

M.S.P Mandal's
Deogiri College, Aurangabad.



Project Report On

“Qurit: Student Query Portal”

Submitted By

Pratik Pramod Arak

Roll No.: 01

Jayesh Santosh Bhavsar

Roll No.: 04

Guided By

Mr. Bhushan S. Kulkarni

Ms. Sana Deshmukh

Submitted To

DEOGIRI COLLEGE, AURANGABAD

Department of Computer Science, IT and Animation

2022-2023

DEOGIRI COLLEGE, AURANGABAD

DEPARTMENT OF COMPUTER SCIENCE, IT AND ANIMATION



CERTIFICATE

*This is to certify that the following candidate of **B.Sc. (I.T) III Year** has satisfactorily completed the project entitled “**Qurit: Student Query Portal**” for the partial fulfillment of the **Bachelor Degree of Dr. Babasaheb Ambedkar Marathwada University, Aurangabad** for the academic year **2022-2023**.*

Submitted By

Pratik Pramod Arak Roll No.: 01

Jayesh Santosh Bhavsar Roll No.: 04

In-charge

Mr.Bhushan S.Kulkarni

Project Guide

Mr.Bhushan S. KulKarni

H.O.D.

Dr. S. N. Helambe

Ms. Sana S. Deshmukh

External Examiner

ACKNOWLEDGEMENT

We take this opportunity to thank our guides Mr. Bhushan S. Kulkarni and Ms. Sana Deshmukh for their kind support and for showing us right direction so that, we could take the right approach in development of our project. This project is our academic landmark and we reconsider ourselves very lucky to have **Mr. Bhushan S. Kulkarni** and **Ms. Sana S. Deshmukh** as our guides in giving it a professional touch.

We thank **our H.O.D. Dr. S. N. Helambe** for his magnificent help and support.

Thanking You,

Pratik Pramod Arak

Jayesh Santosh Bhavsar

INDEX

Sr. No.	Content	Page No.
1	Introduction to HTML	1
2	Introduction to CSS	1
3	Introduction to JavaScript	1
4	Introduction to PHP	1
5	Introduction to MySQL	1
6	Introduction to Project	1
7	System analysis study report i. User Interview ii. Problem Defination	1
8	Hardware & Software Requirements	1
9	Scope of system	1
10	Data Flow Diagram	1
11	Feasibility study i. Environmental Feasibility ii. Economic Feasibility iii. Technical Feasibility	1
12	Testing i. Black Box Testing ii. White Box Testing iii. Cross Browser Testing	1
13	Source Code	1
14	Output Form	1
15	Conclusion	1
16	Bibliography	1

1. INTRODUCTION TO HTML

1. Hypertext Mark-up Language (HTML) is the standard mark-up language to create webpage.

It is written in the form of HTML language consisting of tags enclosed in angle bracket (like). HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although some tags represent empty elements and so are unpaired, for example ``. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags).

2. What is HTML?

- HTML stands for Hyper Text Mark-up Language.
- A mark-up language is a set of mark-up tags.
- HTML documents are described by HTML tags.
- Each HTML tag describes different document content.

Web Browser can read HTML files and compose them into visible or audible web pages. Browsers do not display the HTML tags and scripts, but use them to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presenting, making it a mark-up language, rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in language such as JavaScript which affect the behavior of HTML web pages.

2. INTRODUCTION TO CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to change the style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG, and XUL. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content, such as semantically insignificant tables that were widely used to format pages before consistent CSS rendering was available in all major browsers. CSS makes it possible to separate presentation instructions from the HTML elements; it provides a list of formatting instructions. For example, a CSS rule might specify that all heading 1 elements should be bold, leaving pure semantic HTML mark-up that asserts this text is a level 1 heading without formatting code such as a tag indicating how such text should be displayed

1. What is CSS?

- CSS stands for Cascading style Sheets.
- CSS defines how HTML elements are to be displayed.
- Styles were added to HTML 4.0 to solve a problem.
- CSS saves a lot of work.
- External Style Sheets are stored in CSS files.

This separation of formatting and content makes it possible to present the same mark-up page in different rendering methods, such as on-screen, in print, by voice (when readout by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to display the web page differently depending on the screen size or device on which it is being viewed.

While the author of a web page typically links to a CSS file within the mark-up file, readers can specify a different style sheet, such as a CSS file stored on their own computer, to override the one the author has specified. If the author or the reader did not link the document to a style sheet, the default style

of the browser will be applied. Another advantage of CSS is that aesthetic changes to the graphic design of a document (or hundreds of documents) can be changed quickly and easily, by editing a few lines in one file, rather than by a laborious (and thus expensive) process of crawling over every document line by line, changing mark-up.

3. INTRODUCTION TO JAVASCRIPT

JavaScript is a dynamic computer programming language. It is most commonly used as part of web browsers, those implementations allow client-side scripts to interact with the user, control the browser, communicate as synchronously, and alter the document content that is displayed. It is also used in server-side network programming with runtime environments such as Node.js, game development and the creation of desktop and mobile applications. With the rise of the single-page web app and JavaScript-heavy sites, it is increasingly being used as a compile target for source-to source compilers from both dynamic languages and static languages. In particular, Emscripten and highly-optimized JIT compilers, in tandem with asm.js which is friendly to AOT compilers like Odin Monkey, have enabled C and C++ programs to be compiled into JavaScript and execute at near native speeds, making JavaScript be considered the —assembly language of the web, according to its creator and others.

JavaScript is classified as a prototype-based scripting language with dynamic typing and first- class functions. This mix of features makes it a multi-paradigm language, supporting object- oriented, imperative, and functional programming style.

Despite some naming, syntactic, and standard library similarities, JavaScript and Java are otherwise unrelated and have very different semantics. The syntax of JavaScript is actually derived from C, while the semantics and design are influences by the Self and Scheme programming languages.

JavaScript is also used in environments that are not web-based, such as PDF documents, site specific browsers, and desktop widgets. Newer and faster JavaScript virtual machines (VMs) and platforms built upon them have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript has been traditionally implemented as an interpreted language, but more recent browsers perform just-in-time compilation.

JavaScript has been standardized in the ECMA script language specification.

4. INTRODUCTION TO PHP

PHP is a server-side scripting language designed for web development but also used as a generalpurpose programming language. As of January 2013, PHP was installed on more than 240 million websites (39% of those sampled) and 2.1 million web servers. Originally created by Ramus Lerdorf in 1994, the reference implementation of PHP (powered by the Zend Engine) is now produced by the PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processors, which is a recursive backronym.

PHP code can be simply mixed with HTML code, or it can be used in combination with various template engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or a Common Gateway Interface (CGI) executable. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web pages; for example, PHP code can generate a web page's HTML code, an image, or some other data. PHP has also evolved to include a commandline interface (CLI) capability and can be used in standalone graphical applications.

The canonical PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

Despite its popularity, no written specification or standard existed for the PHP language until 2014, leaving the canonical PHP interpreter as a de facto standard. Since 2014, there is on-going work on creating a formal PHP specification.

1. CROSS PLATFORM: -

Interchange between different lines in a metro system, see cross-platform interchange. In computing, cross-platform, or multi-platform, is an attribute conferred

2. HTML-EMBEDED: -

PHP code is written in files containing a mixture of PHP instructions and HTML code.

3. SERVER SIDE: -

Server-side (commonly referred to as SS) refers to operations that are performed by the server in a client-server relationship in computer networking.

Typically, a server is a computer program, such as a web server, that runs on a remote server, reachable from a user's local computer or workstation. Operations may be performed server-side because they require access to information or functionality that is not available on the client, or require typical behavior that is unreliable when it is done client-side. Server-side operations also include processing and storage of data from a client to a server, which can be viewed by a group of clients.

Advantage: This lightens the work of your client. This also protects your SAMP server from crackers.

4. A WEB SCRIPTING LANGUAGE: -

A Scripting language or script language is programming languages that supports scripts, programs written for a special run-time environment that can interpret (rather than compile) and automate the execution of tasks that could alternatively be executed one-by-one by a human operator. Environments that can be automated through scripting include software applications, web pages within a web browser, the shells of operating systems (OS), and embedded systems. A scripting language can be viewed as a domain-specific language for a particular environment; in the case of scripting an application, this is also known as an extension language. Scripting languages are also sometimes referred to as very high-level programming language, as they operate at a high level of abstraction, or as control languages, particularly for job control languages on mainframes

The term scripting language is also used loosely to refer to dynamic high-level general-purpose language, such as Perl, Tcl, and Python, with the term —script often used for small programs (up to a few thousand lines of code) in such languages, or in domain-specific languages such as the textprocessing languages sed and AWK. Some of these languages were originally developed for use within a particular environment, and later developed into portable domain-specific and generalpurpose languages. Conversely, many general-purpose languages have dialects that are used as scripting languages. This article discusses scripting languages in the narrow sense of languages for a specific environment; dynamic, general-purpose, and high-level programming languages are discussed at those articles.

The spectrum of scripting languages ranges from very small and highly domain-specific languages to general-purpose programming languages used for scripting. Standard examples of scripting languages for specific environments include: Bash, for the Unix or Unix-like operating systems; ECMA Script (JavaScript), for web browsers; and Visual Basic for Applications, for Microsoft Office application. Lua is a language designed and widely used as an extension language. Python is a general-purpose language that is also commonly used as an extension language, while ECMA Script is still primarily a scripting language for web browsers, but is also used as a general-purpose language. The Emacs Lisp dialect of Lisp (for the

Emacs editor) and the Visual Basic for applications dialect of Visual Basic are examples of scripting language dialects of general-purpose languages. Some game systems, notably the Trainz franchise of Railroad simulators have been extensively extended in functionality by scripting extensions.

5. CHARACTERISTICS: -

In principle any language can be used as a scripting language, given libraries or bindings for a specific environment. Scripting is a property of the primary implementations and uses of a language, hence the ambiguity about whether a language is a scripting language for languages with multiple implementations. However, many languages are not very suited for use as scripting languages and are rarely if ever used as such.

Typically, scripting languages are intended to be very fast to pick up and author programs in. This generally implies relatively simple syntax and semantics. For example, it is uncommon to use Java as a scripting language due to the lengthy syntax and restrictive rule about which classes exist in which files – contrast to Python, where it is possible to briefly define some functions in a file. A scripting language is usually interpreted from source code or byte code. By contrast, the software environment the scripts are written for is typically written in a compiled language and distributed in machine code form. Scripting languages may be designed for use by end users of a program – end-user development – or may be only for internal use by developers, so they can write portions of the program in the scripting language. Scripting languages abstract their users from variable types and memory management.

Scripts are often created or modified by the person executing them, though they are also often distributed, such as when large portions of games are written in a scripting language. In many implementations a script or portions of one may be executed interactively on a command line.

6. HISTORY: -

Early mainframe computers (in the 1950s) were non-interactive, instead using batch processing. IBM's Job Control Language (JCL) is the archetype of languages used to control batch processing. The first interactive shells were developed in the 1960's to enable remote operation of the first timesharing systems, and these used shell scripts. Calvin Moores in his TRAC language is generally credited with inventing command substitution, the ability to embed commands in scripts that when interpreted insert a character string into the script. Multics calls these active functions. Louis Pouzin wrote an early processor for command scripts called RUNCOM for CTSS around 1964.

5. INTRODUCTION TO MYSQL

MySQL officially, but also called —My Sequel is (as of March 2014) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius's daughter, My. The SQL acronym stands for Structured Query Language.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open-source web application software stack (and other AMP' stacks). LAMP is an acronym for Linus, Apache, MySQL, and PHP/Python. Free- software open-source projects that require a full-featured database management system often use MySQL.

