DEVELOPING A WEB-BASED PLATFORM THAT WILL AID PROGRAMMERS IN SOLVING PROGRAMMING ERRORS.

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

NDUNG'U ALEX META

C026-01-1294/2019

A project submitted to the department of COMPUTER SCIENCE in the school of COMPUTER SCIENCE AND INFORMATION TECHNOLOGY in partial fulfillment of the requirements for the award of the degree of BACHELOR OF SCIENCE IN COMPUTER SCIENCE of DEDAN KIMATHI UNIVERSITY OF TECHNOLOGY.

DECEMBER 2022

DECLARATION

This proposal is my original work and has not been presented for any other award to the best of my knowledge.

Name: ….....

Signature: ..... Date: .....

This proposal has been submitted for examination with my approval as supervisor.

Name: ….....

Signature: ..... Date: .....

DEDICATION

I dedicate my Final Year Project to my family and friends. Special gratitude to my loving parents, Wilson Meta and Joyce Muniu, whose words of encouragement and push for tenacity ring in my ears.

I also dedicate my Final Year Project to my friends, who supported me throughout the process and gave me ideas on improving the project.

ACKNOWLEDGEMENT

I wish to first thank the Almighty God for the strength and support He has given me throughout the research and development of this project. I would also like to thank my family and friends for their support during this period. May God bless you all.

ABSTRACT

Every programmer spends substantial time fixing bugs throughout the development life cycle. However, it is at times very frustrating to find solutions and could even take weeks to resolve. This study focuses on developing a platform where programmers can easily find the solutions they are looking for.

The functional and non-functional requirements were generated from data collected from the questionnaires, record reviews, and interviews. Agile methodology was used to develop the system since it allows changes far into the development cycle. Figma was the tool used to design the user interface of the website. Due to the enormous amounts of data expected to be processed in the platform, the python programming language was chosen since it is flexible and fast in handling vast amounts of data. For the database, PostgreSQL was selected since it is fast and easy to configure and design.

Testing involved two steps in every sprint in the agile methodology. The first was going through the code with a fine-tooth comb to find any errors made in the sprint process. The second phase involved user response where several users engaged with the system and pointed out the mistakes they observed. Such kinds of platforms should be built to ensure adequate resources are available for programmers to correct these mistakes

Table of Contents

DEVELOPING A WEB-BASED PLATFORM THAT WILL AID PROGRAMMERS IN SOLVING PROGRAMMING ERRORS. 1

BACHELOR OF SCIENCE IN COMPUTER SCIENCE 1

NDUNG'U ALEX META 1

C026-01-1294/2019 1

DECLARATION i

DEDICATION ii

ACKNOWLEDGEMENT iii

ABSTRACT iv

LIST OF FIGURES viii

CHAPTER ONE 1

1 INTRODUCTION 1

1.1 Background 1

1.2 Statement of the problem 3

1.3 Objectives 4

1.3.1 General Objective 4

Specific Objectives 4

1.4 Research Questions 5

1.5 Justification 6

1.6 Scope 6

2 LITERATURE REVIEW 7

2.1 Introduction 7

2.2 Case Studies 7

2.2.1 Case Study 1: Stack Overflow 7

2.2.2 Case Study 2: Quora 8

2.2.3 Case Study 3: Reddit 9

2.2.4 Case Study 4: Code Project 10
2.3 Summary 11
2.4 Research Gap 11
2.5 Proposed Methodology 11
3 METHODOLOGY 12
3.1 Introduction 12
3.2 Fact-Finding Techniques 12
3.2.1 Interviews 12
3.2.2 Record Review and Background Reading 13
3.2.3 Questionnaires 13
3.3 Software Design – Software Development Procedures 14
3.3.1 Planning Stage 14
3.3.2 Analysis Stage 15
3.3.3 Design Stage 15
3.3.4 Development Stage 15
3.3.5 Testing Stage 16
3.3.6 Implementation and Integration Stage 16
3.3.7 Maintenance Stage 16
3.4 Preliminary Data Processing and analysis 17
3.4.1 Editing 17
3.4.2 Coding 17
3.4.3 Data Entry 18
4 ANALYSIS AND DESIGN 19
4.1 Introduction 19
4.2 Requirement Analysis 19
4.2.1 Functional Requirements 19
4.2.2 Non-functional Requirements 22

4.3 Data Analysis 24

4.4 System Analysis 29

4.4.1 Use Case Analysis 29

4.4.2 Class Diagram 29

4.4.3 Data Flow Diagram 30

4.4.4 Context Diagram 31

4.4.5 Sequence Diagram 32

4.4.6 System Flow Chat 33

4.4.7 Entity Relationship Diagram 34

4.4.8 Data Dictionaries 35

4.5 System Design 41

4.5.1 Favicon 41

4.5.2 Navigation 41

4.5.3 Authentication 43

4.5.4 New Questions Page 43

4.5.5 Question Details 45

4.5.6 Posting A Question 50

4.5.7 My Questions 51

5 TESTING AND IMPLEMENTATION 52

5.1 System Implementation 52

5.1.1 Identify The Problem 52

5.1.2 Plan Ahead 52

5.1.3 Solution Design 53

User Experience Design (UX) - 53

User Interface Design (UI) - 53

Figma was the tool used for 53

5.1.3.3.1 Logical Model 53

5.1.3.3.2 Physical Model 54

5.1.4 Programming 54

5.2 Testing 55

5.2.1 Unit Testing 55

5.3 Deployment 60

6 CONCLUSION AND RECOMMENDATION 61

6.1 Discussion 61

Agile Methodology 61

6.2 Limitations 61

6.2.1 Limited Time 61

6.2.2 Skill Set and Experience 61

For this platform to serve all members of the programming community despite their level of experience, the resources to be built into it needed to be top-tier. However, the level of experience needed to achieve this is senior level or higher, presenting a significant challenge in the implementation stage. 61

6.3 Recommendations 62

6.4 Conclusion 62

REFERENCES i

APPEDICES v

QUESTIONARE v

SAMPLE CODE v

LIST OF FIGURES

Figure 1 Agile Cycle 14

Figure 2 Functional Requirements 21

Figure 3 Non Functional Requirements 23

Figure 4 How long have you been programming? 24

Figure 5 Which skill level would you put yourself in 24

Figure 6 How Many Programming Languages Do You use? 25
Figure 7 What has been your experience with programming 25
Figure 8 How Often Do You Face Bugs 26
Figure 9 How Long Do You Take To Debug These Errors 26
Figure 10 What Do You Think Mostly Influence The Time Stated Above 27
Figure 11 Which Tool Do You Use To Find The Bugs Solutions 27
Figure 12 What Has Been Your Experience While Using The Selected Tool 28
Figure 13 Use Case Diagram 29
Figure 14 Class Diagram 30
Figure 15 DFD Diagram 31
Figure 16 Context Diagram 32
Figure 17 Sequence Diagram 33
Figure 18 System Flow Chat 34
Figure 19 Entity Relationship Diagram 35
Figure 20 Data Dictionary Users Table 36
Figure 21 Profile Table 37
Figure 22 Languages Table 37
Figure 23 Frameworks Table 38
Figure 24 Tags Table 39
Figure 25 Questions Table 40
Figure 26 Gallery Table 40
Figure 27 Favicon 41
Figure 28 The Navigation 42
Figure 29 Navigation Dropdown 42
Figure 30 Navigation Active 42
Figure 31 New Questions Page 44
Figure 32 New Questions Page Active 45
Figure 33 User in the Question Detail Page 45
Figure 34 The structure of the question 46
Figure 35 Current user response space 47
Figure 36 Submit Answer 48

Figure 37 An answer from a random user 49
Figure 38 Post A Question 50
Figure 39 My Questions 51
Figure 40 Development Process 52
Figure 41 Database Structure 54
Figure 42 Choose API 55
Figure 43 Google API 56
Figure 44 Post-Question Test 57
Figure 45 Post Success 57
Figure 46 Send Request Popup 58
Figure 47 Request Received 59

CHAPTER ONE

1 INTRODUCTION

1.1 Background

In computer programming and engineering, debugging is a multistep process that involves identifying a problem, isolating the source of the problem, and correcting the problem or determining a way to work around it(Matt Heusser, n.d.). Despite the experience level, ensuring one's code is bug-free takes substantial time. The engineering and design of the code plays a critical factor in either preventing or causing lots of stress and force in the development life cycle(O'Dell, n.d.). Also, the more lines of code there are, the higher the chances of spending more time debugging.

Debugging their code can get tricky and frustrating for beginners since they have yet to grasp the rules behind their programming language. Despite there being tools like *Stack Overflow* and *Code Project* to help them debug their code, it still takes time because they must first understand how these tools work.

Most of the time, these tools are not welcoming for unaccustomed users. Since they have been there long, they have been advanced to suit the high demand. As a result, they become too much for inexperienced users even if there are available tutorials online. This implies that they will not use the power of these tools to their fullest potential.

Posting questions in these tools does not necessarily imply that they will be answered. This is because there are millions of questions being asked simultaneously. Moreover, there is no assurance that you will get the exact answer you are looking for after you ask your question. In other circumstances, users on these platforms who consider themselves experts ignore questions they judge as simple.

Platforms like *Stack Overflow* have a reputation system that favors the experts and destroys the beginners. Newcomers who ask questions that seem simple or ignorant are voted down harshly. This negates the purpose for which these platforms were built. Other platforms like *Quora* and *Reddit* have shifted from being question-and-answer forums to social media-like platforms. As a result, so much irrelevant data is floating in their databases hence frustrating the process of answer finding.

Many users go into these platforms to pour out their anger. There are so many cases of cyberbullying and social critics. This destroys serious users who genuinely want to find answers. In addition, it becomes difficult to interact genuinely with these platforms due to the fear of critics.

Locally, most programmers do not even bother asking questions on these platforms when faced with errors. Instead, they prefer scrolling through samples of answers related to their question in hopes of getting a suitable solution. This slow and tedious method slows their project's development life cycle. Other developers prefer to ask for help from colleagues. This is relatively effective but only if, by chance, these colleagues are more experienced in that language or happen to know the answer.

1.2 Statement of the problem

A software bug is an error, flaw, or fault in computer software's design, development, or operation that causes it to produce an incorrect or unexpected result or behave in unintended ways(*What Are Software Bugs? | TotalView*, n.d.). There is no way to know how long debugging takes; it could be minutes, hours, or even days.

Although tools like Stack Overflow and Code Project are available to help programmers debug their code, it is still an excruciating process. There is no assurance that you will get a response if you ask questions in these forums. Furthermore, you might get answers, but they are irrelevant.

This tool will provide a platform where all kinds of developers, despite their level of experience, can interact and get answers to their problems. Unlike other platforms, this tool will have a live chat module which will significantly increase the chances for these developers to get the answers they might be looking for.

1.3 Objectives

1.3.1 General Objective

To provide a platform for programmers to ask and answer questions concerning programming with services that will enhance one-to-one engagement.

Specific Objectives

(i) To evaluate how the structure and composition of a programming question affect how effectively it is answered.

(ii) To evaluate how a programming language's syntax and rules affect how a question in that language is structured.

(iii) To evaluate how interactive services like chats improve engagement in the programming community.

(iv) To investigate how to share screen complements gaps left by other interactive services.

(v) To evaluate how effective online payment services are and how often users unlock premium services in online platforms.

1.4 Research Questions

(i) How does the structure and composition of a programming question affect how effectively it is answered?

(ii) How do a programming language's syntax and rules affect how a question in that language is structured?

(iii) How do interactive services like chats improve engagement in the programming community?

(iv) How would a share screen module complement the gaps left by other interactive services?

(v) How practical are online payment services, and how often do users unlock premium services on online platforms?

1.5 Justification

Software development is a considerable factor affecting the growth and development of the economy in the world today. This implies that the smoother the software development process is, the more the economy grows. Bugs are inevitable factors that hinder the rapid release of software(*Reasons Why There Are Software Bugs/Defects in Software*, n.d.). This tool will aid developers in finding solutions to the bugs they face. Adding the live chat and screen share will further assist those with questions to find precise answers from other developers.

1.6 Scope

This study mainly focuses on software developers. Despite the level of experience, bugs are a universal factor that hinders the rapid delivery of software from these developers. Furthermore, beginners will significantly benefit from this study since they can interact with more experienced programmers.

2 LITERATURE REVIEW

2.1 Introduction

There are a couple of questions-and-answers platforms already available. This study will consider *Stack Overflow, Code Project, Reddit,* and *Quora*.

2.2 Case Studies

2.2.1 Case Study 1: Stack Overflow

Stack Overflow is a question-and-answer website for professional and enthusiast programmers(Sachs Jason, n.d.). Developers use tags referring to technologies to label questions to support searching and filtering. Furthermore, it has machine learning models like the random forest to classify questions further depending on the tags from different users.

It has a reputation system in which users earn points when they either answer questions correctly or ask questions requiring lots of research. This system plays a crucial role in creating job opportunities for them. The downside to this is that the reverse also happens. If a user asks a question that can be referred to as easy, they are voted down. It could even escalate to the point where their account is closed.

Their ask-questions template offers many options, from code blocks to bold writings and even italics. However, these options are all cramped up in one section; hence this page is considered somewhat intimidating. The best approach would have been to separate these options to make it easy to use and interact with. Their view answer template is detailed and well-designed; understanding the questions is amazingly easy.

2.2.2 Case Study 2: Quora

Quora is a community-based questions and answers website and app(Montti Roger, n.d.). Every content on the site is generated by users, meaning it is created, edited, and organized by the same people who use the website. It allows users to create social networks and follow topics that interest them. Questions can only be asked and answered by people with an account already created.

Quora's ask questions section is more of a pop-up rather than a whole template. This means that the style in which you can pose your questions is greatly limited. For instance, there is no way to add a code block as you would in *Stack Overflow*. Also, compared to *Stack Overflow,* where you can only ask questions, here you can create a post that is more of a social media platform feature.

The space feature enables you to create communities and curate collections around specific topics of interest. Here, users can discuss topics they commonly like, creating a sense of community. Another feature of Quora is the voting option. If a user sees a response that they think perfectly answers the question presented, they can upvote it. On the other hand, if they meet an answer that is inaccurate and uninformative, they can downvote it.

The number of upvotes and downvotes on a particular response will affect its visibility. If a response has many upvotes, it is seen as valid and Quora will push it to the top of the response list. If an answer has more downvotes, users see it as invalid and Quora will not prioritize it for others to see.

2.2.3 Case Study 3: Reddit

Reddit is a vast network of communities created, run, and populated by you, the Reddit users(*Content Policy - Reddit*, n.d.). Also, it is a platform where users can ask and answer questions in the form of posts. Unlike *Stack Overflow,* which is all about programming, Reddit incorporates all kinds of questions, even the unprofessional caliber.

It has a fantastic design compared to *Stack Overflow* which is cramped. Everything is typically straightforward when you log in. The create post page has many items you could add to your post from images, videos, and even code snippets. A user can either select markdown mode or fancy pants mode depending on how much information your post or question will contain. However, this many options could be frustrating for a beginner since there is no preview option.

Using Reddit would be very frustrating for a programmer since most users are after social media services and entertainment. This sort of question could even get voted down, which would bury it in tones and tones of data; hence, it would not get answered. The only solution would be to join or create a coding community in which all the content would be about a particular programming language.

2.2.4 Case Study 4: Code Project

Code Project is a community for computer programmers with articles on different topics and programming languages(*CodeProject. A Guide. - CodeProject*, n.d.). Here, programmers can ask and answer programming questions related to any language. They can read articles related to any language and topic they choose. Also, they can start or join discussions related to programming.

The home page after logging in is majorly made up of articles. However, it is cramped and brutal to distinguish precisely what you want. Users can filter the articles they wish to read by clicking on the keyword they are interested in. This is barely effective since it drops all the data on you for you to travel through the pages and find what you want. The search bar is at the top right, which is too tiny to use.

Asking a question template is relatively easy to use. It has a section where you type the question followed by what you have tried, which is an excellent approach compared to the rest of the case studies. However, like Stack Overflow, a user has all the options crumped together in one tiny space. Unlike Reddit, here you can preview your question and make the relevant changes. Also, a user can create a discussion if that question requires a forum to find the solution. Unfortunately, the discussion page is poorly designed; hence very difficult to follow through with the conversation.

2.3 Summary

All of our case studies are perfect examples of question-and-answer forums. However, there are also significant differences between them. Both *Code Project* and *Stack Overflow* are majorly dedicated to software developers, while *Quora* and *Reddit* are more of social media platforms. They also have a couple of similarities, especially regarding the design of the adding questions section.

The interface designs of Stack Overflow and Code Project have many flaws. For example, both of this platform's home pages are cramped with data that is difficult to interpret, especially for beginners. *Quora* and *Reddit* on the other hand are shifting to be more like social media platforms; hence it is getting harder and harder to get answers from them.

2.4 Research Gap

All these case studies bring an idea of community and discussion. However, there is only so much help beyond this. The one who has asked the question must scroll through all the answers to find the correct answer.

2.5 Proposed Methodology

To close this gap, a live chat system accompanied by a screen share service would ease finding the correct answer. Programmers would not need to scroll through so many questions; they would only have to get clarification directly from those who have supplied the solutions.

3 METHODOLOGY

3.1 Introduction

System methodology structures, plans, and controls the process of developing an information system. It improves the system development productivity and quality by improving the management and control of the software development process. This chapter entails the fact-finding techniques and the development life cycle involved in system development.

3.2 Fact-Finding Techniques

Fact-finding techniques are a process of collecting data and information based on techniques that contain a sampling of existing documents, research, observation, questionnaires, interviews, prototyping, and joint requirements planning(*Fact Finding Techniques || Fact Finding Techniques for Requirements*, n.d.). The fact-finding techniques used for the research were interviews, record reviews, and questionnaires. These techniques were used in the early stages of the System Development Life Cycle, including the system analysis phase, design, and post-implementation review.

