

1. Introduction

In today's time, inspite of technology advancement still all the admission process be it be the college or school admission it is being carried out manually using paper and pen which is very inefficient and utilizes much efforts and time. In this present century of technology where computers are being used everywhere, they are being used as an alternative to pen and paper. The database can digitally store more information about users in less space and in very less time the very basic principle behind the necessity automated college online admission system is peaceful observation of institutes. This tool can easily store the details of students such as fee details, documents details and admission details. This Student Database has been designed taking considering the practical needs to manage a student's data. Its design focuses on 2 types of users:

- Admin
- Students

The Database follows a unique event flow developed for such a system. Communication between the student/parents and the institution management is the sole purpose of this software along with reducing the paper work to some extent. This system benefits the administrator to access and verify the information of students. This Automated Online College Admission Management system benefits the administrator as he can send pushing notifications and make announcements which are directly received by the students and parents. The concept of Personal Registration Number or PRN is also included using which the each student gets one inimitable Identification. This id will help in future to access information or find student among several students of the college.

The objective of online college admission system is to reduce the manual work and excess time taken for admission process. Student can login into the system by signing up. They can fill the form of admission, revaluation, and photocopy. Also the details regarding the updated information of their admission is displayed on the window. The online college admission system ease the admission process by maintaining the database and retrieving the information of student easily. This system aims at being efficient and user-friendly.

2. Literature Survey

This paper aims to develop an Online Intranet College Management System (CMS) which is useful to any education institution. The system (CMS) is an Intranet based application that can be accessed throughout the institution or a specified department. This system aims to monitor the attendance of students for the college. Any information regarding college is accessible to students as well as staff members. The staff uploads their and students attendance and also the marks of the students are maintained. Easy access to information is given to registered users. CMS aims to provide information to all the levels of management in any institution.

Student Information Management System (SIMS) aims to provide an interface to maintain student information. Educational institutes or colleges can use this system to maintain information of students. The Student information system maintains all kind of details regarding students, college, course, batch, placements, academic progress report and other resource related details too. Student details can be tracked from day one to last day which can be useful to maintain records.

Global systems for mobile communication is considered as the reliable and efficient technology for most of the technological devices. GSM used is to know the information about the student whereabouts and his activities completely. The RFID is used to integrate the parts of the student in order to track the student there itself.

Information system is useful in accessing student's data from anywhere and anytime. It is one of the user-friendly ways of accessing information. Many organizations spend a lot of cost in developing such information systems for the reliability in their organization. This project develops student information system which focusses on recording, storing and updating the student details. The new system uses the Rapid Application Development model for the software development. The database will be storing all the student records. This system provide ease to the users as well as professor for arranging the lectures.

Biometric is one of the most reliable technology in today's time. This system provides a number of advantages to the security aspect as well as people. It also eliminates the buddy-punching which was a major problem in earlier biometric systems. In earlier biometric systems the process was carried out by matching the fingerprints with the one stored in the database which was a time taking process. This paper aims at developing an automated online attendance system by reconstructing the fingerprints by minutiae templates. This automated system also manages and maintains a record of attendance for a particular

period of time. The system is also evaluated in order to identify the success rate of the attacks and after evaluation it can be seen that type-1 and type-2 attacks can be effectively.

3. System Analysis

3.1 Existing System

The main aim of our project is to advance the application which will be useful in the college admission system. The Automated Online College Admission Management System Project comprises the rank allotment, documents upload, admittance procedure of student initial from when the student takes admittance in college in first year to final year of the student. The requisite of the student is to: Login to the system through the first page of the application. View/change his/her details. Can get help through the help of choice to view different features of the system. Admin login is present through which the official person can verify the details, modify and update the branches and maintain the student's information.