1. INTERFACES: -

MySQL is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL database or manage data contained within the databases. Users may use the include command line tools, or use SQL —front-end, desktop software and web applications that create and manage MySQL databases, build database structures, back up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

2. GRAPHICAL: -

The official MySQL Workbench is a free integrated environment developed by MySQL AB that enables users to graphically administer MySQL databases and visually design database structures. MySQL Workbench is available in places of the previous package of software, development (replacing MySQL Query Browser) and Database administration (replacing MySQL Administrator). MySQL Workbench is available in two editions, the regular free and open source Community Edition which may be downloaded from the feature set of the Community Edition. Third-party proprietary and free graphical administration applications (or frontends are available that integrate with MySQL and enable users to work with database structure and data visually. Some well-known front ends, in alphabetical order, are:

- Adminer – a free MySQL front end written in one PHP script, capable of managing multiple databases, with many CSS skins available.
- Database Workbench – a software application for development and administration of multiple relational databases including MySQL, with interoperability between different database systems.
- DB Edit – a free front end for MySQL and other databases

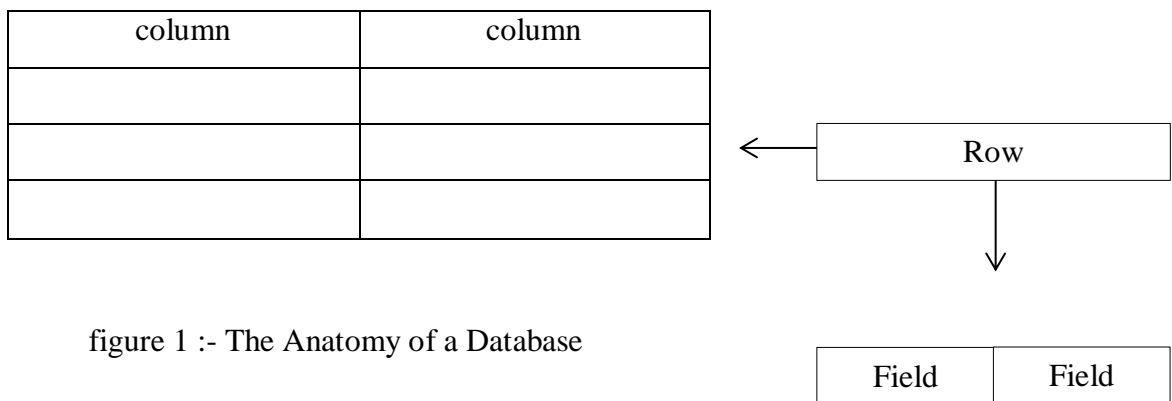
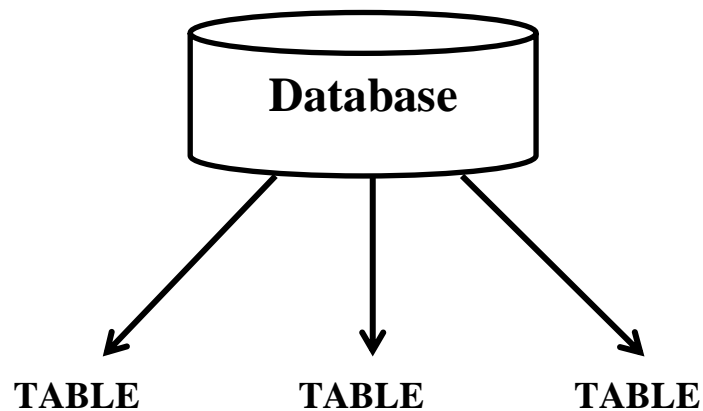


figure 1 :- The Anatomy of a Database

MySQL is more than just a database. It is a system that manages databases. It controls who can use them and how they are manipulated. It logs actions and runs continuously in the background. This is different from what you may be used to. Most people think about Microsoft Access or Lotus Approach what they think about databases. These are databases, but they are not management systems. A DBMS can contain many databases. Users connect -to the database server and issue requests.

The database Server queries its databases and returns the request to the issuers. Databases, such as Approach and Access, are a step down from this type of system. They share their files with multiple users, but there is no interface controlling the connections or answering requests. There are many uses for a DBMS such as MySQL. Uses can range from help desk systems to Web site applications. The important thing to remember is that MySQL is large enough and quick enough to function in almost any situation. Where it finds itself most comfortable is the Enterprise.

6. INTRODUCTION TO PROJECT

A Question and Answer Software is a software that focuses on answering the questions one might put up on an online platform. These are usually implemented by large organizations which aim on implementing a platform where users can clear their doubts about their respective fields. It varies from small scale to large scale or from topic to topic. Many of these platforms may restrict access to either their employees or make it a public one.

The disadvantages of the existing systems are less security, searching efficiency is less, chances of faking answer. One may access these sites/applications from another's system in case they are not public or access the ones which are and post answers which aren't relevant. Sometimes people even lie on such platforms and there might not be any checker to cross-analyze these answers. Now one might also want similar answers to their question which may resolve their query beforehand. Problems which are undertaken are first, making sure that there is more security and safety. Second, helping the users to search answers for similar questions which may answer their question beforehand and even highlight other important points worth knowing.

7. SYSTEM ANALYSIS STUDY REPORT

1. USER INTERVIEW

At the bachelor level, student didn't got the proper reference for their study in computer science; due to this some have lack of knowledge regarding the concept of computers.

When we try to find users view regarding the computer science websites then communication done as:

Q1: Are Qurit website user friendly?

Ans: Developers think that the websites are done in higher languages. But the new user can't comprehend such websites.

Q2: Is the website's easy to navigate?

Ans: Yes, the website UI is very simple and hence makes it easy for the user to navigate across the website.

Q3: Does the website allow user to share their knowledge?

Ans: Yes, any end user can share their thoughts, feedbacks and comments through this website.

Q4: Does all internet browsers support this site?

Ans: This website has passed cross browser test and hence, can run on any browser.

2. PROBLEM DEFINATION

Question and Answer systems play an extremely important role in our day to day life as it helps one gain information and find answers to any sort of query one might have. The users put up questions expecting an answer and also go through previously asked questions where they might find their answer or even get more information about a similar topic. As the increase in population is so rapid and with the number of questions people may have, it's highly unlikely that there can be a question no one can answer at all. Also, we know that altruism does not want every user to just give the answers or to have answer with good quality that too with a less waiting time Internet being such an important source of information, which has such vast amount of data and its constantly increasing with no pause, the users mainly rely on it to find answers from search engines.

8. HARDWARE & SOFTWARE REQUIREMENT'S

HARDWARE REQUIREMENTS: -

The Specified hardware requirements are as follows:

1. The CPU must contain all IIS services
2. It must be Intel I3 processor.
3. It must have 128MB or above RAM capacity.
4. Internet Connection.
5. System Bus: - 32 BIT
6. HDD: - 40GB

SOFTWARE REQUIREMENTS: -

The Specified software requirements are as follows:

1. The operating system must be WIN XP, WIN 7 or above because the required speed is very fast so that the user can operate it easily.
2. Micro media Dreamweaver, PHP
3. Xampp Server
4. MySQL Database
5. Web Server: Internet Explorer 6.0 or any HTTP browser.

9. SCOPE OF SYSTEM

A Question and Answer Software is a software that focuses on answering the questions one might put up on an online platform. These are usually implemented by large organizations which aim on implementing a platform where users can clear their doubts about their respective fields. It varies from small scale to large scale or from topic to topic. Many of these platforms may restrict access to either their employees or make it a public one.

The disadvantages of the existing systems are less security, searching efficiency is less, chances of faking answer. One may access these sites/applications from another's system in case they are not public or access the ones which are and post answers which aren't relevant. Sometimes people even lie on such platforms and there might not be any checker to cross-analyze these answers. Now one might also want similar answers to their question which may resolve their query beforehand. Problems which are undertaken are first, making sure that there is more security and safety. Second, helping the users to search answers for similar questions which may answer their question beforehand and even highlight other important points worth knowing.

10. DATA FLOW DIAGRAMS (DFD's)

The DFD is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data generated by the system. The main reason why the DFD technique is so popular is probable because of the fact that DFD is very simple formalism- it is simple to understand and use. A DFD model uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among these functions.

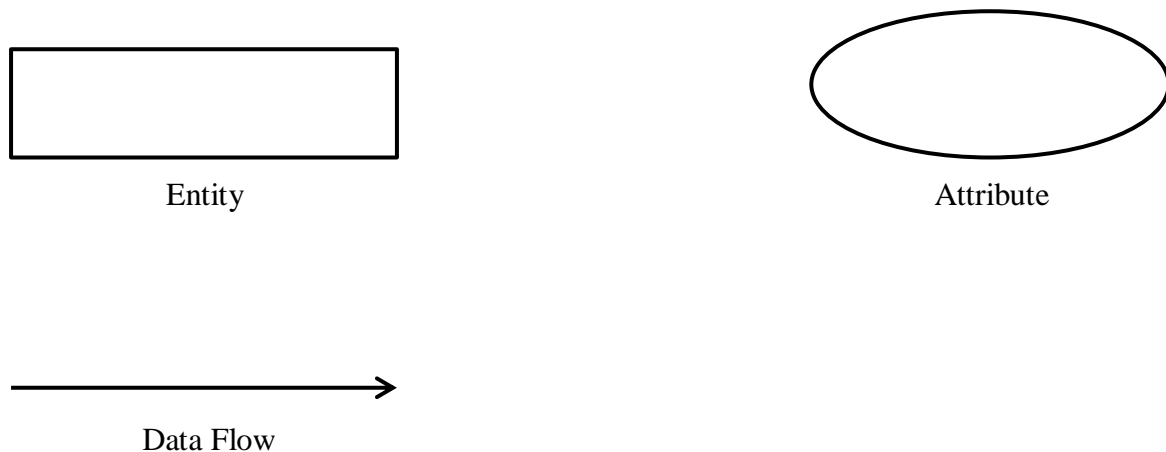


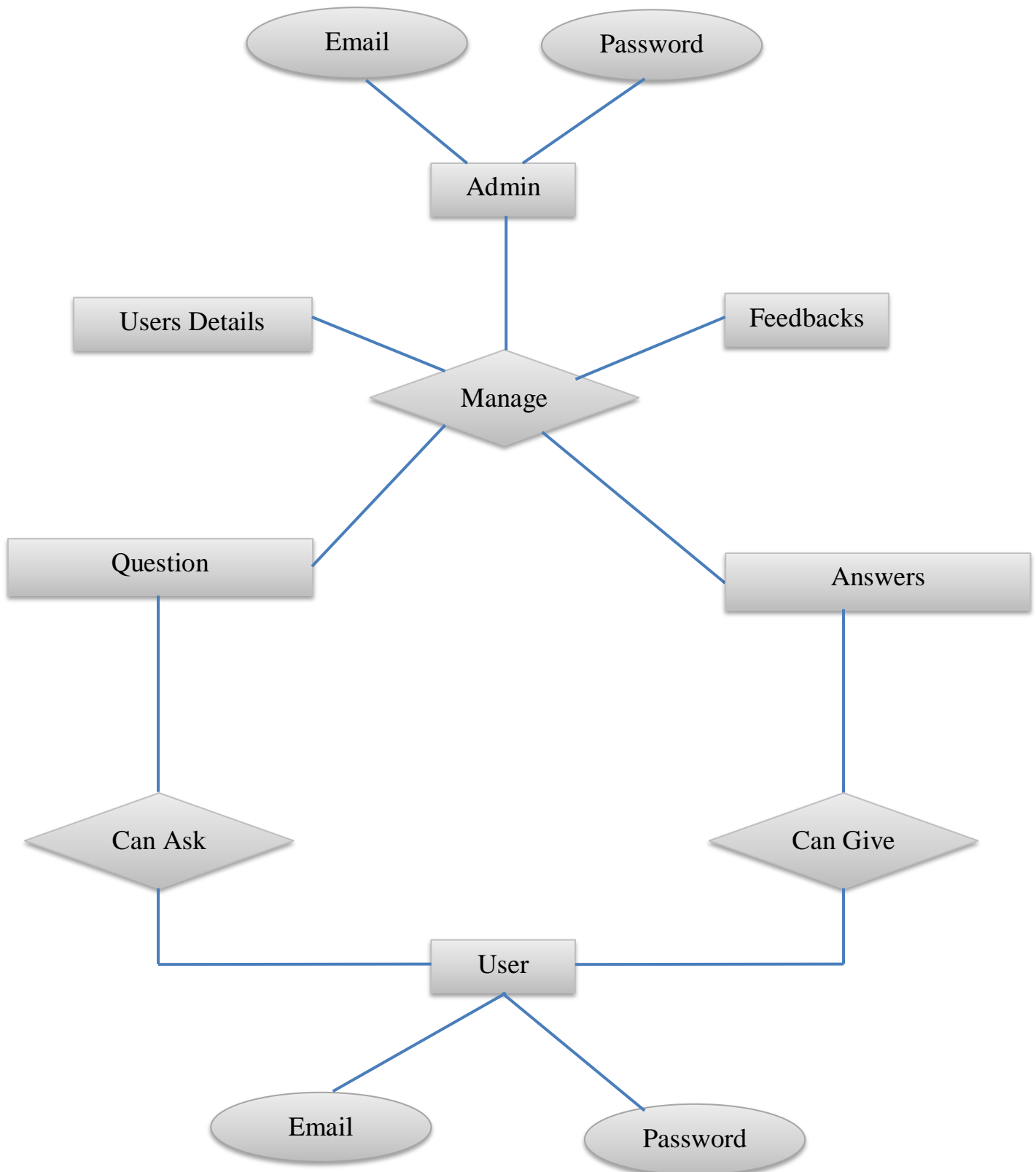
Fig., Symbols used for designing DFD's

Starting with a set of high level functions that a system performs, a DFD model hierarchically represents various sub functions. In fact, any hierarchical model is a simple to understand.

