# EXAMWIZ

**A project report submitted in partial fulfillment of the requirements for the award of the degree of**

# BACHELOR OF COMPUTER SCIENCE

**By**

# DRUPAD M D DVAVSCS027

**GOUTHAM SUNDAR S S DVAVSCS029**

# M P NAVANATH DVAVSCS014

**UndertheGuidanceof**

# Mrs. Deepa Ramachandran



# DEPARTMENT OF COMPUTER SCIENCE

**ST.JOSEPH’S COLLEGE(AUTONOMOUS),DEVAGIRI,CALICUT MARCH 2024**



**ST. JOSEPH’S COLLEGE (AUTONOMOUS)**

**DEVAGIRI, CALICUT**

**EXAMWIZ**

Report of the project submitted to St. Joseph’s College (Autonomous) Devagiri, Calicut

Affiliated to the University of Calicut

**DRUPAD M D (REG NO. DVAVSCS027)**

**GOUTHAM SUNDAR S S (REG NO. DVAVSCS029)**

**M P NAVANATH (REG NO. DVAVSCS014)**

Programme: Bachelor of Computer Science

Semester: VI

DEPARTMENT OF COMPUTER SCIENCE

March 2024

Principal Head of the Department Supervisor

College Seal

# PROJECT REPORT ON EXAMWIZ



## CERTIFICATE

This Is To Certify That The Project Report Titled

# EXAMWIZ

Submitted By

# DRUPAD M D DVAVSCS027

**GOUTHAM SUNDAR S S DVAVSCS029**

# M P NAVANATH DVAVSCS014

St.Joseph’sCollege(Autonomous)Devagiri, Calicutinpartialfulfillmentoftherequirementsforthe award of degree Bachelor of Computer Science during the Year 2021-2024.

InternalExaminer ExternalExaminer

# DECLARATION

We hereby declare that the project entitled “**EXAMWIZ**” has been undertaken by us for the award of the degree of Bachelor of Computer Science. We have completed this project under the guidance of Mrs. Deepa Ramachandran Department of Computer Science,St. Joseph’sCollege(Autonomous),Devagiri, Calicut.

Place: CalicutDate:

# DRUPAD M D DVAVSCS027

**GOUTHAM SUNDAR S S DVAVSCS029**

# M P NAVANATH DVAVSCS014



**CERTIFICATE**

This is to certify that the project report titled **EXAMWIZ**a project work done by them during the academic year 2023-2024 under my guidance and supervision in partial fillmentofthe requirement of Bachelor of ComputerScience.

Place:Devagiri Mrs.Deepa Ramachandran

Date: (Signature)

# ACKNOWLEDGEMENT

We wish to express our sincere gratitude to **Dr. Boby Jose** , Principal of St. Joseph’sCollege (Autonomous), Devagiri, Calicut for providing necessary facilities for developing your project.

We are also grateful to Ms.Asha Unnikrishnan , Head of the Department in charge,as well as Mrs.Deepa Ramachandran,our internal guide for the irimmense support and valuable suggestions.

We are also grateful to faculty members of,RissTechnologies,Calicut For providing assistance to do this work in their organization.

# DRUPAD M D DVAVSCS027

**GOUTHAM SUNDAR S S DVAVSCS029**

# M P NAVANATH DVAVSCS014

# SYNOPSIS

This is a simple yet effective academic tool for students of all phases. The basic operation of this is to accept Syllabi, a question paper (for identifying pattern) and Study materials(optional) to generate multiple Model Question papers with answer key for practice and evaluation for students and others for the proficiency in their area of study.

The features of this software include (but are not confined to:Generating Question papers,Including reading and oral questions,Finding enough study materials based on the syllabus provided,Generating Answer keys for self-evaluation,Mock test facility,Multiple exam question generator.

The software may include the following increments: ,Predicting the academic level of the student,Voice detection for Speaking input.

# TABLEOFCONTENTS

**CERTIFICATE I Ⅱ**

[**DECLARATION III**](#_3znysh7)

**CERTIFICATE II. IV**

**CERTIFICATE III V**

[**ACKNOWLEDGEMENT V**](#_2et92p0)**I**

[**SYNOPSIS VI**](#_1ljsd9k)**I**

**CHAPTERI**

1. **INTRODUCTION** 
   1. AN OVERVIEW 1
   2. INTRODUCTION MISSION OF THE PROJECT 1
   3. [BACKGROUND STUDY](#_44sinio)
      1. THE ORGANIZATION PROFILE 1
      2. STUDY ON EXISTING SYSTEM 2
      3. DISADVANTAGES OF EXISTING SYSTEM 2

**CHAPTER II** 3

1. **SYSTEM ANALYSIS**  4
   1. [STUDY PROPOSED SYSTEM 4](#_3as4poj)

[2.1.1.ADVANTAGES OF PROPOSED SYSTEM 5](#_49x2ik5)

* 1. USER REQUIREMENT SPECIFICATION 5
  2. SOFTWARE REQUIREMENT SPECIFICATION 7
  3. FEASIBILITY STUDY 9
  4. [SYSTEM SPECIFICATION 10](#_4f1mdlm)
     1. REQUIREMENTS FOR DEVELOPING APP 11
     2. REQUIREMENTS FOR USING APP 11

2.6 [COST ESTIMATION AND SCHEDULING 11](#_3tbugp1)

[**CHAPTER III**](#_37m2jsg)

1. **DESIGN AND DEVELOPMENT PROCESS** 12
   1. [FUNDAMENTAL DESIGN CONCEPTS 13](#_46r0co2)
   2. [DESIGN NOTATIONS 14](#_3l18frh)
      1. [DATA FLOW DIAGRAM 14](#_206ipza)
   3. [DESIGN PROCESS 21](#_45jfvxd)
      1. [DATABASE DESIGN 21](#_1egqt2p)
      2. [NORMALIZATION 26](#_2dlolyb)
      3. [INPUT DESIGN 27](#_3cqmetx)
      4. [OUTPUT DESIGN 28](#_1rvwp1q)

**CHAPTER 1V**

1. **CODING** 29

4.1 FRONT END 30

4.2 BACK END 31



**CHAPTERV**

1. **TESTING AND IMPLEMENTATION** 33

* 1. [TESTING 34](#_3q5sasy)
     1. TESTING METHODOLOGIES 34
     2. [DIFFERENT TESTING 35](#_34g0dwd)

5.1.3 [TEST CHART 39](#_4h042r0)

* 1. [QUALITY ASSURANCE POLICIES 42](#_1baon6m)
     1. [GENERIC RISKS 42](#_pkwqa1)
  2. [SYSTEMIMPLEMENTATION 43](#_39kk8xu)
     1. IMPLEMENTATION PROCEDURES 45
  3. [SYSTEM MAINTENANCE 45](#_2nusc19)

**CHAPTER VI**

1. **CONCLUSION** 47

6.1 CONCLUSION48

6.2 [SCOPE FOR FURTHER ENHANCEMENTS 48](#_2fk6b3p)

6.3 BIBLIOGRAPHY 48

[**ANNEXURE-A** 49](#_upglbi)

1. [INPUT AND OUTPUT DESIGN 50](#_4du1wux)
2. [SAMPLESOURCE 55](#_184mhaj)

[**ANNEXURE-B** 58](#_3s49zyc)

[ABBREVIATION 59](#_36ei31r)

# LISTOFTABLES

**SNo Title Page**



| **3.3.1.1** | **LoginTable** | **21** |
| --- | --- | --- |
| **3.3.1.2** | **Advocate Table** | **21** |
| **3.3.1.3** | **User Table** | **22** |
| **3.3.1.4** | **Booking Table** | **22** |
| **3.3.1.5** | **Case Table** | **23** |
| **3.3.1.6** | **Case Allocate Table** | **23** |
| **3.3.1.7** | **Case Report Table** | **23** |
| **3.3.1.8** | **Chat Table** | **24** |
| **3.3.1.9** | **Complaint Table** | **24** |
| **3.3.1.10** | **Daily Report Table** | **24** |
| **3.3.1.11** | **Feedback Table** | **25** |
| **3.3.1.12** | **Law Table** | **25** |
| **3.3.1.13** | **Next Hearing Table** | **25** |

# CHAPTER-

# INTRODUCTION

## EXAMWIZ : AN OVERVIEW

In response to the increasing demand for efficient examination processes in our digital age, ExamWiz emerges as a comprehensive web application. From preparation to exam management, it caters to the needs of students, educators, and institutions alike. For students, ExamWiz accepts various study materials, generating multiple model question papers for diverse practice. Educators benefit from streamlined assessment creation, with intuitive tools ensuring relevance and adherence to standards. The platform introduces grading automation, saving time and ensuring accuracy. Exam management is centralized, providing a user-friendly interface for scheduling, monitoring, and conducting exams. Results generation is quick and insightful, offering valuable data for performance analysis. With its user-friendly design, ExamWiz redefines academic proficiency in the modern, digital landscape.

## INTRODUCTION MISSION OF PROJECT

Welcome to ExamWizz, a pioneering project revolutionizing education with an innovative examination platform.

ExamWizz aims to simplify and optimize the examination process, leveraging technology to empower students, support educators, and enhance institutional efficiency. Our commitment is to redefine academic proficiency through user-friendly interfaces, diverse study material acceptance, and automated grading, fostering a dynamic learning environment for success.

## BACKGROUND STUDY

## THE ORGANIZATION PROFILE

REGIONAL TECHNOLOGIES is a rapidly growing company that provides professional IT services. We are one of the largest and Best software development companies in Kerala with focus on .Net, PHP, Java, Software testing, SEO and Web Design. As a leader in providing Offshore Software Development and related services, REGIONAL TECHNOLOGIES functions from an offshore set-up, based in Kerala, India. Our reputation enables us to serve in terms of Outsourced Software Development, Web Development, designing of websites and their corresponding development. Particularly high end developments using Microsoft .NET, Java J2EE platforms, ASP, ASP.NET, PHP development as well as VB.6.0 Development adjoins us to a particularly niche sector of the global servicing. We have worked for and have provided services to the 6 of the top 10 Fortune 500 companies of the world. We aim to provide our clients with comprehensive, end-to-end technology solutions that give them an advantage over the competition. From building applications that increase your productivity, to internet-enabling your business for maximum profits.

## MISSION

Our mission is to continuously optimize our customers' business through our world-class solutions, services and products. We ensure the success of our company by constantly and consistently satisfying our customers, shareholders and employees.With Continuous research and development we aim to make available latest technology and concepts and tailored it to our valuable client’s requirement and enable them to march on the path of success towards the leader discussions on the phone as needed.

## STUDY EXISTING SYSTEM

existing system:

The existing system is a manual system. There is no system to automate the processes of an advocate agency. In the existing system, from the client registration to the allocation of the advocates is done manually.

## DISADVANTAGES OF EXISTING SYSTEM

* + - * It is a time consuming process
      * possibility of errors are high.
      * Scheduling advocates is complicated.
      * Searching for a particular file is very difficult.

# CHAPTER-II SYSTEM ANALYSIS

# 2. SYSTEM ANALYSIS

A system study is a process of studying a procedure to identify the objectives and purposes of a system as well as to analyze the existing system’s problems and drawbacks. The system analysis will create systems and procedures that will achieve these objectives as well as solutions to the problems in an efficient way. It is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements to the system.Training,experienceandcommonsensearerequiredforthecollectionoftheinformationneeded to do theanalysis.

