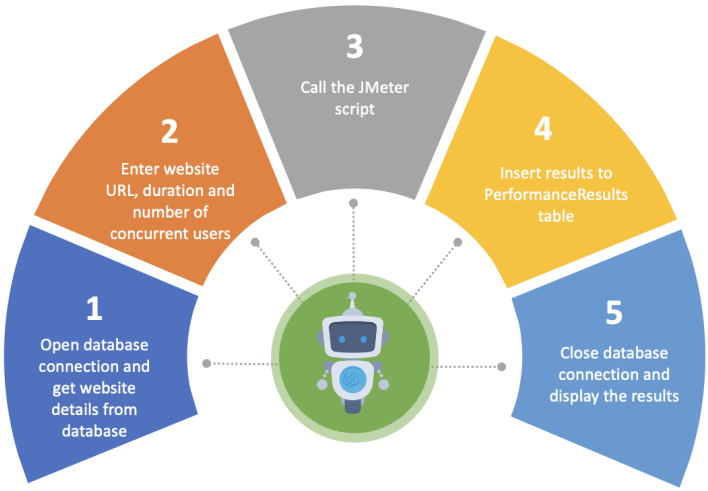


Figure 3.5: Lifecycle of the Performance Testing Bot

3. **Accessibility Testing Bot:** Accessibility Testing Bot is the key bot for Accessibility tests. The bot starts by connecting to our database to retrieve information about the target website. The bot then enters the website's URL with no extra details. The third step is calling the Wave tool API and sending the required parameters. When the scan is completed, the Bot inserts the results into the AccessibilityResults table. Then, the Bot closes the connection and displays the scanning results to the user. Figure 3.6 shows a lifecycle of the Accessibility Testing Bot.

Figure 3.6: Lifecycle of the Accessibility Testing Bot

3.7 Selected Websites for Usability Testing

To verify our framework using bots, we selected 16 Palestinian local websites from four categories: Higher Educational Institutions websites (HEI), Governmental websites (GW), News websites (NW), and eCommerce websites (CW). Table 3.3 shows the Country Rank, Global Rank, Monthly Visit, Monthly Unique Visitors and Visit Duration of each website from for the year 2021 according to SimilarWeb statistics.

Table 3.3: Traffic Details of the Selected Websites

Website's Symbol Allocated	Category	Country Rank	Global Rank	Monthly Visits	Monthly Unique Visitors	Visit Duration
HEI1	HEI	50	84,706	274,588	40,168	00:09:29
HEI2	HEI	26	42,061	1.389M	284,765	00:06:52
HEI3	HEI	41	66,253	872,793	101,348	00:04:57
HEI4	HEI	67	27,645	433,959	50,804	00:08:28
GW1	GW	237	271,489	191,401	78,923	00:04:46
GW2	GW	325	237,687	285,278	121,878	00:02:16
GW3	GW	3,205	808,812	29,738	17,574	00:02:24
GW4	GW	1,218	512,945	46,391	26,421	00:03:43
NW1	NW	12	20,100	2.511M	409,300	00:08:02
NW2	NW	10	9,245	7.551M	2.354M	00:06:20
NW3	NW	333	96,938	400,905	202,241	00:05:04
NW4	NW	1,204	358,897	195,334	60,081	00:01:59
CW1	CW	1,713	1,127,961	24,338	14,252	00:02:31
CW2	CW	3,505	369,517	31,464	17,333	00:07:09
CW3	CW	1,765	717,231	47,881	23,560	00:02:17
CW4	CW	1,463	1,116,374	7,317	<5,000	00:14:58

All experiments were run on a Windows Server with the specifications of 12 GB RAM, 4 CPU Cores, 9.6 GHz Total CPU Power, 1,000 mbps Network Port and 200 GB SSD Disk Space.

Chapter 4: Experimental Results and Analysis

4.1 Security Testing Results

The following sections show the results of running the RPA bots of Web Security and CMS Testing and SSL Testing.

4.1.1 Web Security and CMS Testing Results

As shown in Table 4.1, about 68% (11/16) of targeted websites have known web software or CMS components such as WordPress, Drubal, jQuery, etc., and 43% (7/16) of targeted websites have outdated software. In comparison, 25% (4/16) reported web software vulnerabilities, and the vulnerable websites are HEI3, NW1, GW3, and CW2.

Table 4.1: Web software security scanning results

Test Type	Test Measure	Number/Total
Web Software Security and CMS Test	**No. of Third-Party Software Used**	11/16
	Third-Party Software Outdated	7/16
	Third-Party Software Vulnerabilities	4/16

Table 4.2 shows the list of all websites, the number of founded web software and CMS components, and the number of outdated software and vulnerabilities in each website. The results show the third-party software or CMS component, obsolete software, and the website's vulnerabilities. In addition to the security grade according to the PCI DSS and EU GDPR guidelines.

Table 4.2: Web software security testing results by website

Web Software Security Test				
Website	**No. of Third-Party Software Used**	**Third-Party Software Outdated**	**Third-Party Software Vulnerabilities**	**Grade**
HEI1	1	1	0	A-

HEI2	0	0	0	B-
HEI3	8	5	13	C
HEI4	15	0	0	A
GW1	1	0	0	C
GW2	0	0	0	C
GW3	3	2	10	F
GW4	2	0	0	B
NW1	5	5	10	F
NW2	0	0	0	C
NW3	0	0	0	B-
NW4	2	1	0	B
CW1	1	1	0	C
CW2	3	3	11	F
CW3	1	0	0	B-
CW4	0	0	0	C

The results of the GDPR testing for each website are listed in Table 4.3. The results showed that more than half of the websites failed the privacy policy test, where there are nine websites with no privacy policy page or it's difficult to access that page on these websites, 43% (7/16) of the websites failed the cookie protection test, 25% (4/16) failed in the cookie disclaimer tests, 25% (4/16) failed in the cookie website security tests, and 12% (2/16) was the failure percentage in the TLS encryption test.

Table 4.3: GDPR testing results

Website	Privacy Policy	Website Security	TLS Encryption	Cookie Protection	Cookie Disclaimer
HEI1	Passed	Passed	Passed	Failed	Passed
HEI2	Passed	Passed	Passed	Passed	Passed
HEI3	Failed	Failed	Failed	Passed	Passed
HEI4	Passed	Passed	Passed	Passed	Passed
GW1	Failed	Passed	Passed	Passed	Passed
GW2	Failed	Passed	Passed	Passed	Passed
GW3	Failed	Failed	Failed	Failed	Failed
GW4	Failed	Passed	Passed	Passed	Passed

NW1	Passed	Failed	Passed	Failed	Passed
NW2	Failed	Passed	Passed	Passed	Passed
NW3	Passed	Passed	Passed	Passed	Passed
NW4	Failed	Passed	Passed	Failed	Failed
CW1	Failed	Passed	Passed	Failed	Passed
CW2	Passed	Failed	Passed	Failed	Failed
CW3	Failed	Passed	Passed	Passed	Passed
CW4	Passed	Passed	Passed	Failed	Failed
Total Failed	**9**	**4**	**2**	**7**	**4**

The results of the PCI DSS tests for each website are listed in Table 4.4. The results showed that 43% (7/16) of the websites failed in Requirement 6.2 verification, and 25% (4/16) failed in Requirement 6.5 verification. In comparison, 68% (11/16) of the websites failed in Requirement 6.5 verification.

Table 4.4: PCI DSS testing results

Website	Requirement 6.2	Requirement 6.5	Requirement 6.6
HEI1	Failed	Passed	Passed
HEI2	Passed	Passed	Passed
HEI3	Failed	Failed	Passed
HEI4	Passed	Passed	Passed
GW1	Passed	Passed	Failed
GW2	Passed	Passed	Failed
GW3	Failed	Failed	Passed
GW4	Passed	Passed	Failed
NW1	Failed	Failed	Failed
NW2	Passed	Passed	Failed
NW3	Passed	Passed	Failed
NW4	Failed	Passed	Failed
CW1	Failed	Passed	Failed
CW2	Failed	Failed	Failed
CW3	Passed	Passed	Failed
CW4	Passed	Passed	Failed
Total Failed	**7**	**4**	**11**

As shown in Table 4.5, HEI website's results showed that two websites are not compliant with EU GDPR and PCI DSS, in addition to other issues in software and headers security tests.

Table 4.5: Security testing results of HEI websites

Website	Software Security Test	EU GDPR Issues	PCI DSS Issues	Content Security Policy Test	Header Security Tests Issues
HEI1	1	1	1	Missing	3
HEI2	0	0	0	Missing	2
HEI3	5	3	2	Missing	2
HEI4	0	0	0	Missing	1

HEI1 is not compliant with EU GDPR and PCI DSS, and it has a total of six security issues, but none of them are significant. We found one outdated component, which is the Drubal CMS. It should be updated to the most recent version in version 9.3.13. However, there are no known security vulnerabilities in the current version. EU GDPR Compliance tests showed one issue in Cookie Protection, where Cookies with personal or tracking information are sent without the Secure flag.

There are other issues in Header Security tests, including missing of Strict-Transport-Security required header, which is important to force browsers to access the website via HTTPS. In addition, the server header and X-Powered-By HTTP headers are missing, and other optional headers like Access-Control-Allow-Origin, Public-Key-Pins, Public-Key-Pins-Report-Only, and Expect-CT.

We found Some HTTP headers related to security and privacy are missing or misconfigured in HEI2 test results like Strict-Transport-Security and X-Content-Type-Options, in addition to some missing optional HTTP security headers including Access-Control-Allow-Origin, Public-Key-Pins, Public-Key, Pins-Report-Only, and Permissions-Policy.

HEI3 has major security issues in the fingerprinted CMS components, including jQuery 3.31, jQuery UI 1.12.1, and Bootstrap 3.3.5, which are outdated and vulnerable to publicly known vulnerabilities. Those components should be updated to the most recent versions. The HEI3 website fails in GDPR compliance tests where the Privacy Policy was not found on the website, and HTTPS encryption is missing. Because of the CMS version issue, it doesn't meet Requirements 6.2 and 6.5 in the PCI DSS Compliance test. HEI3 has header security issues like HEI1 and HEI2, where required HTTP headers are missing (X-Frame-Options and X-Content-Type-Options). Some optional HTTP headers (Access-Control-Allow-Origin, Expect-CT, and Permissions-Policy) in addition to an issue in the X-Powered-By header as the web server discloses its version, potentially facilitating further attacks against it in addition to other security issues in cookies where some cookies have missing secure flags or attributes.

HEI4 passed the Web Software Security, EU GDPR, and PCI DSS security tests but with one issue in Header Security Tests as Strict-Transport-Security and X-Frame-Options are missing. Figure 4.1 shows the security testing results of HEI websites.