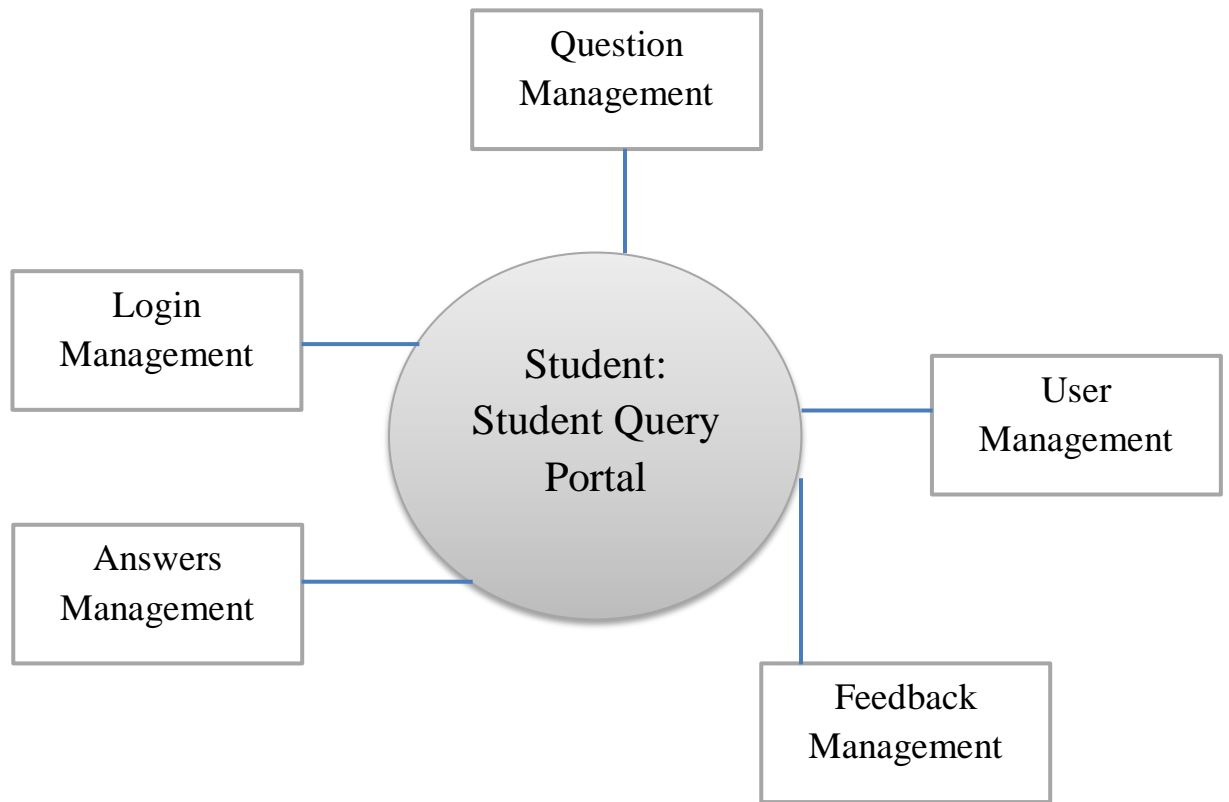
Human mind is such that it can easily understand any hierarchical model of a system - because in a hierarchical model, starting with a very simple and abstract model of a system, different details of the system can be slowly introduced through different hierarchies.

- **Attribute Symbol:** An attribute is represented using a circle.
- **Entity Symbol:** External entity symbol is represented by a rectangle.
- **Data Flow Symbol:** Data flow symbol is represented by an arc or arrow.

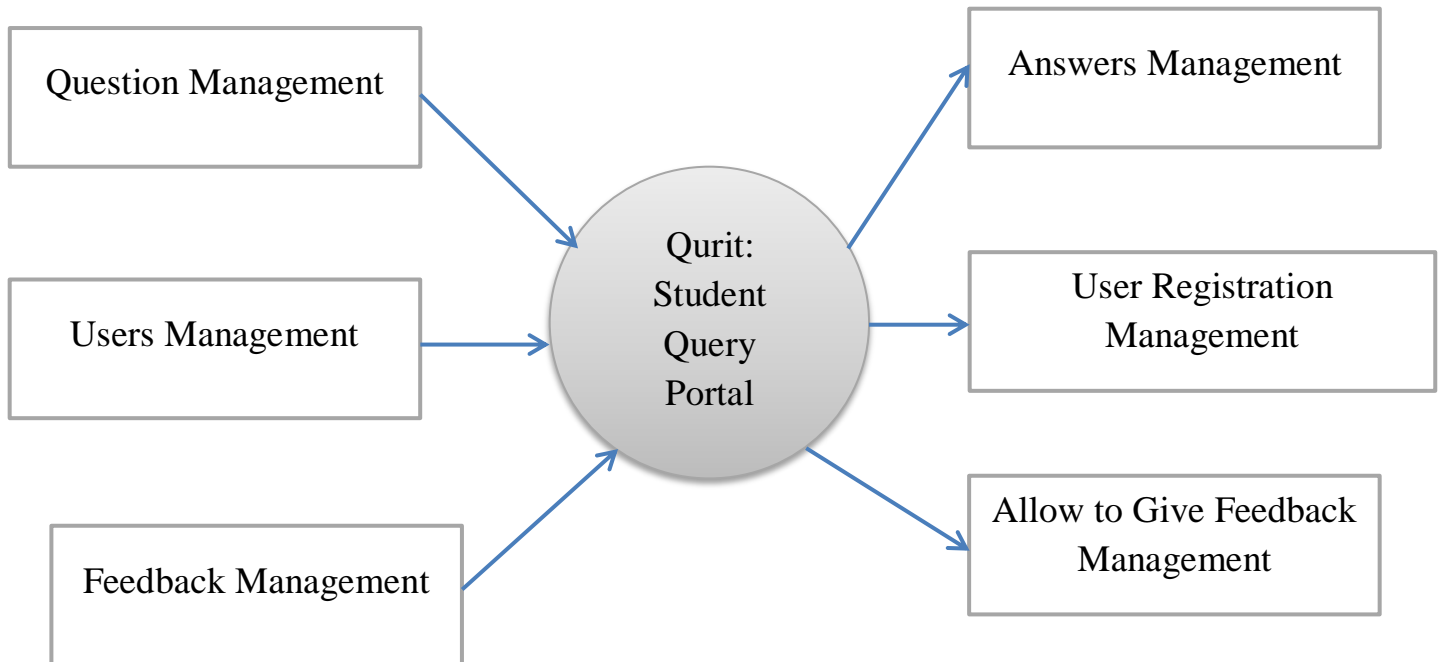
ER DIAGRAM



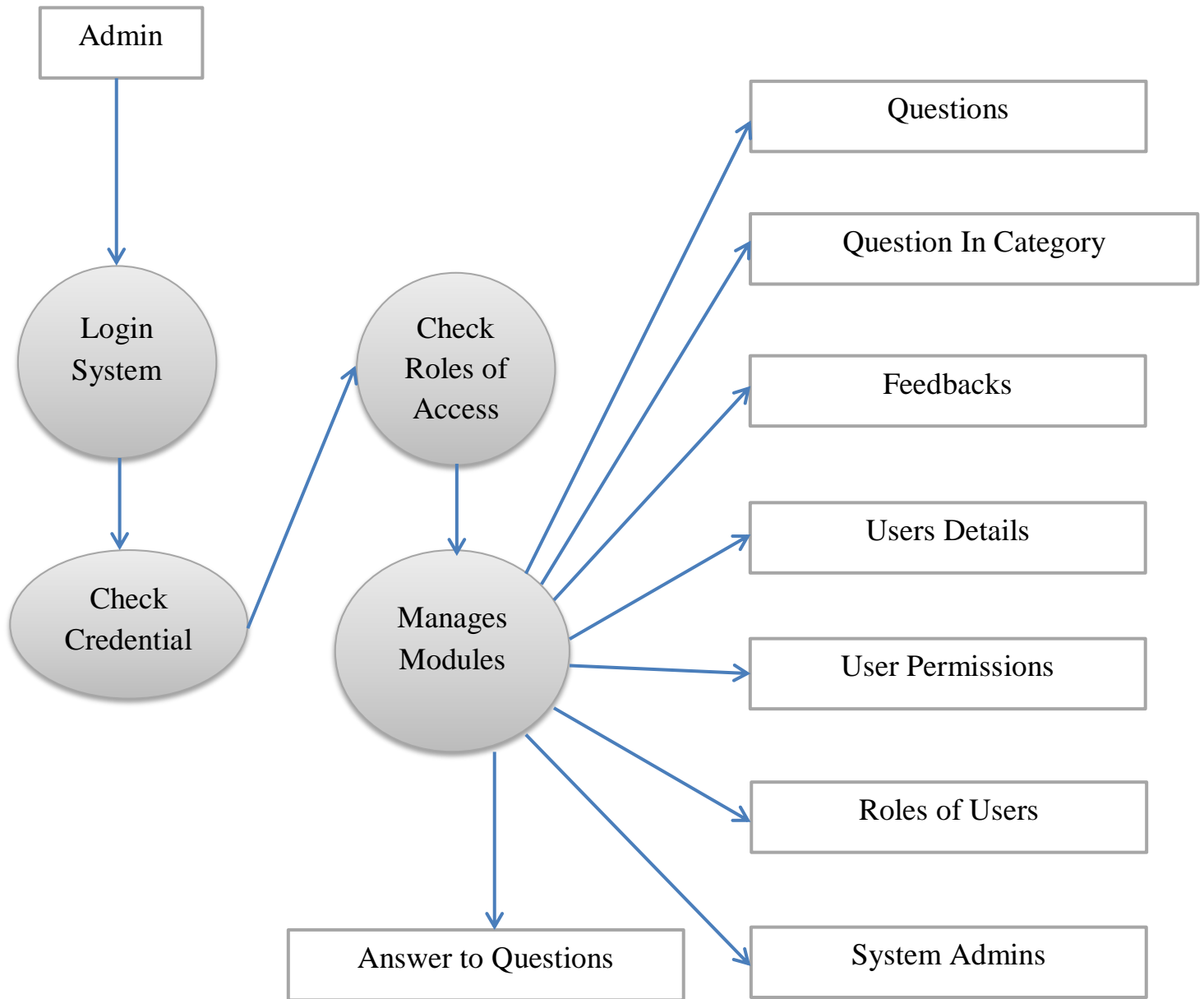
Zero Level DFD – Result Analyzer



First Level DFD – Result Analyzer



Second Level DFD – Result Analyzer



DATA DICTIONARY

Every DFD model of a system must be accompanied by a data dictionary stock. A data dictionary lists all data items appearing in the DFD model of a system. The data items listed include all data flows and the contents of all data stores appearing on all the DFDs in the DFD model of a system. It may be recalled that the DFD model of a system typically consist of several DFDs, namely level 0 DFD, level 1 DFD, level 2 DFDs, etc. However, a single data dictionary should capture all the data appearing in all the DFDs constituting the DFD model of a system. A data dictionary lists the purpose of all data items and definition of all composite data items in terms of their component data items

A data dictionary item plays very important role in any software development process because of the following reasons:

1. A data dictionary provide a standard terminology for all relevant for use by all engineers working in the project. A consistent vocabulary for data items is very important, since in large projects different engineers have a tendency to use different terms to refer to the same data, which unnecessarily cause confusion.
2. The data dictionary provides the analyst with a means to determine the definition of different data structure in terms of their component elements.

DATA DICTIONARY TABLE

category table:

Field Name	Data Type	Constraints
id	int(11)	Primary
name	varchar(255)	no

like table:

Field Name	Data Type	Constraints
id	int(11)	Primary
user_id	int(11)	Index
question_id	int(11)	Index

questions table:

Field Name	Data Type	Constraints
id	int(11)	Primary
question	varchar(255)	no
category_id	int(11)	Index
user_id	int(11)	Index
date_careation	datetime	no

message table:

Field Name	Data Type	Constraints
user_id	int(11)	Primary
question_id	ind(11)	Primary / Index
date	varchar(100)	no
respomse	varchar(500)	no

user table:

Field Name	Data Type	Constraints
id	int(11)	Primary
name	varchar(255)	no
email	varchar(255)	no
password	varchar(255)	no
image	varchar(255)	no
gender	char(1)	no
date_creation	date	no
role	varchar(255)	no

11. FEASIBILITY STUDY

There are three parts of the feasibility study that are follows

- a. Environmental Feasibility
- b. Economic feasibility
- c. Technical Feasibility

a) ENVIRONMENTAL FEASIBILITY:

Any person with minimum computer knowledge can use this Website.

- 1) Windows XP & Windows 7.
- 2) Internet Connection.

b) ECONOMIC FEASIBILITY:

Website requirements are the foundation from which the quality is measured. Then he is ready to spend on the system as much as required. He earns nearly 25 thousand a month. He is spending as much required for system.

c) TECHNICAL FEASIBILITY:

If Website confirms to its requirements, but fails to meet requirements, website quality is suspected. So a good designer should give good website quality considering their factors.

12. TESTING

Testing a program consists of subjecting the program to a set of tests inputs and observing if the program behaves as expected. If the program fails to behave as expected, then the conditions under which failure occurs are noted for a later debugging and correction.

The following are some commonly used terms associated with testing.

1. A failure is a manifestation of an error but the mere presence of an error may not necessarily lead to a failure.
2. A test case is the triplet [I,S,O], where I is the data input to the system, S is the state of the system at which the data is input, and O is the expected output of the system.
3. A test suite is the set of all the test cases with which a given software product is to be tested.

