

M.SHAHZAD

DEVELOPMENT OF A WEB-BASED APPLICATION FOR LEARNING C
PROGRAMMING WITH GAMIFICATION

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY



MUHAMMAD HAMZA SHAHZAD

A MASTER OF SCIENCE
THESIS
IN
THE DEPARTMENT OF COMPUTER ENGINEERING

JULY 2023

ATILIM UNIVERSITY 2023

DEVELOPMENT OF A WEB-BASED APPLICATION FOR LEARNING C
PROGRAMMING WITH GAMIFICATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY

BY

MUHAMMAD HAMZA SHAHZAD



IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF COMPUTER ENGINEERING

JULY 2023

Approval of the Graduate School of Natural and Applied Sciences, Atılım University.

Prof. Dr. Ender
KESKİNKILIÇ
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of **Master of Science in Computer Engineering, Atılım University.**

Assoc. Prof. Dr. Gökhan
ŞENGÜL
Head of Department

This is to certify that we have read the thesis DEVELOPMENT OF A WEB-BASED APPLICATION FOR LEARNING C PROGRAMMING WITH GAMIFICATION submitted by MUHAMMAD HAMZA SHAHZAD and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Damla Topallı
Supervisor

Examining Committee Members:

Assoc. Prof. Dr. Cansu Çiğdem Ekin
Computer Eng. Department, Atılım University

Asst. Prof. Dr. Damla Topallı
Computer Eng. Department, Atılım University

Assoc. Prof. Dr. Elif Polat Hopcan
Department of Computer Education and Instructional
Technologies, İstanbul Üniversitesi-Cerrahpaşa

Date: July 3, 2023

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Muhammad Hamza Shahzad

Signature:

ABSTRACT

DEVELOPMENT OF A WEB-BASED APPLICATION FOR LEARNING C PROGRAMMING WITH GAMIFICATION

Shahzad, Muhammad Hamza

MSc., Department of Computer Engineering

Supervisor : Asst. Prof. Dr. Damla Topalli

July 2023, 64 Pages

The development of programming skills, particularly in programming languages like C, is crucial for the software developers and computer science students. Traditional methods of teaching programming often rely on textbooks and lectures, which may not engage students effectively. The main aim of this thesis is to address this problem by developing a web-based application for learning C programming with gamification elements. The web-based application offers an interactive and gamified learning environment including game elements such as a time counter, scores, achievements (i.e. medals, cups) and leaderboard, to engage students and facilitate their understanding of programming concepts. These game elements motivate students and make the learning experience more enjoyable. In order to evaluate the effectiveness of the web-based application, an experimental study is conducted with 122 students from the departments of Computer, Software and Information Systems Engineering, learning C programming, during 2022-2023 Spring Semester. The students were asked to use this application to answer 10 multiple choice questions regarding C Programming language. Then, the study measures overall satisfaction of the students with the application by using a “System Usability Scale” survey to gain insights into students' experiences and suggestions for improvement of the web based application. The results of this study from the usability perspective implies that, 81% of the students agreed that the developed system is easy to use, and 79% of them agreed that

most people would learn to use this system very quickly. 62% of them mentioned that they would like to use this system frequently. Only 20% of the students thinks that the system is unnecessarily complex. The findings of this study contribute to the field of computer science education by exploring the use of gamification in teaching programming languages. The web-based application offers an innovative and interactive approach to learning C programming, making it more engaging and effective for students.

Keywords: C programming, gamification, web-based application, system usability



ÖZ

OYUNLAŞTIRMA İLE C PROGRAMLAMAYI ÖĞRETEN BİR WEB TABANLI UYGULAMANIN GELİŞTİRİLMESİ

Shahzad, Muhammad Hamza

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi : Dr. Öğr. Üyesi Damla Topallı

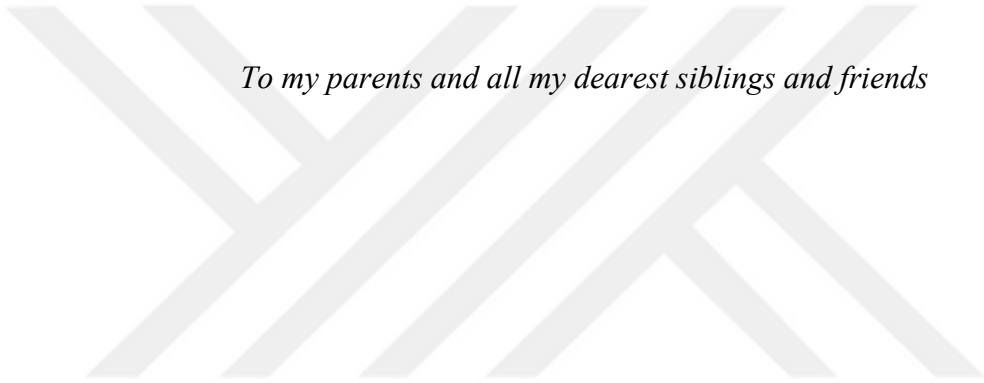
Temmuz 2023, 64 Sayfa

Programlama becerilerinin geliştirilmesi, özellikle C gibi programlama dillerinde, yazılım geliştiricileri ve bilgisayar bilimi öğrencileri için önemlidir. Programlama öğretiminde geleneksel yöntemler genellikle kitaplara ve ders notlarına dayanmaktadır. Bu yöntemler öğrencilerin konuları etkin olarak öğrenmeleri için yeterli olmayabilir. Bu tezin temel amacı, oyunlaştırma unsurlarını içeren bir web tabanlı uygulama geliştirerek bu uygulamanın kullanılabilirliğinin araştırılmasıdır. Geliştirilen web tabanlı uygulama, programlama kavramlarının anlaşılmasını kolaylaştırmak ve öğrencileri öğrenme sürecine etkin olarak dahil etmek için interaktif ve oyunlaştırma ile zenginleştirilmiş bir öğrenme ortamı sunar. Bu oyun unsurları, öğrencileri motive eder ve öğrenme deneyimini daha eğlenceli hale getirir. Bu kapsamda yaptığımız çalışmada oyunlaştırma unsurlarını içeren web tabanlı uygulama geliştirilmiştir. Bu uygulamanın etkinliğini değerlendirmek için, 2022-2023 Bahar Dönemi'nde C programlama dersi alan Bilgisayar, Yazılım ve Bilişim Sistemleri Mühendisliği bölümlerinden 122 öğrenciyle bir deneysel çalışma gerçekleştirilmiştir. Öğrencilerden C Programlama diliyle ilgili 10 çoktan seçmeli soruyu yanıtlamak için bu uygulamayı kullanmaları istenmiştir. Ardından, öğrencilerin uygulama ile genel memnuniyeti bir "Sistem Kullanılabilirlik Ölçeği" anketi kullanılarak ölçülmüş ve öğrencilerin deneyimleri ve geliştirme önerileri hakkında bilgi edinilmiştir. Bu çalışmanın kullanılabilirlik açısından sonuçlarına göre, çalışmaya katılan öğrencilerin %81'i geliştirilen sistemin kolay kullanılabilir olduğunu ve %79'u birçok kişinin bu

sistemi kullanmayı kolaylıkla öğrenebileceklerini belirtmişlerdir. Öğrencilerin %62'si bu sistemi sıklıkla kullanmak isteyeceklerini belirtmişlerdir. Öğrencilerin yalnızca %20'si sistemin gereğinden fazla karmaşık olduğunu düşünmektedir. Bu çalışmanın bulguları, programlama dillerini öğretmede oyunlaştırmanın kullanımını inceleyerek bilgisayar bilimi eğitimine katkıda bulunmaktadır. Dolayısıyla geliştirilen bu web tabanlı uygulamanın, C programlamayı öğrenmede yenilikçi ve etkileşimli bir yaklaşım sunarak, öğrenme deneyimini öğrenciler için daha etkileyici ve etkili hale getirebileceği görülmüştür.

Anahtar Kelimeler: C programlama, oyunlaştırma, web tabanlı uygulama, sistem kullanılabilirliği.





To my parents and all my dearest siblings and friends

ACKNOWLEDGMENTS

I am deeply grateful to my supervisor, Asst. Prof. Dr. Damla Topallı, for her unwavering support during every stage of this thesis. Her commitment, guidance, and mentorship have been invaluable in the preparation of this work, and I truly appreciate her efforts.

I would like to extend my gratitude to the members of the examination committee; Assoc. Prof. Dr. Cansu Çiğdem Ekin and Assoc. Prof. Dr. Elif Polat Hopcan for their valuable time and effort in reviewing this thesis. Their insightful comments and suggestions have greatly contributed to the improvement of this work.

Lastly, I would like to express my deepest appreciation to my parents and family members who have been with me every step of the way throughout this long journey. I am grateful to have their unwavering support, love, and encouragement which have been crucial in my success.

Shahzad, Muhammad Hamza

TABLE OF CONTENTS

ABSTRACT	iii
ÖZ	v
ACKNOWLEDGMENTS.....	viii
TABLE OF CONTENTS.....	ix
LIST OF TABLES	xi
LIST OF FIGURES.....	xii
LIST OF SYMBOLS/ABBREVIATIONS	xiii
CHAPTER 1.....	1
INTRODUCTION.....	1
CHAPTER 2.....	3
BACKGROUND OF THE STUDY	3
2.1. Gamification in Engineering Education	3
2.2. Gamification in Learning Programming Languages (PL).....	5
2.3. Evaluating System Usability of Gamified-Web Applications.....	9
CHAPTER 3.....	12
METHODOLOGY	12
3.1. Research Questions	12
3.2. Participants	12
3.3. Development of the Web Application- System Design	13
3.3.1 Web Application	13
3.3.2 Web Application’s Architectural Pattern.....	14
3.3.3 Types of Web Applications	14
3.3.4 Types of Web Application’s Architectures	15

3.3.5 Our Web Application’s Type and Architecture	16
3.3.6 Technical Stack	17
3.3.7 Technical Terms	17
3.3.8 Software Development Lifecycle Methodology for the System.....	18
3.3.9 Use Case Diagram of the System.....	19
3.4. Research Procedure	19
CHAPTER 4.....	23
RESULTS	23
4.1. Results on the Developed Web-Based System.....	23
4.2. Results on the Usability of the Web-Based System	32
CHAPTER 6.....	38
DISCUSSION	38
CHAPTER 7.....	43
CONCLUSION.....	43
Limitations of Study	44
Future Work.....	44
REFERENCES.....	45
APPENDIX A	49
C PROGRAMMING ACTIVITY: MULTIPLE CHOICE QUESTIONS.....	49
APPENDIX B	52
SAMPLE CONSENT FORM AND SYSTEM USABILITY SCALE (SUS).....	52
APPENDIX C	54
SAMPLE CONSENT FORM AND SYSTEM USABILITY SCALE (SUS) IN TURKISH	54
APPENDIX D	56

LIST OF TABLES

Table 3. 1 Participants of the Study	13
Table 4. 1 Questionnaire Results for Item-3	33
Table 4. 2 Questionnaire Results for Item-5	33
Table 4. 3 Questionnaire Results for Item-7	34
Table 4. 4 Questionnaire Results for Item-9	34
Table 4. 5 Questionnaire Results for Item-2	35
Table 4. 6 Questionnaire Results for Item-4	35
Table 4. 7 Questionnaire Results for Item-6	36
Table 4. 8 Questionnaire Results for Item-8	36
Table 4. 9 Questionnaire Results for Item-10	36
Table 4. 10 Questionnaire Results for Item-1	37

LIST OF FIGURES

Figure 3. 1 API Call Diagram	17
Figure 3. 2 Use Case Diagram of the System	19
Figure 4. 1 View of Entrance Screen	24
Figure 4. 2 View of Notice Component on Entrance Screen.....	24
Figure 4. 3 View of the Input Fields in Application	25
Figure 4. 4 Diagram containing validation method for the keys in database schema	25
Figure 4. 5 Start Button Design.....	25
Figure 4. 6 View of Question Screen.....	26
Figure 4. 7 The Leaderboard Panel Design.....	27
Figure 4. 8 View of Score and Time Component	27
Figure 4. 9 View of Question-Answer Component	28
Figure 4. 10 View of Feedback Component	28
Figure 4. 11 View of the Question Screen with Feedback Element	29
Figure 4. 12 View of the Finish Screen.....	30
Figure 4. 13 View of the Finish Screen with Feedback Elements	30
Figure 4. 14 The Design of the Database Schema	31
Figure 4. 15 Table in the Database.....	32
Figure 4. 16 CSV File from the Database	32

LIST OF SYMBOLS/ABBREVIATIONS

PL	Programming Language
UI	User Interface
SUS	System Usability Scale
CMS	Content Management System
HTML	Hyper Text Markup Language
CSS	Cascade Style Sheets
MVC	Model View Controller
SSR	Server-Side Rendered
PWA	Progressive Web Application
SPA	Single-Page Application
API	Application Programming Interface

CHAPTER 1

INTRODUCTION

In today's digital era, programming skills have become increasingly valuable and sought after in various industries. Aspiring software developers and computer science students often begin their journey by learning foundational programming languages such as C. However, traditional methods of teaching programming, which rely on textbooks and lectures, may not effectively engage students or foster a deep understanding of the subject matter. To address this challenge, there is a growing interest in developing innovative approaches that make programming education more interactive, engaging, and enjoyable.

One such approach is the integration of gamification elements into the learning process. Gamification involves incorporating game-like elements, such as points, levels, achievements, and challenges, into non-game contexts to motivate and engage users. By applying gamification principles to programming education, it is possible to create a more immersive and interactive learning experience that captivates students' attention and enhances their understanding and retention of programming concepts.

In this study, we aim to develop a web-based application for learning C programming with gamification elements. The web-based application will provide an interactive platform where students can engage with programming concepts, practice their coding skills, and receive immediate feedback. The incorporation of gamification elements, such as rewards and challenges, will motivate students to progress through the learning modules and foster a sense of accomplishment.