3.2.1 Interviews

An interview is the most used technique to collect information from face-to-face interviews. The purpose of the interview was to find, verify, and clarify facts, motivate end-users involved, identify requirements, and gather ideas and opinions(*Fact Finding Techniques || Fact Finding Techniques for Requirements*, n.d.). Both newbies and experienced programmers were interviewed to identify how bugs affect their productivity. The interviews mainly took place in universities with many programming students. Also, developers from established companies were interviewed to understand both worlds better.

3.2.2 Record Review and Background Reading

The information related to the system is published in the sources like websites, newspapers, magazines, journals, documents, etc. This record review helps the analyst to get valuable information about the system. Various websites visited included the stack overflow website, code project website, and Quora.

3.2.3 Questionnaires

Questionnaires are helpful fact-finding techniques to collect information from many users. Users fill in the questions the system analyst gives and then give the answers back to the system analyst. Google forms is a free online software that allows users to create questionnaires and send them to other users(*The Beginner's Guide to Google Forms*, n.d.). The only requirement is a google account. This greatly aided data collection since most programmers uses their google account to sign into platforms like GitHub and Stack Overflow.

3.3 Software Design – Software Development Procedures

The system-development life cycle enables developers to transform a newly-developed project into an operational one. Agile methodology was used in the implementation of this system. It is a project management framework that breaks projects down into several dynamic phases, commonly known as sprints.

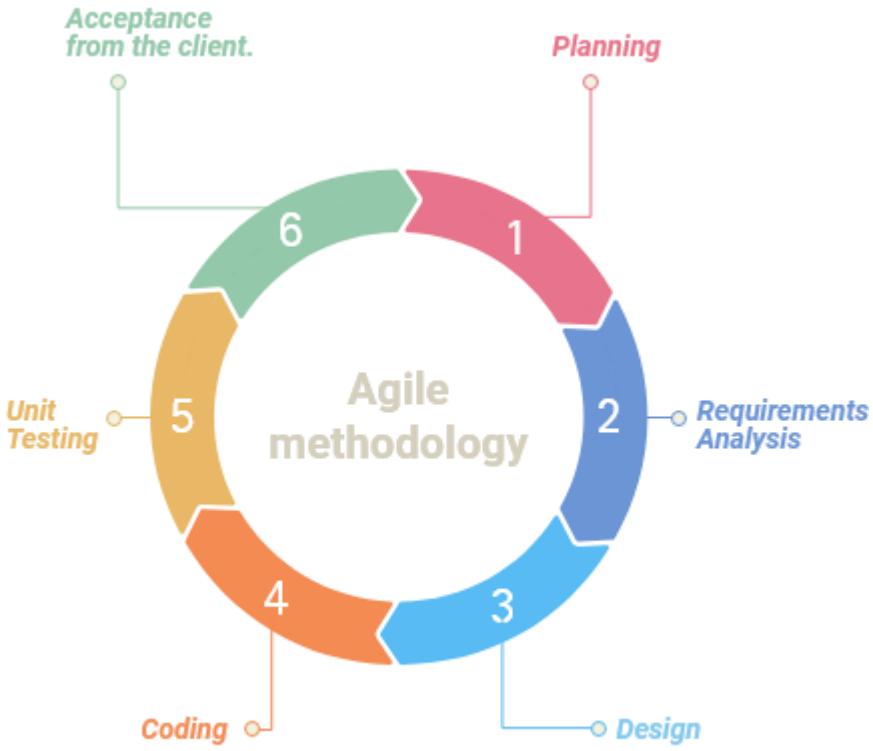

Figure 1 Agile Cycle

3.3.1 Planning Stage

It is the phase in which the developer plans for the upcoming project. It helps to define the problem and scope of any existing systems and determine the objectives for their new designs. By developing an effective outline for the upcoming development cycle, they will theoretically catch problems before they affect development. Fonts and themes used in the design

stage were selected, requirements were outlined, and appropriate tools to accomplish the development were chosen.

3.3.2 Analysis Stage

The analysis stage includes gathering all the specific details required for a new system and determining the first ideas for prototypes.

The developer may:

- Define any prototype system requirements.
- Evaluate alternatives to existing prototypes.
- Perform research and analysis to determine the needs of end-users.

The requirements were analyzed to generate the functional and non-functional requirements.

3.3.3 Design Stage

The design stage is a necessary precursor to the main developer stage. Developers will first outline the details for the overall application alongside specific aspects, such as its:

- User interfaces
- System interfaces
- Network and network requirements
- Databases

Figma was the tool used to design the system's UX and UI, and *inter* was the font applied.

3.3.4 Development Stage

The development stage is where the developer writes code and builds the application according to the earlier design documents and outlined specifications. The product program code is built per the design document specifications. In theory, prior planning and outlining should

make the actual development phase relatively straightforward. Visual Studio Code was the editor used in the coding process. It is built by Microsoft and is compatible with Windows, Linux, and Mac. PostgreSQL, an open-source database with a strong reputation for its reliability, flexibility, and support of open technical standards was used. Django, written in python, was the framework the platform was built on.

3.3.5 Testing Stage

Building software is not the end. Now it is tested to ensure there are no bugs and that the end-user experience will not be negatively affected. During the testing stage, the developer reviews their software with a fine-tooth comb, noting any bugs or defects that need to be tracked, fixed, and retested.

3.3.6 Implementation and Integration Stage

After testing, the overall design for the software will come together. Different modules or designs will be integrated into the primary source code through developer efforts, usually by leveraging training environments to detect other errors or defects. The information system is integrated into its environment and eventually installed. After this stage, the software is theoretically ready for market and may be provided to any end-users.

3.3.7 Maintenance Stage

The SDLC does not end when software reaches the market. The developer now moves into maintenance mode and begins practicing any activities required to handle issues reported by end-users. Furthermore, the developer is responsible for implementing changes the software might need after deployment. This can include handling residual bugs that could not be patched before launch or resolving new issues that crop up due to user reports. Larger systems may require longer maintenance stages compared to smaller systems.

3.4 Preliminary Data Processing and analysis

The objectives of preliminary data analysis are to edit the data to prepare it for further analysis, describe the key features of the data, and summarize the results(*Data Preparation and Preliminary Data Analysis*, n.d.).

3.4.1 Editing

The usual first step in data preparation is to edit the raw data collected through the questionnaire. Editing detects errors and omissions, corrects them where possible, and certifies that minimum data quality standards have been achieved. The purpose of editing is to generate data that is: accurate; consistent with the intent of the question and other information in the survey; uniformly entered; complete; and arranged to simplify coding and tabulation. Since Google Forms was the questionaries used, minimum errors in answering occurred; hence no need for editing.

3.4.2 Coding

Coding involves assigning numbers or other symbols to answers so the responses can be grouped into limited classes or categories. Specifically, coding entails the assignment of numerical values to each response for each question within the survey. Classifying data into limited categories sacrifices some data detail but is necessary for efficient analysis. The google forms response panel automatically transforms the responses into a form that can be statistically analyzed.

3.4.3 Data Entry

Once the questionnaire is coded appropriately, researchers input the data into a statistical software package. This process is called data entry. There are various methods of data entry. Manual data entry or keyboarding remains a mainstay for researchers who need to create a data file immediately and store it in a minimal space on various media. Manual data entry is highly error-prone when complex data is being entered; therefore, verifying the data or at least a portion of it becomes necessary. The advantage of google forms is that data entry is unnecessary since as the responses are coming in, they are automatically entered and classified.

3.4.4 Data Cleaning

Data cleaning focuses on error detection, consistency checks, and the treatment of missing responses. The first step in the data cleaning process is to check each variable for data that are out of the range or otherwise called logically inconsistent data. Such data must be corrected as they can hamper the overall analysis process.

As the name suggests, descriptive statistics describe the data's characteristics and provide an initial analysis of any violations of the assumptions underlying the statistical techniques. It also helps in addressing specific research questions(*Data Preparation and Preliminary Data Analysis*, n.d.). This analysis is essential because many advance statistical tests are sensitive to violations in the data.

The descriptive tests provide clarity to the researchers as to where and how the violations are occurring within the dataset. Descriptive statistics include the mean, standard deviation, range of scores, skewness, and kurtosis. With the use of google forms, these statistics are calculated without the researcher’s involvement, and accurate results are given in the form of pie charts and bar graphs.

4 ANALYSIS AND DESIGN

4.1 Introduction

A system is a group of interacting or interrelated elements that act according to a set of rules to form a unified whole(Bogan Christine, n.d.). A system, surrounded and influenced by its environment, is described by its boundaries, structure, and purpose and expressed in its functioning(Bogan Christine, n.d.). Systems are subjects of the study of systems theory(Bogan Christine, n.d.).

System analysis can be defined as a deep analysis of a part of the structure of a module that has been designed before(Pedamkar, n.d.-b). It is conducted to study a system or its features to identify its objectives. It also is a problem-solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose. System design means making any module or a part of the structure from scratch and building it completely without estimation(Pedamkar, n.d.-b).

4.2 Requirement Analysis

Requirements analysis or requirements engineering is a process used to determine the needs and expectations of a new product(Ramachandran Saranya, 2022). It involves frequent communication with the stakeholders and end-users of the product to define expectations, resolve conflicts, and document all the essential requirements.

4.2.1 Functional Requirements

They indicate what a software system must do and how it must function; they are product features that focus on user needs(Jafari Ladan, 2020). Functional requirements are a part of requirement analysis (also known as requirements engineering), an interdisciplinary field of engineering that concerns the design and maintenance of complex systems(*What Is Functional Requirements? - Definition from WhatIs.Com*, 2022).

Functional requirements describe the desired end function of a system operating within normal parameters to ensure the design is adequate to make the desired product and that the end product reaches its potential for the design to meet user expectations(*What Is Functional Requirements? - Definition from WhatIs.Com*, 2022).

Functional Requirement	Description	Example
Authorization Levels	These functions determine various system access levels and decide who can CRUD (change, read, update, or delete) information.	• Users can create an account but cannot delete the account. • The admin can delete users from the system. • Users can post questions and delete them but only after the consent of the admin. • Users can post answers to the questions asked. • Admin can create languages to be used as tags by users. • Admin can create frameworks associated with these languages.
External interfaces	These functions concern the external interface of systems other than the primary system.	• User should either get authenticated via GitHub API or LinkedIn API. • Payment for the premium feature should be either via PayPal or M-Pesa.

Search/Reporting Requirements	This requirements section will tell you how users can search and retrieve data.	• A robust machine learning search engine should be available for users to find questions and answers. • The admin should be able to search for specific users.
Databases	The elements and formats you should use when defining what data needs storing in a system.	The following will be stored in the database: • Users/Profiles • Questions • Answers • Chats • Payment Transactions • Languages • Frameworks
Transaction corrections, adjustments, and cancellations	These requirements examine every transaction's entry, changing, deleting, canceling, and error checking.	• Users will make payments for the premium feature via PayPal or M-Pesa • Users should be able to cancel the premium payment transaction. • Users should be able to change their payment credentials.

Figure 2 Functional Requirements

4.2.2 Non-functional Requirements

A nonfunctional requirement is an attribute that dictates how a system operates. It makes applications or software run more efficiently and illustrates the system's quality (9 Nonfunctional Requirements Examples | Indeed.Com, n.d.). Nonfunctional requirements differ from functional requirements in the following ways:

- **Mandatory vs Non-Mandatory** - Unlike functional requirements, nonfunctional features are not mandatory for a system to operate. Instead, these features can help differentiate an application from other products on the market (9 Nonfunctional Requirements Examples | Indeed.Com, 2021).
- **Basic operations vs Additional features** - Functional requirements encompass what a system does, while nonfunctional requirements cover how a system completes a task. For example, the functional duty of a camera is to take pictures. The nonfunctional duty is to take pictures with enhanced focus and clarity (9 Nonfunctional Requirements Examples | Indeed.Com, 2021).
- **Intended Purpose vs Customer Expectations -** While functional requirements focus on the purpose of an application, nonfunctional requirements center on the users' expectations, such as the product's performance (9 Nonfunctional Requirements Examples | Indeed.Com, 2021).

Non-Functional Requirements	**Description**	**Examples**
Speed	Speed determines how fast an application responds to commands	• The search engine incorporates machine learning to aid in categorizing the search result. • Data from the database is retrieved via Ajax calls which speeds up the get process.

		• Data is posted via Ajax call, so the page does not refresh after a submission.
Security	Security features are necessary to protect sensitive information in the application.	• The system requires users to create accounts to ask and answer questions. • Authentication is done via GitHub or LinkedIn API since they are well-built and secure.
Portability	Portability means how effectively a system performs in one environment compared to another.	• The website is highly responsive and can be used on different-sized devices.
Compatibility	Highly compatible systems typically function well when other applications run on a device.	• The website loads in all available web search engines.
Usability	Usability refers to the ability to use a particular product	• The website has informative icons to help users navigate the pages easily.
Localization	A localized application has features that match the geographical location of its users	• The website uses English as its prime language since most programmers are conversant with it.

Figure 3 Non Functional Requirements

4.3 Data Analysis

Data analysis is collecting, modeling, and analyzing data to extract insights that support decision-making(Calzon Bernardita, 2022). The following bar graphs and pie charts represent the data obtained from the questionnaires and interviews.

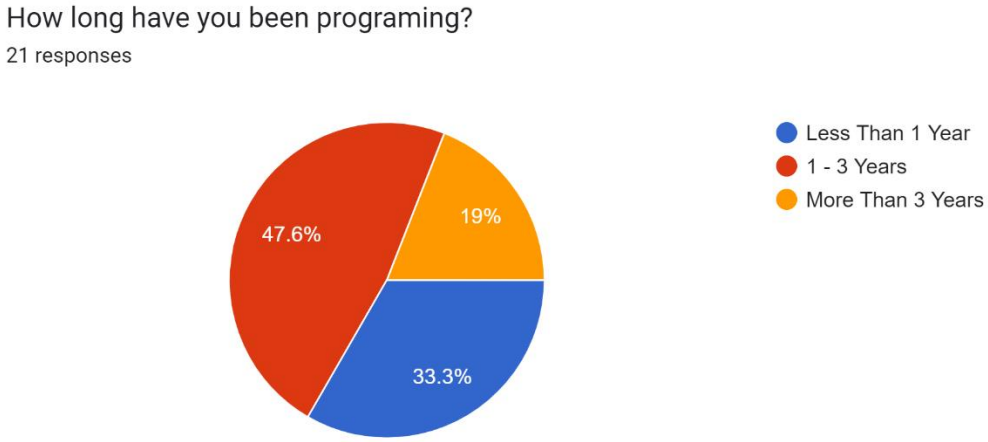

Figure 4 How long have you been programming?

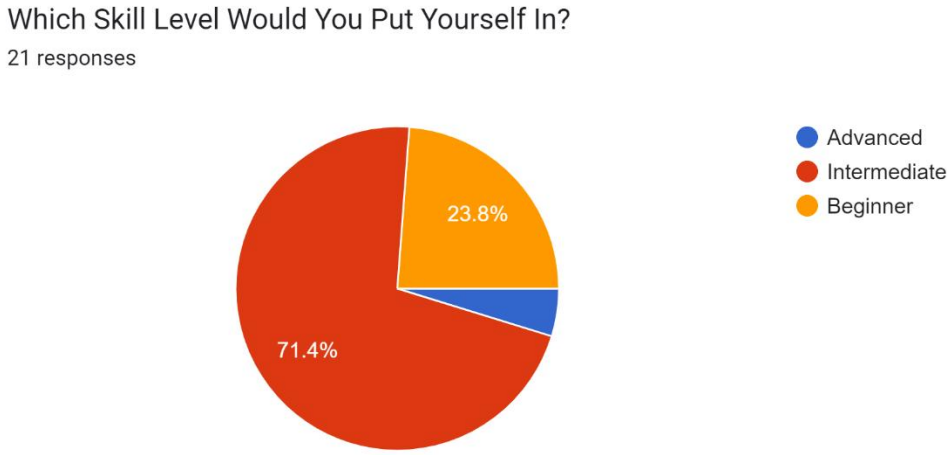

Figure 5 Which skill level would you put yourself in

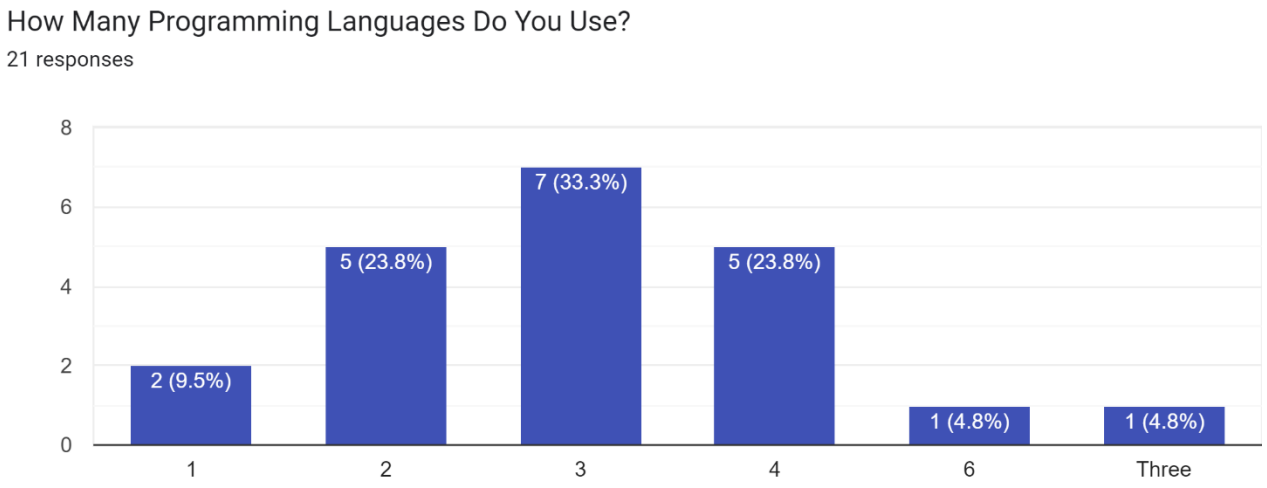

Figure 6 How Many Programming Languages Do You use?

