HETAUDA CITY COLLEGE

Tribhuvan University

Institute of Science and Technology

Quizy

A PROJECT REPORT

Submitted To

Department of Computer Science and Information Technology

Hetauda City College

Under the Supervision of

Er. Sulav Paudel

In Partial fulfillment of the requirements for the Bachelor's Degree in Computer Science and Information Technology

Submitted by

Jenesh Bhagat Pradhananga [11574/073]

Pawan Waiba [11577/073]

Sanjay Khatri [11582/073]

Feb, 2021

Hetauda, Nepal

HETAUDA CITY COLLEGE

Tribhuvan University
Institute of Science and Technology

Date: 2077/10/23

Supervisor's Recommendation

I hereby recommend that this project prepared under my supervision by Sanjay Khatri, Jenesh Bhagat Pradhananga and Pawan Waiba entitled "**Quizy**" in partial fullfillment of the requirements for the degree of B.Sc in Computer Science and Information Technology be processed for the evaluation.

……………………………...
Er. Sulav Paudel
Hetauda City College
Chisapani, Heatauda-4

HETAUDA CITY COLLEGE

Tribhuvan University

Institute of Science and Technology

Date: 2077/10/23

Letter of Approval

This is to certify that this project prepared by Sanjay Khatri, Jenesh Bhagat Pradhanaga and Pawan Waiba entitled "**Quizy**" in partial fulfillment of the requirements for the degree of BSc. in Computer Science and Information Technology has been well studied. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

………………………

Er. Sulav Paudel

Supervisor

Hetauda City College

………………………

Mr. Saroj Poudel

Head of Department

Hetauda City College

…………………………

Mr. Binod Sharma

External Examiner

Abstract

The **Quizy** is a web application for to take online quiz in an efficient manner. The main objective of **Quizy** is to efficiently evaluate the candidate thoroughly through a fully automated system that not only saves lot of time but also gives fast results. It also helps to prepare the quiz faster and in managed manner. Despite using traditional methods of pens and paper for quiz, it's fun to be play with less effort and more ease. Mainly, this web application is suitable for school level students to make them intelligent in the lots of various type of questions.

Quizy consists of mainly two modules: Player, Host and Audience. The host module has privilege to manage the quiz where the player can simply play the game. The Host can create question and attach questions to the quiz to which they can host it. While hosting the quiz, the host generate a unique pin, which is used by players to join that quiz. Anyone can join the quiz if they have the required pin and can participate the quiz. After player joins a quiz, the player has to wait for the host to start the quiz. The players are shown question with four option when the host starts the quiz. Only the player with the turn can answer the question and get scored accordingly. The score is reflected as soon as the player answer the question. The Audience screen is for the live update of the ongoing game like the player's turn and their scores.

Acknowledgement

The success of this project "**Quizy**" would not have been possible without the kind support and assistant of many individuals. We are immensely blessed to get this all along the duration of our project. We would like to extend our profound gratitude to each and every one of them.

We are highly thankful to **Hetauda City College** for providing us this opportunity, as well as guidance, supervision and friendly environment for the successful completion of the project. We also admire the effort of BSc. CSIT coordinator **Mr. Saroj Poudel**, without his supporting role, the project would have been nowhere near to completion.

We would like to express our appreciation to our project supervisor **Er. Sulav Paudel** who took interest in our project and guided us throughout the project by providing all necessary ideas, information and advice for the development of the functional web application.

Our thanks and appreciation also go to all the teaching staff of Hetauda City College for their constant support and encouragement which helped us successfully complete our project. We are fortune enough to get support and encouragement from each and every one of our colleagues in developing the project. We are also grateful to our family members who provide us constant support along the duration of this project.

Last but not least, we would like to thank each team member whose active involvement, hard work and collaborative nature made this project come out of the idea to the real world.

Table of Contents

List of Abbreviation

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
COCOMO	Constructive Cost Model
CSS	Cascading Style Sheets
CRUD	Create Read Update and Delete
DFD	Data Flow Diagram
ERD	Entity Relationship Diagram
HTML	Hypertext Markup Language
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
MIS	Management Information System
MIT	Massachusetts Institute of Technology
MVC	Model View Controller
PDO	PHP Data Objects
PHP	PHP Hypertext Preprocessor
RDBMS	Relational Database Management System
SQL	Structured Query Language
UML	Unified Modeling Language
XML	Extensible Markup Language
XP	Extreme Programming

List of Figures

List of Tables

1. Introduction

1.1.Relevance

‘**Quizy**’ is a web application that has general questions related to current affairs and computer. It has multiple choice questions and it also calculate score of each correct answer. It is good for students of every age group it helps in increasing general knowledge about world, sport and computers etc. User need to register first then login into the ‘**Quizy**’. The authenticated user is called a host. The host have to login to manage their quizzes. The host can prepare questions and attach them to the quiz they want. After creating their desire quiz, they can host the quiz and share the authentic pin of the quiz to the participants. Only the participants with the key can join the quiz.

1.2.Problem Statement

‘Quizy’ is a web application developed to conduct or organize quiz based on time constraints. **‘Quizy’** system is accessed by entering the username and password which is added to the database. Only the host has access to credential, so they can create and manage quizzes. The player can join the quiz with the pin provided by the host. Quiz is started by the host by displaying on question with four options. The players then have to choose the options. If the answer is correct, score is incremented by one and no negative marks for wrong answer. Quiz system will finally direct to the score page and final score will be displayed along with the names of the player.

1.3.Objectives

The main objective of **‘Quizy’** is to digitalize the system and facilitate a user-friendly environment for all users and reduces the manual effort. In the past days quiz is conducted manually but in further resolution of the technology we are able to generate the score and pose the queries automatically. The functional requirements include to create users that are going to participate in the quiz, automatic score and administrative tasks like add, delete, update for admin privilege users. In this application, all the permissions lie with the administrator i.e., specifying the details of the quiz, customizing the questions and hosting the quiz.

1.4.Scope and Limitation

1.4.1. Scope

The use of technology is increasing day by day, we all depends on technology, and we use various technologies to accomplish specific tasks in our lives. Todays, various type of games are played online and various organizations organize online games and, **‘Quizy’** is one of those online game with lots of knowledge. Participants can gain knowledge while having fun in the game.

1.4.2. Limitation

Technology, the one obvious limitation is that not every classroom has a computer, let alone access to a computer lab with one computer for each student.

Quality, the astounding variety of materials available in the internet – not all of them are good quality. Some may have mistaken; others may not be challenging enough.

2. Literature Review

Currently, education is evolving from a traditional perspective to a digital and web-based perspective characterized by the technological advances in the society in general and particularly in the academic world. Quiz game application can be considered as relevant source to build complementary education material. Generally, the application allows building online quiz games, which can be used in educational areas to develop questionnaires for reviewing the concepts already covered during the class. Here, the literature review consists of two existing system that provides quiz game-play as follow:

A. Infuse Learning

Infuse Learning is a system available from any device with Internet connection. With Infuse Learning, teacher could develop quizzes with several different types of questions. To create a quiz, teachers have to create an account and login. Once the quiz has been developed, a number is assigned to that quiz and student could gain access to the quiz anywhere and anytime. Students only have to insert the numbers of the quiz and their name. But however, Infuse Learning does not allow to compete among other participants/students.

