School of Information Technology and Engineering at the ADA University

School of Engineering and Applied Science
at the George Washington University

COMPARATIVE ANALYSIS OF AZERBAIJANI E-GOVERNMENT WEBSITES FOR EFFICIENCY IMPROVEMENT

A Thesis
Presented to the Graduate Program of Computer Science and Data Analytics
of the School of Information Technology and Engineering
ADA University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Computer Science and Data Analytics
ADA University

By
Emil Imanov

April, 2023

THESIS ACCEPTANCE

This Thesis by: Emil Imanov
Entitled: *Analysis of Azerbaijani e-government websites for efficiency improvement*

has been approved as meeting the requirement for the Degree of Master of Science in Computer Science and Data Analytics of the School of Information Technology and Engineering, ADA University.

Approved:

(Advisor)	28 Apr 2023 (Date)
(Program Director)	28 Apr 2023 (Date)
(Dean)	28 Apr 2023 (Date)

ABSTRACT

This Master Thesis analyzes the current status of e-government portals in Azerbaijan, specifically selected five modules of e-taxes.gov.az websites. The analysis is comparative, where there are selected 5 international highest-ranking e-tax portals according to the UN to compare the selected criteria. The analysis includes selection of performance metrics and collect data on response time and availability of services. In addition, it involves front-end analysis, accessibility, and back-end analysis. Some steps consist of using ready tools such as webpage response time checker and webpage stack extraction. The front-end analysis method is the python code, which collects HTML data from webpages to check for best practices. According to the analysis, front end of some modules needs to be modernized. To collect the availability and performance data webpage monitoring tool was used. The data proved, that the services are down during the peak times. The back end was checked to understand the stack used, such as programming languages, frameworks, libraries, servers, and security certificates. This had shown that each module was developed totally different, and there is not standardized approach to the development of the services. This creates challenges on cross services data sharing.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to all my university professors, who had played a significant role in my personal and professional development through the courses. They enabled me to be able to complete the master thesis, where I applied all the knowledge gained from them in the University. Separate gratitude for my mentor Professor Abzetdin Adamov, who directed me all the time and provided his valuable comments for successful completion of this Master Thesis.

My sincere thanks go to my fellow researchers and colleagues, who have provided me with invaluable assistance and intellectual stimulation during the course of this research. I am particularly thankful to colleagues and public sector experts who generously shared their knowledge, resources, and expertise to help me overcome various challenges in my research.

In conclusion, I am grateful to everyone who has played a part in making this thesis a reality, and I dedicate this work to all those who have supported me along the way.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Explanation
E-Gov	Electronic Government portal
SME	Small and Medium Enterprise
E-Taxes	The web-based application, where the user reports taxes
ASAN login	The login system provided by the Government of Azerbaijan
ID	Identification
ERP	Enterprise Resource Planning
ICT	Information Communication Technologies
DEA	Data Envelopment Analysis
SSPF	Social Security Protection Fund
HSTS	HTTP Strict Transport Security
HTTPS	Hypertext Transfer Protocol Secure
MUI	Material User Interface
UI	User Interface
NA	Not Available

1 INTRODUCTION

The growing trend of e-government has gained significant attention in recent years due to the potential benefits of increased efficiency, cost savings, and better service delivery. In recent years Digital government has become a powerful tool in the hands of public sector officials to demonstrate the maturity level of their administrations. It is the responsibility of the government to make e-government systems user-centric and accessible to each citizen and organization, ultimately saving valuable time for both the private and public sectors. The automation of the Government services saves valuable time not only for private sector but also decreases the amount manual work of human resources of public sector agencies. However, despite significant investment in digital government, 60-85% of projects fail [1]. Furthermore, Digital Government is more dependable on the governmental budget, as compared to ERP, Financial, and e-commerce applications implemented by the enterprises [2]. This often leads to the development of low-quality digital government services with subpar design and numerous errors.

In today's world the main beneficials of information technologies are corporations and governments, who collect data, analyze them, and make decisions. To improve the administrative processes, make information public, increase the efficiency of the system the Government pushes public sector agencies to improve the services day by day [3]. But as the technologies evolve, both, businesses and public sectors struggle adapting new technologies. Increasingly, technologies evolve very fast, but Digital Government projects get limited financing to update the software and increase the efficiency of e-Gov portals. Moreover, the main potion of the users of the e-Government systems are SMEs and Microbusinesses, who struggle providing correct tax and other reporting to the government, since the services are of low quality. In fact, the e-Gov portals are not so user friendly, which complicates the reporting process for SMEs and microbusinesses. In addition, many heterogenous systems in e-Gov portals result to inefficient data flow between the independent heterogeneous systems and leads to data duplication, which creates another challenge to the Government such as abundance of the investment to the services without effective payback from them.

Many developed countries on the Governmental level investing time and money to create an architectural and legal basis for all public sector agencies on how to develop, implement, and manage data and information systems. To measure the success of the implementations, to analyze the effectiveness of the e-Gov portals many tools for web measurement and monitoring exist. The Republic of Azerbaijan has made strides in developing its e-government infrastructure, focusing on providing services through digital platforms. The government has established an e-Gov portal for tax and salary reporting services to facilitate the reporting process for all big corporates, SMEs, and microbusinesses, as required by the Azerbaijan Tax Code. In recent years, Azerbaijan has significantly increased its digital services [4], and the government has expanded the number of services provided through web-based applications.

Within this study, we will use measurement tools to analyze the current situation of the e-Gov portals of electronic tax services of the Azerbaijan Republic. This analysis will allow us to measure the current situation and compare it to the government electronic systems of developed countries with established ICT architectural principles on a governmental level. By doing so, we hope to identify opportunities for improvement and provide recommendations for a more user-centric and efficient e-government infrastructure in Azerbaijan.

1.1 Definition of the problem

The practice in the business environment shows that as the time is passing and more solutions appear, businesses face with technical difficulties because of the variety number of software and hardware solutions. One of the examples is that business environments nowadays

have many tech systems and solutions, such as social network activities, ERP systems, CRM solutions, Mail services, Web pages, and governmental portals. The businesses, especially SMEs and Microbusinesses are having difficulties to manage all these services, which causes for them to be one step behind the competitors. Moreover, the main users of e-government systems, such as SMEs and microbusinesses, often struggle to provide correct tax and other reporting to the government since the services are of low quality and not user-friendly. This results in inefficient data flow between independent heterogeneous systems, leading to data duplication and creating challenges for the government in terms of investment returns on services.

Azerbaijan, as one of the developing countries, is facing challenges in developing an architectural basis for software services provided to the private sector. The country is committed to offering high-quality digital services to citizens and enterprises, aiming to strengthen trust and connection between the public and private sectors. The root of the problem stems from the highly fragmented public sector, characterized by a collection of independent and heterogeneous systems. Various agencies and ministries are keen to create their own software services and store data independently, resulting in a lack of information exchange and smooth intercrossing process flow between public sector services. This fragmentation creates difficulties for the private sector in providing user-friendly reports. To address these limitations and inefficiencies, there is a need for a single, integrated web-platform that would enhance hardware usage efficiency, reduce data duplication, and minimize manual work for the private sector.

One of the main pain points of the businesses is dealing with their reporting to the public sectors. Even though Azerbaijan is building digital portals to supervise country economy, unfortunately, there is no architectural solution, which will guarantee that the platform will be easy to implement and use. The information systems should be easy to navigate, have attractive design, goal oriented, and effective [5].

The financial information systems are expensive solution and challenging to implement [6]. On the other side, there are plenty amount of heterogeneous governmental portals, where businesses must report and proceed tax or other commission transactions. In most cases many heterogeneous systems cause data duplication and redundant data transmission between the systems, which adds additional efforts on dealing with the services, and does not allow reusing the solutions between the agencies.

Meanwhile, financial information systems' user friendliness is dependent on the business financial possibilities, governmental portals are regulated by public sector officials, who do not invest enough financing for making the platforms user friendly. Due to this fact, it is difficult to navigate through these portals and report all the required reporting. (Example: US visa application portal is not user friendly). Study shows that, since e-governments are mandatory to use platforms, the platform owners do not want to improve, since in any case it will be used [2]. Inefficient software building causes additional costs and loads the systems.

1.2 Significance of the Problem

The research problem is the inefficiency of existing e-government websites in Azerbaijan, which hinders their effectiveness and usability. The lack of a common digital architecture as a framework for public sector e-Gov services and the implementation of independently developed systems without alignment cause issues such as hardware inefficiency, data duplication, and manual work duplication in the private sector. Moreover, the current web-based applications need to be evaluated to measure their effectiveness, and tools for evaluating them empirically must be identified. Therefore, the research questions for this study are as follows:

1. What is the current state of e-government websites in Azerbaijan, and what are the inefficiencies and issues present in their architecture?
2. How can the effectiveness of current e-government websites be evaluated, and what tools can measure this effectiveness empirically?

In recent years, Azerbaijan has significantly increased its digital services, and the government continues to expand the number of services it provides. These web-based applications need to be evaluated, and appropriate evaluation tools must be determined to measure their effectiveness. This evaluation will empirically prove that current web applications are outdated and need to be implemented under a single architectural basis.

The web-portal has several interconnected modules with separate login processes for each module. For example, to switch between the webmail module and report submission, users must log out from one module and log in to another with the same credentials. Moreover, if there is an unread e-mail in the webmail module and the user first signs in to the reporting module, the system prompts an error and requests that e-mails be read before submitting any reports. If the user closes the reporting module without properly signing out and returns to the login page, they will not be able to log in to any modules, as the system prompts that there is an active session and asks to try again after 10 minutes. These findings in the system have prompted the need to analyze the e-Gov portal, especially tax services.

1.3 Review of Significant Research

The main objective of this study is to analyze the efficiency of the Azerbaijani e-government websites and propose recommendations for improvement. Specific research objectives include:

1. To identify the current status of e-government websites in Azerbaijan and their shortcomings in terms of usability and functionality.
2. To analyze the user experience of the selected e-government websites in terms of ease of use, accessibility, and satisfaction.
3. To identify the factors affecting the efficiency of e-government websites and propose solutions for improvement.

The problem is significant to research for several reasons. First of all, it will provide an overview of the current state of e-taxes.gov.az websites and identify their shortcomings. This will help the public agencies and Government to prioritize their efforts for improvement. Second, the study will provide insights into the user experience of e-government websites, which will help to improve the usability and accessibility of these websites from the user perspective. Third, the study will analyze the front end, architecture, security aspects of the portal for improving the efficiency of the websites, which will help to increase the efficiency of government services and reduce administrative costs. The study will also contribute to the existing literature on e-government and user experience by providing a case study from Azerbaijan.

1.4 Assumptions and limitations

There are some limitations to this study. One of the main challenges is the unknown software and hardware architecture of the existing Digital Government systems in public sector agencies, which were implemented as standalone solutions. This lack of documentation of the backend design, including software requirements and software design documents, makes it difficult to measure the existing systems from the backend perspective. For example, we can measure the functionality of e-taxes.gov.az from the private sector user perspective and analyze the efficiency of the system. However, there are no publicly available Software Requirements Document and Software Design Document for the e-taxes.gov.az portal to read and analyze the architecture. This limitation restricts our ability to estimate the algorithms and patterns

design of the existing portals. Despite this challenge, we plan to use webpage measurement tools and conduct user tests to analyze these portals and identify areas for improvement.

2 LITERATURE REVIEW

The emergence of e-government has been revolutionized by the way governments interact with citizens and businesses, offering a more efficient, accessible, and transparent platform for the delivery of public services[7]. As nations around the world strive to improve their e-government systems, the evaluation of their efficiency becomes a critical aspect of this process. In this context, the present literature review is aimed at exploring the current state of Azerbaijani e-government websites and identifying opportunities for efficiency improvement.

This literature review will be initiated by examining the concept of e-government and its potential benefits, focusing on the aspects of efficiency, transparency, and accessibility[8]. Challenges and barriers faced by developing countries, such as Azerbaijan, in implementing e-government systems[9] will then be delved into. A thorough understanding of these challenges is essential for the identification of potential areas of improvement in Azerbaijani e-government websites.

Subsequently, the role of digital architecture in enabling seamless integration and interoperability of e-government systems, crucial for enhancing efficiency, will be explored[10]. The best practices and lessons learned from other countries that have successfully implemented efficient e-government systems will be discussed, drawing insights that could be applied to the Azerbaijani context[11].

Various evaluation methods and tools used to measure the efficiency of e-government websites will be examined[12]. These evaluation techniques will provide a solid foundation for the assessment of the current state of Azerbaijani e-government websites and the identification of potential areas for improvement.

By reviewing the existing literature on e-government efficiency and digital architecture, this literature review will illuminate the current state of Azerbaijani e-government websites and lay the groundwork for the development of recommendations and strategies to enhance their efficiency, ultimately contributing to more effective and user-friendly public services for the citizens and businesses of Azerbaijan[13].

2.1 Conceptual framework and theoretical background

In recent years, e-government has become a global phenomenon that has changed the way governments interact with citizens [14]. The concept of e-government is a complex one, with many definitions and interpretations. According to the United Nations, e-government refers to "the use of information and communication technologies (ICTs) to improve the activities of public sector organizations, the provision of services to citizens, and the interaction between the government and its stakeholders" [15]. E-government is seen as a means to improve government efficiency, transparency, and accountability [16].