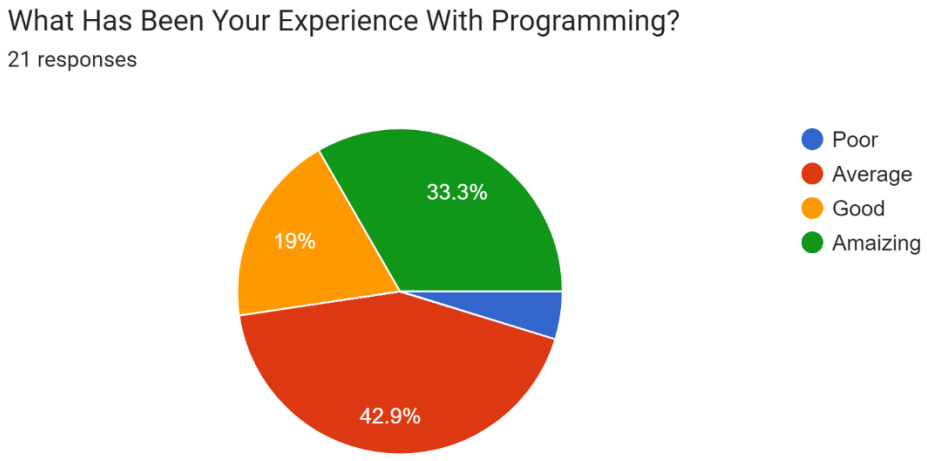

Figure 7 What has been your experience with programming

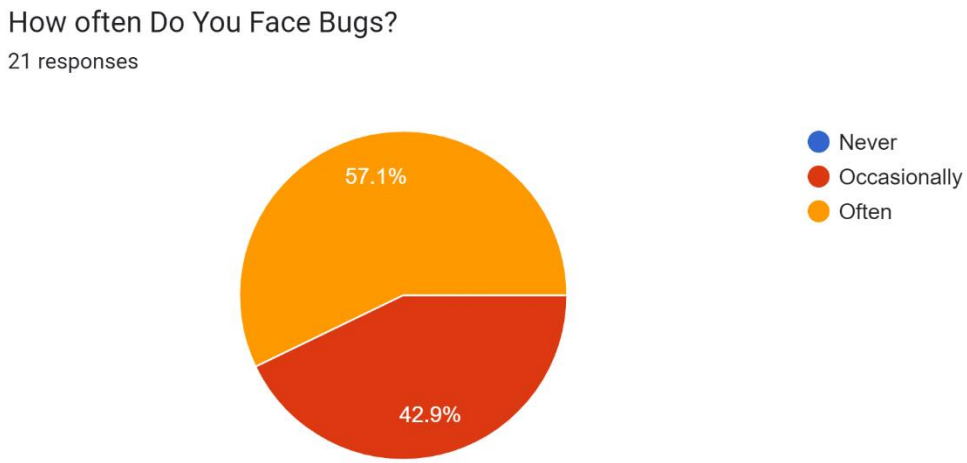

Figure 8 How Often Do You Face Bugs

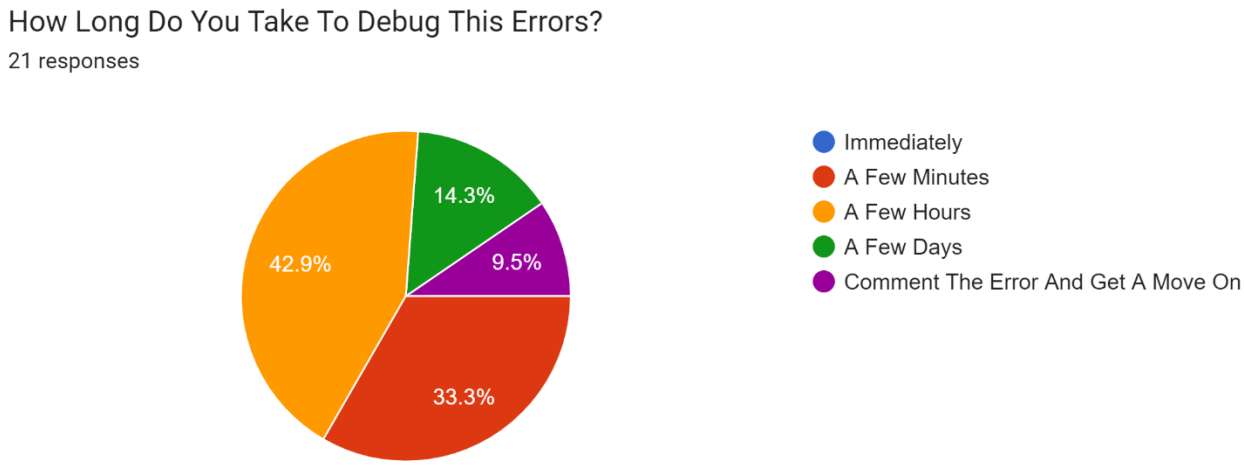

Figure 9 How Long Do You Take To Debug These Errors

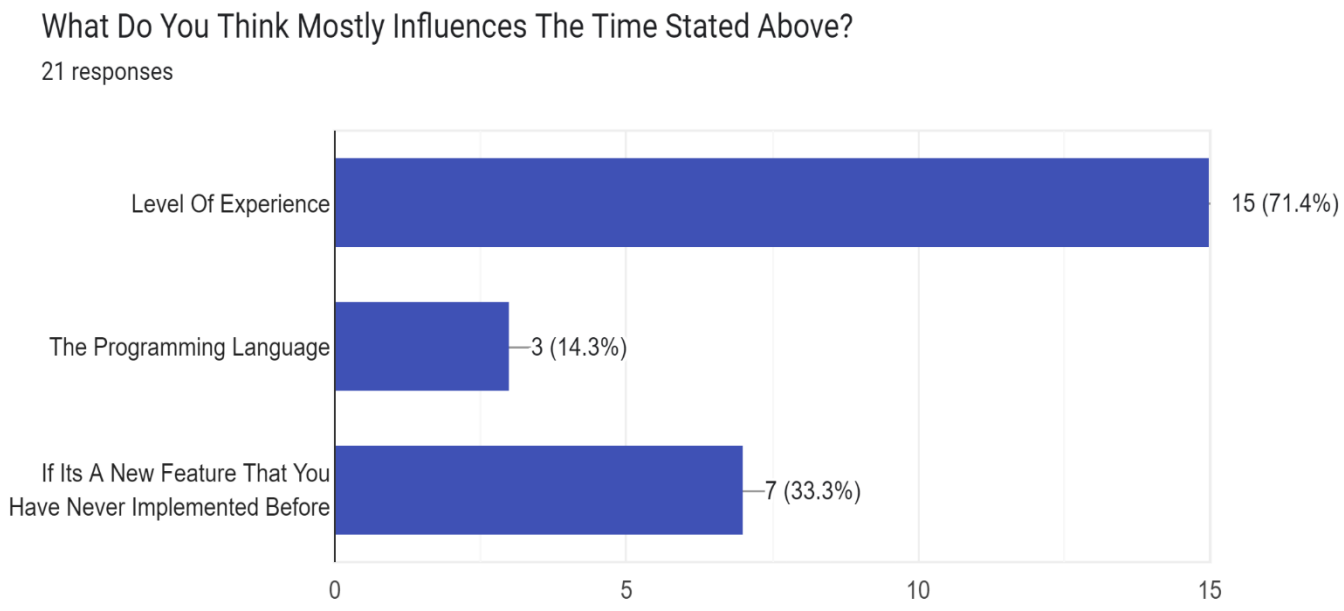

Figure 10 What Do You Think Mostly Influence The Time Stated Above

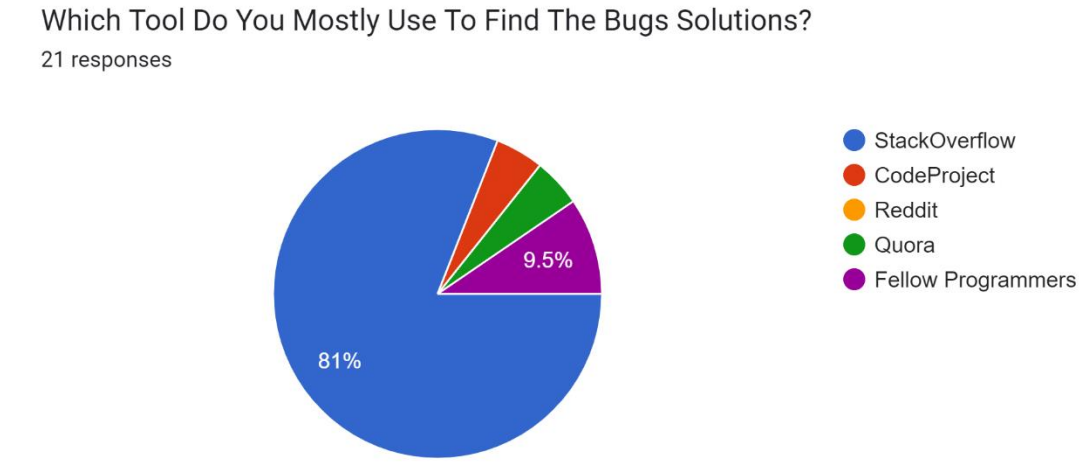

Figure 11 Which Tool Do You Use To Find The Bugs Solutions

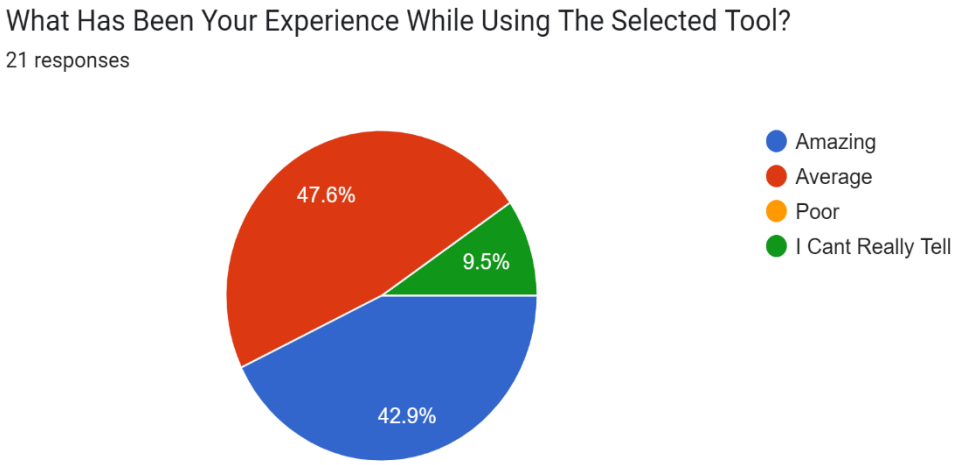

Figure 12 What Has Been Your Experience While Using The Selected Tool

4.4 System Analysis

System analysis can be defined as a deep analysis of a part of the structure of a module that has been designed before(Pedamkar, n.d.-b). It is conducted to study a system or its parts to identify its objectives. It also is a problem-solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

4.4.1 Use Case Analysis

Use case analysis, in simple terms, represents the various ways a software would react based upon the input it receives(*Use Case Analysis: Tutorial & Examples | Study.Com*, n.d.). Consider the example of a person interacting with a social networking site. The various ways in which he or she interacts and the results derived from that interaction could be captured as a use case analysis(*Use Case Analysis: Tutorial & Examples | Study.Com*, n.d.).

Figure 13 Use Case Diagram

4.4.2 Class Diagram

The class diagram is one of the types of UML diagrams used to represent the static diagram by mapping the structure of the systems using classes, attributes, relations, and operations between the various objects(Pedamkar, n.d.-a).

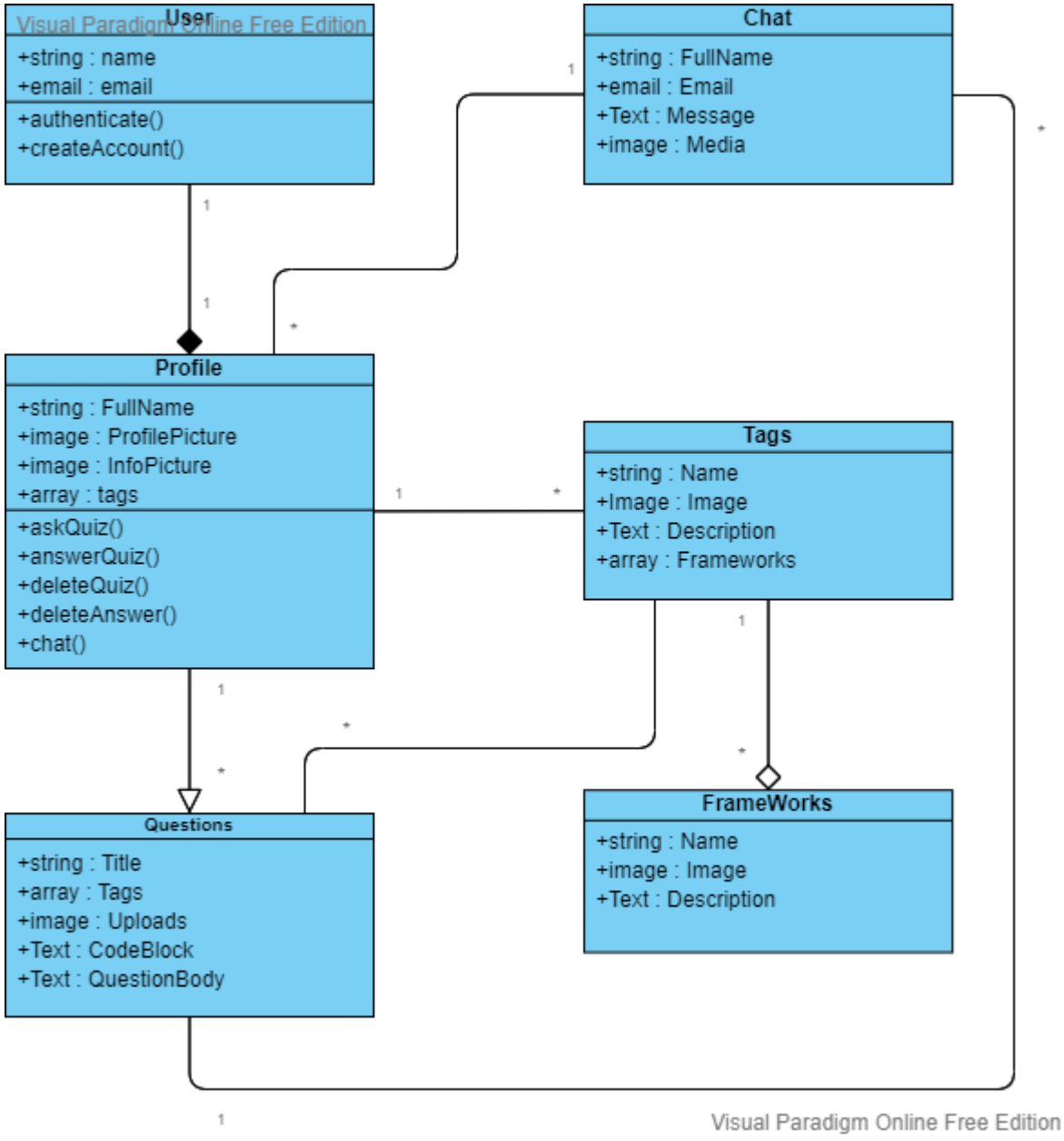

Figure 14 Class Diagram

4.4.3 Data Flow Diagram

A data flow diagram (DFD) maps out the flow of information for any process or system(*What Is a Data Flow Diagram | Lucidchart*, n.d.). It uses defined symbols like rectangles,

circles, and arrows, plus short text labels, to show data inputs, outputs, storage points, and the routes between each destination(*What Is a Data Flow Diagram | Lucidchart*, n.d.).#

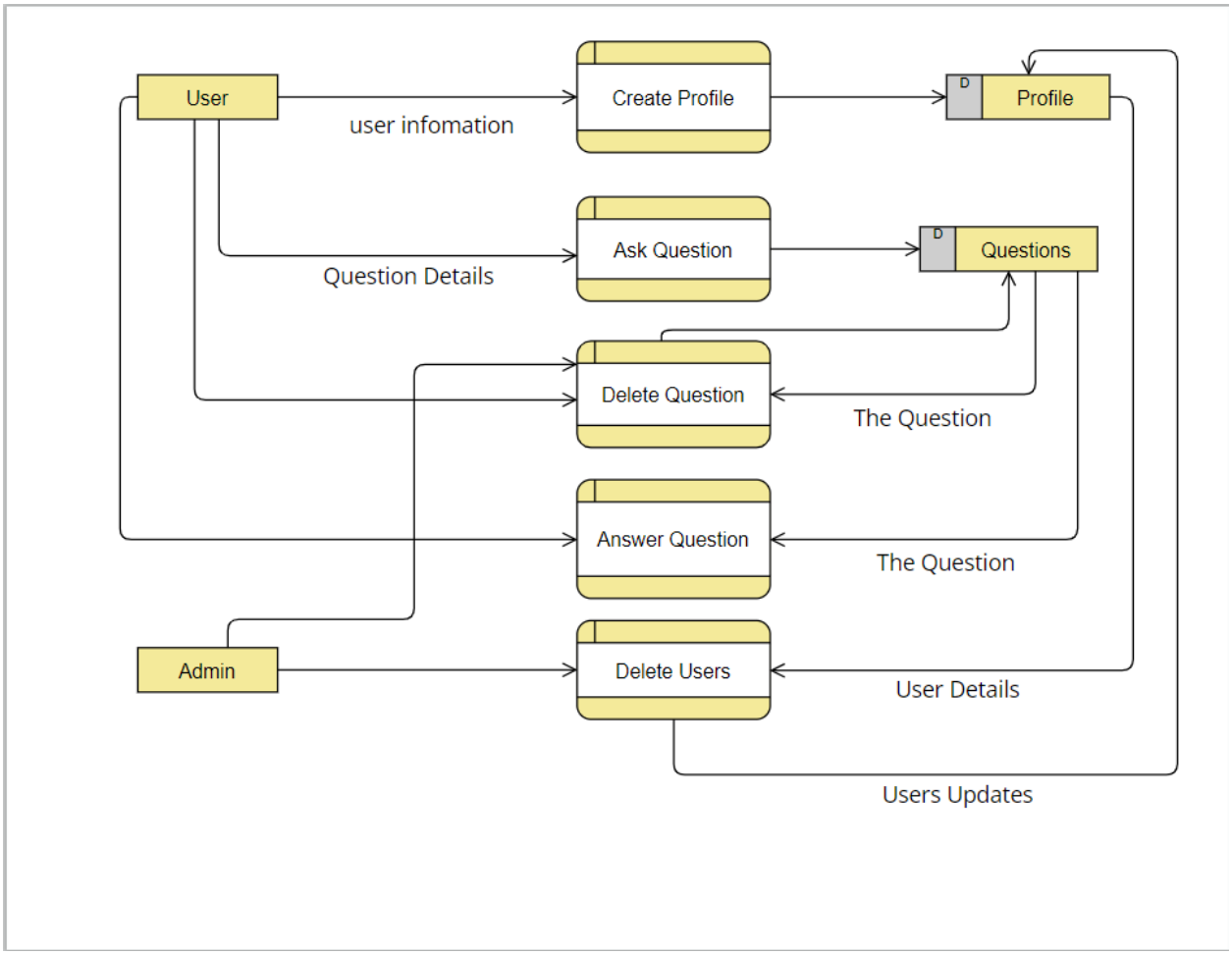

Figure 15 DFD Diagram

4.4.4 Context Diagram

A context diagram visually represents the relationship between data and business processes(Opinaldo Norlyn, n.d.). This diagram has three main components: external entities, system processes, and data flow. It provides the factors and events you need to consider when developing a system(Opinaldo Norlyn, n.d.).