1) BLACK BOX TESTING:

In black-box testing, test cases are designed from an Examination of the input/output values only and no Knowledge of Design or code is required.

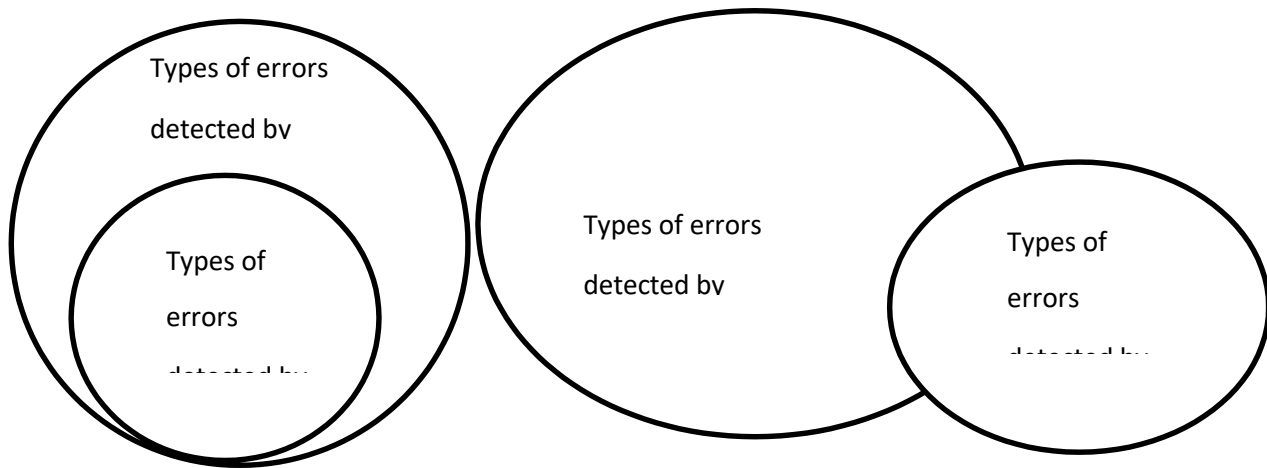
The following are the two main approaches to designing black-box test cases:

- Equivalence class partitioning
- Boundary value analysis

2) WHITE BOX TESTING:

There are several white-box testing strategies. We will discuss some important ones in the following section. Each testing strategy is based on some heuristic. One white-box testing strategy is said to be stronger than another strategy, if all types of errors detected by the first testing strategy (say B) are also detected by the second testing strategy (say A), and the second strategy additionally detects some more types of errors. When two testing strategies detect errors that are different at least with respect to some types of errors, they are then called complimentary. The concepts of stronger and complimentary testing are a schematically Illustrated in the figure. In the practical design of

test cases, if a stronger testing is performed, then the weaker.



A Is stronger testing strategy than B
testing strategies

A & B are complimentary

3) CROSS-BROWSER TESTING:

With regard to scripts, which are the most common usage, the term cross-browser is often confused with multi-browser (see jQuery). Multi-browser scripts can only be expected to work in environments where they have been demonstrated to work (due to assumptions based on observing a subset of browsers). Most publicly available libraries and frameworks are multi-browser scripts and list the environments (typically popular browsers in use at the time and in their default configurations) where they can be expected to work.

Multi-browser scripts virtually always approach obsolescence as new browsers are introduced, features are deprecated and removed, and the author's assumptions are invalidated; therefore, multi-browser scripts have always required periodic maintenance. As the number of browsers and configurations in use has grown, so has the frequency of such maintenance.

13. SOURCE CODE

1. config.php

```
<?php
$server = "localhost";
$user = "root";
$pass = "";
$dbName = "jayesh";
?>
```

2. connection.php

```
<?php
function connexionBdd() {
    require("config.php");
    try {
        $co = new PDO("mysql:host=" . $server . ";dbname=" . $dbName, $user, $pass);
        $co->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    }
    catch(PDOException $e) {
        die("Erreur: " . $e->getMessage());
    }
    return $co;
}
?>
```

3. home.php

```
<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Qurit | Home Page</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <link rel="stylesheet" type="text/css" href="styles/accueil.css?v=<?php echo
time();?>">
    <link rel="stylesheet" type="text/css" href="styles/navbarStyle.css?v=<?php echo
time();?>">
    <script src='https://kit.fontawesome.com/a076d05399.js' crossorigin='anonymous'></script>
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
    <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
</head>
<body>
    <?php
        session_start();
        if (!isset($_SESSION["pseudo"])) {
            header("Location: connexion.php");
            exit();
        }
        require("DB/connexion.php");
        function getCateg($id) {
            $co = connexionBdd();
            $query = $co->prepare("SELECT nom FROM categories WHERE id=:id");
            $query->bindParam(":id", $id);
            $query->execute();
            $result = $query->fetch();
            return $result["nom"];
        }
        function getAuteur($id) {
            $co = connexionBdd();
            $query = $co->prepare("SELECT pseudo FROM utilisateurs WHERE id=:id");
            $query->bindParam(":id", $id);
            $query->execute();
            $result = $query->fetch();
            return $result["pseudo"];
```

```

    }
    function getAvatar($id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT avatar FROM utilisateurs WHERE id=:id");
        $query->bindParam(":id", $id);
        $query->execute();
        $result = $query->fetch();
        return $result["avatar"];
    }
    function changeLikeIcon($like_question_id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT * FROM likes WHERE utilisateur_id=:pseudo_id AND
question_id=:question_id");
        $query->bindParam(":question_id", $like_question_id);
        $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);
        $query->execute();
        $row = $query->rowCount();
        if ($row == 1) {
            return "fas fa-heart";
        } else {
            return "far fa-heart";
        }
    }
}
?>
<?php include("lqnavbar.php"); ?>
<section>
    <div class="container">
        <h3 style="text-align: center; margin-bottom: 2rem;">welcome <?php echo
$_SESSION["pseudo"]; ?> on Qurit!</h3>
        <?php
        $co = connexionBdd();
        $query = $co->query("SELECT * FROM questions");
        $results = $query->fetchAll();
        foreach (array_reverse($results) as $result) {
            $query = $co->prepare("SELECT count(*) from repondre WHERE
questions_id=:question_id");
            $query->bindParam(":question_id", $result[0]);
            $query->execute();
            $responseNumber = $query->fetch();
            $query = $co->prepare("SELECT * from likes WHERE
question_id=:question_id");
            $query->bindParam(":question_id", $result[0]);
            $query->execute();
            $likesNumber = $query->rowCount();
            $likeIcon = changeLikeIcon($result[0]);
            echo "<div class='toast show' role='alert' aria-live='assertive' aria-
atomic='true'>",

```



```

        "<div class='toast-header'>",
        "<strong class='mr-auto'><img class='avatar' src='" .
getAvatar($result[3]) . "'> <a href='profil.php?pseudo_id=$result[3]'
style='color:#000;'>" . getAuteur($result[3]) . "</a> | $responseNumber[0] reply | " .
getCateg($result[2]) . " | <a href='like.php?question_id=$result[0]'><i class='$likeIcon'
style='color:red;'></i></a> $likesNumber</strong>
        <small>$result[4]</small>",
        "</div>",
        "<div class='toast-body'>",
        "<a href='question.php?question_id=$result[0]'
style='color:#000;'>$result[1]</a>",
        "</div>",
        "</div>";
    }
    ?>
</div>
</section>
<?php
    require("footer2.php");
    ?>
</body>
</html>

```

4. admin_page.php

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <title>LiveQuestion</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.bundle.min.js"></scr
ipt>
    <link rel="stylesheet" type="text/css" href="styles/admin.css">
    <script src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
    <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
  </head>
  <body>
    <?php
session_start();
if (!isset($_SESSION["pseudo"])) || $_SESSION["pseudo_role"] != "admin") {
  header("Location: connexion.php");
  exit();
}
?>
<?php include("lqnavbar.php"); ?>
    <section>
      <div class="container">
        <h3 style="text-align: center; margin-bottom: 2rem;"> Delete the unwanted
questions!</h3>
        <?php
require("DB/connexion.php");
function getCateg($id) {
  $co = connexionBdd();
  $query = $co->prepare("SELECT nom FROM categories WHERE id=:id");
  $query->bindParam(":id", $id);
  $query->execute();
  $result = $query->fetch();
  return $result["nom"];
}
function getAuteur($id) {
  $co = connexionBdd();
  $query = $co->prepare("SELECT pseudo FROM utilisateurs WHERE id=:id");
```

```

        $query->bindParam(":id", $id);
        $query->execute();
        $result = $query->fetch();
        return $result["pseudo"];
    }
    function getAvatar($id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT avatar FROM utilisateurs WHERE id=:id");
        $query->bindParam(":id", $id);
        $query->execute();
        $result = $query->fetch();
        return $result["avatar"];
    }
    function changeLikeIcon($like_question_id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT * FROM likes WHERE utilisateur_id=:pseudo_id
AND question_id=:question_id");
        $query->bindParam(":question_id", $like_question_id);
        $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);
        $query->execute();
        $row = $query->rowCount();
        if ($row == 1) {
            return "fas fa-heart";
        } else {
            return "far fa-heart";
        }
    }
    }
    $co = connexionBdd();
    $query = $co->query("SELECT * FROM questions");
    $results = $query->fetchAll();
    foreach (array_reverse($results) as $result) {
        $query = $co->prepare("SELECT count(*) from repondre WHERE
questions_id=:question_id");
        $query->bindParam(":question_id", $result[0]);
        $query->execute();
        $responseNumber = $query->fetch();
        $query = $co->prepare("SELECT * from likes WHERE question_id=:question_id");
        $query->bindParam(":question_id", $result[0]);
        $query->execute();
        $likesNumber = $query->rowCount();
        $likeIcon = changeLikeIcon($result[0]);
        echo "<div class='toast show' role='alert' aria-live='assertive' aria-
atomic='true'>",
            "<div class='toast-header'>",
            "<strong class='mr-auto'><img class='avatar' src='" .
getAvatar($result[3]) . "'> <a href='profil.php?pseudo_id=$result[3]'>" .
getAuteur($result[3]) . "</a> | $responseNumber[0] reply | " . getCateg($result[2]) . " |

```

```

<a href='like.php?question_id=$result[0] '><i class='$likeIcon'></i></a>
$likesNumber</strong>
    <small>$result[4]</small>",
    "<a href='delete.php?question_id=$result[0] '><button class=''
style='    background: #fdc500;' type='button'>
DELETE</button></a>",
    "</div>",
    "<div class='toast-body'>",
    "<a href='question.php?question_id=$result[0] '>$result[1]</a>",
    "</div>",
    "</div>";
}
?>
</div>
</section>
</body>
</html>

```

5. connection.php

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Qurit | Login</title>
    <link rel="stylesheet" type="text/css" href="styles/connexionStyle.css?v=<?php echo
time();?>">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
    <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
  </head>
  <body>
    <?php
require("DB/connexion.php");
session_start();
$co = connexionBdd();
$message = "";
if (isset($_POST["submit"])) {
    $pseudo = $_POST["pseudo"];
    $password = hash('sha256', $_POST["password"]);
    $query = $co->prepare("SELECT * FROM utilisateurs WHERE pseudo=:pseudo and
mot_de_passe=:password");
    $query->bindParam(":pseudo", $pseudo);
    $query->bindParam(":password", $password);
    $query->execute();
    $result = $query->fetchAll();
    $rows = $query->rowCount();
    if ($rows == 1) {
        $_SESSION["pseudo"] = $pseudo;
        $_SESSION["pseudo_id"] = $result[0]["id"];
        $_SESSION["pseudo_role"] = $result[0]["role"];
        $_SESSION["pseudo_email"] = $result[0]["email"];
        $_SESSION["pseudo_avatar"] = $result[0]["avatar"];
        header("Location: admin.php");
    } else {
        $message = "User Name or Password is incorrect";
    }
}
```

```

}
?>
<div class="wrapper">
  <div class="loginBox">
    <h1>LOGIN</h1>
    <h3><?php echo $message; ?></h3>
    <form action="" method="post">
      <p>User Id</p>
      <input type="text" name="pseudo" placeholder="Enter your user id">
      <p>Password</p>
      <input type="password" name="password" placeholder="Enter password">
      <input type="submit" name="submit" value="LOGIN">
      <p style="text-align:center;"><span class="lienExterne">you didn't have
account? <a href="inscription.php">Register!</span></a>
      </p>
    </form>
  </div>
</div>
</body>
</html>