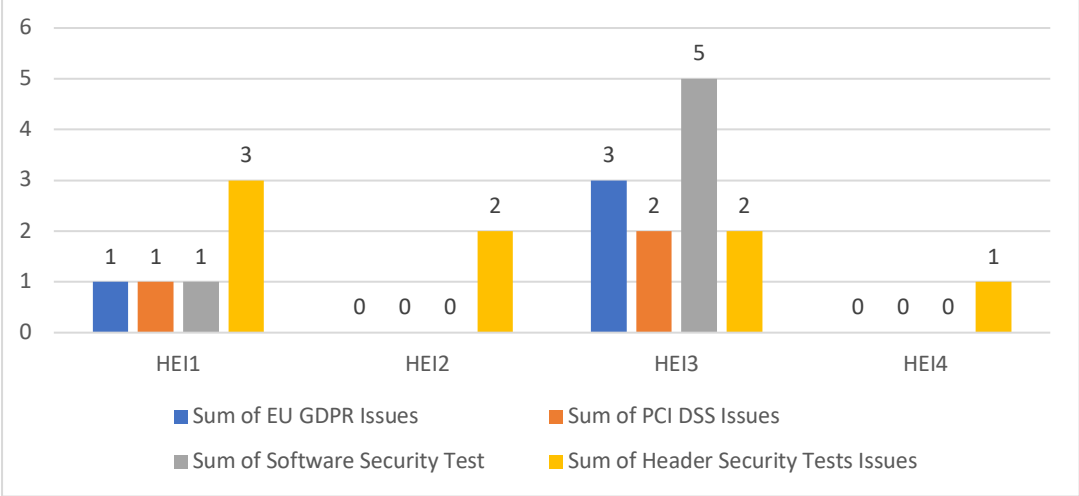


Figure 4.1: Security testing results of HEI websites

For the results of the governmental websites, Table 4.6 shows the Security Testing Results of Governmental Websites, where none of the websites passed the EU GDPR and PCI DSS tests.

Table 4.6: Security testing results of Governmental websites

Website	Software Security Test Issues	EU GDPR Issues	PCI DSS Issues	Content Security Policy Test	Header Security Tests Issues
GW1	0	1	1	Missing	2
GW2	0	1	1	Missing	1
GW3	2	5	2	Missing	1
GW4	0	1	1	Missing	1

We found that none of the websites have a privacy policy, which is required to pass the GDPR compliance test. TRACK and CUSTOM headers are enabled in GW1, GW2, and GW3, making the web server vulnerable to XST (Cross-Site-Tracing) attacks. Regarding the CUSTOM header, the server appears to allow any HTTP method, which could pose a security risk. No WAF was discovered by PCI DSS testing on GW1, GW2, and GW4, which violates Requirement 6.6.

Security header tests have discovered other security issues in the governmental websites, such as the missing of some required HTTP headers (e.g., Strict-Transport-Security, X-Frame-Options, and X-Content-Type-Options), as well as the CI SESSION cookie, which lacks the Secure, HttpOnly, and SameSite flags.

GW3 fails to meet GDPR or PCI DSS compliance criteria because of two outdated old, fingerprinted CMS components: jQuery 2.0.0 and Bootstrap 3.3.6. On the other hand, GW4 has a better security grade one issue due to a missing privacy policy, failing to meet requirement 6.6 since no WAF was discovered on the website. Figure 4.2 shows the web scanning and security testing results of Governmental websites.

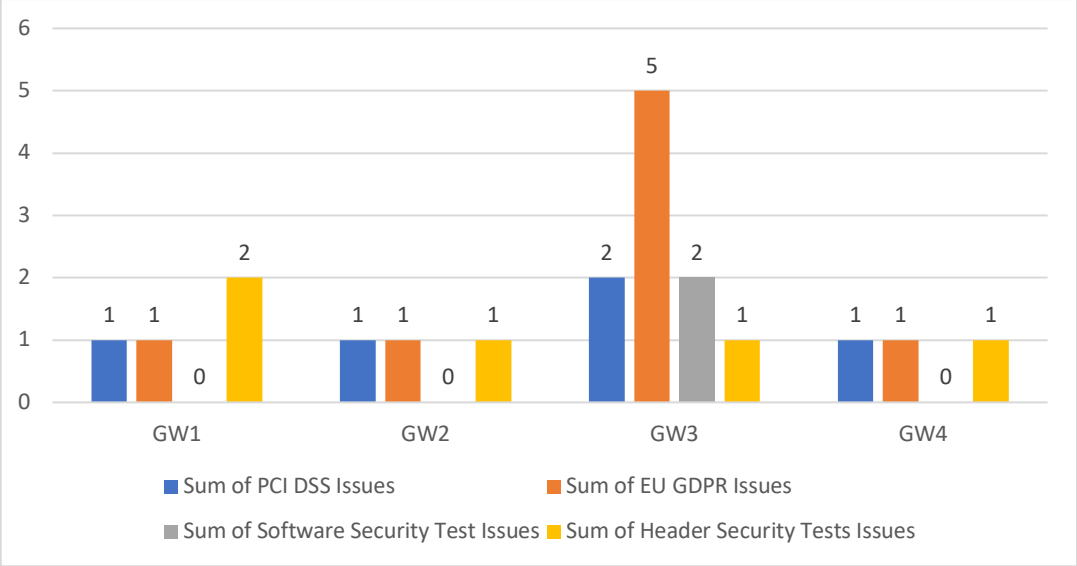


Figure 4.2: Web Scanning and Security Testing Results of Governmental Websites

Table 4.7 shows the Security Testing Results of News Websites, where none of the websites passed the PCI DSS test while only NW3 passed the EU GDPR test.

Table 4.7: Security testing results of news websites

Website	Software Security Test	EU GDPR Issues	PCI DSS Issues	Content Security Policy Test	Header Security Tests Issues
NW1	4	2	3	Missing	5
NW2	0	1	1	Missing	1
NW3	0	0	1	Missing	2
NW4	1	3	2	Missing	0

News websites tests' results showed that NW1 has a critical security issue: TRACK and CUSTOM headers are enabled, and the jQuery 3.3.1 and Bootstrap 3.3.7 CMS versions are old and vulnerable to publicly known vulnerabilities and should be changed to the most recent versions. Furthermore, cookies containing personal or tracking data are transferred without the Secure mark, which violates EU GDPR rules. In addition, the website fails all PCI DSS compliance tests because it does not meet the 6.2, 6.5, and 6.6 requirements. Furthermore, the website has various HTTP header misconfigurations and flaws, such as the configuration

of ACCESS-CONTROL-ALLOW-ORIGIN and several cookies with missing secure flags or attributes.

NW1, NW2, and NW4 have failed the EU GDPR compliance tests as none of these websites have a privacy policy on their websites, or it is not easily accessible. In addition, required HTTP headers are missing in NW1, NW2, and NW3 (e.g., Strict-Transport-Security, X-Frame-Options). Moreover, all news websites have failed the PCI DSS compliance tests, where none of them passed the requirement 6.6 compliance test. Furthermore, NW1 fails to meet any of the three conditions, whereas NW2 only meets requirement 6.5. Figure 4.3 shows the security testing results of News websites.

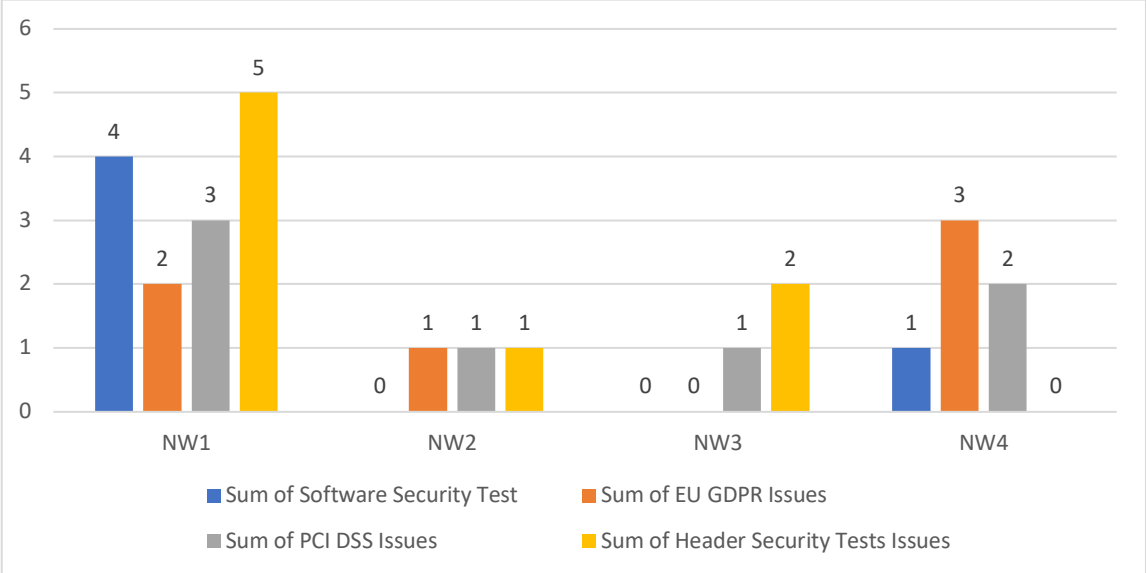


Figure 4.3: Security testing results of News website

Table 4.8 shows the Security Testing Results of eCommerce Websites, where none of the websites passed the PCI DSS and EU GDPR tests.

Table 4.8: Security testing results of eCommerce websites

Website	Software Security Test	EU GDPR	PCI DSS	Content Security Policy Test	Header Security Tests
CW1	1	1	2	Missing	1
CW2	3	3	3	Missing	2

CW3	0	1	1	Missing	1
CW4	0	2	1	Missing	1

eCommerce website results showed that two eCommerce websites had outdated CMS components; CW1 uses outdated PrestaShop 1.6.0.9 CMS, and no known security vulnerabilities are found in the current version. It should be upgraded to the most recent version, 1.7.8.6. CW2 has three outdated components which are vulnerable to publicly known vulnerabilities. It uses jQuery 2.1.4, Moment.js 2.17.1, and Bootstrap 3.3.7, and all of these components should be updated to the most recent versions.

The issues are different from one website to another, as in GPDR compliance tests, Privacy Policy was not found in CW1 and CW3. CW1, CW2, and CW4 Cookies protection tests failed because the cookies with personal or tracking information are sent without the Secure flag. The Cookie disclaimer test failed for CW2 and CW4 as Third-party cookies or cookies with tracking information were sent, but no cookie disclaimer was found on the website. Regarding PCI DSS compliance tests, the Requirement 6.6 test failed for all eCommerce websites. Requirement 6.2 test failed for CW1 and CW2, while Requirement 6.5 failed due to known security vulnerabilities in current CMS versions.

In Header Security tests, we found that all eCommerce websites' HTTP headers related to security and privacy are missing or misconfigured. Furthermore, X-Powered-By testing failed for CW2 and CW3 as the web server discloses its version, potentially facilitating further attacks against it. In addition to the SERVER header issue in CW4, which is usually sent by websites to advertise their version.