## STUDY PROPOSED SYSTEM

The proposed system solves the disadvantages of the existing system. A study of the existing system and identification of its drawbacks lead to the designing of the proposed system. The new system is designed to be very user friendly, easy to use and also to provide fast retrieval of data.

In the proposed system, there are three main modules. They are the Administrator module , Advocate module, and Client module. Security is provided through hierarchical division of authorities. The entire system is monitored by the administrator. Only the administrator has the power to register new advocates to the agency and to provide authorization rights for them. They can view the case list, client list, different reports like payment report and scheduling of advocates. There is also an option for viewing feedback from advocates in the concern and the opinions, suggestions and complaints from the clients. In the Entry-level module client registration, case registration, uploading of case files, allocation of cases to appropriate advocates, payment entry, rack registration with specific shell number or using the case number, case file allocation, case scheduling entry and the payment report are done. Clients are registered using a one-time registration process.

The Advocate module deals with the features like viewing the allocated case, providing daily reports, giving feedback to the admin, viewing the case scheduling, searching files in the rack with specific shell number, preparation and uploading of case notes, and also an option like forum which helps them to get suggestions from the client. In the Client module, a client, who will be allocated with a login id and password at the time of registration, can view his/her case status. She/he can give the suggestions to the concerned advocate with the help of a forum. A chatting section is also provided. Also there is an option that the client can provide the feedback on the services. There is a Public Section which provides the information about the advocates in the agency, case list handled by the agency and also the contact of the agency is provided. This section helps the advocate to attract the clients more than that they do with their words.

## ADVANTAGES OF PROPOSED SYSTEM

* + - * User friendly
      * Provide fast retrieval of data.
      * Files or datas are more organized.
      * Advocates can be scheduled easily.
      * Time saving.
      * Possibility of errors are comparatively low.

## USER REQUIREMENT SPECIFICATION

Afterthoroughanalysis,Oursystemhasbeenpresentedwiththefollowingmodules:

1. Admin
2. Advocate
3. User

## ADMIN

Inthismoduletheadminhastheprivilegetocontrolthesystem.Administhemainashecanviewusersand advocates.

* + Login:Admincanlogintothewebsiteusingtheirusernameandpassword.
  + Add and manage Advocates: admin can add and manage advocates.
  + View Users: Admin can view users
  + View booked cases : admins can view the advocates booked by the users.
  + case allocate:admin allocates case to the advocates that the user has booked.
  + View Feedback: admin can view feedback from users about the application.
  + View Complaints and reply: admin can view complaints and reply to users.
  + View Daily Report: admin can view daily report



## ADVOCATE

* + Login: Advocate canlogintothewebsiteusingtheirusernameandpassword provided by the admin.
  + View allocated case: advocates can view allocated cases.
  + Chat : advocates can chat with users.
  + Add & manage Case report: can update case report
  + Add & Manage next hearing date: advocates can add and manage next hearing date
  + Add & Manage Daily report: advocates can add and manage daily reports.
  + Add and manage law: advocates can add and manage law

## USER

* + Registration: Users must register to the application.
  + Login :Theusermustsignupbeforeloggingintotheapplication.
  + Case register: users can register case
  + Chat: users can chat with advocates
  + Feedback: can give feedback about the application.
  + Complaints & reply:
  + View case report: user can view the report of the case
  + View next hearing: user can view next hearing date
  + View advocate: user can view advocates
  + View law: user can view law
  + Book advocates : can book advocates.



## SOFTWARE REQUIREMENTS SPECIFICATION

Oncethesystemrequirements arefoundthenwehavetodeterminewhetheraparticularsoftwarepackagefitsfor thosesystemrequirements.

## PyCharmIDE2017.1.2

PyCharm is a dedicated Python Integrated Development Environment (IDE) providing a wide range of essential tools for Python developers, tightly integrated to create a convenient environment for productive Python, Web, and data science development. It is developed by the Czech company Jet Brains.It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web developmentwithDjangoaswellasdatasciencewithAnaconda.PyCharmiscross-platform,withWindows, macOS and Linux versions. PyCharm provides API so that developers can write their own plugins to extend PyCharm features. Several plugins from other JetBrains IDEa also work with PyCharm. There are more than 1000 plugins which are compatible with PyCharm.JetBrains asdevelopedPyCharmasacross-platformIDEforPython.In Addition to supporting versions 2.x and 3.x of Python, PyCharm is also compatible withWindows,Linux,andmacOS.Atthesametime,thetoolsandfeaturesprovide by PyCharm help programmers to write a variety of software applications in Python quickly and efficiently. The developers can even customize the PyCharm UI according to their specific needs and preferences. Also, they can extend the IDE by choosing from over 50 plug-ins to meet complex project requirements. PyCharm makes it easier for developers to implement both local and global changes quickly and efficiently. The developers can even take advantage of the refactoring options provided by the IDE while writing plain Python Code and working with Python frameworks. They can avail the rename and move refactoring forfiles,classes,functions,methods,properties,parameters,andlocal/global variables.Likewise,theycan improve code quality by extracting variables,fields,constants,and parameters. Also, PyCharm allows programmers to break up longer classes and methods through extract methods. PyCharm makes it easier for programmers to write various webapplicationsinPythonsupportingwidelyusedwebtechnologieslikeHTML,CSS,JavaScript,TypescriptandCoffeeScript.Thewebdeveloperscanusetheliveeditingpreview option provided by the IDE to view a single web page simultaneously in the editorandbrowser.Atthesametime,theliveeditfeatureprovidedbytheIDEenablesprogrammerstoseethechangesmadetothecodeinstantaneouslyonawebbrowser.

PyCharm further allows developers to avail a JavaScript debugger as well as CoffeeScriptand Typescript editors. It even simplifies isomorphic web application development by supporting both Angularjs JSand NodeJS.

## FLASK

Flask is a web application framework written in Python. It is developed by Armin Ronacher,who leads an international group of Python enthusiasts named Pocco. Flask is based on theWerkzeugWSGItoolkitandJinja2templateengine.Flaskisamicrowebframeworkwritten in Python. It is classified as a micro framework because it does not require particulartoolsorlibraries.Ithasnodatabaseabstractionlayer,formvalidation,oranyothercomponents where pre-existing third-party libraries provide common functions. However,Flask supports extensions that can add application features as if they were implemented inFlaskitself.Extensionsexistforobject-relationalmappers,formvalidation,uploadhandling, various open authentication technologies and several common framework relatedtools.FlaskisconsideredmorePythonicthantheDjangowebframeworkbecauseincommonsituationstheequivalentFlaskwebapplicationismoreexplicit.Flaskisalsoeasyto get started with as a beginner because there is little boilerplate code for getting a simpleapp up and running. Flask was originally designed and developed by Armin Ronacher as anApril Fool's Day joke in 2010. Despite the origin as a joke, the Flask framework becamewildly popular asan alternative toDjango projects with theirmonolithic structure anddependencies.

## ANDROID STUDIO

Android is a [mobile operating system](https://en.wikipedia.org/wiki/Mobile_operating_system) based on a modified version of the [Linux kernel](https://en.wikipedia.org/wiki/Linux_kernel) and

other [open-source](https://en.wikipedia.org/wiki/Open-source_software) software, designed primarily for [touchscreen](https://en.wikipedia.org/wiki/Touchscreen) mobile devices such

as [smartphones](https://en.wikipedia.org/wiki/Smartphone) and [tablets](https://en.wikipedia.org/wiki/Tablet_computer). Android is developed by a consortium of developers known as

the [Open Handset Alliance](https://en.wikipedia.org/wiki/Open_Handset_Alliance), though its most widely used version is primarily developed

by [Google](https://en.wikipedia.org/wiki/Google). It was unveiled in November 2007, with the first commercial Android device,

the [HTC Dream](https://en.wikipedia.org/wiki/HTC_Dream), being launched in September 2008.

At its core, the operating system is known as Android Open Source Project (AOSP)[[4]](https://en.wikipedia.org/wiki/Android_(operating_system)#cite_note-4) and

is [free and open-source software](https://en.wikipedia.org/wiki/Free_and_open-source_software) (FOSS) primarily licensed under the [Apache License](https://en.wikipedia.org/wiki/Apache_License).

However most devices run on the [proprietary](https://en.wikipedia.org/wiki/Proprietary_software) 

## FEASIBILITY STUDY

A feasibility study is an assessment of the practicality of a proposed system or project. Afeasibility study aims to objectively and rationally uncover weakness and strengths of anexistingbusinessorproposedventure,opportunitiesandthreatspresentinthenaturalenvironment, the resources required to carry through, and ultimately the prospects of success.Initssimplestterms,thetwocriteriatojudgefeasibilityarecostrequiredandvaluetobe

attained. A feasibility study evaluates the project’s potential for success; therefore, perceived objectivity is an important factor in the credibility of the study for potential investors and lending institutions. It must therefore be conducted with an objective, unbiased approach to provide information upon which decisions can be based. The feasibility study is done in thesephases.

* TechnicalFeasibility
* EconomicFeasibility
* OperationalFeasibility

## TECHNICAL FEASIBILITY

It investigates the technical feasibility of each implementation alternative. It analyzes and determines whether the solution can be supported by existing technology or not. The analyst determines whether current technical resources be upgraded or added that fulfill the new requirements. The analyst must find out whether current technical resources can be upgraded or added in a manner that fulfills the request under considerations. Our software is more user friendly. We are using python language for coding. So, it is easy to understand and it is more readable.

## ECONOMICAL FEASIBILITY

The purpose of an economic feasibility study is to demonstrate the net benefit of a proposedprojectforacceptingordisbursingelectronicfunds/benefitsandcoststotheagency,otherstate agencies and the general public as a whole. It is evaluating the effectiveness of candidatesystembyusingcost/benefitanalysismethod.Itdemonstratesthenetbenefitsfromthecandidate system in terms of benefits and costs to the organization. Software is said to be economically feasible if it focuses on issues like cost incurred on software development toproducelongtermgainsforanorganization,costrequiredtoconductfullsoftware

investigation, cost of hardware, software, development team, and training.Our application can be economically feasible since it can be used by all sections of the society.

## OPERATIONAL FEASIBILITY

Itdetermineswhetherthesystemisoperatingeffectivelyonceitisdevelopedandimplemented. It ensures that the management should support the proposed system and its working feasible in the current organizational environment. It analyzes whether the users will be affected and they accept the modified or new business methods that affect the possible system benefits. It also ensures that the computer resources and network architecture of candidate system are workable.

The proposed system is easily understandable.the maintenance and working of proposed system need less human effort. The application can be easily operated via smartphone thus this project passes these entire tests for feasibility and thus found feasible.