```

6. delete.php

```
<?php
session_start();
if (!isset($_SESSION["pseudo"]) || $_SESSION["pseudo_role"] != "admin") {
    header("Location: connexion.php");
    exit();
}
require("DB/connexion.php");
$co = connexionBdd();
$query = $co->prepare("DELETE FROM likes WHERE question_id=:question_id");
$query->bindParam(":question_id", $_GET["question_id"]);
$query->execute();
$query = $co->prepare("DELETE FROM repondre WHERE questions_id=:question_id");
$query->bindParam(":question_id", $_GET["question_id"]);
$query->execute();
$query = $co->prepare("DELETE FROM questions WHERE id=:question_id");
$query->bindParam(":question_id", $_GET["question_id"]);
$query->execute();
if ($query) {
    header("Location: admin.php");
} else {
    echo "Error";
}
?>
```

7. footer.php

```
<head>
  <link rel="stylesheet" type="text/css" href="styles/footerStyle.css?v=<?php echo
time();?>">
</head>
<footer>
  <div class="borderFooter"></div>
  <div class="container">
    <div class="row">
      <div class="col-md-8">
        <p>© 2023 <b>Privacy Policy</b> and <b>Terms of Service</b>
        </p>
      </div>
      <div class="col-md-4">
        <ul>
          <li><a href=""></a></li>
          <li><a href=""></a></li>
          <li><a href=""></a></li>
          <li><a href=""></a></li>
        </ul>
      </div>
    </div>
  </div>
</footer>
```


8. footer2.php

```
<head>
  <link rel="stylesheet" type="text/css" href="styles/footerStyle.css?v=<?php echo
time();?>">
</head>
<footer style="background: #eeeeee;">
  <div class="borderFooter"></div>
  <div class="container">
    <div class="row">
      <div class="col-md-8">
        <p>© 2023 <b>Privacy Policy</b> and <b>Terms of Service</b>
        </p>
      </div>
      <div class="col-md-4">
        <ul>
          <li><a href=""></a></li>
          <li><a href=""></a></li>
          <li><a href=""></a></li>
          <li><a href=""></a></li>
        </ul>
      </div>
    </div>
  </div>
</footer>
```

9. index.php

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Qurit</title>
  <link rel="stylesheet" type="text/css" href="styles/indexStyle.css?v=<?php echo
time();?>">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
  <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
  <script type="text/javascript" src="script/indexScript.js"></script>
  <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
</head>
<body>
  <header>
    <?php
      require("navbar.php");
    ?>
    <div class="container headerBlock">
      <div class="figureAccueil">
        <div class="figureAccueil2"></div>
      </div>
      <div class="row">
        <div class="col-md-6">
          <h1>Qurit is a place to gain and share knowledge.</h1>
          <p>It's a platform to ask questions and connect with people who
contribute unique insights and
          quality answers.</p>
          <a href="redirection.php" class="btn bouton headerBouton"><span
class="headButton">LogIn</span></a>
        </div>
        <div class="col-md-6">
          
        </div>
      </div>
    </div>
  </header>
  <div class="borderBottomHeader"></div>
  <section id="section1">
```

```

<div class="container">
  <div class="row">
    <div class="col-md-4" style="text-align: center;">
      
      <h5>Post the questions</h5>
      <p>Post the question eazly<br> only you wan't to sign up.</p>
    </div>
    <div class="col-md-4" style="text-align: center;">
      
      <h5>Asnwer the questions</h5>
      <p>Answer the posted questions <br> if you konw the answer.</p>
    </div>
    <div class="col-md-4" style="text-align: center;">
      
      <h5>Find your quries eazly</h5>
      <p>Find your quries in few seconds<br> only you wan't to sign up.
      </p>
    </div>
  </div>
</div>
</section>
<section id="Section3">
  <div class="embed-responsive video_player embed-responsive-16by9">
    <iframe class="embed-responsive-item" src="img/main_v.mp4"></iframe>
  </div>
</section>
<section id="Section4">
  <div class="contenuApp">
    <div class="texteApp">
      <h4>About Us</span></h4>
      <br>
      <p>Qurit: student query portal is a platform where student post the query
as a question and other
          users give the answer easily. It is help to student in many type of
query like theoretical, coding,
          mathematics, etc.
      <br>
      This is a introduction to new era. This website is a thought of
bringing a give new choice to
          student solve their queries in this new digital generation.
      </p>
    </div>
  </div>
</section>
<div class="borderBottomSection4"></div>
<?php
  require("footer.php");

```

```
        ?>  
</body>  
</html>
```

10. sign_in.php

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Qurit | Sing In</title>
    <link rel="stylesheet" type="text/css" href="styles/inscriptionStyle.css?v=<?php
echo time();?>">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
    <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
  </head>
  <body>
    <?php
require("DB/connexion.php");
function has_special_chars($string) {
    return preg_match('/[^a-zA-Z\d]/', $string);
}
function passwordVerification($pass) {
    if (strlen($pass) >= 8 && has_special_chars($pass) > 0) {
        return true;
    } else {
        return false;
    }
}
function userExist($user) {
    $co = connexionBdd();
    $query = $co->prepare("SELECT * FROM utilisateurs WHERE pseudo=:pseudo");
    $query->bindParam(":pseudo", $user);
    $query->execute();
    $row = $query->rowCount();
    if ($row > 0) {
        return true;
    } else {
        return false;
    }
}
}
```

```

function emailExist($mail) {
    $co = connexionBdd();
    $query = $co->prepare("SELECT * FROM utilisateurs WHERE email=:email");
    $query->bindParam(":email", $mail);
    $query->execute();
    $row = $query->rowCount();
    if ($row > 0) {
        return true;
    } else {
        return false;
    }
}

$message = "";
if (isset($_POST["submit"])) {
    if (isset($_POST["pseudo"], $_POST["email"], $_POST["password"],
$_POST["passwordverif"], $_POST["genre"])) {
        $pseudo = $_POST["pseudo"];
        $email = $_POST["email"];
        $password = $_POST["password"];
        $passwordverif = $_POST["passwordverif"];
        $passwordhash = hash('sha256', $_POST["password"]);
        $genre = $_POST["genre"];
        date_default_timezone_set('Europe/Paris');
        $date = date('Y-m-d H:i:s');
        if ($password == $passwordverif) {
            if (passwordVerification($password)) {
                if (userExist($pseudo) == false) {
                    if (emailExist($email) == false) {
                        $co = connexionBdd();
                        $query = $co->prepare("INSERT into utilisateurs (pseudo,
email, mot_de_passe, avatar, genre, date_inscription, role) VALUES (:pseudo, :email,
:password,
'https://www.tenforums.com/geek/gars/images/2/types/thumb_15951118880user.png', :genre,
:date, 'membre')");
                        $query->bindParam(":pseudo", $pseudo);
                        $query->bindParam(":email", $email);
                        $query->bindParam(":password", $passwordhash);
                        $query->bindParam(":genre", $genre);
                        $query->bindParam(":date", $date);
                        $query->execute();
                        if ($query) {
                            $message = "Your account has been successfully
registered";
                        }
                    } else {
                        $message = "The email address is already in use";
                    }
                }
            }
        }
    }
}

```

```

        } else {
            $message = "Username is already in use";
        }
    } else {
        $message = "The password must contain 8 characters including 1
special character";
    }
    } else {
        $message = "Passwords do not match";
    }
}
}
?>
<div class="wrapper">
    <div class="loginBox">
        <h1>Sign In</h1>
        <h3><?php echo $message; ?></h3>
        <form action="" method="post">
            <input type="text" name="pseudo" placeholder="Enter User Name">
            <input type="email" name="email" placeholder="Enter Email Id">
            <p>Genre :</p>
            <div class="radioInput">
                <label for="Male">Male</label>
                <input type="radio" style="width: 10%;" name="genre" value="M">
                <label for="Female">Female</label>
                <input type="radio" style="width: 10%;" name="genre" value="F">
            </div>
            <input type="password" name="password" placeholder="Enter The Password">
            <input type="password" name="passwordverif" placeholder="Confirm The
Password">
            <input type="submit" name="submit" value="Sing In">
            <p class="para">You already have account? <a href="connexion.php"
>LogIn!</a></p>
        </form>
    </div>
</div>
</html>

```

11. like.php

```
<?php
session_start();
if (!isset($_GET["question_id"], $_SESSION["pseudo_id"])) {
    header("Location: accueil.php");
} else {
    $pseudo_id = $_SESSION["pseudo_id"];
    $question_id = $_GET["question_id"];
}
require("DB/connexion.php");
$co = connexionBdd();
$query = $co->prepare("SELECT * FROM likes WHERE utilisateur_id=:pseudo_id AND
question_id=:question_id");
$query->bindParam(":pseudo_id", $pseudo_id);
$query->bindParam(":question_id", $question_id);
$query->execute();
$row = $query->rowCount();
if ($row == 1) {
    $query = $co->prepare("DELETE FROM likes WHERE utilisateur_id=:pseudo_id AND
question_id=:question_id");
    $query->bindParam(":pseudo_id", $pseudo_id);
    $query->bindParam(":question_id", $question_id);
    $query->execute();
} else {
    $query = $co->prepare("INSERT INTO likes (utilisateur_id, question_id) VALUES
(:pseudo_id, :question_id)");
    $query->bindParam(":pseudo_id", $pseudo_id);
    $query->bindParam(":question_id", $question_id);
    $query->execute();
}
header("Location:" . $_SERVER['HTTP_REFERER']);
exit();
?>
```

12. logout.php

```
<?php
session_start();
$_SESSION = array();
header("Location: connexion.php");
?>
```


13. navbar.php

```
<link rel="stylesheet" type="text/css" href="styles/navbarStyle.css?v=<?php echo
time();?>">
<script src='https://kit.fontawesome.com/a076d05399.js' crossorigin='anonymous'></script>
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    
    <a class="navbar-brand" href="index.php" style="color:#000;">Qurit</a>
    <div class="collapse navbar-collapse" id="navbarColor">
        <ul class="navbar-nav login-btn" style="align-items: center;">
            <li class="nav-item">
                <a class="nav-link nav-anc" href="accueil.php" style="color:#000; font-
weight: 500;">Questions</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="nouvellequestion.php" style="color:#000; font-
weight: 500;">Post question</a>
            </li>
            <li>
                <?php
                    if ($_SESSION["pseudo_role"] == "admin") {
                        echo " <a class='nav-link' href='admin.php' style='color:#000; font-
weight: 500;'>Delete Question</a>";
                    }
                <?>
            </li>
        </ul>
        <ul class="navbar-nav login-btn">
            <li class="nav-item dropdown" style="margin-left: 7rem;">
                <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="#"
role="button" aria-haspopup="true"
                aria-expanded="false" style="color:#000; font-weight: 500;"><i
class='fas fa-user-alt'></i></a>
                <div class="dropdown-menu">
                    <a class="dropdown-item" href="profil.php" style="color:#000; font-
weight: 500;">View Profile</a>
                    <a class="dropdown-item" href="logout.php" style="color:#000; font-
weight: 500;">Log Out</a>
                </div>
            </li>
        </ul>
    </div>
</nav>
```

14. question.php

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <title>LiveQuestion</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.bundle.min.js"></scr
ipt>
    <link rel="stylesheet" type="text/css" href="styles/modifier.css">
    <script src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
    <link rel="icon" href="img/favicon.png" type="image/png">
  </head>
  <body>
    <?php
    session_start();
    if (!isset($_SESSION["pseudo"]) && !isset($_SESSION["pseudo_id"])) {
      header("Location: connexion.php");
      exit();
    }
    include("lqnavbar.php");
    require("DB/connexion.php");
    $messageP = "";
    $messageE = "";
    $messagePA = "";
    $messageA = "";
    $co = connexionBdd();
    function pseudoUnique($pseudoStr) {
      $co = connexionBdd();
      $query = $co->prepare("SELECT * FROM utilisateurs WHERE pseudo=:pseudo");
      $query->bindParam(":pseudo", $pseudoStr);
      $query->execute();
      $row = $query->rowCount();

      if ($row == 1) {
        return false;
      } else {
        return true;
      }
    }
  </body>
</html>
```

```

}
function emailUnique($emailStr) {
    $co = connexionBdd();
    $query = $co->prepare("SELECT * FROM utilisateurs WHERE email=:email");
    $query->bindParam(":email", $emailStr);
    $query->execute();
    $row = $query->rowCount();
    if ($row == 1) {
        return false;
    } else {
        return true;
    }
}
function has_special_chars($string) {
    return preg_match('/[^a-zA-Z\d]/', $string);
}
function passwordVerification($pass) {
    if (strlen($pass) >= 8 && has_special_chars($pass) > 0) {
        return true;
    } else {
        return false;
    }
}
function oldPasswordVerification($pass) {
    $pass = hash('sha256', $pass);
    $co = connexionBdd();
    $query = $co->prepare("SELECT * FROM utilisateurs WHERE id=:id AND
mot_de_passe=:pass");
    $query->bindParam(":id", $_SESSION["pseudo_id"]);
    $query->bindParam(":pass", $pass);
    $query->execute();
    $row = $query->rowCount();
    if ($row == 1) {
        return true;
    } else {
        return false;
    }
}
if (isset($_POST["submitPseudo"])) {
    if (!empty($_POST["pseudo"])) {
        if (pseudoUnique($_POST["pseudo"])) {
            $query = $co->prepare("UPDATE utilisateurs SET pseudo=:pseudo WHERE
id=:pseudo_id");
            $query->bindParam(":pseudo", $_POST["pseudo"]);
            $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);
            $query->execute();
            if ($query) {

```