The findings showed that all the selected samples failed the Required HTTP Headers tests and that some HTTP headers relating to security and privacy are missing or incorrectly defined across all websites. Most websites, for example, did not set the Strict-Transport-Security, X-Frame-Options, or X-Content-Type-Options headers. Another crucial HTTP Headers test is X-Powered-By, which hackers may use to target a certain technology or version.

We found that the servers of six websites reveal their version, potentially allowing for more assaults against them.

Table 4.9 shows list of failed security tests in addition to the number of failed websites by category in GDPR compliance, PCI DSS compliance, HTTP Headers tests, in addition to extra tests in Content Security Policy and Cookies Privacy and Security Analysis tests that can be marked as Industry Best Practice tests.

Table 4.9: Failed Security Tests Statistics by Website's Category

Category	Test Type	Num. of Failed HEI Websites	Num. of Failed Governmental Websites	Num. of Failed News Websites	Num. of Failed eCommerce Websites	Total
GDPR	**Privacy Policy**	1	3	2	2	8
	Website Security	1	1	1	1	4
	TLS Encryption	1	1	0	0	2
	Cookie Protection	1	1	2	3	7
	Cookie Disclaimer	0	1	1	2	4
PCI DSS	**Requirement 6.2**	2	1	2	2	7
	Requirement 6.5	1	1	1	1	4
	Requirement 6.6	0	3	4	4	11
HTTP Headers	**Required HTTP Headers**	4	4	4	4	16
	Server Header	1	0	1	1	3
	X-Powered-By	2	1	1	2	6
	Access-Control-Allow-Origin	0	0	1	0	1
	Strict-Transport-Security	0	0	1	1	2
Others	**Content Security Policy**	4	4	4	4	16
	Cookies Privacy and	3	2	2	3	10

	Security Analysis					

Figure 4.4 shows the most founded security issues in all websites as well as the percentage of infected websites with these issues.

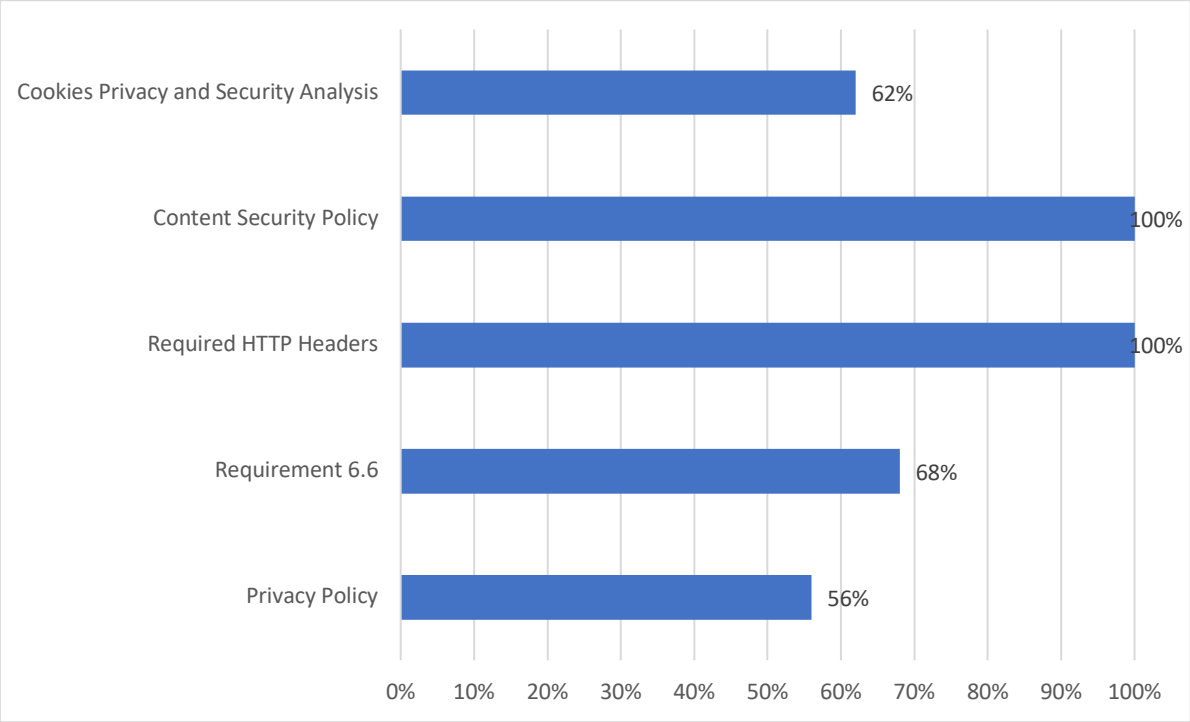


Figure 4.4: Most founded security issues detected by Bot security model

4.1.2 SSL Testing Results

As shown in Table 4.10, SSL testing results showed that 56% (9/16) of the websites failed in the PCI DSS tests, 37% (6/16) failed in both HIPAA and NIST tests

Table 4.10: Number of failed websites in SSL testing by website's category

Category	PCI DSS Failed Websites	HIPAA Failed Websites	NIST Failed Websites
HEI Websites	3	3	3
Governmental Websites	0	0	0
News websites	3	2	2
eCommerce Websites	3	1	1

Total	9	6	6

Table 4.11 shows the list of all websites and the number of founded issues in PCI DSS, HIPAA, and NIST tests.

Table 4.11: SSL testing results by website

Website	Grade	PCI DSS	HIPAA	NIST
HEI1	B-	2	9	9
HEI2	B+	2	2	3
HEI3	A-	2	2	2
HEI4	A	0	0	0
GW1	A+	0	0	0
GW2	A+	0	0	0
GW3	A+	0	0	0
GW4	A	0	0	0
NW1	A-	2	2	2
NW2	A	1	0	0
NW3	B	2	2	3
NW4	A+	0	0	0
CW1	A-	1	0	0
CW2	A-	1	0	0
CW3	A	1	0	0
CW4	A	0	2	2

The results showed that 25% (4/16) of the websites have a grade of A+, 31% (5/16) of the websites have a grade of A, and 25% (4/16) of the websites have a grade of A-, while 19% (3/16) have a grade lower than A-.

HEI1, HEI2, and HEI3 all have TLS 1.0 enabled, vulnerable to man-in-the-middle attacks, risking the integrity and authentication of data sent between a website and a browser [50]. HTTP Strict Transport Security is not enforced on HEI1, HEI3, or HEI4 servers, and this should be enabled to force the users to access the website over HTTPS. The server in HEI2 enables a client-initiated secure renegotiation, which could be risky and open the door to Denial-of-Service attacks. We found that HEI4's server is not set to handle OCSP stapling for

its RSA certificate, which provides for better verification of the certificate validation status during OCSP stapling verification. Figure 4.5 shows the SSL Testing Results of HEI Websites.

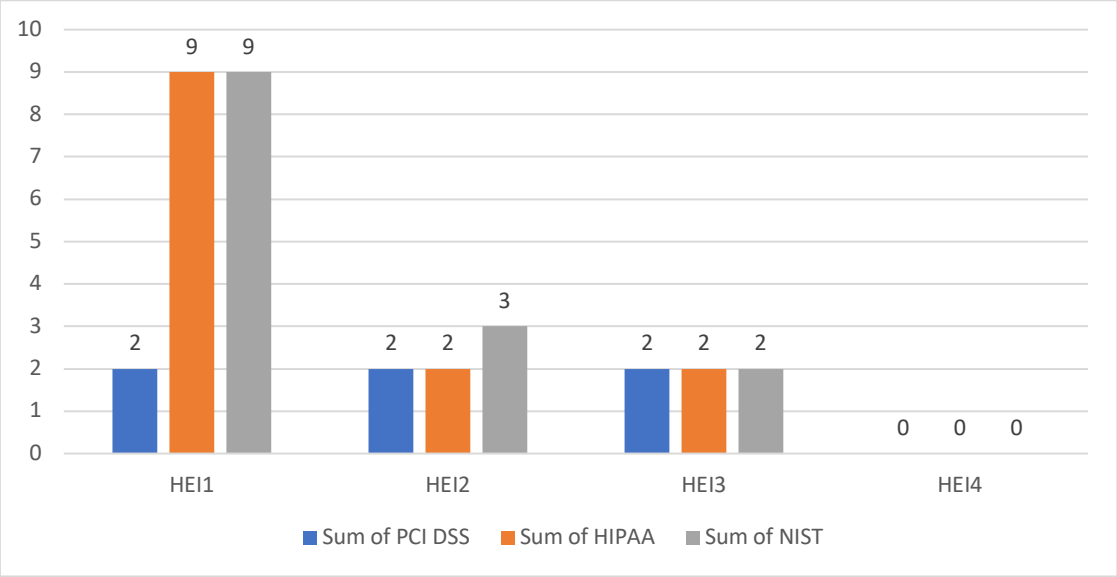


Figure 4.5: SSL testing results of HEI websites

For the governmental websites, servers support the most recent and secure TLS protocol version of TLS 1.3, and they all complain to PCI DSS, HIPAA, and NIST.

Only NW4 website from the news websites category is PCI DSS, HIPAA, and NIST compliant, whereas all other news websites failed the complaint tests due to the use of TLS 1.0, which is not PCI DSS, HIPAA, or NIST compliant. Furthermore, HTTP Strict Transport Security is not enforced by GW1, GW2, or GW3. Another issue with GW3 is that the server allows for client-initiated secure renegotiation, which could be dangerous and enable Denial-of-Service attacks. Figure 4.6 shows the SSL Testing Results of News Websites.