The development of this web-based application involves designing a user-friendly interface, creating interactive coding exercises, and curating comprehensive learning materials. The application will offer a step-by-step approach, starting from the basics of C programming and gradually advancing to more complex topics. By providing a structured learning path and interactive exercises, the application aims to cater to learners of varying proficiency levels and ensure a comprehensive understanding of C programming.

To evaluate the effectiveness of the web-based application, a pilot study will be conducted with a group of students learning C programming. The study will assess students' engagement levels, learning outcomes, and overall satisfaction with the application. Qualitative feedback will also be collected to gain insights into students' experiences and gather suggestions for further improvement.

The findings of this study have the potential to contribute to the field of computer science education by exploring the integration of gamification in programming language learning. The web-based application aims to provide an innovative and interactive learning environment that enhances students' engagement and understanding of C programming. The insights gained from the pilot study will inform future refinements and improvements to the application, ultimately benefiting students and educators in the field of programming education.

CHAPTER 2

BACKGROUND OF THE STUDY

Gamification offers a promising approach to enhance engineering education and the learning of programming languages like C. With the help of gamification, an immersive and motivating learning environment can be created that promotes active engagement, immediate feedback, and a sense of accomplishment. By incorporating gamified elements into the educational process, educators can effectively motivate and guide students to develop their programming skills. This section summarized the literature review findings of the importance of gamification in engineering education, in learning programming languages and discussed the usability of such systems.

2.1. Gamification in Engineering Education

Gamification has emerged as a powerful and innovative pedagogical approach in engineering education, revolutionizing the way students learn, particularly when it comes to mastering complex programming languages like C. This transformative educational strategy infuses elements of game design and mechanics into the traditional learning process, resulting in a highly engaging, interactive, and enjoyable experience for students [1]. By leveraging the motivational power of gamification, educators have successfully motivated students to actively participate in their learning journey, set ambitious goals, and achieve meaningful objectives, thereby fostering a deeper and more enduring sense of engagement with the subject matter and ultimately leading to remarkable improvements in overall learning outcomes [2][3][4].

Beyond its remarkable success in engineering education, gamification has sparked considerable interest across various domains of education, including the specialized realm of software engineering. A noteworthy study by Alhammad et al. (2018) undertook a systematic mapping of published articles spanning from 2011 to June 2017, conducting a comprehensive exploration of the implementation and impact of gamification in engineering education. Their exhaustive analysis offered a panoramic view of the gamification elements employed and their profound effects on students'

learning experiences. Notably, the research yielded overwhelmingly positive results, affirming that the integration of gamification in engineering education produces more beneficial outcomes than negative or negligible effects [5]

Moreover, taking a more nuanced perspective on the subject, Milosz and Milosz [6] conducted a meticulous examination of a staggering 168 research articles published between 2012 and 2019, with a specific focus on the utilization of gamification in engineering education. Their research sought to discern the diverse applications of gamification elements across various engineering disciplines and shed light on the most frequently adopted elements. The findings were enlightening, with Computer Engineering standing out as the foremost field embracing gamification, with prominent elements such as Leaderboards and Badges/Medals prominently utilized to enhance the learning experience.

Additionally, Nah et al. (2014) contributed significantly to the gamification literature through their comprehensive literature review, encompassing a broader spectrum of education as a whole [7] In their meticulous analysis, they identified and scrutinized an extensive array of game elements utilized within the context of gamified applications, encompassing elements ranging from simple Points systems to sophisticated Leaderboards and Feedback mechanisms. Through a thorough assessment of the impact of these gamification elements on various crucial aspects such as student engagement, participation, motivation, and academic performance, the researchers underscored the immense potential of gamification in transforming the educational landscape. Their pioneering work served as an inspiring call to action for educators and practitioners alike, encouraging them to explore the vast possibilities of gamified applications specifically tailored to educational contexts. As the popularity of gamification continues to surge and the educational landscape evolves, further exploration of its comprehensive impact on various disciplines becomes paramount, facilitating the holistic assessment and optimization of its benefits in fostering effective and dynamic learning environments that empower students to excel and thrive.

Bartel and Hagel [8] propose the creation of gamified and competency-based learning activities in Software Engineering courses, aiming to revolutionize the way students engage with complex concepts and design patterns. To empirically validate their

approach, they meticulously conducted an experimental study with active participation and feedback from a diverse group of students. The results of their study illuminated the immense potential of incorporating gamification design elements into the learning process. It was evident that the strategic use of gamified techniques not only facilitated a deeper understanding of design patterns but also had a remarkable impact on students' motivation to excel in their studies.

Building on Bartel and Hagel's pioneering work, other researchers such as Souza et al.[9] have delved into the realm of gamification in Software Engineering education. Utilizing diverse gamification components, including badges, leaderboards, and interactive games, these scholars sought to create an engaging and dynamic learning environment. Their approach centered on reinforcing positive behaviors, encouraging healthy competition, and instilling a sense of achievement among the students.

To assess the effectiveness of their implementations, Souza et al. conducted comprehensive surveys, gathering valuable feedback from the participants [9]. The overwhelmingly positive responses from the students demonstrated the significant impact of integrating gamification elements into the Software Engineering courses. Notably, the incorporation of badges for completing milestones, leaderboards to track progress, and interactive games for practical learning resonated exceptionally well with the learners, fostering a sense of camaraderie and a competitive spirit.

These studies collectively underscore the immense potential of gamified approaches in revolutionizing Software Engineering education. By tapping into students' intrinsic motivation, curiosity, and desire for meaningful achievement, gamification proves itself to be an effective tool in enhancing the learning experience and empowering students to become proficient and enthusiastic software engineers. As educators continue to explore and refine these innovative pedagogical techniques, the future of Software Engineering education promises to be an exciting and transformative one, with gamification playing a pivotal role in shaping the technologists of tomorrow.

2.2. Gamification in Learning Programming Languages (PL)

Gamification, with its profound impact on engineering education, has proven to be an invaluable tool, particularly in the realm of programming languages like C. By incorporating elements of game design and mechanics, the traditional learning process

is transformed into an immersive, interactive, and enjoyable experience, engaging students on a deeper level [1] The benefits are evident as gamification successfully overcomes the initial complexities and challenges associated with learning programming concepts. By breaking down these intricate topics into smaller, more manageable tasks or levels within a game-like environment, students experience a sense of progression and achievement with each completed level, fostering a powerful sense of confidence and motivation [10]

Khaleel et al. (2019) conducted an extensive and enlightening study [11] that delved into the profound impact of gamification on student motivation in learning programming languages. Implementing the experiment with students at a university, they established three groups, including a control group and two experimental groups exposed to different degrees of gamification elements. The findings unequivocally demonstrated that the incorporation of gamification elements significantly heightened student motivation and engagement when compared to traditional teaching methods. Through gamification, students developed a genuine enthusiasm for learning and a renewed dedication to their academic pursuits.

Building upon these promising findings, Palaniappan et al. (2022) conducted further research [12] that sought to elucidate the inherent advantages of gamification in the context of learning a programming language. Employing gamification elements within an online learning environment, learners' knowledge was assessed through tests designed to evaluate their understanding. The results resonated with previous studies, solidifying the notion that gamification serves as a potent motivational catalyst, particularly for new students with little or no prior experience in programming. Through the stimulating effects of gamification, these students were inspired to embark on their learning journey with zest and determination, overcoming any initial hesitations or apprehensions.

Expanding the scope of gamification's impact on programming education, Khaleel et al. (2020) contributed to the growing body of research [13] by exploring the effects of gamification and student engagement when learning a programming language. Deploying a dedicated website as a platform to teach programming concepts, the research illuminated a significant disparity in the effectiveness and participation of students when compared to traditional teaching methods. The gamified approach once

again emerged as a compelling and impactful medium in encouraging active student involvement, capturing their interest, and fostering long-term commitment to the learning process.

Moreover, gamification serves as an invaluable tool in facilitating immediate feedback and rewards, which are essential components of the learning process. The feedback loop created through gamification empowers students to closely monitor their progress, identify areas for improvement, and receive timely and positive reinforcement for their achievements, fostering a continuous cycle of self-directed learning and improvement. [5]

In addition to its benefits in individual learning, gamification also cultivates a sense of healthy competition and collaboration among students, further enriching the learning environment. The incorporation of elements such as Leaderboards and badges sparks friendly competition, motivating students to push their boundaries and excel, resulting in elevated academic achievement and a sense of accomplishment. Simultaneously, collaborative features like team challenges or group activities foster a spirit of teamwork and cooperation, creating an atmosphere of knowledge sharing and peer-to-peer learning. These collaborative experiences not only contribute to academic growth but also equip students with essential soft skills, such as communication, teamwork, and leadership, that are vital for their future professional success in engineering and beyond. [12]

Sprint and Cook noted that students often lack focus and feel uncomfortable in introductory programming courses, leading to high dropout rates. [14] Sarkar et al. [10] found that students perceive programming as boring and tedious, lacking the element of fun. Even when novice programmers grasp the concepts, they struggle to apply them due to insufficient practice [15] Mastering programming requires considerable effort, time, and practice before students can effectively write programs and comprehend underlying principles.

To address this concern and motivate students to enhance their performance, researchers have explored the gamification theory in the field of learning computer programming. Gamification provides an innovative approach that leverages game design elements and mechanics to engage learners in a dynamic and interactive learning environment. Ibanez et al. [5] conducted a study that highlighted gamification

as a mechanism to keep students motivated and engaged during programming learning. By integrating elements such as points, levels, badges, and challenges into programming exercises, students are incentivized to progress and achieve milestones, transforming the learning experience into an exciting journey. Similarly, game-based assignments were found to increase motivation and programming skills among students [16]. By turning programming tasks into game-like scenarios and providing instant feedback and rewards for successful completion, students are motivated to persist and improve their skills.

Moreover, Sprint and Cook [14] conducted a survey comparing traditional teaching methods to a gamified approach in computer programming courses, observing a higher throughput of the learning process with the use of gamification. These findings collectively demonstrate the potential of gamification to revitalize introductory programming education, fostering a sense of accomplishment and engagement among students, thereby reducing drop-out rates and promoting a deeper understanding of programming concepts.

Several studies have highlighted the positive impact of gamification on students' engagement in programming courses. Harrington and Chaudhry [17] Latulipe et al. [18], and Marín et al. [1] conducted independent investigations and consistently found that integrating gamification elements in programming education leads to increased student interest and participation. The introduction of gamified elements, such as virtual badges for completing specific coding challenges, leaderboards to track progress and compare achievements with peers, and interactive coding games that simulate real-world scenarios, proved to be highly effective in motivating students to actively participate in the learning process. By transforming traditionally dry and abstract programming concepts into interactive and rewarding experiences, gamification bridges the gap between theory and practice, promoting a deeper understanding of programming principles.

Kumar and Khurana [19] also emphasized the potential of gamification in helping programming students learn and gain a complete understanding of concepts without solely relying on textbooks. Immediate feedback and group learning were also identified as important drivers of motivation. This collective body of research showcases the significant positive impact of gamification on programming education,

encouraging educators to explore and integrate these innovative pedagogical techniques to create engaging and effective learning experiences for their students. By leveraging gamification's inherent potential to transform the learning journey into an enjoyable and immersive experience, educators can create a positive and empowering environment that nurtures students' passion for programming, instilling a lifelong love for learning and continuous improvement. With the ever-increasing demand for skilled programmers in various industries, integrating gamification in programming education stands as a promising approach to cultivate a new generation of enthusiastic and competent software developers.

2.3. Evaluating System Usability of Gamified-Web Applications

In addition to the benefits of gamification in engineering education and learning programming languages, the usability of gamified systems plays a significant role in their effectiveness. [7] Usability refers to the ease of use, user-friendliness, and overall user experience of a system or application. In terms of usability, some of the critical aspects are as listed below:

- *Interface Design*: The user interface should be intuitive, visually appealing, and well-organized. Clear navigation menus, understandable icons, and logical layout contribute to a positive user experience. The system should provide easy access to various features, and levels.
- *Interactivity and Responsiveness*: The gamified system should respond promptly to user actions, providing immediate feedback on the correctness of programming solutions or progress in the game. It should encourage active participation and engagement by allowing students to interact with the learning materials and game elements smoothly.
- *Progress Tracking and Assessment*: The system should offer mechanisms to track and visualize student progress, allowing them to monitor their performance, achievements, and areas for improvement. Clear indicators of progress, such as levels, points, or badges, can motivate students and provide a sense of accomplishment.

- *Adaptability and Customization:* A well-designed gamified learning system should allow customization options, such as adjusting the difficulty level, language or choosing specific programming topics.
- *Technical Reliability:* The system should be stable, reliable, and accessible across different devices and platforms. Students should be able to access the gamified learning environment without technical issues or limitations. [20]

Khaleel et al. in 2018, delved into the benefits of gamification in the realm of learning. Through the implementation of a web-based application and a comprehensive survey, the researchers sought to gain valuable insights into the usability and effectiveness of gamification as a learning tool [21].

The web-based application used in the study served as the platform for incorporating gamified elements into the learning process. By integrating game-like mechanics, such as points, badges, and leaderboards, the researchers aimed to create an engaging and motivating learning environment. To evaluate the system's effectiveness, the survey was carefully designed, comprising various items that assessed different aspects, including ease of use, interface design, feedback mechanisms, and students' intention to continue using the gamified approach.

The study's findings unveiled a promising connection between the usability of the gamified learning platform and students' motivation to learn. Participants demonstrated a heightened interest and enthusiasm for the subject matter when exposed to the gamified approach, which positively impacted their learning motivation. The gamification elements appeared to tap into intrinsic motivators, fostering a sense of achievement, competition, and enjoyment, all of which contributed to a more productive and rewarding learning experience. Consequently, these positive outcomes shed light on the potential of gamification to transform traditional learning methodologies and offer a fresh perspective on optimizing educational practices. [21]



CHAPTER 3

METHODOLOGY

This chapter explains the design and methodology of the system which was used in this study to investigate and achieve the proposed thesis objectives and through the subchapters of this chapter, we would understand the methodology of the experimental study to get feedbacks from the students who used the web-based system when taking the C Programming course and the main research questions of the thesis are discussed in this perspective.

