PES

UNIVERSITY

CELEBRATING 50 YEARS

Step-Up

Prepared By: Sparsh Pal

SRN: PES1PG21CA086

Guide: Dr. Thenmozhi S

Assistant Professor

Signature:

Date:

CONTENTS

1. INTRODUCTION
1.1 PROBLEM DEFINITION
1.2 PROPOSED SOLUTION
1.3 PURPOSE
1.4 SCOPE

2. LITERATURE SURVEY
2.1 EXISTING SYSTEM
2.2 FEASIBILITY STUDY

3. HARDWARE AND SOFTWARE REQUIREMENTS
3.1 HARDWARE REQUIREMENTS
3.2 SOFTWARE REQUIREMENTS

4. SOFTWARE REQUIREMENTS SPECIFICATION
4.1 USERS
4.2 FUNCTIONAL REQUIREMENTS
4.3 NON-FUNCTIONAL REQUIREMENTS

5. SYSTEM DESIGN
5.1 PROCESS FLOW DIAGRAM
5.2 CONTEXTFLOW DIAGRAM
5.3 CLASS DIAGRAM

6. DETAILED DESIGN
6.1 USE CASE DIAGRAM
6.2 ACTIVITY DIAGRAM

7. IMPLEMENTATION
7.1 Screen Shots

ABSTRACT:

Today all the work at the time of admission of the students is done manually by ink and paper, which is very slow and consuming much efforts and time. It is required to Design of a Computerized Automated Student Admission System, to speed up and make it easy to use system. Student admissions are a vital part of any university's running because students are what keep a University alive.

The student admission is one of the most important activities within a university as one cannot survive without students. A poor admissions system can mean fewer students being admitted into a university because of mistakes or an overly slow response time. The process begins with a potential student completing an application form through the Universities and Colleges Admissions Service, the first step for students is to apply directly to the university through a custom online form.

This project's aim is to automate the system, pre checking the inclusion of all required material and automatically ranking each student's application based on a number of criteria. These criteria include the ranking of their university, their grade at said university and their language grade Certificate.

The data used by the system is stored in a database that will be the center of all information held about students and the base for the remainder of the process after the initial application has been made.

This enables things to be simplified and considerably quickened, making the jobs of the people involved easier. It supports the current process but centralizes it and makes it possible for decisions to be made earlier and easier way

1. INTRODUCTION

Step-Up is a web enabled application that helps educational institutions manage the student enrollment process effortlessly. It lets admission teams capture student inquiries, check their eligibility, follow-up, collect documents, and complete the application process digitally. The admission management system also allows students to apply online, check their application status, submit documents and pay fees online.

Needless to say that the admission cycle is long and cumbersome. From student lead generation, document collection, selection, and admissions – everything involves a lot of paperwork. Digital solutions for admission management.

An efficient student management system makes the enrollment process easier for the students and counselors. With paperless applicant portals, you can make the admission journey easy for students because a student-friendly form will always have a higher completion rate. With the help of a student portal, applicants can:

- Fill the admission form at their convenience
- Select their course
- Schedule screening exams and interviews
- Pay the application fee
- Book virtual, physical campus tours, and continuously engage with counselors with the help of a dashboard.

With the help of Step-Up, you can connect all processes and teams. You can build an admission workflow and map student journeys, for example, inquiry, applications, counseling, enrollment, etc., with the relevant department.

- **PROBLEM DEFINITION**:

Now-a-days the admission process for institutes comprises of several stages like conducting exam, quizzes and interviews all these process consumes much effort and time both by institute and students. Also, there are few candidates who just applies for the college and have no intention to get enroll.

1.2 PROPOSED SOLUTION:

Step-Up primarily focuses on admission of new student, generate the fee payment receipts, scanning and uploading the students documents, saving it with permanent unique id provided, providing the details regarding institute, courses and conducting all the admission process in an efficient and concise manner.

Step-Up collects the details of student and accordingly allot them for quick-tests afterwards student can upload all the required documents which will be verified by the faculties and student can go for the interview round.

This web enabled application provides a layer between the aspiring candidates and the institute for shortlisting really interested candidates. This will make admission process concise.

- **PURPOSE:**

Every year, students in thousands stand in queues for collecting admission / application forms and then again for submitting the admission / application forms. This leads to problems in - managing the applications, handle queries, & distribution of forms, collection of forms and then short-listing the candidates resulting in annoyed parents and students alike. In addition to this, the admission process is not transparent, leading to possibility for widespread malpractice.

The goal of **'Step-Up'** is to computerize the admission management system structure, its related operation, and functionality to rectify these weaknesses. Also, the purpose is to provide support to the administration and admission seeking candidates by providing a faster, transparent, and easy way of maintaining records and utilize them for reference and further proceedings.

1.4 SCOPE:

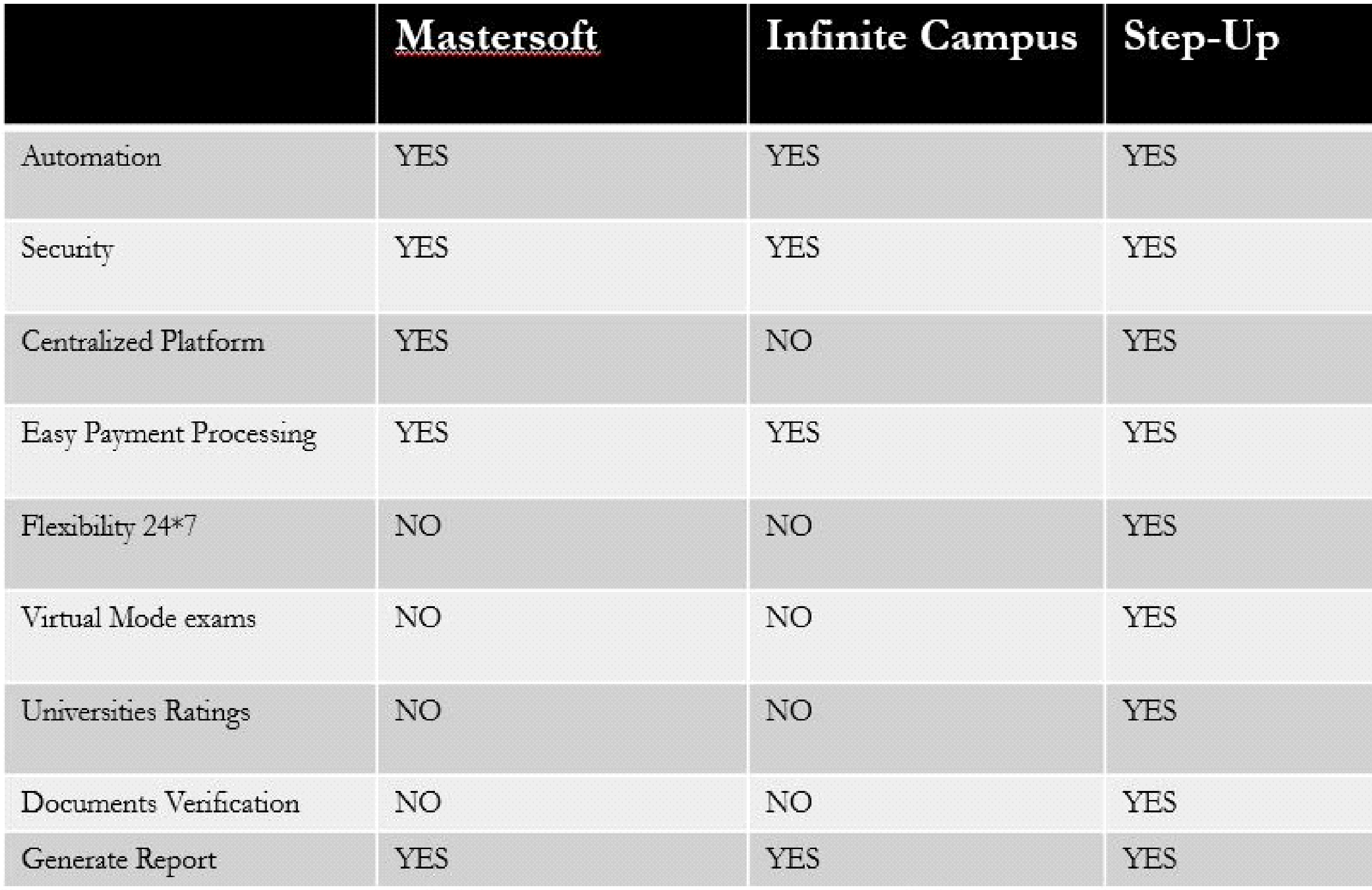
Scope of Step-Up is to ready portal that should be demonstrated to the Admission Committee of the college before final selection of the service provider. After the section of the service provider selected party must present a trial run to the Admission Committee before the commencement of receiving online applications.