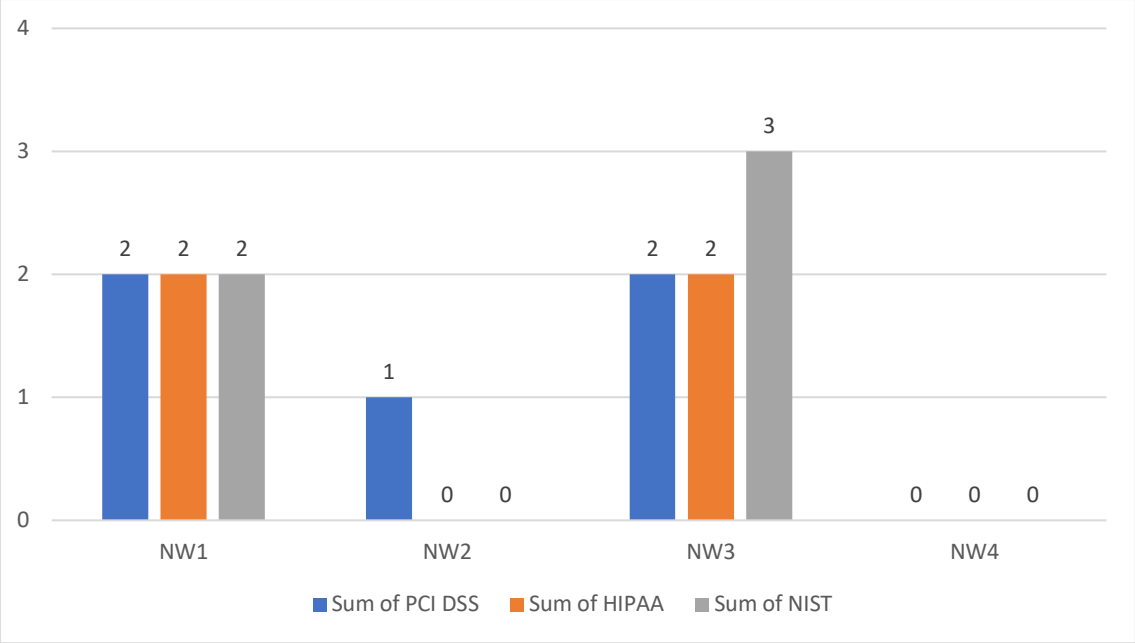


Figure 4.6: SSL testing results of News websites

TLS 1.0 is enabled on CW1, CW2, and CW3 servers in the eCommerce websites category, which is non-compliant with PCI DSS and NIST. CW1's server does not support OCSP stapling for its RSA certificate, and both CW1 and CW4 violate NIST rules because neither server supports the Extended Master Secret (EMS) extension for TLS version 1.2. EMS adds further security to SSL sessions and protects against some MitM attacks. In addition, the CW2 and CW3 servers do not enforce HTTP Strict Transport Security, which is beneficial because it forces users to use HTTPS to access the website. Figure 4.7 shows the SSL Testing Results of eCommerce Websites.

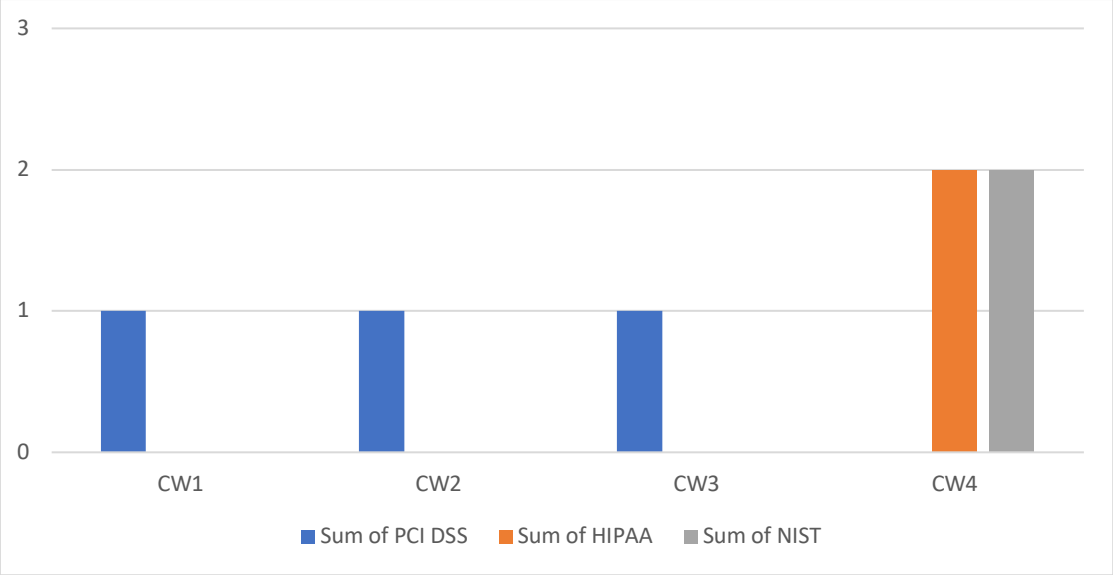


Figure 4.7: SSL testing results of eCommerce websites

Table 4.12 shows the PCI DSS, HIPAA and NIST, and Industry Best Practice failed items as well as number of failed websites by category.

Table 4.12: Failed SSL testing items by website's category

Category	Test Type	Num. of Failed Educational Websites	Num. of Failed Governmental Websites	Num. of Failed News Websites	Num. of Failed Commercial Websites	Total (Percentage)
PCI DSS	Supported Ciphers	3	0	2	0	**5 (31%)**
	Supported Protocols	3	0	3	3	**9 (56%)**
HIPAA and NIST	Ocsp Stapling	1	0	1	2	**4 (25%)**
	Supported Ciphers	3	1	2	0	**6 (37%)**
	Supported Protocols	3	0	3	3	**9 (56%)**
	Supporting Extended Master Secret	0	0	0	2	**2 (12%)**

Industry Best Practices	**Server Provides HSTS**	3	3	3	3	**12 (75%)**
	Not supporting of client-initiated secure renegotiation	0	0	1	1	**2 (12%)**
	HTTP to HTTPS redirection	1	0	2	1	**4 (25%)**
	Supporting TLSv1.3	1	0	0	2	**3 (19%)**

4.2 Performance Testing Results

The automated user experience and performance testing were conducted using the Performance Testing bot integrated with JMeter through the .NET framework. The performance experiments were executed using the two most common networks in Palestine: the 3G network and the LAN network with a speed of 8 Mbps. The first experiment's result of running the script using a 3G network was as shown in Table 4.13

Table 4.13: Performance testing results using 3G network

Website	Number of Completed Samples	Avg Response Time (msec)	Min (msec)	Max (msec)	Error Rate (%)	Throughput	Network Received (KB/sec)
HEI1	9123	10387	5605	23488	0.0	8.1	1253.7
HEI2	40566	6105	2259	12427	0.0	35.9	2383.5
HEI3	21845	5670	2016	15143	0.0	19.3	1556.1
HEI4	6005	13106	6277	18580	2.4	5.3	1944.6
GW1	3841	12585	4184	31773	0.0	3.4	2856.0
GW2	11284	10476	3199	21394	0.0	9.9	3265.5
GW3	4081	14489	5408	14136	0.0	3.6	396.9
GW4	19441	11078	1979	9809	2.2	17.1	1978.2
NW1	10088	6384	4373	23004	0.0	8.8	8637.3
NW2	7441	7532	2902	18869	0.0	6.5	4143.3
NW3	9843	8584	4332	17882	1.4	8.6	3250.8

NW4	6487	10979	4652	19813	0.0	5.7	1946.7
CW1	32402	4406	1704	7458	0.0	28.4	636.3
CW2	7684	12473	3143	22097	1.6	6.7	3672.9
CW3	3844	19769	4157	36113	1.1	3.4	4002.6
CW4	5047	16261	5183	23793	0.0	4.4	1707.3

The results shows that the average response time for all websites was 10604 msec using the 3G network, the CW1 website has the lowest average response time with 4406 msec, and the average response time of HEI websites was 8817 msec, while it was 12157 msec for governmental websites, 8370 msec for news websites, and 13227 msec for eCommerce websites. Figure 4.8 shows the average response time for all websites.

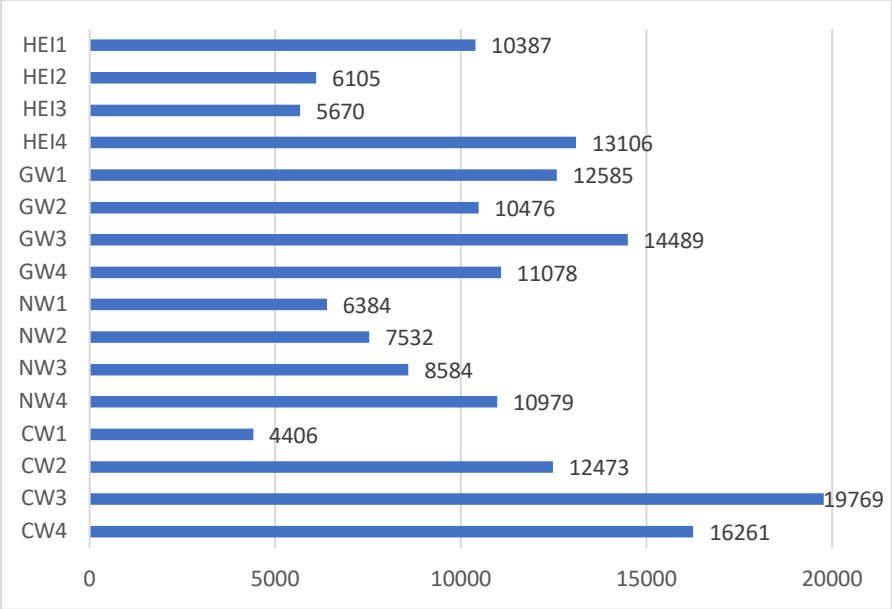


Figure 4.8: Average response time in msec by website using 3G network

As show in Figure 4.9, HEI2 has the highest throughput at 35.9, while GW2 and CW3 have the lowest throughput at 3.4. On the other hand, the average throughput of HEI websites was 17.2, 8.5 for governmental websites, 7.4 for news websites, and 10.7 for eCommerce websites.

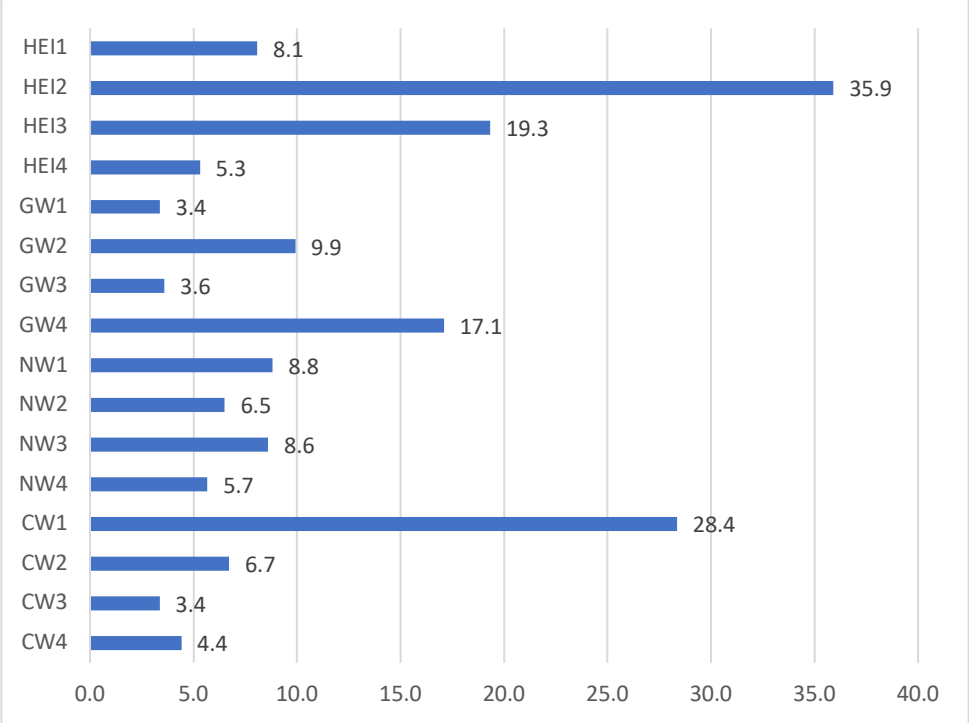


Figure 4.9: Throughput by website using 3G network

The results show that 31% (5/16) of the websites have returned errors to some users due to the high traffic in which these users were unable to access the website because they received a 503 Service Unavailable server error response code, which indicates that the server is not ready to handle the requests. Regarding the total number of completed samples, HEI2 has the highest number of completed samples, 40566 samples, while GW1 has the lowest number of completed samples with 3841 samples. Figure 4.10 shows a total number of completed samples for all websites.