```

        $messageP = "Modification effectue";
        $_SESSION["pseudo"] = $_POST["pseudo"];
    }
} else {
    $messageP = "Ce pseudo existe déjà";
}
} else {
    $messageP = "Les champs ne peuvent pas être vides";
}
}
if (isset($_POST["submitEmail"])) {
    if (!empty($_POST["email"])) {
        if (emailUnique($_POST["email"])) {
            $query = $co->prepare("UPDATE utilisateurs SET email=:email WHERE
id=:pseudo_id");
            $query->bindParam(":email", $_POST["email"]);
            $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);
            $query->execute();
            if ($query) {
                $messageE = "Modification effectue";
                $_SESSION["pseudo_email"] = $_POST["email"];
            }
        } else {
            $messageE = "Cet email existe déjà";
        }
    } else {
        $messageE = "Les champs ne peuvent pas être vides";
    }
}
if (isset($_POST["submitPass"])) {
    if (!empty($_POST["passOld"]) && !empty($_POST["passNew"])) {
        if ($_POST["passOld"] != $_POST["passNew"]) {
            if (passwordVerification($_POST["passNew"])) {
                if (oldPasswordVerification($_POST["passOld"])) {
                    $passNew = hash('sha256', $_POST["passNew"]);
                    $query = $co->prepare("UPDATE utilisateurs SET mot_de_passe=:pass
WHERE id=:id");
                    $query->bindParam(":id", $_SESSION["pseudo_id"]);
                    $query->bindParam(":pass", $passNew);
                    $query->execute();
                    if ($query) {
                        $messagePA = "Modification effectuée";
                    }
                } else {
                    $messagePA = "Votre ancien mot de passe est incorrect";
                }
            } else {

```

```

        $messagePA = "Votre nouveau mot de passe doit contenir 8 caractères
et au moins 1 caractère special";
    }
    } else {
        $messagePA = "Votre nouveau mot de passe ne peut pas être le même que
l'ancien";
    }
    } else {
        $messagePA = "Les champs ne peuvent pas être vides";
    }
}
if (isset($_POST["submitAvatar"])) {
    if (!empty($_POST["avatar"])) {
        $query = $co->prepare("UPDATE utilisateurs SET avatar=:avatar WHERE
id=:pseudo_id");
        $query->bindParam(":avatar", $_POST["avatar"]);
        $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);
        $query->execute();
        if ($query) {
            $messageA = "Modification effectuée";
            $_SESSION["pseudo_avatar"] = $_POST["avatar"];
        }
    } else {
        $messageA = "Les champs ne peuvent pas être vides";
    }
}
?>
<div class="container">
    <div class="boxModif">
        <h1><span>Modifier mon profil :</span></h1>
        <h3>Modifier mon pseudo:</h3>
        <form action="" method="POST">
            <input type="text" name="pseudo" value="<?php echo $_SESSION['pseudo']
?>">
            <input type="submit" name="submitPseudo" value="Modifier">
            <p><?php echo $messageP; ?></p>
        </form>
        <h3>Modifier mon email :</h3>
        <form action="" method="POST">
            <input type="email" name="email" value="<?php echo
$_SESSION['pseudo_email'] ?>">
            <input type="submit" name="submitEmail" value="Modifier">
            <p><?php echo $messageE; ?></p>
        </form>
        <h3>Modifier mon avatar (lien web uniquement) :</h3>
        <form action="" method="POST">

```

```

        <input type="url" name="avatar" value="<?php echo
$_SESSION['pseudo_avatar'] ?>">
        <input type="submit" name="submitAvatar" value="Modifier">
        <p><?php echo $messageA; ?></p>
    </form>
    <h3>Modifier mon mot de passe :</h3>
    <form action="" method="POST">
        <label for="passOld">Votre ancien mot de passe</label>
        <input type="password" name="passOld">
        <label for="passNew">Votre nouveau mot de passe</label>
        <input type="password" name="passNew">
        <input type="submit" name="submitPass" value="Modifier">
        <p><?php echo $messagePA; ?></p>
    </form>
</div>
</div>
</body>
</html>

```

15. navbar.php

```

<link rel="stylesheet" type="text/css" href="styles/navbarStyle.css?v=<?php echo
time();?>">
<nav class="navbar navbar-expand-lg fixed-top navbar-dark bg-dark">
    
    <a class="navbar-brand" href="index.php">Qurit</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarColor"
        aria-controls="navbarColor" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarColor">
        <ul class="navbar-nav login-btn" style="margin-left: 75%;">
            <li class="nav-item dropdown" >
                <a class="btn btn-secondary" type="submit" href="redirection.php">LogIn</a>
            </li>
        </ul>
    </div>
</nav>

```

16. add_new_question.php

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>LiveQuestion</title>
    <link rel="stylesheet" type="text/css" href="styles/nouvellequestion.css?v=<?php
echo time();?>">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
    <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
  </head>
  <body>
    <?php
      session_start();
      if (!isset($_SESSION["pseudo"])) {
        header("Location: connexion.php");
        exit();
      }

      require("DB/connexion.php");
      $message = "";

      //Ajout de la question
      $co = connexionBdd();

      if (isset($_POST["submit"])) {
        if (isset($_POST["title"], $_POST["categorie"])) {
          $pseudo_id = $_SESSION["pseudo_id"];
          $title = $_POST["title"];
          $categorie = $_POST["categorie"];
          date_default_timezone_set('Europe/Paris');
          $date = date('Y-m-d H:i:s');

          $query = $co->prepare("INSERT into questions (titre, categorie_id,
auteur_id, date_creation) VALUES (:title, :categorie, :pseudo_id, :date)");
          $query->bindParam(":title", $title);
          $query->bindParam(":categorie", $categorie);
```

```

        $query->bindParam(":pseudo_id", $pseudo_id);
        $query->bindParam(":date", $date);
        $query->execute();
        if ($query) {
            $message = "Your question has been correctly sent !";
        }
    }
}
?>
<?php include("lqnavbar.php"); ?>
<h3>Add a new question:</h3>
<form action="" method="POST">
    <div class="form-outline mb-4">
        <label class="col-form-label" for="title">Write a question:</label>
        <input type="text" name="title" id="title" class="form-control">
    </div>
    <div class="form-outline mb-4">
        <label class="form-label" for="categorie">Question Catagory:</label>
        <select class="custom-select" name="categorie" id="categorie">
            <?php
                //Récuperation des categories
                $query = $co->prepare("SELECT * FROM categories");
                $query->execute();
                $results = $query->fetchAll();
                foreach ($results as $result) {
                    echo "<option value='$result[0]'">$result[1]</option>";
                }
            ?>
        </select>
    </div>
    <input type="submit" name="submit" class="rosebutton" style="background: #fdc500;">
</form>
<p><?php echo $message; ?></p>
</body>
</html>

```


17. profile.php

```
<!DOCTYPE html>
<html lang="fr">

<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>LiveQuestion</title>
    <link rel="stylesheet" type="text/css" href="styles/profil.css?v=<?php echo
time();?>">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css?v=<?php echo
time();?>">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css?v=<?php echo time();?>">
    <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
    <script src="script/profilScript.js"></script>
</head>

<body style="background:#eee;">
    <!-- NAVBAR -->
    <?php
        session_start();
        if (!isset($_SESSION["pseudo"])) {
            header("Location: connection.php");
            exit();
        }

        if (isset($_GET["pseudo_id"])) {
            $id = $_GET["pseudo_id"];
        } else {
            $id = $_SESSION["pseudo_id"];
        }

        //Si l'utilisateur consulte son profil autorisé les modifications
        if ($id == $_SESSION["pseudo_id"]) {
            $modifiable = true;
        } else {
            $modifiable = false;
        }
    </?php>

```

```

include("lqnavbar.php");

require("DB/connexion.php");

function getCateg($id) {
    $co = connexionBdd();
    $query = $co->prepare("SELECT nom FROM categories WHERE id=:id");
    $query->bindParam(":id", $id);
    $query->execute();
    $result = $query->fetch();
    return $result["nom"];
}

function getAuteur($id) {
    $co = connexionBdd();
    $query = $co->prepare("SELECT pseudo FROM utilisateurs WHERE id=:id");
    $query->bindParam(":id", $id);
    $query->execute();
    $result = $query->fetch();
    return $result["pseudo"];
}

function getAvatar($id) {
    $co = connexionBdd();
    $query = $co->prepare("SELECT avatar FROM utilisateurs WHERE id=:id");
    $query->bindParam(":id", $id);
    $query->execute();
    $result = $query->fetch();
    return $result["avatar"];
}

function changeLikeIcon($like_question_id) {
    $co = connexionBdd();
    $query = $co->prepare("SELECT * FROM likes WHERE utilisateur_id=:pseudo_id AND
question_id=:question_id");
    $query->bindParam(":question_id", $like_question_id);
    $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);
    $query->execute();
    $row = $query->rowCount();
    if ($row == 1) {
        return "fas fa-heart";
    } else {
        return "far fa-heart";
    }
}

```

```

//Affichage des informations du profil
$co = connexionBdd();

$query = $co->prepare("SELECT * FROM utilisateurs WHERE id=:id");
$query->bindParam(":id", $id);
$query->execute();
$result = $query->fetch();
?>
<section>
    <div class="container">
        <div class="main-body">
            <div class="row gutters-sm">
                <div class="col-md-12 mb-3">
                    <div class="card">
                        <div class="card-body">
                            <div class="d-flex flex-column align-items-center text-center">
                                
                                <div class="mt-3">
                                    <h4>
                                        <?php echo $result['pseudo']; ?>
                                    </h4>
                                    <p class="text-secondary mb-1">Gender :
                                        <?php if ($result['genre'] == "F") { echo "Female ♀"; }
else { echo "Male ♂"; }; ?>
                                    </p>
                                    <p class="text-muted font-size-sm">Registration date:
                                        <?php echo $result['date_inscription']; ?>
                                    </p>

                                <?php
//Affichage des boutons de modifications
if ($modifiable) {
    ?>
        <div class="modifyDiv">
            <button type="button" class="btn btn-primary
modifyButton"
                onclick="location.href='modifier.php';">Edit Profile
            <i
                class="fas fa-user-cog"></i></button>
            <button type="button" class="btn btn-outline-primary
modifyButton" onclick="submitDelete()">Delete Account <i class="fas fa-user-
times"></i></button>
            <form action="" method="POST" name="deleteForm">
                <input type="text" name="submitDelete"
style="display: none;">

```



```

        $query = $co->prepare("SELECT * FROM questions WHERE auteur_id=:id");
        $query->bindParam(":id", $id);
        $query->execute();
        $results = $query->fetchAll();
        foreach (array_reverse($results) as $result) {
            $query = $co->prepare("SELECT count(*) from repondre WHERE
questions_id=:question_id");
            $query->bindParam(":question_id", $result[0]);
            $query->execute();
            $responseNumber = $query->fetch();
            $query = $co->prepare("SELECT * from likes WHERE
question_id=:question_id");
            $query->bindParam(":question_id", $result[0]);
            $query->execute();
            $likesNumber = $query->rowCount();
            $likeIcon = changeLikeIcon($result[0]);
            echo "<div class='toast show' role='alert' aria-live='assertive' aria-
atomic='true' >",
                "<div class='toast-header' style='width: 100%;'>",
                "<strong class='mr-auto'><img class='avatar' src='" .
getAvatar($result[3]) . "'> <a href='profil.php?pseudo_id=$result[3]'
style='color:#000;'>" . getAuteur($result[3]) . "</a> | $responseNumber[0] reply | " .
getCateg($result[2]) . " | <a href='like.php?question_id=$result[0]'><i class='$likeIcon'
style='color:red;'></i></a> $likesNumber</strong>
                <small>$result[4]</small>",
                "</div>",
                "<div class='toast-body'>",
                "<a href='question.php?question_id=$result[0]'
style='color:#000;'>$result[1]</a>",
                "</div>",
                "</div>";
        }

    ?>

</div>
</section>
<?php
    require("footer2.php");
    ?>
</body>

</html>