One of the key components of e-government is the web portal. E-government websites serve as a portal for citizens and enterprises to access government services and information [17]. Therefore, it is important that e-government websites are designed to meet the needs of citizens and enterprises, are easy to use, and provide reliable and accurate information [18].

To design and evaluate e-government websites, several models and frameworks have been proposed in the literature. One such framework is the e-government website evaluation framework (EGWEF) proposed by Alshehri et al. [19]. The EGWEF is a comprehensive framework that consists of 10 dimensions, including usability, content, and security, that are used to evaluate e-government websites.

In addition to the EGWEF, several other models and frameworks have been proposed for evaluating e-government websites. For example, the e-government service quality (EGSQ)

model proposed by Lin and Lee [20] focuses on the quality of e-government services. The model consists of three dimensions: information quality, system quality, and service quality.

Another important aspect of e-government is the concept of digital divide. The digital divide refers to the unequal distribution of access to and use of ICTs among different groups in society [21]. According to a report by the World Bank, the digital divide can be divided into four dimensions: access, usage, quality of access, and social and economic impacts [22].

Therefore, in this thesis, the conceptual framework and theoretical background will focus on e-government website evaluation frameworks, the e-government service quality model, and the digital divide. These frameworks will provide a theoretical basis for evaluating the efficiency of Azerbaijani e-government websites and will guide the development of research questions and hypotheses.

Several studies have been conducted to analyze the effectiveness of e-government services in different countries. However, there is limited research on the efficiency of e-government services in Azerbaijan, and the available studies mainly focus on the legal and policy framework of e-government in the country.

Previous research has identified several gaps and challenges in the implementation of e-government services in Azerbaijan. For instance, the lack of a common digital architecture for public sector services has resulted in inefficiencies and duplication of work [23]. Furthermore, the absence of proper documentation and design of existing e-government systems has made it difficult to evaluate their efficiency from a technical perspective.

2.2 Key concepts and definitions

To gain a deeper understanding of the context and language used in this thesis, we must first clarify some essential terms related to e-government and efficiency.

E-government encompasses the application of digital technologies, such as the internet, to offer government services and information to citizens, businesses, and various stakeholders [24]. The implementation of e-government has the potential to enhance government efficiency, transparency, and accountability while also promoting greater citizen involvement and interaction [25].

In the realm of e-government, efficiency refers to the degree to which digital technologies contribute to streamlining government processes, reducing expenses, and elevating service quality [26]. User-friendly, secure, and universally accessible e-government systems should be available to all citizens, irrespective of their socio-economic standing or geographical location [27].

Digital architecture represents the foundational framework that allows the integration and interoperability of e-government systems [28]. A well-structured digital architecture can enable the smooth exchange of data and information among various government agencies and departments, ultimately leading to more efficient and effective service provision [29].

Throughout this thesis, the fundamental concepts of e-government, efficiency, digital architecture, and related terms will be employed in the examination and enhancement of Azerbaijan's existing e-government systems.

2.3 E-government models and architectures

E-government models and architectures refer to the various frameworks and structures that governments use to deliver public services to citizens electronically. Over the years, different models and architectures have been proposed and implemented in various countries. This section discusses some of the most used e-government models and architectures. There are 3 different e-government models employed in various countries: Government-to-Citizen, Government-to-Business, and Government-to-Government [30].

E-Government Models:

1. The G2C (Government-to-Citizen) Model

The G2C model is the most commonly used e-government model, where government agencies provide services directly to citizens. The main aim of this model is to improve the accessibility, transparency, and accountability of government services to citizens [31].

2. The G2B (Government-to-Business) Model

The G2B model is focused on delivering services to businesses and private sector organizations. The aim of this model is to improve the efficiency and effectiveness of business processes by reducing bureaucracy and streamlining the interactions between businesses and government agencies [35].

3. The G2G (Government-to-Government) Model

The G2G model focuses on inter-agency cooperation and information sharing. It is used to improve coordination between different government agencies, reduce duplication of efforts, and enhance the delivery of government services [32].

The are many architectures proposed by different countries for e-government service provision. There are 4 architectures selected and described, which are centralized, decentralized, federated, and service-oriented architectures [33].

a) Centralized Architecture

In a centralized architecture, a single agency or department is responsible for the delivery of e-government services. This model is relatively easy to implement and manage, but it can lead to issues related to scalability and interoperability [34].

b) Decentralized Architecture

In a decentralized architecture, different government agencies are responsible for the delivery of e-government services in their respective domains. This model offers greater flexibility and autonomy, but it can lead to issues related to standardization and coordination.

c) Federated Architecture

In a federated architecture, different government agencies maintain their own e-government systems, but there is a shared infrastructure that enables data and information exchange between them. This model offers a balance between centralization and decentralization, and it can lead to better coordination and interoperability.

d) Service-Oriented Architecture (SOA)

Service-Oriented Architecture (SOA) is a modular and flexible approach to e-government architecture, where government services are delivered through reusable and interoperable components. This model enables the integration of disparate systems and promotes the development of new services through the reuse of existing components.

Overall, the selection of an appropriate e-government model and architecture is dependent on the specific needs and requirements of a government agency. A careful evaluation of the available options should be conducted to ensure the successful implementation of e-government services.

2.4 Efficiency improvement in e-government websites

Efficiency improvement is an important aspect of e-government websites, as it can enhance user satisfaction and reduce costs for both users and governments [34]. In order to improve the efficiency of e-government websites, it is essential to consider various factors, including website design, website content, website functionality, website accessibility, website security, and website usability [35].

According to the UN last survey on E-government, the Denmark, Finland, Republic of Korea, Sweden, and Iceland are the highest-ranking e-gov portals in the service provision quality [13]. Figure 1 is the table of top 15 highest ranking countries according to the UN.

Figure 1: 2022 United Nations E-Government Surveys. (https://desapublications.un.org/sites/default/files/publications/2022-11/Web%20version%20E-Government%202022%20November%2010.pdf)

Country name	Rating class	Region	OSI	HCI	TII	EGDI (2022)	EGDI (2020)
Denmark	VH	Europe	0.9797	0.9559	0.9795	0.9717	0.9758
Finland	VH	Europe	0.9833	0.9640	0.9127	0.9533	0.9452
Republic of Korea	VH	Asia	0.9826	0.9087	0.9674	0.9529	0.9560
New Zealand	VH	Oceania	0.9579	0.9823	0.8896	0.9432	0.9339
Sweden	VH	Europe	0.9002	0.9649	0.9580	0.9410	0.9365
Iceland	VH	Europe	0.8867	0.9657	0.9705	0.9410	0.9101
Australia	VH	Oceania	0.9380	1.0000	0.8836	0.9405	0.9432
Estonia	VH	Europe	1.0000	0.9231	0.8949	0.9393	0.9473
Netherlands	VH	Europe	0.9026	0.9506	0.9620	0.9384	0.9228
United States of America	VH	Americas	0.9304	0.9276	0.8874	0.9151	0.9297
United Kingdom of Great Britain and Northern Ireland	VH	Europe	0.8859	0.9369	0.9186	0.9138	0.9358
Singapore	VH	Asia	0.9620	0.9021	0.8758	0.9133	0.9150
United Arab Emirates	VH	Asia	0.9014	0.8711	0.9306	0.9010	0.8555
Japan	VH	Asia	0.9094	0.8765	0.9147	0.9002	0.8989
Malta	VH	Europe	0.8849	0.8734	0.9245	0.8943	0.8547

For comparison purposes e-gov portals, specifically tax services of 5 high ranking countries were selected.

Website design is an important factor that affects the efficiency of e-government websites. Good website design can improve user satisfaction, reduce user errors, and increase user productivity [36]. Website content is another factor that affects the efficiency of e-government websites. Good website content can provide users with relevant and useful information, which can help them to complete tasks more efficiently [37]. Website functionality is also important, as it can provide users with the necessary tools and functions to complete tasks effectively [38].

Website accessibility is a critical factor in improving the efficiency of e-government websites. Accessibility can ensure that all users, regardless of their abilities, can access and use the website effectively [39]. Website security is also essential, as it can protect users' personal and sensitive information from unauthorized access [40]. Finally, website usability is a crucial factor that affects the efficiency of e-government websites. Usability can ensure that users can navigate and use the website easily and efficiently [41].

In summary, improving the efficiency of e-government websites requires considering various factors, including website design, website content, website functionality, website accessibility, website security, and website usability. By addressing these factors, governments can enhance user satisfaction, reduce costs, and provide better services to their citizens.

2.5 User-centered design and evaluation methods

User-centered design (UCD) and evaluation methods play an essential role in the development of user-friendly and effective e-government websites. UCD is a design approach that emphasizes the needs and preferences of users in the design and development process [42]. It aims to improve user satisfaction, efficiency, and effectiveness by involving users throughout the design process and considering their feedback in the decision-making process [43].

UCD is a process that involves several steps, including user research, requirements gathering, design, prototyping, and evaluation [44]. Each step is iterative and involves the continuous feedback of users to ensure that the website meets their needs and preferences.

Several evaluation methods can be used to assess the usability and effectiveness of e-government websites. These include heuristic evaluation, cognitive walkthrough, and usability testing [45]. Heuristic evaluation involves the use of a set of principles to evaluate the interface design and identify potential usability issues [46]. Cognitive walkthrough focuses on the user's cognitive process and assesses the website's ease of use and learnability [47]. Usability testing involves observing users performing tasks on the website and collecting feedback on their experience [48].

- Navigation and Structure

The portal should have a clear and intuitive structure, with a navigation system that enables users to easily find the information they need. A user-friendly design should enable users to quickly locate the service they are looking for and allow them to complete their transactions quickly and easily.

- Clarity of Language

Language used in the portal should be clear, concise and jargon-free. The language used should be easy to understand for all users, regardless of their educational background. To ensure that all users can understand the content, the portal should be available in multiple languages.

- User Feedback

The portal should enable users to provide feedback on their experience. This can help improve the user experience of the portal by addressing any issues that users may have encountered. Additionally, feedback can be used to identify common issues that are affecting users and make necessary changes to the design and functionality of the portal.

- Accessibility

E-government portals should be designed with accessibility in mind, ensuring that users with disabilities can access the portal and its services. This includes making the portal compatible with assistive technologies like screen readers and ensuring that it conforms to relevant accessibility standards.

- Performance and Reliability

E-government portals should be designed to be fast, reliable, and secure. A slow and unreliable portal can be frustrating for users and may cause them to abandon their transactions. Additionally, security should be built into the portal from the start to ensure the privacy and confidentiality of user information.

The use of UCD and evaluation methods can lead to improved user satisfaction and efficiency, as well as reduced errors and costs associated with redesign and maintenance [49].

3 METHODOLOGY

The methodology section plays a crucial role in any research project, as it outlines the approach and techniques used to gather and analyze data, ensuring the credibility and reliability of the findings. In the context of this study, which aims to analyze the efficiency of Azerbaijani e-government websites, a comprehensive methodology will be employed to examine various aspects of these websites, including their front-end and back-end components, user interface, and other relevant factors.

This section will provide a detailed overview of the methodology adopted for this research, including the data collection and analysis techniques employed to achieve the research objectives. We will begin by discussing the front-end analysis, focusing on factors such as search engine optimization (SEO), website layout, and accessibility. This analysis will help to assess the visibility and usability of the e-government websites, which are essential elements of an efficient online platform.

Subsequently, we will delve into the back-end analysis, examining the underlying architecture and infrastructure that support the e-government websites. This will be achieved using various tools and techniques to analyze the performance, security, and scalability of the websites' back-end systems. By doing so, we aim to identify potential bottlenecks and weaknesses that may hinder the efficiency of the e-government platforms.

In addition to the front-end and back-end analyses, we will also investigate the user interface (UI) of the e-government websites. This will involve assessing the websites' design, navigation, and overall user experience, as these aspects play a significant role in determining the efficiency and effectiveness of the e-government services provided. Through the evaluation of the user interface, we will be able to identify areas of improvement that can enhance the overall experience of users interacting with the e-government websites.

In conclusion, the methodology section will provide a comprehensive framework for the systematic analysis of Azerbaijani e-government websites, incorporating various techniques to assess their efficiency from multiple perspectives. By employing this robust methodology, we will be better equipped to draw meaningful insights and recommendations to improve the efficiency of e-government services in Azerbaijan, ultimately benefiting both the government and the citizens it serves.

3.1 Research design and approach

The research design of the study is mixed approach, including quantitative and qualitative analysis. The study will utilize an analysis of front end design, usability testing, and interviews with e-government experts to collect qualitative data. In addition, the quantitative analysis will be done by using web analyzer tools, to collect server response time, web data size, server availability. Moreover, the qualitative analysis will include to determine the technology stack of the websites.

3.1.1 User Centered Design and Front End

The user interfaces of e-Government portals were meticulously analyzed with respect to user-friendliness, usability, functionality, and accessibility. Emphasis was placed on evaluating the ease with which users could navigate the websites, the intuitiveness of the design, and the efficiency of various functions provided by the portals. Additionally, the accessibility of these e-Government websites was assessed to ensure that they catered to a diverse range of users, including those with disabilities. In addition, technology stack also was collected to evaluate the development design. The e-gov webpages UCI will be analyzed using the following criteria.

- ***Navigation and Structure***
- ***Clarity of Language***
- ***User Feedback***
- ***Accessibility***
- ***Performance and Reliability***

The HTML tags were analyzed in two ways, by viewing source code from browser's inspect element tool and with the help of python code "html_tags.py" which is in the Appendix A. The page source investigation was commenced with an assessment of the basic structure of each HTML document, which begins with the **<!DOCTYPE html>** declaration and continues with the opening and closing **<html>** tags. Within these tags, the focus was on two main sections: **<head>** and **<body>** tags. In the **<head>** section, the analysis was emphasized on meta information such as the title, character encoding, stylesheets, and scripts. Subsequently, the **<body>** section, where the actual content of the webpage is housed was investigated. To thoroughly evaluate the content and layout, the special attention was paid to HTML tags and

their attributes, such as headings, paragraphs, lists, links, images, tables, and forms. This in-depth analysis had shown the quality and organization of the content presented on each e-Government website.

3.1.2 Performance and availability

The webpage data for 10 websites were collected using a combination of Python scripts that utilized various libraries and methods to extract relevant information. 10 pages were consisting of 5 Azerbaijani and 5 globally highest-ranking e-tax services.

The extracted information was organized into a table for each website, which was used to compare and analyze the various attributes of the websites. Overall, the use of Python scripts allowed for efficient and consistent collection of webpage data, which facilitated the analysis and comparison of the websites. The scripts were flexible and could be easily modified to extract additional information or metrics as needed.

The availability and response times of e-Government portals were monitored using the Site24x7 tool, a comprehensive website performance monitoring solution. The primary objective was to observe the trend of responsiveness during peak usage times.

3.2 Data collection and sources

The 31st of March is the deadline for the declaration of the incomes for enterprises. During this month all entities fill the yearly tax reporting form. Considering the fact, that many companies keep it “till last minute”, they all log in at the very last time to fill the form. On these days the server of e-taxes.gov.az are very loaded and suffer from difficulties.

This is the reason, why only 5 web modules of e-taxes.gov.az were selected among all Azerbaijani e-government portals. They are listed in the Table 1.

Table 1: Selected Azerbaijani E-Gov websites

E-Government	Definition	Title
e-taxes.gov.az	Tax home page	İnternet Vergi İdarəsi
vroom.e-taxes.gov.az	Document work, mailing	Virtual Ödəyici Xidməti
e-taxes.gov.az/vedop2	Tax declaration	NA
qaime.e-taxes.gov.az	Invoice	E-QAİMƏ
vp.e-taxes.gov.az	SSPF	Vergi, MDSS, İS

For the purposes of comparison, 5 other e-tax services of highest-ranking countries also were selected. The table 2 shows the list of these webpages.

Table 2: Selected high-ranking E-Gov websites

E-Government	Definition	Title
skat.dk/data.aspx	Denmark	Businesses - Skat.dk
vero.fi/en/e-file/mytax	Finland	MyTax - vero.fi
www.ird.govt.nz	New Zealand	Inland Revenue - Te Tari Taake
skatteverket.se/servicelankar	Sweden	In English | Skatteverket
skatturinn.is/english	Iceland	Iceland Revenue and Customs

3.2.1 User Centered Design and Front End

To analyze the user interface, the webpages were accessed and used. They were compared with international best practices. The inefficient steps were determined, which are unnecessary and do not add business value to the completion of the task, which can be removed. The qualitative data for this study will be collected through interviews with e-government experts

as well. The experts are selected based on their experience and knowledge of e-government website design and implementation in Azerbaijan. The interviews were conducted using a semi-structured interview guide designed to collect data on best practices, architecture, challenges, and recommendations for improving the efficiency of e-government websites.

To analyze the front end, the HTML codes of the webpages were analyzed. The HTML were accessed through the browser inspect element instrument. From the HTML the SEO optimization, HTML structure, and best practice application was analyzed.

3.2.2 Performance and Availability

The scripts were divided into several categories, each focusing on a specific aspect of the webpage:

- The requests and BeautifulSoup libraries were used to retrieve and parse the HTML content of each webpage. The scripts extracted information such as the title tag, meta description tag, meta keywords tag, H1 tag, canonical URL, and robots tag, which were organized into a table for each website. (Appendix A, Appendix B)
- The OpenSSL library was used to extract SSL security information such as the certificate issuer, subject, not before date, and not after date. The scripts also utilized the ssl and socket libraries to retrieve SSL certificate information from the server (Appendix C).
- The time and requests libraries were used to calculate various performance metrics such as response time, page load time, DNS lookup time, and connection time. The scripts used these metrics to evaluate the overall performance of each webpage (Appendix D).
- The requests library was used to retrieve the webpage content, and the len() function was used to calculate the size of the webpage in bytes. The scripts used this information to compare the size of each webpage and evaluate their efficiency (Appendix D).

By employing Site24x7, valuable data regarding the portals' ability to handle increased user traffic and maintain optimal performance during periods of high demand was collected. This analysis provided crucial insights into the robustness and efficiency of e-Government portals. The period of data collection was March of 2023.

3.3 Data analysis and tools

Data analysis is an essential part of any research study and involves the systematic examination of data collected during the research process. This study will employ both qualitative and quantitative data analysis methods to analyze the data collected from the survey, usability testing, HTMLs, python codes, and website monitoring tools. The following subsections will describe the data analysis tools that will be used in this study.

3.3.1 Qualitative Data Analysis

The qualitative data collected Wappalyzer and Builtwith. These data will consist of back end and development stack such as programming language, web server, Security, libraries, and frameworks. Another tool was semi structured interviews with experts and developers of e-gov portals. The questions were related to architecture and integration between cross services. The frontend information was collected using browser developer tools to see the source codes.

Table 3 shows the information collected from Wappalyzer. In addition, Wappalyzer had shown only for qaime.e-taxes.gov.az UI framework as Bootstrap. For vp.e-taxes.gov.az Wappalyzer had shown UI framework as MUI, which is analog for Bootstrap.

Table 3: Wappalyzer

E-Government	Prog Lang	Security	SSL/TLS	JS library	Web Server
e-taxes.gov.az	Java	HSTS	DigiCert	jQuery	Apache
vroom.e-taxes.gov.az	Java	HSTS	DigiCert	NA	Apache
e-taxes.gov.az/vedop2	Java	HSTS	DigiCert	jQuery	NA
qaime.e-taxes.gov.az	PHP	HSTS	NA	jQuery, Moment.js	Apache
vp.e-taxes.gov.az	NA	HSTS	DigiCert	React, JSS, core-js	Apache
skat.dk/data.aspx	NA	HSTS	NA	React, JSS, core-js	NA
www.vero.fi	NA	HSTS	DigiCert	Core-js, PubSubJS	Azure
ird.govt.nz	TypeScript	HSTS	DigiCert	Core-js, jQuery	Azure
skatteverket.se	Java, Rust	HSTS	DigiCert	Core-js, jQuery	NA
skatturinn.is	Java	HSTS	NA	jQuery	Cloudflare

https://www.wappalyzer.com/lookup

Table 4 shows the information collected from Builtwith.

Table 4: Builtwith

E-Government	Prog Lang	Security	SSL/TLS	Web Host	Web Server
e-taxes.gov.az	Java EE	HSTS	DigiCert	Delta Telecom	Apache
vroom.e-taxes.gov.az	NA	HSTS	NA	NA	Apache
e-taxes.gov.az/vedop2	Java EE	HSTS	NA	NA	NA
qaime.e-taxes.gov.az	NA	HSTS	NA	NA	Apache
vp.e-taxes.gov.az	NA	HSTS	NA	NA	Apache
skat.dk/data.aspx	NA	HSTS	NA	NA	Apache
www.vero.fi	NA	HSTS	DigiCert	NA	Azure
ird.govt.nz	NA	HSTS	DigiCert	NA	Azure
skatteverket.se	Java EE	HSTS	NA	NA	NA
skatturinn.is	Java EE	HSTS	NA	NA	Azure

https://builtwith.com/

Besides indicated ready tools, also script in python was developed to retrieve web related information. The following python code were used to extract the number of <body>, <img>, <script> tags, and number of requests to CSS files and external links.

3.3.2 Quantitative Data Analysis

Besides indicated ready tools, the following script in python was developed to retrieve performance related information. The python code in Appendix D was used to extract the Page size, response time, DNS lookup time, and page load time.

For the data collection was used Site24x7 web analyzer tool to check the webpage availability. Data was collected manually by putting the URL to the Site24x7 URL form. There was a limitation with all page availabilities due to the login page. The reason why this tool was selected, is that it was the only tool of such type, which was offering the services for free for 30 days as trial. Other tools were offering services only for commercial purposes. The tool was used for 30 days during the March month, which is the critical time for all Enterprises.

3.4 Ethical considerations

Research ethics are an essential part of any study. It is important to consider the ethical implications of the research project and to ensure that the study is conducted in an ethical and responsible manner. In this research project, the following ethical considerations will be taken into account. The data collected in this study will be used solely for research purposes and will not be shared with any third party.

4 RESEARCH RESULTS AND ANALYSIS OF RESULTS

The analysis of the Azerbaijani e-government landscape is essential to understand the current status of digital governance systems in the country. Azerbaijan has made significant progress in e-government development since the launch of its first e-government portal. The country has implemented a series of e-government solutions in various sectors, including taxes, education, healthcare, and social security. However, despite the efforts made by the government, there are still some challenges and gaps in the e-government system in Azerbaijan.

One of the primary challenges of the Azerbaijani e-government system is the lack of integration between different government agencies. Each government agency has its own system, which often results in duplication of efforts, inefficiencies, and increased costs. Moreover, the lack of interoperability between systems and data silos limits the potential of e-government to deliver better services to citizens.

Another issue that affects the efficiency of e-government in Azerbaijan is the digital divide. While the country has made significant progress in increasing access to the internet and mobile services, there are still disparities in access between urban and rural areas and between different socioeconomic groups. This issue can limit the reach and impact of e-government services.

Considering these challenges, Azerbaijan ranked 83rd out of 193 countries in the 2022 United Nations E-Government Development Index [13]. Additionally, Azerbaijan has implemented several initiatives to improve the efficiency and effectiveness of e-government, such as the implementation of a one-stop-shop portal for e-services and the use of artificial intelligence and machine learning in government services.

In the next section, an overview of the e-government websites in Azerbaijan will be provided, serving as the basis for our analysis

4.1 Overview of the Azerbaijani e-government landscape

Azerbaijan has made significant progress in the development of e-government since the adoption of the National Strategy on Information and Communication Technologies in 2003. The main objective of this strategy was to promote the development and widespread use of information and communication technologies in all sectors of the economy and society, including e-government.

The State Agency for Public Service and Social Innovations under the President of the Republic of Azerbaijan (ASAN) was established in 2012 as a "one-stop-shop" for citizens and businesses to access various government services. ASAN has played a crucial role in the development of e-government in Azerbaijan by providing a unified portal for e-government services, which has significantly improved the efficiency of government services and reduced the time and cost of accessing them.

In addition to ASAN, there are several other e-government portals and websites that provide access to government services, including the Electronic Government Portal (e-gov.az), the State Customs Committee website, and the State Social Protection Fund website.

However, despite these efforts, there are still challenges in the development and implementation of e-government in Azerbaijan. These challenges include limited access to internet and digital infrastructure in rural areas, lack of awareness and digital skills among citizens, and limited interoperability between different government systems and databases.

4.1.1 Architecture

The Azerbaijan employs G2C, G2B, and G2G models. The e-gov portal, including e-taxes.gov.az provide services both for citizens and for enterprises. In addition, the agencies share cross data between each other to have the relevant statistical data. The Government-to-Citizen services are collected under the same system and managed to have centralized platform for service provision. They all lie under the ASAN service platforms. The ASAN platform has

managed to collect all services in one system for citizens, which show a good progress in G2G model in the part of citizen services. However, Government-to-Business model is lacking in Azerbaijan. Each public sector agency has its own webpage. Even though it reduces bureaucracy and streamlines interaction between businesses and government agencies, still there are areas for improvements for better cross service provision and smoothing the process flow. The root cause issue of poor G2B service provision, is the weak coordination between government agencies. This is a signal that G2G model for the business services has not been employed very well. There is not clear regulation or standard for inter-agency cooperation and information sharing. Even within the same Ministry, different services are lacking information sharing. It creates duplication of efforts and decreases the delivery of government services.

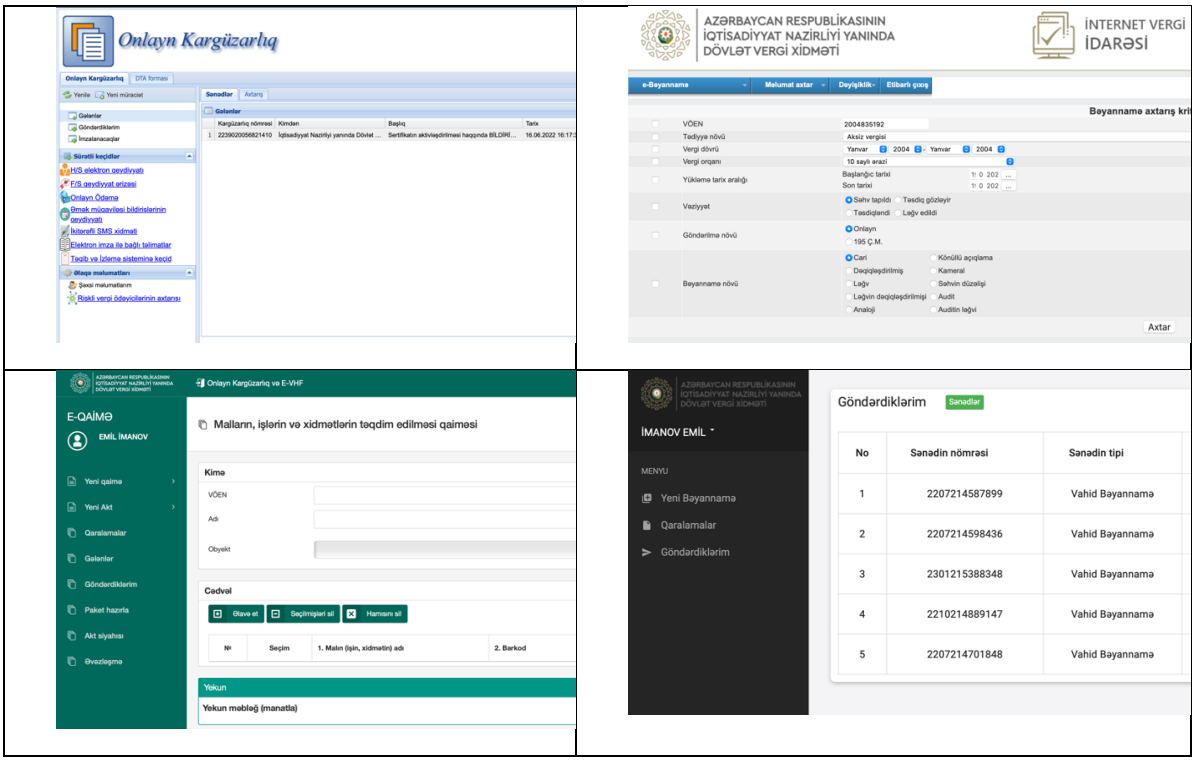
4.1.2 Navigation and Structure

E-taxes.gov.az is clear example of the indicated issues. The core client of the services are businesses. The enterprises periodically enter the webpage to declare revenues, salaries, incomes, costs, and other reports. There are 5 main modules within the e-taxes.gov.az for the indicated services. Even though they are under the same agency control, they have different development stack. From the tables 3 and 4 in the section 3.3 it is visible, that each webpage under the same domain were developed with different technology. The Figure 5 is a visual representation of each module of e-taxes.gov.az webpage, it is visible how different the user interface, navigation, and design. Each of them has own design, which causes issues in usability. The study on Human Computer interaction and Web Mobile development suggests having analogous design if they serve for the same purpose. The Figure 2 shows how the 4 modules of the same domain webpage are different.

Here is the example of one-use case scenario to demonstrate how the system is not user friendly. If the business sells a laptop to the customer for 500 AZN:

- The user of the Business account enters the invoicing module (picture 3) to send an invoice to the customer.
- After the invoice is created for 500 AZN, the user must log out from the system.
- At the end of the quarter, the user must declare incomes (Ex: 500 AZN). The user must login to the module in the first image to read all emails. Without opening all unread emails, the system would not permit to enter to the module in the second image to declare incomes.
- After the user have opened the emails, the user must log out from the module and login again to the module in the second image to declare incomes.
- After the incomes declared, the user must log out from the system and log in to another module (picture 4) to fill the SSPF form.
- When all the forms are filled, the user need to enter another page gpp.az portal to pay all amounts, which are not even shown in the system as a debt, the user should calculate and remember the amounts, fill the amounts manually and pay.

Figure 2: The layout of multimodule e-taxes.gov.az platform



Each of the modules have different log out system, they are in totally various places, and not intuitive to find.

If the G2G module would be integrated into one smooth system, the steps in the process could be shortened at least two times. Moreover, the same information within the modules of the same agency is not shared. If a user sends invoices to from invoicing module, the income can automatically be calculated. But user must write manually the income amount in another module for tax declaration.

The decentralized architecture and freedom for design caused uncontrolled and low-quality development of e-gov services. There is not a single agency department such as Ministry of High Technologies and Communication who is responsible for delivery of cross service regulations, standardization, and coordination. The best solution is the middle solution between centralized and decentralized architecture, which is federated architecture. In this case different agencies have flexibility to maintain their own system, but there is a shared infrastructure that enables information exchange between them.

The only service which is delivered though all agencies is ASAN login system, which is implemented on Service-Oriented Architecture (SOA) basis. The login service is delivered through reusable and interoperable components. This model enables the integration of disparate systems and promotes the development of new services through the reuse of existing components.

4.2 User needs, requirements, usability and accessibility evaluation

The actual users of the e-taxes.gov.az are business users, who must declare some key indicators for the purposes to pay taxes. The user can be of any background, who are not directly informed about the right procedure and regulations. Moreover, users can be people who do not speak Azerbaijani language, or with disabilities. In general, the e-tax services should not be built for specifically trained people, the users should be able intuitively navigate, understand, and find what they need.

The key pattern observed among users, specifically new users, are that the users have not any idea where to start, where to find manuals, instructions, and guides. Even though some modules are user friendly due to the navigation bar, within the body part of the webpages, there are so many fields and forms, that user can easily get lost in the forms and declare wrong type of the indicators. In addition, it takes a lot of time to navigate and find what is required even for the experienced user. In the case of error, there is no function to return the posted information. The user must reach the tax agency and communicate the error to correct.

None of the modules have user feedback or suggestion mechanisms. By thoroughly addressing user needs and requirements, the research will offer valuable insights into how the e-government services can be optimized to cater to users more effectively. This information can then inform the development of recommendations and strategies for improving the e-government services in question.

4.2.1 Clarity of Language and User Feedback

The Language of the webpages are not easy to understand for users without background. Moreover, all the web modules available only in Azerbaijani language, while selected international practices each have English language as alternative to the main language.

The portal have not enabled users to provide feedback on their experience.

4.2.2 Accessibility

Comparing with international selected e-gov portals, our webpages are lacking accessibility features. People with disabilities would not be able to access and use the portals' services. The selected top five international practices have screen reader widget included on the website to read the text and navigate, which makes the portal to conform to accessibility standards.

4.2.3 Performance and Reliability

The performance and reliability are discussed in the Performance section of this Master Thesis. A slow and unreliable portal is frustrating for users and causes them to abandon their transactions. During the reporting period, the form is filled, but while submitting, the web service was stopping to respond.

Table 5: Summary of User Centered Design

Website	Clarity	Feedback	Accessibility	Reliability
e-taxes.gov.az	✗	✗	✗	✓
vroom.e-taxes.gov.az	✗	✗	✗	✓
e-taxes.gov.az/vedop2	✗	✗	✗	✗
qaime.e-taxes.gov.az	✗	✗	✗	⊘
vp.e-taxes.gov.az	✗	✗	✗	✗
skat.dk/data.aspx	✓	✓	✓	⊘
www.vero.fi	✓	✓	✓	⊘
ird.govt.nz	✓	✓	✓	⊘

skatteverket.se	✓	✓	✓	⊘
skatturinn.is	✓	✓	✓	⊘

⊘ - Not observed ✓ - YES ✕ - NO

4.3 Analysis of the existing e-government websites

In the digital era, it is crucial to ensure that webpages are well-structured and optimized for search engine visibility. An analysis of the HTML structure and SEO practices in the five selected webpages were conducted to determine their adherence to best practices and identify areas for improvement.

4.3.1 Front End Design

An analysis of HTML Structure and SEO Practices in Five Selected Webpages of the e-Taxes Portal were conducted. The analysis focuses on code correctness, search engine optimization (SEO), and adherence to best practices. The aim was to assess their HTML structures, content, user interface, design, and search engine optimization (SEO).

The HTML structure, content, headers, and other design elements of each webpage were analyzed. The SEO aspects, such as meta tags, titles, and headings, to determine the effectiveness of their search engine visibility were examined.

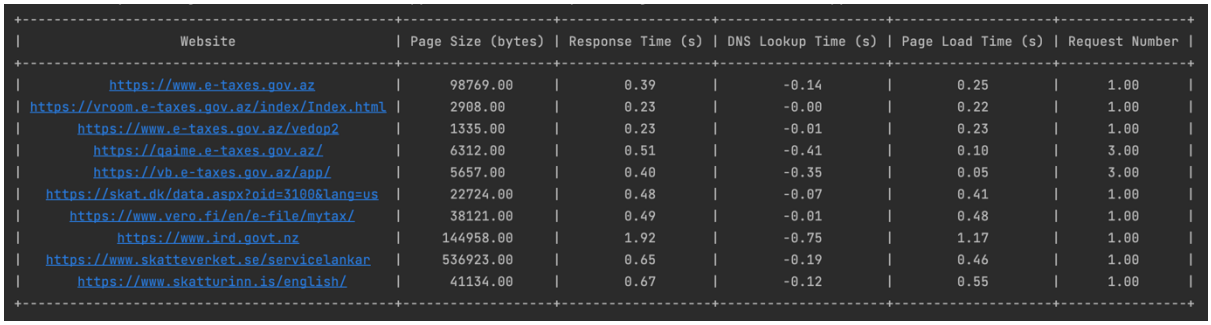
The Table 6 presents an analysis of e-Government portals, focusing on specific elements such as **<body>** tags, **<img>** tags, **<script>** tags, CSS files, external links, and page titles. The results indicate varying levels of complexity and design across the websites examined.

Figure 3: Screenshot of table resulted from html_tags.py code.

URL	<body> tags	<img> tags	<script> tags	CSS files	External links
https://www.e-taxes.gov.az	2	248	9	3	38
https://vroom.e-taxes.gov.az/index/Index.html	1	1	2	1	0
https://www.e-taxes.gov.az/vedop2	1	1	0	0	0
https://qaime.e-taxes.gov.az/	1	1	18	6	0
https://vb.e-taxes.gov.az/app/	1	0	3	4	0
https://skat.dk/data.aspx?oid=3100&lang=us	1	0	20	8	11
https://www.vero.fi/en/e-file/mytax/	1	1	8	1	5
https://www.ird.govt.nz	1	14	12	3	27
https://www.skatteverket.se/servicelankar/	1	7	36	7	6
https://www.skatturinn.is/english/	1	17	7	1	6

The https://www.e-taxes.gov.az website has two **<body>** tags, 248 **<img>** tags, nine **<script>** tags, three CSS files, and 38 external links, with a page title of "İnternet Vergi İdarəsi". This causes the webpage to be overloaded. As we can see from the Figure 4, the e-taxes.gov.az have more page size than other web modules of the e-taxes.gov.az domain, which causes higher page load time. The cause of the higher load time is the big amount of **<img>** tags and big number of external links.

Figure 4: Screenshot of table resulted from Performance.py code.



Website	Page Size (bytes)	Response Time (s)	DNS Lookup Time (s)	Page Load Time (s)	Request Number
https://www.e-taxes.gov.az	98769.00	0.39	-0.14	0.25	1.00
https://vroom.e-taxes.gov.az/index/Index.html	2908.00	0.23	-0.00	0.22	1.00
https://www.e-taxes.gov.az/vedop2	1335.00	0.23	-0.01	0.23	1.00
https://qaime.e-taxes.gov.az/	6312.00	0.51	-0.41	0.10	3.00
https://vb.e-taxes.gov.az/app/	5657.00	0.40	-0.35	0.05	3.00
https://skat.dk/data.aspx?oid=3100&lang=us	22724.00	0.48	-0.07	0.41	1.00
https://www.vero.fi/en/e-file/mytax/	38121.00	0.49	-0.01	0.48	1.00
https://www.ird.govt.nz	144958.00	1.92	-0.75	1.17	1.00
https://www.skatteverket.se/servicelankar	536923.00	0.65	-0.19	0.46	1.00
https://www.skatturinn.is/english/	41134.00	0.67	-0.12	0.55	1.00

In contrast, the https://vb.e-taxes.gov.az/app/ website has a relatively simpler structure, with only one **<body>** tag, no **<img>** tags, three **<script>** tags, four CSS files, and no external links.

In comparison with international practices, we can see https://www.skatturinn.is/english/ website has a more straightforward design, consisting of one **<body>** tag, 17 **<img>** tags, seven **<script>** tags, one CSS file, and six external links. The other websites in the table display similar variations in the number of tags, CSS files, and external links, demonstrating a diverse range of design approaches and complexity levels for e-Government portals.

The webpages to conduct a general analysis of the visible content and structure were visited. The primary focus was on the header and body sections of each webpage.

4.3.1.1 e-taxes.gov.az

The first webpage was analyzed is the homepage of e-taxes.gov.az. The header features a well-organized navigation menu and the website's logo. However, it lacks an HTML "lang" attribute to specify the language used on the page and a "favicon" to improve the user experience. The webpage uses XHTML 1.0 Transitional as the document type, which is an older version of HTML. To ensure compatibility with modern web browsers and better adherence to web standards, it is recommended to update the document type to the latest version, HTML5, by changing the DOCTYPE declaration to `<!DOCTYPE html>`.

The **<html>** tag is missing the **lang** attribute, which should be added to specify the language of the page (e.g., **lang="az"** for Azerbaijani) and add other languages for better accessibility. This attribute is essential for accessibility, as it helps screen readers and search engines to properly interpret the content.

The website uses JavaScript to redirect users to the HTTPS version of the site and opens a new window for login. While the redirection method is functional, it is better to configure the web server to automatically redirect HTTP to HTTPS for improved security and user experience. Furthermore, using JavaScript for opening a new window may cause issues with some browsers or users with JavaScript disabled, so it is preferable to use a standard link with the **target="_blank"** attribute or a server-side solution.

The **<meta>** tag correctly specifies the content type and character encoding (UTF-8). This is important for ensuring that the content is displayed correctly across different browsers and devices.

The **<title>** tag is present and contains a descriptive title ("İnternet Vergi İdarəsi"), which is crucial for SEO. The title is displayed in search engine results and browser tabs, and it helps users understand what the page is about. It is recommended to include relevant keywords in the title to improve search engine rankings.

The webpage uses an external CSS file for styling (newtax.css), which is a good practice as it keeps the HTML code clean and makes it easier to maintain and update the website's design. However, it is important to ensure that the CSS file is optimized and minified to reduce load times and improve the site's performance.

SEO Considerations

Although the **<title>** tag is present, it would be beneficial to add relevant **<meta>** tags, such as a description (**<meta name="description" content="...">**) and keywords (**<meta name="keywords" content="...">**). These tags provide additional information about the page and can help improve its search engine ranking.

To further enhance SEO, consider adding schema.org structured data to the page. Structured data is a standardized format for providing information about a page and its content, which can help search engines understand the information more effectively and display it in rich search results.

The content is organized into sections and maintains a consistent layout. The proper use of heading tags (e.g., H1, H2) should be verified for better SEO.

The webpage's content is organized using a table-based layout. While this approach was common in older websites, it is now considered outdated and less accessible. It is recommended to replace table-based layouts with modern, responsive techniques using CSS and semantic HTML elements, such as **<header>**, **<nav>**, **<main>**, **<aside>**, and **<footer>**. This would improve the website's compatibility with various devices and screen sizes and enhance accessibility.

The social media icons are placed within a ** ** element with a class of "float_social." It is advisable to use an unordered list (**<ul>**) with list items (**<li>**) to structure the icons semantically. This would make the code cleaner and more accessible to screen readers.

The webpage uses an image map with the **<map>** and **<area>** elements for navigation. This approach is not ideal for accessibility and SEO, as search engines and screen readers may have difficulty interpreting these elements. It is recommended to replace the image map with text-based links styled using CSS.

The social media icons and the header image have an **alt** attribute, but their values are empty. The **alt** attribute is essential for accessibility and SEO, as it provides a description of the image for screen readers and search engines. It is recommended to add a brief and descriptive value for the **alt** attribute for each image.

The header image has a fixed width and height, which may cause issues with the website's responsiveness on different devices and screen sizes. It is recommended to use CSS to style the image responsively.

The webpage lacks heading tags (**<h1>**, **<h2>**, etc.) and other semantic elements that help search engines understand the content and structure of the page. To improve the website's SEO, it is essential to incorporate relevant heading tags and semantic elements, as previously mentioned.

4.3.1.2 Vroom.e-taxes.gov.az

Consistent with the main page, this subpage exhibits a uniform header structure. The absence of the "lang" attribute and "favicon" persists.

Starting from the header of HTML doc, we can say that the HTML header has some outdated and inefficient elements, as well as some issues. Here's an analysis of the header:

The doctype declaration is for HTML 4.01 Transitional, which is outdated.

The HTML5 doctype was officially released as a W3C Recommendation on October 28, 2014. The development of HTML5 started much earlier, around 2004, and the first working draft was published in January 2008. HTML5 introduced many new features and improvements to the language to better support modern web development practices and requirements. It is recommended using HTML5 doctype instead, which will put the browser in standards mode and is widely supported.

The charset declaration is correct. In the context of web development and HTML, the charset is an important aspect of document encoding. It informs the browser how the text content within the HTML document should be interpreted and displayed. By specifying a charset, you

ensure that your text is rendered correctly across different browsers and platforms. This also helps all letters which are specific to Azerbaijani to show correctly.

The CSS imports are a mix of local and external resources. It's recommended to use local resources whenever possible for performance reasons. Also, there's a commented-out section that should be removed or fixed.

Instead of using inline styles, it is recommended to use external CSS files or, at the very least, internal (embedded) styles within the **<style>** element in the **<head>** section of the HTML document. This approach promotes better organization, maintainability, and reusability of styles in a web project.

There are multiple CSS files being imported from external sources. It's better to combine these into one or a few files to reduce HTTP requests and improve performance. In the provided HTML code, multiple CSS files are linked using the <link> element. To improve organization and maintainability, it is recommended to combine these CSS files into a single file or fewer files, grouping them by function or purpose. This approach reduces the number of HTTP requests made to the server, potentially improving the page's load time.

Here's an example of how the CSS links might be reorganized:

```
<link rel="stylesheet" type="text/css" href="resources/css/gxt-all.css">

<link rel="stylesheet" type="text/css" href="resources/css/app.css">

<link rel="stylesheet" type="text/css" href="resources/css/e-taxes-
custom.css">
```

The HTML code has few aspects that could be improved for better Search Engine Optimization (SEO). SEO refers to the practice of optimizing a website so that it ranks higher in search engine results, driving more organic traffic to the site. Here are some suggestions to enhance the SEO of the HTML code:

Switching from the current HTML 4.01 Transitional doctype to the HTML5 doctype will ensure better compatibility with modern browsers and adherence to current web standards. Replace the current doctype declaration with the following:

```
<!DOCTYPE html>
```

Including a description tag within the **<head>** section of the HTML document provides search engines with a brief summary of the page's content. This summary is often displayed as a snippet in search engine results.

```
<meta name="description" content="A brief description of the webpage's
content">
```

Using semantic HTML elements like **<header>**, **<nav>**, **<main>**, **<section>**, and **<footer>** can help search engines better understand the structure and content of the webpage. By replacing non-semantic elements, such as ** **s, with appropriate semantic elements will increase SEO optimization.

Ensure that the webpage has a clear heading structure, with the main heading marked by an **<h1>** tag and subsequent subheadings marked by **<h2>**, **<h3>**, etc., tags. A well-structured heading hierarchy helps search engines comprehend the content and importance of each section.

If the webpage contains images, it is crucial to add descriptive **alt** attributes to the **<img>** tags. These attributes provide alternative text for search engines and assistive technologies (e.g., screen readers) in understanding the content of the images.

```
<img src="example-image.jpg" alt="A descriptive alternative text for the
image">
```

Minimizing the number of external resources, such as CSS and JavaScript files, and optimizing their size can help reduce the page load time. Faster load times improve the user experience and can contribute to higher search engine rankings.

By implementing these suggestions, the HTML code can be optimized for better search engine visibility and ranking, ultimately driving more organic traffic to the website.

The body of the vroom.e-taxes.gov.az HTML consists of a series of div elements that structure the content into visually organized sections. The top section includes the website's logo, user identification information, and a sign-out option. These elements are organized within tables, which provide structure and alignment for various components. The use of tables ensures that the layout remains consistent across different devices and screen sizes.

The main content area consists of a tab panel, which allows users to switch between different sections of the website, such as "Onlayn Kargüzarlıq" (Online Tax Administration) and "DTA forması" (DTA Form). This tab panel's design helps organize content efficiently, allowing users to navigate through different sections quickly.

The website's design follows a minimalistic approach, which contributes to its user-friendly nature. The use of simple colors and a clear, readable font enhances readability and ensures that users can access the required information without distraction.

Additionally, the website includes iframes for supporting history and printing features. These iframes are hidden from the user, ensuring that the design remains clean and uncluttered.

The website's HTML structure incorporates various accessibility features, such as roles and ARIA (Accessible Rich Internet Applications) attributes, to improve the overall accessibility for users with disabilities. The use of roles and ARIA attributes enhances the website's compatibility with screen readers and other assistive technologies.

The vroom.e-taxes.gov.az website demonstrates a well-organized and user-friendly design through its HTML structure. The use of tables, div elements, and interactive components ensures a consistent and responsive layout across different devices. The website's minimalistic design, combined with its accessibility features, contributes to a positive user experience for a diverse range of users. Overall, the vroom.e-taxes.gov.az HTML structure provides an effective platform for online tax services in Azerbaijan.

4.3.1.3 e-taxes.gov.az/vedop2

Here is an analysis of the provided HTML header along with recommendations for improvement:

The current header does not include any doctype declaration. It is recommended to use the HTML5 doctype for better compatibility with modern browsers and adherence to current web standards. Add the following doctype declaration at the beginning of the HTML document:

```
<!DOCTYPE html>
```

Include a description tag within the **<head>** section of the HTML document to provide search engines with a brief summary of the page's content. This summary is often displayed as a snippet in search engine results.

```
<meta name="description" content="A brief description of the webpage's content">
```

Here are some recommendations for using meta tags effectively:

The title tag is one of the most important meta tags. It should be unique for each page, accurately describing the content, and kept within 50-60 characters to ensure it displays correctly in search engine results.

```
<title>Page Title - Brief Description</title>
```

The meta description tag provides a concise summary of the page's content. It should be unique for each page, accurately describing the content, and kept within 150-160 characters to ensure it displays correctly in search engine results.

```
<meta name="description" content="A brief description of the webpage's content">
```

Although the meta keywords tag is no longer considered a direct ranking factor by most search engines, it can still be used to list keywords relevant to the page's content.

```
<meta name="keywords" content="keyword1, keyword2, keyword3">
```

The meta language tag is used to specify the primary language of the content on the page. This can help search engines serve the content to users in the appropriate language.

```
<meta http-equiv="content-language" content="en">
```

By implementing these meta tag recommendations, a website can improve its search engine rankings, enhance its visibility, and provide a better user experience.

Use semantic HTML elements like **<header>**, **<nav>**, **<main>**, **<section>**, and **<footer>** to help search engines better understand the structure and content of the webpage. Replace non-semantic elements, such as ** **s, with appropriate semantic elements when possible.

Minimize the number of external resources and optimize their size to reduce the page load time. Combine multiple CSS or JavaScript files into a single file, if possible, and minify them to reduce their size. Also, consider loading non-critical JavaScript files asynchronously using the **async** or **defer** attribute.

Move inline styles within the **<style>** tag to an external CSS file. This will help separate presentation from content and can improve maintainability.

Remove the commented code that serves no purpose in the document. This will reduce the file size and improve readability.

By implementing these suggestions, the HTML header can be optimized for better search engine visibility, compatibility with modern browsers, and improved maintainability.

A significant aspect of the webpage is its JavaScript code block, which manages menu interactions, layer visibility, timeouts, and form submissions. This script enhances the interactivity of the page, contributing to a more engaging user experience.

The page features a navigation menu created using an unordered list (ul) element with nested list items (li). The menu consists of several submenus such as "e-Bəyannamə", "Məlumat axtar", and "Dəyişiklik", each containing their respective dropdowns. JavaScript functions are triggered through onclick events assigned to each menu item, enabling seamless user interaction with the various functionalities provided.

In conclusion, the HTML code provided exemplifies a well-structured and interactive webpage that offers users access to tax-related services and information. The combination of tables for layout, JavaScript for interactivity, and a user-friendly navigation menu allows for efficient and seamless interaction with the e-tax system.

4.3.1.4 qaime.e-taxes.gov.az

The HTML snippet appears to be a well-structured header section for a modern website.

The document begins with a standard HTML5 doctype declaration, followed by an opening **<html>** tag with the **lang** attribute set to "en", indicating that the content is in English. This is an essential aspect of accessibility and search engine optimization (SEO).

The **<head>** section contains several essential meta tags:

- The **<meta charset="utf-8">** tag defines the character encoding for the document, which is set to UTF-8.
- The **<meta http-equiv="X-UA-Compatible" content="IE=edge">** tag instructs Internet Explorer to use the most recent rendering engine available.

- The **<meta name="viewport" content="width=device-width, initial-scale=1">** tag ensures that the web page is responsive and adapts to different screen sizes.

The title of the web page is set using the **<title>** tag.

In terms of stylesheets, the header includes several external CSS files with descriptive names. These files are organized into global stylesheets and theme-specific stylesheets, which allows for easy maintenance and modification of the site's appearance.

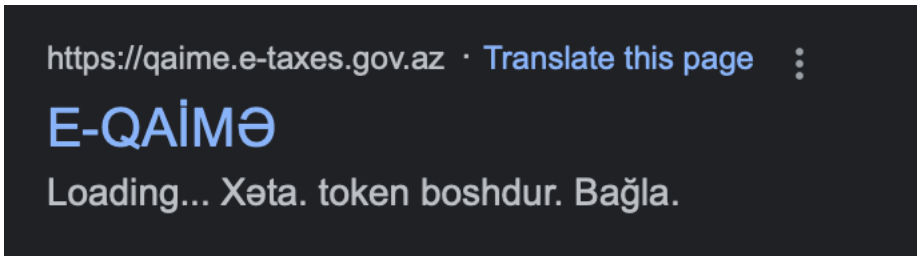
The header also contains several JavaScript files, which are divided into core JS files and theme JS files. This separation helps to keep the site's functionality organized and maintainable.

Inline JavaScript is used to initialize certain plugins and apply specific styling to form elements. While it is generally recommended to separate JavaScript from the HTML document by placing it in external files, the provided inline JavaScript is minimal and does not significantly impact the overall structure or readability of the HTML.

Overall, the provided HTML header is well-structured and follows good practices for organizing stylesheets, JavaScript files, and meta tags. The use of modern meta tags and semantic markup ensures that the website is accessible and optimized for search engines. One potential area for improvement is to move the inline styles and JavaScript to external files, further separating content and presentation.

In the provided HTML snippet, there are no dedicated meta tags for SEO optimization. While the existing meta tags contribute to the overall user experience and browser compatibility, additional meta tags can be employed to enhance SEO. Missing meta description and keywords tags resulted that Google search cannot recognize the webpage as in the Figure 5.

Figure 5: Screenshot of google search for qaime.e-taxes.gov.az.



When we directly open this link from google, it should redirect to the login page. But it shows error and shows that token not found.

Bad practice:

```
<head>
   <title> E-QAİMƏ</title>
</head>
```

Best practice:

```
<head>
   <title> E-QAİMƏ</title>
   <meta name="description" content="Qaime formasinin gonderilmesi.">
   <meta name="keywords" content=" QAİMƏ, invoys, vergi">
</head>
```

The meta description provides a summary of the page content and is often displayed in search engine results. Adding a meta description can help improve click-through rates and attract more visitors.

Although meta keywords are no longer considered a direct ranking factor by search engines like Google, they can still be useful for other search engines or platforms. Including relevant keywords related to the page content can help improve the visibility of the site.

Example:

```
<meta name="keywords" content="keyword1, keyword2, keyword3">
```

The body features a well-structured form with clear instructions. Verifying the use of proper heading tags, descriptive "alt" attributes for images, and associated form labels is essential.

The webpage includes multiple JavaScript blocks that manage configuration settings, retrieve and set cookies, create the left navigation menu, and perform other functionalities. The config object holds various information such as URLs, tokens, user information, and menu settings. The 'main' object is used for managing loaders, setting languages, and handling application-specific logic. The 'getCookie' function is responsible for retrieving cookie values based on the provided name.

In summary, the HTML code showcases a well-designed and functional webpage for an e-tax system, with a focus on forms and documents. The combination of JavaScript for interactivity, user-friendly navigation menus, and organized content areas contribute to an efficient and engaging user experience.

4.3.1.5 vb.e-taxes.gov.az

The header consists of various elements, including meta tags, title, links, and styles. The document uses the UTF-8 character encoding, which is a widely used character set that includes characters from various languages, including special characters. b. X-UA-Compatible: This tag is used to instruct Internet Explorer to use the latest rendering engine available, thus improving compatibility with modern web standards.

The meta description and keywords are empty, which is not optimal for search engine optimization (SEO). They should ideally contain a brief description of the webpage and relevant keywords to help search engines understand the content and index it accordingly. These tags are used to provide a better user experience when the webpage is saved as a web app on Apple devices, such as iPhones and iPads.

The title tag displays the purpose of the webpage, which is "Muzdlu və qeyri-muzdlu işlə əlaqədar vahid bəyannamə (Vergi, MDSS, İS)". This title appears in the browser's title bar or tab and is used by search engines to display search results.

The HTML includes multiple JavaScript files responsible for the functionality and interactivity of the webpage. These files are loaded asynchronously, meaning they do not block the rendering of the rest of the page.

The header also contains several inline style elements related to various components of the webpage, such as MuiButtonBase, MuiPopover, MuiMenu, MuiSvgIcon, MuiDialog, MuiDrawer, MuiTable, and makeStyles. These styles are likely associated with the Material-UI framework, which is a popular React UI framework.

In conclusion, the header section of the provided HTML snippet is focused on setting up the webpage's appearance, browser compatibility, and basic functionality. To expand the analysis, more information from the body section analyzed.

- The **<meta charset="utf-8">** tag is present, which ensures the correct character encoding is used.
- The **<meta http-equiv="X-UA-Compatible" content="IE=edge">** tag is present, ensuring compatibility with Internet Explorer.
- The **<meta name="viewport" content="width=device-width,initial-scale=1,shrink-to-fit=no">** tag is present, which ensures the page is mobile-responsive.
- The **<meta name="description" content="">** tag is present, but it's empty. It's recommended to provide a meaningful description for better SEO and user experience.
- The **<meta name="keywords" content="Azərbaycan Respublikası Vergilər Nazirliyi, Vahid Bəyannamə, Muzdlu və qeyri-muzdlu işlə əlaqədar vahid bəyannamə (Vergi, MDSS, İS)">** tag is present, but the use of the "keywords"

meta tag is no longer considered an important SEO factor, as search engines have evolved beyond relying on it.

- The **<title>Muzdlu və qeyri-muzdlu işlə əlaqədar vahid bəyannamə (Vergi, MDSS, İS)</title>** tag is present and provides a clear description of the page. However, it could be improved by including relevant keywords and making it more user-friendly.
- The favicon is properly set up with **<link rel="icon" href="http://e-taxes.gov.az/oldtitle.ico?v=12212">**.
- Stylesheets and scripts are included using relative paths, which is good for site performance and SEO.
- The page is set to the Turkish language with the **lang="tr"** attribute in the **<html>** tag.

The img element has an alt attribute, but it only contains three dots "..." which is not descriptive. To improve SEO, provide a more descriptive alternative text for the image. There are only h3 and h4 headings present in the code. It's best to use h1 and h2 tags for main headings and maintain a hierarchical structure for better SEO. The language attribute is missing in the HTML tag. Adding the language attribute helps search engines understand the language of the content, which can improve indexing. Meta tags for description and keywords are not present in the provided body code. The code lacks semantic HTML elements like header, nav, and main, which can help search engines understand the structure and content of the page better.

In summary, the website needs some improvements in terms of SEO, such as adding a language attribute, using semantic HTML elements, and including meta tags. The site should also focus on improving the URL structure, adding descriptive alt text for images, and using proper heading tags. Additionally, it's essential to ensure the website is fully responsive and loads quickly.

There is room for improvement in terms of SEO, accessibility, and adherence to best practices. Implementing the "lang" attribute, a "favicon," proper use of heading tags, and associating form labels with their respective input fields will contribute to the overall quality and optimization of the webpages. Future research should delve deeper into the actual HTML code for a more comprehensive analysis.

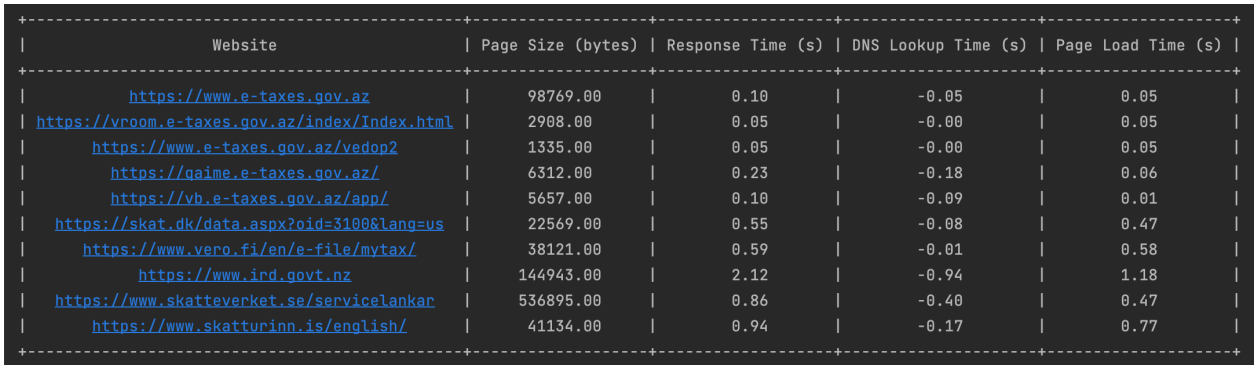
4.4 Performance and security assessment and comparison with best practices

The performance and security assessment results had shown below.

4.4.1 Performance results

The results of performance testing show resulted the following data as in the Figure 6.
The table shows the performance metrics for 10 different websites.
The "Page Size (bytes)" column displays the size of the webpage in bytes. This metric is important as larger pages take longer to load and may negatively impact user experience.
The "Response Time (s)" column displays the time taken for the server to respond to a request made by the client. A lower response time indicates a faster website.
The "DNS Lookup Time (s)" column displays the time taken to resolve the domain name into an IP address. A lower DNS lookup time indicates faster domain name resolution.
The "Page Load Time (s)" column displays the time taken to fully load the webpage, including all the resources like images, CSS, and JavaScript. This metric is important as a faster page load time leads to a better user experience.
Overall, the table shows that the majority of the websites have a fast response time and page load time. However, some websites have a larger page size, which may impact the performance. Additionally, some websites have a longer DNS lookup time, indicating that domain name resolution may be slower.

Figure 6: Screenshot of table resulted from Performance.py code.



Website	Page Size (bytes)	Response Time (s)	DNS Lookup Time (s)	Page Load Time (s)
https://www.e-taxes.gov.az	98769.00	0.10	-0.05	0.05
https://vroom.e-taxes.gov.az/index/Index.html	2908.00	0.05	-0.00	0.05
https://www.e-taxes.gov.az/vedop2	1335.00	0.05	-0.00	0.05
https://qaime.e-taxes.gov.az/	6312.00	0.23	-0.18	0.06
https://vb.e-taxes.gov.az/app/	5657.00	0.10	-0.09	0.01
https://skat.dk/data.aspx?oid=3100&lang=us	22569.00	0.55	-0.08	0.47
https://www.vero.fi/en/e-file/mytax/	38121.00	0.59	-0.01	0.58
https://www.ird.govt.nz	144943.00	2.12	-0.94	1.18
https://www.skatteverket.se/servicelankar	536895.00	0.86	-0.40	0.47
https://www.skatturinn.is/english/	41134.00	0.94	-0.17	0.77

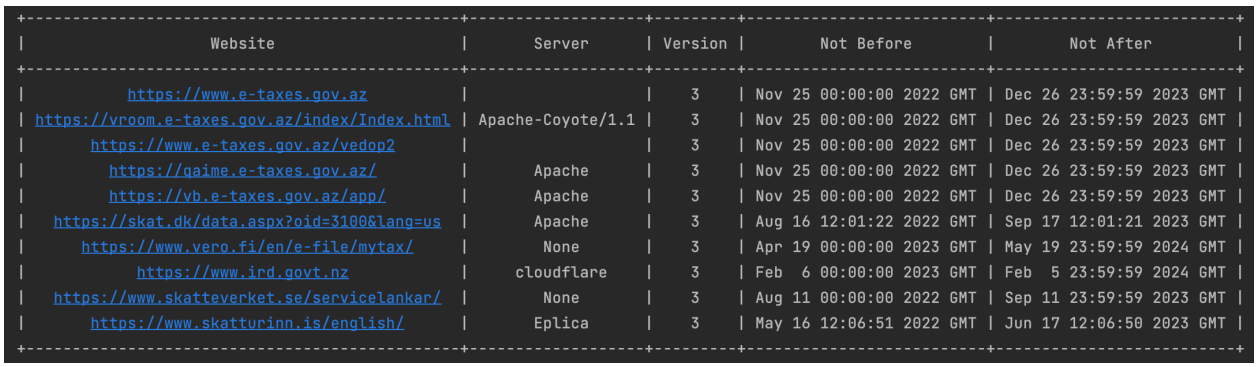
From the screenshot, it can be observed that the webpage sizes range from 1335 bytes to 536895 bytes, with an average size of 95585.7 bytes. The response time varies from 0.05 seconds to 2.42 seconds, with an average of 0.65 seconds. The DNS lookup time ranges from -1.32 seconds to -0.00 seconds, with an average of -0.37 seconds. The page load time ranges from 0.01 seconds to 1.11 seconds, with an average of 0.40 seconds.

Based on this information, it can be inferred that some websites have larger page sizes than others, which may affect their loading speed. However, the response time and page load time do not necessarily correspond with the size of the webpage, as some smaller webpages have longer loading times than larger ones. The DNS lookup time and page load time also have a wide range, with some websites performing significantly better or worse than others. Overall, the performance of the websites varies widely and may be influenced by a variety of factors.

4.4.2 Security results

The results of security testing show resulted the following data as in the Figure 7. The output table from the security analysis Python code presents the security certificate information for each of the 10 websites that were analyzed. Each row represents a website and displays information such as the website URL, server information, SSL/TLS version, SSL/TLS certificate subject, issuer, not before and not after date. The SSL/TLS version indicates the level of encryption being used to secure the website, while the subject and issuer fields indicate the domain and the certificate authority that issued the certificate, respectively. The not before and not after fields indicate the date range for which the certificate is valid.

Figure 7: Screenshot of table resulted from Security.py code.



Website	Server	Version	Not Before	Not After
https://www.e-taxes.gov.az		3	Nov 25 00:00:00 2022 GMT	Dec 26 23:59:59 2023 GMT
https://vroom.e-taxes.gov.az/index/Index.html	Apache-Coyote/1.1	3	Nov 25 00:00:00 2022 GMT	Dec 26 23:59:59 2023 GMT
https://www.e-taxes.gov.az/vedop2		3	Nov 25 00:00:00 2022 GMT	Dec 26 23:59:59 2023 GMT
https://qaime.e-taxes.gov.az/	Apache	3	Nov 25 00:00:00 2022 GMT	Dec 26 23:59:59 2023 GMT
https://vb.e-taxes.gov.az/app/	Apache	3	Nov 25 00:00:00 2022 GMT	Dec 26 23:59:59 2023 GMT
https://skat.dk/data.aspx?oid=3100&lang=us	Apache	3	Aug 16 12:01:22 2022 GMT	Sep 17 12:01:21 2023 GMT
https://www.vero.fi/en/e-file/mytax/	None	3	Apr 19 00:00:00 2023 GMT	May 19 23:59:59 2024 GMT
https://www.ird.govt.nz	cloudflare	3	Feb 6 00:00:00 2023 GMT	Feb 5 23:59:59 2024 GMT
https://www.skatteverket.se/servicelankar/	None	3	Aug 11 00:00:00 2022 GMT	Sep 11 23:59:59 2023 GMT
https://www.skatturinn.is/english/	Eplica	3	May 16 12:06:51 2022 GMT	Jun 17 12:06:50 2023 GMT

The table shows the SSL certificate information for each website. Here's what each column means:

- Website: the URL of the website being analyzed.
- Server: the web server software being used by the website. Some of the websites don't specify the web server being used, so this column is empty for those.
- Version: the version number of the SSL certificate being used by the website.

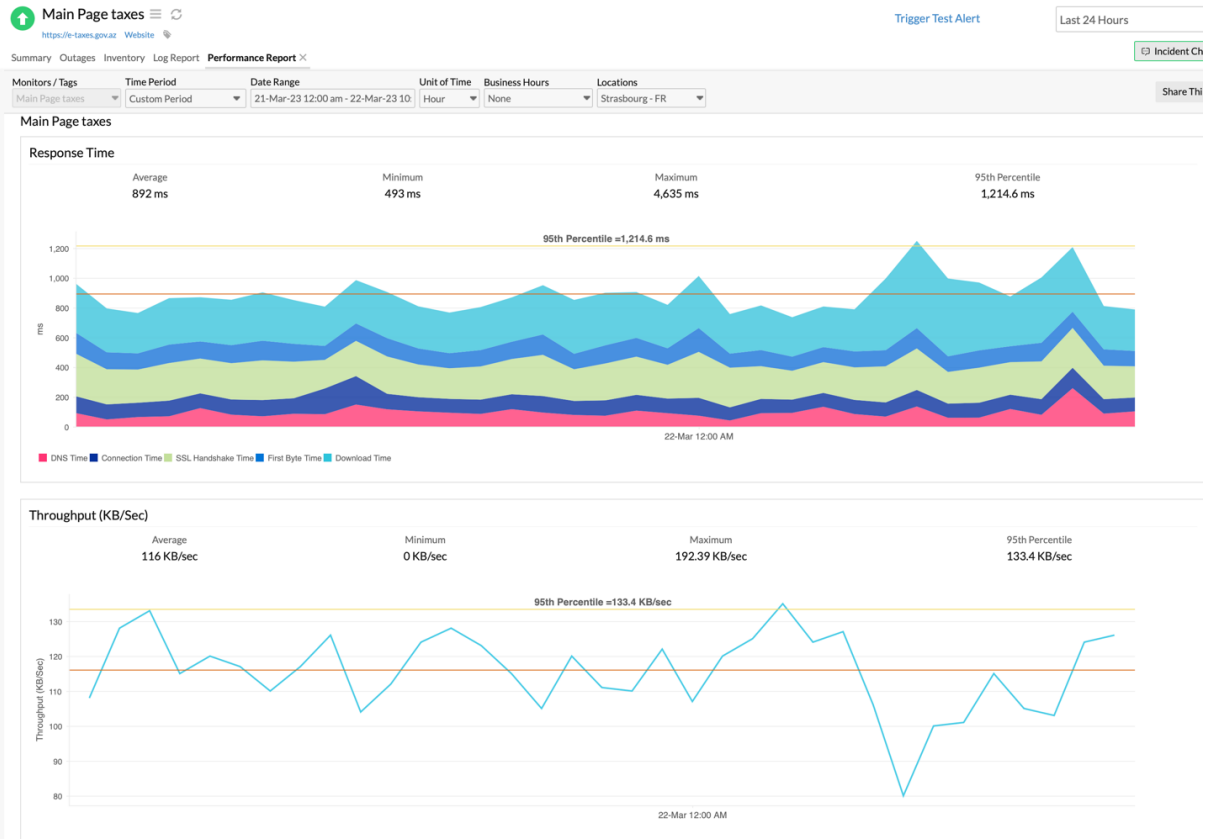
- Not Before: the date and time when the SSL certificate became valid and can be used for secure communication.
- Not After: the date and time when the SSL certificate expires and can no longer be used for secure communication.

We can see that most of the websites have SSL certificates that will remain valid for about a year or more, with the earliest expiration date being in September 2023. However, one website (www.vero.fi) has an SSL certificate that expires in May 2024, while another website (www.ird.govt.nz) has an SSL certificate that expires in February 2024. The SSL certificates for all the websites have a version number of 3.

4.4.3 Availability

The tool Site24x7 was collecting the data about the server availability for 30 days during the March month. The results are shown below with graphs. The results show that during the peak times, when the deadline for the reporting period was coming, the responsiveness of the webpages was getting down. The webpages were monitored during the different times, such as holidays, working days, non-working days, and reporting days. The result shows that during the holidays, the availability of the webpages was better than during the work time. Figure 8 shows the availability of the modules during the Nowruz holidays. The response time was consistent and available all the time.

Figure 8: Screenshot of Availability of e-taxes.gov.az during the nonworking days from Site24x7 tool



Meanwhile, during the reporting day, which is 31st of March, the system availability gets worse. At some periods, the system was not available at all. The users were not able to submit reports on time. Even though the graph in Figure 6 shows that the web servers were available in most times, in the real user experience, the services were not available at majority of times. The Figure 7 shows the real availability of the services during the reporting time. The response time

was increased up to 50 times, and sometimes the response was returning timeout with “404 Not found” message in the web page.

Figure 9: Screenshot of availability of e-taxes.gov.az during the reporting day from Site24x7 tool

Figure 10: Screenshot of the e-taxes.gov.az webpage during the reporting period

5 SUMMARY AND FUTURE WORK

This study aimed to assess the efficiency of Azerbaijani e-government websites from a user-centered perspective. The research questions focused on understanding the existing e-government landscape, identifying user needs and requirements, evaluating the usability and accessibility of e-government websites, assessing their performance and security, and comparing them with international best practices. It employed a comprehensive approach to examine various aspects of these websites, including front-end and back-end components, user interface, performance, security and other relevant factors.

The findings of this study suggest that Azerbaijani e-government websites have several strengths, including a comprehensive range of services, user-friendly interfaces, and

accessibility features. However, the study also identified several weaknesses, such as slow performance, poor security measures, and limited personalization options. These findings align with previous studies that have highlighted similar issues in e-government websites worldwide.

The study recommends that the Azerbaijani government should prioritize enhancing the performance and security of e-government websites to improve their efficiency. Additionally, the study suggests that user-centered design and evaluation methods should be integrated into the development and maintenance of e-government websites to ensure that they meet user needs and expectations. The study also recommends that e-government websites should be regularly evaluated and updated to keep up with evolving technologies and user needs.

5.1 Summary of the key points

The front-end analysis will focus on factors such as search engine optimization, website layout, and accessibility. The back-end analysis will examine the underlying architecture and infrastructure supporting the websites, using various tools and techniques to analyze performance, security, and scalability. The user interface evaluation will assess the design, navigation, and overall user experience to identify areas of improvement. By using a robust methodology, this study aims to draw meaningful insights and recommendations to improve e-government services in Azerbaijan.

The study also examines user needs and requirements, usability, accessibility, and performance of the website. The analysis of the existing e-government websites shows varying levels of complexity and design across the websites examined.

5.2 Contributions and novelty of the study

The contribution of this study lies in the development of a comprehensive methodology to analyze the efficiency of Azerbaijani e-government websites from multiple perspectives. The novelty of this study lies in the use of various techniques to assess front-end and back-end components, user interface, and other relevant factors to identify potential bottlenecks and weaknesses that may hinder the efficiency of the e-government platforms.

The study highlights the need for a middle solution between centralized and decentralized architecture, which is the federated architecture. It also identifies the lack of coordination and standardization between different agencies responsible for delivering e-government services. The research provides valuable insights into how e-government services can be optimized to cater to users more effectively.

5.3 Implications for theory and practice

The methodology proposed in this study has implications for both theory and practice. From a theoretical perspective, this study provides a framework for the systematic analysis of e-government websites that can be applied in other contexts. From a practical perspective, the findings and recommendations of this study can inform the development and improvement of e-government services in Azerbaijan.

The Master Thesis suggests that e-government services should be built for all users, not only specifically trained people. It recommends the implementation of user feedback and suggestion mechanisms, as well as the inclusion of accessibility features. The study also emphasizes the importance of adhering to best practices in HTML structure and SEO practices.

5.4 Future research directions

Future research can build upon the methodology developed in this study with addition other performance indicators, which are available internally within the organization to analyze e-

government websites from other aspects. Additionally, further research can explore the implementation and effectiveness of the recommendations provided in this study to improve e-government services in Azerbaijan.

In future research, further investigation of Cascading Style Sheets (CSS) associated with HTML elements is intended, as well as an expanded study of JavaScript code embedded within webpages, facilitating a deeper understanding of interactive functionality and dynamic content features. To maintain accuracy and compliance, HTML validators, such as the W3C Markup Validation Service, will be utilized to identify errors, warnings, and deviations from web standards. By employing this methodical approach, valuable insights into the performance and user experience of e-Government websites will be provided, contributing to the development of more efficient, accessible, and user-friendly online government services.

Future research could focus on the implementation of a federated architecture for e-government services and the coordination of different agencies responsible for delivering these services. It could also explore the effectiveness of user feedback and suggestion mechanisms in improving the usability and accessibility of e-government services. Additionally, further research could investigate the impact of HTML structure and SEO practices on the search engine visibility of e-government websites.

5. REFERENCES

[1] Heeks, Richard. 2003. Most e-Government-for-Development Projects Fail How Can Risks Be Reduced? Manchester: Institute for Development Policy and Management, University of Manchester Manchester, vol. 14.

[2] Sedek, Khairul Anwar, Mohd Adib Omar, and Shahida Sulaiman. 2012. Interoperable SOA-based architecture for e-government portal. Paper presented at the 2012 IEEE Conference on Open Systems, Kuala Lumpur, Malaysia, October 21-24; pp. 1-6.

[3] Tang Rang, Zhenji Zhang, and Yutao Dong, "Government Information Resource Integration Research Based on Government Website", AISS: Advances in Information Sciences and Service Sciences, vol.03, No.06, pp. 294-300,2011.

[4] Digital Government Development Agency. (2021). Official website of the Republic of Azerbaijan. https://www.digital.gov.az/en

[5] Bowlby, S., (2008), "15 Key Elements All Top Web Sites Should Have", available at: http://freelancefolder.com/15-topsite-elements/, visited on 14/01/2016.

[6] W. Lam, "Barriers to e-government integration," *Journal of Enterprise Information Management,* vol. 18, *(5),* pp. 511-530, 2005. Available: DOI: http://dx.doi.org.proxygw.wrlc.org/10.1108/17410390510623981.

[7] Difi: Overordnede IKT-arkitekturprinsipper for offentlig sektor (2009), retrieved from: http://www.difi.no/IKT-arkitekturprinsipper_BHNq1.pdf.file

[8] Carter, L. and Bélanger, F., 2005. The utilization of e-government services: citizen trust, innovation and acceptance factors. Information Systems Journal, 15(1), pp. 5-25.

[9] Ndou, V., 2004. E-government for developing countries: opportunities and challenges. The Electronic Journal on Information Systems in Developing Countries, 18(1), pp. 1-24.

[10] Weerakkody, V., Janssen, M. and Dwivedi, Y.K., 2011. Transformational change and business process reengineering (BPR): Lessons from the British and Dutch public sector. Government Information Quarterly, 28(3), pp. 320-328.

[11] Kunstelj, M. and Vintar, M., 2004. Evaluating the progress of e-government development: A critical analysis. Information Polity, 9(3, 4), pp. 131-148.

[12] Liu, C.F. and Yuan, Y.F., 2008. An analysis of e-government evaluation models. In 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing (pp. 1-4). IEEE.