An applicant should be able to fill up the form directly without any login. But a user id and password should be provided to applicant before payment of online application fee. Form structure should consist of all the data needed for registration (old form may be used as reference).

Step-Up aim is to automate the system, pre-checking the inclusion of all required material and automatically ranking each student's application based on a number of criteria. It also supports the current process but centralizes it and makes it possible for decisions to be made earlier and easier way.

LITERATURE SURVEY:

EXISTING SYSTEMS:



	Mastersoft	Infinite Campus	Step-Up
Automation	YES	YES	YES
Security	YES	YES	YES
Centralized Platform	YES	NO	YES
Easy Payment Processing	YES	YES	YES
Flexibility 24*7	NO	NO	YES
Virtual Mode exams	NO	NO	YES
Universities Ratings	NO	NO	YES
Documents Verification	NO	NO	YES
Generate Report	YES	YES	YES

FEASIBILITY STUDY:

An important outcome of the preliminary investigation is to determine that whether the system requested is feasible. Depending on the results of the preliminary investigation the survey is expended to a more detailed feasibility study. Feasibility study is a test of a system proposal according to its workability, impact on the organization, ability to meet user needs, and effective use of resources. The objective of feasibility study is not to solve the problem but to acquire a sense of its scope. During the study, the problem definition is crystallized and aspects of the problem to be included in the system are determined. Consequently, costs and benefits are estimated with greater accuracy at this stage.

Objectives of Feasibility Study

The main objectives of feasibility study are:

- To identify the deficiencies in the current system.
- To determine objectives of the proposed system.
- To acquire a sense of scope of the system.
- To identify the responsible users.
- To determine whether it is feasible to develop the new system.

3. HARDWARE AND SOFTWARE REQUIREMENTS:

3.1 HARDWARE REQUIREMENTS:

Processor: Intel I3 IV 2.4 GHz and above.

RAM: 4 Gb RAM or above

Hard Disk: 50 Gb or above

3.2 SOFTWARE REQUIREMENTS:

Database: (MongoDB 5.0)

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL) which is deemed non-free by several distributions.

FRONT END: (ReactJS 17.0.2)

React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies. React can be used as a base in the development of single-page, mobile, or server-rendered applications with frameworks like Next.js. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality

BACKEND : (NodeJS 15v, ExpressJS 4.17.2)

Node.js is an open-source server environment. Node.js is cross-platform and runs on Windows, Linux, Unix, and macOS. Node.js is a back-end JavaScript runtime environment. Node.js runs on the V8 JavaScript Engine and executes JavaScript code outside a web browser.

Node.js lets developers use JavaScript to write command line tools and for server-side scripting. The functionality of running scripts server-side produces dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aim to optimize throughput and scalability in web applications with many input/output operations, as well as for real-time Web applications (e.g., real-time communication programs and browser games)

The Node.js distributed development project was previously governed by the Node.js Foundation, and has now merged with the JS Foundation to form the OpenJS Foundation. OpenJS Foundation is facilitated by the Linux Foundation's Collaborative Projects program.

4. SOFTWARE REQUIREMENTS SPECIFICATION:

4.1 USER:

Admin: Admin will be having secure login and Admin is the person who can select/reject the candidates. He is the one who can update the details of the candidate, who had applied. Admin will shortlist the students for quiz/interviews for final selection and declare the results.

Candidate: Candidate at first, he will register for the portal and then he will be having a secure login. After logging in he will be able to enter his/her details and upload the required documents.

4.2 FUNCTIONAL REQUIREMENTS:

Sign-Up :

User must sign-up by clicking on Sign-Up button, user must fill the information form and click on Submit button and system will verify the information.

Login :

User must key in the username and password and press Sign In button to login. User can press Forget User-pass button. System will verify the login information

Apply Online :

User can apply online by click on apply online button. User must fill the information form. User can return back to main page by click on Home button.

View Application Status :

User can view application status by click on view button. User can return back to main page by click on Home button.

Update Application Status :

Administrator can update application status by click on update button. The system will be send notification about updating to the user email. Administrator can return back to main page by click on Home button.

Update Personal Information:

User must press Update Information button. User can update the information form and press Submit button.

Manage Users:

Administrator can add user account. Administrator can remove user account. Administrator can update user information

4.3 Non-Functional Requirements:

Usability: The system must be simple and easy to understand.

Reliability: The system should be trustworthy and reliable in

providing all the functionalities.