B. Verso

Verso is based on the collaboration principle to share ideas. The strategy of Verso is that students could not see the contributions and responses made by other students in their classes until they have posted their own idea or responses. Moreover, when the responses are visible, students could not know who have posted each response, which encourages participation and sharing of ideas, but doesn’t support the concept of quiz at all. It can be considered a platform for sharing and contributing ideas and thoughts but cannot be used for competition among participants for some quality fun time.

3. Requirement Analysis

Before making any new system, we have to take consideration of different requirements. Since our project is based on the digitalization of online quiz contest. Hence, we use the following techniques to analysis the requirement.

3.1.Data Collection Methods

Data collection is the systematic approach of gathering and measuring information from a verity of source to get accurate and complete picture of an area of interest. We have done the data collection for our Quizy with various techniques.

3.1.1. Questionnaires

A questionnaire is a very popular tool for data collection, where a number of respondents are asked to answer a set of question either written or verbal about a pertinent topic. Questionnaire is done by creating a set of questions for the targeted respondents specifying the information needed. To collect the information for Quizy we have prepared a set of questions for the respondent.

- ✓ Do the existing quiz systems are free to play?
- ✓ Are the systems easy to use and convenient for school environment?

3.1.2. Interaction

Interaction was done with various normal users which help us to gain vital information and views on the aspect of using online donating system over manual system. It helps us in gathering information, combined perspectives and opinion. Interaction allowed us to get directly through their ideas, demands and expectations that significantly enhance the design and development of our web application.

3.1.3. Internet Research and observation

We search different websites as a part of our research. We observe and analysis how they offer services. This research helps us to understand about the existing system and features they couldn't address.

3.2.Requirement Specifications

Requirement specification can be broken down into functional requirements and non-functional requirements.

3.2.1. Functional Requirements

Functional requirements define the function and components of the system. It also describes the technical details, accepting inputs, processing and what a system is going to achieve.

i) Login module

This module is provided with email and password for user to login, manage and host the quiz.

- ✓ **Input-** Host email and password.
- ✓ **Process-** On correct credential, if the host is approved by the admin is redirect to their dashboard.
- ✓ **Output-** Registered host can manage the questions (create, edit, delete), attach them to quiz and host the quiz.

ii) Admin module

Admin module also provide email and password with which admin can access and control the system.

- ✓ **Input-** Email and password.
- ✓ **Process-** Manage the hosts and categories for the quiz.
- ✓ **Output-** Administrator will maintain the host and have privilege to ban the host in case of violation of the application.

iii) Player Module

Here, participant (players) can join the quiz and play.

- ✓ **Input-** Name with pin of the quiz.
- ✓ **Output-** Access to question with option to choose and answer.

3.2.2. Non-Functional Requirements

Non-functional requirements describe how the system works. Following are the non-functional requirements of our system:

i) Performance

This is the part where depends on the speed of internet. This is about time expectation. Mouse click is the priority for the players for choosing their answer and submitting.

ii) Speed

This is the important part of it. The application calculates the score in real time and reflects them in application as soon as the player submits their answer.

3.3.Server/Client Requirement

Quizy system is based on client server architecture. Server/Client requirement helps us to outline the minimum software and hardware needed to deploy our project.

3.3.1. Server Requirements

The values below refer to the minimum available hardware and software required to run Quizy. The Laravel framework has a few system requirements. (Server Requirements, 2017).

Table 3.3-a: Minimum Hardware Requirements for Server

Processor	2.4 GHz or above
RAM	Greater than or equal to 2GB
Hard Disk	80GB or above
Cache Memory	512MB or above

Table 3.3-b: Minimum Software Requirements for Server

Basic	PHP Version greater than or equal to 5.6.4, OpenSSL PHP Extension, PDO, PHP Extension, PHP Extension, Tokenizer PHP Extension, XML PHP Extension
Back End	MySQL Server 2008
Operating System	Windows or any compatible operating system
Web Server	Apache, Nginx or any compatible web server

3.3.2. Client Requirements

The values below refer to the minimum hardware and software requirements need to client.

Table 3.3-c: Minimum Hardware Requirements for Clients

Processor	1.8GHz or any compatible processor
RAM	256 or above

Table 3.3-d: Minimum Software Requirements for Clients

Operating System	Windows or any compatible operating system
Browser	Any modern web browser

3.5.Feasibility Analysis

Before the system is developed, it must be investigated to find out whether the system is feasible or not. Feasibility study helps to decide if the project is worth the investment of time and money. It determines the potential success of the project. We performed following feasibility test for our project:

3.5.1. Technical Feasibility

The main objective of a technical feasibility study is to determine whether a certain plan of action is feasible or not. A technical feasibility assessment should be applied to our project; it helps us to know whether the project can be done technically or not. The technical feasibility concerns with equipment and software that will successfully satisfy the requirements. Our system is being developed using following technologies:

Table 3.5-a: List of major technologies used in Quizy

Front-End	HTML 5, CSS3, Bootstrap Vue
Backend	Laravel, SQL
Client-Side Script	Vue JS
For data interchange	AJAX, JSON

The project will be less risky because our group members are familiar with above technologies and the systems.

3.5.2. Economic Feasibility

Economic feasibility is considering as the bottom line for most system. We have to choose the technology that easily fits in the budget. Basic Constructive Cost Model (Basic COCOMO) is used for cost estimation of our project (COCOMO Model, 2013).

Quizy is going to build using PHP language. We are trying to deal with developing well understood application program, the size of development team is small and the team member are unexperienced in developing similar type of projects.

The web application is available and accessible via the Internet which makes it economically feasible for user. Users only require an Internet connection to access contents included in the web application. Moreover, there are no charges of any sorts associated with contents in the web application.

Webpage Name	Number of Lines of Code
Login	145 LOC
Sign Up	210 LOC
User Management	146 LOC
Quizzes Page	213 LOC
Questions Page	180 LOC
Player Page	190 LOC
Home Page	110 LOC
Host Dashboard	50 LOC
Registration Page	40 LOC
Home Page	130 LOC
Category Management	145 LOC
Total Line of Code = 2127	

Total Kilo Line of Code (KLOC) = 2767/1000 = 2.127

Effort = 3.0 x $(\text{KLOC})^{1.12}$ PM = 3.0 x $2.127^{1.12}$ = 7.35 Person Months

Time to develop the software (T_{dev}) = 2.5 x $(\text{Effort})^{0.35}$ Months = 2.5 x $(7.35)^{0.35}$ = 4.9 months

Cost required to develop the software = 4.9 x 8000 = Rs.39200

Basic COCOMO shows that the project will be completed in 6 months at the budget of Rs.40000.

3.5.3. Operational Feasibility

Operational feasibility is mainly related to the human involvement. The system solves the problem and fulfills the aspects of operational feasibility study.

3.5.4. Schedule Feasibility

Evaluation of time is the most important consideration in the development of project. Schedule feasibility helps to identified how long will each activity take and how long will the entire project take? Project schedule split project into task and estimate time and resources required to complete each task. The main purpose of scheduling is to identify the task dependencies and is to avoid delays caused by one task waiting for another to complete. Typically, Gantt chart is used for monitoring and tracking the project progress.

4. System Design

System Design describes the components, modules, system activities, and system architecture of Quizy. Various system design aspect has been followed and these are explained below:

4.1.System Flowchart

Below flowchart shows the overall system flow of Quizy.