## SYSTEM SPECIFICATION

* + 1. **REQUIREMENTS DEVELOPING APP**

HARDWARE:

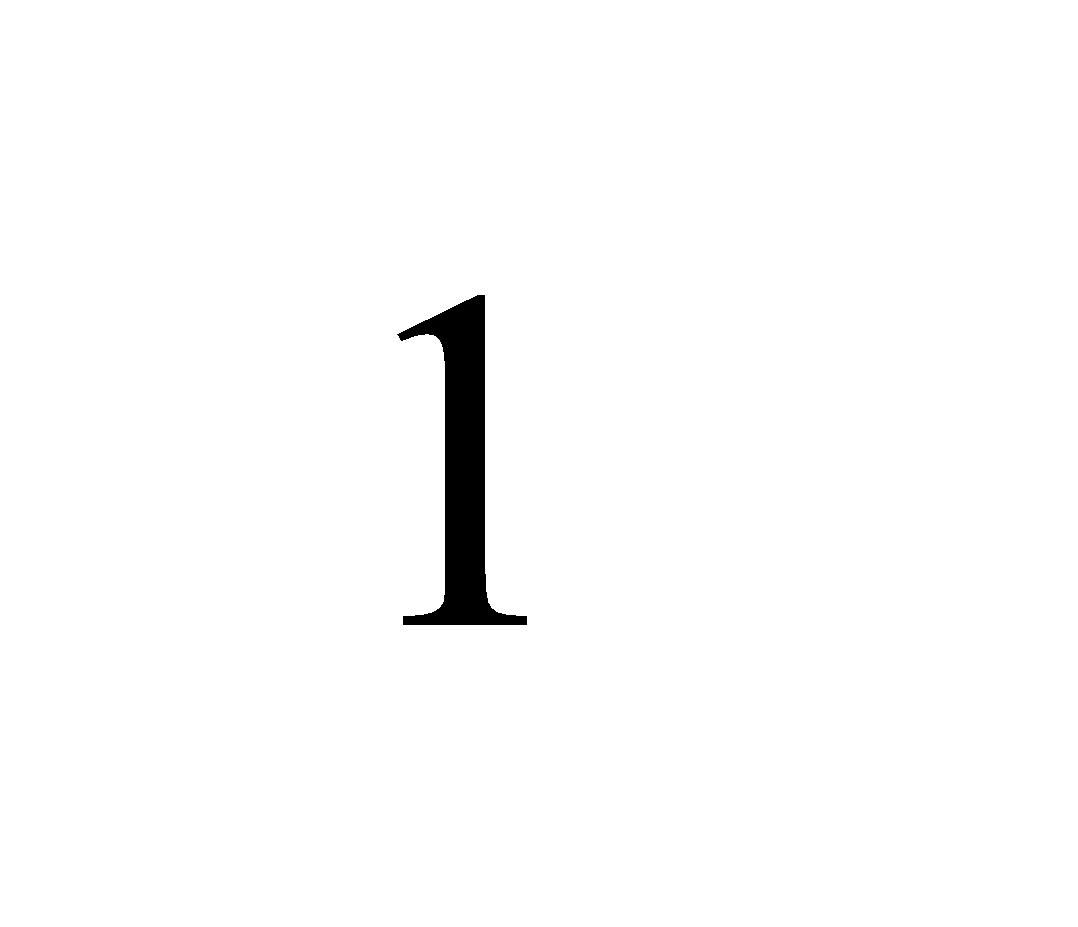
* PROCESSOR:IntelCOREi3(4TH GEN)and aboveorAMDDualCoreorequivalent.
* RAM:Minimum4GBandAbove
* RAM:Minimum 2GB,4GBRecommended
* DisplayResolution:1280x800px

SOFTWARE**:**

OS:MicrosoftWindows7andabove(64 Bit)Apps:

* + -AndroidStudio
  + -AndroidEmulator
  + -PyCharmIDE





## REQUIREMNTSFORUSINGAPP

HARDWARE:

* PROCESSOR:QualcommSnapdragon400SeriesandaboveorEquivalentExynosorMediaTek Processor
* RAM:Minimum2GBandAbove
* ROM:Minimum100MBStorage
* INTERNET:Required

SOFTWARE:

OperatingSystem:AndroidVersion5.0andabove

## COST ESTIMATION & SCHEDULING

* H/W:40000
* S/W:5000
* PERSONNEL:3000
* OTHER CHARGES:2000

## SCHEDULING

* Requirements Collection-2 weeks
* Analysis And Design-2 weeks
* Coding-4 weeks
* Testing-1 week



# CHAPTER-III

**DESIGN AND DEVELOPMENT PROCESS**

## FUNDAMENTAL DESIGN CONCEPTS

Over the course of software engineering history, a collection of essential design principles has emerged. These principles serve as a cornerstone for software designers, providing them with a solid foundation upon which they can build more advanced design methodologies. These concepts are integral to both traditional and object-oriented software development approaches.:

## Abstraction

Abstraction simply means to hide the details to reduce complexity and increases efficiency or quality. Different levels of abstraction are necessary and must be applied at each stage of the design process so that any error that is present can be removed to increase the efficiency of the software solution and to refine thesoftwaresolution.

## Architecture

Architecture in software design refers to the methodology of structuring something. In the context of software, it involves the arrangement of different elements and the handling of data within the structure. These elements interact with each other and utilize the structure's data as part of the architectural process..

## Separation Of Concerns

The separation of concerns is a design principle that proposes breaking down a complex problem into smaller, more manageable pieces. By doing so, each piece can be addressed or optimized independently. This approach reduces the effort and time required to solve the problem, making it more manageable overall.

## Modularity

Modularity involves breaking down a system or project into smaller components to simplify its complexity. Similarly, in design, modularity entails subdividing a system into smaller parts, allowing for independent creation. These parts can then be utilized across different systems to perform various functions..

## InformationHiding

Information hiding involves concealing data to prevent access by unauthorized parties. In software design, this is accomplished by structuring modules so that the information they contain is inaccessible to other modules..



## Refinement

Refinement is the process of improving something to eliminate impurities and enhance its quality. In software design, refinement involves developing or presenting the software or system in a detailed manner, elaborating on its structure and functionality. It's crucial for identifying and reducing errors..

## Object-OrientedDesignConcepts

The object-oriented (OO) paradigm is extensively utilized in contemporary software engineering. OO design incorporates various concepts like classes and objects, inheritance, messages, and polymorphism, among others, as its distinguishing features..

## DESIGN NOTATIONS

## DATA FLOW DIAGRAM

A graphical representation is employed to depict and analyze the flow of data within a system, whether manual or automated, encompassing processes, data storage, and system delays. Data flow diagrams serve as a central tool, forming the foundation for the development of other components. The transition of data from input to output via processes can be logically described independently of the system's physical elements. These logical data flow diagrams illustrate the actual implementation and movement of data among individuals, departments, and workstations. DFDs represent one of the most crucial modeling tools used in physical design, illustrating the data flow through various processes within the system.



# NotationsusedinDFD

InDFD,there are four symbols,they as follows:

* + - * Process



* + - * DataStore



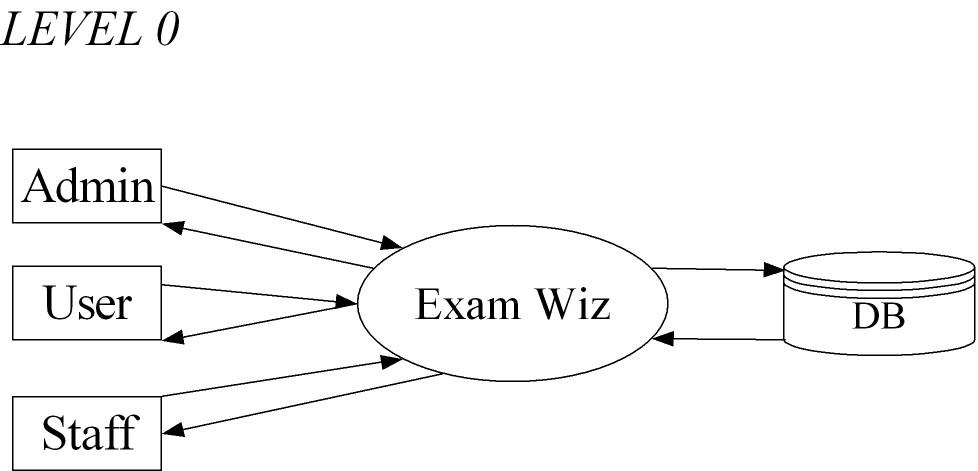
* + - * ExternalEntity



* + - * Direction Of Flow

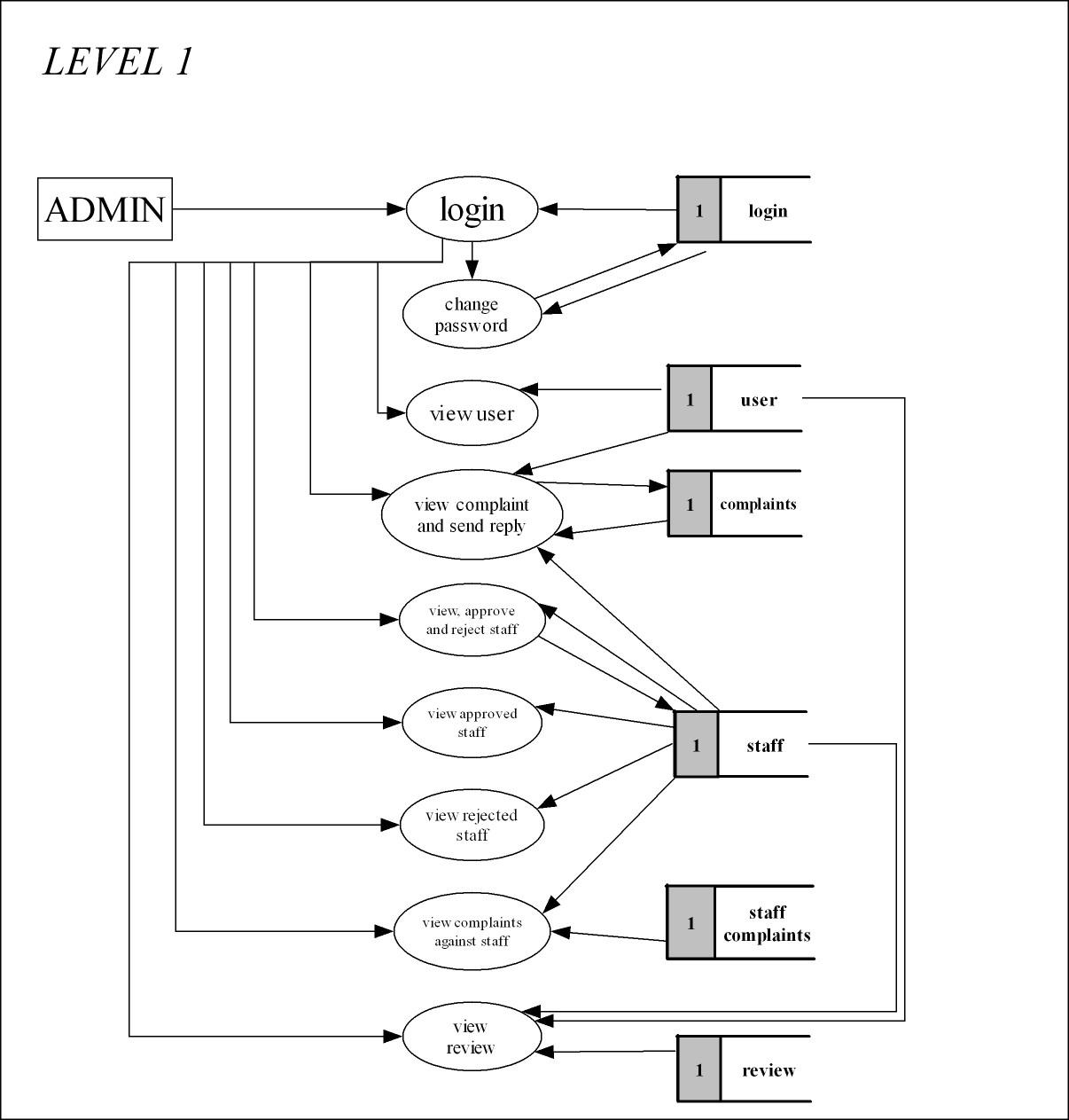


## LEVEL 0 DFD

****

**Figure 3.2.1**

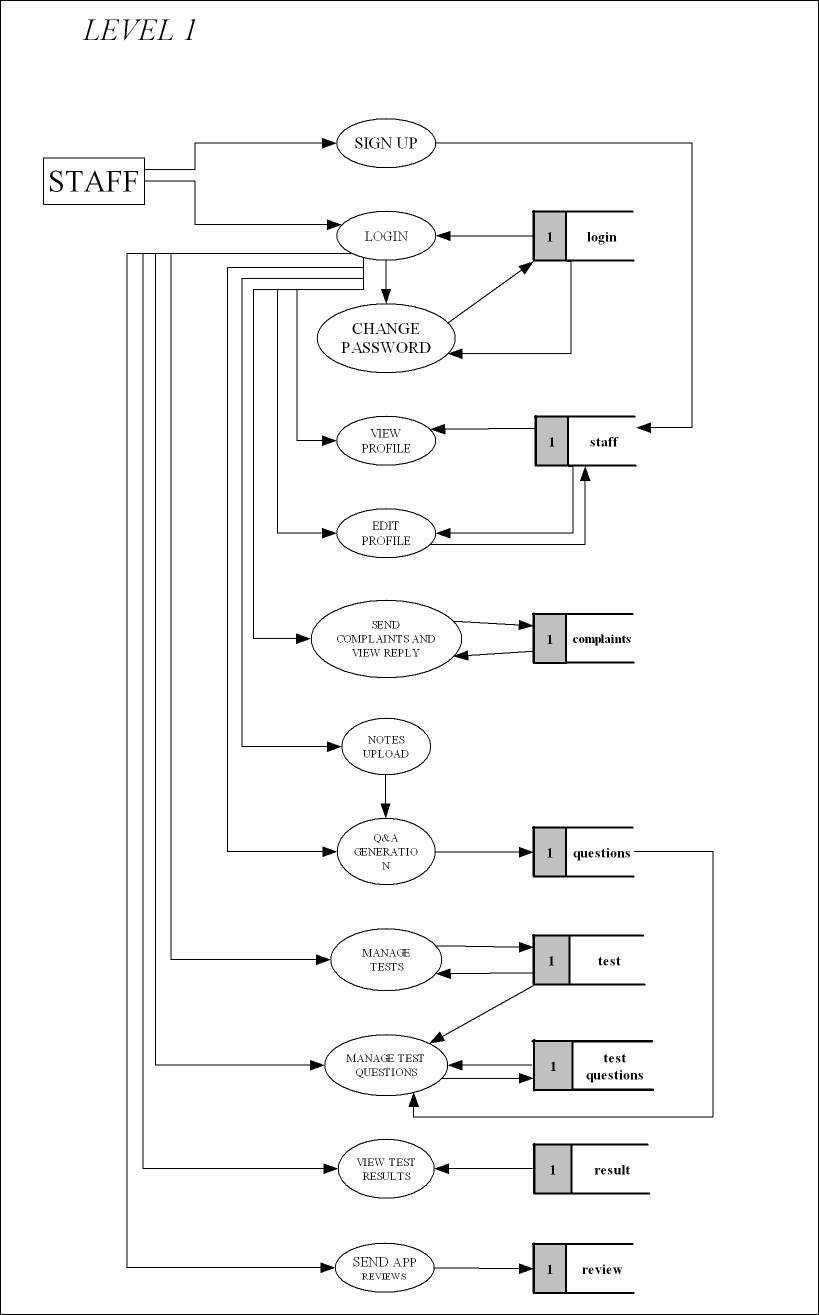
## LEVEL 1.1DFD(Admin)

****

**Figure 3.2.1.1**



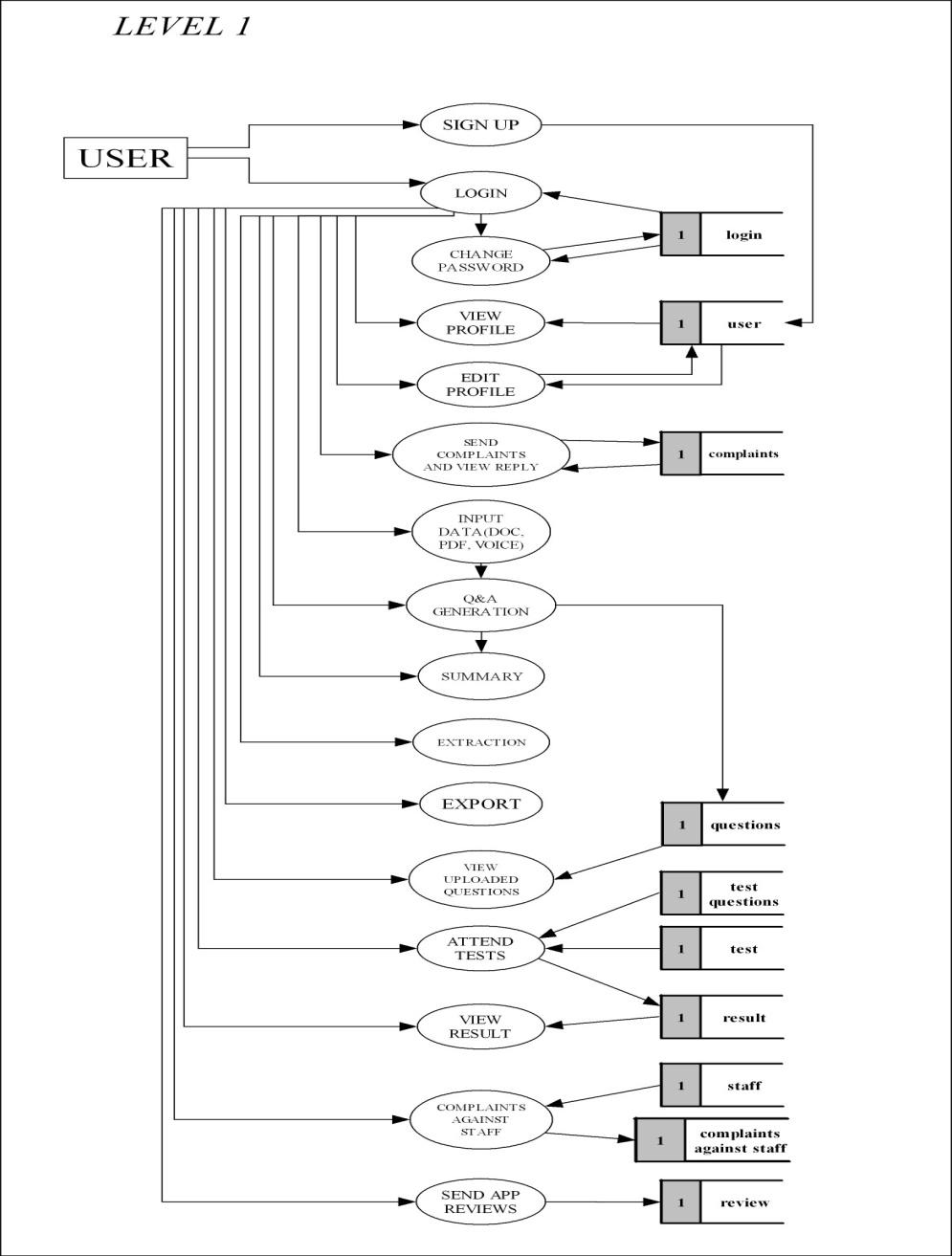
## LEVEL 1.2DFD(STAFF)

****

**Figure 3.2.1.2**



## LEVEL 1.3DFD(User)



**Figure 3.2.1.3**



## Figure 3.2.2.1 DIAGRAM

**Figure 3.2.2.1**

## DESIGN PROCESS

## DATABASE DESIGN

A table is a structured data format that arranges information into rows and columns, facilitating both storage and presentation of data. Databases typically consist of multiple tables, each tailored for a specific purpose and containing its own set of fields based on the required data. In database tables, fields are represented as columns, and individual entries, or records, as rows. Accessing specific values from a table involves requesting data from individual columns and rows. Primary key fields are utilized to uniquely identify records within a table.

## Table3.3.1.1:LOGIN TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | big | 20 | Primary Key |
| **username** | VARCHAR | 100 | Not Null |
| **password** | VARCHAR | 100 | Not Null |
| **Type** | VARCHAR | 30 | Not Null |

**Table 3.3.1.2: USER TABLE**

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **name** | VARCHAR | 100 | Not Null |
| **gender** | VARCHAR | 100 | Not Null |
| **dob** | DATE |  | Not Null |
| **photo** | VARCHAR | 300 | Not Null |
| **mail** | VARCHAR | 100 | Not Null |
| **phone** | BIGINT | 20 | Not Null |
| **place** | VARCHAR | 100 | Not Null |
| **current\_education\_status** | VARCHAR | 100 | Not Null |
| **LOGIN\_id** | BIGINT | 20 | Foreign Key |