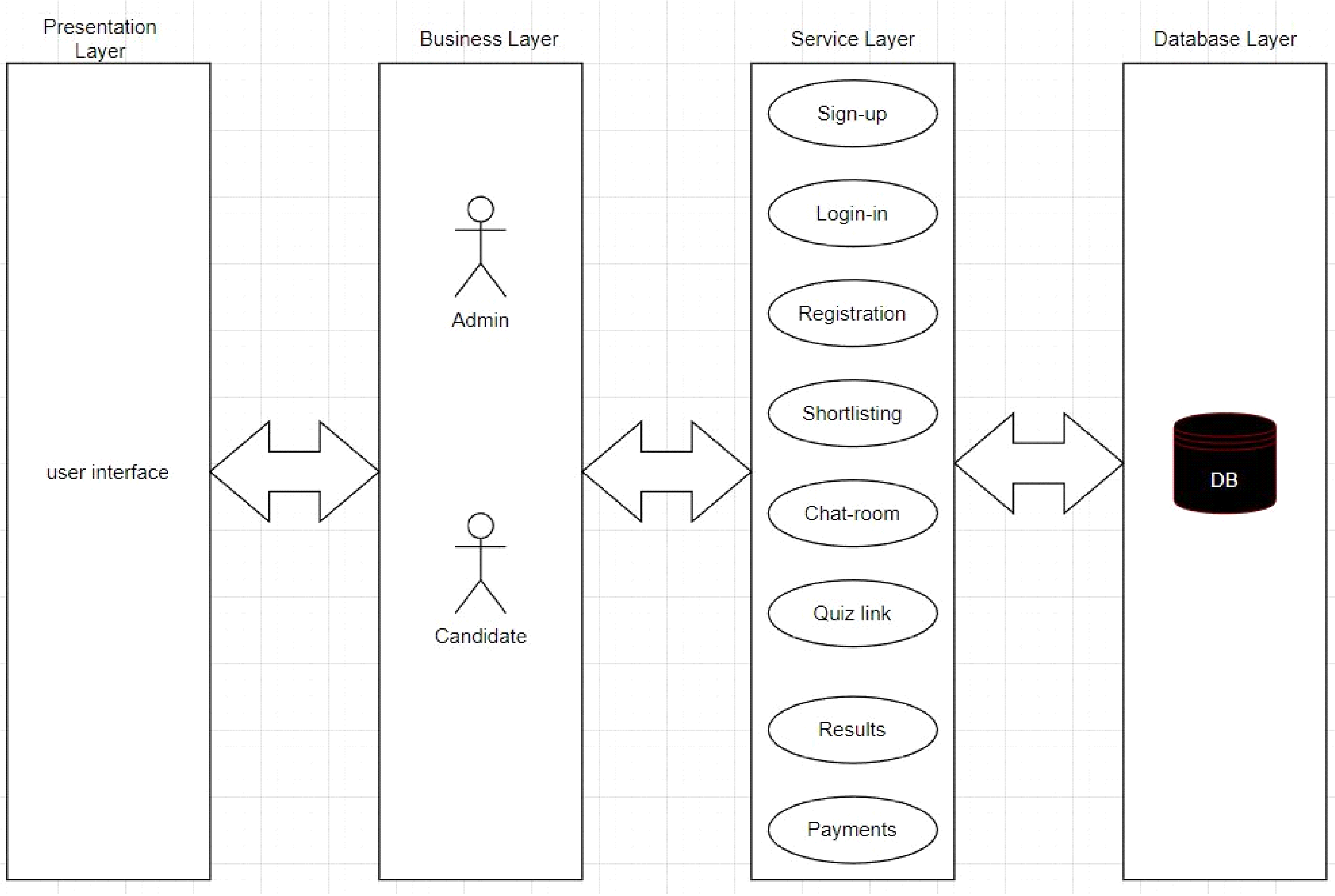
Security: All the users have separate login credentials and no one

alter others details.

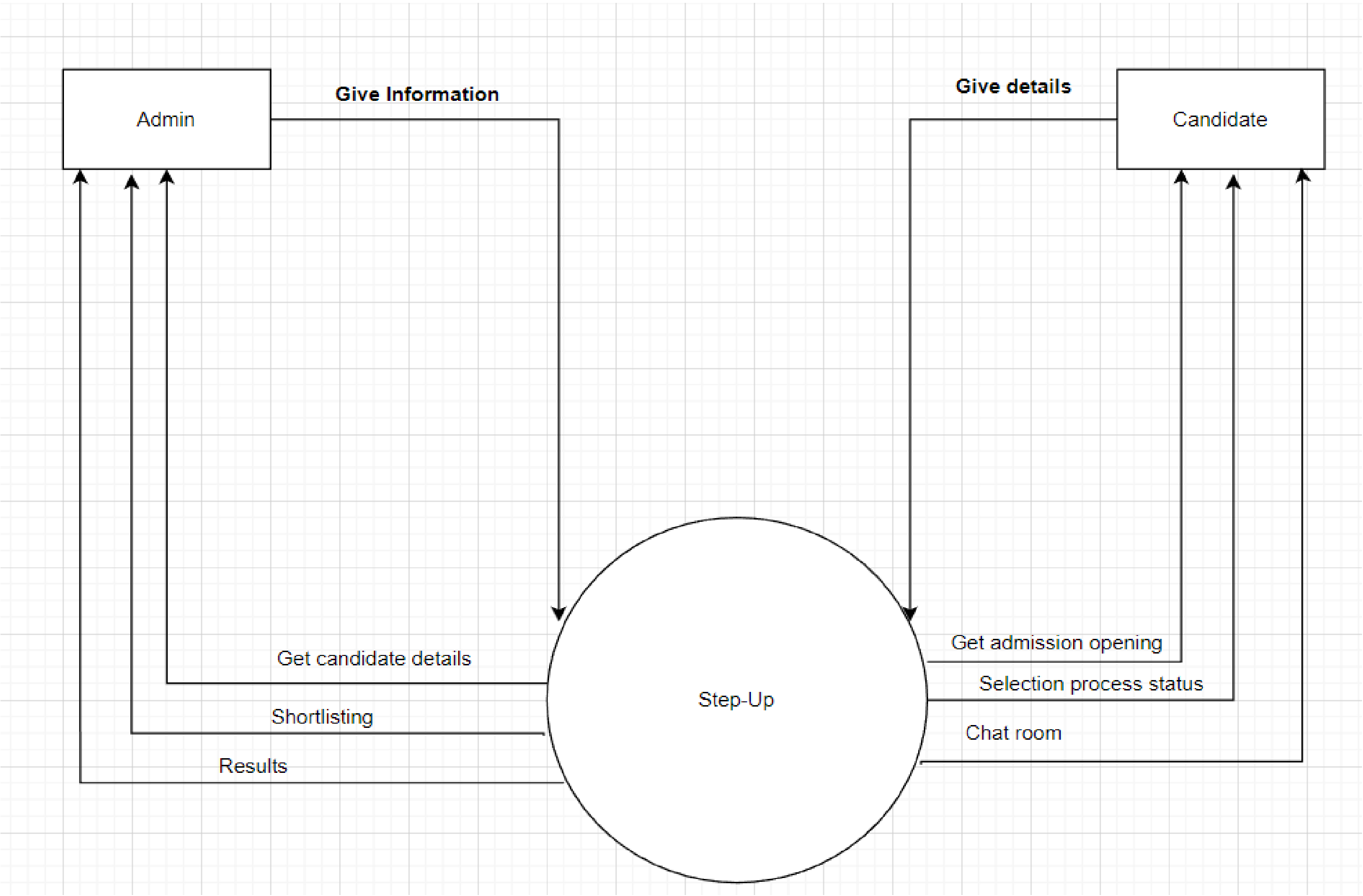
Flexibility: The system should be flexible enough to allow modifications at any point of time.