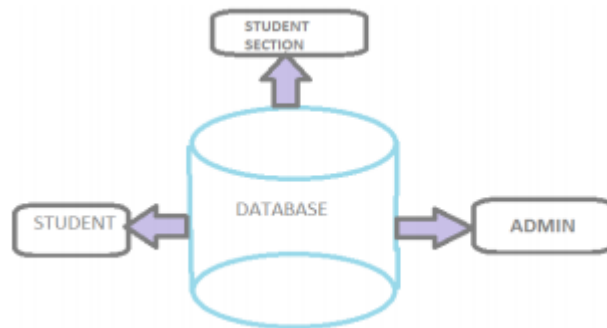
3.2 Proposed System

Need of proposed system:

The need of this system is to maintain proper user-friendly communication. The scheme, which is created, now generates all the details that are recalled manually. Once the details are saved into the system there is no need for numerous persons to deal with distinct sections. Only a particular person is enough to preserve all the reports. The haven can also be given as per the implementing needs of the user and those needs are: big volumes of data can be stored with ease. Records stored are modified with much ease without taking numerous efforts. Stored data and procedures can be easily modified. Smooth calculations are done, and the Main important benefit from this system is it provides a good way of managing the thing in a better implemented way and large amount of work can be done in less interval of time

System design:

This plays with how the data flow through the various processes, comprehensive and detailed flow graph and the procedure flow of the front end and back end make of the student information running system. There is one database which deals with the student section, student counselling process which holds its verification and validation process and admin section which holds the authorities for all the processes to be implemented.

SYSTEM ARCHITECTURE:**3.3 Modules Description****USER**

The student who wants to seek admission is the user of the online system. The student registers himself/herself with the system and becomes authenticated user after the login process. The students can seek admission in 2nd, 3rd and 4th year of engineering college as per their academic year. The admission form is filled by the student after logging in the system and the documents such as previous years mark sheets, Aadhar card, caste certificates, etc. are scanned and uploaded in the system. The application forms for provisional admission, photocopy and revaluation can be filled the user in the similar fashion. All these details can be accessed by the admin via server. If the admin verifies the form and grants permission to the user for admission, a message will be generated and sent to the student ID through server. If an unauthenticated user tries to login in the system, he/she is not permitted to do so.

SERVER

The Server acts as a medium between the authenticated user (student) and the admin. All the details entered, and documents uploaded by the student is stored in the server (Wamp Server). The admin can access these details via server. When the admin verifies the student details and grants permission for regular admission/provisional admission/photocopy/revaluation process, a message will be generated to the user sent by admin via server.

ADMIN

The admin plays a pivotal role in the online admission system. It is the admin who provides authentication to the user in the online process. The details entered and the documents uploaded by the authenticated user is accessed by the admin via server. These details are checked, verified and authenticated by the admin. If the details entered and documents uploaded are genuine, the admin approves the form and a message is sent to the user via server. The modification rights in the system to the user is granted by the admin. The admin acts as a supreme authority to approve, accept and reject the application forms for admission.

3.4 Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are,

3.5. Economic Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased

3.6 Operational Feasibility

Are you into the production of “things”? Perhaps, your answer would be yes. We naturally don’t call them things; instead, we call them products, services, or systems. Using the term “things” sounds foreign because you can’t just drop them into an area without touching them. They need to be connected to an existing service or business. These “things” are an extension of the organization where they are produced.

3.7 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead

to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system

3.8 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system

4. System Requirements Specification

4.1 Introduction

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well-ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers

4.2 Purpose

In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

4.3 Functional Requirements

For developing the application, the following are the Software Requirements:

1. Python
2. anaconda

Operating Systems supported

1. Windows 7
2. Windows XP
3. Windows 8

Technologies and Languages used to Develop

1. Python

Debugger and Emulator

- Any Browser (Particularly Chrome)

Hardware Requirements

For developing the application the following are the Hardware Requirements:

- Processor: Pentium IV or higher
- RAM: 256 MB

Space on Hard Disk: minimum 512MB

4.4 Non Functional Requirements

- Any Browser (Particularly Chrome)

4.5 Input & Output Design

4.5.1 Input design:

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.

Methods for preparing input validations and steps to follow when error occur

OBJECTIVES

1.Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3.When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

4.5.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.

Confirm an action

4.6 Hardware Requirements

For developing the application the following are the Hardware Requirements:

- Processor: Pentium IV or higher
- RAM: 256 MB

Space on Hard Disk: minimum 512MB

4.7 Software Requirements

For developing the application the following are the Software Requirements:

3. Python

4. anaconda

Operating Systems supported

4. Windows 7
5. Windows XP
6. Windows 8

Technologies and Languages used to Develop

Python

5. System Design

5.1 System Design

The purpose of the design phase is to plan a solution of the problem specified by the requirement document. This phase is the first step in moving from the problem domain to the solution domain. In other words, starting with what is needed, design takes us toward how to satisfy the needs. The design of a system is perhaps the most critical factor affecting the quality of the software; it has a major impact on the later phase, particularly testing, maintenance. The output of this phase is the design document. This document is similar to a blueprint for the solution and is used later during implementation, testing and maintenance. The design activity is often divided into two separate phases: System Design and Detailed Design.