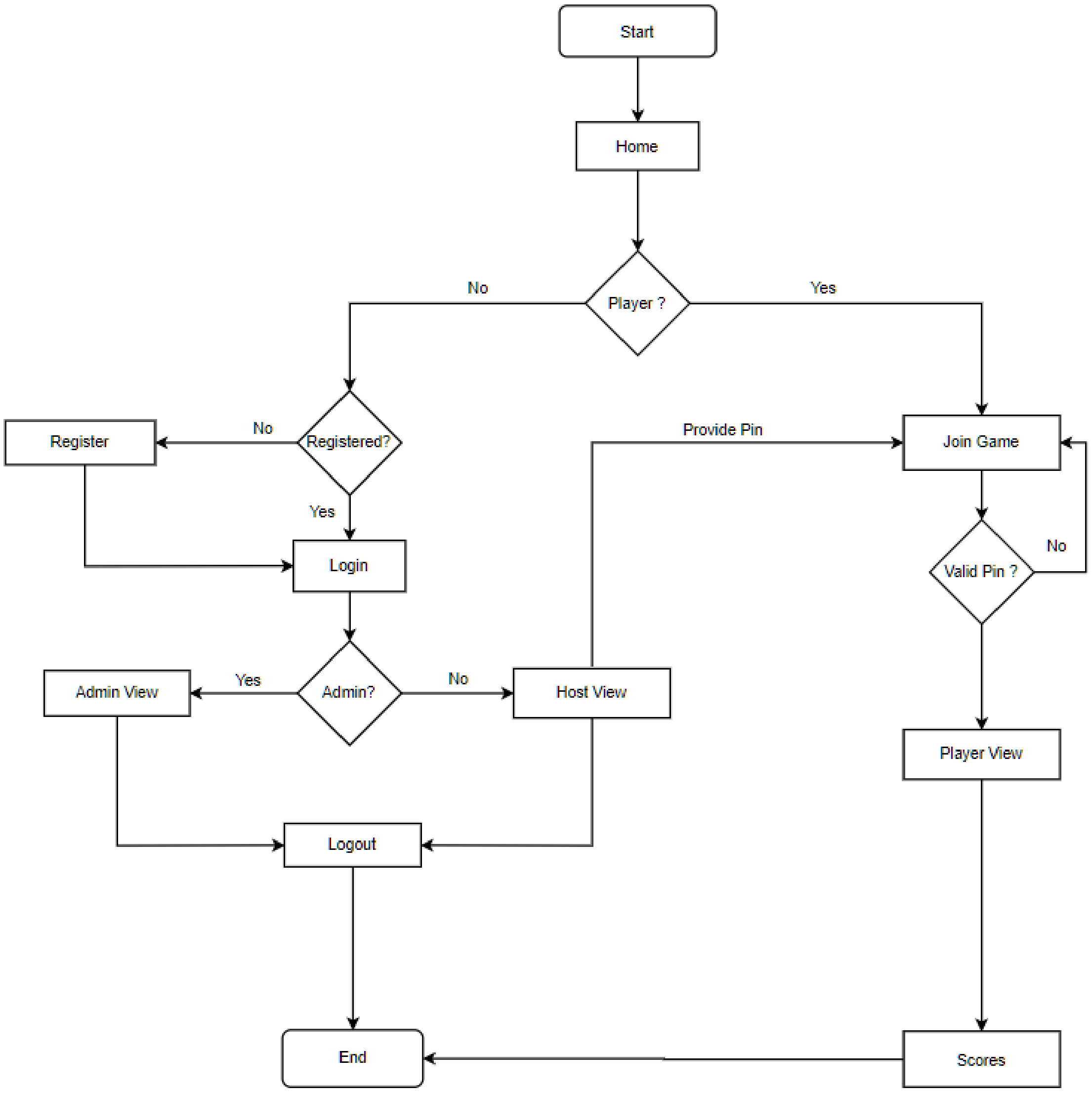

Figure 4-1: System flow chart

4.2.Use Case Diagram

The purpose of use case diagram is to capture the dynamic aspect of system. Use case diagram consists of use cases, actors, and their relationships. Use case diagram helps to get the outside view of our system, it also helps to identify the external and internal factors influencing the system. Following use case shows the dynamic aspect of proposed system.

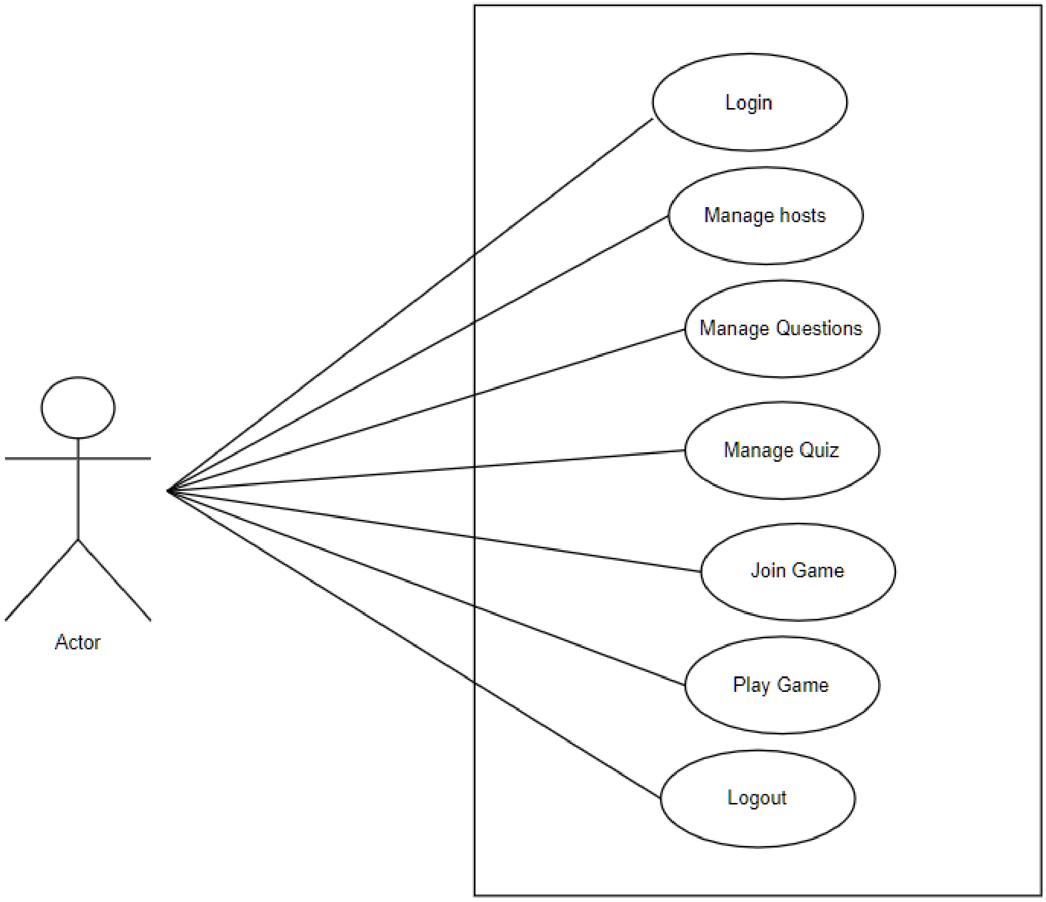

Figure 4-2: Use Case Diagram for main feature

4.3.Data Flow Diagram

Data Flow Diagram shows the flow of data in a system. Data Flow Diagram (DFD) consists of entities, processes, data store and data flow (Data Flow Diagram Notation, 2009).