5. SYSTEM DESIGN

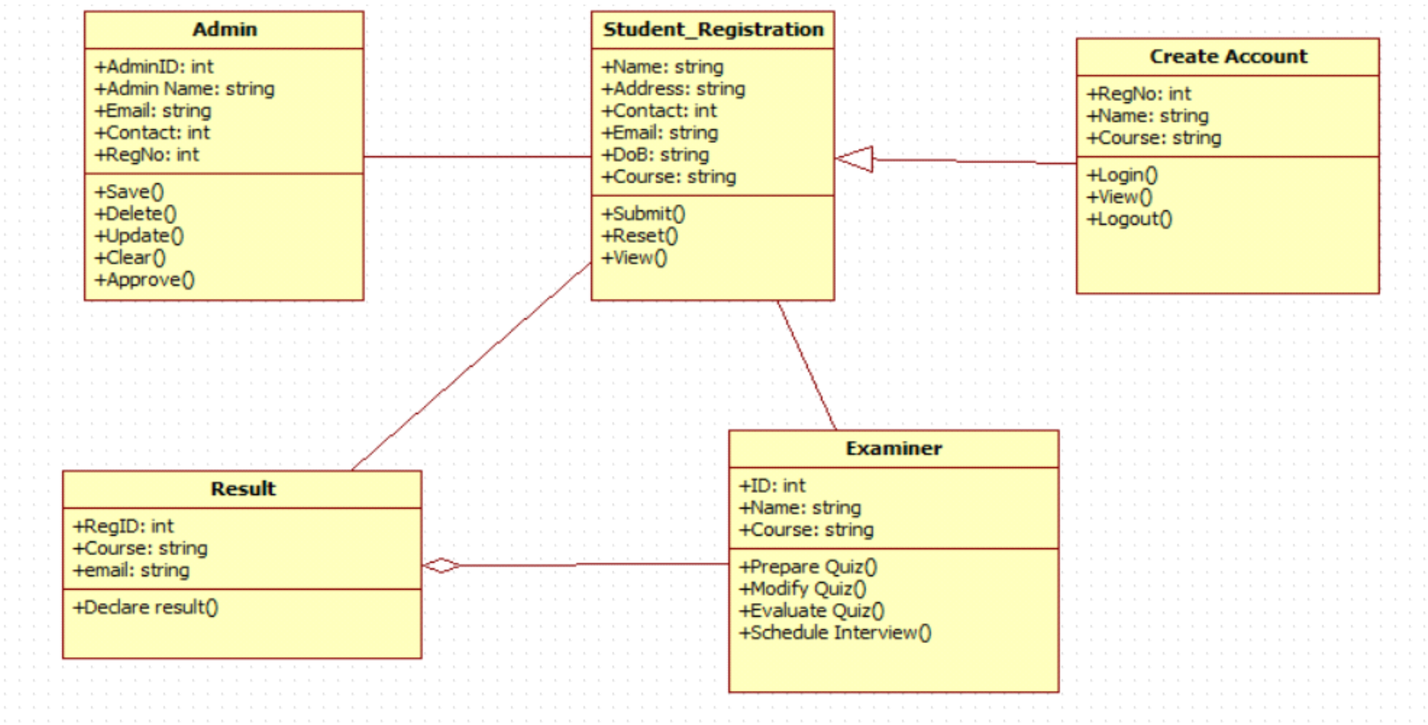
5.1 Architecture Diagram:



5.2 CONTEXTFLOW DIAGRAM:

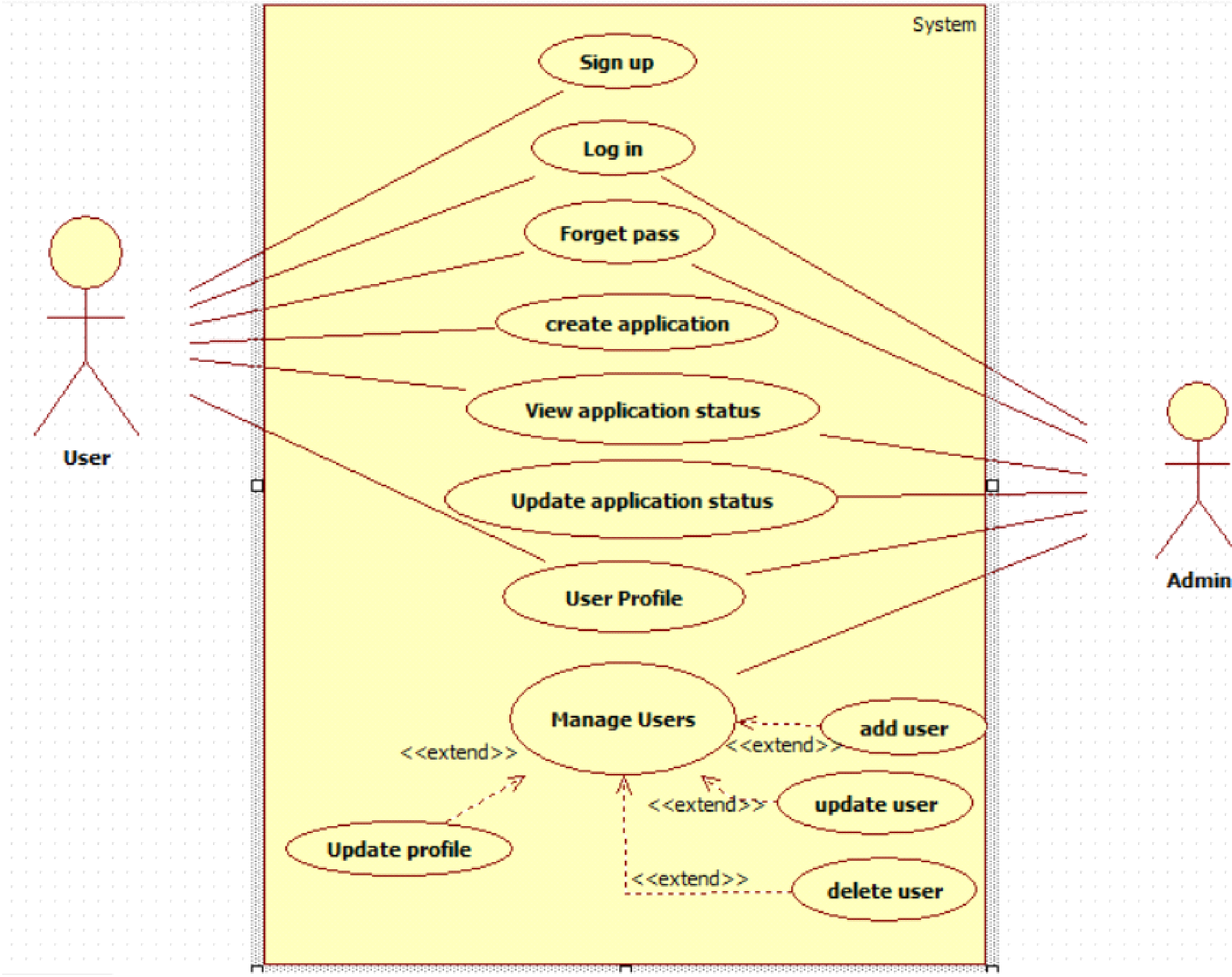

Admin
Give Information
Give details
Candidate
Get candidate details
Shortlisting
Results
Step-Up
Get admission opening
Selection process status
Chat room