System Design also called top-level design aims to identify the modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired results. At the end of the system design all the major data structures, file formats, output formats, and the major modules in the system and their specifications are decided.

During Detailed Design, the internal logic of each of the modules specified in system design is decided. During this phase, the details of the data of a module are usually specified in a high-level design description language, which is independent of the target language in which the software will eventually be implemented.

In system design the focus is on identifying the modules, whereas during detailed design the focus is on designing the logic for each of the modules. In other words, in system design the attention is on what components are needed, while in detailed design how the components can be implemented in software is the issue.

Design is concerned with identifying software components specifying relationships among components. Specifying software structure and providing blue print for the document phase. Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. In such a manner, the interaction between parts is minimal clearly specified.

During the system design activities, Developers bridge the gap between the requirements specification, produced during requirements elicitation and analysis, and the system that is delivered to the user.

Design is the place where the quality is fostered in development. Software design is a process through which requirements are translated into a representation of software

5.2 UML Diagrams (9 types)

Any complex system is best understood by making some kind of diagrams or pictures. These diagrams have a better impact on our understanding. If we look around, we will realize that the diagrams are not a new concept but it is used widely in different forms in different industries.

We prepare UML diagrams to understand the system in a better and simple way. A single diagram is not enough to cover all the aspects of the system. UML defines various kinds of diagrams to cover most of the aspects of a system. You can also create your own set of diagrams to meet your requirements. Diagrams are generally made in an incremental and iterative way. There are two broad categories of diagrams and they are again divided into subcategories –

- Structural Diagrams
- Behavioral Diagrams

Structural Diagrams

The structural diagrams represent the static aspect of the system. These static aspects represent those parts of a diagram, which forms the main structure and are therefore stable.

These static parts are represented by classes, interfaces, objects, components, and nodes. The four structural diagrams are –

Class diagram

Object diagram

Component diagram

Deployment diagram

Class Diagram

Class diagrams are the most common diagrams used in UML. Class diagram consists of classes, interfaces, associations, and collaboration. Class diagrams basically represent the object-oriented view of a system, which is static in nature. Active class is used in a class diagram to represent the concurrency of the system.

Class diagram represents the object orientation of a system. Hence, it is generally used for development purpose. is is the most widely used diagram at the time of system construction.

Object Diagram

Object diagrams can be described as an instance of class diagram. Thus, these diagrams are closer to real-life scenarios where we implement a system. Object diagrams are a set of objects and their relationship is just like class diagrams. They also represent the static view of the system.

The usage of object diagrams is similar to class diagrams but they are used to build prototype of a system from a practical perspective.

Component Diagram

Component diagrams represent a set of components and their relationships. These components consist of classes, interfaces, or collaborations. Component diagrams represent the implementation view of a system. During the design phase, software artifacts (classes, interfaces, etc.) of a system are arranged in different groups depending upon their relationship. Now, these groups are known as components. Finally, it can be said component diagrams are used to visualize the implementation.

Deployment Diagram

Deployment diagrams are a set of nodes and their relationships. These nodes are physical entities where the components are deployed. Deployment diagrams are used for visualizing the deployment view of a system. This is generally used by the deployment team. Note – If the above descriptions and usages are observed carefully then it is very clear that all the diagrams have some relationship with one another. Component diagrams are dependent upon the classes, interfaces, etc. which are part of class/object diagram. Again, the deployment diagram is dependent upon the components, which are used to make component diagrams.

Behavioral Diagrams

Any system can have two aspects, static and dynamic. So, a model is considered as complete when both the aspects are fully covered.

Behavioral diagrams basically capture the dynamic aspect of a system. Dynamic aspect can be further described as the changing/moving parts of a system.

UML has the following five types of behavioral diagrams –

- Use case diagram

- Sequence diagram
- Collaboration diagram
- Statechart diagram
- Activity diagram

Use Case Diagram

Use case diagrams are a set of use cases, actors, and their relationships. They represent the use case view of a system. A use case represents a particular functionality of a system. Hence, use case diagram is used to describe the relationships among the functionalities and their internal/external controllers. These controllers are known as actors.