Table 4.3-a: Data Flow Diagram Notation

Notations	Represent	Description
□	Entity	Source and destination of information data
○	Data Process	Activities and action taken on the data
=	Data Store	Data storage
→ ←	Data Flow	Data movement

4.3.1. Level 0 Data Flow Diagram

Level 0 DFD also known as context diagram. This diagram contains only one process node.

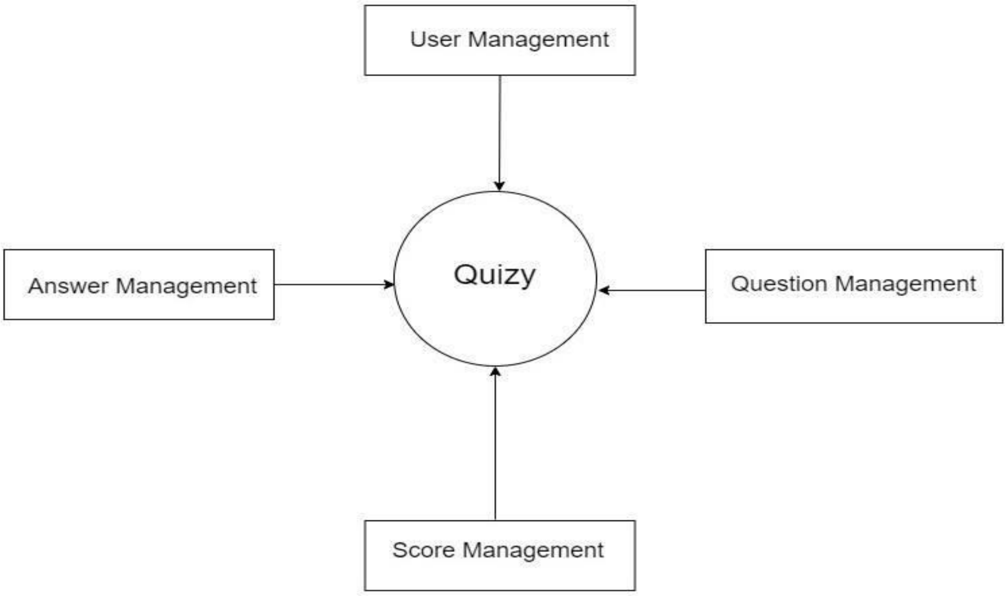

Figure 4.3-a: Level 0 Data Flow Diagram

4.3.2. Level 1 Data Flow Diagram

Level 1 Data flow diagram provides a more details than context level diagram. Level 1 DFD method is used to specify the functional requirement for the system design. It describes the business process, user perceived information, a workflow, and a decision rule for system.

The level 1DFD shows only the major high-level processes in the system interact. Each process on the level 0 DFD can be decomposed into more explicit DFD, called Level 1 diagram, which shows how it operate in getter details.

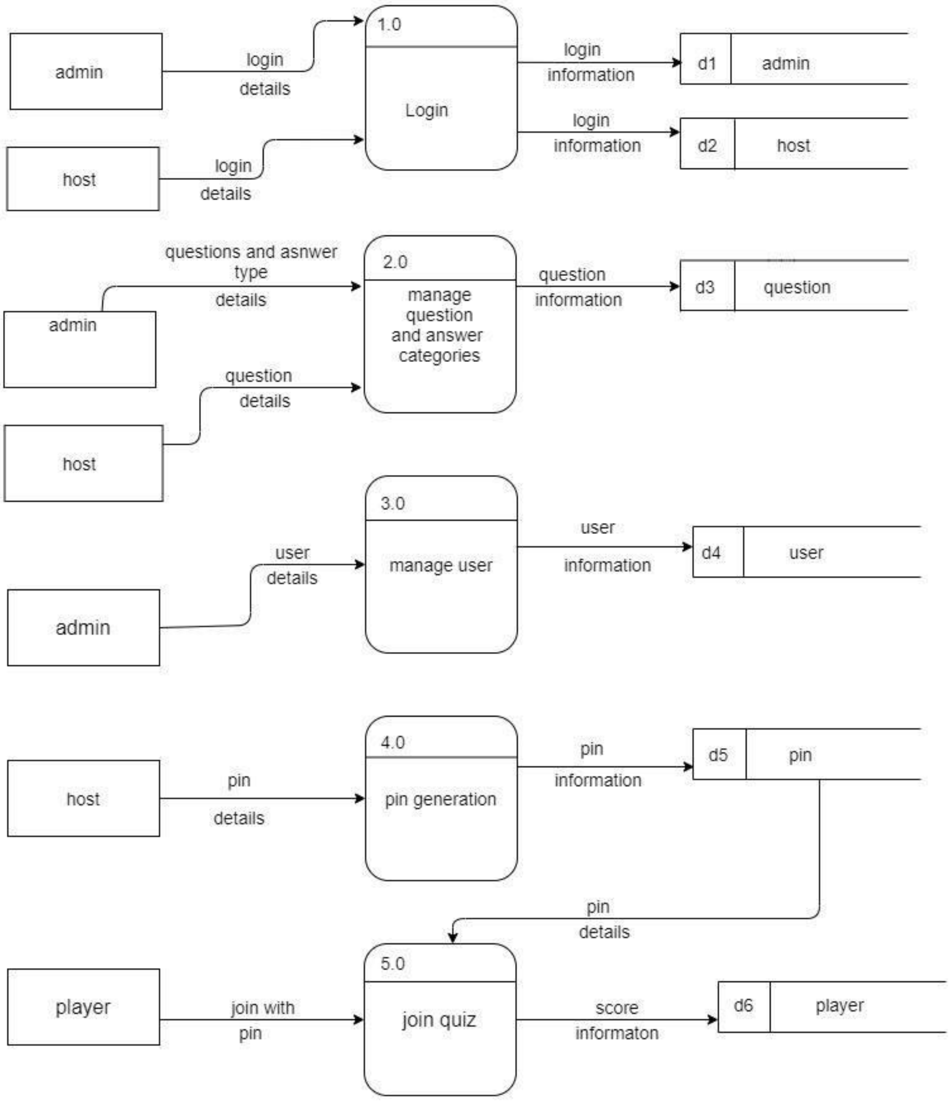

Figure 4.3-b: Level 1 Data Flow Diagram

4.4.Entity Relationship Schema

Entity Relationship Schema is tabular reorientation that describes how entities are related to each other. ER Schema of Quizy is given below:

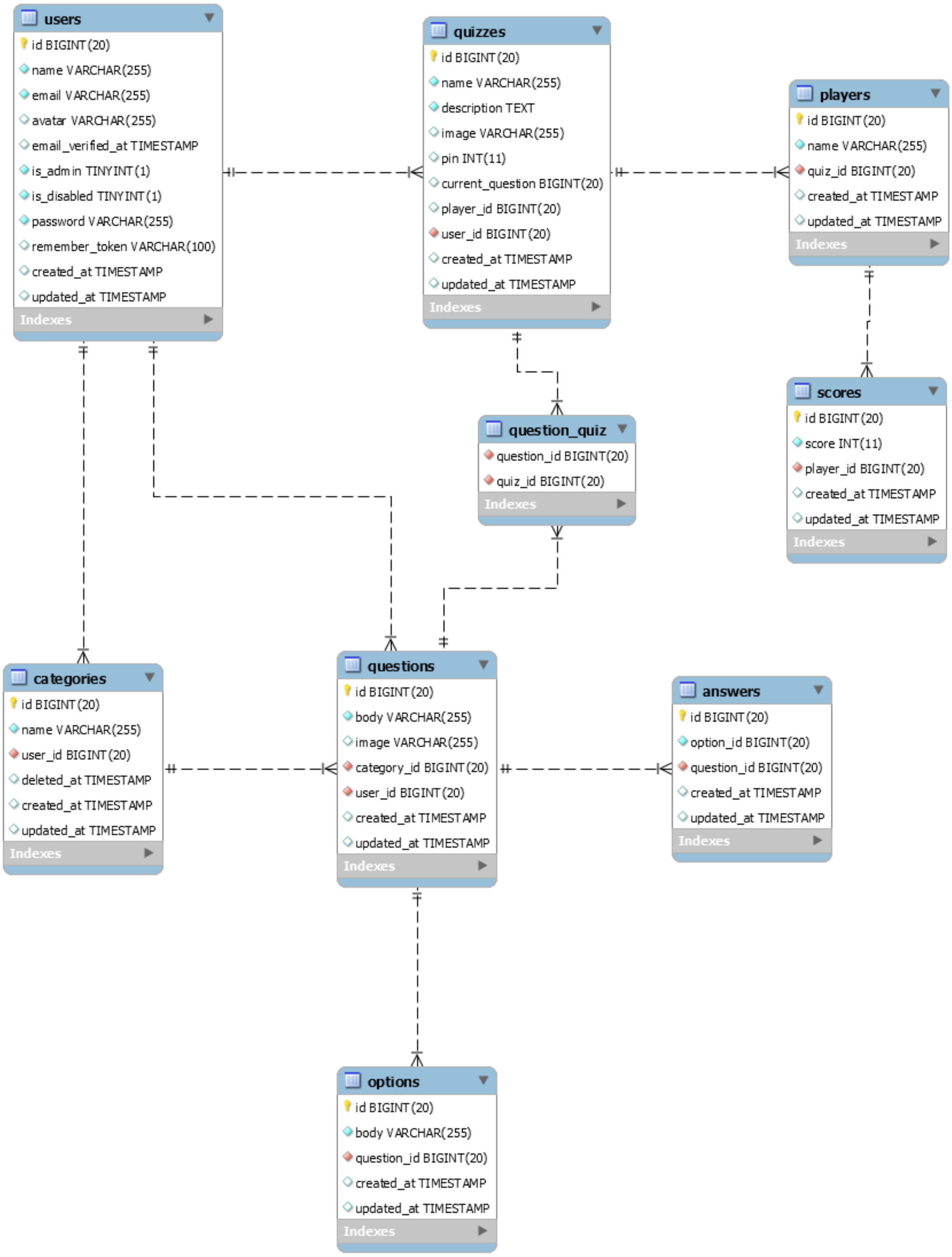

Figure 4-3: Entity Relationship Schema

4.5.Gantt Chart

Table 4.5-a: Gantt Chart

#	Task Name	Start	Duration	End
1.1	Team Formulation	Apr-10	1	Apr-11
1.2	Project Idea and Title Selection	Apr-12	11	Apr-23
1.3	Proposal Writing	Apr-23	5	Apr-28
2.1	Project Research	Apr-29	8	May-6
2.2	System Design	May-7	10	May-17
2.2	Flowchart	May-18	4	May-22
2.3	ER Diagram	May-23	3	May-26
2.4	DFD	May-27	3	May-30
2.5	Use Case Diagram	June-1	3	June-4
3.1	Front End Login & Registration View	June-5	4	June-9
3.2	Admin View	June-10	7	June-17
3.3	Host View	June-18	5	June-23
4.1	Backend Authentication	June-5	6	June-11
4.2	Admin Module	June-12	6	June-18
4.3	Quiz Module	June-19	4	June-23
4.4	Question Module	June-24	5	June-29
4.5	Player Module	June-30	6	July-5
5.1	Login & Registration Integration	June-12	2	June-14
5.2	Admin Integration	June-19	5	June-24
5.3	Host Integration	Jun-25	8	July-3
5.4	Player Integration	July-5	8	July-13
6.1	Unit Testing	July-14	7	July-21
6.2	System Testing	July-22	6	July-28
7	Report Writing	July-30	15	Aug-14

5. System Development

5.1.Software Development Model

5.1.1. Agile Development Model

Agile Method is an umbrella term that refers a group of development process, and not any single model of software development. Agile method is the combination of iterative and incremental development model. It breaks the product into small increment builds and these builds are developed in iterative manner (Hughes, Cotterell, & Mall, Agile Development, 2015).

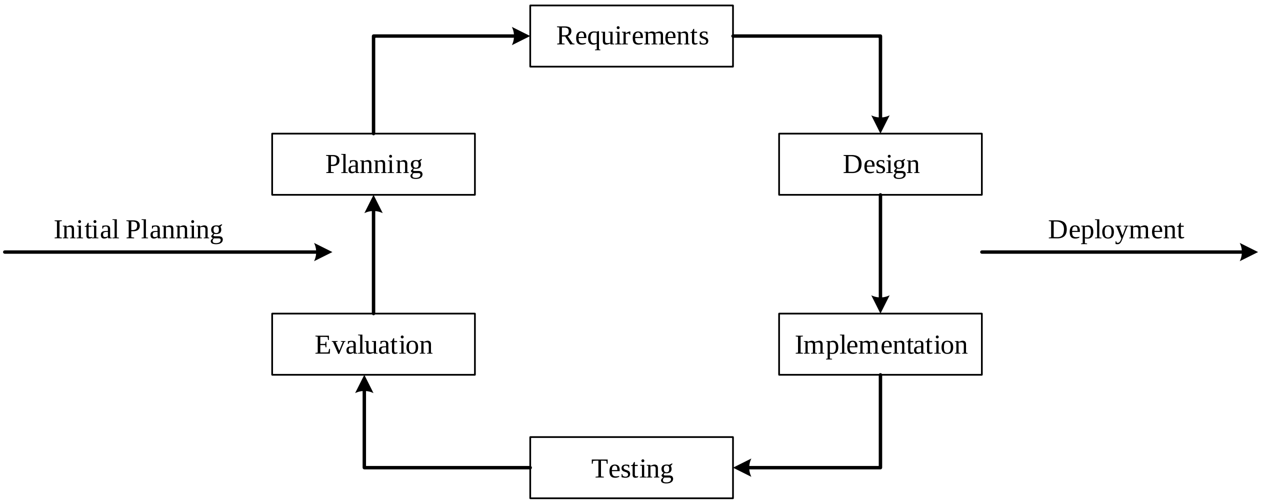

Figure 5-1: Agile Development

Agile development eliminates heavy weight implementation methodologies. In Agile development process featured requirements are decomposed into several parts that can be incrementally developed. Each incremental part is developed over an iteration. We use extreme programming method of agile development as the process model for our project.