```

18. question.php

```
<!DOCTYPE html>
<html lang="fr">

<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>LiveQuestion</title>
    <link rel="stylesheet" type="text/css" href="styles/question.css?v=<?php echo
time();?>">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="script.js"></script>
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css">
    <link rel="icon" type="image/png" sizes="16x16" href="img/favicon-16x16.png">
    <style>
        .toast {
            max-width: 100%;
            max-height: 7rem;
            width: 40%;
        }

        .toast.show {
            margin-top: 2% !important;
        }
        .toast{
            width: 50%;
            margin: auto;
        }
        h3{
            text-align: center;
            padding-top: 2rem;
        }
    </style>
</head>

<body>
    <!-- NAVBAR -->
    <?php
        session_start();
        include("lqnavbar.php");
```

```

?>
<section>
  <div class="container">
    <?php
      if (!isset($_SESSION["pseudo"], $_GET["question_id"])) {
        header("Location: connexion.php");
        exit();
      }

      require("DB/connexion.php");
      $message = "";

      function getAuteur($id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT pseudo FROM utilisateurs WHERE id=:id");
        $query->bindParam(":id", $id);
        $query->execute();
        $result = $query->fetch();
        return $result["pseudo"];
      }

      function getCateg($id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT nom FROM categories WHERE id=:id");
        $query->bindParam(":id", $id);
        $query->execute();
        $result = $query->fetch();
        return $result["nom"];
      }

      function getAvatar($id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT avatar FROM utilisateurs WHERE id=:id");
        $query->bindParam(":id", $id);
        $query->execute();
        $result = $query->fetch();
        return $result["avatar"];
      }

      function changeLikeIcon($like_question_id) {
        $co = connexionBdd();
        $query = $co->prepare("SELECT * FROM likes WHERE utilisateur_id=:pseudo_id
AND question_id=:question_id");
        $query->bindParam(":question_id", $like_question_id);
        $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);

```

```

        $query->execute();
        $row = $query->rowCount();
        if ($row == 1) {
            return "fas fa-heart";
        } else {
            return "far fa-heart";
        }
    }
}

//Affichage de la question
$co = connexionBdd();

$query = $co->prepare("SELECT * FROM questions WHERE id=:id");
$query->bindParam(":id", $_GET["question_id"]);
$query->execute();
$result = $query->fetch();
$query = $co->prepare("SELECT count(*) from repondre WHERE
questions_id=:question_id");
$query->bindParam(":question_id", $result[0]);
$query->execute();
$responseNumber = $query->fetch();
$query = $co->prepare("SELECT * from likes WHERE question_id=:question_id");
$query->bindParam(":question_id", $result[0]);
$query->execute();
$likesNumber = $query->rowCount();
$likeIcon = changelikeIcon($result[0]);
echo "<div class='toast show' role='alert' aria-live='assertive'
style='width:100%;' aria-atomic='true' >",
    "<div class='toast-header' style='width: 100%;'>",
    "<strong class='mr-auto'><img class='avatar' src='" . getAvatar($result[3]) .
"'> <a href='profil.php?pseudo_id=$result[3]' style='color:#000;'>" .
getAuteur($result[3]) . "</a> | $responseNumber[0] reply | " . getCateg($result[2]) . " |
<a href='like.php?question_id=$result[0]'><i class='$likeIcon'
style='color:red;'></i></a> $likesNumber</strong>
<small>$result[4]</small>",
    "</div>",
    "<div class='toast-body'>",
    "<a href='question.php?question_id=$result[0]'
style='color:#000;'>$result[1]</a>",
    "</div>",
    "</div>";
?>

<div class="row" style="    flex-direction: column;">
    <h3 style=" margin: 0rem 1rem;">Answer :</h3>
    <?php
        //Affichage des réponses

```



```

        $query = $co->prepare("SELECT * FROM repondre WHERE questions_id=:id");
        $query->bindParam(":id", $_GET["question_id"]);
        $query->execute();
        $results = $query->fetchAll();
        foreach (array_reverse($results) as $result) {
            echo "<div class='toast toast1 show' role='alert' aria-
live='assertive' aria-atomic='true'>",
                "<div class='toast-header'>",
                "<strong class='mr-auto'><img class='avatar' src='" .
getAvatar($result[0]) . "'> <a href='profil.php?pseudo_id=$result[0]'>" .
getAuteur($result[0]) . "</a></strong>",
                "<small>$result[2]</small>",
                "</div>",
                "<div class='toast-body'>",
                "$result[3]",
                "</div>",
                "</div>";
        }
    ?>
<?php
//Enregistrement de la réponse
if (isset($_POST["submit"])) {
    $reponse = $_POST["reponse"];
    date_default_timezone_set('UTC');
    $date = date('Y-m-d H:i:s');

    $query = $co->prepare("INSERT INTO repondre (utilisateurs_id,
questions_id, date, reponse) VALUES (:pseudo_id, :question_id, :date, :reponse)");
    $query->bindParam(":reponse", $reponse);
    $query->bindParam(":date", $date);
    $query->bindParam(":pseudo_id", $_SESSION["pseudo_id"]);
    $query->bindParam(":question_id", $_GET["question_id"]);
    try {
        $query->execute();
        if ($query) {
            $message = "response not send!";
        }
    } catch (Exception $e) {
        $message = "response has been send!";
    }
}
?>

<!-- FORMULAIRE -->
<div class="">
    <h3>Respond to the question :</h3>

```

```

        <div class="row" style="width: 50%;margin: auto;">
            <form action="" method="post" style="width: 100%; margin-left:
1rem;">
                <label for="reponse">Your answer :</label><br>
                <textarea id="reponse" class="form-control" row="20"
                    style="height: 6rem; border: 2px solid black;"
name="reponse"></textarea>
                <br>
                <input type="submit" name="submit" class="rosebutton">
            </form>
            <p><?php echo $message; ?></p>
        </div>
    </div>
</div>
</div>
</section>
</body>

</html>

```

19. redirection.php

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>LiveQuestion</title>
        <link rel="icon" href="img/favicon.png" type="image/png">
    </head>
    <body>
        <?php
            session_start();

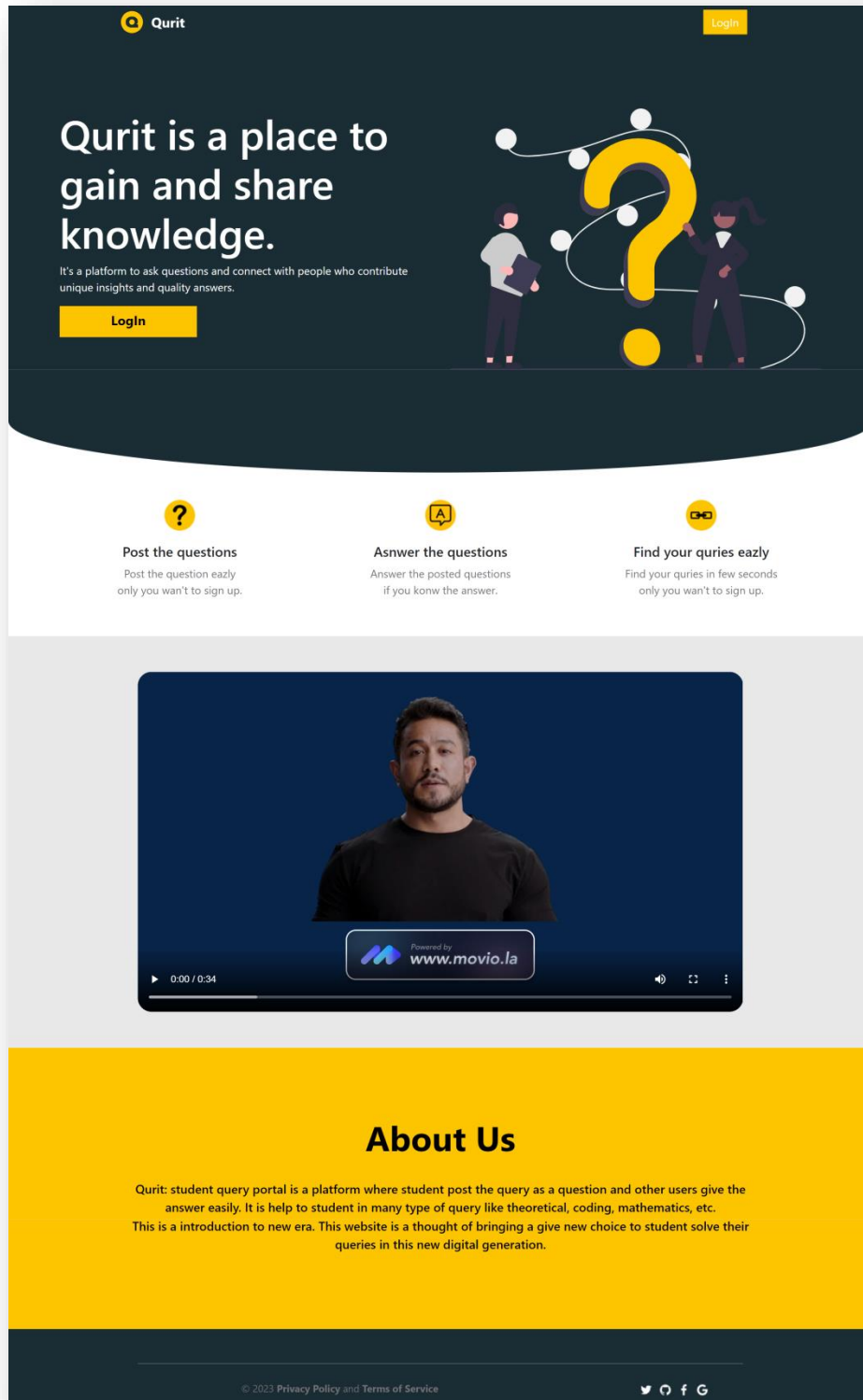
            if (!isset($_SESSION["pseudo"])) {
                header("Location: connexion.php");
            } else {
                header("Location: accueil.php");
            }
        ?>