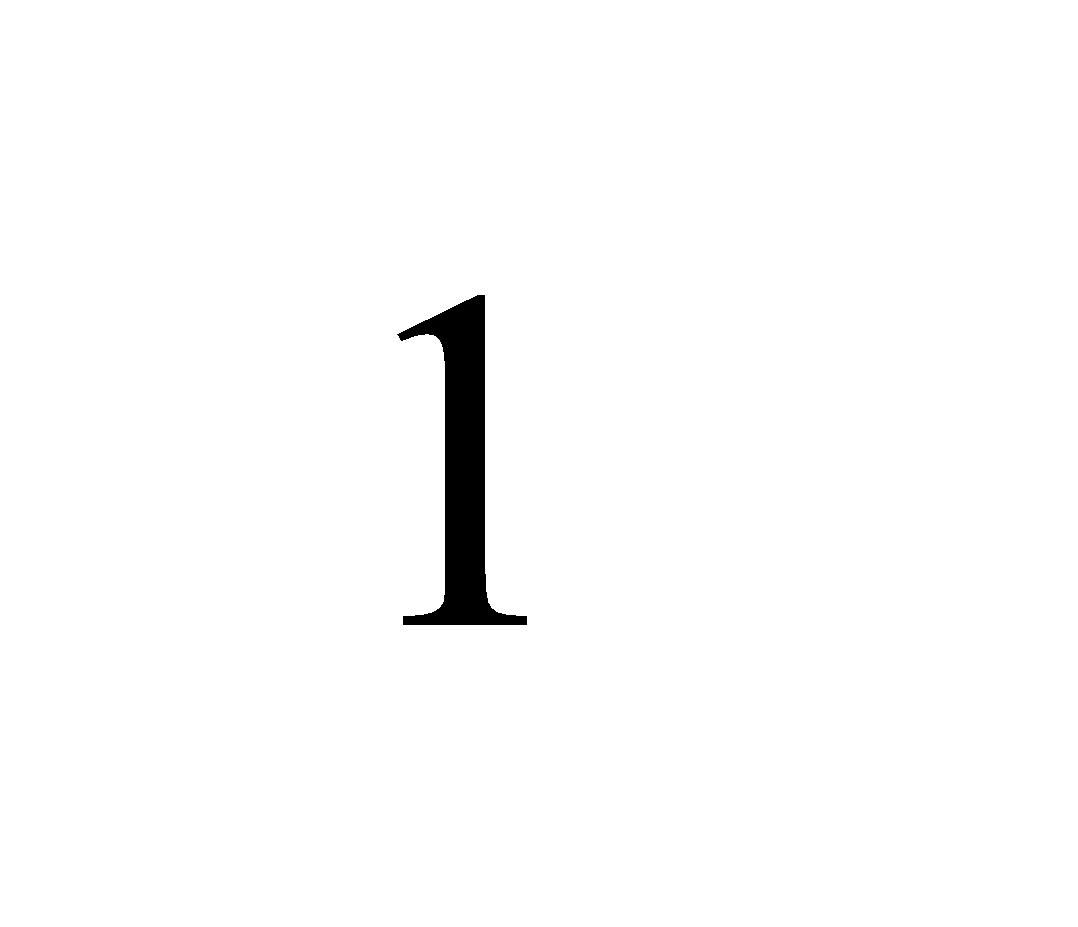


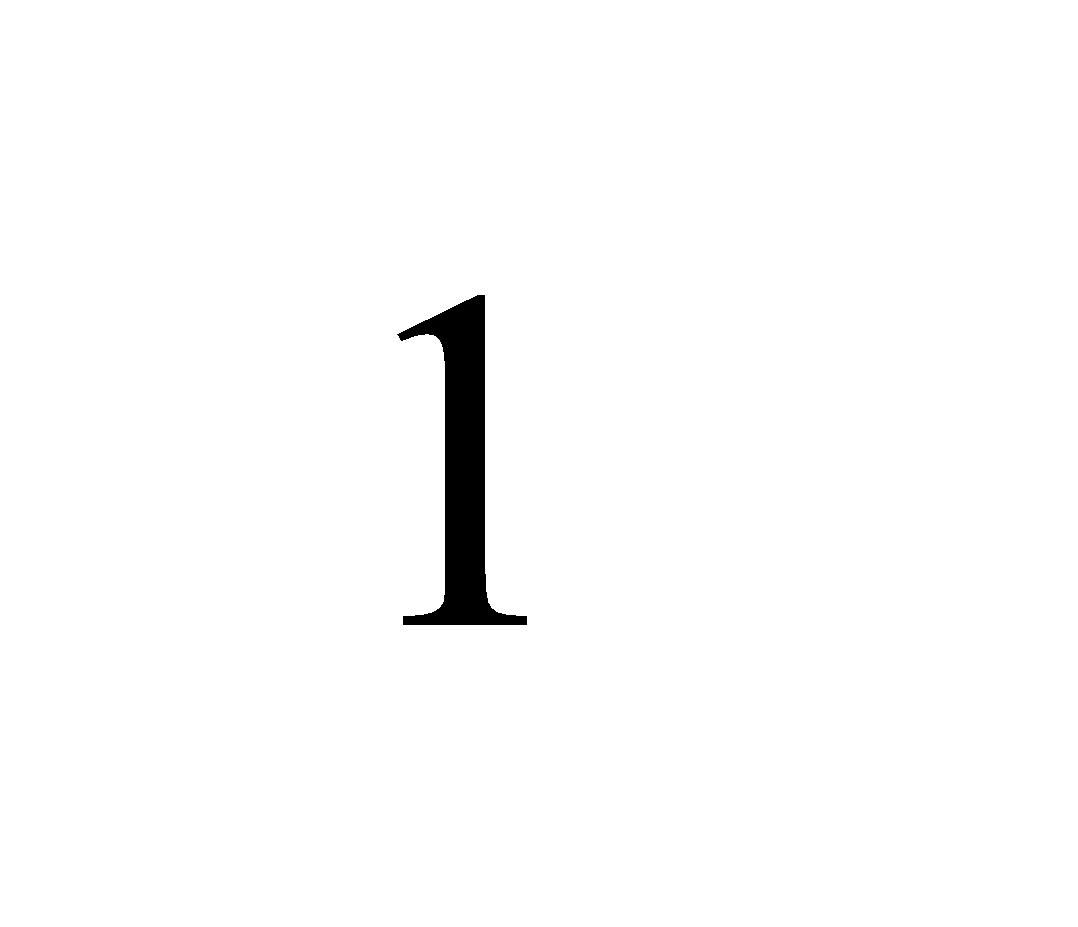
## Table 3.3.1.3: STAFF TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **name** | VARCHAR | 100 | Not Null |
| **gender** | VARCHAR | 100 | Not Null |
| **dob** | DATE |  | Not Null |
| **photo** | VARCHAR | 300 | Not Null |
| **mail** | VARCHAR | 100 | Not Null |
| **phone** | BIGINT | 20 | Not Null |
| **house name** | VARCHAR | 100 | Not Null |
| **place** | VARCHAR | 100 | Not Null |
| **district** | VARCHAR | 100 | Not Null |
| **pincode** | INT | 11 | Not Null |
| **state** | VARCHAR | 100 | Not Null |
| **qualification** | VARCHAR | 100 | Not Null |
| **proof** | VARCHAR | 300 | Not Null |
| **status** | VARCHAR | 100 | Not Null |
| **current\_institution** | VARCHAR | 100 | Not Null |
| **LOGIN\_id** | BIGINT | 20 | Foreign Key |

**Table 3.3.1.4: TEST TABLE**

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **STAFF\_id** | BIGINT | 20 | Foreign Key |
| **testname** | VARCHAR | 100 | Not Null |
| **date** | DATE |  | Not Null |
| **time** | TIME | 6 | Not Null |







## Table 3.3.1.5: QUESTION TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **question** | VARCHAR | 3000 | Not Null |
| **answer** | VARCHAR | 3000 | Not Null |
| **LOGIN\_id** | BIGINT | 20 | Foreign Key |

## Table 3.3.1.6: RESULT TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **TEST\_id** | BIGINT | 20 | Foreign Key |
| **USER\_id** | BIGINT | 20 | Foreign Key |
| **mark** | BIGINT | 20 | Not Null |

## Table 3.3.1.7:TEST QUESTIONS TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **TEST\_id** | BIGINT | 20 | Foreign Key |
| **QUESTION\_id** | BIGINT | 20 | Foreign Key |
| **option1** | VARCHAR | 100 | Not Null |
| **option2** | VARCHAR | 100 | Not Null |
| **option3** | VARCHAR | 100 | Not Null |
| **option4** | VARCHAR | 100 | Not Null |
| **correct\_answer** | VARCHAR | 100 | Not Null |



## Table 3.3.1.8:REVIEWS TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **date** | DATE |  | Not Null |
| **type** | VARCHAR | 300 | Not Null |
| **review** | VARCHAR | 300 | Not Null |
| **rating** | VARCHAR | 5 | Not Null |
| **LOGIN\_id** | BIGINT | 20 | Foreign Key |

## Table 3.3.1.9:COMPLAINT TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **LOGIN\_id** | BIGINT | 20 | Foreign Key |
| **date** | DATE |  | Not Null |
| **complaint** | VARCHAR | 300 | Not Null |
| **reply** | VARCHAR | 300 | Not Null |
| **status** | VARCHAR | 100 | Not Null |

## Table 3.3.1.10:STAFF COMPLAINT TABLE

| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| --- | --- | --- | --- |
| **id** | BIGINT | 20 | Primary Key |
| **USER\_id** | BIGINT | 20 | Foreign Key |
| **STAFF\_id** | BIGINT | 20 | Foreign Key |
| **complaint** | VARCHAR | 300 | Not Null |
| **status** | VARCHAR | 100 | Not Null |
| **date** | DATE |  | Not Null |



## Table 3.3.1.11:FEEDBACK

## Table 3.3.1.12:LAW

## Table 3.3.1.13:NEXT HEARING DATETABLE



## NORMALIZATION

Database Normalization is a methodical technique for structuring data within a database. It involves decomposing tables to eliminate data redundancy and to address undesirable anomalies such as Insertion, Update, and Deletion Anomalies. This multi-step process transforms data into tabular form, removing duplicated data from relational tables. In our design, tables have been normalized up to the third normalization form, which includes :

* FirstNormalForm(1NF)
* SecondNormal Form(2NF)
* ThirdNormalForm(3NF)

## FirstNormalForm

Our application complies with the requirements of the first normal form (1NF), which stipulates that a relation should not contain composite or multi-valued attributes. In other words, for a relation to be in first normal form, every attribute within it must be single-valued. The first normal form states that:

* Every column in the table must be unique
* Separate tables must be created for each set of related data.
* Each table must be identified with a unique column or concatenated columns known as the primary key.
* No rows maybe duplicated
* No columns may be duplicated
* No row/column intersections contain a null value

## Second Normal Form

A relation achieves second normal form (2NF) only when it meets the conditions of first normal form (1NF) for the primary key and when every non-primary key attribute is fully dependent on the primary key alone. If a non-key attribute does not rely on the primary key, it should be extracted from the relation and placed in a separate one. In other words, all fields in a table that is in 2NF are directly related to the primary key.

## Third Normal Form

A relation achieves third normal form (3NF) when it satisfies the conditions of being in second normal form (2NF), and additionally, non-key attributes do not depend on other non-key attributes. Storing and retrieving data from the database are essential components of system design. At the analysis stage, data elements and structures to be stored are identified, structured, and integrated to create the data storage and retrieval system..

## INPUT DESIGN

Input design serves as the bridge between the information system and the user, encompassing the development of specifications and procedures for data preparation. These steps are essential for transforming transaction data into a usable format for processing, which can involve inspecting the computer, reading data from written or printed documents, or inputting data directly into the system. The primary focus of input design is to regulate the input quantity, minimize errors, prevent delays, streamline processes, and maintain simplicity. Additionally, input design aims to ensure security, user-friendliness, and privacy retention.



## OUTPUT DESIGN

Quality output refers to meeting the end user's requirements and presenting information clearly. In any system, the outcomes of processing are communicated to users and other systems through outputs. Output design entails determining how information is displayed for immediate use and also generating hardcopy output. It serves as the primary and direct source of information for users. Effective and intelligent output design enhances the system's ability to assist user decision-making..



# CHAPTER-IV

**CODING**



## 4.1 FRONTEND:

**HTML**

Hypertext Markup Language (HTML) serves as the universal markup language for crafting web pages. Its purpose is to outline the framework of a webpage, employing various elements to instruct browsers on how to render content. HTML is widely embraced by developers for website creation due to its straightforward nature, making it accessible and uncomplicated to grasp..

**CSS**

Cascading Style Sheets (CSS) is a language utilized for defining the visual appearance of a document authored in a markup language like HTML or XML (including XML variations such as SVG, MathML, or XHTML). As a foundational component of the World Wide Web, CSS, alongside HTML and JavaScript, plays a crucial role.

CSS is engineered to facilitate the segregation of content and presentation, encompassing aspects like layout, colors, and typography. This division enhances content accessibility, affords greater flexibility and precision in delineating presentation attributes, allows for the reuse of formatting across multiple web pages by specifying CSS in a separate .css file, thus minimizing redundancy and complexity in structural content. Furthermore, caching the .css file enhances page loading speed between pages that share the same formatting, optimizing overall user experience.