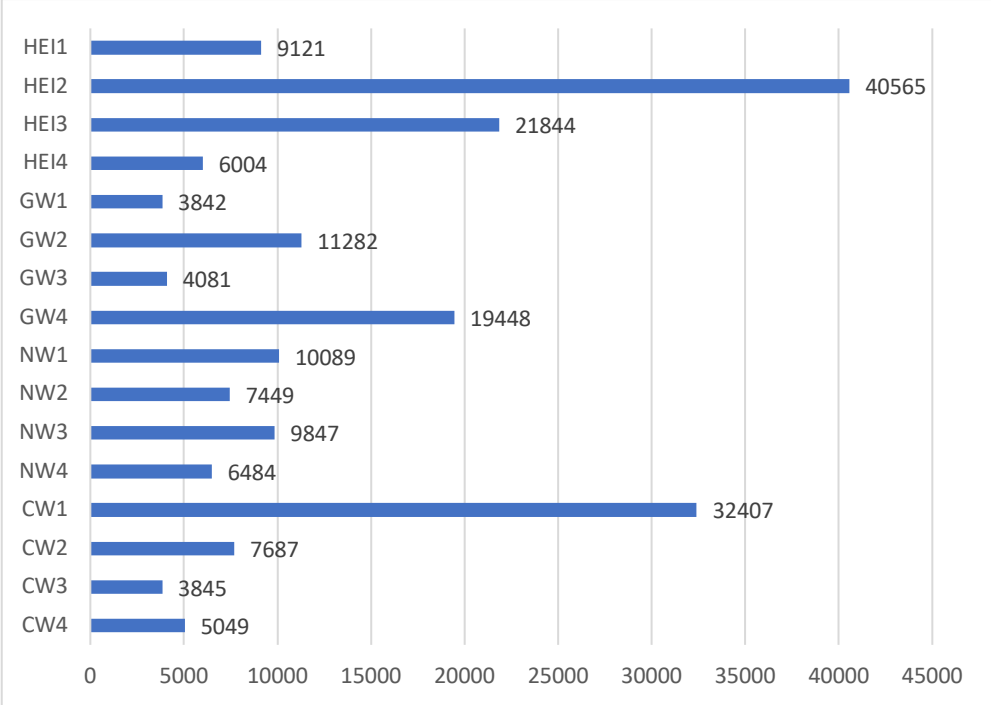


Figure 4.10: Number of completed samples by website using 3G network

We executed the same script with the same configurations on all websites using an 8 Mbps LAN network. Table 4.14 shows the experiment's results for all websites.

Table 4.14: Performance testing results using 8 Mbps LAN network

Website	Total Completed Samples	Avg Response Time (msec)	Min (msec)	Max (msec)	Error Rate (%)	Throughput	Network Received (KB/sec)
HEI1	11402	8814	3811	19740	0.0	9.2	1253.7
HEI2	54685	2566	1069	7888	0.0	45.8	2383.5
HEI3	28609	3768	2353	7774	0.0	24.4	1556.1
HEI4	10209	9516	6297	15716	3.2	8.2	1944.6
GW1	9449	8189	3483	41250	0.6	6.3	2720
GW2	17043	4189	1478	25561	0.0	13.7	3110
GW3	5647	8917	5905	12396	0.0	4.4	396.9
GW4	21487	3381	1243	6605	6.8	17.2	1978.2
NW1	17363	5125	3528	67360	0.6	13.2	8637.3
NW2	15325	5856	2168	103820	0.0	9.2	4143.3
NW3	13844	6282	2874	40194	1.9	11.6	3250.8
NW4	12569	7368	4486	23862	0.0	9.9	1946.7
CW1	41408	2271	1543	6464	0.0	35.1	636.3
CW2	12204	7903	4443	15110	1.9	9.7	3672.9

CW3	8162	12652	3349	27512	2.5	6.1	4002.6
CW4	9766	6535	1972	13988	2.7	8.2	3208.8

The average response time for all websites was 6458 msec using the 8 Mbps LAN network. The CW1 has the lowest average response time with 2271 msec, the average response time of HEI websites was 6116 msec, while it was 7026 msec for governmental websites, 6734 msec for news websites, and 7346 msec for eCommerce websites. Figure 4.11 shows the average response time for all websites.

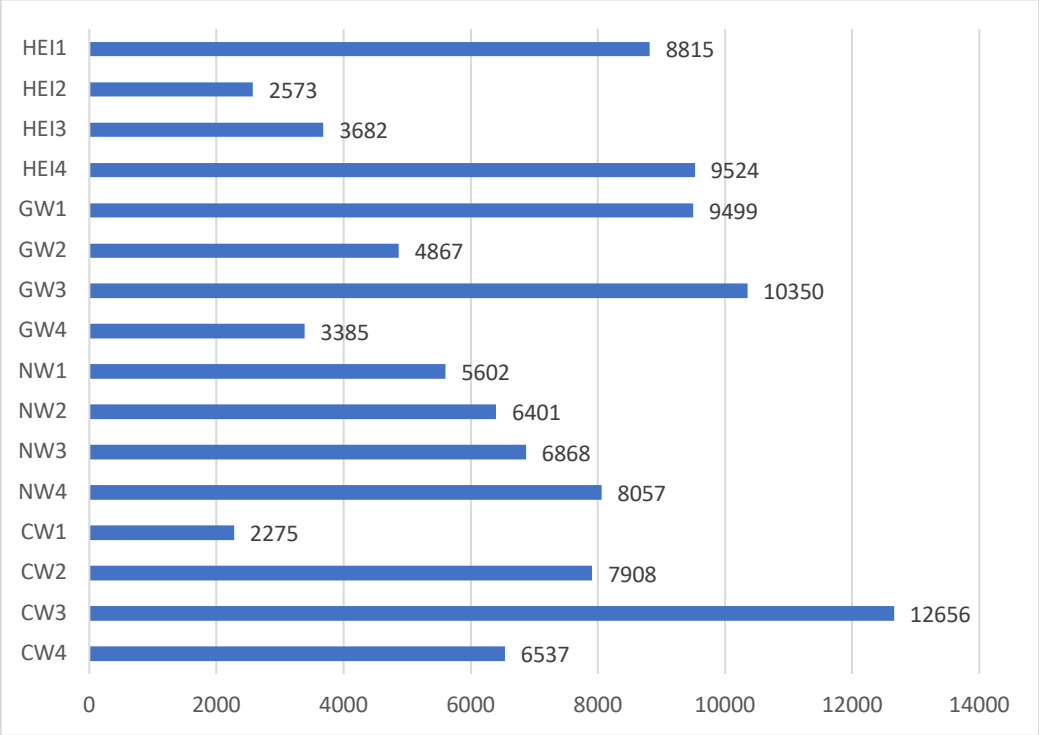


Figure 4.11: Average response time in msec by website using 8 Mbps LAN network

As shown in figure 4.12, HEI2 has the highest throughput as it was 45.8, while GW3 has the lowest throughput of 4.4. The average throughput of all websites was 14.5, it was 21.9 for HEI websites, while it was 10.4 for governmental websites, 11 for news websites, and 14.8 for eCommerce websites.

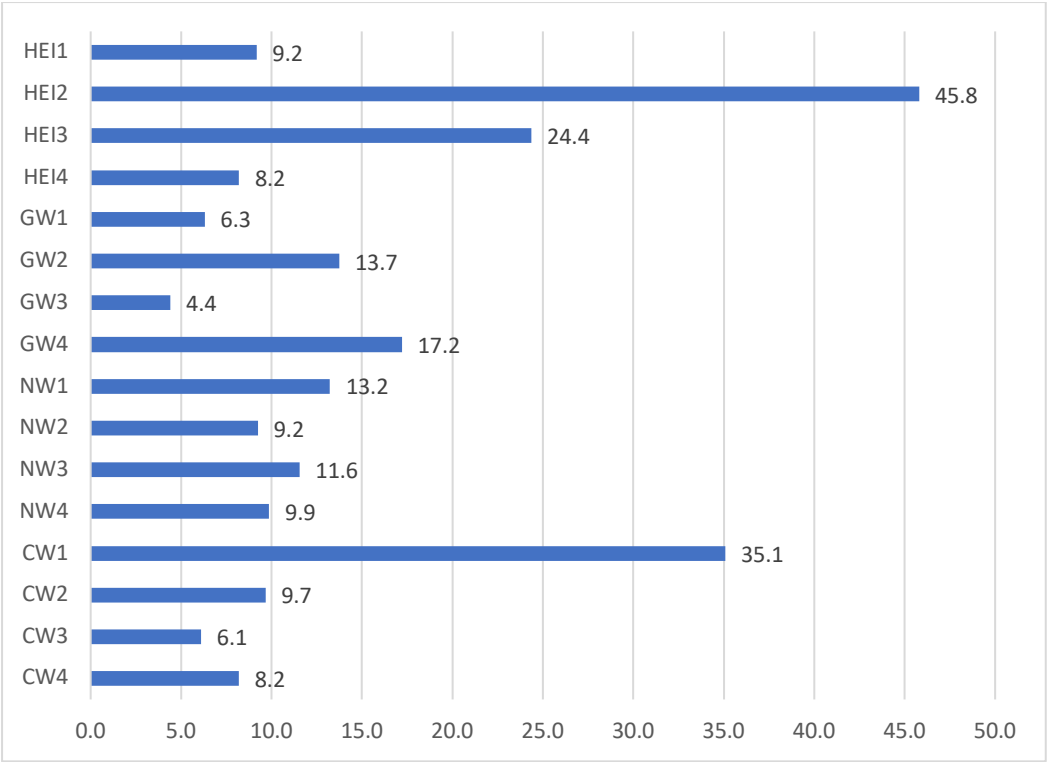


Figure 4.12: Throughput by website using 8 Mbps LAN network

HEI2 has the highest total completed samples with 54683 samples, while GW3 has the lowest number of completed samples with 5646 samples. Figure 4.13 shows the total number of completed samples for all websites.