5.3 CLASS DIAGRAM:



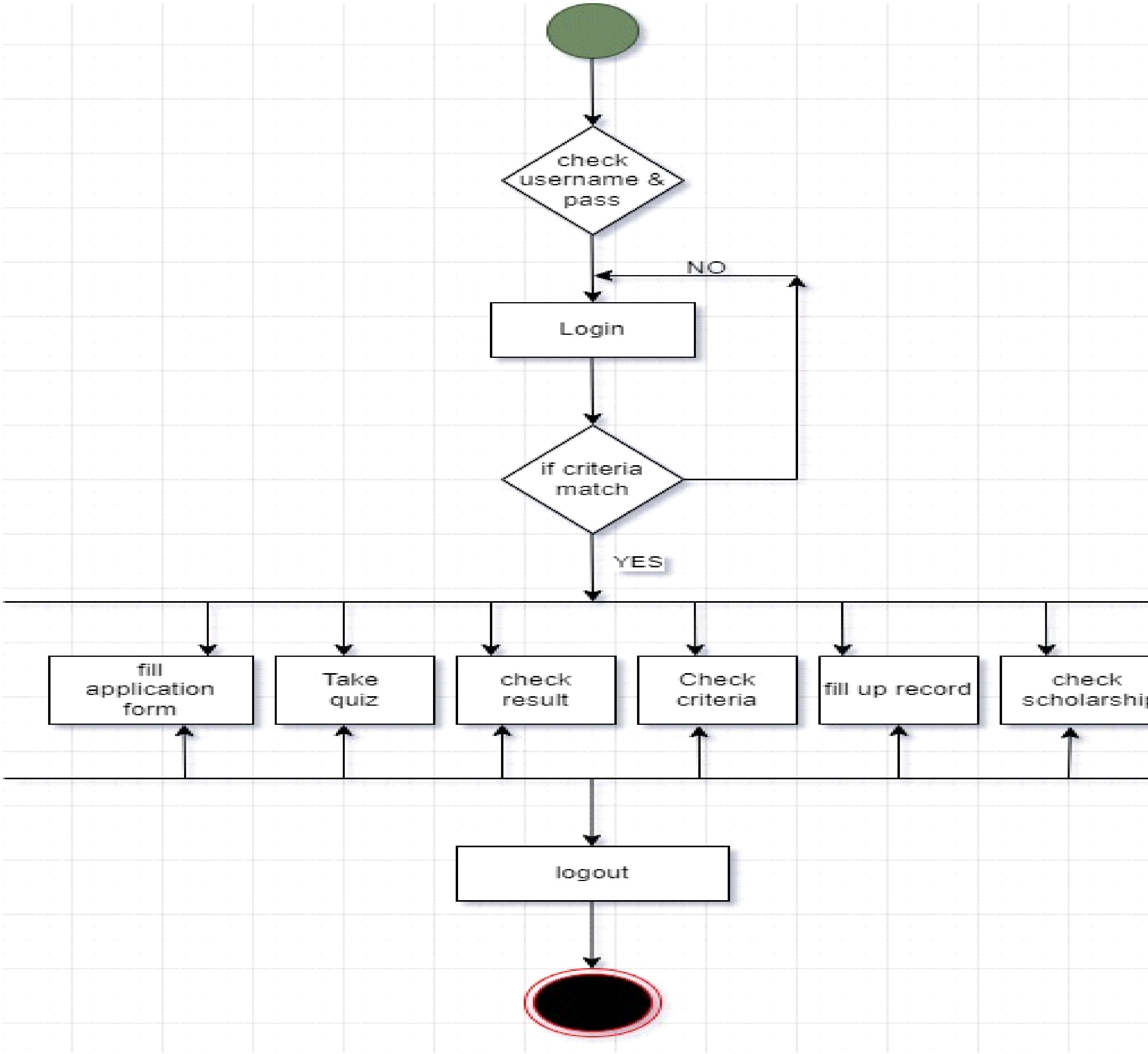
6. DETAILED DESIGN

6.1 USE CASE DIAGRAM:

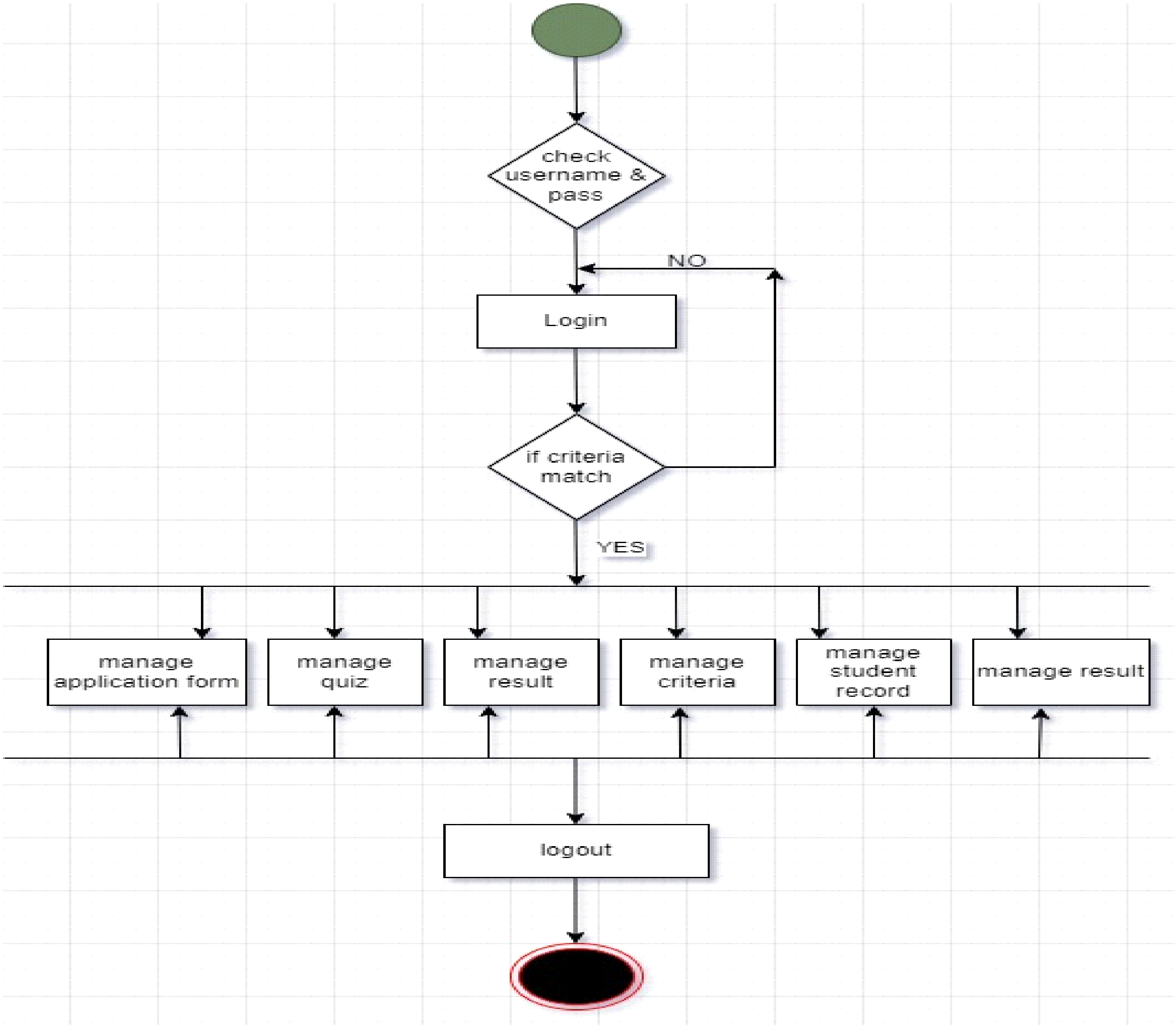

System
Sign up
Log in
Forget pass
create application
View application status
Update application status
User Profile
Manage Users
add user
update user
delete user
Update profile
<<extend>>
<<extend>>
<<extend>>
<<extend>>
User
Admin

6.2 ACTIVITY DIAGRAM:

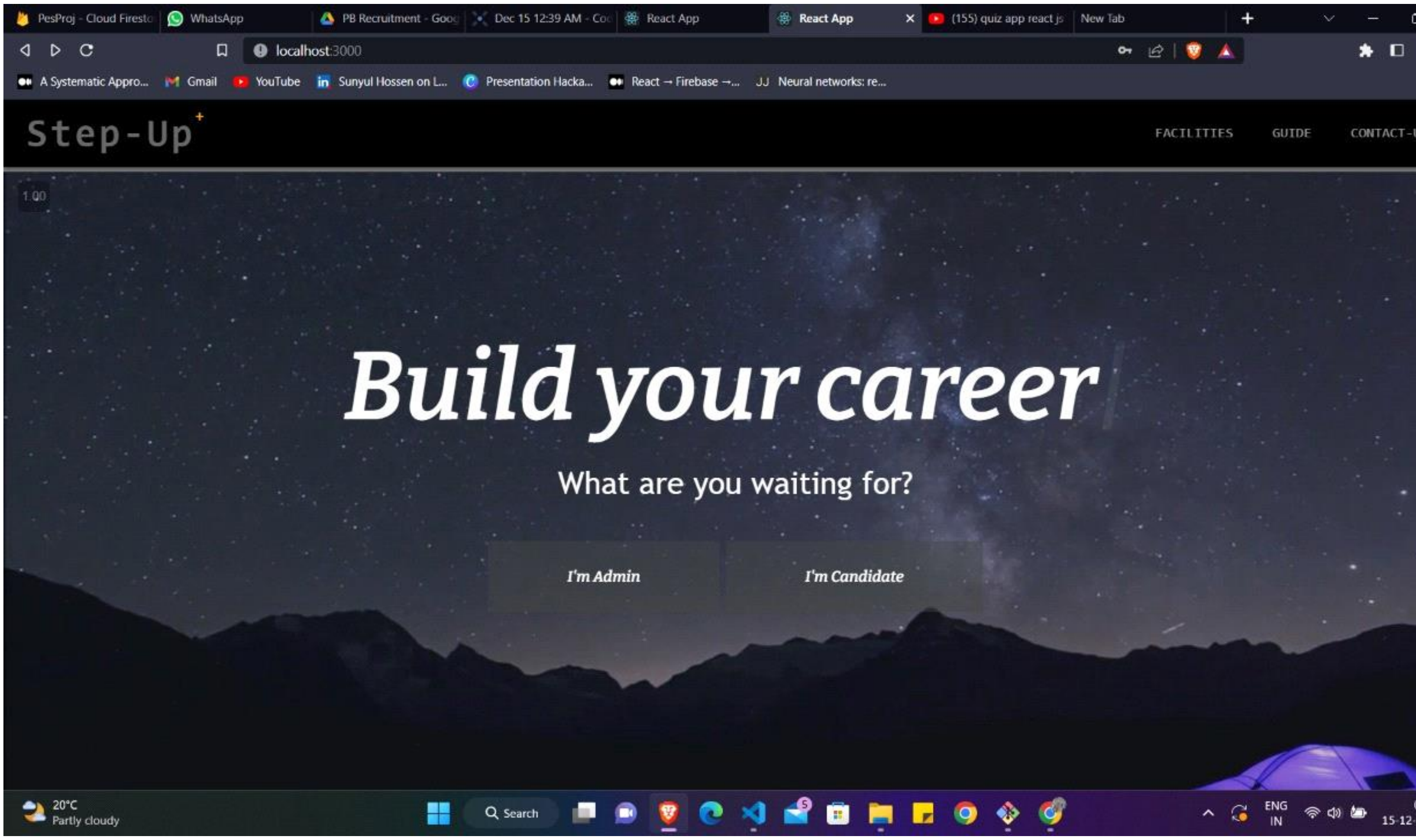
User side :


check username & pass
NO
Login
if criteria match
YES
fill application form
Take quiz
check result
Check criteria
fill up record
check scholarshi
logout

Admin side :

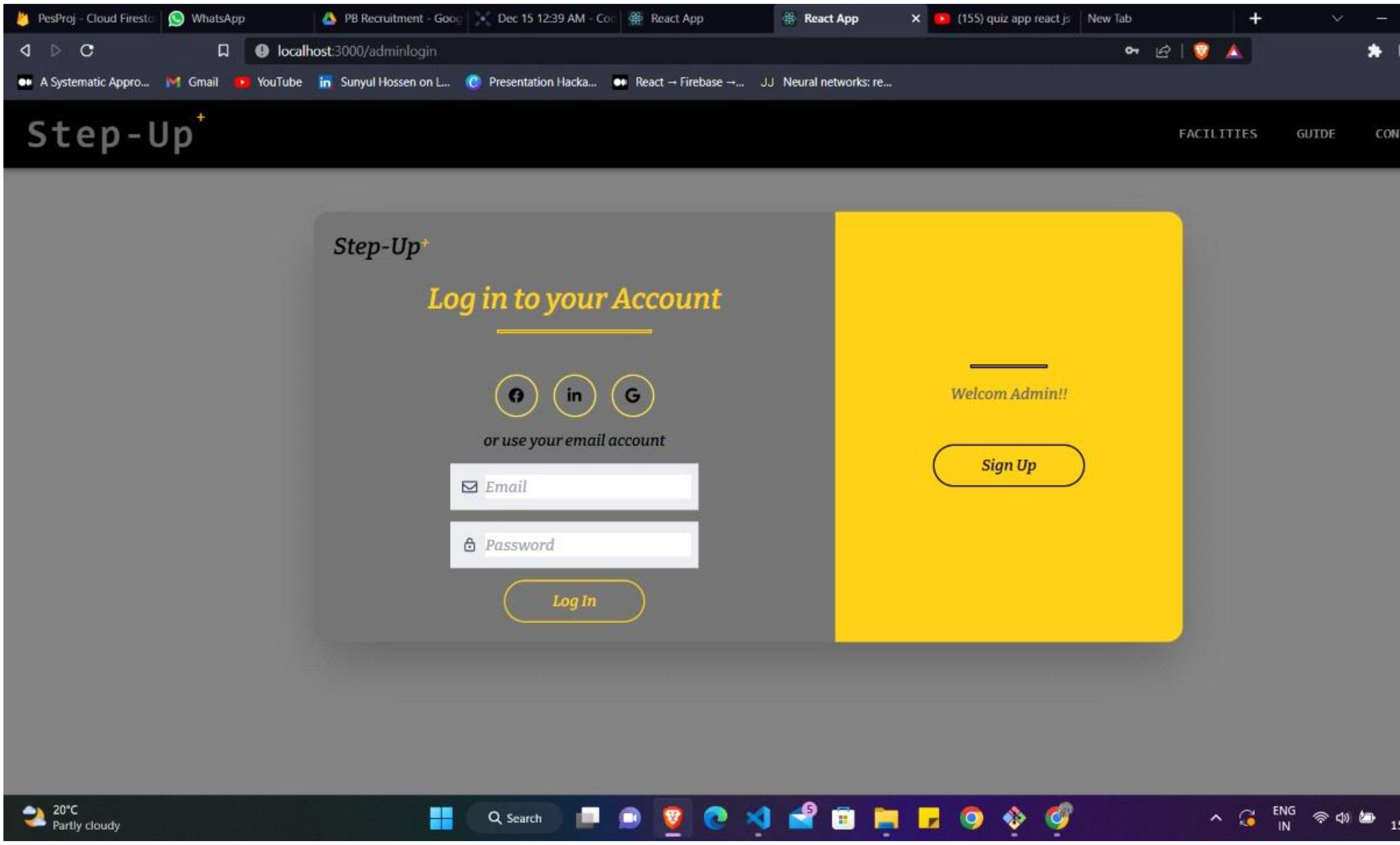

check username & pass
NO
Login
if criteria match
YES
manage application form
manage quiz
manage result
manage criteria
manage student record
manage result
logout

7.2 SCREEN SHOTS:



Build your career

What are you waiting for?



Step-Up+

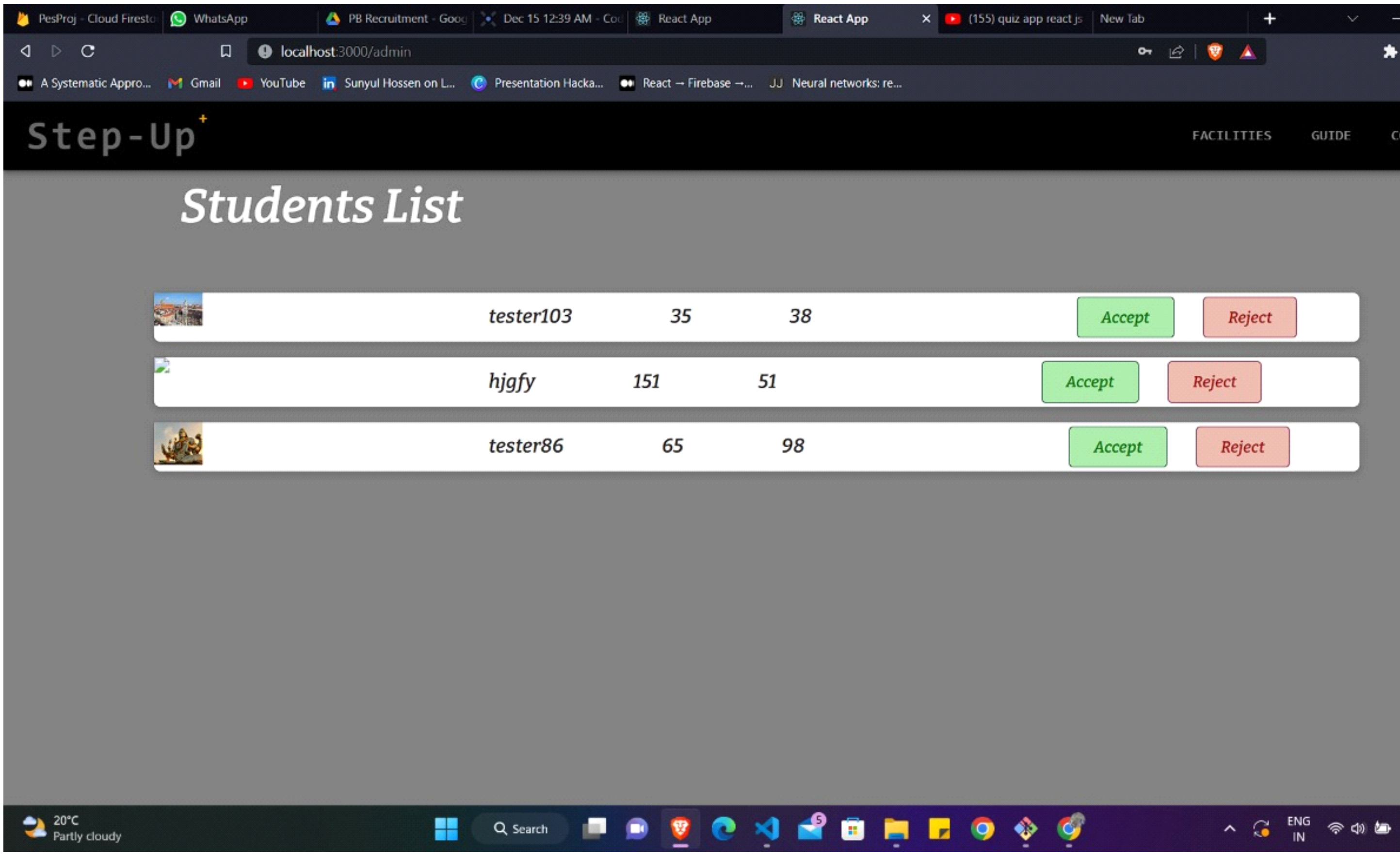
Log in to your Account

or use your email account

Email

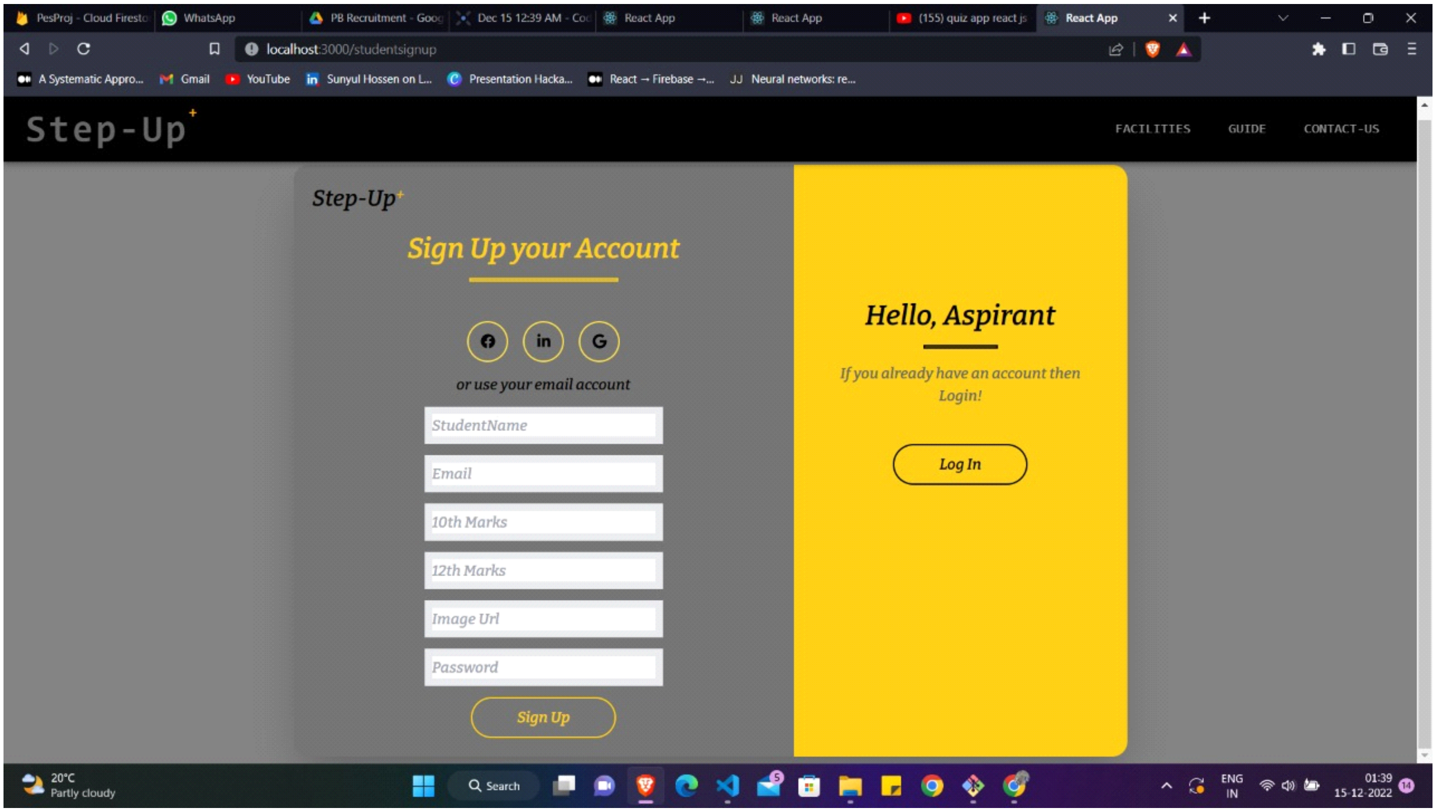
Password

Welcom Admin!!