**JAVASCRIPT**

JavaScript, often abbreviated as JS, is a high-level, versatile programming language primarily used for enhancing interactivity and functionality in web pages. As a cornerstone technology of the web alongside HTML and CSS, JavaScript enables dynamic updates, interactive features, and complex behavior within web applications. It runs on the client side, typically within a web browser, but can also be executed on the server side through platforms like Node.js. With its broad adoption and extensive ecosystem of libraries and frameworks, JavaScript has become indispensable in modern web development, empowering developers to create dynamic, responsive, and interactive web experiences..



## 4.2 BACKEND:

**PYTHON**

Python is a versatile programming language that supports multiple paradigms, including object-oriented programming, structured programming, functional programming, and aspect-oriented programming. It features dynamic typing and utilizes reference counting and a cycle-detecting garbage collector for memory management. Python also offers dynamic name resolution, binding method and variable names during program execution. Known for its ease of learning and platform independence, Python boasts extensive libraries and supports both procedural-oriented programming (POP) and object-oriented programming (OOP). Hence, we have chosen Python as the backend language for our web development.

.

## PYCHARM IDE

PyCharm is a popular integrated development environment (IDE) specifically designed for Python programming. Developed by JetBrains, it offers a wide range of features to enhance productivity and streamline the development process. PyCharm provides smart code completion, syntax highlighting, and code analysis to help developers write clean and efficient code. It also includes powerful debugging tools, version control integration, and support for web development frameworks like Django and Flask. With its user-friendly interface and extensive plugin ecosystem, PyCharm is a preferred choice for Python developers seeking a comprehensive development environment.



**MySQL**

MySQL is an open-source relational database management system (RDBMS) supported by Oracle and built around Structured Query Language (SQL). It is compatible with various platforms such as Linux, UNIX, and Windows. While MySQL finds application in diverse domains, it is particularly renowned for its association with web applications and online publishing. Offering high performance and memory efficiency, MySQL is known for its speed and user-friendliness. Given its simplicity and efficiency, we have opted for MySQL as the backend database solution.

TECHNOLOGIES USED:

Machine Learning (ML) transformers

Machine Learning (ML) transformers are a groundbreaking architecture in natural language processing (NLP) and machine translation. They employ a mechanism known as self-attention to process input data in parallel, allowing for more efficient learning and improved performance on sequential tasks. Transformers have revolutionized the field of NLP by enabling the training of deep neural networks on large-scale datasets, leading to state-of-the-art results in tasks such as language translation, text generation, and sentiment analysis. They have become a cornerstone in many modern NLP applications and have greatly contributed to advancements in the field.



# CHAPTER-V

**TESTING AND IMPLEMENTATION**

## 5.1 TESTING

Software testing involves the systematic examination of a system to uncover errors, discrepancies, or deviations from the specified requirements. It is broadly divided into two categories: functional testing and non-functional testing. Ideally, testing should commence as early as possible in the development process to minimize rework and ensure the delivery of bug-free software to clients. However, the exact starting point for testing within the Software Development Life Cycle (SDLC) can vary. Testing may begin as early as the Requirements Gathering phase and continue throughout the development process until the software is deployed in production. The timing of testing activities can also be influenced by the chosen development model. For instance, in the Waterfall model, testing typically begins after the development phase, whereas in the V-model, testing is conducted concurrently with development activities.

## 5.1.1 TESTING METHODOLOGY MANUAL TESTING

Manual testing is a software testing approach where test cases are executed manually, without the use of automated tools. Testers follow end-user perspectives to ensure that the application aligns with the requirements outlined in the documentation. Test cases are meticulously planned and executed to achieve near-complete coverage of the software application. Additionally, manual test reports are generated without automation.

This method is fundamental as it uncovers both apparent and latent defects in the software. Defects, defined as discrepancies between expected and actual outputs, are identified by testers and reported for resolution by developers.

Prior to automated testing, manual testing is imperative for newly developed software. Despite requiring substantial time and effort, manual testing guarantees software reliability. Unlike automated testing, manual testing relies solely on manual testing techniques and does not require knowledge of automated testing tools.

Test engineers who conduct manual testing gain insights into the application from an end-user perspective, facilitating the creation of accurate test cases and prompt feedback on the application's performance.

## AUTOMATION TESTING

Automation testing involves executing test cases using specialized automation tools. These tools control the test case execution and compare actual results with expected outcomes. Automation testing demands significant resources and financial investment.

Primarily, automation testing focuses on repetitive actions, such as regression tests. The tools employed in automation testing serve various purposes, including automated GUI interaction, data setup generation, defect logging, and product installation. The objective of automation testing is to decrease manual test cases while retaining their importance. Automation tools can record test suites, allowing testers to replay them as needed without human intervention.

## 5.1.2 DIFFERENT TESTING

* + - * UnitTesting
      * IntegrationTesting
      * SystemTesting
      * AcceptanceTesting
      * ValidationTesting

## UNITTESTING

The initial stage of testing is referred to as unit testing, where individual modules undergo examination against the specifications outlined during design. Unit testing is crucial for verifying module functionality and assessing internal logic. It is performed across various modules within the project, enabling immediate identification and rectification of errors to enhance program clarity. Testing is integrated into the programming stage itself, ensuring each module operates satisfactorily as per expectations. For instance, the Login page is subjected to testing under three distinct states: positive input, negative input, and zero input. While testing with positive and negative inputs yields expected outcomes, zero input may lead to unexpected results. This underscores the importance of thoroughly testing different states of the code.



## INTEGRATION TESTING

The next level of testing, known as integration testing, involves systematically examining the construction structure of the system. Simultaneously, tests are performed to detect errors within the interface. It is not guaranteed that software modules, which exhibit satisfactory results when run individually, will also perform flawlessly when integrated. Therefore, individual modules are retested, and the results are validated to assess their integration with one another. This testing activity essentially evaluates the design and emphasizes the interaction between modules..

* TopDownIntegration

This approach is incremental in nature and focuses on constructing the program structure step by step. Modules are integrated by progressing downwards through the control hierarchy, starting with the main program module. Subordinate modules are then incorporated into the structure, following either a depth-first or breadth-first manner.

* BottomUpIntegration

This approach initiates the construction and testing with the modules positioned at the lowest level in the program structure. By integrating modules from the bottom upwards, the necessary processing for subordinate modules at any given level is consistently accessible, thereby eliminating the need for stubs.

## SYSTEM TESTING

Testing is a process aimed at confirming the development of a correct system, with the objective of uncovering any faults within it. This activity is not limited to the post-development phase but is ideally conducted concurrently with all stages of system development, beginning with requirements specification. The results of testing, once collected and analyzed, offer a qualitative assessment of software quality and reliability. They also serve as a foundation for potential design modifications if deemed necessary. A project is considered incomplete without thorough testing.



System testing involves verifying if the developed system aligns with the initial objectives and requirements. It entails experimental testing with test data to validate that the system operates as per the specified requirements. Once the system is confirmed to be functional, it is then tested with actual data to assess its performance.

## ACCEPTANCE TESTING

Acceptance testing is a method used to assess whether the software system fulfills the specified requirements. Its primary objective is to evaluate the system's alignment with business requirements and confirm if it meets the necessary criteria for delivery to end users.

## VALIDATION TESTING

Validation testing is conducted for each input form to ensure that only permissible values are entered. This process involves testing with incorrect values to validate whether errors are properly detected and handled. Incorrect values are expected to be rejected, and any identified errors are addressed accordingly. Validation verifies if the system aligns with requirements, fulfills intended functions, and meets the organization's objectives and user needs. This validation occurs throughout various testing phases, including feature testing, integration testing, system testing, load testing, compatibility testing, stress testing, and others.

## Front-EndValidation

All validation processes are implemented on the server to prevent unauthorized access or manipulation of the database through alternative clients. The frontend validation is also essential to enhance efficiency and prevent inappropriate data access on the server.

Frontend validation primarily serves to aid data entry and provide contextual messages. This ensures a smooth data entry experience for users and reduces the need for repeated validation checks. Front-end validation encompasses all input verification in modern applications, offering users prompt feedback on potential issues.

In the user interface (UI), basic input validation such as mandatory field checks and email address validity verification is conducted. Additionally, UI controls are updated or disabled based on the validation outcomes..

## Back-EndValidation

Backend validation is equally crucial, as it must guarantee the validity of incoming data. Moreover, depending on the architecture, the middle-tier business logic is often reused across various components. Therefore, it is essential to ensure that the rules applied remain consistent regardless of the frontend logic enforcement.



## 5.1.3 TEST CHART

A test case is a document containing a defined set of test data, preconditions, expected outcomes, and post conditions. It is crafted for a specific test scenario to verify compliance with particular requirements.

The test case serves as the initial step for executing tests. After applying a set of input values, the application produces a definitive outcome and reaches a conclusion, also referred to as the execution postcondition..

## Test Case For Login:



| Test ID | TESTCASE | TESTDATA | EXPECTEDRESULT | ACTUALRESULT | REMARK |
| --- | --- | --- | --- | --- | --- |
| 1 | check admin login with valid data | Email:[admin@gmail.com](mailto:admin@gmail.com)  password :admin | adminshouldlogintotheadmindashboard | login successful | Pass |
| 2 | check admin login with invalid data | username:admin@gmail.compassword:abcd | admin should fail to login to admin dashboard | login failed | Pass |
| 3 | check admin login by emptying  the data and login button pressed | nodata | alerted that field is empty | trigger an error alert | Pass |
| 4 | check  Advocateloginwithvaliddata | user name:[user@gmail.com](mailto:user@gmail.com)  password:user | advocateshouldlogintotheadmindashboard | login successful | Pass |
| 5 | check | username: | advocate  should | login  failed | Pass |

|  | Advocate Login with invalid data | [user@gmail.com](mailto:user@gmail.com)  password:abcd | fail to login to admin dashboard |  |  |
| --- | --- | --- | --- | --- | --- |
| 6 | checkAdvocatelogin byemptyingthedataandlogin button pressed | nodata | alerted thatfield is empty | trigger an error alert | Pass |
| 7 | check  Userloginwithvaliddata |  | member should login  to the android application |  |  |
| 8 | check User login with invalid data |  | member  shouldfail  to login to android  application |  |  |
| 9 | checkUserlogin byemptyingthedataandlogin button pressed |  | alerted thatfield is empty |  |  |



**Test Case For Registration(user):**



| Test Id | TestCase | TestData | ExpectedResult | ActualResult | Remark |
| --- | --- | --- | --- | --- | --- |
| 1 | check registration form  by emptying  the  data and submit button pressed | no data | form shouldfail  to enter data and the user should alerted to enter remaining data | trigger an error alert | Pass |
| 2 | check mobile number field by entering non numeric. | testchar | user should prompt to enter a valid mobile number | trigger an error alert | Pass |
| 3 | Entering Invalid Email Format | abc.gmail.com | Prompt to enter a valid email id | trigger an error alert | Pass |

## 5.2 QUALITY ASSURANCE POLICIES

## 5.2.1 GENERIC RISKS

“Risk is future uncertain events with a probability of occurrence and potential for loss” Risk Identification and management are the main concerns in every software project. Effective Analysis of software risks will help to effective planning and assignments of work. Risks are identified,classified and managed beforetheactualexecution ofthe program.

TheseRisksareclassifiedintodifferentcategories:

1. **Schedule Risk:** Project schedule slips when project tasks and schedule release risks arenotaddressedproperly.Schedulerisksmainlyaffectaprojectandfinallyoncompanyeconomyandmay lead to projectfailure.