    </body>
</html>

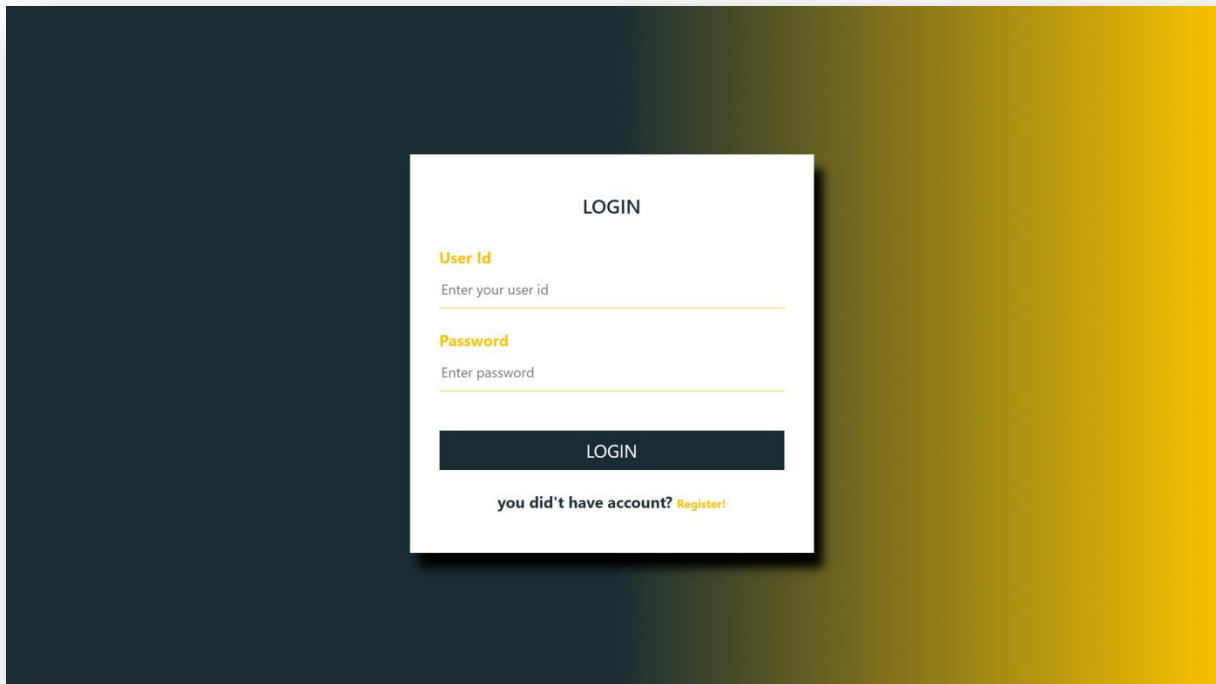
```

14. OUTPUT FORMS

1. Landing Page



2. Login Page



A login page mockup with a dark blue and yellow gradient background. A white login form is centered, featuring a 'LOGIN' title, 'User Id' and 'Password' labels, input fields, a 'LOGIN' button, and a 'Register!' link.

LOGIN

User Id

Enter your user id

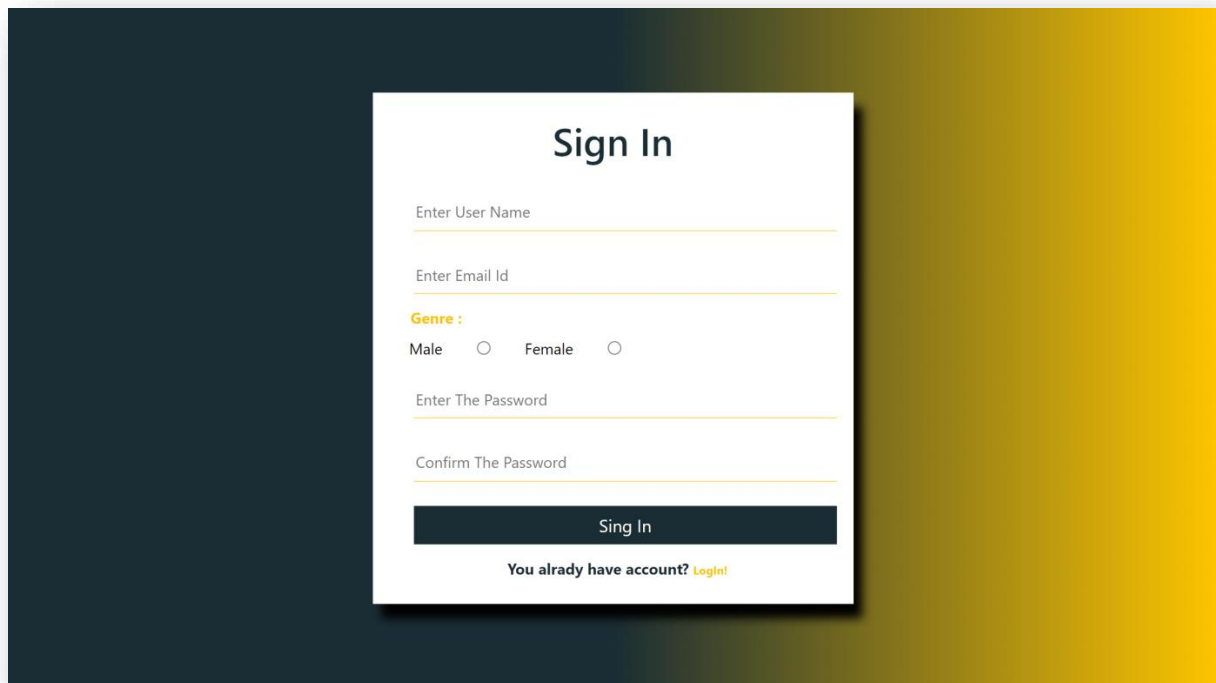
Password

Enter password

LOGIN

you didn't have account? [Register!](#)

3. Registration Page



A registration page mockup with a dark blue and yellow gradient background. A white registration form is centered, featuring a 'Sign In' title, 'Enter User Name', 'Enter Email Id', 'Genre' (Male/Female), 'Enter The Password', 'Confirm The Password', a 'Sing In' button, and a 'Login!' link.

Sign In

Enter User Name

Enter Email Id

Genre :

Male ☐ Female ☐

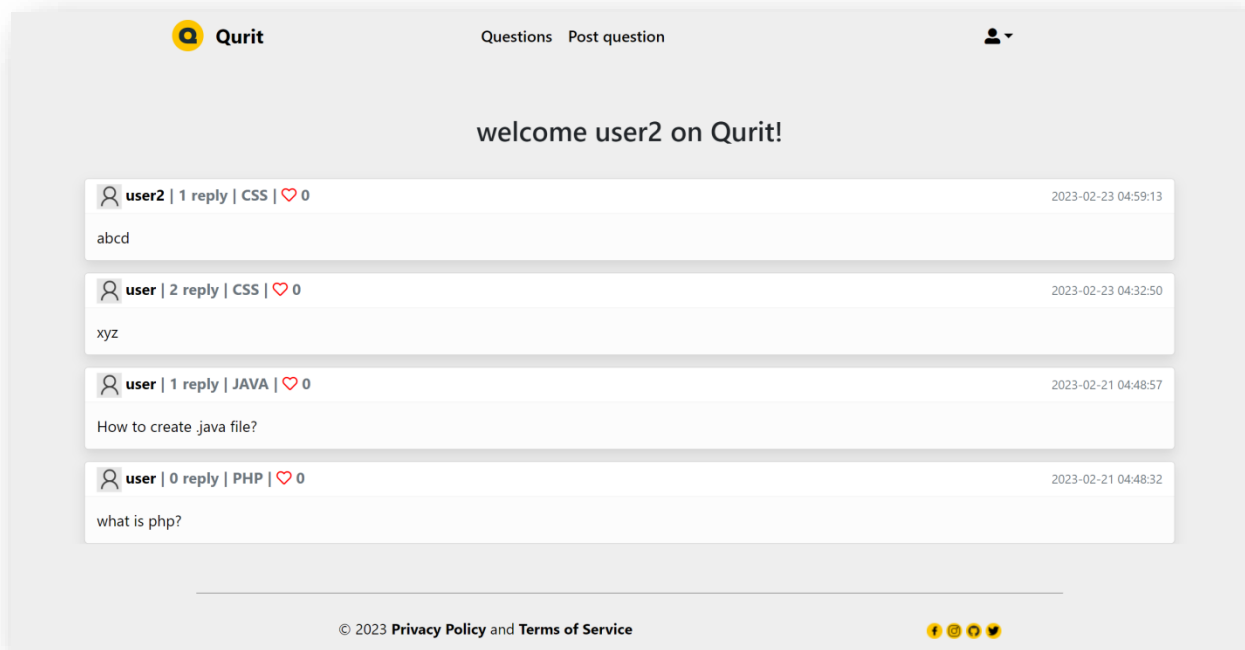
Enter The Password

Confirm The Password

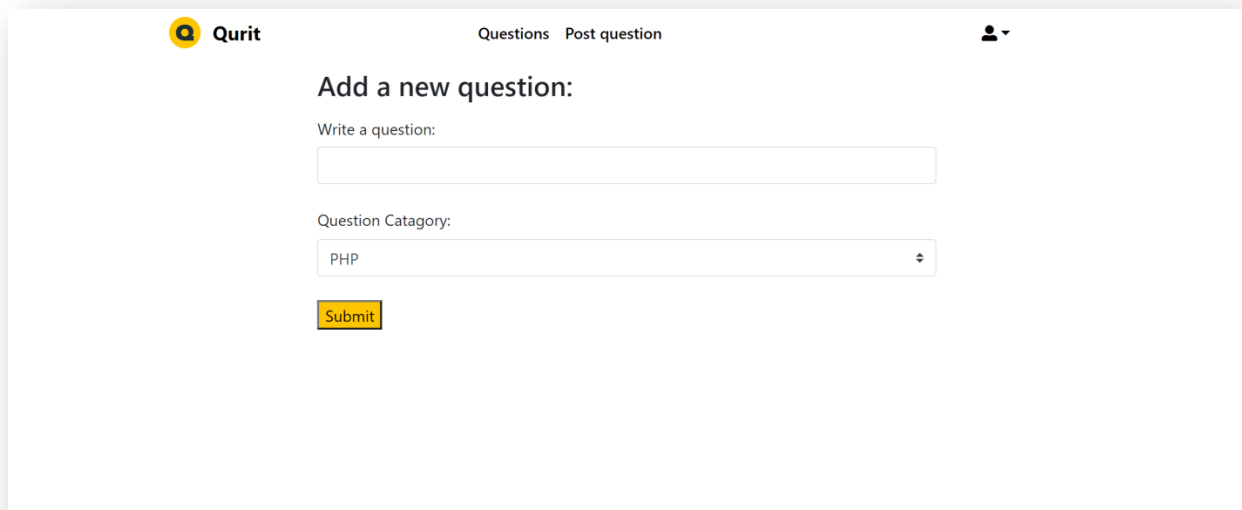
Sing In

You already have account? [Login!](#)


4. Question View Page





5. Question Add Page



6. Replay To Question Page

 Qurit


Questions Post question Delete Question

 user2 | 1 reply | CSS |  0

2023-02-23 04:59:13

abcd

Answer :

 user

2023-02-23 04:19:06


xyz

Respond to the question :



Your answer :


Submit

7. User Profile Page

 Qurit

Questions Post question


 




user2



Gender : Male ♂

Registration date: 2023-02-23 04:35:29

Edit Profile 

Delete Account 





My questions asked:

 user2 | 1 reply | CSS |  0

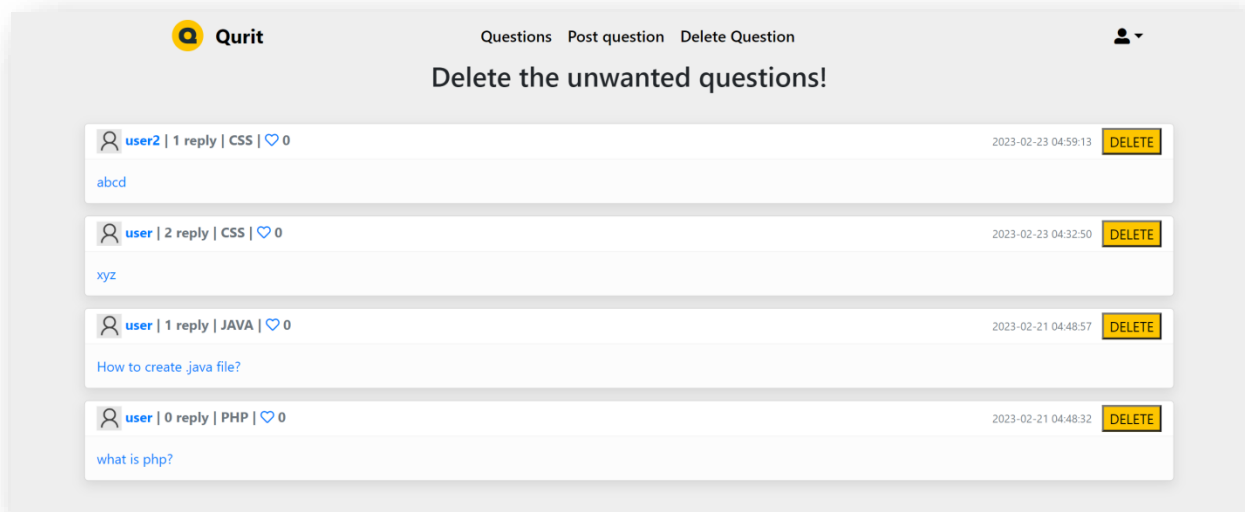
2023-02-23 04:59:13

abcd

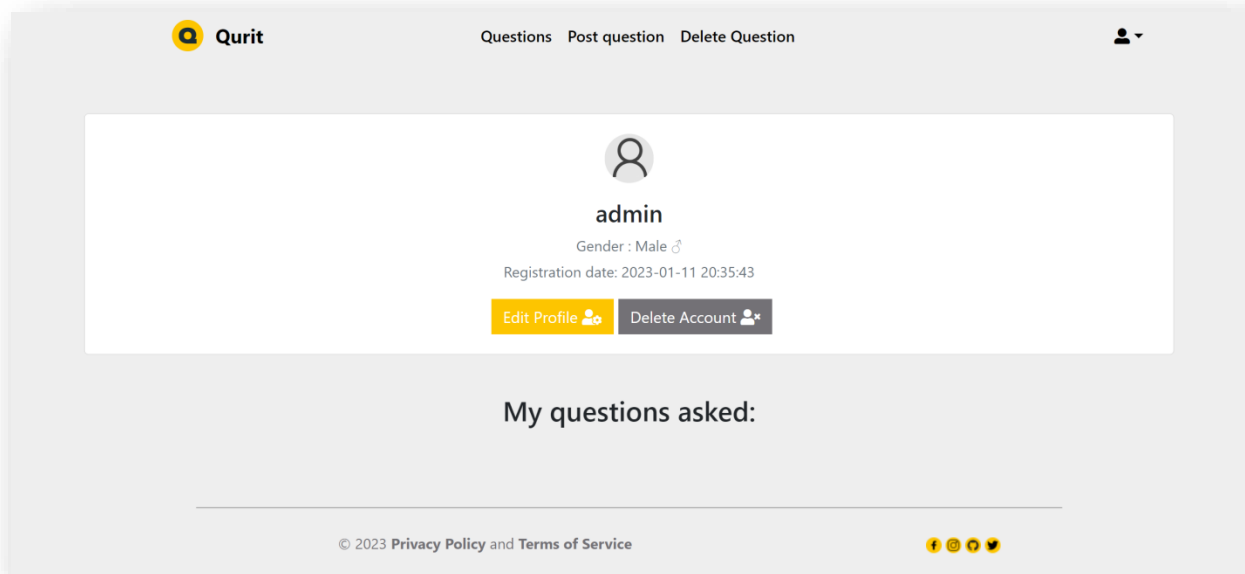
© 2023 Privacy Policy and Terms of Service

8. Admin Question Delete Page



9. Admin Profile Page



15. CONCLUSION

Easy and user-friendly website containing services related to student queries. Here we have a platform accessible everywhere. A specific and all requirements meted fluent website where we can search for questions and make it available for student/user. The query is being developed to ease the traditional question finding method by making it work through online mode. This project will help in avoiding the time-consuming manual process and will convert it in an automated process.

16. BIBLIOGRAPHY

Website Reference:

<https://www.freepik.com/>

<https://www.w3schools.com/html/>

Book Reference:

- Beginning Php6, Apache, MySQL Web Development by Boronczyk Timothy
- Learning PHP, MySQL & JavaScript with j Query, CSS & HTML5 by Robin Nixon