3.1. Research Questions

The following research questions are aimed to be answered in this study:

RQ1. How can a web-based application be developed by using gamification elements for learning programming languages?

RQ2. In terms of usability of the developed web-based application with the integration of gamification elements, are the students satisfied with the application for learning C language?

RQ3. Do the students prefer to use the system frequently?

3.2. Participants

An experimental study is conducted with 122 students from the departments of Computer, Software, and Information Systems Engineering at Atilim University. These students are registered to the “CMPE113-Computer Programming-I” course in the 2022-2023 Spring semester. Most of the students (104 students, 85%) are first year students in the university, where there are also 16 students (13%) from second year and 2 students (2%) from third year students. There are 38 female students with the

average age of 20.29, and 84 male students with the average age of 20.56. The detailed information about the students according to their departments are given in Table 3.1.

Table 3. 1 Participants of the Study

	Female	Male	TOTAL
Computer Eng.	14	28	42
Information Systems Eng.	15	24	39
Software Eng.	9	32	41
TOTAL	38	84	122

3.3. Development of the Web Application- System Design

To ensure high usability, gamified learning systems can create a seamless and engaging learning experience for students. A user-friendly interface, clear presentation of content, interactivity, progress tracking, adaptability, and technical reliability contribute to the overall usability of such systems, enhancing their effectiveness in facilitating learning and skill development in programming languages like C.

3.3.1 Web Application

A web application also phrased as web app is used as a software with the help of tools and programming languages such as HTML, Javascript and CSS. As the term ‘web’ depicts, these apps can be accessed through a web browser in a desktop computer or other devices such as a mobile phone or a tablet. Also, these applications can perform a range of tasks including streaming media, shopping, and managing financial service. Henceforth, in our case, it can also be used in education sector.

To understand the web apps better we can deduce that, the term web app cannot be taken as a website/webpage because the web app and a website can have different expectations, requirements, and usages for the end users. To clear the blurred line, websites are typically designed to provide information and take simple inputs from the user and on the other hand, a web app is designed to be more dynamic and provides useful interactive content, in turn engaging the users. [22] As we have understood what

web app is, we can discuss how the web application really is, with the help of its architecture; in other words, the components involved and how they are interlinked together.

3.3.2 Web Application's Architectural Pattern

The modern application can have a 3-tier pattern also famous as Model View Controller (MVC) model and can be deduced as layered architecture which involves a data layer, application, and a presentation layer. Let's discuss them in more detail:

1. **Data Layer:** This layer is required to have a database or a like connected to it, which acts as a service to pass all the data used in the app and is responsible for the storage.
2. **Application Layer:** This layer is used for the logic involved in the app, for instance, it can manipulate the user input and executes the business rules on it through a server running continuously to provide the services. Hence, it is a required piece in the app.
3. **Presentation Layer:** This layer acts as a user interface which a user sees when he uses the application. It is responsible to provide navigation in the app as well as the visual elements involved. In turn, the important aspect in the application. [23]

We have discussed our chosen application's architecture in detail. However, before we dive into the types of architectures available to us and compare it to our selected architecture, it is important to mention the types of web application, because the types of web application can have specific requirements when it comes to their architecture.

3.3.3 Types of Web Applications

The web application can range from different types depending upon the requirements of the stakeholders. The web application itself can be bare bone to ultra-complex, representing enterprise-level. However, 3 types of web applications will be useful for our understanding, so we can differentiate easily between them.

1. Server-side rendered (SSR) application: This type of app is sourced directly from web server, because of their faster page loads enhancing the user's experience, these apps are secure and can be scaled easily. However, it is hard to implement and for complex views on presentation layer, these apps can take initially a longer rendering time; or for a page to load so the HTML can be sent directly from the server. [24]
2. Progressive web application (PWA): Progressive web application can provide a user experience which behaves like a native app, rich and immersive for the user and user does not need to install it, based on its nature. However, it is also complex to develop and to maintain. [25]
3. Single-page Application (SPA): This type of application interacts with the user by dynamically painting the view, also known as client-side rendering. Hence a user does not need to wait for the page to load and it is easier to implement so the development is not complex. It can be used to build complex user interfaces which can provide all the great functionalities including, faster response time and rich user experience. [26]

As we have gained understanding in the types of web applications. Let's enlighten ourselves with web application's architecture through their types, so we can also understand the importance of choosing the right architecture for our application.

3.3.4 Types of Web Application's Architectures

Although web applications can be architected in various ways depending upon the variables such as cost and the selected type of web application, however there are three main ways an application can be architected and it should be noted that, hybrid architectures are also available to meet the requirements.

1. Monolithic Architecture: This is the most basic type of architecture for the web applications because it relies on a single code base for all the components and logic, including business, user interface and database access. Hence, they all share the same runtime environment, and it simplifies the development.

However, monolithic architecture is not suitable for a large and complex web app, used for a large number of concurrent users and needs to be scaled. [27]

2. **Microservices Architecture:** Microservice architecture, on the contrary to monolithic, breaks down large, complex application into smaller and manageable components. This helps developers to manage the components better, because they can be deployed, scaled and are testable independently so, developers can make changes easily and different developers with specialized skills can work on their component, in turn improving the whole application. [28]
3. **Serverless Architecture:** When it comes to managing the architecture, it can be time consuming and can require special skills also, the cost can be a factor so a serverless architecture takes care of the application's architecture by automatically scaling the components of the web application. Hence it is efficient because it can scale when required. For example, when a lot of users are accessing the web app at the same time it will increase the number of servers otherwise, if users are low in number, it can scale down. [29]

3.3.5 Our Web Application's Type and Architecture

There are numerous factors to consider when it comes to selecting the correct type of web application and its architecture. The factors including main use, complexity of the app, stakeholder's needs, and expectations. User traffic, experience and skills of developers and the cost are also important factors to consider.

In our case, we selected SPA with the serverless architecture, because the application required a complex, user-friendly interface with faster response time and, our 3-tier layers were managed on cloud-servers due to high number of concurrent users, meeting the requirements. Hence, making our application highly scalable.

Since we have already shed light on the top-level view of our app. Let's dive into technical details and understand the libraries/packages or simply the web technologies we have used in the development of our application.

3.3.6 Technical Stack

We have used Typescript for the development of our application because it is a widely used language which could run our servers as well as, paint our view on presentation layer. Hence, one language was enough for our needs, and we did not require any resource other than the proficiency in the language and in the main packages including React and also the skills for deployment on cloud.

For the UI, we used Material UI to build a beautiful and user-friendly design of our application and connected our presentation layer to the data layer using Strapi's API, running the Node.js server and we utilized Amazon Web Services (AWS) for the deployment of our application.

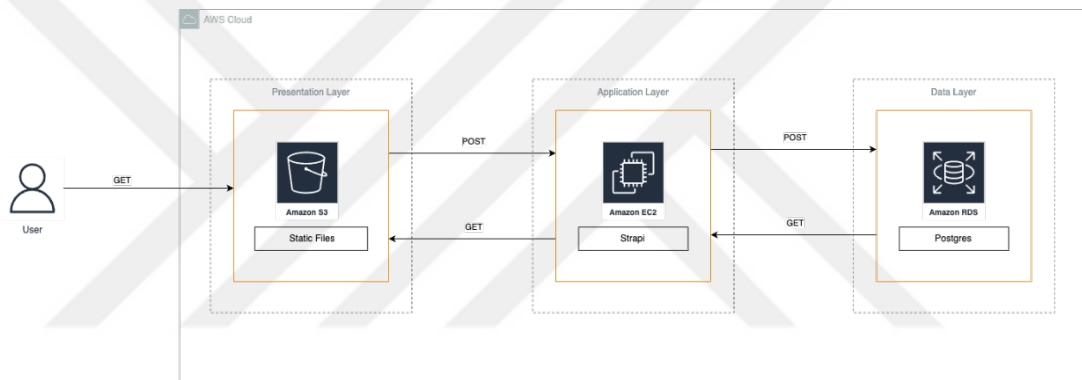


Figure 3. 1 API Call Diagram

3.3.7 Technical Terms

Javascript: is a programming language. [30]

Typescript: is a subset of Javascript, used to build large applications with type-safe system. [31]

React: is a library written in Javascript language to build reusable User Interface (UI) components. [32]

Material UI: is a library of components to build design of the UI that features, the implementation of Google's Material Design system. [33]

Node.js: is an open-source server environment which can run Javascript on server. [34]

Strapi: is an open-source Content Management System (CMS) which enables content to be created, managed, and exposed through its Application Programming Interface (API). [35]

Amazon Web Services (AWS): is a cloud service provider. [36]

3.3.8 Software Development Lifecycle Methodology for the System

There are plenty of models available to alter or develop a software product, consisting of planned activities to meet the expectations of the stakeholders. The following models are used widely by the software industry: Waterfall Model, Spiral Model, V-Model, Big Bang Model, Rapid Prototyping Model, Iterative Model, Incremental Model and Agile Model. [37] We chose Agile model which contains the processes of iterative and incremental models. The detail about the chosen model is provided in more detail below.

Agile Model

Agile model takes a different approach when compared to other tradition model such as waterfall. Due to its flexibility and adaptability, it has become popular with time. It believes in principles also known as Agile Manifesto, which was published in 2001, which clearly outlines the need of a working software to communicate with customer rather than just depending on documentation for the requirements whereas continuous interaction with the customer is important to get proper requirements. Also, motivation and self-organization plays an important role in development as well as quick response to changes in the software. [38]

Agile is based on an adaptive method when the requirements are not clear and detailed planning is not needed. [39] In our case, it was suitable because we did not plan on how the frontend/presentation layer would look like. We released the application in cycle, meaning each feature was built incrementally and deployed consequently, highlighting the main feature of agile model. Hence, it proved to be a success, because it provided great flexibility in development and the required features were delivered in the application.

3.3.9 Use Case Diagram of the System

A use case diagram is a diagram type of a Unified Modeling Language (UML), which can condense the detailed functions of your system's users (referring as actors) and their interactions with the system. To design our application, the use case diagram for a student to use the quiz application is given in Figure 3.2.

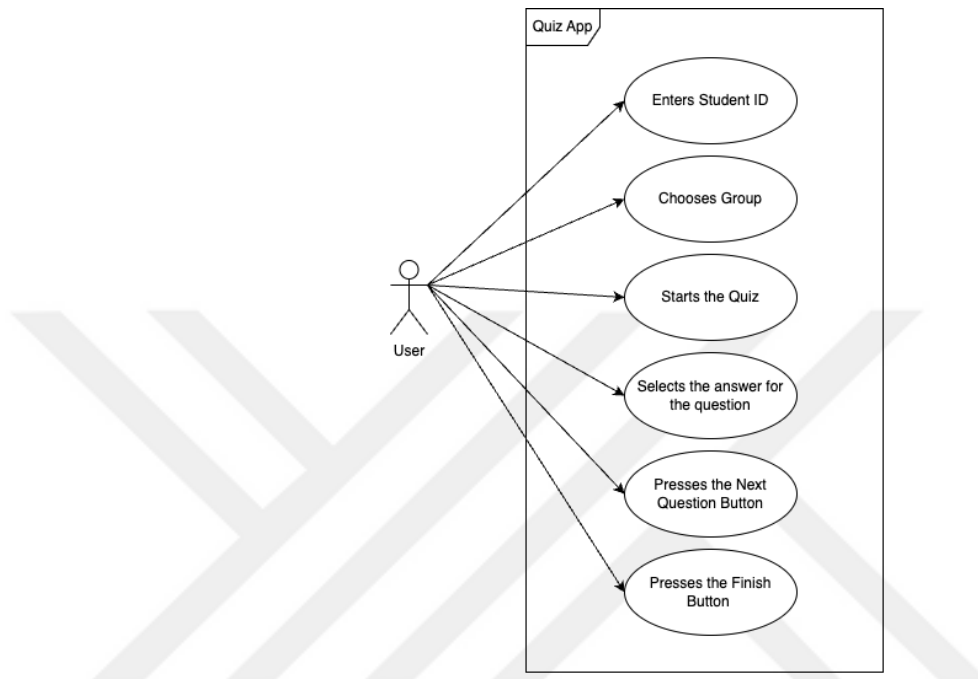


Figure 3. 2 Use Case Diagram of the System

3.4. Research Procedure

Considering the following research questions, we have conducted an experimental study with 122 students from the departments of Computer, Software and Information Systems Engineering.

RQ1. How can a web-based application be developed by using gamification elements for learning programming languages?

In order to answer this research question, a web-based application is developed for the assessment of the students' knowledge in a C programming course. A multiple-choice question test is prepared with 10 questions (see Appendix A). This web-based environment also includes gamification elements like the duration passed to answer each question, the students' scores and the leaderboard.

RQ2. In terms of usability of the developed web-based application with the integration of gamification elements, are the students satisfied with the application for learning C language?

To understand the students' perceptions and getting feedbacks regarding the application, a System Usability Scale (SUS) survey (see Appendix B, Appendix C) is applied to the students. Usability is an important software quality attribute, based on the concept "is it designed for the user?". If the system is designed with higher usability, users can learn the content rapidly and keep this information in mind for longer periods of time, which directly links to reduced training time.

RQ3. Do the students prefer to use the system frequently?

Similarly, to answer this research question, an item "I think that I would like to use this system frequently" in System Usability Scale (SUS) survey (see Appendix B, Appendix C) is used and the data is collected by using a paper-based survey.

Experimental Study

The experimental study took place in the four laboratories of the Computer Engineering Department. The students are given 30 minutes in total to answer the multiple-choice test. The research procedure steps for the experimental study are described as follows:

1. On the day of the experiment, the students came to the laboratory and randomly assigned to one of the laboratories.
2. They opened the computers and the web address of the application is shared with the students. A demo is made by the Instructor to all the students to provide a general idea of the system and show how the system works. All of the students are informed about the requirements of the experimental study and the features of the developed system.
3. When the web application opened, students should enter their student ids in the application into a textbox named as User Id. Their personal information is not asked regarding the privacy concerns. Two versions of the web-based application is developed game (with gamification elements) and Normal (without gamification elements). In our current study, all students were asked to play the game version since we gather their feedbacks regarding a web-based environments' usability by using the SUS survey. In further research, both versions can be used to make a group comparison

to better understand their attitude regarding the gamified version. There is also a short notice on the use of application on the entrance screen of the application. There were remarks that the students have to answer 10 multiple choice questions and for each question they were given 3 minutes to answer.