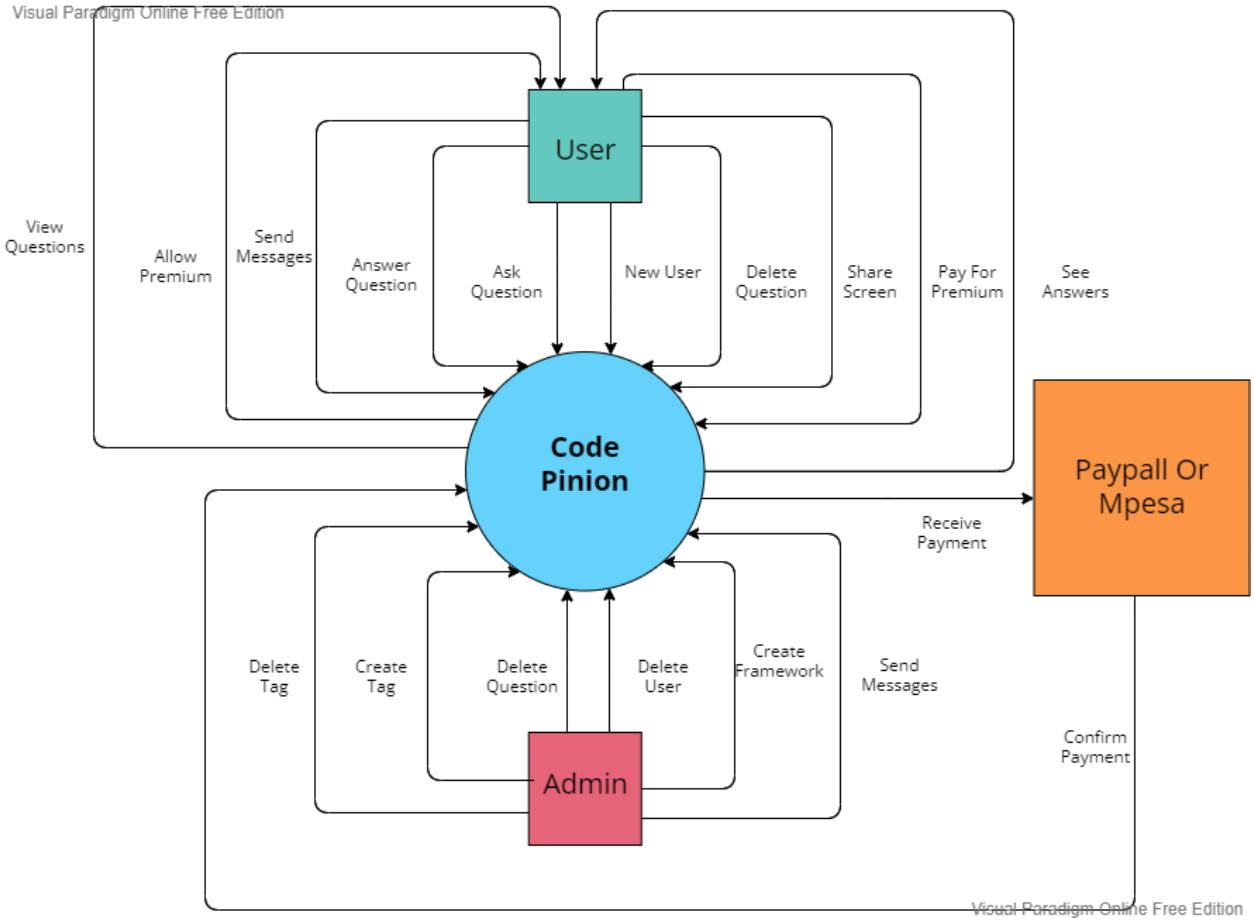

Figure 16 Context Diagram

4.4.5 Sequence Diagram

Sequence diagrams describe class interactions in terms of exchanging messages over time(Sequence Diagrams - What Is a Sequence Diagram? n.d.). They are also *event diagrams*. A sequence diagram is an excellent way to visualize and validate various runtime scenarios. These

can help predict how a system will behave and discover responsibilities a class may need to have in modeling a new system(Sequence Diagrams - What Is a Sequence Diagram? n.d.).

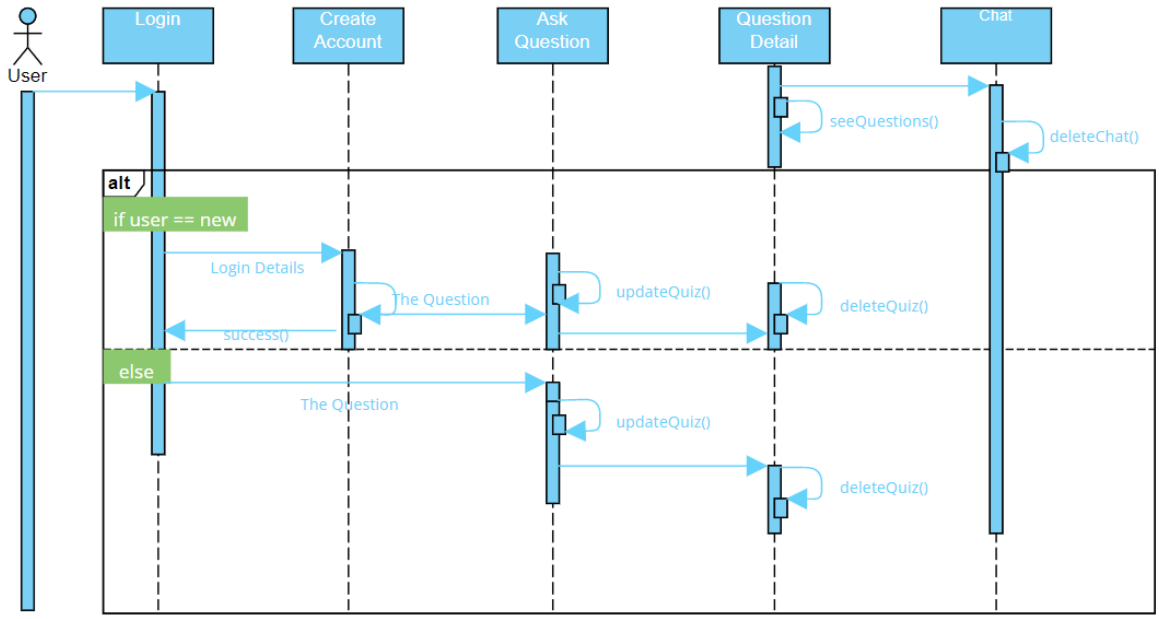

Figure 17 Sequence Diagram

4.4.6 System Flow Chat

A flowchart is a visual representation of the sequence of steps and decisions needed to perform a process(*Flowchart - Process Flow Charts, Templates, How To, and More*, n.d.). Each step in the sequence is noted within a diagram shape. Connecting lines and directional arrows link steps. This allows anyone to view the flowchart and logically follow the process from beginning to end(*Flowchart - Process Flow Charts, Templates, How To, and More*, n.d.).

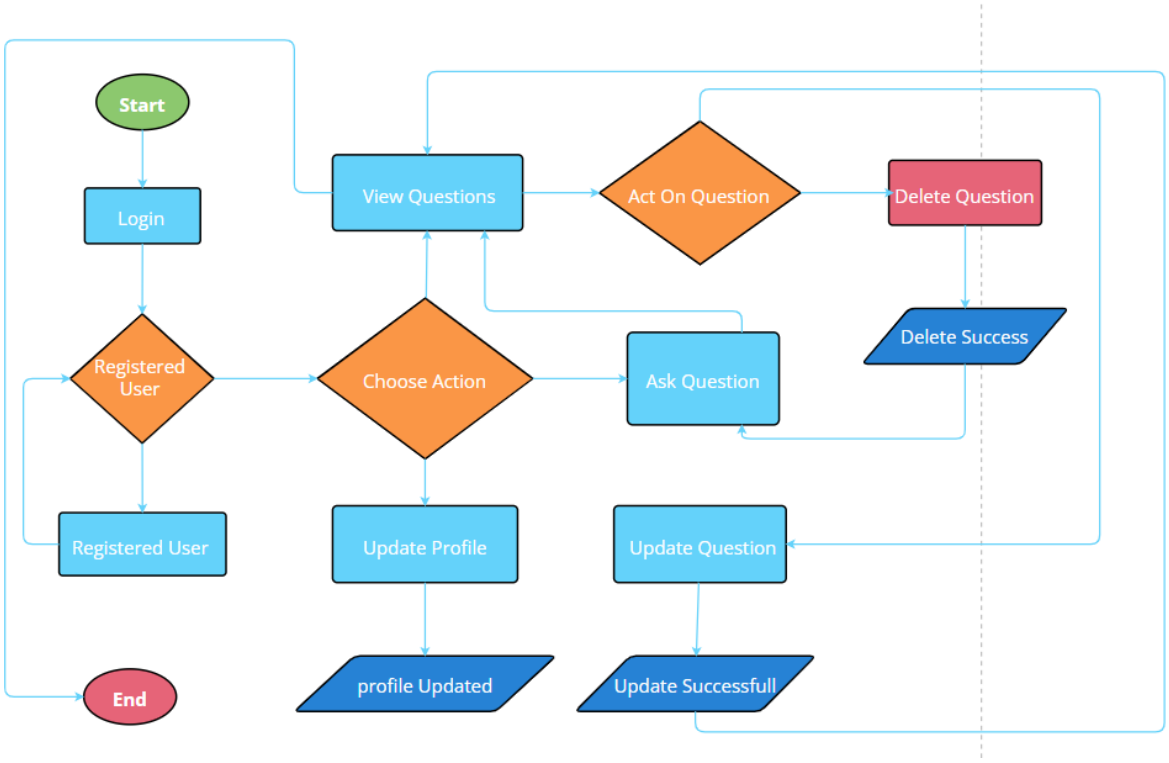

Figure 18 System Flow Chat

4.4.7 Entity Relationship Diagram

An entity relationship diagram is a flowchart that illustrates how entities (people, objects, or concepts) relate to each other inside a system(Roxanna Evan, 2022). To capture an intuitive picture of a system, ER diagrams use symbols such as triangles, rectangles, diamonds, ovals, and lines that display the relationships between entities(Roxanna Evan, 2022). A typical entity diagram mirrors grammatical structure: entities are expressed as nouns, and relationships are portrayed as verbs(Roxanna Evan, 2022).

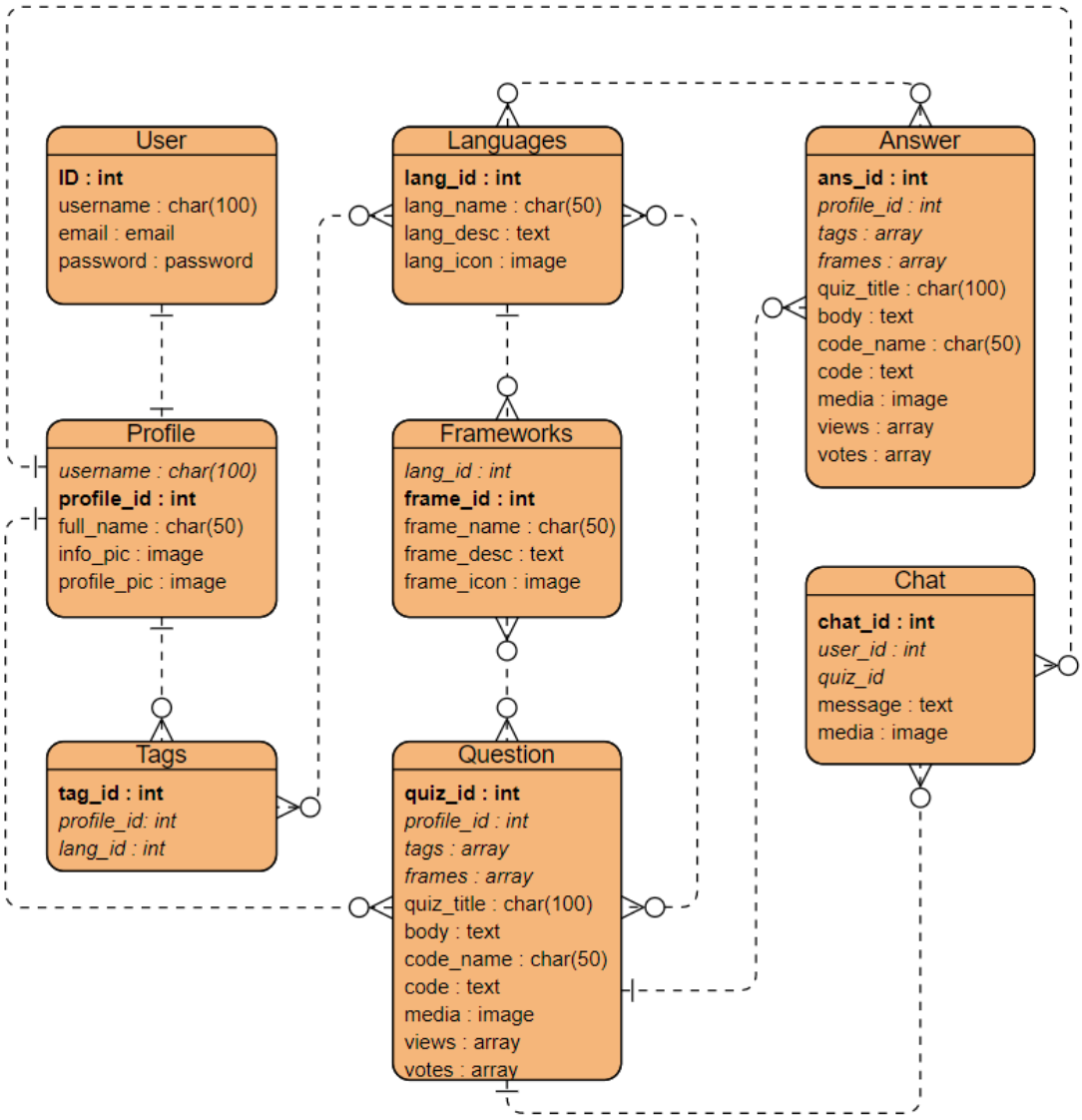

Figure 19 Entity Relationship Diagram

4.4.8 Data Dictionaries

Data Dictionary is a significant component in the structured analysis model of the system(*Data Dictionaries in Software Engineering - GeeksforGeeks*, n.d.). It lists all the data items appearing in DFD. A data dictionary in Software Engineering means a file or a set of files that includes a database's metadata (holds records about other objects in the database), like data ownership, relationships of the data to another object, and some other data(*Data Dictionaries in Software Engineering - GeeksforGeeks*, n.d.)**.**

4.4.8.1 Data Dictionary Users Table

Field Name	Field Size	Data Type	Data Format	Description	Example
user_id	---	int	---	The user id	1001
username	50	char	---	The username of a user	TheMeta
email	50	text	---	The email of the user	alex@gmail.com
superuser	---	boolean	---	If the user has admin rights	yes
staff	---	boolean	---	If the user is a staff or not	no
groups	---	array	---	Which roles the user has	admin
permissions	---	array	---	What the user is allowed to do	Delete users

Figure 20 Data Dictionary Users Table

4.4.8.2 Data Dictionary Profile Table

Field Name	Field Size	Data Type	Data Format	Description	Example
profile_id	---	int	---	The id of the profile owed by a specific user	123
full_name	50	char	---	The full name of the user owning the profile	Alex Meta
profile_pic	---	image	---	The profile image	---

info_pic	---	image	---	Image to illustrate more about the user	---
create	---	date	dd/mm/yyyy	The date the profile was created	01/09/2022
update	---	data	dd/mm/yyyy	Any date the profile is updated	01/09/2022

Figure 21 Profile Table

4.4.8.3 Data Dictionary Language Table

Field Name	**Field Size**	**Data Type**	**Data Format**	**Description**	**Example**
lang_id	---	int	---	The id of the language	12
lang_name	50	char	---	The name of the language	Python
lang_icon	---	image	---	The icon of the language	---
lang_desc	---	text	---	Short description of the language	It is an oop language.
create	---	date	dd/mm/yyyy	The date the language was added to the database	01/09/2022
update	---	date	dd/mm/yyyy	The date any changes were made to the language	01/09/2022

Figure 22 Languages Table

4.4.8.4 Data Dictionary Frameworks table

Field Name	Field Size	Data Type	Data Format	Description	Example
language	50	char	---	The language the framework is built on	Python
frame_id	---	int	---	The id of the framework	12
frame_name	50	char	---	The name of the framework	Django
frame_icon	---	image	---	The icon of the framework	---
frame_desc	---	text	---	Short description of the framework	It is for website development.
create	---	date	dd/mm/yyyy	The date the framework was added to the database	01/09/2022
update	---	date	dd/mm/yyyy	The date any changes were made to the framework	01/09/2022

Figure 23 Frameworks Table

4.4.8.5 Data Dictionary Tags Table

Field Name	Field Size	Data Type	Data Format	Description	Example
tag_id	---	int	---	The id of the tag	23
user	50	char	---	The user owning the tag	TheMeta
Tags	---	array	---	The languages that the user will ask	Python, C++

				questions about	

Figure 24 Tags Table

4.4.8.6 Data Dictionary Questions Table

Field Name	Field Size	Data Type	Data Format	Description	Example
quiz_id	---	int	---	The id of the question	12
profile	50	char	---	The profile asks the question.	TheMehta
Tags	---	array	---	The languages that the question involves	Python, HTML
frameworks	---	array	---	The frameworks the question is about	Django
quiz_title	100	char	---	The title of the question	How to write a function in python.
Body	---	text	---	The details of the questions	---
code_1	50	char	---	The language the code block is about.	Python
codeBlock_1	---	text	---	The actual lines of code	print(“hello”)
quiz_views	---	array/int	---	All the profiles that have viewed the question	40

quiz_votes	---	array/int	---	The number of votes the question got	50
created	---	date	dd/mm/yyyy	The date the question was posted	09/09/2022
update	---	date	dd/mm/yyyy	The date the question was altered	09/09/2022

Figure 25 Questions Table

4.4.8.7 Data Dictionary Gallery Table

Field Name	Field Size	Data Type	Data Format	Description	Example
gallery_id	---	int	---	The id of the gallery of a question	109
question_id	---	int	---	The id of the question owns the gallery	23
body_media	---	image	---	The images involved in asking a particular question	---
update	---	date	dd/mm/yyyy	The date the question was posted	09/09/2022
update	---	date	dd/mm/yyyy	The date the question was altered	09/09/2022

Figure 26 Gallery Table

4.5 System Design

System design is the process of defining the elements of a system such as the architecture, modules, and components, the different interfaces of those components, and the data that goes through that system(*What Is System Design? - Definition from Techopedia*, 2014). It is meant to satisfy a business or organization's specific needs and requirements by engineering a coherent and well-running system(*What Is System Design? - Definition from Techopedia*, 2014).

4.5.1 Favicon

A favicon is a small square image representing your website in web browsers(*What Is a Favicon and Why Does Your Website Needs One? | Upwork*, 2021). Favicons can be composed of a company's logo, initials, or other identifying imagery(*What Is a Favicon and Why Does Your Website Needs One? | Upwork*, 2021).

Figure 27 Favicon

4.5.2 Navigation

Website navigation is the act of clicking and looking through resources on the internet, such as the various pages that make up a website(Website Navigation: Definition, Importance, and Tips | Indeed.Com, 2021). Users navigate websites using a web browser and click on links that transport

them to other pages when clicked(Website Navigation: Definition, Importance, and Tips | Indeed.Com, 2021).

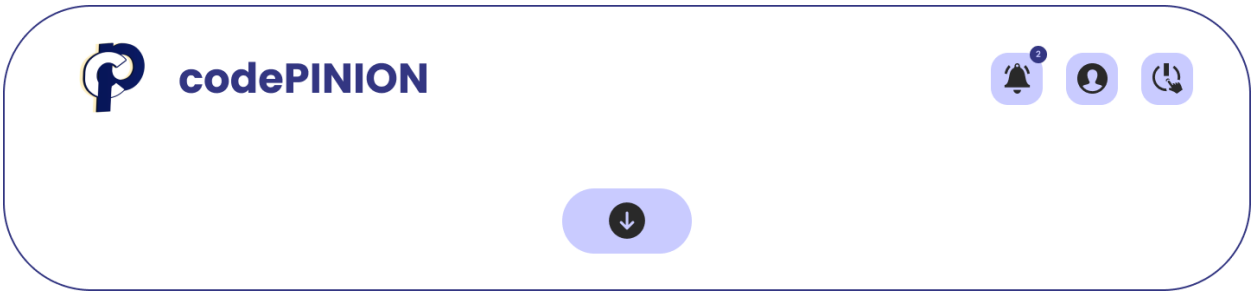

Figure 28 The Navigation

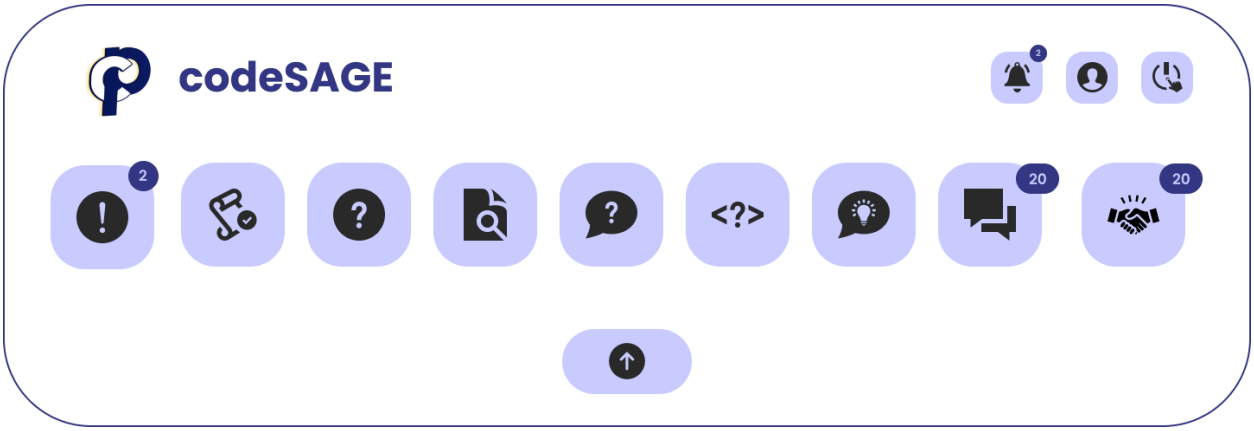

Figure 29 Navigation Dropdown

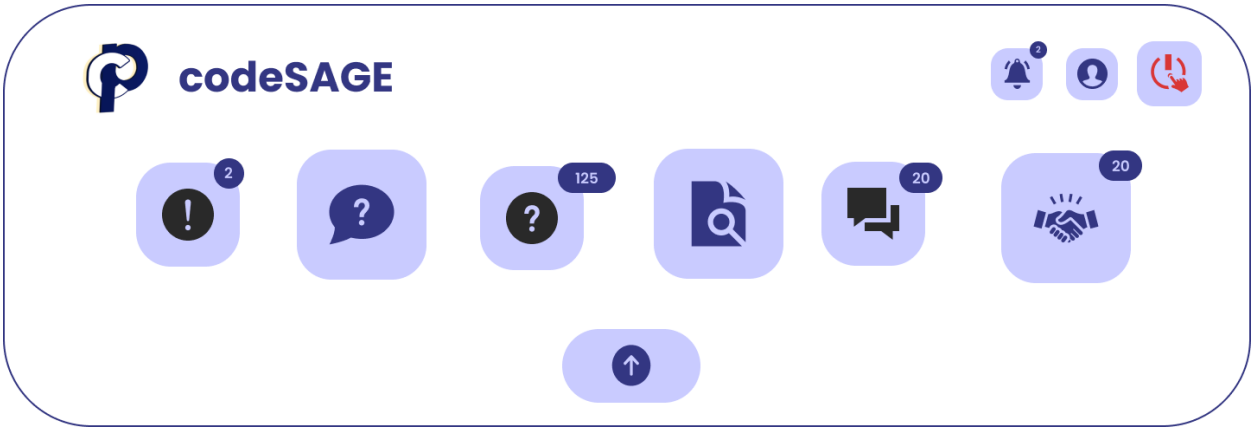

Figure 30 Navigation Active

4.5.3 Authentication

Authentication is recognizing a user's identity(*What Is Authentication? Definition of Authentication, Authentication Meaning - The Economic Times*, 2022). It is the mechanism of associating an incoming request with a set of identifying credentials(*What Is Authentication? Definition of Authentication, Authentication Meaning - The Economic Times*, 2022). The credentials provided are compared to those on a file in a database of the authorized user's information on a local operating system or within an authentication server(*What Is Authentication? Definition of Authentication, Authentication Meaning - The Economic Times*, 2022). Users will either use LinkedIn or GitHub to sign in.

4.5.4 New Questions Page

After a question is posted, it will appear on the new questions page where a user can click on it to view its details and answer it.

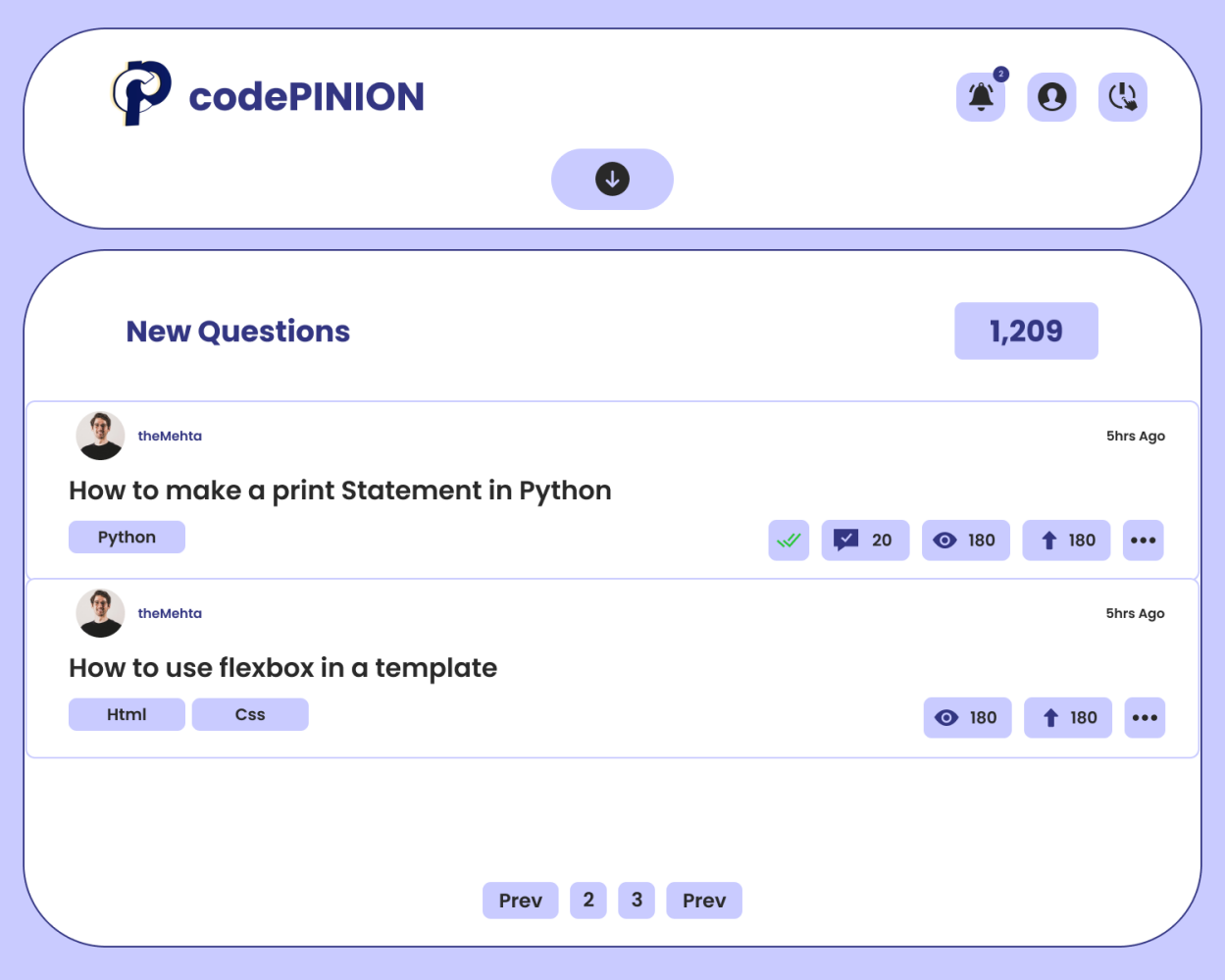

Figure 31 New Questions Page

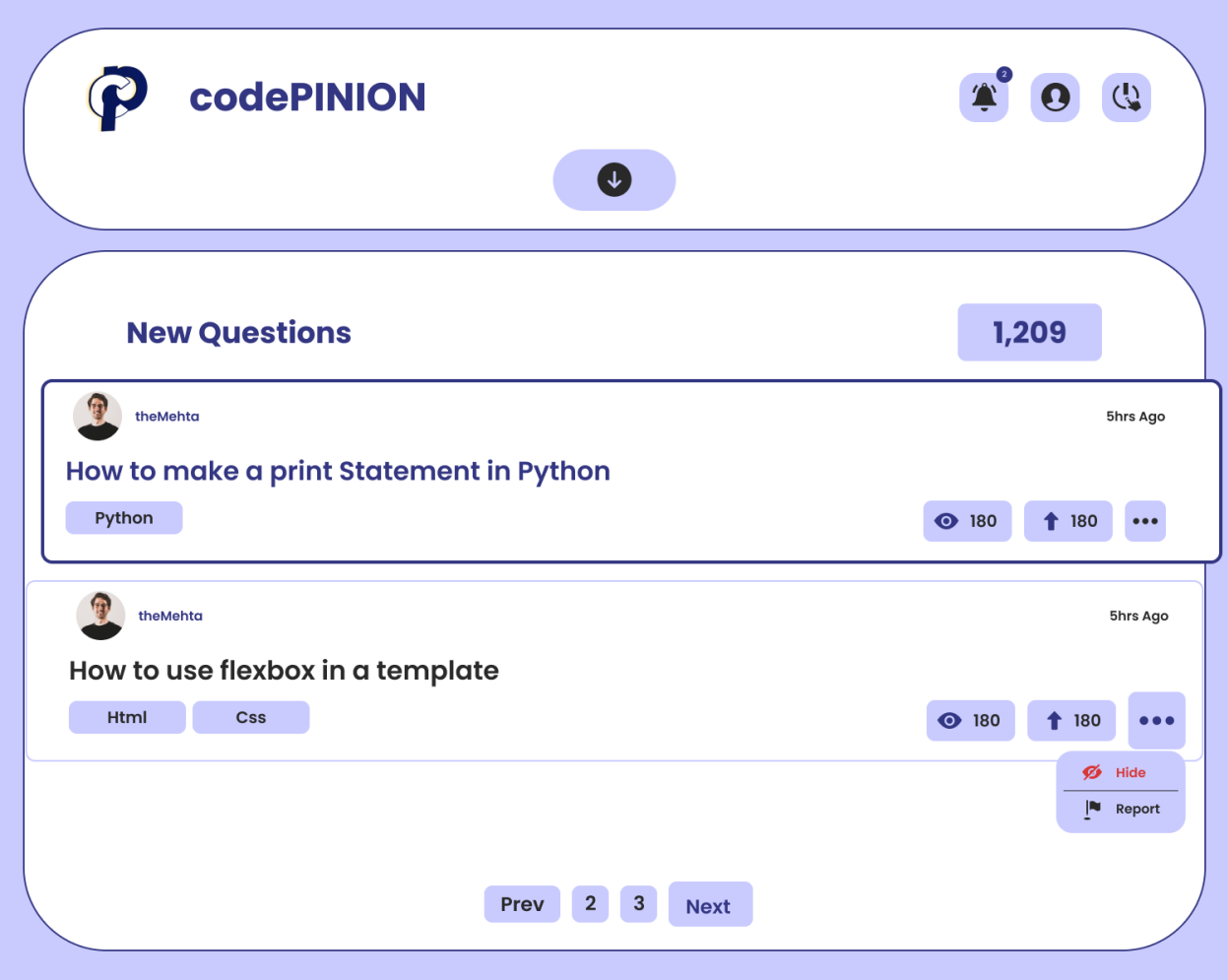

Figure 32 New Questions Page Active

4.5.5 Question Details

After clicking on a random question, the user will be taken to this page to see the full details of the question. Also, they will be able to respond to the questions asked.