Sequence Diagram

A sequence diagram is an interaction diagram. From the name, it is clear that the diagram deals with some sequences, which are the sequence of messages flowing from one object to another.

Interaction among the components of a system is very important from implementation and execution perspective. Sequence diagram is used to visualize the sequence of calls in a system to perform a specific functionality.

Collaboration Diagram

Collaboration diagram is another form of interaction diagram. It represents the structural organization of a system and the messages sent/received. Structural organization consists of objects and links. The purpose of collaboration diagram is similar to sequence diagram. However, the specific purpose of collaboration diagram is to visualize the organization of objects and their interaction.

State chart Diagram

Any real-time system is expected to be reacted by some kind of internal/external events. These events are responsible for state change of the system. State chart diagram is used to represent the event driven state change of a system. It basically describes the state change of a class, interface, etc. State chart diagram is used to visualize the reaction of a system by internal/external factors.

Activity Diagram

Activity diagram describes the flow of control in a system. It consists of activities and links. The flow

can be sequential, concurrent, or branched. Activities are nothing but the functions of a system. Numbers of activity diagrams are prepared to capture the entire flow in a system. Activity diagrams are used to visualize the flow of controls in a system. This is prepared to have an idea of how the system will work when executed.

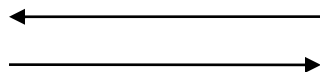
Note – Dynamic nature of a system is very difficult to capture. UML has provided features to capture the dynamics of a system from different angles. Sequence diagrams and collaboration diagrams are isomorphic hence they can be converted from one another without losing any information. This is also true for State chart and activity diagram

5.3 data flow diagrams

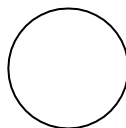
A graphical tool used to describe and analyze the movement of data through a system manual or automated including the process, stores of data, and delays in the system. Data Flow Diagrams are the central tool and the basis from which other components are developed. The transformation of data from input to output, through processes, may be described logically and independently of the physical components associated with the system. The DFD is also known as a data flow graph or a bubble chart.

DFDs are the model of the proposed system. They clearly should show the requirements on which the new system should be built. Later during design activity this is taken as the basis for drawing the system's structure charts. The Basic Notation used to create a DFD's are as follows:

1. Dataflow: Data move in a specific direction from an origin to a destination.

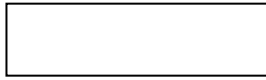


2. Process: People, procedures, or devices that use or produce (Transform) Data. The physical component is not identified.

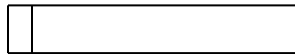


3. Source: External sources or destination of data, which may be People, programs, organizations

or other entities.



4. Data Store: Here data are stored or referenced by a process in the System.



6. Implementation

6.1 USER

The student who wants to seek admission is the user of the online system. The student registers himself/herself with the system and becomes authenticated user after the login process. The students can seek admission in 2nd, 3rd and 4th year of engineering college as per their academic year. The admission form is filled by the student after logging in the system and the documents such as previous years mark sheets, Aadhar card, caste certificates, etc. are scanned and uploaded in the system. The application forms for provisional admission, photocopy and revaluation can be filled the user in the similar fashion. All these details can be accessed by the admin via server. If the admin verifies the form and grants permission to the user for admission, a message will be generated and sent to the student ID through server. If an unauthenticated user tries to login in the system, he/she is not permitted to do so.

6.2 SERVER

The Server acts as a medium between the authenticated user (student) and the admin. All the details entered and documents uploaded by the student is stored in the server (Wamp Server). The admin can access these details via server. When the admin verifies the student details and grants permission for regular admission/provisional admission/ photocopy/ revaluation process, a message will be generated to the user sent by admin via server.

6.3 ADMIN

The admin plays a pivotal role in the online admission system. It is the admin who provides authentication to the user in the online process. The details entered and the documents uploaded by the authenticated user is accessed by the admin via server. These details are checked, verified and authenticated by the admin. If the details entered and documents uploaded are genuine, the admin approves the form and a message is sent to the user via server. The modification rights in the system to the user is granted by the admin. The admin acts as a supreme authority to approve, accept and reject the application forms for admission.