Extreme Programming

Extreme Programming is one of the agile methodologies for system development. Here changes in requirements can be easily handled. It improves the quality and responsiveness of system application. Fundamental principle of extreme programming is rapid feedback, simplicity, incremental change and quality work (Hughes, Cotterell, & Mall, Extreme Progrmming, 2015).

Extreme Programming (XP) has been declared to be a new way of software development: a lightweight methodology, which is efficient, low-risk, flexible, predictable, scientific, and distinguishable from any other methodology. In the core of XP practices are programming activities, with strong emphasis on oral communications, automated tests, pair programming, storytelling culture and collective code-ownership at any time in the XP project. The paper gives an overview of XP practices and raises some serious concerns regarding their role in conceptual modeling and code generation; which directly affects software architecture solutions (Extreme programming and its development practices, 2000). There are five major phases in this agile methodology:

i) Planning

The first step we are going to take is planning. We will convert the requirements into the iteration which is usually a small part of the system functionalities. We will focus on deadline of the iteration so that we will be able to complete the requirements on time.

ii) Designing

After the planning we will start designing the overall system. Here we will include object-oriented design models (UML) to illustrate the overall functionalities of the system. This design models will show the active users of the system and their involvement in every process. Entity Relationship graph and data dictionary are developed at design part.

iii) Coding

This is the most important phases that we should take into account while developing the system. Here we will do pair coding for developing the system using one machine so that at the end we will be able to produce high quality code and at the same time reduce the overall cost. It will also help us to reduce the conflict and optimize the functionalities of the system.

iv) Testing

The testing is done to verify and validate the Quizy system. The objective of testing is to find as many bugs as possible and fix them so that it is free from any programming and logical error and the results would obtain as expected from the system.

v) Feedback

Feedback is the important phase where the developers directly get feedback from the customer or the project manager regarding the project. We took the feedback from each other as well as our supervisor after completing each module. This is the reason we can enhance our system make it more applicable as per the demand of the market.

5.2.Front End Tools

A "front-end" application is one that application users interact with directly. The front end is an interface between the user and the back end. In simple words we can understand the concept of front and back end with the help of below given definition. In client/server applications, the client part of the program is often called the front end and the server part is called the back end. While creating a front end, different components relating to the software development were used. These components are explained below.

5.2.1. HTML5/CSS3

HTML5 is a markup language used to design the webpages. It uses predefined tags to create the webpages. It is run using a web browser. We have used HTML5 to create the front-end of our system. We specially choose HTML5 for designing front-end of Quizy since it is the latest version and it contain number our new elements that helped to make our application more interactive.

CSS is a style sheet language that is used to describe the appearance and formatting of a web document, which is written in a markup language. It enhances the presentation of the web document. We used CSS3 to make Quizy front-end more attractive and to give the viewer better experience with its all-new feature.

5.2.2. Bootstrap Vue

Bootstrap Vue, based on the world's most popular framework - Bootstrap v4, for building responsive, mobile-first sites using Vue.js. Bootstrap Vue is used in our application as it is easy to use and understand. It makes the Quizzy responsive and adds more attractive features. (BootStrap Vue, 2020)

5.2.3. Vue

Vue is a progressive framework for building user interfaces. Unlike other monolithic frameworks, Vue is designed from the ground up to be incrementally adoptable. The core library is focused on the view layer only, and is easy to pick up and integrate with other libraries or existing projects. On the other hand, Vue is also perfectly capable of powering sophisticated Single-Page Applications when used in combination with modern tooling and supporting libraries. (VueJs, 2020)

5.3.Back End Tools

Back-end refers to everything the user can't see in the browser, like database and servers. Back-end developers or programmer is required to create a dynamic website. A dynamic web application is constantly changing and updated in real time. Quizy is developed by using following back-end technologies:

5.3.1. Laravel Framework

Laravel framework is a clean and well-structured PHP framework for web application development. It is a free, open-source framework and distributed under the MIT License. Laravel was created by Taylor Otwell and initial version was released in June 2011. Laravel provides powerful tools needed for large and small, robust application (Laravel Framework Introduction). The main features of Laravel are explained below:

i) Artisan Command

Artisan commands help developer to quickly create skeleton code for doing some administrative tasks, there are several common tasks that we have to do respectively during the development process such as the creation of database schema and build their migrations, creation of controller, middleware, views etc. We can create, delete, update database, controller class, middleware class etc. by entering few command-line codes (Artisan Development, 2016). Some of them are listed in

Table 5.3-a: List of artisan command and description

Artisan Command	Description
php artisan list	To view a list of all available Artisan commands
php artisan cache: clear	To flush the application cache
php artisan key: generate	To set the application key
php artisan help migrate	To view a help screen, precede the name of the command
php artisan tinker	To enter the Tinker environment
php artisan –version	To view the current version of your Laravel
php artisan make: controller	To create controller class
php artisan make: middleware	To create middleware class
php artisan make: provider	To create provider class
php artisan make: test	To create test class
php artisan serve	To Start the development server with default options
php artisan migrate: install	Creates the migration repository
php artisan migrate: make	To create the migration
php artisan migrate: refresh	Resets and reruns all the migrations
php artisan migrate: reset	Rollback all the database migrations
php artisan migrate: rollback	Rollback the last database migration
php artisan migrate: status	To show the status of each migration
php artisan view: clear	Clear all compiled view files
php artisan route: cache	Create a route cache file for faster route registration
php artisan route: list	List all registered routes

ii) Laravel MVC Pattern

Basically, Laravel is a fully Model-View-Controller (MVC) compliant framework. MVC pattern of Laravel ensures clarity of logic and presentation, and also helps in improving the performance, security, and allowing the better documentation. The MVC architecture pattern let the developer write a code that can be divided on the basis of the following three things:

- ✓ **Model**

  A Model is the layer by which developer can manipulate data, it lies between the data and application. The data itself can be stored in various types of database systems such as Maria DB, MS Access, MySQL, SQL Lite or Excel files etc.

- ✓ **View**

  Views are the visual or presentation layer of our web application, view is responsible for presenting the data that the Controller received from the model. In Laravel, Blade Templates are used as view.

- ✓ **Control**

  The primary function of controller is to handle requests and return appropriate responses to the correct Views. Controller acts as the middle man between Model and View. Controller is the logical layer of the application. Controller can receive data from view then process data and write the data to database, and redirecting users to appropriate routes.

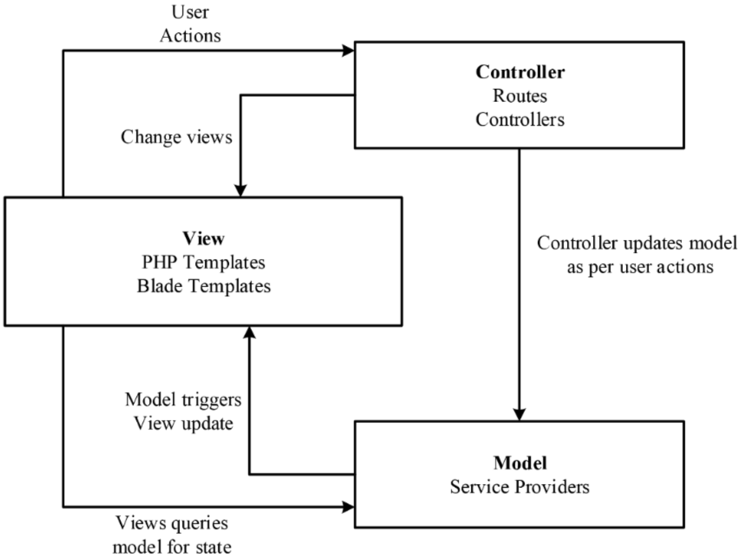

Figure 5-2: Model-View-Control

iii) Structure of Laravel application

The default Laravel application structure is intended to provide a great starting point for both large and small applications. Of course, we are free to organize our application whatever we like. Laravel imposes almost no restrictions on where any given class is located - as long as Composer can auto load the class. The below Figure 6.2-B (Directory Structure, 2017) shows the structure of Laravel application:

```
├── app                  # Your Laravel application
│   ├── Commands
│   ├── Console
│   ├── Events
│   ├── Exceptions
│   ├── Handlers
│   ├── Http
│   ├── Providers
├── artisan              # Artisan command-line utility
├── bootstrap
│   ├── app.php
│   ├── autoload.php
│   └── cache
├── composer. json
├── config
├── database
│   ├── migrations       # Database migration classes
│   └── seeds            # Database seeder classes
├── package.json
├── phpspec.yml
├── phpunit.xml          # PHPUnit configuration file
├── public
│   ├── assets
│   ├── favicon.ico
│   ├── index.php        # Entry point to run the application
│   ├── packages
│   ├── robots.txt
├── resources
│   ├── assets
│   ├── lang
│   └── views            # Contain blade template
├── server.php           # A lightweight local development server
├── storage
│   ├── app              # App storage, like file uploads etc
│   ├── framework        # Framework storage (cache)
│   └── logs             # Application-generated logs
├── tests                # Contain all test cases
└── vendor               # All third-party libraries installed by Composer
```

Figure 5-3: Laravel application structure

5.3.2. MySQL

MySQL is an Oracle-backed open-source relational database management system (RDBMS) based on Structural Query Language (SQL). It is the most popular open-source database. It is highly noted for its quick processing, proven reliability, ease and flexibility of use. It also complies on a number of platforms. (Back up and Recovery, n.d.)

All the database of Quizy is created using MySQL. It gives the flexibility of open source, high performance and data security. It also supports large databases. Comprehensive transactional support, complete workflow control and compatible with multiple platforms are some other features of MySQL that attracted us to use it for Quizy.

5.3.3. Pusher

Pusher empowers developers with APIs to create collaboration & communication features in their web and mobile apps. Pusher channels provides real-time communication between servers, apps and devices. Channels is used for real time charts, real-time, real-time maps, multiplayer gaming, and many other kinds of UI updates. We used pusher in our project for real time updating of the scores of players during the game session. (Pusher, 2020)

5.3.4. Apache Server 2.4.35

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards. The Apache HTTP Server ("httpd") was launched in 1995 and it has been the most popular web server on the Internet since April 1996. It has celebrated its 20th birthday as a project in February 2015 (Apache Server, 2017).

Apache Web Server is open-source Web server creation, deployment and management software. Initially developed by a group of software programmers, it is now maintained by the Apache Software Foundation. Apache Web Server is designed to create Web servers that have the ability to host one or more HTTP-based websites. Notable features include the ability to support multiple programming languages, server-side scripting, an authentication mechanism and database support. Apache Web Server can be enhanced by manipulating the code base or adding multiple extensions.

5.4.Others Tools

5.4.1. PHP Storm JetBrains

Php Storm is a powerful editor for PHP, HTML and JavaScript with on-the-fly code analysis, error prevention and automated refactoring for PHP and JavaScript code. It is built on JetBrains' IntelliJ IDEA platform. We are found this IDE very user friendly and using throughout the development of the project. We are using the Cracked version of Php Storm.

5.4.2. Windows Visio 2016

Microsoft Visio (formerly Microsoft Office Visio) is a diagramming and vector graphics application and is part of the Microsoft Office family. The product was first introduced in 1992, made by the Shapeware Corporation. It was acquired by Microsoft in 2000. We are using this tool to draw different graphical representation (Flowchart, ER-Diagram, Data Dictionary, UML Diagrams, Use Case Diagrams) for our project.

5.5.Graphical User Interface

5.5.1. Home Page

This page is rendered when the application is loaded first. Here user can join the quiz of which they should have pin of. The visitor can join the game with their name and the pin associated with the quiz they want to join. After entering the valid pin, the user is redirected to the game page, where they can play the quiz.

5.5.2. User Login Page

This is the page which is rendered when the host wants to login. Only registered users can login through this page. With success correct credentials, the user is redirected to the dashboard of Host. Register Page

This page is used to register a new host. Only host are allowed to host quiz, thus one should register to host quiz. Once the user registers with the required credentials, they will have to wait for their account approval by the admin. Admin Page

This is used by admin to manage the host in the application. The admins can also manage the categories used in the quiz. Only the user with authorized credentials can have access to this page.

5.5.3. Host Page

The Host page is where the quiz is made by authenticated user. They can manage the questions with multiple categories, can add them in quiz and host the quiz. Hosts can create multiple questions and attach them to quizzes.

5.5.4. Game Page

The Game page is where the quiz is played. The player with authentic pin joins the quiz. After the host starts the quiz, the player is presented with a question with four options. The player has to select their option and submit. Based on their answer, the score is reflected on the screen.

6. Testing

6.1. Unit Testing

Unit testing is the testing of individual software component or module. These tests are usually written by the developers of the module or programmer themselves. We performed the testing of test case by providing the input to the system then analyze the result.

6.1.1. Test case for host login

Table 6.1-a: Test case for host login

SN	Description	Input test data	Expected Result	Actual Result	Pass / Fail
1.	Enter valid data for email and password	Email: user@quizy.com Password: password	Should redirect to the host dashboard	Display the host dashboard	Passed
2.	Enter invalid email and password	Email: user@usr.com Password: password2	Error message display as the email and password you entered is incorrect	These credentials do not match our records.	Passed
3.	Email or password field empty	Username: test@test.com Password:	An error message should be displayed	Error message display as the input field must not be empty	Passed

6.1.2. Test case for admin login

Table 6.1-b: Test case for admin login

SN	Description	Input test data	Expected Result	Actual Result	Pass / Fail
1.	Enter valid data for email and password	Username: admin@quizy.com Password: password	Should redirect to admin page.	Display User Management page	Passed
2.	Enter invalid email and password	Email: dsad Password: dasdasda	Error message display as the username and password you entered is incorrect	These credentials do not match our records.	Passed
3.	Email or password field empty	Email: example@ab.com Password:	An error message should be displayed	Please fill out this field.	Passed

6.1.3. Test case for player

Table 6.1-c: Test case for player

SN	Description	Input test data	Expected Result	Actual Result	Pass / Fail
1.	Enter unique name and authentic pin	Name: John Doe Pin: 12345	Should redirect to game page.	Displays Welcome text and list of other participants	Passed
2.	Enter name and unauthentic pin	Name: Jane Doe Pin: 00000	Error message should be display	Invalid Pin	Passed
3.	Enter existing name and pin	Name: John Doe Pin: 12345	An error message should be displayed	Player name already exists	Passed

6.2.Integration Testing

The Integration testing part of a testing methodology is the testing of the different modules/components that have been successfully unit tested when integrated together to perform specific tasks and activities. The test is often done on both the interfaces between the components and the larger structure being constructed, if its quality property cannot be assessed from its components. After integrating the requirements, we tested it, it was fine and satisfactory.