Students List

	tester103	35	38	Accept	Reject
	hjgfy	151	51	Accept	Reject
	tester86	65	98	Accept	Reject



Step-Up+

Sign Up your Account

or use your email account

StudentName

Email

10th Marks

12th Marks

Image Url

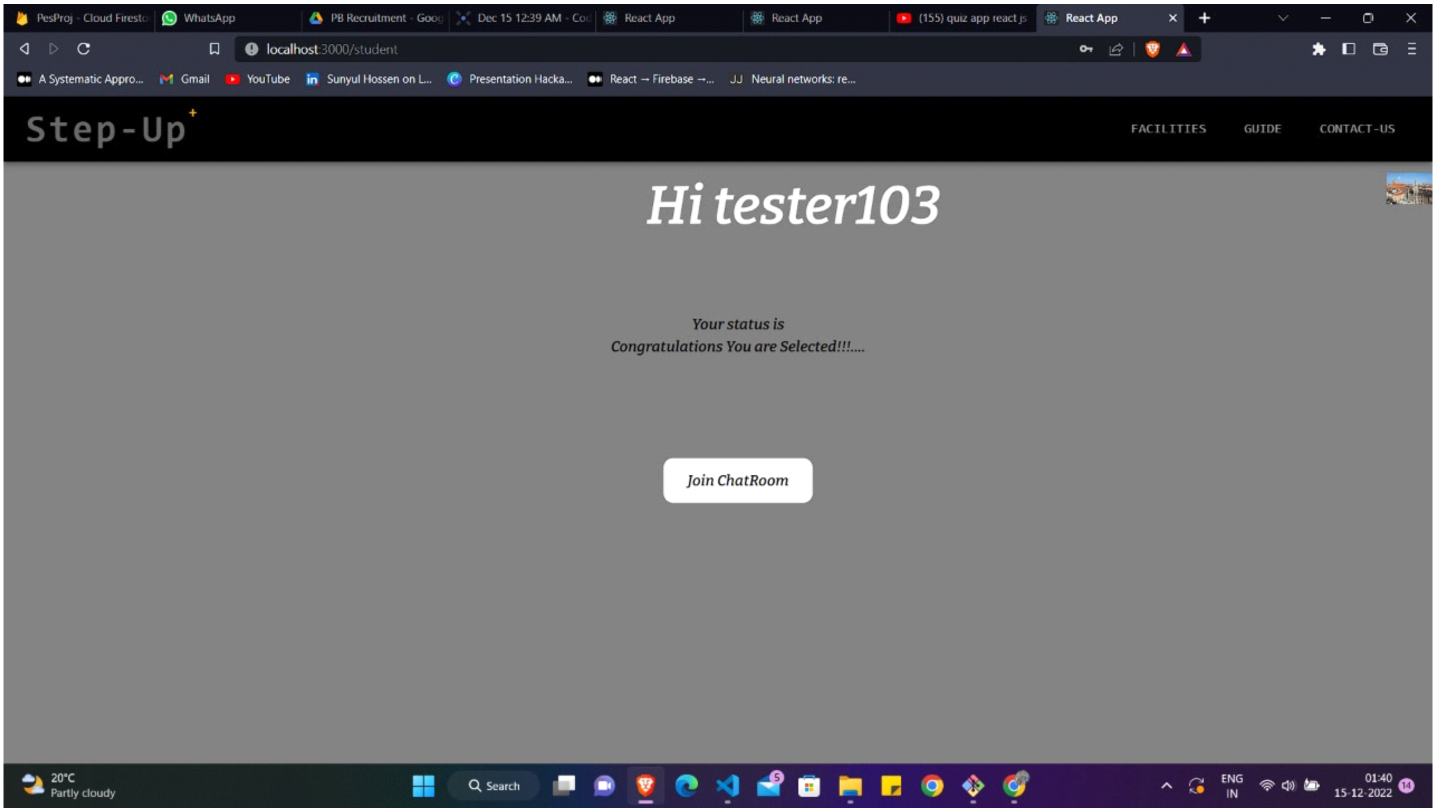
Password

Sign Up

Hello, Aspirant

If you already have an account then Login!

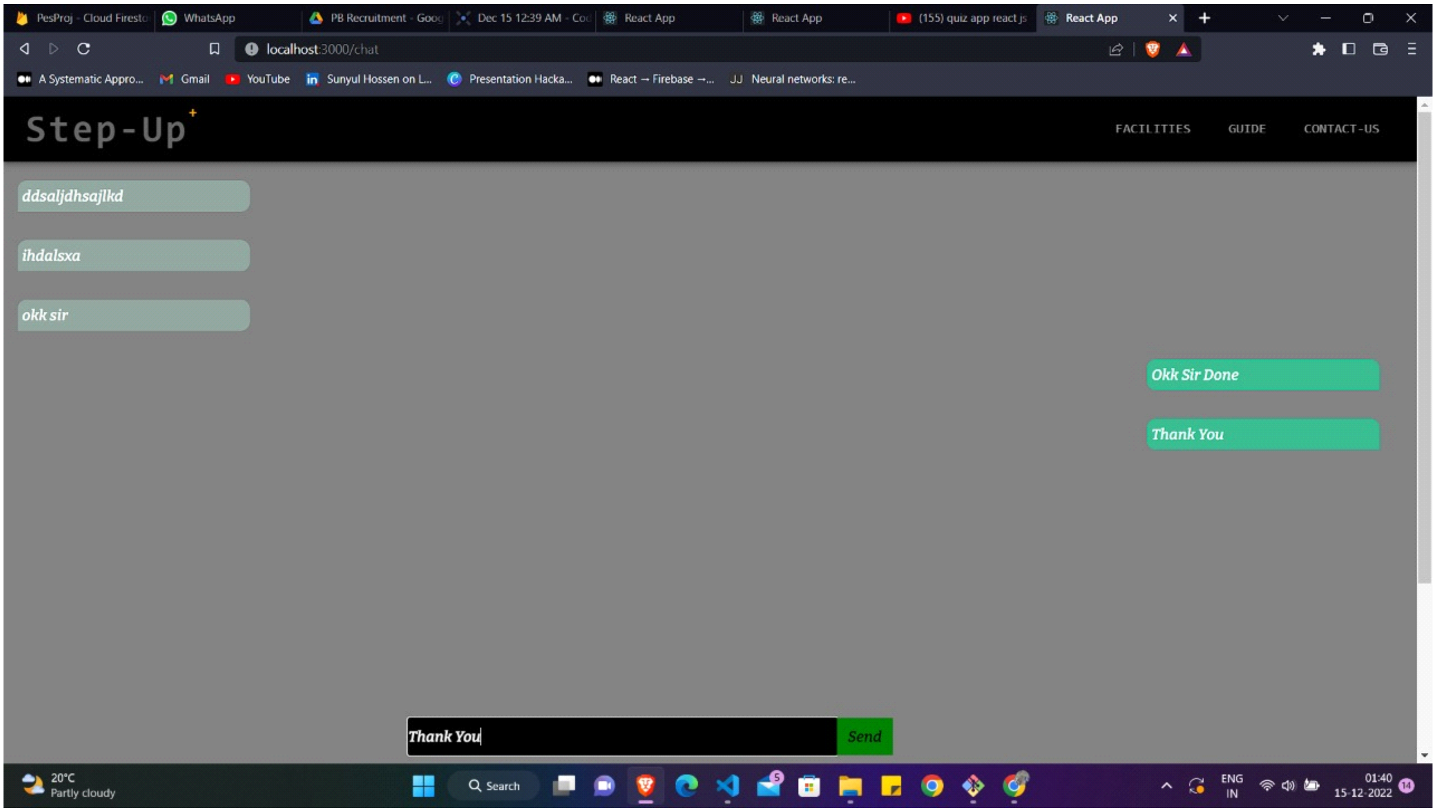
Log In



Hi tester103

Your status is
Congratulations You are Selected!!!....

Join ChatRoom



ddsaljdhsajlkd

ihdalsxa

okk sir

Okk Sir Done

Thank You

Thank You