4. After entering the student Id and choosing the Game group, they should press the “Start” button to start the application.

5. The students answer 10 multiple choice questions regarding C Programming by using a web-based application in computer laboratories. They can see the time remaining for each question from application on their screens. The questions included in the application covered the following topics on C programming:

- C Language Elements- 2 questions
- Input/ Output Statements- 1 questions
- Selection Statements- 2 questions
- Loop Statements- 3 questions
- Functions- 2 questions

6. To see the next question students should press on the “Next” button on the application. Once a question is answered, it is recorded to the database and the student cannot get back to a previous question.

7. During the experimental study, the students can also see their current progress from the score and the leaderboard panel on the screen.

8. After the experiment finished, the user id, type (game version), question asked, the students’ answer, time taken to answer this question (in seconds), the score value, is the answer correct (TRUE or FALSE) values are recorded into the database for each student.

9. At the end of the experiment, the students are asked to fill out the System Usability Scale (SUS) survey (see Appendix B, Appendix C). It is a paper-based survey then after the students completed the survey, they handed in the papers to the instructor in the laboratory.

In the System Usability Scale (SUS) survey (see Appendix B, Appendix C), there were ten questions prepared in a 5-point likert scale format that participants asked to answer in one of the five alternatives (1: Strongly Disagree, 2: Disagree, 3: Neither Agree Nor Disagree, 4: Agree, 5: Strongly Agree). However, to better summarize their opinions

in the results section, their answers are evaluated in three levels: the participants' responses as 1: Strongly Disagree and 2: Disagree are together evaluated as Disagree (1-2), Neither Agree Nor Disagree (3), the participants' responses as 4: Agree, 5: Strongly Agree are together evaluated as Agree (4-5).



CHAPTER 4

RESULTS

This thesis focuses on the development of a web-based gamified learning environment whereas the results are also presented from usability perspective of the students. In this chapter, the results were provided in two main subsections. Firstly, the details for the development of the web based system is given. Secondly, the results of the “System Usability Scale (SUS)” survey is presented to better understand the students’ perceptions regarding this application and their attitudes towards using such web-based environments for learning C programming language with gamification.

4.1. Results on the Developed Web-Based System

In order to answer the first research question “RQ1. How can a web-based application be developed by using gamification elements for learning programming languages?”, the details of the developed web-based application is given in this subsection. The application code snippets are also available with the descriptions in Appendix D.

System Interface

To give detailed information about the developed system, firstly, we intent to mention the User Interface (UI) of the system. The UI plays an important role to define the usability of the system; hence we have presented the screens starting from the entrance to the system and finishing or last screen shown on the system. As well as, the components which have been used in the creation of screens are also explicitly mentioned so the design of system and the interface can be fully understood.

Entrance Screen

A student enters the student id and chooses “Game” group. A non-game environment is also created for comparison but not used in the current study. All students have attended the “gamified” version of the quiz, to take their feedbacks about the usability of the gamified web-based application. The entrance screen is seen as given in Figure 4.1.

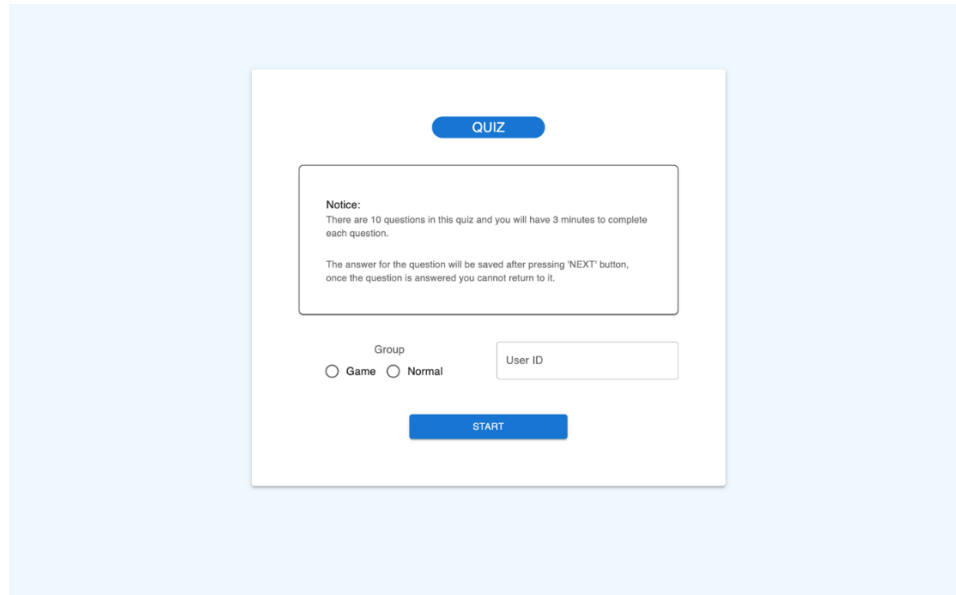


Figure 4. 1 View of Entrance Screen

Information regarding the components on the entrance screen is given below:

1. Notice

To provide relevant information regarding the multiple-choice test, a notice is provided as seen in Figure 4.2 on the entrance screen giving information about the number of questions and duration. Also, they were informed that returning to a previous question is not available in this system.

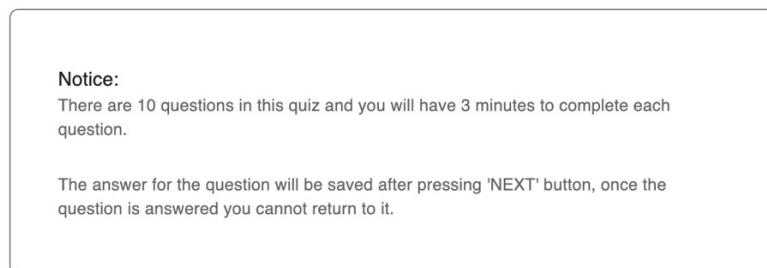


Figure 4. 2 View of Notice Component on Entrance Screen

2. Input Fields

To obtain the group information two radio buttons are placed in the applications' entrance screen as shown in Figure 4.3. Additionally, a textbox is used to gather the User ID information, which is the student ID of the students. This field is mandatory and exactly 11 characters should be entered (check for validity is done to obtain the correct information for the student IDs).

Group
 Game Normal

User ID

Figure 4. 3 View of the Input Fields in Application

User should enter an 11-digit student id and choose the group information to start the quiz application. This information is validated in the system as shown in the following Figure 4.4:

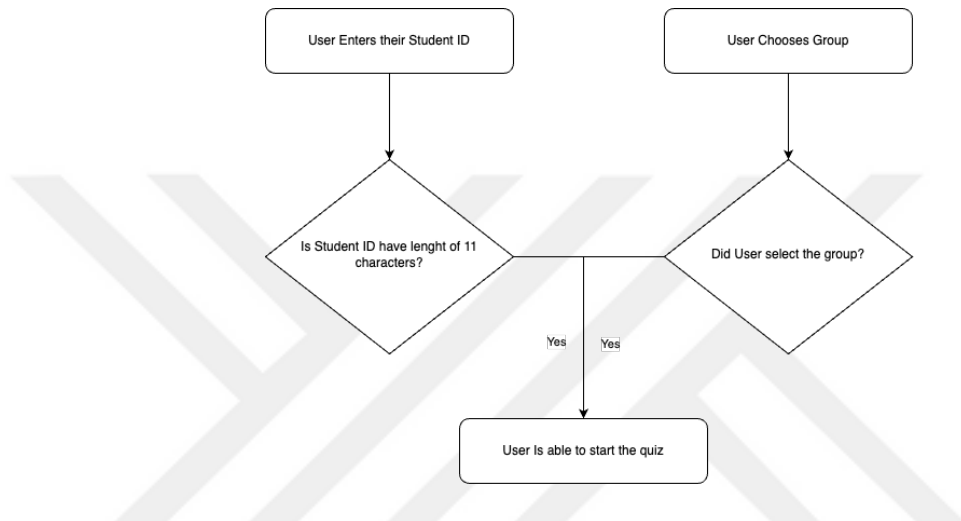


Figure 4. 4 Diagram containing validation method for the keys in database schema

3. Button

To start the application, after reading the notice and entering the required information, the user should click on the start button as shown in Figure 4.5.



Figure 4. 5 Start Button Design

Question Screen

Then, a student starts the application. Each question is shown on a separate page in the application. The question number, remaining time, score and leaderboard and can be seen as shown in Figure 4.6. To navigate through the next question, the user should click on the “Next” button.

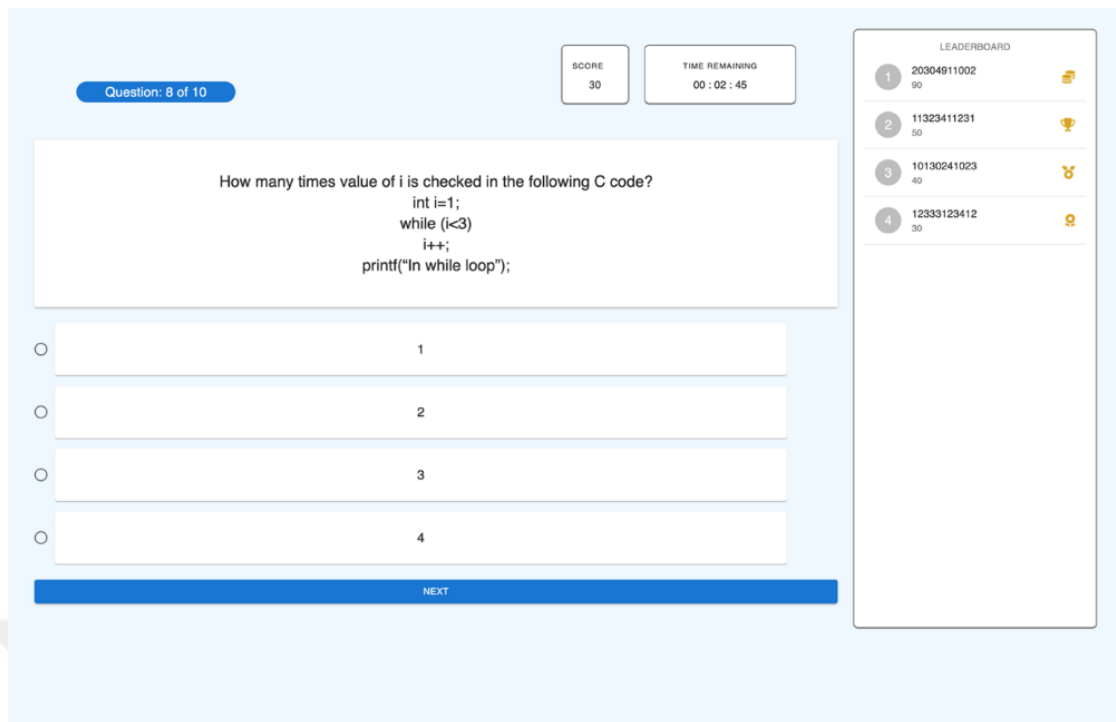


Figure 4. 6 View of Question Screen

As seen from the question view, following game elements were used in the application:

- Score
- Time
- Leaderboard
 - Score of participants
 - Medals
 - Own Score Compared
 - Own Medal

The score and type of medal changes with question being answered correctly. Information regarding the components on the question screen is given below:

1. Leaderboard

The students can see their progress while solving the test, using the leaderboard panel located on the right side of the question screen as seen in Figure 4.7.

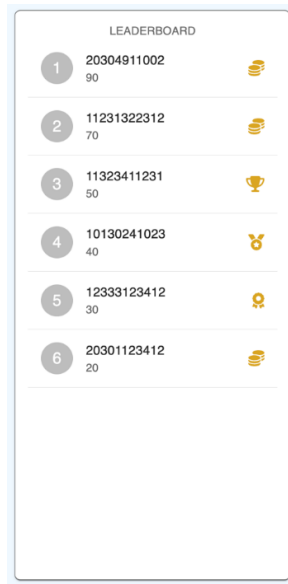


Figure 4. 7 The Leaderboard Panel Design

In Figure 4.7, the student IDs, their current scores are seen on the leaderboard screen. After reaching a limit score, they are awarded by a different icon (as seen on the right side of the panel) such as a medal, or a cup.

2. Score and Time Component

On the top of the question screen, the current score of the user and the remaining time for each question is seen as in Figure 4.8.

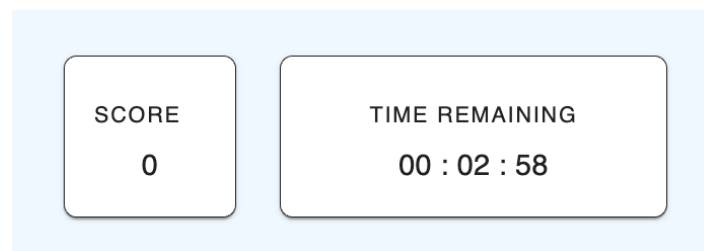


Figure 4. 8 View of Score and Time Component

3. Question and Answer Component

In the middle of the question screen, question & answer component is residing as seen in Figure 4.9. There is a text area showing the question, four radio buttons for the multiple choices of the question and to submit the answer there is a “Next” button.

Which of the following logical expressions has the value false if a and b have been declared as follows? int a=5, b=2;

(a >= b)&&(b != 0)

(a % b)|| (b == 0)

1 || !0

(b - 2 != 0)

NEXT

Figure 4. 9 View of Question-Answer Component

4. Feedback Element

The feedback is a pop-up element, seen when the student clicked on the “Next” button as seen in Figure 4.10. This element is used to notify the user that their answer is recorded to the database.