6.3.System Testing

The system testing part of a testing methodology involves testing the entire system for errors and bugs. This test is carried out by interfacing the hardware and software components of the entire system, and then testing it as a whole. This testing is listed under the black-box testing method, where the software is checked for user expected working conditions as well as potential exception and edge conditions.

6.3.1. Black Box Testing

Black box testing is software testing where functionality of the software is tested without looking the internal code structure, knowledge of internal paths and the implementation details. It typically involves running through every input to verify that the result is as expected. We have decided to perform Equivalence Class Testing and Boundary Value Testing for our system. Our system has successfully passed these vary test and ready for the implementation on the real world.

6.3.2. White Box Testing

White box testing is software testing where functionality of the system is ignored and only focuses on the code and the structure of the code. This testing is done to check whether all the code implements correctly. To ensure this to happen in our system we performed Branch Coverage technique and our system passed this testing as every code is working as expected.

7. Implementation and Result Analysis

This chapter covers the implementation stage of the Quizy System. The implementation phase involves system implementation in the Real-World Environment and to ensure its proper functionality. Before the system can be implemented, certain requirements must be meet. The Quizy System requires different hardware requirements for the server specified in the Table 2.4-a and software requirements for the server specified in the Table 2.4-b. As for the client requirements, the client must have an access to browser supported computing device and working internet connection.

After the requirements are meet, the system can be implemented and can be used to carry out day-to-day transaction. Beside these, the system must have PHP installed and MySQL server for the system to work. PHP is required as the system is developed using PHP language and the MYSQL server is used to handle the database operation. The system is developed in MVC pattern using Laravel framework. The system needs to be hosted on the WWW so that anyone can have access to the system from anywhere on the globe. (Armel, 2014)

Quizy is designed and developed in follow components: - admin, host, and player components. The admin components are only accessible to the administrator with authorized credentials. The administrator can manage all hosts as well as categories used for quiz. The admin has the privilege to approve or ban host in the application. The host has privilege of creating and conducting quizzes. The host can create questions, attach them to quiz and host the quiz. When the host conduct a quiz, a unique pin is generated, which is required for participants to join the game. The participant component is used by the participants to join the quiz conducted by the host. The participant must have the pin to join the game, which is provided by the host. After the participant join the game, they are displayed with a question with four options. The participant must submit their answer and they will be scored as soon as they submit in real time. They will get score only on correct answer with conditional visuals on the display.

8. Conclusion and Future Enhancement

Quizy is a web-based quiz game that aims to provide the student’s beloved game a digital platform. Quizy is developed for the ease of creating and playing quizzes as it saves the time managing quiz and is accessible from anywhere with just the help of web browser and internet. To build a system a variety of requirements are needed. To gather the requirements for our system we used the basic methods like questionnaires, interaction, internet research and observation. We get to know about the views of the people, their needs and expectation from this survey. We even learnt about the system that is currently running in the real-world scenario. It helped us to find what services they provide and how. The study of the current system motives us to build Quizy and provide those services that the current system lacks.

Quizy implements agile development model to develop. In the agile model, the feature requirements are decomposed into several small parts that can be incrementally developed. Agile development model adopts iterative approach. Each incremental part is developed over iteration. Agile focuses on the iterative process with small team of developers, deploying pair programming and having a customer representative in the team. Under agile development model we are using extreme programming approach. XP has been declared to be a new way of software development: a lightweight methodology, which is efficient, low-risk, flexible, predictable, scientific and distinguishable from any other methodology.

The development of Quizy needed a number of technologies. We used HTML, CSS, Bootstrap, Semantic UI and jQuery to design our frontend and Laravel framework, MySQL, Apache server to design our backend. During the design and development of our system we learnt to use these development tools. We got the knowledge about the use and implement of these tools and also learn the flow of these tools. The Laravel frame work was the new workspace for us but as we used it, we get the idea of what MVC pattern is and how to implement it in a system design. As every system need to validate and verified, we also, put Quizy through different testing methods and the outcomes were obtained as expected from the system. So, this system is flawlessly validated and verified.

Quizy is a web-based application for playing quiz online and get result in real time. Currently system is limited to web application only but it can be further extended as mobile application as well as Desktop application. This application has a great scope academically and in student's growth. With digitization of quiz game, one can save time without the compromise of knowledge and benefits this game has always given students.

As we all know, every system is not perfect in this world. We are also aware that every system needs to make some agreement or limit it feature to fully develop. We have to admit that the current Quizy lacks some of the objectives due limitation off technologies we can use.

9. References

Armel, J. (2014). *Web application development with Laravel PHP.* Helsinki: Metropolia University of Applied Sciences. Retrieved from https://www.theseus.fi/bitstream/handle/10024/74052/Author.pdf

Apache Server. (2017, October 23). Retrieved from apache.org: https://httpd.apache.org/

Artisan Development. (2016, August 23). Retrieved from Laravel: https://laravel.com/docs/5.0/commands

Back up and Recovery. (n.d.). Retrieved from MySQL: https://dev.mysql.com/doc/refman/5.7/en/backup-and-recovery.html/

BootStrap Vue. (2020, Feb 27). Retrieved from Introduction to BootstrapVue: https://bootstrap-vue.org/

Data Flow Diagram Notation. (2009). In S. Tayal, B. B. Agarwal, & M. Gupta, *Software Enginnering & Testing* (pp. 62-63). Jones and Bartlett Publisher.

Directory Structure. (2017, January 24). Retrieved from Laravel: https://laravel.com/docs/5.4/structure

Hughes, B., Cotterell, M., & Mall, R. (2015). Agile Development. In B. Hughes, M. Cotterell, & R. Mall, *Software Project Management* (pp. 88-89). Delhi: McGraw Hill Education (India) Private Limited.

Hughes, B., Cotterell, M., & Mall, R. (2015). Extreme Progrmming. In B. Hughes, M. Cotterell, & R. Mall, *Software Project Management* (5th Edition ed., pp. 89-92). Delhi: McGraw Hill Education (India) Private Limited.

Juric, R. (2000). Extreme programming and its development practices. *22nd International Conference on Information Technology Interfaces* (pp. 97-104). Pula, Croatia: IEEE. Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=915842&isnumber=19790

Kumar, R. (2013, March). COCOMO Models for E-commerce Application. *International Journal of Computational Science, Engineering & Technology*, 25. Retrieved from Eclatjournals: http://eclatjournals.com/ijcset/documents/Vol-I_Issue-II/Vol-I_Issue-II_Paper-4.pdf

Laravel Framework Introduction. (2020, February 27). Retrieved from laravel.com: https://laravel.com/

Pusher. (2020, Apr 5). Retrieved from Powering realtime experiences for mobile and web: https://pusher.com/

Resig, J. (2015, May). John Resig: Building JQuery. (S. Charles, Interviewer) www.computer.org. Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7111910&isnumber=7111853

Server Requirements. (2017, January 24). Retrieved from Laravel.com: https://laravel.com/docs/5.4/installation#server-requirements

VueJs. (2020, Feb 28). Retrieved from The Progressive Javascipt Framework: https://vuejs.org/