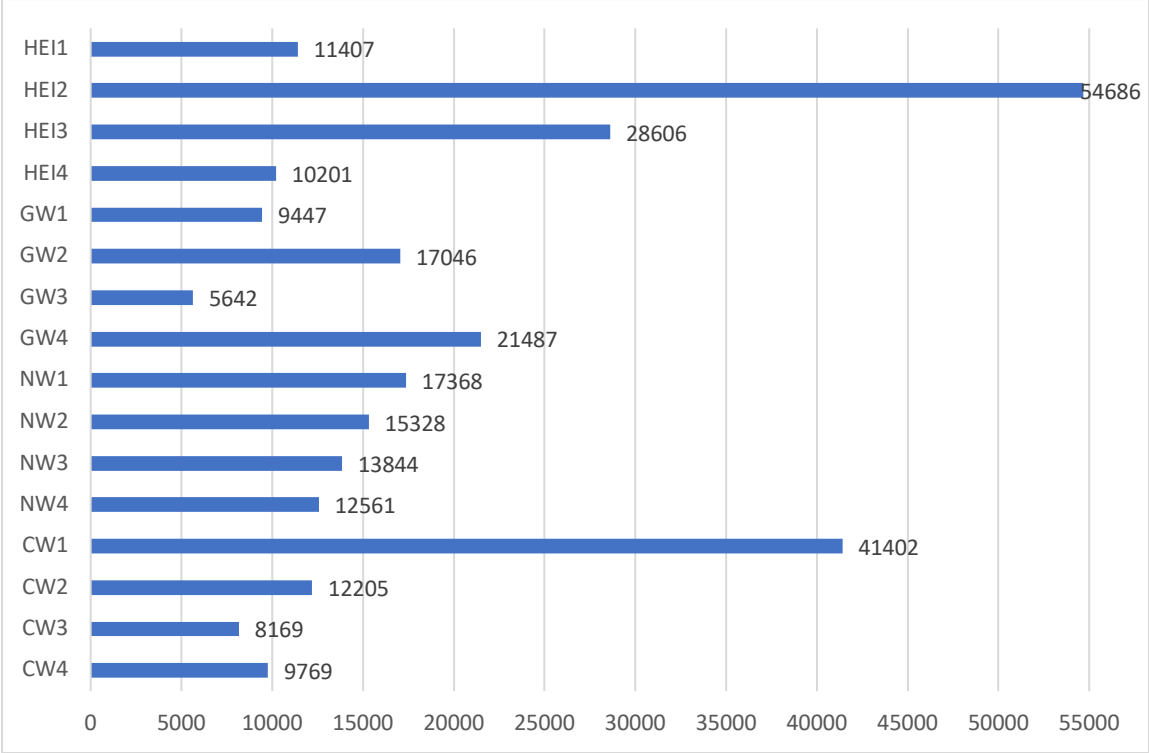


Figure 4.13: Total Number of completed samples by website using 8 Mbps LAN network

The results show that 50% (8/16) of the websites have returned errors to some users due to the high traffic; these users were unable to access the website because they received a 503 Service Unavailable server error response code, which indicates that the server is not ready to handle the requests.

4.3 Accessibility Testing Results

The results of the accessibility tests showed that there are many accessibility issues in all selected websites, where most of the websites have a high number of contrast errors, errors, and alerts. For example, NW1 has the highest number of errors with 280 errors, CW3 has another high number of errors with 200 errors, and CW4 has the lowest number of errors with 7. In addition, CW3 has the highest number of contrast errors with 162, while CW1 has the lowest number of contrast errors with four.

NW3 has the highest number of alerts with 572, NW1 has the highest number of features with 224 features, and it has the highest number of structural elements with 259. While NW4 has the highest number of ARIA with 313 items. Table 4.15 shows the accessibility results for all websites

Table 4.15: Accessibility testing results

Website	Errors	Contrast Errors	Alerts	Features	Structural Elements	ARIA
HEI1	50	42	81	47	68	201
HEI2	39	88	36	19	75	65
HEI3	63	32	39	25	76	99
HEI4	31	68	47	39	49	150
GW1	33	35	53	10	61	4
GW2	11	50	124	36	87	24
GW3	38	38	45	0	47	5
GW4	36	13	16	19	115	135
NW1	280	27	553	224	259	27
NW2	11	40	36	88	17	8
NW3	8	49	572	156	195	11
NW4	80	20	145	14	110	313
CW1	7	4	20	13	33	0
CW2	11	124	26	39	40	1
CW3	200	162	112	67	130	7
CW4	7	40	20	13	33	0

Figure 4.14 shows HEI websites' errors, contrast errors, and alerts. In this category, HEI3 has the highest number of errors with 63 errors, HEI2 has the highest number of contrast errors with 88 errors, and HEI has the highest number of alerts with 81 alerts.

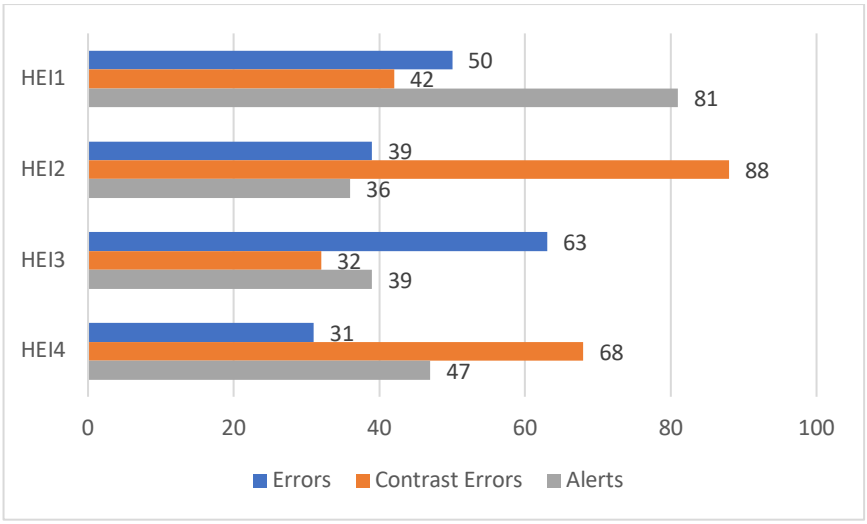


Figure 4.14: Total number of errors, contrast errors, and alerts in HEI websites

Figure 4.15 shows Governmental websites' errors, contrast errors, and alerts. In this category, GW3 has the highest number of errors with 38 errors, GW2 has the highest number of contrast errors with 50 errors, and it has the highest number of alerts as well with 124 alerts.

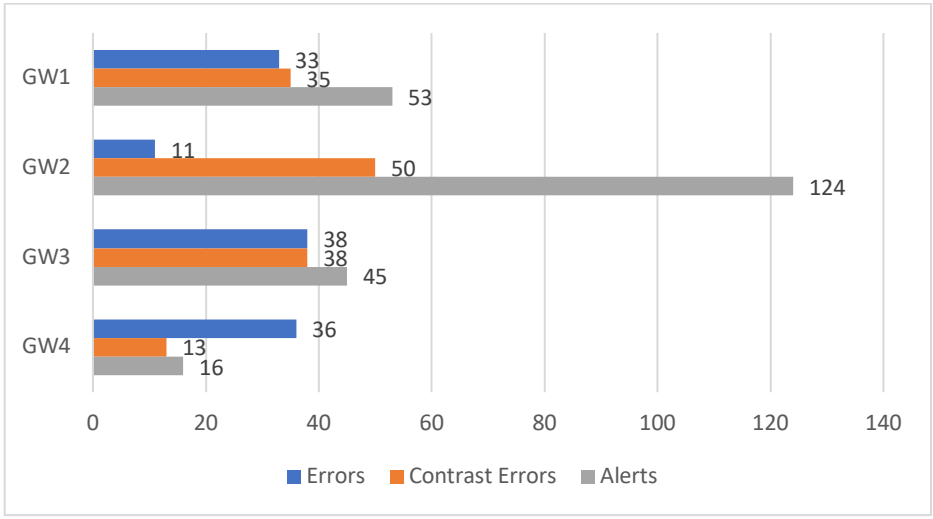


Figure 4.15: Total number of errors, contrast errors, and alerts in Governmental websites

Figure 4.16 shows News websites' errors, contrast errors, and alerts. In this category, NW1 has the highest number of errors with 280 errors, NW3 has the highest number of contrast errors with 49 errors, and it has the highest number of alerts with 572 alerts.

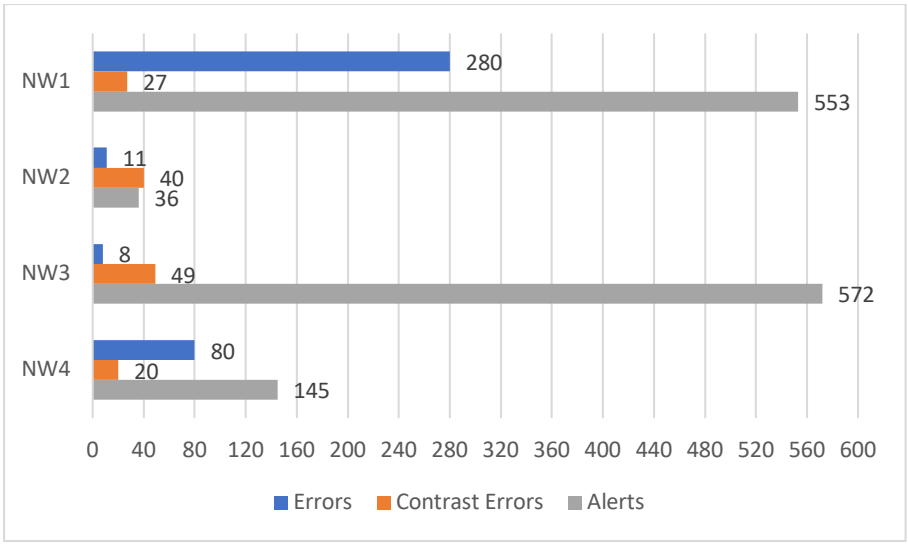


Figure 4.16: Total number of errors, contrast errors, and alerts in News websites

Figure 4.17 shows eCommerce websites' errors, contrast errors, and alerts. In this category, CW3 has the highest number of errors with 200 errors, the highest number of contrast errors and alerts, and 162 contrast errors and 112 alerts.