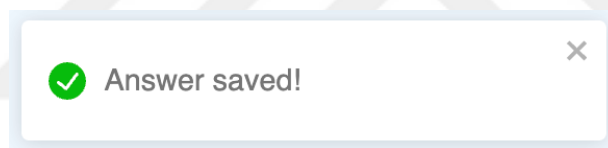


Figure 4. 10 View of Feedback Component

The view of the Question Screen with Feedback Element is given in Figure 4.11. The pop-up information is seen on the bottom-right corner of the screen as a feedback to the user.

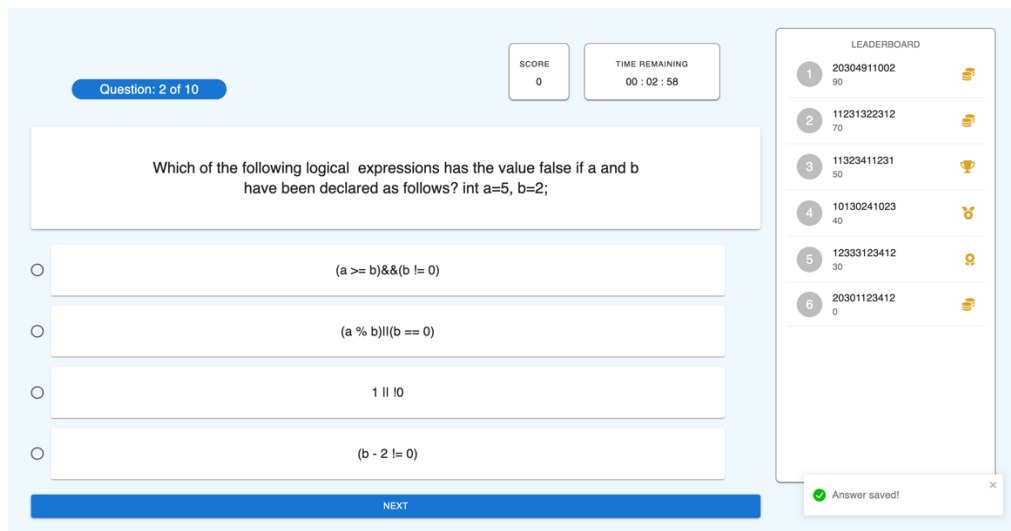


Figure 4. 11 View of the Question Screen with Feedback Element

Final Screen

After the user finish the quiz activity, a message appears on the screen as shown in Figure 4.12. There is also a button for re-starting the application. If the student wants to do the exercises again, the application will start from the entrance screen.

In further development, a statistical results panel can be added to this design to show the students' progress, such as the number of correct and incorrect answers, overall score and ranking. The score and number of correct and incorrect answers are kept in the applications' database so feedbacks regarding their performances can be evaluated by the instructors based on reporting facilities.

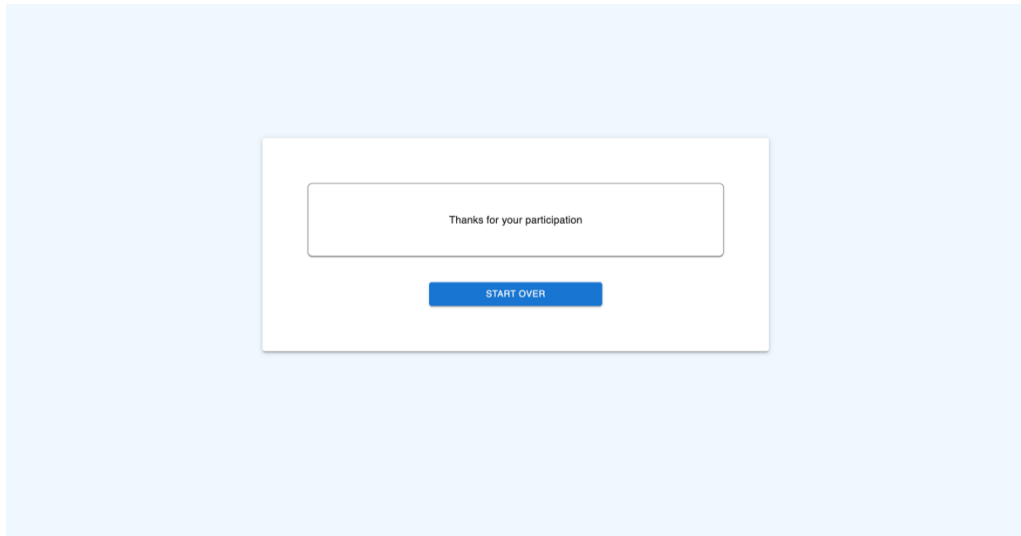


Figure 4. 12 View of the Finish Screen

Additionally, when a student finished the application, two pop-up information windows are seen on the bottom right corner, one is for the last answer to be saved and the other is a notification that the overall quiz application is finished successfully. The view of the Finish Screen with feedback elements is seen in Figure 4.13.

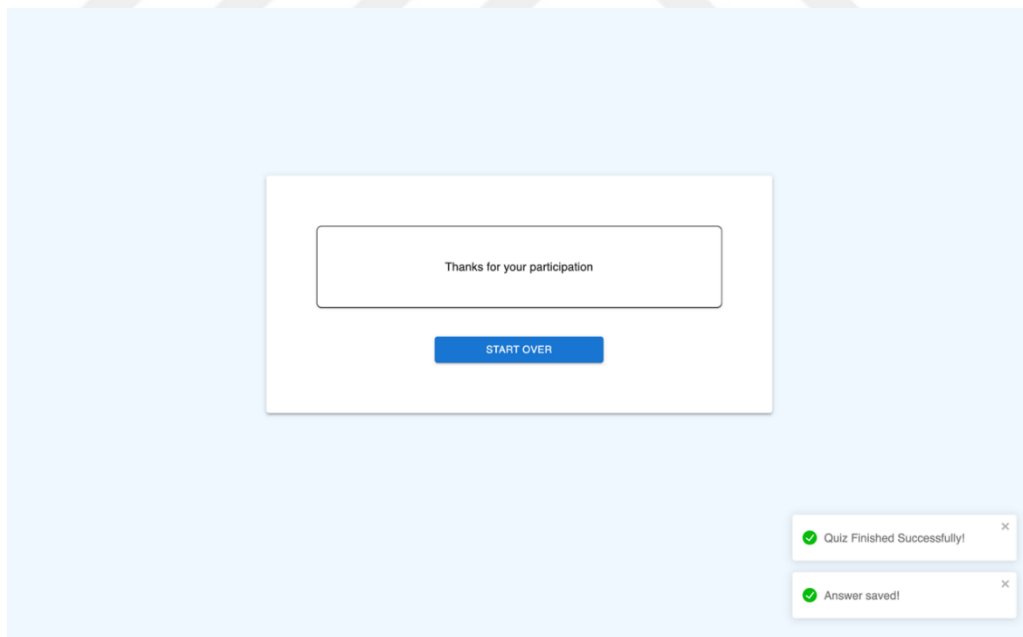
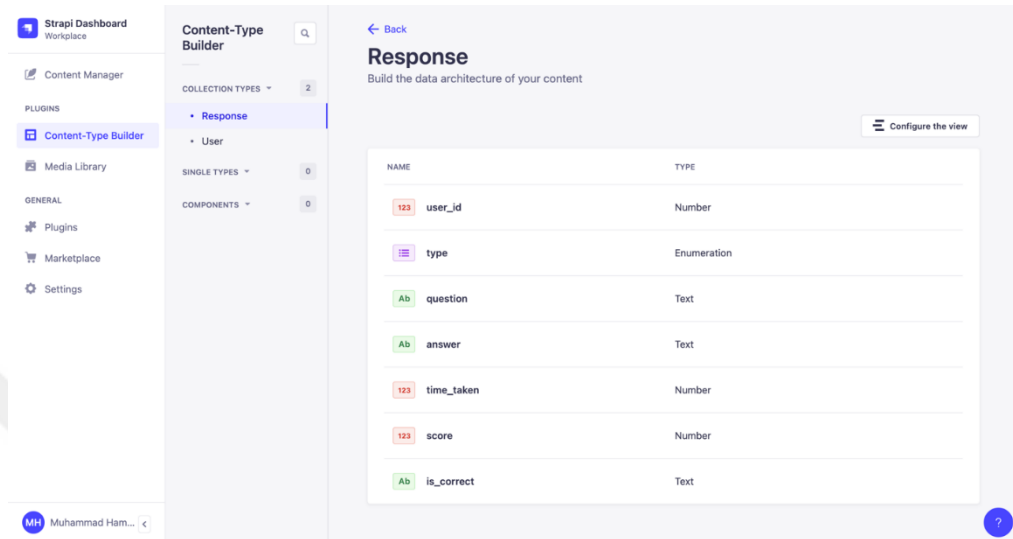


Figure 4. 13 View of the Finish Screen with Feedback Elements

After the application attempt finishes, the students' responses, and some measures like time taken to answer each question, the success information (correct or wrong) is kept in a database system. The design of the database schema in the Content Management System (CMS) is shown in Figure 4.14.



The screenshot shows the Strapi Dashboard interface. On the left is a sidebar with navigation options: Strapi Dashboard Workplace, Content Manager, PLUGINS (Content-Type Builder, Media Library), GENERAL (Plugins, Marketplace, Settings), and a user profile for Muhammad Ham... On the right, the 'Content-Type Builder' section is active, showing the 'Response' collection type. Below this, a table displays the database schema for the 'Response' collection type.

NAME	TYPE
123 user_id	Number
Ab type	Enumeration
Ab question	Text
Ab answer	Text
123 time_taken	Number
123 score	Number
Ab is_correct	Text

Figure 4. 14 The Design of the Database Schema

After the data is recorded in database, the information is stored in a table named “responses” and can be exported as a .csv file for further analysis as shown in Figure 4.15 and Figure 4.16, respectively.

id	user_id	type	question	answer	time_taken	score	is_correct	created_at	updated_at	published_at	created_by_id	updated_by_id
1	2030491002	game	Which of the following is a valid variable name according to C identifier naming rules?	Sum	40	10	true	1686934537005	1686934537005	1686934537003	AKELI	AKELI
2	2030491002	game	Which of the following logical expressions has the value false if a and b have been declared as follows? int a=5, b=2;	(b - 2 != 0)	9	10	true	1686934546163	1686934546163	1686934546162	AKELI	AKELI
3	2030491002	game	Fill in the blanks: the program finds the multiplication of two variables if the first one of these variables is greater than the second, otherwise calculates summation of these numbers? ... result = num1 * num2; ... result = num1 + num2;	if (num1 > num2) - else	8	0	false	1686934554817	1686934554817	1686934554816	AKELI	AKELI
4	2030491002	game	Given the following code segment: switch (ch) case 'A': area=side*side; printf("Area is %d", area); break; case 'P': perimeter=4*side; printf("Perimeter is %d", perimeter); break; default: printf("Wrong input!");	Wrong input!	6	10	true	1686934560921	1686934560921	1686934560920	AKELI	AKELI
5	2030491002	game	Suppose you want to print the double variable profit right-justified by using two decimal places after the decimal point. Which one of these printf statements will do?	printf("your profit is %7.2f Turkish liras %", profit);	6	10	true	1686934570000	1686934570000	1686934569998	AKELI	AKELI
6	2030491002	game	Which of the following loops execute the statements at least once?	Do...while	18	10	true	1686934589014	1686934589014	1686934589012	AKELI	AKELI
7	2030491002	game	What is the output of the following code block program? int i, j, k=2; for(i=1; i<=k; i++) for(j=1; j<=i; j++) printf("%d", j); printf("\n");	***	17	10	true	1686934606910	1686934606910	1686934606909	AKELI	AKELI
8	2030491002	game	How many times value of i is checked in the following C code? int i=1; while (i<=3) i++; printf("in while loop");	3	10	10	true	1686934617589	1686934617589	1686934617582	AKELI	AKELI
9	2030491002	game	findMax is a function that accepts three double parameters and returns no value. Three double variables num1, num2 and num3, have already been declared and initialized in main. What is the correct statement that calls findMax by sending num1, num2 and num3?	findMax (num1, num2, num3);	8	10	true	1686934626361	1686934626361	1686934626359	AKELI	AKELI
10	2030491002	game	Which one is the correct call statement of a function my_function which accepts two parameters and returns integer type variable?	int num = my_function (a, b);	8	10	true	1686934635346	1686934635346	1686934635343	AKELI	AKELI
11	10130241023	game	Which of the following is a valid variable name according to C identifier naming rules?	Sum	5	10	true	1686934663666	1686934663666	1686934663665	AKELI	AKELI
12	10130241023	game	Which of the following logical expressions has the value false if a and b have been declared as follows? int a=5, b=2;	(b - 2 != 0)	10	10	true	1686934674551	1686934674551	1686934674549	AKELI	AKELI
13	10130241023	game	Fill in the blanks: the program finds the multiplication of two variables if the first one of these variables is greater than the second, otherwise calculates summation of these numbers? ... result = num1 * num2; ... result = num1 + num2;	while (num1 > num2) - else	4	0	false	1686934678810	1686934678810	1686934678809	AKELI	AKELI
14	10130241023	game	Given the following code segment: switch (ch) case 'A': area=side*side; printf("Area is %d", area); break;	Wrong input!	13	10	true	1686934692355	1686934692355	1686934692355	AKELI	AKELI