7. Technology Description

7.1 PYTHON

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Interactive Mode Programming

Invoking the interpreter without passing a script file as a parameter brings up the following prompt –

```
$ python
```

```
Python 2.4.3 (#1, Nov 11 2010, 13:34:43)
```

```
[GCC 4.1.2 20080704 (Red Hat 4.1.2-48)] on linux2
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>>
```

Type the following text at the Python prompt and press the Enter –

```
>>> print "Hello, Python!"
```

If you are running new version of Python, then you would need to use print statement with parenthesis as in `print ("Hello, Python!")`;. However in Python version 2.4.3, this produces the following result –

Hello, Python!

Script Mode Programming

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension `.py`. Type the following source code in a `test.py` file –

Live Demo

```
print "Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows –

```
$ python test.py
```

This produces the following result –

Hello, Python!

Let us try another way to execute a Python script. Here is the modified `test.py` file –

Live Demo

```
#!/usr/bin/python
```

```
print "Hello, Python!"
```

We assume that you have Python interpreter available in /usr/bin directory. Now, try to run this program as follows –

```
$ chmod +x test.py    # This is to make file executable
```

```
$ ./test.py
```

This produces the following result –

```
Hello, Python!
```

8.Coding

```

1 import firebase as firebase
2 from flask import *
3 import os
4 import pyrebase
5
6 config = {
7     "apiKey": "AIzaSyAIs5H0uNkfoGulr75eQK2t6jNVT5T1t8",
8     "authDomain": "cams-da440.firebaseio.com",
9     "databaseURL": "https://cams-da440.firebaseio.com",
10    "projectId": "cams-da440",
11    "storageBucket": "cams-da440.appspot.com",
12    "messagingSenderId": "592415369968"
13 }
14
15 firebase = pyrebase.initialize_app(config)
16 db = firebase.database()
17
18 app = Flask(__name__)
19
20 global active, credentials
21 active = None
22
23 @firebase
24 from firebase import firebase
25 firebase = firebase.FirebaseApplication('https://cams-da440.firebaseio.com/', None)
26 credentials = firebase.get('/credentials', None)
27 cutoff = firebase.get('/cutoff', None)
28 applications = firebase.get('/application', None)
29
30 @app.route('/', methods=['POST', 'GET'])
31 def home():

```