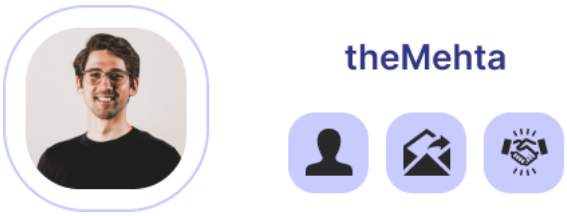

Figure 33 User in the Question Detail Page

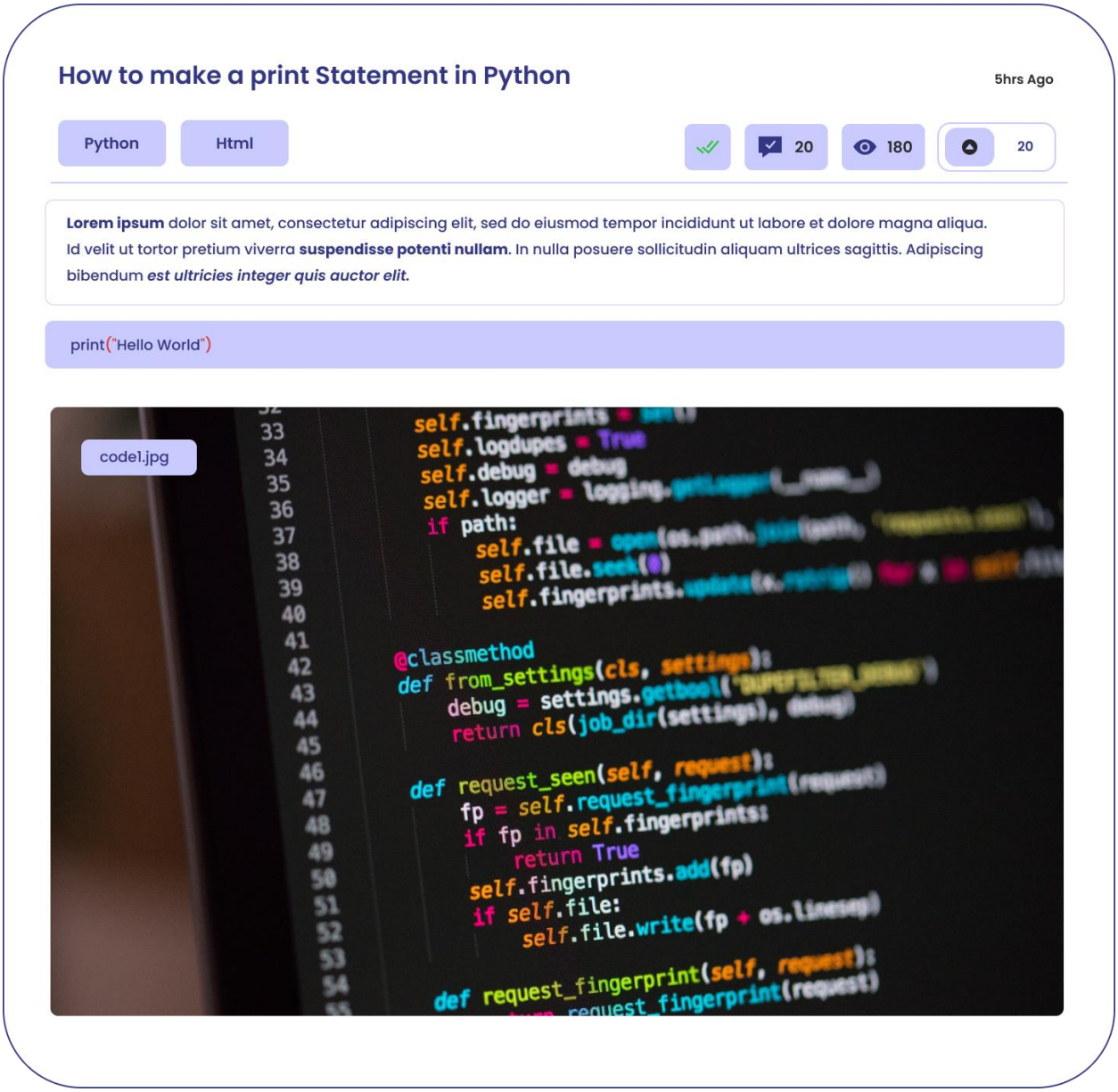

Figure 34 The structure of the question

The current user will respond in the space below.

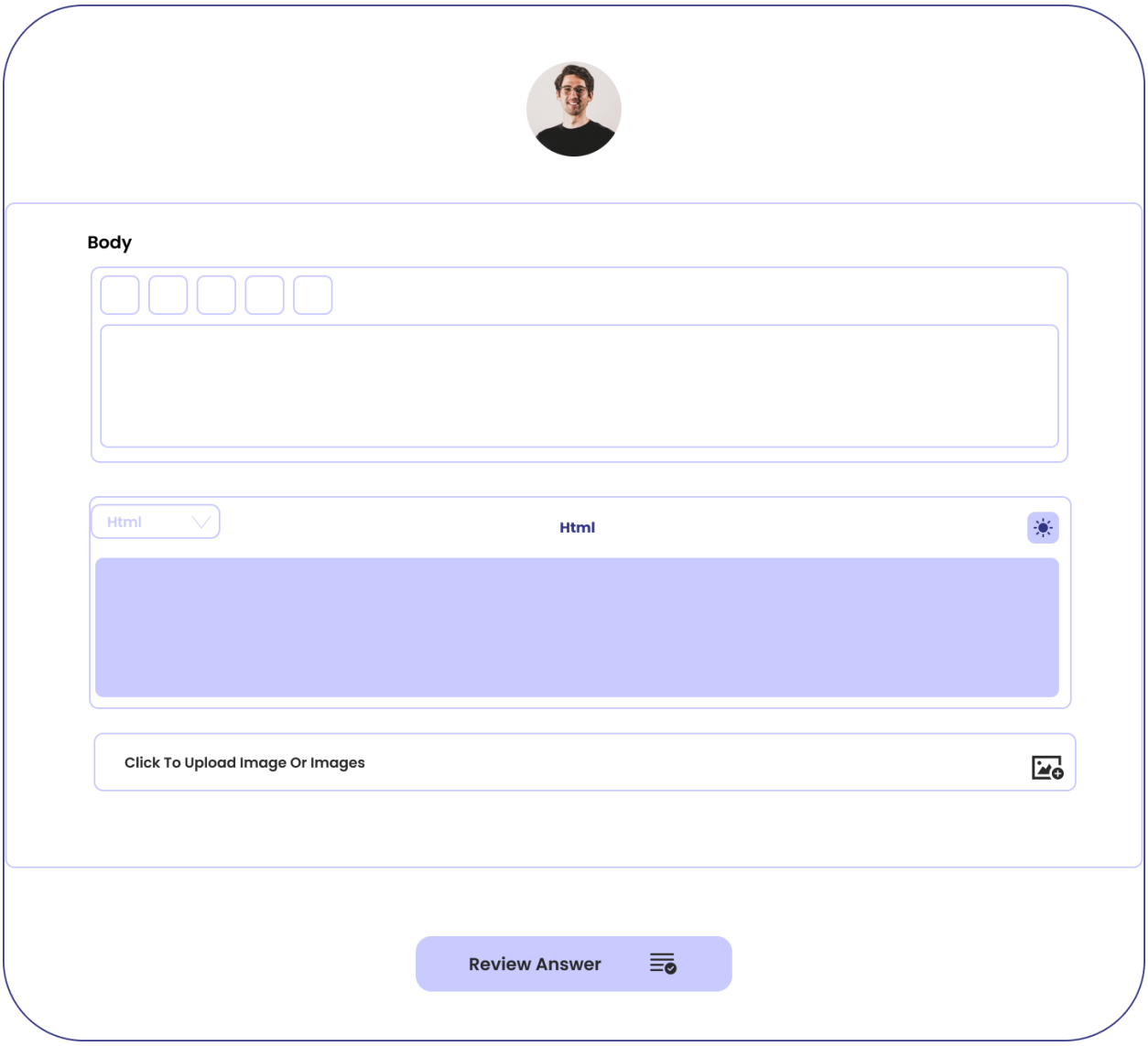

Figure 35 Current user response space

After typing the answer, the user will submit it on this next activity.

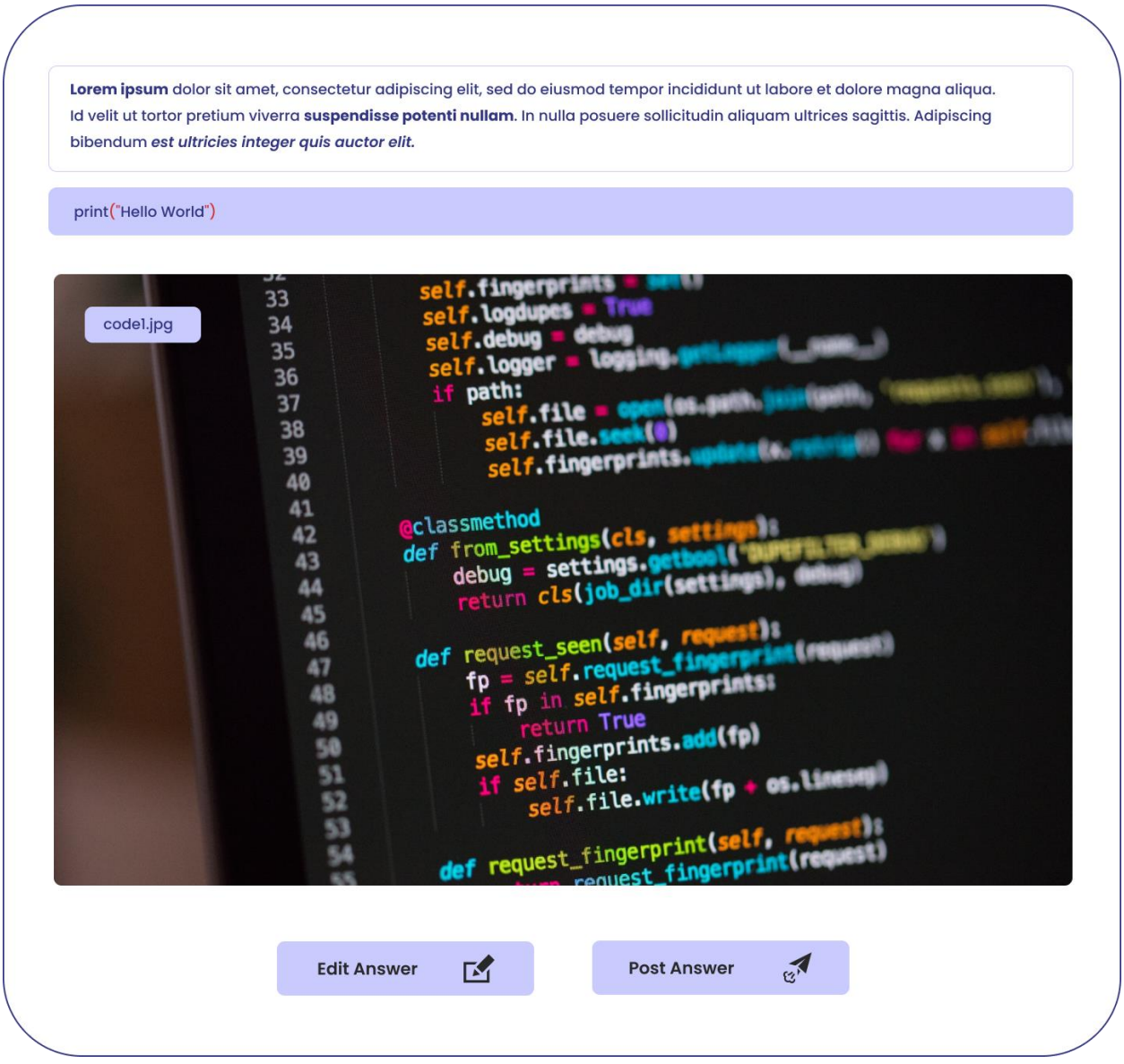

Figure 36 Submit Answer

Below is how the answers from other users will appear on the website.

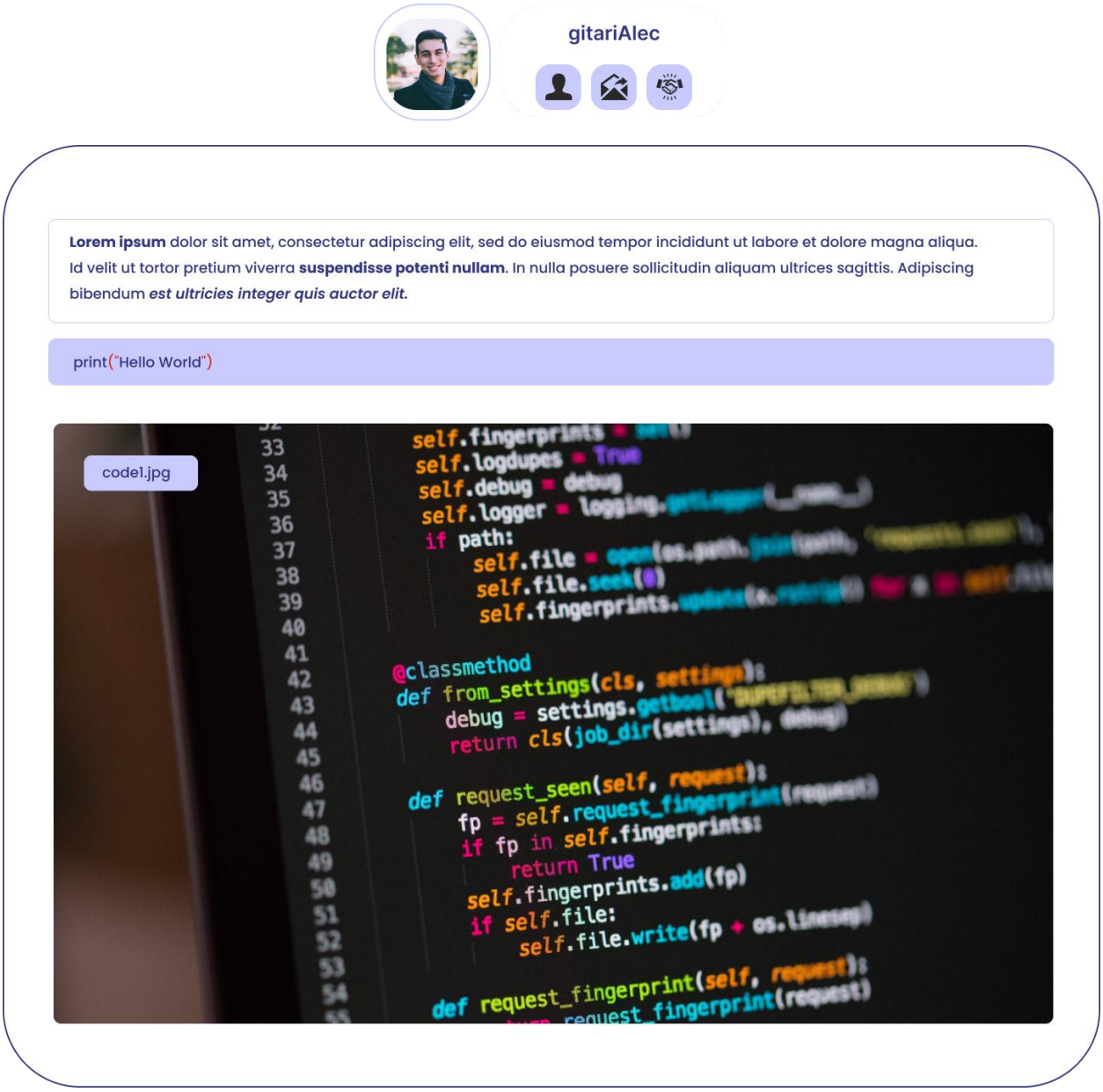

Figure 37 An answer from a random user

4.5.6 Posting A Question

The current user can type their question in the space below.

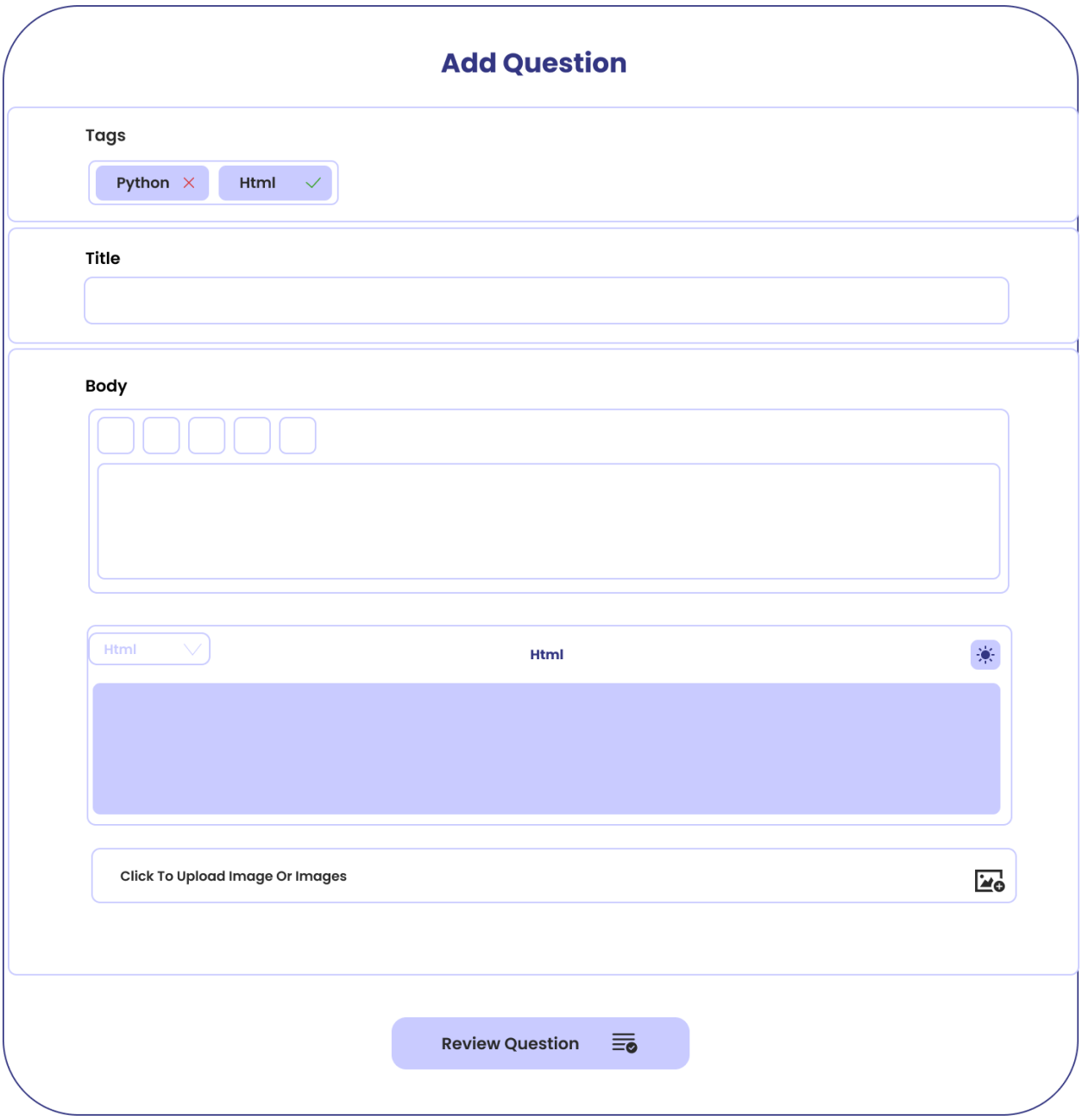

Figure 38 Post A Question

4.5.7 My Questions

Users can see all of the questions they have asked on this platform.

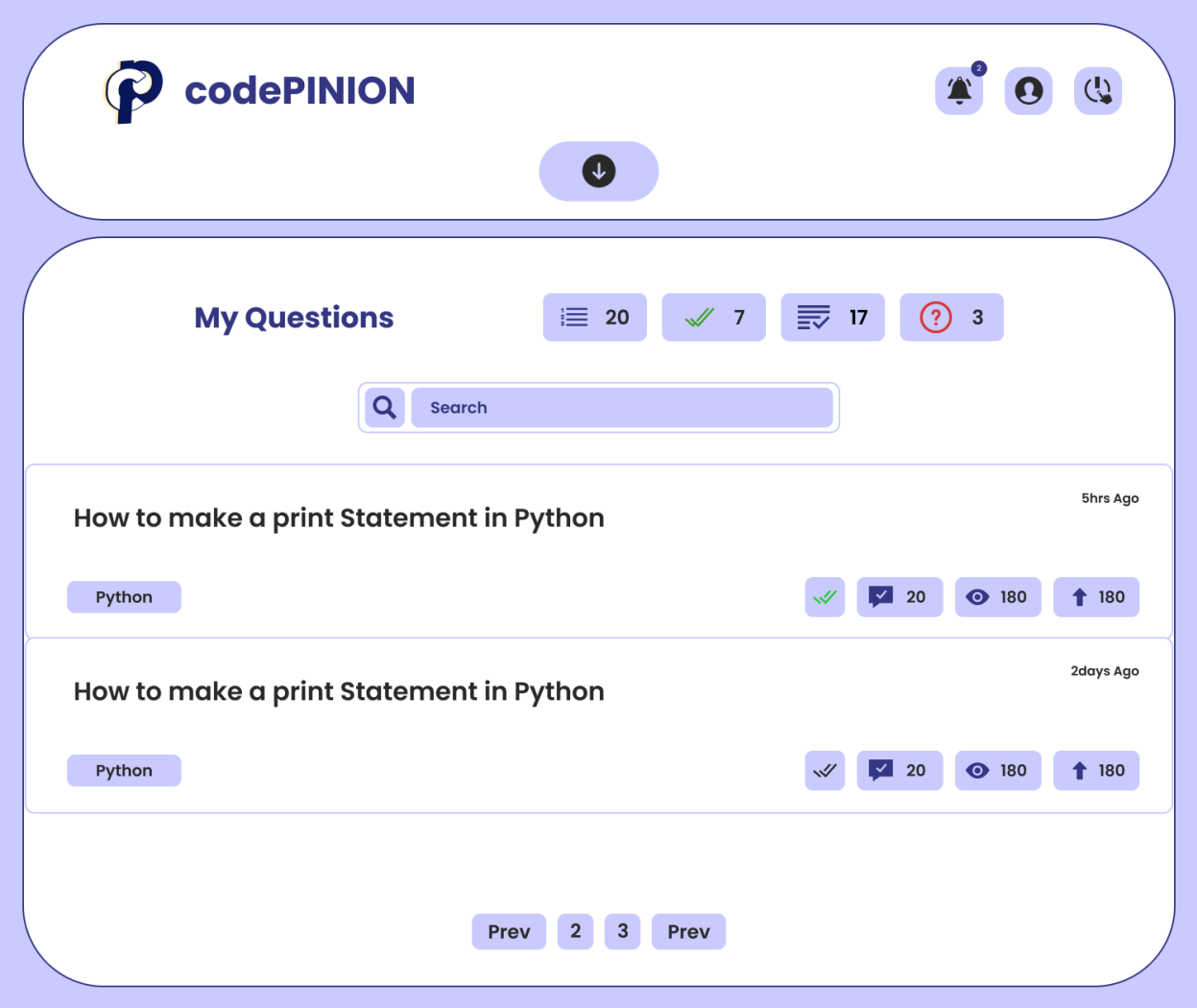

Figure 39 My Questions

5 TESTING AND IMPLEMENTATION

- **System Testing -** verifying the accuracy and completeness of project deliverables before they are released to the customer. It is an essential part of quality assurance and helps to ensure that the final product meets the customer's expectations(*Understanding Project Testing and Its Phases in Project Quality Management*, n.d.-a).
- **System Implementation -** a set of procedures performed to complete the contained in the approved systems design document and to test, install, and begin to use the new or revised Information System(*Introduction to Systems Implementation | Open Textbooks for Hong Kong*, 2016).
- **System Deployment -** Deployment in software and web development means pushing changes or updates from one deployment environment to another(What Is Deployment in Software & Web Development? n.d.).

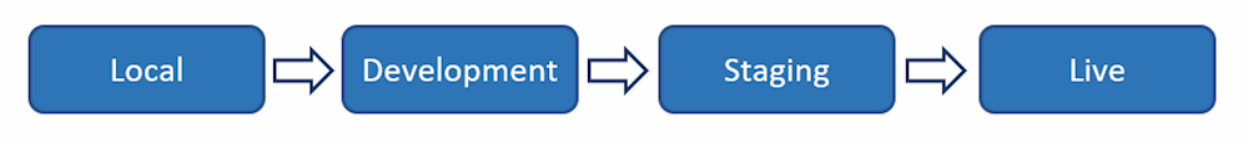

Figure 40 Development Process

5.1 System Implementation

5.1.1 Identify The Problem

The first step in any implementation process is to identify the problem to be solved. To manifest the **Statement of the problem**, thorough **LITERATURE REVIEW** around available question and answer platforms was carried out, and the weaknesses identified formulated the problem statement.

5.1.2 Plan Ahead

After the problem statement is identified, the next step is to plan how the system will be implemented. The following questions were the guide in the planning stage.

a) Which SDLC **METHODOLOGY** is the best for implementation?
b) Which tools are going to be involved in implementation?
c) Which languages are going to be used?

d) How long will the project take?
e) Which are the problems that might be met in the SDLC?

5.1.3 Solution Design

Solution design involves the following steps:

5.1.3.1 System Architecture Design

System Architecture is the model that conceptually defines the system's views, structure, and behavior. In other words, it represents and describes how the system works and communicates with other system components in general(*System Architecture Diagram: A Complete Tutorial | EdrawMax*, n.d.). **System Analysis** was performed to give way for UX/UI and database design.

5.1.3.2 UX/UI Design

- **User Experience Design (UX) -** The process of creating products or services that provide meaningful experiences for users, involving many different areas of product development, including branding, usability, function, and design(*What Is UX Design? Differences Between UX and UI Design | Columbia Engineering Boot Camps*, n.d.).
- **User Interface Design (UI) -** UI design refers to the interfaces users engage with. The UI design process may include buttons or widgets, text, images, sliders, and other interactive elements(*What Is UX Design? Differences Between UX and UI Design | Columbia Engineering Boot Camps*, n.d.).

Figma was the tool used for **System Design**.

5.1.3.3 Database Design

Database design can be generally defined as a collection of tasks or processes that enhance the designing, development, implementation, and maintenance of enterprise data management system(*Database Design - Javatpoint*, n.d.).

5.1.3.3.1 Logical Model

This stage primarily concerns developing a model based on the proposed requirements. The entire model was designed on paper without implementing or adopting DBMS considerations.

5.1.3.3.2 Physical Model

The physical model is concerned with the practices and implementations of the logical model. Django framework provides an inbuilt module called *Models*, where classes that represent tables in a database are coded.

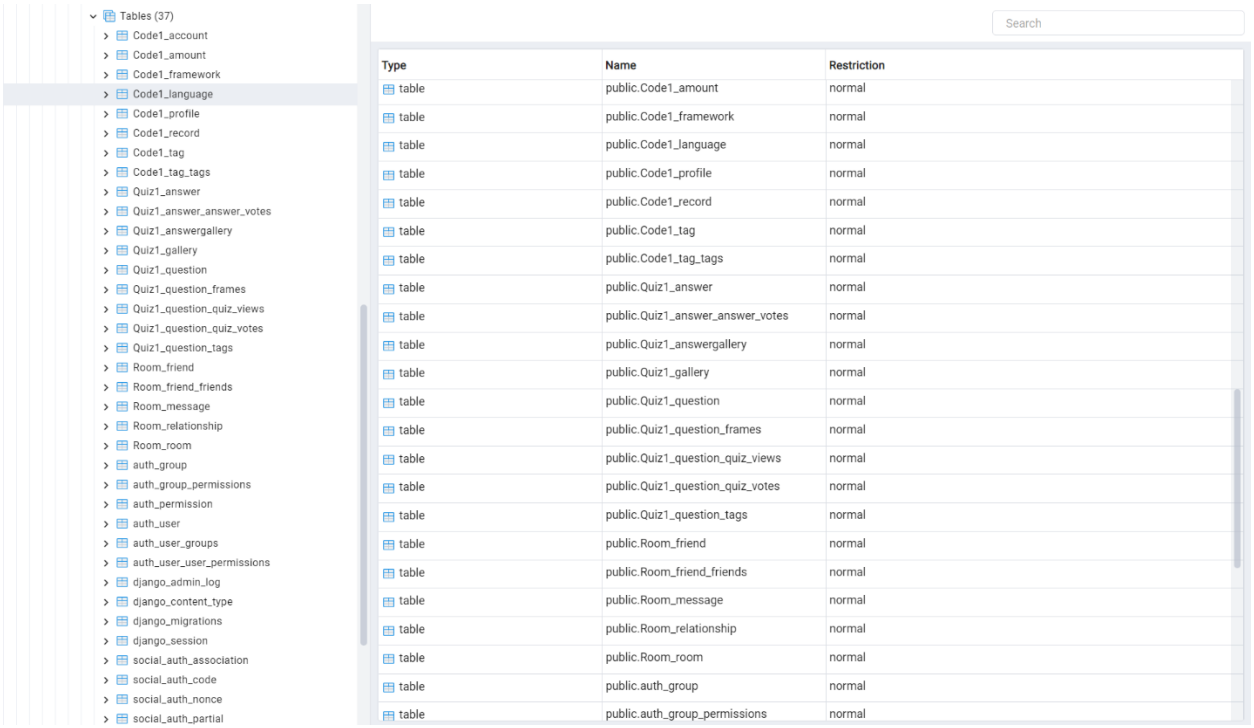

Figure 41 Database Structure

5.1.4 Programming

Programming is the central part of the system development process. Django, written in python programming language, was the framework best suited for this project since python has excellent modules that can manipulate vast amounts of data efficiently and effectively. PostgreSQL was the best choice for the database due to how fluid it interacts with Django.

5.2 Testing

5.2.1 Unit Testing

Unit testing tests individual code units to ensure they are working as intended and functioning correctly(*Understanding Project Testing and Its Phases in Project Quality Management*, n.d.-b).

5.2.1.1 Test Cases

5.2.1.1.1 Test Case 1

Purpose: Authentication

Prerequisites: GitHub Account, Google Account, LinkedIn Account

Procedure: A user clicks a link that redirects them via an API to an authentication page of the provider they have clicked on.

Result: If the credentials are correct, the user gets redirected back to the platform as authenticated.

Verdict: Authentication Success

Failure: The user gets a pop-up message on the platform they are getting authenticated if they enter the wrong email or password

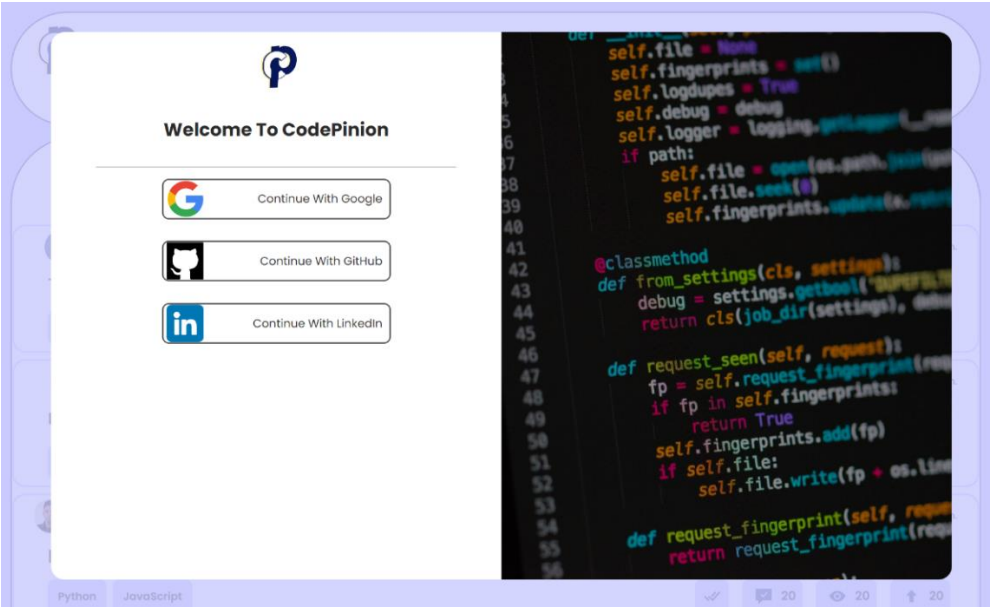

Figure 42 Choose API

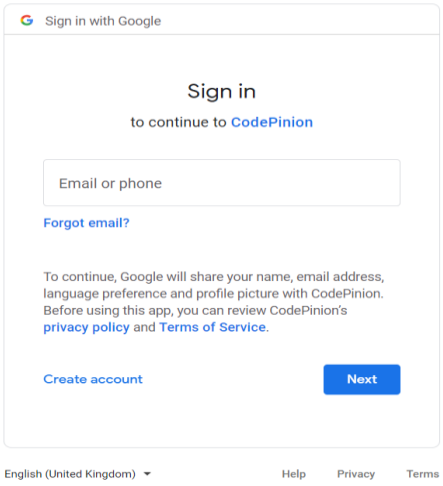

Figure 43 Google API

5.2.1.1.2 Test Case 2

Purpose: Posting A Question

Prerequisites: Tags, Frameworks, Question Title, Question Body, Code Snippets, Images.

Procedure: A user selects the Tags and frameworks involved in the question. They then fill in the title and body and write snippets where the error occurs or the question involves them; they drag-drops they may have and click the post button.

Result: A success message pops up, and this new question is queried in the next ajax call.

Verdict: The question was Posted Successfully.

Failure: The error occurs if a user tries to post a question without a title.

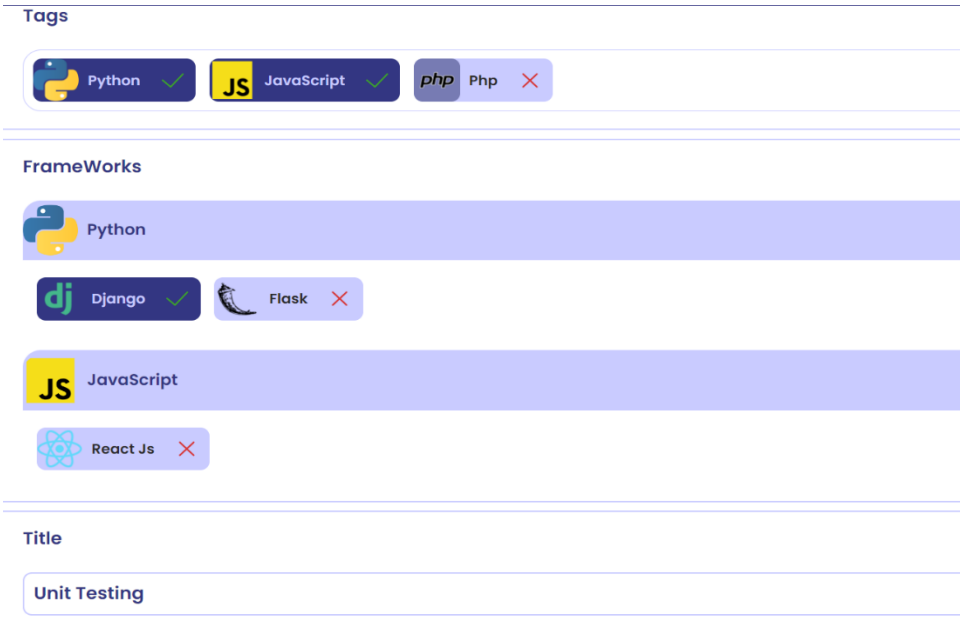

Figure 44 Post-Question Test

Figure 45 Post Success

5.2.1.1.3 Test Case 3

Purpose: Updating user profile

Prerequisites: Full Name, Bio, Tags, Profile Picture, Information Picture

Procedure: A user changes the Profile Picture and background Picture, inputs a new name and bio, selects or unselects the tags and clicks on the update button.

Result: The user gets an update successful pop-up message

Verdict: Update Successfully

Failure: An error may occur if the ajax call is unsuccessful or a component in the profile table is not updated, but its data was posted from the front end.

5.2.1.1.4 Test Case 4

Purpose: Send Connection Request

Prerequisites: User and new connect id.

Procedure: The user clicks on the handshake icon, and a new section appears where the user is asked to send the request. The user clicks on send, and the connect receives the request to accept or decline.

Result: The connect receives a notification request where they can either accept or decline the request.

Verdict: Request Sent Successfully.

Failure: An error may occur if the id of the connect is not retrieved correctly; hence the request goes to the wrong user.