Figure 4. 15 Table in the Database

id	user_id	type	question	answer	time_taken	score	is_correct	created_at	updated_at	published_at	created_by_id	updated_by_id
1	2030491002	game	Which of the following is a valid variable name according to C identifier naming rules?	Sum	40	10	TRUE	1686934537005	1686934537005	1686934537003	AKELI	AKELI
2	2030491002	game	Which of the following logical expressions has the value false if a and b have been declared as follows? int a=5, b=2;	(b - 2 != 0)	9	10	TRUE	1686934546163	1686934546163	1686934546162	AKELI	AKELI
3	2030491002	game	Fill in the blanks: the program finds the multiplication of two variables if the first one of these variables is greater than the second, otherwise calculates summation of these numbers? ... result = num1 * num2; ... result = num1 + num2;	if (num1 > num2) - else	8	0	FALSE	1686934554817	1686934554817	1686934554816	AKELI	AKELI
4	2030491002	game	Given the following code segment: switch (ch) case 'A': area=side*side; printf("Area is %d", area); break; case 'P': perimeter=4*side; printf("Perimeter is %d", perimeter); break; default: printf("Wrong input!");	Wrong input!	6	10	TRUE	1686934560921	1686934560921	1686934560920	AKELI	AKELI
5	2030491002	game	Suppose you want to print the double variable profit right-justified by using two decimal places after the decimal point. Which one of these printf statements will do?	printf("your profit is %7.2f Turkish liras %", profit);	6	10	TRUE	1686934570000	1686934570000	1686934569998	AKELI	AKELI
6	2030491002	game	Which of the following loops execute the statements at least once?	Do...while	18	10	TRUE	1686934589014	1686934589014	1686934589012	AKELI	AKELI
7	2030491002	game	What is the output of the following code block program? int i, j, k=2; for(i=1; i<=k; i++) for(j=1; j<=i; j++) printf("%d", j); printf("\n");	***	17	10	TRUE	1686934606910	1686934606910	1686934606909	AKELI	AKELI
8	2030491002	game	How many times value of i is checked in the following C code? int i=1; while (i<=3) i++; printf("in while loop");	3	10	10	TRUE	1686934617589	1686934617589	1686934617582	AKELI	AKELI
9	2030491002	game	findMax is a function that accepts three double parameters and returns no value. Three double variables num1, num2 and num3, have already been declared and initialized in main. What is the correct statement that calls findMax by sending num1, num2 and num3?	findMax (num1, num2, num3);	8	10	TRUE	1686934626361	1686934626361	1686934626359	AKELI	AKELI
10	2030491002	game	Which one is the correct call statement of a function my_function which accepts two parameters and returns integer type variable?	int num = my_function (a, b);	8	10	TRUE	1686934635346	1686934635346	1686934635343	AKELI	AKELI
11	10130241023	game	Which of the following is a valid variable name according to C identifier naming rules?	Sum	5	10	TRUE	1686934663666	1686934663666	1686934663665	AKELI	AKELI
12	10130241023	game	Which of the following logical expressions has the value false if a and b have been declared as follows? int a=5, b=2;	(b - 2 != 0)	10	10	TRUE	1686934674551	1686934674551	1686934674549	AKELI	AKELI
13	10130241023	game	Fill in the blanks: the program finds the multiplication of two variables if the first one of these variables is greater than the second, otherwise calculates summation of these numbers? ... result = num1 * num2; ... result = num1 + num2;	while (num1 > num2) - else	4	0	FALSE	1686934678810	1686934678810	1686934678809	AKELI	AKELI
14	10130241023	game	Given the following code segment: switch (ch) case 'A': area=side*side; printf("Area is %d", area); break;	Wrong input!	13	10	TRUE	1686934692355	1686934692355	1686934692355	AKELI	AKELI

Figure 4. 16 CSV File from the Database

4.2. Results on the Usability of the Web-Based System

In order to answer the second research question “RQ2. In terms of usability of the developed web-based application with the integration of gamification elements, are the students satisfied with the application for learning C language?”, the results of the

survey about “System Usability Scale (SUS)” is given to the students to get their feedback about the web-based gamified application. Students’ responses were collected through a paper-based questionnaire. In the following section, the results of the “System Usability Scale (SUS)” survey (see Appendix B, Appendix C) is presented. The survey included 10 items and grouped under three subsections: positive usability, negative usability, and frequency of use. The results taken from questionnaire were analyzed descriptively.

Typically, the survey employs a five-point Likert scale format, however, to facilitate interpretation, responses 1-strongly disagree and 2-disagree are combined and are evaluated as "Disagree," while items 4-agree and 5-strongly agree are combined and are evaluated as "Agree."

To better understand the usability of the system in terms of ease of use, integration of functions, quick learning, and user confidence, the responses of the students to the following four items of the questionnaire (Questionnaire Items: 3, 5, 7 and 9- see Appendix B) were analyzed descriptively.

- *I thought the system was easy to use.*

Table 4. 1 Questionnaire Results for Item-3

	Item-3	%
Disagree (1-2)	14	12%
Neither Agree Nor Disagree (3)	9	8%
Agree (4-5)	96	81%
Total Number of Responses	119	

As identified from Table 4.1, Majority of the participants (81%) agreed that the system was easy to use. Only 12% of the students do not agree with the idea that system is easy to use.

- *I found the various functions in this system were well integrated.*

Table 4. 2 Questionnaire Results for Item-5

	Item-5	%
Disagree (1-2)	14	11%
Neither Agree Nor Disagree (3)	28	23%
Agree (4-5)	80	66%
Total Number of Responses	122	

As noticed from Table 4.2, 66% of the participants found that various functions in this system were well integrated, whereas 23% of them do not agree or disagree with this view. The results show that the majority of the users are satisfied with the system from the efficiency perspective and agreed that there are no inconsistencies in the system.

- *I would imagine that most people would learn to use this system very quickly.*

Table 4. 3 Questionnaire Results for Item-7

	Item-7	%
Disagree (1-2)	14	11%
Neither Agree Nor Disagree (3)	12	10%
Agree (4-5)	96	79%
Total Number of Responses	122	

As seen from Table 4.3, 79% of the participants found that most people would learn to use this system very quickly as to 11% of them does not agree with this perspective.

- *I felt very confident using the system.*

Table 4. 4 Questionnaire Results for Item-9

	Item-9	%
Disagree (1-2)	18	15%
Neither Agree Nor Disagree (3)	19	16%
Agree (4-5)	85	70%
Total Number of Responses	122	

As viewed from Table 4.4, 70% of the participants found that they feel confident when using the system, however 15% of them do not agree with this view.

In the following part, the applications' usability is evaluated according to the complexity, dependence on technical support, and inconsistency, students' feedback to the following five items of the questionnaire (Questionnaire Items: 2, 4, 6, 8 and 10- see Appendix B) are analyzed descriptively.

- *I found the system unnecessarily complex.*

Table 4. 5 Questionnaire Results for Item-2

	Item-2	%
Disagree (1-2)	80	67%
Neither Agree Nor Disagree (3)	16	13%
Agree (4-5)	24	20%
Total Number of Responses	120	

As seen from Table 4.5, 67% of the participants disagreed that they found the system unnecessarily complex, whereas only 20% of them agree with this view.

- *I think that I would need the support of a technical person to be able to use this system.*

Table 4. 6 Questionnaire Results for Item-4

	Item-4	%
Disagree (1-2)	76	62%
Neither Agree Nor Disagree (3)	18	15%
Agree (4-5)	28	23%
Total Number of Responses	122	

As observed from Table 4.6, 62% of the participants disagree with the proposal that they need the support of a technical person to be able to use this system, compared to 23% of them thinks that they need assistance.

- *I thought there was too much inconsistency in this system.*

Table 4. 7 Questionnaire Results for Item-6

	Item-6	%
Disagree (1-2)	79	66%
Neither Agree Nor Disagree (3)	20	17%
Agree (4-5)	21	18%
Total Number of Responses	120	

As viewed from Table 4.7, 66% of the participants disagree with the idea that there are too many inconsistencies within the system, whereas only 18% of them agreed with this view.

- *I found the system very cumbersome to use.*

Table 4. 8 Questionnaire Results for Item-8

	Item-8	%
Disagree (1-2)	66	54%
Neither Agree Nor Disagree (3)	31	25%
Agree (4-5)	25	20%
Total Number of Responses	122	

As seen from Table 4.8, More than half (54%) of the participants disagree with the idea that the system is very cumbersome to use, however 25% of them neither agree nor disagree with this perspective. This item is mostly related to the user satisfaction which shows that they are mostly satisfied with this system.

- *I needed to learn a lot of things before I could get going with this system.*

Table 4. 9 Questionnaire Results for Item-10

	Item-10	%

Disagree (1-2)	70	57%
Neither Agree Nor Disagree (3)	25	20%
Agree (4-5)	27	22%
Total Number of Responses	122	

As seen from Table 4.9, More than half (57%) of the participants disagree with the idea that they needed to learn a lot of things before they could get going with this system, whereas 22% of them agreed with this opinion.

Lastly, to answer the last research question “RQ3. Do the students prefer to use the system frequently?”, the students’ responses to the following item (Questionnaire Item:1-see Appendix B) is analyzed descriptively. This item of the survey is related to the usefulness of the system. So, with this question, the students evaluated if the web-based application is useful for them or not. The results of this survey item is presented in Table 4.10.

- *I think that I would like to use this system frequently.*

Table 4. 10 Questionnaire Results for Item-1

	Item-1	%
Disagree (1-2)	17	14%
Neither Agree Nor Disagree (3)	29	24%
Agree (4-5)	75	62%
Total Number of Responses	121	

As noticed from Table 4.10, 62% of the participants agreed with the idea that they would like to use this system frequently, however only 14% of them disagreed with this idea. In another words, they found the system useful for them.

CHAPTER 6

DISCUSSION

The use of gamification in educational settings has gained significant attention due to its potential to enhance student engagement and learning outcomes. This article focuses on the development and evaluation of a web-based application that incorporates gamification elements to facilitate the learning of C programming language. The aim is to provide an interactive and innovative learning environment that caters to learners of varying proficiency levels, ensuring a comprehensive understanding of the language.

The development process of the web-based application involved several key components. Firstly, a user-friendly interface was designed to create an intuitive and accessible learning platform. Interactive coding exercises were incorporated to actively engage learners and encourage hands-on practice. Additionally, comprehensive learning materials were curated to provide a structured learning path, starting from the fundamentals of C programming, and progressing to more complex topics. This step-by-step approach ensures a gradual and comprehensive understanding of the subject matter.

To assess the effectiveness of the web-based application, a pilot study will be conducted with a group of students learning C programming. The study aims to evaluate students' engagement levels, learning outcomes, and overall satisfaction with the application. Quantitative data will be collected through surveys and performance assessments, while qualitative feedback will be gathered to gain valuable insights into students' experiences and to collect suggestions for further improvement.

Based on this data gathered from the SUS survey, the main findings of this study regarding the usability of such web-based systems can be listed as:

According to the results on factors contributing to the positive usability of the system in terms of ease of use, integration of functions, quick learning, and user confidence:

- Majority of the participants (81%) agreed that the system was easy to use.

- 66% of the participants found that various functions in this system were well integrated.
- 79% of the participants found that most people would learn to use this system very quickly.
- 70% of the participants found that they feel confident using the system.

The survey results reveal highly positive feedback regarding the usability of the system. A significant majority of participants, comprising 81%, expressed agreement that the system was easy to use. This indicates that the design and interface of the web-based application effectively provided to users' needs and facilitated a smooth learning experience. Furthermore, 66% of the participants acknowledged the seamless integration of various functions within the system, highlighting its cohesive and well-organized nature. The finding that 79% of participants believed that most people would quickly learn to use the system demonstrates its intuitive design and user-friendly approach. Additionally, a substantial 70% of participants reported feeling confident when using the system, implying that it instilled a sense of competence and ease of navigation. Collectively, these positive survey responses provide compelling evidence of the system's high usability, affirming its effectiveness in facilitating a user-friendly and accessible learning environment for C programming.

Based on the results on the factors leading to the negative usability of the system in terms of complexity, dependence on technical support, and inconsistency:

- 67% of the participants disagreed that they found the system unnecessarily complex.
- 62% of the participants disagree with the idea that they need the support of a technical person to be able to use this system.
- 66% of the participants disagree with the idea that there are too much inconsistencies within the system.
- 54% of the participants disagree with the opinion that the system is very cumbersome to use.
- 57% of the participants disagree with the view that they needed to learn a lot of things before they could get going with this system.

The survey results indicate a generally favorable perception of the system's usability, with participants expressing a low level of disagreement regarding various aspects. A

notable 67% of participants disagreed that they found the system unnecessarily complex, suggests that the system effectively presented information and features are in a clear and understandable manner. Similarly, 62% of participants disagreed with the idea that they needed the support of a technical person to use the system, indicating that it was designed to be accessible to users with diverse backgrounds and skill levels. Moreover, 66% of participants disagreed that there were too many inconsistencies within the system, indicating a relatively coherent and consistent user experience.

Based on the results on the factor influencing the likelihood and desire of users to frequently utilize the system:

- 62% of the participants agreed with the idea that they would like to use this system frequently.

The finding that 62% of the participants agreed with the view of wanting to use the system frequently opens an intriguing discussion on the potential impact and user satisfaction associated with the web-based application. This high percentage suggests that the system effectively captivated users and fostered a desire for continued engagement. The positive inclination towards frequent usage implies that the system's design, interactive coding exercises, and comprehensive learning materials were successful in creating an engaging and rewarding learning experience for the participants. The desire to utilize the system regularly could also indicate that it met the participants' needs for practicing and advancing their C programming skills.

As motivation theorists have long argued, intrinsic motivation plays a crucial role in shaping students' learning experiences and academic achievements. When individuals are genuinely interested and engaged in the educational process, they are more likely to be proactive in seeking knowledge, persist in challenging tasks, and display a deeper level of understanding [40]. This intrinsic drive fosters a sense of ownership and personal investment in their learning journey, leading to better outcomes and a higher level of overall satisfaction with their educational experience.

Within the realm of instructional design, incorporating motivational elements has been shown to yield positive results. [41] By integrating motivational design principles into the learning environment, educators can create experiences that resonate with students, ignite their curiosity, and evoke a genuine interest in the subject matter. This, in turn,

leads to heightened engagement and a more focused approach to learning, facilitating better retention and application of knowledge.

The present study's focus on intrinsic motivation and learning achievement shed light on the dynamic relationship between these two variables within different learning contexts. The findings revealed that when learning activities were designed to captivate students' curiosity and stimulate their interest, and when students expressed satisfaction with the system's functionality, such as in an outdoor educational environment, their intrinsic motivation levels surged. As a result, they displayed a greater commitment to their studies, exhibited a higher level of enthusiasm, and experienced a more profound sense of achievement in their learning endeavors.

In line with our results, prior research by Liu and Chu [42] also demonstrated a strong positive correlation between learning motivation and academic performance. This observation further reinforces the significance of intrinsic motivation as a driving force behind students' learning success.

The findings from the present study contribute to the growing body of research that emphasizes the pivotal role of motivation in educational settings. They highlight the importance of creating learning environments that not only cater to students' academic needs but also foster a sense of joy, curiosity, and self-directed learning. This can be achieved through purposeful instructional design, interactive learning experiences, and the thoughtful integration of technologies that enhance engagement.

Furthermore, the study's alignment with previous research [21], which reported a positive motivational level supporting the usability of the application, strengthens the validity and reliability of the current findings. It suggests that a motivational approach not only enhances usability but also positively influences students' overall learning experiences and outcomes.

Henceforth, understanding and harnessing the power of intrinsic motivation is essential for educators and instructional designers seeking to optimize students' learning experiences. By incorporating motivational design elements and creating stimulating learning environments, educators can foster a deep sense of engagement, curiosity, and motivation, ultimately leading to improved learning outcomes and academic achievements for their students. Further investigation into the specific

elements of the system that contributed to this positive sentiment can provide valuable insights for the refinement and expansion of the application, while also highlighting the potential for long-term user engagement and satisfaction.



CHAPTER 7

CONCLUSION

In conclusion, the integration of gamification in learning C programming through the development of a web-based application holds immense potential to revolutionize programming education. By creating an innovative and interactive learning environment, the application addresses challenges faced by students in grasping the intricacies of C programming and fosters their engagement and understanding.

The user-friendly interface of the web-based application ensures that learners, regardless of their prior programming experience, can navigate through the materials seamlessly. The step-by-step approach, starting from the basics and gradually advancing to complex topics, provides a structured learning path that accommodates learners of varying proficiency levels. This systematic progression not only builds a solid foundation but also encourages learners to challenge themselves and expand their programming skills.

The inclusion of interactive coding exercises within the application promotes active learning and hands-on practice, enabling learners to apply theoretical knowledge in practical scenarios. By actively engaging with the coding exercises, learners can gain valuable experience and develop problem-solving skills, both of which are crucial for success in programming.

Through the experimental study, which assesses students' engagement levels, learning outcomes, and overall satisfaction, the effectiveness of the web-based application will be evaluated. The study's findings will provide insights into the impact of gamification on students' motivation, knowledge retention, and programming proficiency. Additionally, qualitative feedback from students will offer valuable suggestions for further improvement and refinement of the application, ensuring it continues to meet the evolving needs of learners.

Limitations of Study

While the integration of gamification in learning C programming through the web-based application offers promising advantages, it is important to acknowledge certain limitations that may affect the generalizability and interpretation of the study's findings. Sample size and selection process can be a limitation for this study, which involves a specific group of students learning C programming and may not represent the entire population of programming students.

Additionally, the study's duration (only covering one semester) may not allow for long-term evaluation of the web-based application's effectiveness and impact. Long-term follow-up studies would be beneficial to understand the sustainability and durability of the learning outcomes achieved through the application.

Future Work

Building upon the findings and limitations of the current study, several areas of future work can be explored to further enhance the integration of gamification in programming education. The potential contributions of this study extend beyond the realm of programming education. By exploring the integration of gamification in the teaching of programming languages, this research can inform the development of educational technologies in various disciplines. As a future work, the principles and strategies employed in this web-based application can be adapted and applied to other domains.

Additionally, comparative studies can be conducted to evaluate the effectiveness of the web-based application against traditional teaching methods or other gamified approaches. This would provide insights into the relative advantages and disadvantages of different instructional approaches in programming education. Deeper and more informative analyzes can be performed with this way. Lastly, a long-term follow-up studies can be conducted to assess the sustained impact of the web-based application on students' learning outcomes, engagement, and programming proficiency over an extended period. This would provide a deeper understanding of the long-term effectiveness and benefits of gamified learning environments.

REFERENCES

- [1] B. Marín, J. Frez, J. Cruz-Lemus, and M. Genero, “An Empirical Investigation on the Benefits of Gamification in Programming Courses,” *ACM Transactions on Computing Education*, vol. 19, no. 1, pp. 1–22, Nov. 2018.
- [2] J. Díaz-Ramírez, “Gamification in engineering education – An empirical assessment on learning and game performance,” *Heliyon*, vol. 6, no. 9, p. e04972, Sep. 2020.
- [3] C.-H. Su, “The effects of students’ motivation, cognitive load and learning anxiety in gamification software engineering education: a structural equation modeling study,” *Multimedia Tools and Applications*, vol. 75, no. 16, pp. 10013–10036, Jul. 2015.
- [4] R. M. Sanso and Cristina Manresa-Yee, “Gamifying an Artificial Intelligence Course in Engineering Education,” *International Journal of Engineering Education*, vol. 32, pp. 513–520, 2016.
- [5] M.-B. Ibanez, A. Di-Serio, and C. Delgado-Kloos, “Gamification for Engaging Computer Science Students in Learning Activities: A Case Study,” *IEEE Transactions on Learning Technologies*, vol. 7, no. 3, pp. 291–301, Jul. 2014.
- [6] M. Milosz and E. Milosz, "Gamification in Engineering Education – a Preliminary Literature Review," *IEEE Global Engineering Education Conference (EDUCON)*, 2020, pp. 1975–1979.
- [7] S. Mohtar, N. Jomhari, N. A. Omar, M. B. P. Mustafa, and Z. M. Yusoff, “The usability evaluation on mobile learning apps with gamification for middle-aged women,” *Education and Information Technologies*, vol. 28, no. 1, pp. 1189–1210, Jul. 2022.
- [8] A. Bartel and G. Hagel, "Gamifying the learning of design patterns in software engineering education," *IEEE Global Engineering Education Conference (EDUCON)*, 2016, pp. 74–79.
- [9] M. R. d. A. Souza, K. F. Constantino, L. F. Veado and E. M. L. Figueiredo, "Gamification in Software Engineering Education: An Empirical Study," *IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*, 2017, pp. 276–284.
- [10] S. P. Sarkar, B. Sarker and S. K. A. Hossain, "Cross platform interactive programming learning environment for kids with edutainment and gamification," *19th International Conference on Computer and Information Technology (ICCIT)*, 2016, pp. 218–222.

- [11] Firas Layth Khaleel, Noraidah Sahari Ashaari, and Tengku Siti Meriam Tengku Wook, "An empirical study on gamification for learning programming language website," *Jurnal Teknologi*, vol. 81, no. 2, pp. 151–162, Feb. 2019.
- [12] K. Palaniappan and N. Md Noor, "Gamification Strategy to Support Self-Directed Learning in an Online Learning Environment," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 17, no. 03, pp. 104–116, Feb. 2022.
- [13] R. Smiderle, S. J. Rigo, L. B. Marques, J. A. Peçanha de Miranda Coelho, and P. A. Jaques, "The impact of gamification on students' learning, engagement and behavior based on their personality traits," *Smart Learning Environments*, vol. 7, no. 1, Jan. 2020.
- [14] G. Sprint and D. Cook, "Enhancing the CS1 student experience with gamification," *IEEE Integrated STEM Education Conference*, 2015, pp. 94–99.
- [15] E. Lahtinen, K. Ala-Mutka, and H.-M. Järvinen, "A study of the difficulties of novice programmers," *ACM SIGCSE Bulletin*, vol. 37, no. 3, pp. 14–18, Jun. 2005.
- [16] H. C. Jiau, J. C. Chen, and K.-F. Ssu, "Enhancing Self-Motivation in Learning Programming Using Game-Based Simulation and Metrics," *IEEE Transactions on Education*, vol. 52, no. 4, pp. 555–562, Nov. 2009.
- [17] B. Harrington and A. Chaudhry, "TrAcademic: Experiences With Gamified Practical Sessions for a CS1 Course," *Proceedings of the ACM Conference on Innovation and Technology in Computer Science Education*, 2017, pp. 1–2
- [18] C. Latulipe, N. B. Long, and C. E. Seminario, "Structuring Flipped Classes with Lightweight Teams and Gamification," *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, 2015, pp. 392–397.
- [19] P. Kumar and K. Balraj, "Gamification in Education—Learn Computer Programming with Fun," *International Journal of Computers and Distributed Systems*, vol. 2, pp. 46–53, 2012.
- [20] K. Magylaitė, K. Kapočius, R. Butleris, and L. Čeponienė, "Towards High Usability in Gamified Systems: A Systematic Review of Key Concepts and Approaches," *Applied Sciences*, vol. 12, no. 16, p. 8188, Aug. 2022.
- [21] Firas Layth Khaleel, Tengku Siti Meriam Tengku Wook, and Noraidah Sahari Ashaari, "Quantifying user experience in using learning gamification website," *Journal of Theoretical and Applied Information Technology*, vol. 96, no. 23, pp. 7783–7793, Dec. 2018.
- [22] M. Jazayeri, "Some Trends in Web Application Development," *Future of Software Engineering (FOSE '07)*, 2007, pp. 199–213.

- [23] D.-P. Pop and A. Altar, “Designing an MVC Model for Rapid Web Application Development,” *Procedia Engineering*, vol. 69, pp. 1172–1179, 2014.
- [24] T. Fadhilah Iskandar, M. Lubis, T. Fabrianti Kusumasari, and A. Ridho Lubis, “Comparison between client-side and server-side rendering in the web development,” *IOP Conference Series: Materials Science and Engineering*, vol. 801, no. 1, p. 012136, May 2020.
- [25] V. S. Magomadov, “Exploring the role of progressive web applications in modern web development,” *Journal of Physics: Conference Series*, vol. 1679, no. 2, p. 022043, Nov. 2020.
- [26] Madhuri A. Jadhav, Balkrishna R. Sawant, and Anushree Deshmukh, “Single Page Application using AngularJS,” *International Journal of Computer Science and Information Technologies*, vol. 6, no. 3, pp. 2876–2879, 2015.
- [27] L. De Lauretis, “From Monolithic Architecture to Microservices Architecture,” *IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, 2019, pp. 93–96.
- [28] N. Alshuqayran, N. Ali, and R. Evans, “A Systematic Mapping Study in Microservice Architecture,” *IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)*, 2016, pp. 41–51.
- [29] D. Taibi, J. Spillner, and K. Wawruch, “Serverless Computing-Where Are We Now, and Where Are We Heading?,” *IEEE Software*, vol. 38, no. 1, pp. 25–31, Jan. 2021.
- [30] C. Yue and H. Wang, “Characterizing insecure javascript practices on the web,” *Proceedings of the 18th international conference on World wide web*, 2009, pp. 961–970.
- [31] G. Bierman, M. Abadi, and M. Torgersen, “Understanding TypeScript,” *ECOOP 2014 – Object-Oriented Programming*, 2014, pp. 257–281.
- [32] Prateek Rawat and Archana N. Mahajan, “ReactJS: A Modern Web Development Framework,” *International Journal of Innovative Science and Research Technology*, vol. 5, no. 11, pp. 698–702, Nov. 2020.
- [33] E. Ikkala, E. Hyvönen, H. Rantala, and M. Koho, “Sampo-UI: A full stack JavaScript framework for developing semantic portal user interfaces,” *Semantic Web*, vol. 13, no. 1, pp. 69–84, Nov. 2021.
- [34] B. Pfretzschner and L. ben Othmane, “Identification of Dependency-based Attacks on Node.js,” *Proceedings of the 12th International Conference on Availability, Reliability and Security*, 2017, pp. 1–6.

- [35] D. Markovic, M. Scekic, A. Bucaioni, and A. Cicchetti, "Could jamstack be the future of web applications architecture?," *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, 2022, pp. 1872–1881.
- [36] S. Mukherjee, "Benefits of AWS in Modern Cloud," Internet: <https://arxiv.org/pdf/1903.03219.pdf>, 07-Mar-2019. [03-Aug-2023].
- [37] T. Bhuvanewari and S. Prabakaran, "A Survey on Software Development Life Cycle Models," *International Journal of Computer Science and Mobile Computing*, vol. 2, pp. 262-267, May. 2013.
- [38] M. STOICA, M. MIRCEA, and B. GHILIC-MICU, "Software Development: Agile vs. Traditional," *Informatica Economica*, vol. 17, no. 4/2013, pp. 64–76, Dec. 2013.
- [39] A. Sinha and P. Das, "Agile Methodology Vs. Traditional Waterfall SDLC: A case study on Quality Assurance process in Software Industry," *5th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*, 2021, pp. 1–4.
- [40] K. A. Miller, E. L. Deci, and R. M. Ryan, "Intrinsic Motivation and Self-Determination in Human Behavior," *Contemporary Sociology*, vol. 17, no. 2, p. 253, Mar. 1988.
- [41] D. M. Gabrielle and R. K. Branson, "The effects of technology-mediated instructional strategies on motivation, performance, and self-directed learning," *EdMedia+ Innovate Learning. Association for the Advancement of Computing in Education (AACE)*, 2003, pp. 2568–2575.
- [42] T.-Y. Liu and Y.-L. Chu, "Using ubiquitous games in an English listening and speaking course: Impact on learning outcomes and motivation," *Computers & Education*, vol. 55, no. 2, pp. 630–643, Sep. 2010.

APPENDIX A

C PROGRAMMING ACTIVITY: MULTIPLE CHOICE QUESTIONS

Q1. Which of the following is a valid variable name according to C identifier naming rules?

- A) int
- B) n_exists?
- C) Sum
- D) My%percent

Q2. Which of the following logical expressions has the value false if a and b have been declared as follows ?

- ```
int a=5, b=2;
```
- A)  $(a \geq b) \&\&(b \neq 0)$
  - B)  $(a \% b) \|\| (b == 0)$
  - C)  $1 \|\| !0$
  - D)  $(b - 2 \neq 0)$

Q3. Fill in the blanks: the program finds the multiplication of two variables if the first one of these variables is greater than the second, otherwise calculate summation of these numbers?

- ```
.....  
result= num1*num2;  
.....  
result= num1+num2;
```
- A) if (num1>num2) – else
 - B) if (num1<num2) – else if
 - C) if (num1<num2) – else if (num1>num2)
 - D) while (num1>num2) – else

Q4. Given the following code segment:

```
switch (ch){  
case 'A': area=side*side;  
    printf("Area is %d", area);  
    break;  
case 'P': perimeter=4*side;  
    printf("Perimeter is %d", perimeter);  
    break;  
default:  
    printf ("Wrong input!");  
}
```

Assume the side is 4 and ch is 'a' what will be the output?