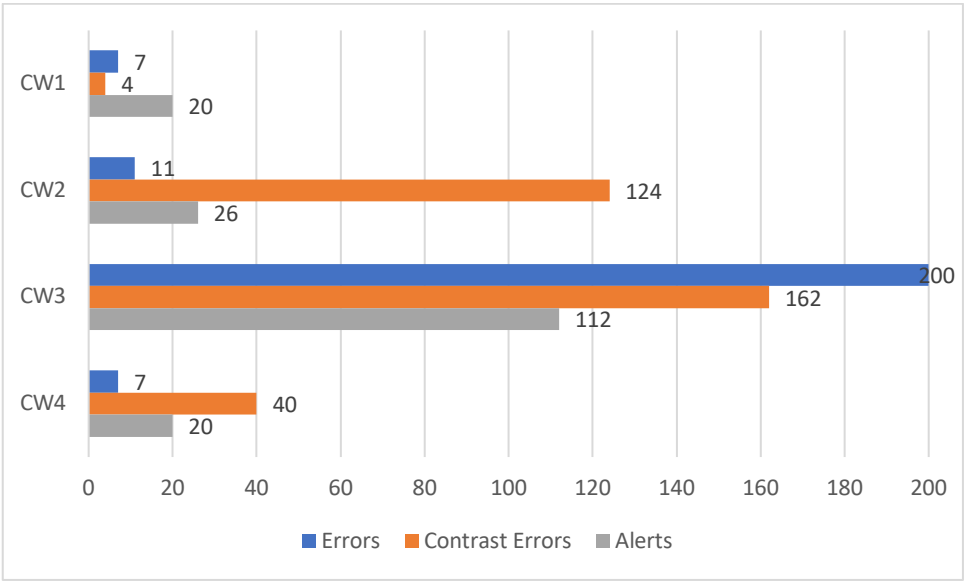


Figure 4.17: Total number of errors, contrast errors, and alerts in eCommerce websites

Chapter 5: Discussion and Conclusion

5.1 Research Limitations

1. The Framework used specific usability measure and doesn't include all HCI standards and guideline.
2. The Framework was limited to website applications, with limited access to internal application resource on the server side.
3. Limited testing resources for testing a high number of concurrent users.
4. The Bots are limited to Automation Anywhere cloud platform.
5. RPA bots Agents do not support virtual machines (VMs).

5.2 Discussion

Web Security testing results showed that the websites HEI3, NW1, GW3, and CW2 have outdated third-party software or CMS components, these components should be updated as soon as possible because these websites are vulnerable to publicly known vulnerabilities. In addition, the security results showed that nine websites had failed the privacy policy test, where they have no privacy policy page or are not easily accessible, which causes failure in the EU GDPR compliance test.

Only two websites (HEI2 and HEI4) passed both EU GDPR and PCI DSS tests. On the other hand, the NW3 website passed the EU GDPR test and failed PCI DSS, while 75% (12/16) of the websites failed both tests. In PCI DSS verification, we found eleven websites do not comply with Requirement 6.6. Only HEI and one Governmental website have passed that test, whereas all other Governmental, News, and ECommerce websites failed Requirement 6.6 verification. eCommerce websites failed the most in the Cookies Protection and Cookies Disclaimer tests, whereas government websites failed the most in the Privacy Policy verification.

The security testing results of HEI websites were better than other categories, where none of the eCommerce and governmental websites passed the EU GDPR and PCI DSS compliance tests. Governmental websites failed the most in the privacy policy test, and eCommerce websites failed the most in the cookie protection test

Another crucial HTTP Headers test is X-Powered-By, which hackers may use to target a specific technology or version. We found that the servers of six websites reveal their version, potentially allowing for more assaults against them.

It's worth noting that all websites have issues with different required HTTP headers. Moreover, all websites failed the Content Security Policy test, which functions as a gatekeeper for the website, limiting where data can come from. Therefore, it's critical to configure it to avoid cross-site scripting vulnerabilities.

In the SSL testing, we found that most websites have an excellent SSL test grade, where around 81% (13/16) of the websites have a grade higher than A-. However, HEI websites have the lowest SSL grade as HEI1 has a grade of B- and HEI2 has a grade of B+. 31% (5/16) of the websites failed in all PCI DSS, HIPAA, and NIST tests, 25% (4/16) of the websites failed only in PCI DSS, while there is one website (CW4) failed in HIPAA and NIST tests, although it’s passed PCI DSS test. Therefore, fixing the highlighted SSL issues is essential to secure online transactions, protect customer data, and stop hackers from accessing or altering data transferred between two systems.

Most government websites had superior SSL test results than other categories, with three governmental websites passing the PCI DSS, HIPAA, and NIST compliance tests. TLS 1.0 is active on nine websites across all samples, which is non-compliant with NIST since SP 800-52 REV. 2 and non-compliant with PCI DSS since June 30, 2018

We found that 75% (12/16) of the websites do not support HSTS. Protecting consumers from passive eavesdroppers and active man-in-the-middle (MITM) attacks is critical. It also

imposes strong security measures such as blocking mixed content and click-through certificate overrides and protecting against web server errors such as loading JavaScript over an insecure connection [53].

The performance results showed that many websites returned errors to some users due to the high traffic, it was the same error for all websites which is 503 Service Unavailable server error response code, which indicates that the server is not ready to handle the requests. Such issues may lead to poor usability, customer dissatisfaction and tarnished brand image.

Due to Internet connectivity limitations in Palestine, we preferred to run the performance scripts using the most common networks in Palestine which are 3G mobile network and 8 Mbps LAN network. the results showed that the LAN network has better speed in loading the web pages where the average response time of all websites in 8 Mbps network was 6458 msec while it was 10604 msec in 3G mobile network. Websites developers should take these networks limitations in their consideration by compressing and optimize the website's images and videos, reducing your redirects, caching the web pages, minifying Minify HTML, JavaScript, and CSS files, and deleting unnecessary plugins.

However, the loading time of the selected websites should be enhanced more, as Kissmetrics founded that 47% of users expect web pages to load in under two seconds, and 40% leave websites that take longer than three seconds to load, and a one-second delay in page response can result in a 7% drop in conversions. As per Google/SOASTA Research, 2017 [51] they reported that the probability of a bounce increases by 32% when page time load goes from 1 to 3, 90% probability of bounce expected when the page time load goes from 1 second to 5 seconds. When Page Time Load goes from 1 to 6 seconds, the probability of bounce increases by 106%, while the probability of a bounce increases by 123% when the page time load goes from 1 to 10 seconds.

The accessibility testing results showed that all websites have critical accessibility issues. As a result, we can conclude that these websites cannot be used effectively or satisfactorily by all stakeholders, particularly disabled prospective users. None of the evaluated websites achieved the required conformance level A. Comparing the number of errors per website, websites NW1 has the highest number of errors, with 280. Next is CW3, with 200 errors. CW3 has the highest number of contrast errors for the contrast errors tests, with 162 errors. Next is CW2, with 124 errors. Based on the above accessibility results, we recommend developers and administrators follow accessibility standards and guidelines like WCAG 2.0 when creating websites with many stakeholders.

5.3 Conclusion and Future Work

In this research, we used RPA Bots to evaluate the performance, security and accessibility based on many international standards and guidelines, and it showed a high capability in evaluating the usability of all websites by integrating the RPA bots with different automation tools and API services. Moreover, the ability to connect the RPA bots with database to store and read the and will help businesses in evaluating the usability of their products more effectively, saving time, money, and effort. The results indicated different major security issues in the many local websites. Moreover, all websites failed in accessibility testing where none of the analyzed websites met the required compliance level A. Other problems indicated in the performance experiments as well where 50% (8/16) of the websites have returned errors to some users in the 8 Mbps experiment due to the high traffic, while the percentage was 31% (5/16) of the websites in the 3G network experiment.

As a future work, we can integrate the Robotic Process Automation Bots with other testing tools that follow other international standards to verify websites and mobile applications as well. in addition to that, developing an RPA integration with eye tracking and heat maps is

an important research area to help developers in verifying the usability of the products more effectively.

References

[1] I. Telecommunication Union, *Measuring digital development - Facts and figures 2021*. 2021.

[2] D. Dang Abstract Author Dang, "Performance testing for best case," 2020.

[3] M. A. G. Andurkar, "Human-computer interaction," *International Research Journal of Engineering and Technology*, 2015, [Online]. Available: www.irjet.net

[4] A. Beirekdar, J. Vanderdonckt, and M. Noirhomme-Fraiture, "KWARESMI-Knowledge-based Web Automated Evaluation with REconfigurable guidelineS optiMIzation KWARESMI 1-Knowledge-based Web Automated Evaluation with REconfigurable guidelineS optiMIzation," 2002. [Online]. Available: https://www.researchgate.net/publication/2564953

[5] A. Beirekdar, M. Keita, M. Noirhomme, F. Randolet, J. Vanderdonckt, and C. Mariage, "LNCS 3585 - Flexible Reporting for Automated Usability and Accessibility Evaluation of Web Sites," 2005. [Online]. Available: http://www.info.fundp.ac.behttp://www.isys.ucl.ac.be/bchi

[6] T. Issa and P. Isaias, "Usability and Human Computer Interaction (HCI)," in *Sustainable Design*, Springer London, 2015, pp. 19–36. doi: 10.1007/978-1-4471-6753-2_2.

[7] L. Masip, C. Martinie, M. Winckler, P. Palanque, T. Granollers, and M. Oliva, "LNCS 7623 - A Design Process for Exhibiting Design Choices and Trade-Offs in (Potentially) Conflicting User Interface Guidelines," 2012.

[8] J. Nielsen and Hoa. Loranger, *Prioritizing Web usability*. New Riders, 2006.

[9] M. Swaak, M. de Jong, and P. de Vries, "Effects of information usefulness, visual attractiveness, and usability on web visitors' trust and behavioral intentions," *IEEE*

International Professional Communication Conference, 2009, doi: 10.1109/IPCC.2009.5208719.

[10] R. Safavi, "Interface design issues to enhance usability of E-commerce websites and systems," *ICCTD 2009 - 2009 International Conference on Computer Technology and Development*, vol. 1, pp. 277–281, 2009, doi: 10.1109/ICCTD.2009.23.

[11] F. Montero, P. González, M. Lozano, and J. Vanderdonckt, "Quality Models for Automated Evaluation of Web Sites Usability and Accessibility."

[12] M. Matera, F. Rizzo, and G. T. Carughi, "Web usability: Principles and evaluation methods," in *Web Engineering*, Springer Berlin Heidelberg, 2006, pp. 143–180. doi: 10.1007/3-540-28218-1_5.

[13] J. L. Cui, "Performance testing for best case," *International Journal of Pavement Engineering*, Jul. 2021.

[14] A. Bhatnagar, A. P. Sinha, and A. Sen, "Role of navigational ability in website visit duration," *Eur J Mark*, vol. 53, no. 5, pp. 972–988, Jun. 2019, doi: 10.1108/EJM-10-2017-0719.

[15] R. Page, T. Ash, and M. Ginty, *Landing page optimization: The definitive guide to testing and tuning for conversions*. John Wiley & Sons, 2012.

[16] B. Arkin, S. Stender, and G. McGraw, "Software penetration testing," *IEEE Secur Priv*, vol. 3, no. 1, pp. 84–87, 2005.

[17] "Web Browser Security - Source Defense." https://sourcedefense.com/glossary/web-browser-security/ (accessed Sep. 02, 2022).

[18] S. L. Henry, "Understanding web accessibility," in *Web Accessibility*, Springer, 2006, pp. 1–51.

[19] "Understanding Success Criterion 3.1.1: Language of Page." https://www.w3.org/WAI/WCAG21/Understanding/language-of-page.html (accessed Sep. 02, 2022).