## Schedulesoftenslipduetothefollowingreasons:

* + Wrong Time Estimation
  + Resourcesarenottrackedproperly.Allresourceslikestaff,systems,skills individuals,etc.
  + Failuretoidentifycomplexfunctionalitiesandtimerequiredtodevelopthosefunctionalities.
  + Unexpected Project Scope Expansions.

## BudgetRisk

* + Wrong Budget Estimation.
  + Cost Overruns
  + Project Scope Expansion

1. **OperationalRisks:** Risksoflossduetoimproperprocessimplementationfailedsystemorsomeexternaleventsrisks. Causes Of OperationalRisks:
   * Failure To Address Priority Conflicts
   * Failure To Resolve The Responsibilities
   * Insufficient Resources
   * No Proper Subject Training
   * No Resource Planning



* + No Communication In The Team.

1. **TechnicalRisks:**Technicalrisksgenerallyleadtofailureoffunctionalityandperformance.Causes of TechnicalRisksare:
   * Continuous Changing Requirements
   * Noadvancedtechnologyavailableortheexistingtechnologyisintheinitialstages.
   * The Product Is Complex To Implement.
   * Difficult Project Modules Integration.
2. **ProgrammaticRisks:** Thesearethe External Risks beyondtheoperationallimits.Thesearealluncertainrisks areoutsidethecontrolofthe program. These External Events Can Be:
   * Running Out Of The Fund.
   * Market Development
   * Changing Customer Product Strategy And Priority
   * Government Rule Changes.

## 5.3 SYSTEMIMPLEMENTATION

System implementation is the final phase i.e., putting the utility into action. Implementation isthe state in the project where theoretical design turned into working system. Implementationinvolves theconversion of a basic application tocomplete replacement with a computersystem. It is the process of converting to a new or revised system design into an operationalone.Duringthedesignphase,theproductsstructure,itsundergoingdatastructures,thegeneralalgorithmsandtheinterfacesandcontrol/datalinkagesneededtosupportcommunication among the various sub structures were established. Implementation process issimply a translation of the design abstraction into the physical realization, using the languageofthetargetarchitecture.

There Are Three Types Of Implementation:

* Implementation of a computer system to replace a manual system.
* Implementation of a new computer system to replace an existing one.
* Implementationofamodifiedapplicationtoreplaceanexistingonecomputer.

The Common Approaches For Implementation Are:

## ParallelConversion

In parallel conversion the existing system and new system operate simultaneously until the project team is confident that the new system is working properly. The outputs from the old system continue to be distributed until the new system has proved satisfactorily parallel conversion is a costly method becauseoftheamountofduplication involved.

## DirectConversion

Under the direct conversion method the old system is discontinued altogether and the new system becomes operational immediately. A greater risk is associated with direct conversion is no backup in thein the case of system fails.

## PilotConversion

A pilot conversion would involve the changing over of the part of the system either in parallel or directly. Use of the variation of the two main methods is possible when part of the systemcanbetreatedasa separate entity.

## UserTraining

After the system is implemented successfully, training of the user is one of the most important subtasks of the developer. For this purpose, user manuals are prepared and handled over to the user to operate the developed system. Thus, the users are trained to operate the developed system.

Inordertoputnewapplicationsystemintouse,thefollowingactivitiesweretakencareof:

* Preparation Of User And System Documentation.
* Conducting User Training With Demo And Handson.
* Testrunforsomeperiodtoensuresmoothswitchingoverthesystemet.

## 5.3.1 IMPLEMENTATION PROCEDURES

The Major Implementation Procedures Are:

* Test Plans
* Training
* Conversion

## 5.4 SYSTEM MAINTENANCE

Themaintenanceisanimportantactivityinthelifecycleofasoftwareproduct.Maintenanceincludesalltheactivitiesaftertheinstallationofsoftwarethatisperformed to keep the system operational. The maintenance phase of a software lifecycle is the time period in which a product performs useful work. Maintenance is classified into four types.

* CorrectiveMaintenance
* AdaptiveMaintenance
* PerfectiveMaintenance
* PreventiveMaintenance

## CORRECTIVE MAINTENANCE

Correctivemaintenancereferstochangesmadetorepairdefectsinthedesign,coding,or implementation of the system. Corrective maintenance is often needed for repairingprocessingorperformancefailuresormakingchangesbecauseofpreviouslyuncorrectedproblemsorfalseassumptions.Mostcorrectivemaintenanceproblemssurface soon after the installation. When corrective maintenance problems surface, theyare typically urgent and need to be resolved to curtail possible interruptions in normalbusiness activities.



## ADAPTIVE MAINTENANCE

Adaptive maintenance involves making changes to an information system to evolve its functionality or to migrate it to a different operating environment. Adaptive maintenance is usually less urgent than corrective maintenance because business and technical changes typically occur some period of time.

## PERFECTIVE MAINTENANCE

Perfectivemaintenanceinvolvesmakingenhancementstoimproveprocessingperformance, interface usability, or to add desired, but not necessarily required, system features. Many system professionals feel that perfective maintenance is not really the maintenance but new development.

## PREVENTIVE MAINTENANCE

Preventive maintenance is the only maintenance activity which is carried out without formal maintenance request from the user. When a software company or maintenance agency realizes that the methodologies used in a program have become obsolete, it may decide to develop or modify parts of the program, which do not conform to the current trends. Of these types, more time and money are spent on perfecting than on corrective and adaptive maintenance together .



# CHAPTER-VI CONCLUSION

## 6.1 CONCLUSION

Legal Advisor is a user friendly app that helps users to find and book the advocates according

to their requirements. Legal Advisor includes the information about the

advocates. This app provides a platform for the users to communicate with the advocates via

online.

## 6.2 SCOPE FOR FURTHER ENHANCEMENTS

Nowadays the role of Legal Consultants is very important because of the popularity of

increasing crimes. So the aim of Legal Advisor is to help the users to find the advocates

according to their requirements. Thus Legal Advisor is useful and very relevant app at

present and future.

## 6.3 BIBLIOGRAPHY

### Textbooks

* Elmasri R.& Navathe S. (2007). Fundamentals of database systems. Boston:Pearson/AddisonWesley.
* Sommerville.(2016).Software Engineering.Harlow:PearsonEducation.
* DarwinI.F.(2017).AndroidCookbook:ProblemsandSolutionsforAndroidDevelopers.Beijing:OReilly Media.