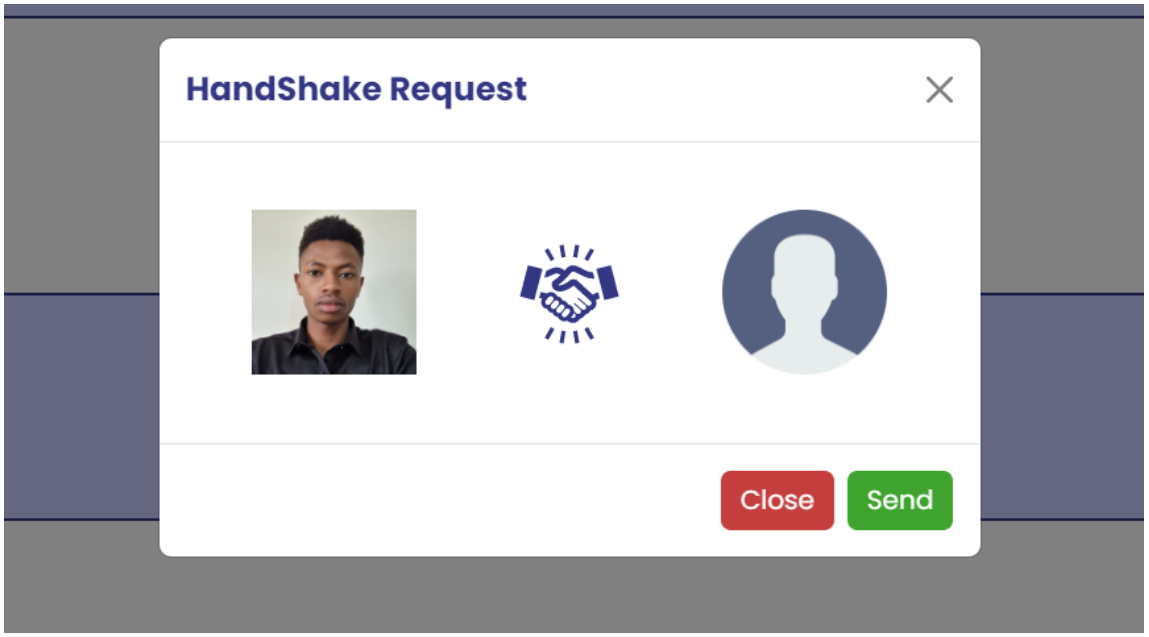

Figure 46 Send Request Popup

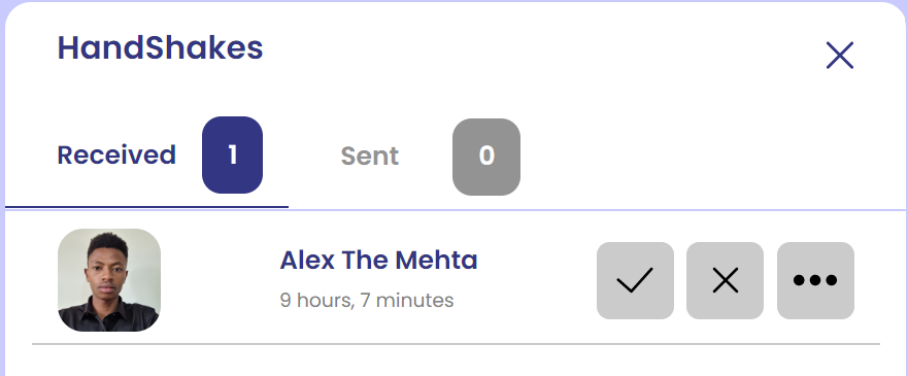

Figure 47 Request Received

5.2.1.1.5 Test Case 5

Purpose: Messaging

Prerequisites: Room Id, Users, Message.

Procedure: A user joins a room with a connection, inputs their message, and clicks send.

Result: The message is received by the connect in real-time.

Verdict: Message Sent Successfully

Failure: An error may occur if the room id is not retrieved accurately since the message will be posted to the wrong room.

5.2.1.1.6 Test Case 6

Purpose: Share Screen

Prerequisites: Room Id, Users, and Video Data.

Procedure: A user joins a room with a connection and clicks on the start sharing button. A screen pops up where the user selects the Screen they want to share to the connect, and the screen is shared.

Result: The connect receives a live feed of the other user's screen.

Verdict: Share Screen Successful

Failure: An error may occur if the connection is already in the room since they will not receive the feed until they rejoin the room.

5.3 Deployment

Deployment was the final step in the development of this platform. The hosting company selected was Host Pinnacle because it is local and uses M-pesa for payment. Since all of the updates were regularly committed to GitHub, all it took to host the website was to clone the repository in c-panel, create a new Postgres database, and finally, it was taken live. All of this costed Ksh 2000.

6 CONCLUSION AND RECOMMENDATION

6.1 Discussion

Software bugs are the biggest obstacle to software development. They are also among the most prominent factors causing beginner programmers to drop out of programming. Although there are platforms where programmers can get solutions to these bugs, these platforms are still not enough to handle the number of issues that need to be solved.

This research aimed to provide a platform where programmers can interact individually to improve how questions are asked and answered. The structure of the question also proved to be a significant influence on how the question was answered. A well-structured question resulted in a well-defined answer. By analyzing the other available platforms, it was concluded that having more one-on-one interactive platforms greatly improved the rate at which questions were answered. Implementing the Chat, Share Screen, and Video Chat modules proved this result true since now users could go beyond the answers posted.

METHODOLOGY proved to be appropriate in the implementation of this project. This is because it allowed alteration as the project was in progress. Furthermore, the release of versions played a big part in the user acceptance stage.

6.2 Limitations

6.2.1 Limited Time

The project's scope was enormous because this platform could potentially serve members of the programming community all over the world. However, the six months allocated for its completion was too short, resulting in so much not being implemented while other modules were not being implemented with the best programming practices.

6.2.2 Skill Set and Experience

For this platform to serve all members of the programming community despite their level of experience, the resources to be built into it needed to be top-tier. However, the level of experience needed to achieve this is senior level or higher, presenting a significant challenge in the implementation stage.

6.3 Recommendations

The current study can be interpreted as a step toward creating a platform capable of providing solutions to all programming errors. However, there are still so many gaps and modules that can be added to better answer questions. The ability to run code snippets posted in the questions and answers could significantly contribute to how effectively these questions and answers are understood by the users and programming community at large.

6.4 Conclusion

Unlike in the earlier years, the programming world has grown and hence the implementation of platforms like stack overflow and Code Project to aid in the solving of newly developing programming errors. However, as this research illustrates, these platforms are insufficient to handle the vast and growing number of programmers and their bugs. With this, new and more advanced platforms need to be developed to cater to the lack of one-to-one interaction among programmers in the programming community.

REFERENCES

Bogan Christine. (n.d.). *What does system mean?* Retrieved August 30, 2022, from https://badass.aussievitamin.com/what-does-system-mean

Calzon Bernardita. (2022, March 12). *What Is Data Analysis? Methods, Techniques, Types & How-To*. https://www.datapine.com/blog/data-analysis-methods-and-techniques/

CodeProject. A Guide. - CodeProject. (n.d.). Retrieved August 29, 2022, from https://www.codeproject.com/info/guide.aspx

Content Policy - Reddit. (n.d.). Retrieved August 29, 2022, from https://www.redditinc.com/policies/content-policy

Data Dictionaries in Software Engineering - GeeksforGeeks. (n.d.). Retrieved September 1, 2022, from https://www.geeksforgeeks.org/data-dictionaries-in-software-engineering/

Data preparation and preliminary data analysis. (n.d.).

Database Design - Javatpoint. (n.d.). Retrieved December 4, 2022, from https://www.javatpoint.com/database-design

Fact Finding Techniques || Fact Finding Techniques for Requirements. (n.d.). Retrieved August 29, 2022, from https://bcisnotes.com/thirdsemester/system-analysis-and-design/fact-finding-techniques-fact-finding/

Flowchart - Process Flow Charts, Templates, How To, and More. (n.d.). Retrieved August 31, 2022, from https://www.smartdraw.com/flowchart/

Introduction to Systems Implementation | Open Textbooks for Hong Kong. (2016, January 19). https://www.opentextbooks.org.hk/ditatopic/25374

Jafari Ladan. (2020, May 12). *What Are Functional Requirements? Types and Examples - WINaTALENT | Blog*. https://winatalent.com/blog/2020/05/what-are-functional-requirements-types-and-examples/

Matt Heusser. (n.d.). *What is debugging?* Retrieved August 29, 2022, from https://www.techtarget.com/searchsoftwarequality/definition/debugging

Montti Roger. (n.d.). *What Is Quora & How It Works*. Retrieved August 29, 2022, from https://www.searchenginejournal.com/what-is-quora/460545/#close

Nonfunctional Requirements Examples | Indeed.com. (2021, May 4). https://www.indeed.com/career-advice/career-development/non-functional-requirements-examples

O'dell, D. H. (n.d.). *Understanding the psychology of learning strategies leads to effective problem-solving skills. The Debugging Mindset*.

Opinaldo Norlyn. (n.d.). *What is A System Context Diagram with Explanation Examples*. Retrieved August 31, 2022, from https://gitmind.com/context-diagram.html

Pedamkar. (n.d.-a). *Class Diagram | Types & Examples | Relationship and Advantages*. Retrieved August 30, 2022, from https://www.educba.com/class-diagram/

Pedamkar. (n.d.-b). *System Analysis And Design | Top 11 Differences You Should Know*. Retrieved August 30, 2022, from https://www.educba.com/system-analysis-and-design/

Ramachandran Saranya. (2022, July 12). *What Is Requirement Analysis: Applications, Techniques and Tools Used [2022 Edition]*. https://www.simplilearn.com/what-is-requirement-analysis-article

Reasons why there are Software Bugs/Defects in Software. (n.d.). Retrieved August 29, 2022, from https://www.qamentor.com/top-10-reasons-why-there-are-bugs-defects-in-software/

Roxanna Evan. (2022, July 7). *What is an entity relationship diagram (ERD)? | MiroBlog*. https://miro.com/blog/entity-relationship-diagram/

Sachs Jason. (n.d.). *My Love-Hate Relationship with Stack Overflow*. Retrieved August 29, 2022, from https://www.embeddedrelated.com/showarticle/741.php

Sequence Diagrams - What is a Sequence Diagram? (n.d.). Retrieved August 31, 2022, from https://www.smartdraw.com/sequence-diagram/

System Architecture Diagram: A Complete Tutorial | EdrawMax. (n.d.). Retrieved December 4, 2022, from https://www.edrawsoft.com/article/system-architecture-diagram.html

The Beginner's Guide to Google Forms. (n.d.). Retrieved August 29, 2022, from https://www.howtogeek.com/434570/the-beginners-guide-to-google-forms/

Understanding Project Testing and Its Phases in Project Quality Management. (n.d.-a). Retrieved December 4, 2022, from https://pm-training.net/project-testing/

Understanding Project Testing and Its Phases in Project Quality Management. (n.d.-b). Retrieved December 4, 2022, from https://pm-training.net/project-testing/

Use Case Analysis: Tutorial & Examples | Study.com. (n.d.). Retrieved August 30, 2022, from https://study.com/academy/lesson/use-case-analysis-tutorial-examples.html

Website Navigation: Definition, Importance and Tips | Indeed.com. (2021, May 18). https://www.indeed.com/career-advice/career-development/website-navigation

What Are Software Bugs? | TotalView. (n.d.). Retrieved August 29, 2022, from https://totalview.io/blog/what-software-bugs

What is a Data Flow Diagram | Lucidchart. (n.d.). Retrieved August 31, 2022, from https://www.lucidchart.com/pages/data-flow-diagram

What Is a Favicon and Why Does Your Website Needs One? | Upwork. (2021, May 7). https://www.upwork.com/resources/what-is-a-favicon

What is Authentication? Definition of Authentication, Authentication Meaning - The Economic Times. (2022, September 2). https://economictimes.indiatimes.com/definition/Authentication

What is deployment in software & web development? (n.d.). Retrieved December 4, 2022, from https://umbraco.com/knowledge-base/deployment/

What is functional requirements? - Definition from WhatIs.com. (2022). https://www.techtarget.com/whatis/definition/functional-requirements

What is System Design? - Definition from Techopedia. (2014, February 14). https://www.techopedia.com/definition/29998/system-design

What is UX Design? Differences Between UX and UI Design | Columbia Engineering Boot Camps. (n.d.). Retrieved December 4, 2022, from https://bootcamp.cvn.columbia.edu/blog/what-is-ux-design/

APPEDICES

QUESTIONARE

Google Form Link

SAMPLE CODE

//JavaScript Posting Question using Ajax

```
//Here we will create a question using ajax call

//here we importthe list containing the tags

import {tagsArray} from './add.js'

//Here we import the array containing the frameworks

import {framesArray} from './add.js'

//This is the title of the question

let quizTitle = document.getElementById('title');

//The rich edit text

let richText = document.getElementById('richEdit');

//This are the code blocks

import {editor1Value,editor2Value,editor3Value,editor4Value} from './code.js'

//Here we collect the languages indicated in the code blocks

let lang1 = document.getElementById('code_1_lang');
```

```
let lang2 = document.getElementById('code_1_lang1');

let lang3 = document.getElementById('code_1_lang2');

let lang4 = document.getElementById('code_1_lang3');

//Here we take the images

let quizMedia = document.getElementById('myInputDrop');

import {inputDrop} from './drop.js'

//Lets create a form using js

let formCreate = document.getElementById('revBtn');

let childEl = document.getElementById('goChangeArea');

//Here we are appending the form

formCreate.addEventListener('click', ()=> {

   setTimeout(function(){

      var parent = childEl.parentNode;

      var wrapper = document.createElement('form');

      parent.replaceChild(wrapper, childEl);

      wrapper.appendChild(childEl);

      wrapper.setAttribute("id","greatForm");

      wrapper.setAttribute("method","POST");

   },1000);
```

```
})


//Here we remove the form
let removeForm = document.getElementById('actBtnEdit');
removeForm.addEventListener('click', ()=> {
    document.getElementById('goChangeQuizNow').click();
    let greatForm = document.getElementById('greatForm');
    let childNew = document.getElementById('goChangeArea')
    //childEl.parentElement.remove()
    setTimeout(function(){
        greatForm.replaceWith(childNew);
    },100);
    //greatForm.removeChild(childNew);
})
//NOW LETS USE AJAX TO POST THE QUESTION
//Lets get the csrf token
let csrf = document.getElementsByName('csrfmiddlewaretoken');
let submitGreatForm = document.getElementById('submitGreatForm');
let createForm = document.getElementById('createForm');
```

```
createForm.addEventListener('click', ()=> {
    submitGreatForm.click();
    //console.log(quizTitle.value)
    console.log(quizMedia.files)
});
//Here we set the csrf token
const getCookie =(name) => {
    let cookieValue = null;
    if (document.cookie && document.cookie !== '') {
        const cookies = document.cookie.split(';');
        for (let i = 0; i < cookies.length; i++) {
            const cookie = cookies[i].trim();
            // Does this cookie string begin with the name we want?
            if (cookie.substring(0, name.length + 1) === (name + '=')) {
                cookieValue = decodeURIComponent(cookie.substring(name.length + 1));
                break;
            }
        }
    }
```

```
    return cookieValue;
}
const csrftoken = getCookie('csrftoken');
//Here we display none the alert
$('#statusSubmit').hide();
submitGreatForm.addEventListener('click', ()=> {
    console.log('click')
    console.log(quizTitle.value)
    if(document.getElementById('greatForm') !== null){
        let greatForm = document.getElementById('greatForm');
        greatForm.addEventListener('submit', e=> {
            e.preventDefault();
            console.log(csrftoken)
            //Now we perform the ajax call
            //First we create form data
            let formData = new FormData();
            //Append to form data
            //for (const tag in tagsArray){
                //formData.append('Tags',tag);
```

```
    //console.log(tag)
//}
for(let item = 0; item < tagsArray.length; item++ ){
    console.log(tagsArray[item])
    formData.append('Tags',tagsArray[item]);
}
for(let item = 0; item < framesArray.length; item++ ){
    console.log(framesArray[item])
    formData.append('Frameworks',framesArray[item]);
}
//formData.append('Frameworks',JSON.stringify(framesArray));
formData.append('Title',quizTitle.value);
formData.append('Body',richText.innerHTML);
formData.append('lang1',lang1.value);
formData.append('code1',editor1Value);
formData.append('lang2',lang2.value);
formData.append('code2',editor2Value);
formData.append('lang3',lang3.value);
formData.append('code3',editor3Value);
```

```
formData.append('lang4',lang4.value);
formData.append('code4',editor4Value);
for (const file of inputDrop.files) {
    formData.append("medias", file);
}
//formData.append('medias',inputDrop.files);
formData.append('csrfmiddlewaretoken', csrf[0].value);
$.ajax({
    type:'POST',
    url:'/greatForm/',
    data: formData,
    processData: false,
    contentType: false,
    success: function(response){
        console.log(response)
        //Scroll to the top of the body
        document.getElementById('actBtnEdit').click();
        $('#statusSubmit').show();
        $('#navigation').hide();
```

```
                setTimeout(function(){
                    //Show success color
                    document.getElementById('colorAlert').classList.add('success');
                    //Show success message
                    document.getElementById('message').innerText = "Question Creation
Successfull";
                },1000);
                setTimeout(function(){
                    $('#statusSubmit').hide(100);
                    $('#navigation').show(100);
                    //$("#newContainer").css('padding-top', '200px');
                },2500);
            },
            error: function(error){
                console.log(error)
                document.getElementById('actBtnEdit').click();
                $('#statusSubmit').show();
                $('#navigation').hide();
```

```
                setTimeout(function(){
                    //Show success color
                    document.getElementById('colorAlert').classList.add('failed');
                    //Show success message
                    document.getElementById('message').innerText = "Question Creation
Failed";
                },1000);
                setTimeout(function(){
                    $('#statusSubmit').hide(100);
                    $('#navigation').show(100);
                },2500);
            }
        })
    });
  }
});
```