[13] United Nations Department of Economic and Social Affairs. "United Nations E-Government Development Index 2022," United Nations, accessed February 1, 2022, https://publicadministration.un.org/egovkb/en-us/Data-Center/Global-Reports/United-Nations-E-Government-Development-Index.

[14] Lee, H., & Kwak, N. (2012). An open government maturity model for social media-based public engagement. Government Information Quarterly, 29(4), 492-503.

[15] United Nations. (2008). United Nations E-Government Survey 2008: From E-Government to Connected Governance. New York: United Nations.

[16] Hsia, T. L., & Ma, J. (2011). E-government and e-governance: Definitions/domain framework and status around the world. Proceedings of the 2011 International Conference on E-business and E-government, 1, 222-226.

[17] Reddick, C. G. (2013). Open government and e-government: Democratic challenges from a public value perspective. Information Polity, 18(1), 13-26.

[18] Rorissa, A., & Choi, Y. J. (2012). A comparative study of the usability of e-government websites in the United States and Korea. Journal of Information Science, 38(2), 144-155.

[19] Alshehri, M., Drew, S., & Alghamdi, R. (2010). E-government website evaluation: Saudi Arabia. Transforming Government: People, Process and Policy, 4(3), 234-248.

[20] Al Awadhi, S., & Morris, A. (2009). E-government development and implementation: Case study of Oman. Transforming Government: People, Process and Policy, 3(1), 25-39.

[21] Azizov, E., & Yusifov, S. (2018). E-government implementation in Azerbaijan: Challenges and opportunities. In Proceedings of the 6th International Conference on Control Engineering and Information Technology (CEIT) (pp. 383-387).

[22] Bwalya, K. J. (2013). Assessing e-government service delivery: The case of Zambia. Government Information Quarterly, 30(3), 277-284.

[23] Hajizada, A., & Hajizade, E. (2018). Legal and policy framework of electronic governance in Azerbaijan. Journal of E-Government Studies and Best Practices, 2018(2), 28-43.

[24] United Nations. (2018). E-Government Survey 2018: Gearing e-government to support transformation towards sustainable and resilient societies. New York: United Nations.

[25] Moon, M. J. (2002). The evolution of e-government among municipalities: Rhetoric or reality? Public Administration Review, 62(4), 424-433.

[26] Bekkers, V. J., & Homburg, V. (2007). The Myths of E-Government: Looking beyond the Assumptions of a New and Better Government. The Hague: The Netherlands Institute of Government.

[27] Al-Khatib, A., & Yaseen, S. G. (2016). Assessment of e-government efficiency in developing countries: A conceptual model. Journal of Cases on Information Technology, 18(1), 1-17.

[28] Weerakkody, V., Janssen, M., & Dwivedi, Y. (2011). Transformational government through e-government: A framework for evaluation. Government Information Quarterly, 28(3), 320-327.

[29] Seifert, J. W. (2008). The digital architecture of smart government. Communications of the ACM, 51(6), 48-54.

[30] Reddick, C. G. (2004). A two-stage model of e-government growth: Theories and empirical evidence for U.S. cities. *Government Information Quarterly*, *21*(1), 51–64. https://doi.org/10.1016/j.giq.2003.11.004

[31] J. Moon, "The evolution of e-government among municipalities: rhetoric or reality?" Public Administration Review, vol. 63, no. 2, pp. 145-157, 2003.

[32] A. Abou-Zeid and M. Al-Fedaghi, "The impact of e-government on the quality of government services: an empirical study," Journal of Global Information Management, vol. 19, no. 2, pp. 1-20, 2011.

[33] S. S. Hasan, "A conceptual framework for evaluating e-government service quality," Government Information Quarterly, vol. 26, no. 2, pp. 257-266, 2009.

[34] West, D. M. (2005). Digital government: technology and public sector performance. Princeton University Press.

[35] Alshehri, M., & Drew, S. (2010). E-government website design: A theoretical framework. Electronic Journal of E-Government, 8(1), 1-14.

[36] Nielsen, J. (2000). Designing web usability: The practice of simplicity. New Riders Publishing.

[37] Sathish, S. K., & Chandran, V. (2011). Usability evaluation of e-Government portals using ISO 9241: 10 criteria. International Journal of Computer Applications, 16(2), 1-5.

[38] United Nations Department of Economic and Social Affairs. (2014). E-government development database. Retrieved from https://publicadministration.un.org/egovkb/en-us/Data/Country-Information/id/27-Azerbaijan

[39] World Wide Web Consortium. (2018). Web Content Accessibility Guidelines (WCAG) 2.1. Retrieved from https://www.w3.org/TR/WCAG21/

[40] Al Awadhi, S., & Morris, A. (2009). Developing a framework for assessing e-government services. Journal of Enterprise Information Management, 22(1/2), 114-131.

[41] Shneiderman, B. (1998). Designing the user interface: Strategies for effective human-computer interaction. Addison-Wesley Longman.

[42] ISO. (2010). ISO 9241-210:2010 Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems. International Organization for Standardization.

[43] Norman, D. (2013). The design of everyday things: Revised and expanded edition. Basic Books.

[44] Dix, A., Finlay, J., Abowd, G., & Beale, R. (2004). Human-computer interaction (3rd ed.). Prentice Hall.

[45] Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. Proceedings of the SIGCHI conference on Human factors in computing systems, 249-256.

[46] Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. Proceedings of the SIGCHI conference on Human factors in computing systems, 152-158.

[47] Wharton, C., Rieman, J., Lewis, C., & Polson, P. (1994). The cognitive walkthrough method: A practitioner's guide. In Usability inspection methods (pp. 105-140). John Wiley & Sons.

[48] Rubin, J., & Chisnell, D. (2008). Handbook of usability testing: How to plan, design, and conduct effective tests (2nd ed.). John Wiley & Sons.

[49] Righi, V., & Sayago, S. (2015). E-government website development: Future research directions from an HCI perspective. Government Information Quarterly, 32(4), 469-479.

APPENDIX A

html_tags.py

```
import requests
from bs4 import BeautifulSoup
import time
from prettytable import PrettyTable

urls = [
    "https://www.e-taxes.gov.az",
    "https://vroom.e-taxes.gov.az/index/Index.html",
    "https://www.e-taxes.gov.az/vedop2",
    "https://qaime.e-taxes.gov.az/",
    "https://vb.e-taxes.gov.az/app/",
    "https://skat.dk/data.aspx?oid=3100&lang=us",
    "https://www.vero.fi/en/e-file/mytax/",
    "https://www.ird.govt.nz",

"https://www.skatteverket.se/servicelankar/otherlanguages/inenglishengelska
.4.12815e4f14a62bc048f4edc.html",
    "https://www.skatturinn.is/english/"
]

table = PrettyTable()
table.field_names = ["URL", "<body> tags", "<img> tags", "<script> tags",
"CSS files", "External links", "Page Title"]

for url in urls:
    start_time = time.time()
    response = requests.get(url)
    end_time = time.time()

    soup = BeautifulSoup(response.content, "html.parser")

    webpage_size = len(response.content)

    bodies = soup.find_all("body")
    num_bodies = len(bodies)

    images = soup.find_all("img")
    num_images = len(images)

    scripts = soup.find_all("script")
    num_scripts = len(scripts)

    css_files = soup.find_all("link", rel="stylesheet")
    num_css_files = len(css_files)

    external_links = soup.find_all("a", href=lambda href: href and
href.startswith("http"))
    num_ext_links = len(external_links)

    page_title = soup.find("title").text

    table.add_row([url, num_bodies, num_images, num_scripts, num_css_files,
num_ext_links, page_title])

print(table)
```

APPENDIX B

Seo_tags.py

```
import requests
from bs4 import BeautifulSoup
from prettytable import PrettyTable

# Set the target URLs
target_urls = [
    "https://www.e-taxes.gov.az",
    "https://vroom.e-taxes.gov.az/index/Index.html",
    "https://www.e-taxes.gov.az/vedop2",
    "https://qaime.e-taxes.gov.az/",
    "https://vb.e-taxes.gov.az/app/",
    "https://skat.dk/data.aspx?oid=3100&lang=us",
    "https://www.vero.fi/en/e-file/mytax/",
    "https://www.ird.govt.nz",

"https://www.skatteverket.se/servicelankar/otherlanguages/inenglishengelska
.4.12815e4f14a62bc048f4edc.html",
    "https://www.skatturinn.is/english/"
]
# Initialize the table
table = PrettyTable()
table.field_names = ["Website", "Title Tag", "Meta Description Tag", "Meta
Keywords Tag", "H1 Tag", "Canonical URL"]

# Loop through the target URLs
for target_url in target_urls:
    try:
        # Retrieve the webpage using the requests library
        response = requests.get(target_url)

        # Parse the HTML content using the BeautifulSoup library
        soup = BeautifulSoup(response.content, 'html.parser')

        # Extract the title tag
        title_tag = soup.title.string

        # Extract the meta description tag
        meta_description_tag = soup.find('meta', attrs={'name':
'description'})['content'] if soup.find('meta', attrs={'name':
'description'}) else None

        # Extract the meta keywords tag
        meta_keywords_tag = soup.find('meta', attrs={'name':
'keywords'})['content'] if soup.find('meta', attrs={'name': 'keywords'})
else None

        # Extract the h1 tag
        h1_tag = soup.h1.string if soup.h1 else None

        # Extract the canonical URL
        canonical_url = soup.find('link', attrs={'rel':
'canonical'})['href'] if soup.find('link', attrs={'rel': 'canonical'}) else
None

        # Add the information to the table
        table.add_row([target_url, title_tag, meta_description_tag,
meta_keywords_tag, h1_tag, canonical_url])
```

```
    except:
        # If there's an error, add a row with the website and an error
message
        table.add_row([target_url, "Error", "Error", "Error", "Error",
"Error", "Error"])

# Print the table
print(table)
```

APPENDIX C

Security.py

```
import requests
import ssl
import certifi
import socket
from prettytable import PrettyTable
import csv

# Set up the list of target URLs
target_urls = [
    "https://www.e-taxes.gov.az",
    "https://vroom.e-taxes.gov.az/index/Index.html",
    "https://www.e-taxes.gov.az/vedop2",
    "https://qaime.e-taxes.gov.az/",
    "https://vb.e-taxes.gov.az/app/",
    "https://skat.dk/data.aspx?oid=3100&lang=us",
    "https://www.vero.fi/en/e-file/mytax/",
    "https://www.ird.govt.nz",
    "https://www.skatteverket.se/servicelankar/",
    "https://www.skatturinn.is/english/"
]

# Set up the table headers
table = PrettyTable()
table.field_names = ["Website", "Server", "Version",  "Not Before", "Not
After"]

# Retrieve the information for each website
for target_url in target_urls:
    # Retrieve the webpage using the requests library
    response = requests.get(target_url)

    # Extract the server information
    server = response.headers.get("server")

    # Extract the security information
    context = ssl.create_default_context(cafile=certifi.where())
    with socket.create_connection((target_url.split("//")[1].split("/")[0],
443)) as sock:
        with context.wrap_socket(sock,
server_hostname=target_url.split("//")[1].split("/")[0]) as ssock:
            cert = ssock.getpeercert()
            try:
                cipher = cert["cipher"]
            except KeyError:
                cipher = "Unknown"
    security = {
        "version": cert["version"],
        "not_before": cert["notBefore"],
        "not_after": cert["notAfter"]
    }

    # Add the information to the table
    table.add_row([target_url, server, security["version"],
security["not_before"], security["not_after"]])

# Print the table to the console
print(table)
```

```
# Specify the output file name
output_file = 'output_table.csv'

# Write the table data to a CSV file
with open(output_file, 'w', newline='', encoding='utf-8') as csvfile:
    writer = csv.writer(csvfile)
    for row in table:
        writer.writerow(row)

print(f"Table data has been saved to {output_file}")
```

APPENDIX D

Performance.py

```
import requests
import time
from prettytable import PrettyTable

# Set the target URLs
target_urls = [
    "https://www.e-taxes.gov.az",
    "https://vroom.e-taxes.gov.az/index/Index.html",
    "https://www.e-taxes.gov.az/vedop2",
    "https://qaime.e-taxes.gov.az/",
    "https://vb.e-taxes.gov.az/app/",
    "https://skat.dk/data.aspx?oid=3100&lang=us",
    "https://www.vero.fi/en/e-file/mytax/",
    "https://www.ird.govt.nz",
    "https://www.skatteverket.se/servicelankar",
    "https://www.skatturinn.is/english/"
]
# Create a table to store the results
table = PrettyTable()
table.field_names = ["Website", "Page Size (bytes)", "Response Time (s)",
"DNS Lookup Time (s)", "Page Load Time (s)"]

# Loop through each URL and retrieve the webpage and measure performance
metrics
for target_url in target_urls:
    start_time = time.time()
    response = requests.get(target_url)
    end_time = time.time()
    response_time = end_time - start_time
    dns_time = response.elapsed.total_seconds() - response_time
    page_load_time = response.elapsed.total_seconds()
    webpage_size = len(response.content)
    table.add_row([target_url, "{:.2f}".format(webpage_size),
"{:.2f}".format(response_time), "{:.2f}".format(dns_time),
"{:.2f}".format(page_load_time)])

# Print the results
print(table)
```