### WebSites

* [www.w3schools.com](http://www.w3schools.com/)
* www.tutorialspoint.com

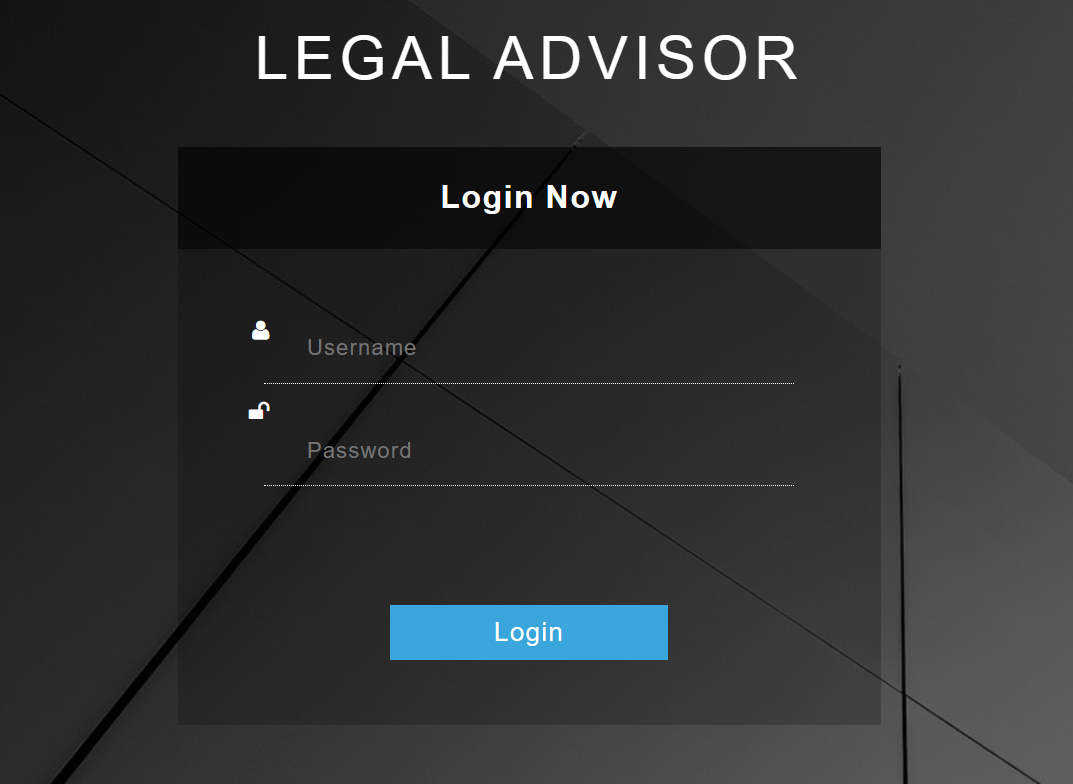


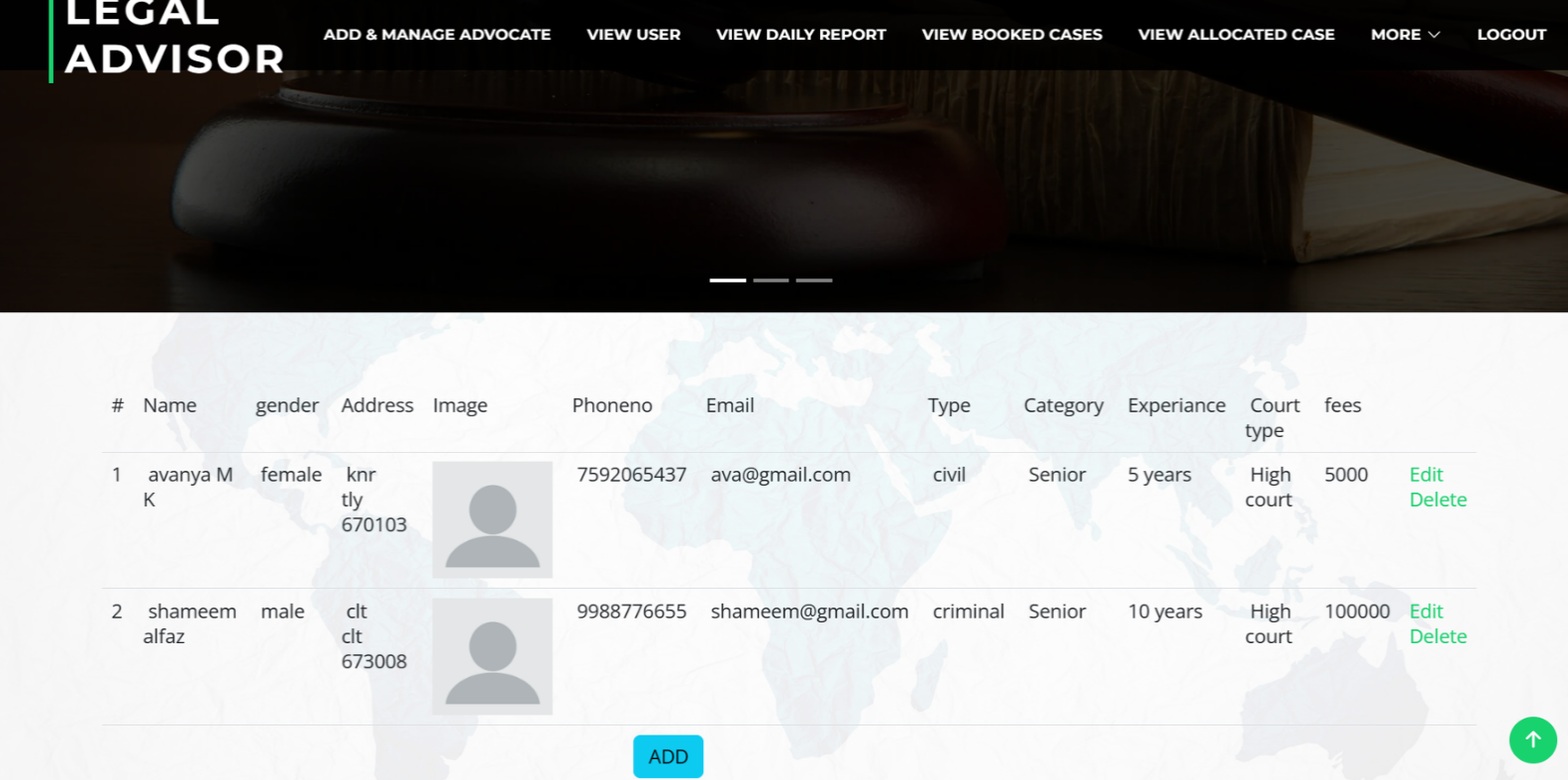
# 

# ANNEXURE-A

**1.INPUT AND OUTPUT DESIGN**

# WEBPART

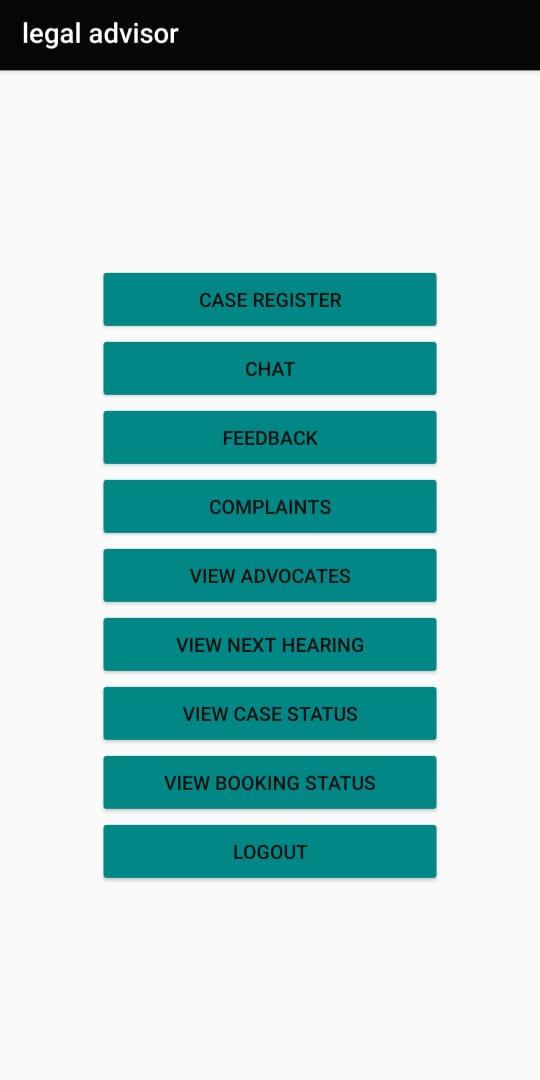
****



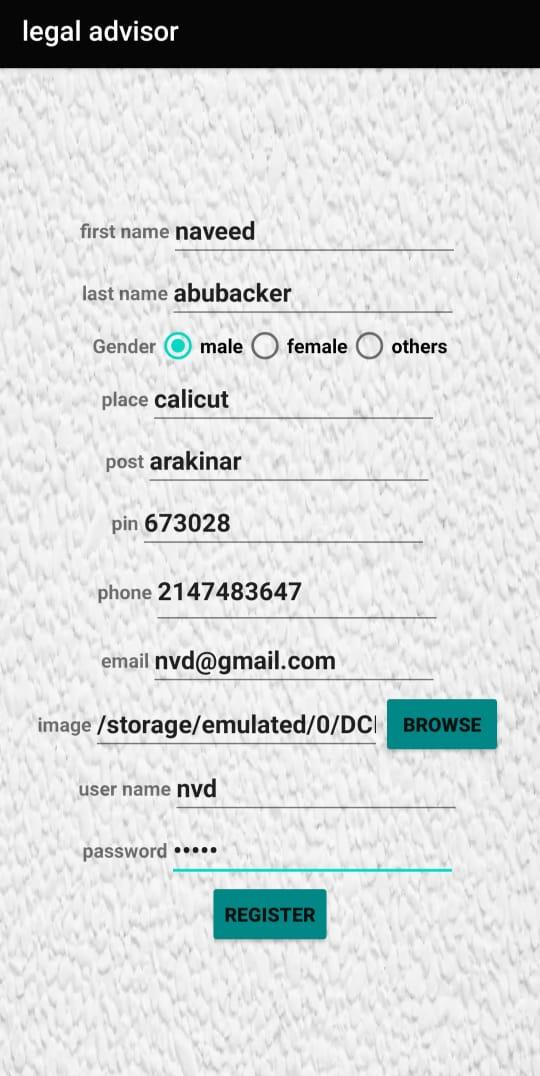
1. **ANDROIDPART**



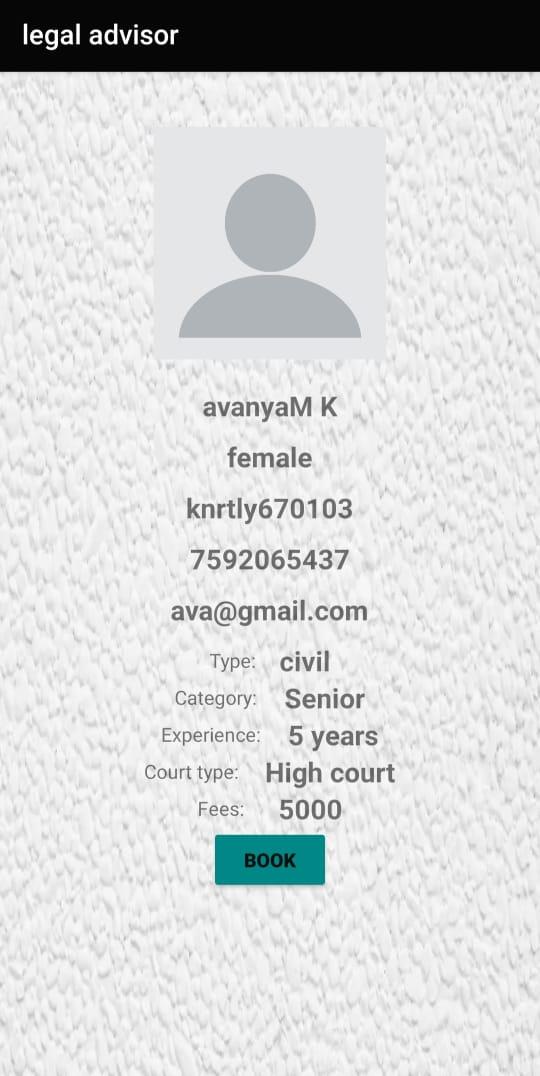












# 

# 2.SAMPLESOURCE

package com.example.legal advisor;  
  
  
import android.content.Intent;  
import android.content.SharedPreferences;  
import android.os.Bundle;  
import android.preference.PreferenceManager;  
import android.support.v7.app.AppCompatActivity;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import com.android.volley.Request;  
import com.android.volley.RequestQueue;  
import com.android.volley.Response;  
import com.android.volley.VolleyError;  
import com.android.volley.toolbox.StringRequest;  
import com.android.volley.toolbox.Volley;  
  
import org.json.JSONException;  
import org.json.JSONObject;  
  
import java.util.HashMap;  
import java.util.Map;  
  
public class login extends AppCompatActivity {  
  
 EditText e1,e2;  
 Button b1,b2;  
 String uname,pass,url;  
 SharedPreferences sh;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_login*);  
 e1 = findViewById(R.id.*editTextTextPersonName2*);  
 e2 = findViewById(R.id.*editTextTextPassword*);  
 b1 = findViewById(R.id.*button2*);  
 b2 = findViewById(R.id.*button3*);  
 sh= PreferenceManager.*getDefaultSharedPreferences*(getApplicationContext());  
  
 b1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 uname = e1.getText().toString();  
  
 pass = e2.getText().toString();  
 if (uname.equalsIgnoreCase("")) {  
 e1.setError("Enter your username");  
 e1.requestFocus();  
 } else if (pass.equalsIgnoreCase("")) {  
 e2.setError("Enter your password");  
 e2.requestFocus();  
 } else {  
 RequestQueue queue = Volley.*newRequestQueue*(login.this);  
 url = "http://" + sh.getString("ip", "") + ":5000/login";  
// url = "http://192.168.63.7:5000/login";  
  
 // Request a string response from the provided URL.  
 StringRequest stringRequest = new StringRequest(Request.Method.*POST*, url, new Response.Listener<String>() {  
 @Override  
 public void onResponse(String response) {  
 // Display the response string.  
 Log.*d*("+++++++++++++++++", response);  
 try {  
 JSONObject json = new JSONObject(response);  
 String res = json.getString("task");  
  
 if (res.equalsIgnoreCase("success")) {  
 String lid = json.getString("id");  
 SharedPreferences.Editor edp = sh.edit();  
 edp.putString("lid", lid);  
 edp.commit();  
 Intent ik = new Intent(getApplicationContext(), home.class);  
 startActivity(ik);  
  
 } else {  
  
 Toast.*makeText*(login.this, "Invalid username or password", Toast.*LENGTH\_SHORT*).show();  
  
 }  
 } catch (JSONException e) {  
 e.printStackTrace();  
 }  
  
  
 }  
 }, new Response.ErrorListener() {  
 @Override  
 public void onErrorResponse(VolleyError error) {  
  
  
 Toast.*makeText*(getApplicationContext(), "Error" + error, Toast.*LENGTH\_LONG*).show();  
 }  
 }) {  
 @Override  
 protected Map<String, String> getParams() {  
 Map<String, String> params = new HashMap<String, String>();  
 params.put("uname", uname);  
 params.put("pswd", pass);  
  
 return params;  
 }  
 };  
 queue.add(stringRequest);  
  
  
 }  
  
  
 }  
 });  
  
 b2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent in=new Intent(getApplicationContext(),register.class);  
 startActivity(in);  
  
  
 }  
 });  
 }  
}



# ANNEXURE-B

**ABBREVIATION**

* **IDE**-IntegratedDevelopmentEnvironment
* **HTML-**HypertextMark-upLanguage
* **SQL**-StructuredQueryLanguage
* **UI**-UserInterface
* **PK**-PrimaryKey
* **FK**-ForeignKey
* **SVGA–**StandardVideoGraphicsArray