```

102 @app.route('/removeStud/<string:d_id>', methods=['POST', 'GET'])
103 def removeStud(d_id):
104     #print(d_id)
105     db.child("application").child(d_id).update({'Status': 'Rejected'})
106     applications = firebase.get('/application', None)
107     return render_template('home_admin.html', u = active, cutoff = cutoff, form = applications)
108
109 @app.route('/addStud/<string:d_id>', methods=['POST', 'GET'])
110 def addStud(d_id):
111     #print(d_id)
112     db.child("application").child(d_id).update({'Status': 'Accepted'})
113     applications = firebase.get('/application', None)
114     return render_template('home_admin.html', u = active, cutoff = cutoff, form = applications)
115
116 @app.route('/application', methods=['POST', 'GET'])
117 def application():
118     global active
119     print(active)
120     if active != None:
121         for i in credentials:
122             if active in credentials[i]['EmailId']:
123                 idno = i
124                 break
125         applications = firebase.get('/application', None)
126         if applications != None:
127             for i in applications:
128                 if i == idno:
129                     return render_template('congrats.html')
130     if request.method == 'POST':

```

9. System Testing

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

9.1 TYPES OF TESTS

Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing:

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

9.2 Test strategy and approach:

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered

10. Output Screens (FORMS & REPORTS)

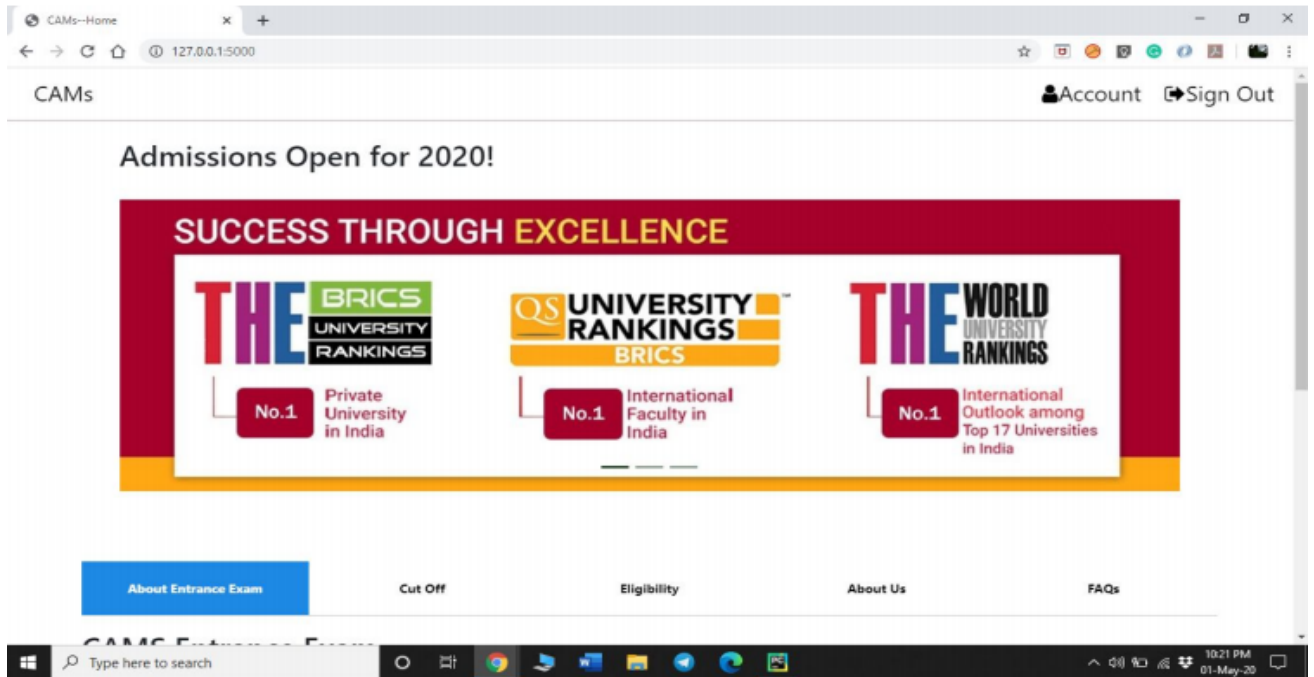


Fig: 10.1 Home Page

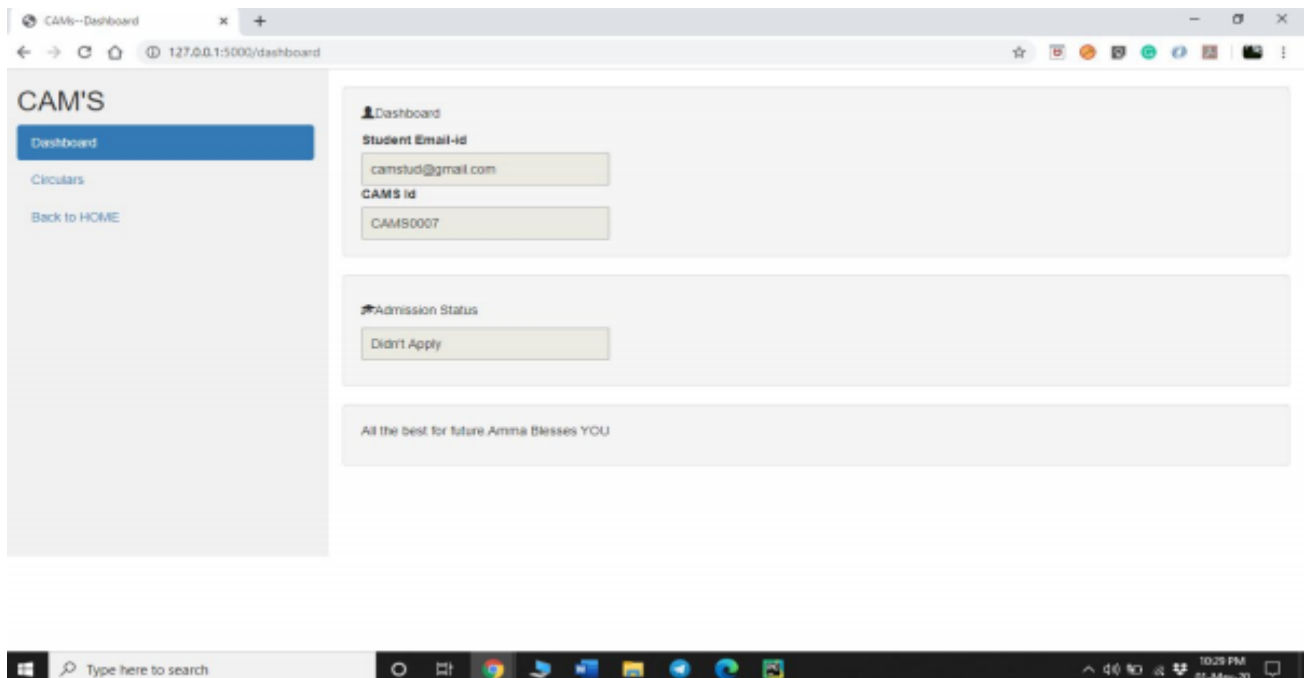


Fig 10.2 Dashboard

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/login'. The page title is 'CAMs' and the user is logged in as 'admin@gmail.com :)' with a 'Sign Out' link. The navigation menu includes 'Add College Details', 'View & Register Students', 'Create CutOff List', and 'View Final Selected Students'. The main heading is 'FILL IN THE FORM TO ADD DETAILS OF THE COLLEGE TO THE DATABASE.' The form contains three sections: 'Add Course' with a dropdown menu set to '--Choose a Branch--', 'Add Fee Structure' with a text input field 'Enter the fee', and 'Add Seats available' with a text input field 'Enter the seats available'. Below the form, a red message states 'By creating additions to the details, changes will be visible to the student's portal.' There are two buttons: a grey 'Confirm' button and a green 'Add Details' button.

10.3 Add course Page

The screenshot shows the same web browser window as before, but the main heading is 'VIEW/ACCEPT/REMOVE STUDENTS WHO APPLIED FOR ADMISSION'. Below this heading is a table with the following data:

CAMS ID	Email	Stream	Marks scored	Cut-Off for the Stream	Accept/Remove
CAMS0007	camstud@gmail.com	Electrical and Electronic Engineering	88	860	Accept Remove

Fig 10.4 List of Application Page

11. Future Enhancement