- A) Area is 16

- B) Perimeter is 16
- C) Wrong input!
- D) None of them

Q5. Suppose you want to print the double variable profit right-justified by using two decimal places after the decimal point. Which one of these printf statements will do?

- A) printf("\nYour profit is %6.4f Turkish liras \n", profit);
- B) printf("\nYour profit is %7.2f Turkish liras \n", profit);
- C) printf("\nYour profit is %-5.3f Turkish liras \n", profit);
- D) printf("\nYour profit is %2.0f Turkish liras \n", profit);

Q6. Which of the following loops execute the statements at least once?

- a) While
- b) Do...while
- c) For
- d) Nested Loop

Q7. What is the output of the following code block program?

```
int i, j, n=3;
for(i=1; i<=n; i++){
    for(j=i; j>=1; j--){
        printf("*");
    }
    printf("\n");
}
```

- a) *
**

- b) ***
**
*
- c) *****
- d) *
**

Q8. How many times value of i is checked in the following C code?

```
int i=1;
while (i<3)
    i++;
printf("In while loop");
```

- A) 1
- B) 2
- C) 3
- D) 4

Q9. findMax is a function that accepts three double parameters and returns no value. Three double variables num1, num2 and num3, have already been declared and initialized in main. What is the correct statement that calls findMax by sending num1, num2 and num3?

- A) findMax (num1, num2, num3);

- B) findMax (num1; num2; num3);
- C) * findMax (num1, num2, num3);
- D) findMax (num1, 2, 3);

Q10. Which one is the correct call statement of a function my_function which accepts two parameters and returns integer type variable?

- A) ch = my_function();
- B) int my_function ();
- C) int num = my_function (a, b);
- D) int num = my_function ();



2. Gender: Female / Male

3. Class: 1 / 2 / 3 / 4

4. Your department: CMPE / SE / ISE

Part II. System Usability Survey

Choose one of the options (from 1 to 5) about the judgments given below in terms of the usability of the developed application.

		STRONGLY DISAGREE	DISAGREE	NEITHER AGREE NOR DISAGREE	AGREE	STRONGLY AGREE
		1	2	3	4	5
1.	I think that I would like to use this system frequently.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2.	I found the system unnecessarily complex.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.	I thought the system was easy to use.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4.	I think that I would need the support of a technical person to be able to use this system.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5.	I found the various functions in this system were well integrated.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6.	I thought there was too much inconsistency in this system.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7.	I would imagine that most people would learn to use this system very quickly.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8.	I found the system very cumbersome to use.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9.	I felt very confident using the system.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10.	I needed to learn a lot of things before I could get going with this system.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Brooke, J. (1996). SUS-A quick and dirty usability scale. Usability evaluation in industry, 189(194), 4-7.

APPENDIX C

SAMPLE CONSENT FORM AND SYSTEM USABILITY SCALE (SUS) IN TURKISH

“The Effect of Gamification on Learning C Programming” Gönüllü Katılım Formu

Bu çalışma Atılım Üniversitesi Bilgisayar Mühendisliği bölümünde Dr. Öğr. Üy. Damla Topallı danışmanlığında yürütülen “The Effect of Gamification on Learning C Language (Oyunlaştırmanın C Programlama Dilini Öğrenmeye Etkisi)” konulu tez çalışması kapsamında yapılmıştır.

Çalışmanın amacı C programlama dili ile geliştirilen çoktan seçmeli bir sınav uygulaması ile ilgili katılımcılardan geri bildirim almak ve bilgi toplamaktır. Çalışmaya katılım tamamen gönüllülük esasıdır. Uygulanacak ankette sizden kimlik belirleyici hiçbir bilgi istenmemektedir. Cevaplarınız tamamıyla gizli tutulacak ve sadece araştırmacılar tarafından değerlendirilecektir; elde edilecek bilgiler tez ve bilimsel yayımlarda kullanılacaktır. Çalışma Atılım Üniversitesi Bilgisayar Mühendisliği Bölümü laboratuvarlarında yapılacaktır. Her katılım seansı en fazla 30 dakika olarak planlanmıştır.

Çalışma kapsamında yapılacak anket ve mülakatlar genel olarak kişisel rahatsızlık verecek soruları içermemektedir. Ancak, katılım sırasında sorulardan ya da herhangi başka bir nedenden ötürü kendinizi rahatsız hissederseniz çalışmayı yarıda bırakıp çıkmakta serbestsiniz. Böyle bir durumda çalışmayı yürüten sorumlu araştırmacıya, çalışmayı/anketi tamamlamadığınızı söylemek yeterli olacaktır. Anket sonunda, bu çalışmayla ilgili sorularınız cevaplanacaktır. Çalışmamıza katılarak bize destek verdiğiniz için şimdiden çok teşekkür ederiz. Çalışma hakkında daha fazla bilgi almak için Atılım Üniversitesi Bilişim Sistemleri Mühendisliği Bölümü öğretim üyelerinden Dr. Öğr. Üyesi Damla Topallı (damla.topalli@atilim.edu.tr) ile iletişim kurabilirsiniz.

Bu çalışmaya tamamen gönüllü olarak katılıyorum ve istediğim zaman yarıda kesip çıkabileceğimi biliyorum. Verdiğim bilgilerin bilimsel amaçlı yayımlarda kullanılmasını kabul ediyorum. (Formu doldurup imzaladıktan sonra uygulayıcıya geri veriniz).

Öğrenci No: _____ Tarih: _____/_____/2023 Lab: _____ Signature: _____
_____ / _____ / 2023 L3009 / L3013 / L3017

Bölüm I. Demografik Bilgiler

1. Yaş:

2. Cinsiyet: Kadın / Erkek

3. Sınıf: 1 / 2 / 3 / 4

4. Bölüm: CMPE / SE / ISE

Bölüm II. Sistem Kullanılabilirlik Ölçeği - System Usability Scale (SUS)

Çalışma kapsamında geliştirilen uygulamanın kullanılabilirliği (Usability) açısından aşağıda verilen yargılar hakkında verilen seçeneklerden (1'den 5'e kadar) size uygun olan birini seçiniz.

		KESİNLİKLE KATILMIYORUM 1	2 KATILMIYORUM	3 KARARSIZIM	4 KATILYORUM	5 KESİNLİKLE KATILYORUM
1.	Bu sistemi sık sık kullanmak isteyeceğimi düşünüyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2.	Sistemi gereksiz karmaşık buldum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.	Sistemin kullanımının kolay olduğunu düşünüyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4.	Bu sistemi kullanabilmek için teknik bir kişinin desteğine ihtiyacım olacağını düşünüyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5.	Bu sistemdeki çeşitli işlevlerin iyi bir şekilde entegre edildiğini düşünüyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6.	Bu sistemde çok fazla tutarsızlık olduğunu düşünüyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7.	Çoğu insanın bu sistemi kullanmayı çok çabuk öğreneceğini tahmin ediyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8.	Sistemi kullanmayı çok hantal buluyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9.	Sistemi kullanırken kendimi rahat hissediyorum.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10.	Bu sisteme geçmeden önce çok şey öğrenmem gerekiyor.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Brooke, J. (1996). SUS-A quick and dirty usability scale. Usability evaluation in industry, 189(194), 4-7.

APPENDIX D

APPLICATION CODE SNIPPETS

D.1 Page Rendering Function

The following code snippet shows the implementation on frontend for rendering different views in JSX used for React framework.

Description: The `renderPages` function is responsible for conditionally rendering different pages based on the state variables `isStart`, `isQuiz`, and `isFinish`. The rendering depends on the values of these variables and whether the data is still loading (“loading” flag). The function returns different components accordingly, allowing the application to navigate between the start page, quiz page, and finish page.

Function Name: `renderPages`

Details of Function:

- **Start Page:** If `isStart` is true, the function returns the `StartPage` component. The `StartPage` component renders the initial page of the quiz and is triggered when the user is about to start the quiz. It receives a `startQuiz` function as a prop, allowing the user to initiate the quiz.
- **Quiz Page:** If `isQuiz` is true, the function checks the value of the input field with the name `type` using `methods.getValues('type')`. If the value is `'game'`, the function returns a grid layout containing two components:
 1. The main quiz page (`QuizPage`) is placed in the larger section of the grid (Grid item `md={9}` `lg={9}`). It receives props like `endQuiz`, `modifyScore`, and `lang` that enable the user to take the quiz, end it, and modify the score.
 2. The leaderboard (`Leaderboard`) is placed in the smaller section of the grid (Grid item `md={3}` `lg={3}`). The `Leaderboard` component receives the `userScore` prop to display the scores of other users.

If the value of `type` is not `'game'`, only the main quiz page is rendered without the leaderboard, taking up the full width of the grid.

- **Finish Page:** If `isFinish` is true, the function returns the `FinishPage` component. The `FinishPage` component renders the page when the quiz is finished, displaying the user's final score and providing an option to reset the quiz. It receives the `onReset` function as a prop to reset the quiz.

Note: The `'loading'` flag is used to prevent rendering until the data is fully loaded, ensuring a smooth transition between different pages.

The `renderPages` function is an essential part of the application's logic that determines the UI flow based on the user's progress and interactions. By using conditional rendering, different components are displayed based on specific conditions, resulting in a dynamic and interactive user experience.

Code:

```
const renderPages = () => {
  if (!loading) {
    if (isStart) {
      return <StartPage startQuiz={startQuiz} />;
    }
    if (isQuiz) {
      if (methods.getValues('type') === 'game') {
        return (
          <Grid container>
            <Grid item md={9} lg={9}>
              <QuizPage endQuiz={endQuiz} modifyScore={setScore}
                lang={language} />
            </Grid>
            <Grid item md={3} lg={3}>
              <Box sx={{ p: 3 }}>
                <Leaderboard userScore={score} />
              </Box>
            </Grid>
          </Grid>
        );
      } else {
```

```

        return (
            <Grid container>
                <Grid item md={12} lg={12}>
                    <QuizPage endQuiz={endQuiz} modifyScore={setScore}
lang={language} />
                </Grid>
            </Grid>
        );
    }
}
if (isFinish) {
    return <FinishPage onReset={onReset} />;
}
};

```

D.2 Leaderboard View

This code snippet demonstrates the frontend implementation of rendering a leaderboard using components from the Material UI library.

Description: The leaderboard component showcases a list of users and their respective scores in a visually appealing and organized manner.

The Material UI library is utilized to create responsive and customizable UI elements, such as tables, avatars, and typography. Each user's name, avatar, and score are dynamically fetched from the backend API and displayed in the leaderboard table. The table can be sorted based on user scores, allowing users to easily see the top performers.

Function Name: Leaderboard

Details of Function:

- The Box component is utilized as a container, styled with Paper, to provide a bordered layout.

- The List component is used to create a vertical list of leaderboard items.
- The ListSubheader displays a centered typography element as the header for the leaderboard.
- Each leaderboard item (user score) is represented as a ListItem, containing the user's avatar, ID, and score.
- The FontAwesomeIcon component renders an icon based on the user's score, using the renderIcon function from the 'utils' module.

Code:

```

<Box
  component={Paper}
  sx={{
    borderColor: 'primary',
    borderRadius: '7px',
    border: 1,
    padding: 2,
    height: '80vh'
  }}>
  <List
    style={{ maxHeight: '100%', overflow: 'auto' }}
    component={'nav'}
    subheader={
      <ListSubheader component="div">
        <Typography textAlign={'center'} variant={'body2'}>
          LEADERBOARD
        </Typography>
      </ListSubheader>
    }>
    {score?.map((object, index) => (
      <div key={index}>
        <ListItem>
          <ListItemAvatar>
            <Avatar>{index + 1}</Avatar>

```

```

    </ListItemAvatar>
    <ListItemText primary={object.id} secondary={object.sum} />
    <FontAwesomeIcon
      icon={renderIcon(object.sum)}
      size={'lg'}
      style={{ color: 'goldenrod' }}
    />
  </ListItem>
  <Divider />
</div>
  )}
</List>
</Box>

```

D.3 API Connection

This code snippet demonstrates a function used in the frontend to send user responses to the backend API.

Description: The `sendResponse` function utilizes the `axios` library to make an HTTP POST request to the server. It sends user response data, including the user ID, answer, question text, response type, correctness, score, and time taken to complete the question.

Function Name: `sendResponse`

Details of Function: The function is designed to access data from the form fields and question object directly. It operates asynchronously and handles responses through the `axios` promise chain. This includes both success and error responses, which are appropriately managed to display toasts as needed.

Following are the variables used in the function:

- The `apiLink` variable should contain the URL for the API endpoint to receive user responses.
- The `getValues` function retrieves form field values using their names (`'userId'`, `'answer'`, `'type'`).

- The question object contains the current question details, such as the question text and correct answer.
- The config variable holds any additional configuration needed for the axios request (e.g., authorization headers).
- The timeTaken variable represents the time taken by the user to answer the question in milliseconds.
- The toast function is used to display success or error messages to the user after sending the response.

Code:

```
const sendResponse = async () => {
  await axios
    .post(
      apiLink,
      {
        data: {
          user_id: getValues('userId'),
          answer: getValues('answer').replace(/\n/g, ' '),
          question: question.question,
          type: getValues('type'),
          is_correct: getValues('answer') === question.correct ? 'true' : 'false',
          score: getValues('answer') === question.correct ? 10 : 0,
          time_taken: timeTaken / 1000
        }
      },
      config
    )
    .then(() => {
      toast.success('Answer saved!');
    })
    .catch(() => {
      toast.error('Answer could not be saved!');
    });
}
```

};