[20] T. Taulli, "The robotic process automation handbook," *The Robotic Process Automation Handbook. https://doi. org/10.1007/978-1-4842-5729-6*, 2020.

[21] B. Axmann and H. Harmoko, "Robotic process automation: An overview and comparison to other technology in industry 4.0," in *2020 10th International Conference on Advanced Computer Information Technologies (ACIT)*, 2020, pp. 559–562.

[22] R. Syed *et al.*, "Robotic process automation: contemporary themes and challenges," *Comput Ind*, vol. 115, p. 103162, 2020.

[23] M. K. Y. Shambour, "Assessing the Usability of Hajj and Umrah Websites," in *2021 International Conference on Information Technology (ICIT)*, 2021, pp. 876–881.

[24] A. Albesher and T. Alhussain, "Privacy and security issues in social networks: An evaluation of Facebook," *ACM International Conference Proceeding Series*, pp. 7–10, 2013, doi: 10.1145/2503859.2503861.

[25] Avrasya University, Institute of Electrical and Electronics Engineers. Turkey Section., Society of Digital Information and Wireless Communications, and Institute of Electrical and Electronics Engineers, *ICDIPC2019 proceedings: 2019 Seventh International Conference on Digital Information Processing and Communications (ICDIPC) : Avrasya University, Trabzon, Turkey, May 2-4, 2019.*

[26] N. Nouman and A. Umer, "Web Navigation and Usability Analysis of Educational Websites in Pakistan," *Proceedings - 2019 7th International Conference on Digital Information Processing and Communications, ICDIPC 2019*, pp. 57–62, May 2019, doi: 10.1109/ICDIPC.2019.8723704.

[27] T. Issa and A. Turk, “Applying Usability and HCI Principles in Developing Marketing Websites,” 2012. [Online]. Available: www.mirlabs.net/ijcisim/index.html

[28] R. Law and C. Ngai, “Usability of travel websites: A case study of the perceptions of Hong Kong travelers,” *Journal of Hospitality and Leisure Marketing*, vol. 13, no. 2, pp. 19–31, Oct. 2005, doi: 10.1300/J150v13n02_03.

[29] Jordan University of Science & Technology, Institute of Electrical and Electronics Engineers. Jordan Section, and Institute of Electrical and Electronics Engineers, *2019 10th International Conference on Information and Communication Systems (ICICS) : 11-13 June, 2019, Jordan University of Science and Technology, Irbid, Jordan.*

[30] Ş. S. Macakoğlu, S. Peker, and İ. T. Medeni, “Accessibility, usability, and security evaluation of universities’ prospective student web pages: a comparative study of Europe, North America, and Oceania,” *Univers Access Inf Soc*, 2022, doi: 10.1007/s10209-022-00869-9.

[31] Y. Akgül, “Accessibility, usability, quality performance, and readability evaluation of university websites of Turkey: a comparative study of state and private universities,” *Univers Access Inf Soc*, vol. 20, no. 1, pp. 157–170, Mar. 2021, doi: 10.1007/s10209-020-00715-w.

[32] B. Csontos and I. Heckl, “Accessibility, usability, and security evaluation of Hungarian government websites,” *Univers Access Inf Soc*, vol. 20, no. 1, pp. 139–156, Mar. 2021, doi: 10.1007/s10209-020-00716-9.

[33] N. Elisa, “Usability, Accessibility and Web Security Assessment of E-government Websites in Tanzania,” 2017. [Online]. Available: http://www.ega.go.tz/

[34] Y. Liu and Z. Li, “Experimental evaluation on government portal website’s usability : To 11 government websites of Zhejiang province,” *2nd International Conference on*

Information Science and Engineering, ICISE2010 - Proceedings, pp. 2076–2078, 2010, doi: 10.1109/ICISE.2010.5688953.

[35] A. Masum, "Security Analysis of Government & Financial Websites of Bangladesh," 2022, doi: 10.5815/ijeme.2022.02.03.

[36] A. A. Ali and M. Zamri Murah, "Security Assessment of Libyan Government Websites," in *2018 Cyber Resilience Conference (CRC)*, Nov. 2018, pp. 1–4. doi: 10.1109/CR.2018.8626862.

[37] M. S. Al-Sanea and A. A. Al-Daraiseh, "Security evaluation of Saudi Arabia's websites using open source tools," *2015 1st International Conference on Anti-Cybercrime, ICACC 2015*, Dec. 2015, doi: 10.1109/ANTI-CYBERCRIME.2015.7351928.

[38] R. Kainda, I. Flechais, and A. W. Roscoe, "Security and usability: Analysis and evaluation," in *ARES 2010 - 5th International Conference on Availability, Reliability, and Security*, 2010, pp. 275–282. doi: 10.1109/ARES.2010.77.

[39] N. A. Bt Mohd and Z. F. Zaaba, "A Review of Usability and Security Evaluation Model of Ecommerce Website," *Procedia Comput Sci*, vol. 161, pp. 1199–1205, Jan. 2019, doi: 10.1016/J.PROCS.2019.11.233.

[40] A. Kwangsawad, A. Jattamart, and P. Nusawat, "The Performance Evaluation of a Website using Automated Evaluation Tools," Dec. 2019. doi: 10.1109/TIMES-iCON47539.2019.9024634.

[41] M. Ciugudean and D. Gorgan, "Methodology for Identification and Evaluation of Web Application Performance Oriented Usability Issues," 2016.

[42] A. Nurshuhada, R. O. M. Yusop, A. Azmi, S. A. Ismail, H. M. Sarkan, and N. Kama, "Enhancing performance aspect in usability guidelines for mobile web application," *International Conference on Research and Innovation in Information Systems, ICRIIS*, vol. December-2019, Dec. 2019, doi: 10.1109/ICRIIS48246.2019.9073617.

[43] “ImmuniWeb – :: Exire Technologies ::” https://www.exiretechnologies.com/htb/ (accessed Sep. 05, 2022).

[44] S. Nagpure and S. Kurkure, “Vulnerability assessment and penetration testing of web application,” in *2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, 2017, pp. 1–6.

[45] M. Luo, P. Laperdrix, N. Honarmand, and N. Nikiforakis, “Time does not heal all wounds: A longitudinal analysis of security-mechanism support in mobile browsers,” 2019.

[46] R. K. Lenka, S. Mamgain, S. Kumar, and R. K. Barik, “Performance analysis of automated testing tools: JMeter and TestComplete,” in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, 2018, pp. 399–407.

[47] S. Matam and J. Jain, *Pro Apache JMeter: web application performance testing*. Apress, 2017.

[48] M. Bajammal and A. Mesbah, “Semantic Web Accessibility Testing via Hierarchical Visual Analysis,” in *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, 2021, pp. 1610–1621.

[49] “Determining Concurrent Users from Web Analytics - LoadView.” https://www.loadview-testing.com/blog/determining-concurrent-users-from-web-analytics/ (accessed Aug. 07, 2022).

[50] P. Saripalli and B. Walters, “Quirc: A quantitative impact and risk assessment framework for cloud security,” in *2010 IEEE 3rd international conference on cloud computing*, 2010, pp. 280–288.

[51] “Google: How Bounce Rate Increases With Longer Mobile Load Times.” http://www.thesempost.com/google-bounce-rate-increases-longer-load-times/ (accessed Aug. 07, 2022).

الملخص

نظام أتمتة عمليات روبوتية Robotic Process Automation لفحص قابلية استخدام المواقع الإلكترونية من حيث سرعة الأداء والأمان.

يتزايد استخدام المواقع الإلكترونية بشكل كبير، وأصبح الوصول اليها أسهل بكثير نظرا لاتساع رقعة انتشار الشبكة العنكبوتية حول العالم. ومما أدى الى وجود حاجة في توفير تجربة مستخدم فعالة وجيدة (UX) والتي أصبحت جزءًا أساسيًا من تطوير التطبيقات، البرامج وبما في ذلك المواقع الإلكترونية. وبالرغم من الاهتمام في توفير تجربة مستخدم فعالة والحرص على رضا زوار الموقع، ألا ان رضا الزبائن عن كثير من المواقع ما زال محل نقاش.

إن تصميم وتطوير تطبيق المواقع الالكترونية المتوافقة مع قابلية الاستخدام ومبادئ وإرشادات UX وHCI يعد مهمة معقدة وتحتاج للكثير من الاختبارات، هناك العديد من ادوات الاختبار المتاحة للتحقق من اتباع هذه المواقع لمبادئ HCI وضمان رضا الزوار. معظم هذه الأدوات لديها صلاحيات محدودة بسبب تكلفة الاستخدام العالية والوقت المستغرق. الهدف من هذا البحث هو تسهيل عملية اختبار تجربة المستخدم وتحسين أداء المواقع الالكترونية وأمانها وإمكانية الوصول إليها من خلال تطوير نظام متكامل لاختبار تجربة المستخدم يعتمد على أتمتة العمليات الروبوتية (Robotic Process Automation) وفحص أداء وسرعة الموقع والأمان وإمكانية الوصول لأي موقع الكتروني في نظام واحد.

ناقشت الدراسات السابقة أهمية اتباع المواقع الالكترونية لإرشادات HCI وقام العديد من الباحثين بإجراء اختبارات على عينات مواقع الكترونية مختلفة كمواقع تعليمية وحكومية وفحص مدى اتباع هذه المواقع لإرشادات HCI واستخدم بعضهم طرقا تقليدية واستخدم آخرون أدوات فحص أتمتة متعددة لفحص الأمان والأداء وغيرها من المعايير والمتغيرات. ولعل أبرز ما لاحظناه هو عدم وجود نظام واحد لفحص جميع هذه المعايير، حيث يتطلب استخدام أكثر من أداة وأكثر من اشتراك حتى يتم فحص جميع المعايير.

في النظام المقترح، قمنا بتطوير روبوتات أتمتة العمليات الروبوتية (Robotic Process Automation Bots) المتصلة والمتكاملة مع أدوات مختلفة لفحص الأداء والأمان وإمكانية الوصول، حيث تم ربط النظام بأداة JMeter لفحص أداء وسرعة الموقع، أداة ImmuniWeb لفحص الأمان، وWAVE لفحص قابلية الوصول حسب معايير وارشادات HCI. لاختبار النظام، قمنا باختيار ستة عشر موقعا الكترونيا محليا فلسطينيا من أربع قطاعات ومجالات مختلفة وهي التعليم العالي

والحكومي والإخباري والتسويق الالكتروني، وقد تم مراجعة النتائج وتسليط الضوء على أبرز المشاكل في كل موقع الكتروني.

من خلال النظام المقترح، وبناء على نتائج الاختبارات المختلفة، وجدنا ان توظيف روبوتات أتمتة العمليات الروبوتية (Robotic Process Automation Bots) هي طريقة فعالة لتحسين تجربة المستخدم وزيادة رضا الزبائن عن الموقع الالكتروني، بالإضافة الى تحسين الأداء، والأمان، وتوفير المال، والوقت والجهد.