The online admission system prototype which was developed is effective in a way that it will automate and make online admission instead of manual. Yet some areas in this study not explored in detail due to time constraint. Some of them are given below in future the possibilities of work to be carried out in this system

1. The payment system can be increased by using the original merchant account for credit card holders.
2. The admission offer letter can be sent online when student is admitted to the university.
3. Student can be assigned academic coordinator to advise them after admission to avoid problems of students facing for coming university and choosing subjects.
4. The admission system security in login can be increased by using CAPTCHA.
5. The System can send the confirmation or information by SMS to cell phone of candidates

12. Conclusion

By this project we are making to heighten and implement the improvements in college counselling system, which is good and an easy way for reduce in hand work and making less handwork necessary. Students' database can be retrieved within quick interval of time proper maintaining of records can be achieved. Our system presently aims on creation of a good management system for the college universities. System will solve the problem of seat status by showing dynamic display of the seat status after every session of counselling, so that student can select their seat preference accordingly in a hassle freeway which will make a clear impact for selecting their desired branch in an quick interval of time. This online counselling management system will do the best for the needful who are at very large distance and will clear information about the process so that counselling can be implemented in a smooth way and by reducing the efforts and by increasing the efficiency in the college work process.

13. References

- [1] HuixiaWang,HuiZan “methodologies of student information management system.” 2010 report on college information processing.978-0-7695- 4196-9/10 IEEE.
- [2] Zhi-gang YUE, You JIN, “The development and design of the student management system based on the network strategies”,2010 International Conference on management system , 978-0- 7695- 4136-5/10 2010 IEEE.
- [3] N. F. Col, (2007). Using Internet technology to improve and simplify counseling about university: The WISDOM website. Maturitas , 57, 95-99.
- [4] M.A. Norasiah and A. Norhayati. “Intelligent student information system”.4th International conference on computer technology proceduress, Shah Alam, singapore, 0-7803-7773-7/03 2003 IEEE.
- [5] Qiu Chang-li 2 Li Jing 3. “The developing procedures of student information counselling system based on B/S architecture”. 978-1- 4577- 1415-3/12 2012IEEE
- [6] S.R.Bharamagoudar¹, Geeta R.B.², S.G.Totad³ “Student online counselling system International Journal of Advanced Research in Computer Engineering Vol. 2, Issue 6, June 2013.
- [7] B. Stein, (2000). Learning information Management System (LIMS) Solution for WebBased counselling system (WBCS). Abgerufen am 19. Jan. 2008 von White Paper: