**THE STATE UNIVERSITY OF ZANZIBAR**

**SCHOOL OF COMPUTING, COMMUNICATION AND MEDIA STUDIES**

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

##### FINAL YEAR PROJECT SYSTEM DOCUMENTATION

##### 2022/2023

##### PROJECT TITLE : NDOTO AJIRA

##### STUDENT NAME : ABDULLA MAKAME ABDALLA

##### REG NO : BITAM/8/20/075/TZ

##### SUPERVISER NAME : MR. AKRAM ALI OMAR

##### YEAR OF STUDY : 2022/2023

##### SUBMISSION DATE : 4/8/2023

# **DECLARATION**

I declare to The State University of Zanzibar (SUZA) that it is my own work and I have done myself, and that it has neither been submitted nor being submitted in any other institute or university.

Abdulla Makame Abdalla. Date

………………………… ……./……/…………..

The above declaration is confirmed by:

Mr. Akram ……. Date

………………………… ……./……/…………..

(Supervisor)

# **ABSTRACT**

This system is one among the web systems that will deal with all aspects about making online application for participating Ndoto Ajira competitions and than ensuring that the time and cost are reduced when making the hole application process and when providing the confirmation.

In this system we may accept or reject the application applied and than providing feedback to the one who made the application so as to show his/her talent, idea or innovation. The applicant may participate individually or in form of group this is due to the Ndoto Ajira Organization provide different types of competitions.

To achieve all of these goals, the study will employ case study as a research design, the study will interview all Ndoto Ajira’s stakeholders with the aim of getting all the details and remove those challenges that they face in their performance.

This document shows the step by step used to create this system, from researching to the implementation.

##### TABLE OF CONTENT

Contents

[**DECLARATION** i](#_Toc141984448)

[**ABSTRACT** ii](#_Toc141984449)

[**CHEPTER ONE** 1](#_Toc141984450)

[**Introduction** 1](#_Toc141984451)

[**1.1** **DESCRIPTION OF THE PROJECT** 1](#_Toc141984452)

[**1.2** **Problem statement** 1](#_Toc141984453)

[**1.3** **Problems Solution and the scope** 2](#_Toc141984454)

[**1.4 Objectives** 2](#_Toc141984455)

[**1.5 Specific objectives:-** 2](#_Toc141984456)

[**1.6 PROJECT BACKGROUND AND MOTIVATION** 3](#_Toc141984457)

[**1.7 Feasibility study report** 3](#_Toc141984458)

[**CHAPTER TWO** 4](#_Toc141984459)

[**Methodology** 4](#_Toc141984460)

[**2.1 Software development approach (object oriented or structured)** 4](#_Toc141984461)

[**2.2 Software development life cycle model(SDLC)** 4](#_Toc141984462)

[**2.3 System Architecture** 5](#_Toc141984463)

[5](#_Toc141984464)

[**2.4 Software development tools** 5](#_Toc141984465)

[**CHAPTER THREE** 6](#_Toc141984466)

[**Requirements Analysis and Modeling** 6](#_Toc141984467)

[**3.1 Requirement determination** 6](#_Toc141984468)

[**3.2 Information gathering techniques** 6](#_Toc141984469)

[**3.3 Functional requirement** 6](#_Toc141984470)

[**3.4 Non – functional requirement** 7](#_Toc141984471)

[**3.5 Requirement Structuring** 8](#_Toc141984472)

[**CHAPTER FOUR** 12](#_Toc141984473)

[**System Design** 12](#_Toc141984474)

[4.1 Architectural design 12](#_Toc141984475)

[4.2 Database Design 13](#_Toc141984476)

[**CHAPTER FIVE** 18](#_Toc141984477)

[**System implementation and testing** 18](#_Toc141984478)

[5.1 Technologies 18](#_Toc141984479)

[5.2 Database implementation 18](#_Toc141984480)

[5.3 Internal Schema of database (database schema) 19](#_Toc141984481)

[5.4 Data Dictionary 19](#_Toc141984482)

[Testing 19](#_Toc141984483)

[5.5 User Interfaces 20](#_Toc141984484)

[5.6 Strength ad Limitation of the system 20](#_Toc141984485)

[5.7 What is covered from requirements 20](#_Toc141984486)

[5.8 What is not covered 20](#_Toc141984487)

[**CHAPTER SIX** 21](#_Toc141984488)

[Conclusion, Recommendations Challenges and References 21](#_Toc141984489)

[**Conclusion** 21](#_Toc141984490)

[**Recommendation** 21](#_Toc141984491)

[My recommendation is for University and Stundents as well:- 21](#_Toc141984492)

[**Challenges** 21](#_Toc141984493)

[**Cost** 21](#_Toc141984494)

[**Error debugging** 21](#_Toc141984495)

[Some time I face a lot of error from the backend (Spring boot) and frontend (angular) so that make me stacking until I solve error so this issue loose my time so much 21](#_Toc141984496)

[**References** 22](#_Toc141984497)

##### TABLE OF FIGURES

[Figure 1System development Life circle 4](#_Toc141984420)

[Figure 2Two tire architecture 5](#_Toc141984421)

[Figure 3use case diagrame of the current system 8](#_Toc141984422)

[Figure 4use case of proposed system 9](#_Toc141984423)

[Figure 5class diagram 10](#_Toc141984424)

[Figure 6 entity relationship diagram 11](#_Toc141984425)

[Figure 7 architectural diagram 12](#_Toc141984426)

[Figure 8 database relational model 13](#_Toc141984427)

[Figure 9 Registration form 16](#_Toc141984428)

[Figure 10 login form 16](#_Toc141984429)

[Figure 11 report page 17](#_Toc141984430)

[Figure 12 interface design sample 17](#_Toc141984431)

[Figure 13 database implementation 18](#_Toc141984432)

[Figure 14 dashboard 20](#_Toc141984433)

##### ABRIVIATIONS

##### NAMS : Ndoto Ajira Management System

##### SUZA : State University of Zanzibar

##### SDLC : Software development life cycle model

##### API : Application Programming Interface

##### SQL : Structured Query Language

##### ERD : Entity Relationship Diagram

# **CHEPTER ONE**

## **Introduction**

## **DESCRIPTION OF THE PROJECT**

Ndoto Ajira Management System (NAMS) is the web based system that is developed for automating Ndoto Ajira Organization’s activities, this system can be accessed through the organization and outside as well as with proper login provided. This system can be used as online system to manage the applicant informations when they make application for Ndoto Ajira’s competition by applying either individual or group to show their talents, dream and innovation. Applicants logging should be able to upload their information as well as their idea and description to show what do their ideas mean.

The Organization may also access/search any information put up by Applicants. The project has been planned to be having the view of distributed architecture, with centralized storage of the database. The application for the storage of the data has been planned using the constructs of MYSQL Database and all the user interfaces have been designed using Angular framework technology.

The standards of security and data protective mechanism have been given a big choice for proper usage.

## **Problem statement**

## 

There has been various problems during the existing of current system because all their operations were done manually and leaded lost of money, wasting time when operating their activities, not only this but also loosing of data, also data retrieval were not easy process. Also the confirmation issue after application were using phone call so they were calling some applicants and they were unreachable at that time.

Due to the explained problems, I have come up with an idea of introdusing a technological management system known as Ndoto Ajira Management System (NAMS) wich will not only help them to solve their current problems but also keeping relationship with the applicants.

## **Problems Solution and the scope**

The proposed system will be an automated by developing a computerized system that will able to record the information about their activities both from external to internal of the Organization.

So the proposed system will help Ndoto Ajira Organization and the applicants who made application in data accessibility, updating and even deleting them when not needed. Also the proposed system will help to confirm the applicants online through their accounts after making online application so applicants may look at their confirmation at any time anywhere.

## **1.4 Objectives**

The main objective of proposed system is keeping records of Ndoto Ajira’s operations in automatically and systematically for accruing and accurate data.

Also there are specific objectives to accomplish the aim (main objective) of this project as follow:-

## **1.5 Specific objectives:-**

1. To design the database for Storing Information.
2. To develop a system that will able to give confirmation to the applicants through their accounts after making applications, in order to reduce common mistake of phone unreachable during the confirmation process.
3. To design the web-based system for their operations
4. To centralize data and information management.
5. To save cost and time during system process occur.
6. To increase data performance (data retrieving).
7. To make system easy to access information at real time and accurate.

## **1.6 PROJECT BACKGROUND AND MOTIVATION**

Currently there is no automatic system that available from Ndoto Ajira Organization in order to keep their data in safe way. The motivation is I come with an automation system (Web based system) that will solve this process, those activities will be solved by using this system.

## **1.7 Feasibility study report**

Feasibility study were conducted in organization where requirements were collected in Ndoto Ajira Organization at Kilimani Zanzibar also at Tunguu during Ndoto Ajira day (event).

* + **Technical**: It is visible because the proposed system will be very easy to interact with users and control it.
  + **Economic**: In proposed system’s development I use software that most of them are open sources, such as MySQL which is used to create and maintain database.

##### Operational: To implement this project is possible since most of the people have computer, so they can use this system anytime and anywhere if there is Internet

# **CHAPTER TWO**

## **Methodology**

## **2.1 Software development approach (object oriented or structured)**

##### In this project I will use Object Oriented Approach for system development because of the following :-

* Its bundle code into a single unit where you can determine the scope of each piece of data.
* The class can inherit attributes and behaviors from another class, you are able to reuse more code.

##### One class can be used to create many objects, all from the same flexible piece of code.

## **2.2 Software development life cycle model(SDLC)**

In this project I appreciate to use **Agile model** because this model allows changes when they are required (Time to time changes).

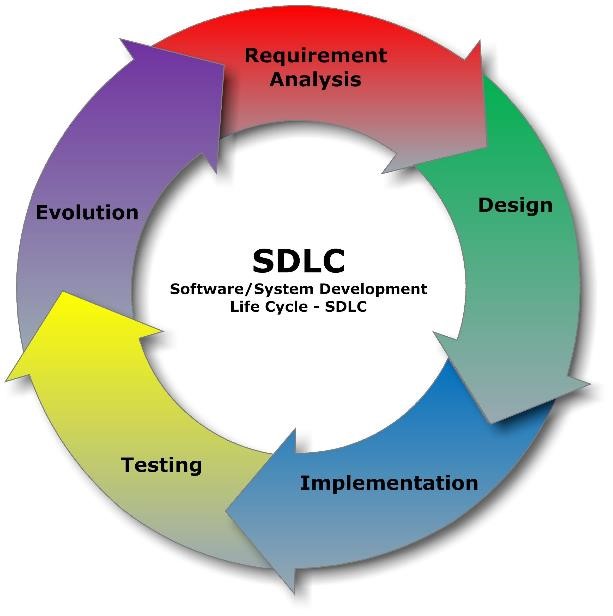


Figure 1System development Life circle

## **2.3 System Architecture**

##### The architecture used in this project is Two Tire Architecture where client side can access information and get response from the Server side and the server side can send the result to the client.

## 

Figure 2Two tire architecture

## **2.4 Software development tools**

##### In this project I’m going to use different software development tools so as to complete the proposed system those system concerning with front-end view and back-end view and they are as follow :-

##### Angular framework for front-end designing.

##### Java language for back-end.

##### Nodejs for javascript and support Angular framework

##### Postman for testing APIs.

##### Visual Studio Code (VSCode) and IntelliJ IDEA for code editing.

##### MYSQL database for creating and maintaining database.

# **CHAPTER THREE**

## **Requirements Analysis and Modeling**

### **3.1 Requirement determination**

##### In a feasibility study period, I have spent a lot of time for studying the function of the current system and how it operates in the Organization, purposely to get the data in accuracy manner.

### **3.2 Information gathering techniques**

##### When gathering requirements, I have used different way and technique for collecting needed requirement. These techniques are as follow:-

* **Interviews:** interview was carried out through means of questions and answers with the Ndoto Ajira’s members and applicants to determine what requirement is needed for the new system to be developed.
* **Observation:** also, an observation was done in order to determine the requirements by looking the processes that happen in the current system.
* **Existing system reviews:** an observation was also carried out to review the current system that are similar to the proposed system, in order to determine how the new system can be implemented.

### **3.3 Functional requirement**

##### Functional requirement these are the requirement that covering the functionalities expected by the users, a complete specification which describes all the functionalities of the system, they are as follow:-

##### The system should allow the applicants to make online application for Ndoto Ajira competitions.

* The system should able to provide the confirmation to the applicants for their application.
* The system should provide the competition advertisement to the applicants.
* The system should enable the applicants to participate in the competition individually or group of people depending on the type of competition.
* The system should enable Ndoto Ajira Organization to accept or reject the idea(application) applied by applicants.
* The system should enable Ndoto Ajira Organization to check different report concerning to their operations, Example to view all people who have ever participated in Ndoto Ajira competition.

### **3.4 Non – functional requirement**

##### These are the requirements that has no important to be entered within the system during system developing period, also doesn’t affect the system operation or performance of the system, these requirement are regarding such as reliability, portability, accessibility, maintainability and usability, they are the following:-

* The system must provide 24 hours online application service.
* The system should support almost all browsers (Internet Explorer, Microsoft Edge, Chrome and Firefox).
* The system will ensure that each applicant may access his/her only individual information for security reason.
* The system will provide access to only legitimate users, it will be secure and authorized person can use it.
* Furthermore, the system is designed in such a way any individual with even little IT knowledge can be able to use it
* The UI will be simple enough for everyone to understand and get relevant information without any special training.
* The system will be able to perform desired tasks in reasonable unit of time and as expected.

##### The system will be easy to maintain and extend, minor modification to the system will not cause harm when the system is running.

### **3.5 Requirement Structuring**

#### **3.6 Process Modelling**

#### Use case diagram to illustrate how current system is implemented

##### 

Figure 3use case diagrame of the current system

##### 3.8 Use Case model (Usercase Diagram and Use case Description) to illustrate how proposed system is implemented

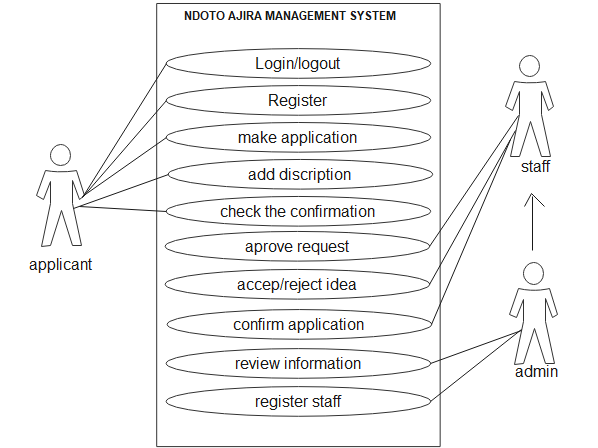


Figure 4use case of proposed system

#### 3.9 **Class diagram**

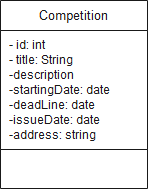
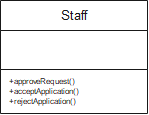
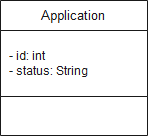
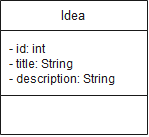
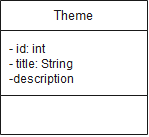
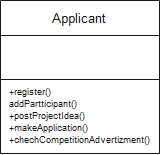
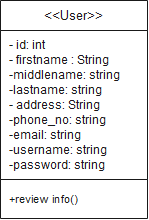
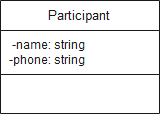


Figure 5class diagram

#### 3.1.1 Entity relationship diagram (ERD)

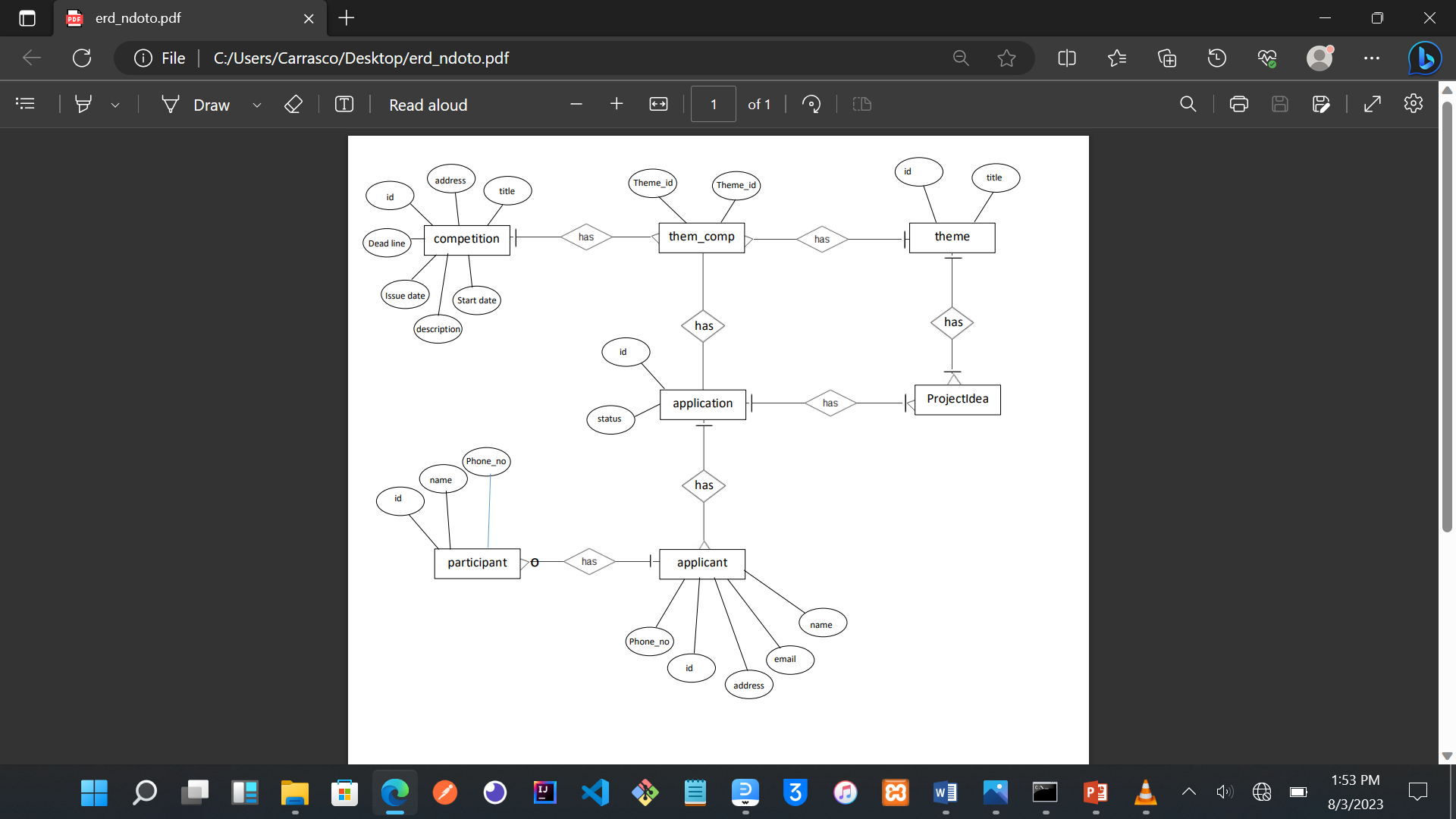


Figure 6 entity relationship diagram

# **CHAPTER FOUR**

## **System Design**

### 4.1 Architectural design

##### 

Figure 7 architectural diagram

### 4.2 Database Design

#### 4.3 Relational Model

##### 

Figure 8 database relational model

#### 4.4 Data Description

**USER:** is one that stand as a super class who inherited by applicant and staff so the information about Staff and Applicant are stored in the User table.

**APPLICANT:** this is the table that inherit form User table and act as Sub class also this is the table that is connected with Idea table because this is the one that provide Idea within the system

**PARTICIPANT**: is the one that depending on the existing of Applicant class

**COMPETITION:** is the table that is used to contain the data used to inform Applicant table when where the competition will take place.

**THEME:** this the table that that store and provide the title of the Competitions

**IDEA:** is the table that contain the title and description about title that the Applicant is aim to provide

#### 4.5 Data Dictionaries

Table Name: The name for the Table.

Column Name: The predefined name for the column.

Description: A detailed description of the contents and purpose for the column

Data Type: The predefined characteristics for the column.

**Table name: User(Applicant & Staff)**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **COLUMN**  **NAME** | **DESCRIPTION** | **DATA TYPE** |
| 1 | id | A unique identification of users table | Long |
| 2 | Name | Names of users (First, middle and last). | String |
| 3 | Address | The location of user is live. | String |
| 4 | Phone | The phone number of user | String |
| 5 | Email | The email address of user | String |
| 7 | Username | This is the user’s credential to get authentication | String |
| 8 | Password | The encrypted password of user | String |

**Table name: Applicant**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **COLUMN**  **NAME** | **DESCRIPTION** | **DATA TYPE** |
| 1 | id | A unique identification | Long |

**Table name: Staff**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **COLUMN**  **NAME** | **DESCRIPTION** | **DATA TYPE** |
| 1 | id | A unique identification | Long |

**Table name: Application**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **COLUMN**  **NAME** | **DESCRIPTION** | **DATA TYPE** |
| 1 | id | A unique identification | Long |
| 2 | Status | Shows the status of the applications | String |

**Table name: Idea**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **COLUMN**  **NAME** | **DESCRIPTION** | **DATA TYPE** |
| 1 | id | A unique identification | Long |
| 2 | title | The name of the idea | String |
| 3 | description | Brief explanation about title | String |

**Table name: Competition**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **COLUMN**  **NAME** | **DESCRIPTION** | **DATA TYPE** |
| 1 | id | A unique identification | Long |
| 2 | title | The heading of the competition | String |
| 3 | description | Brief explanation about title | String |
| 4 | startingDate | When the competition exist | Date |
| 5 | Deadline | The last date for application | Date |
| 6 | IssueDate | When the competition take palce | Date |
| 7 | Address | The location where the competition exist | String |

#### 4.6 User Interface Design

#### 4.7 Forms and Reports

#### **Registration form sample**

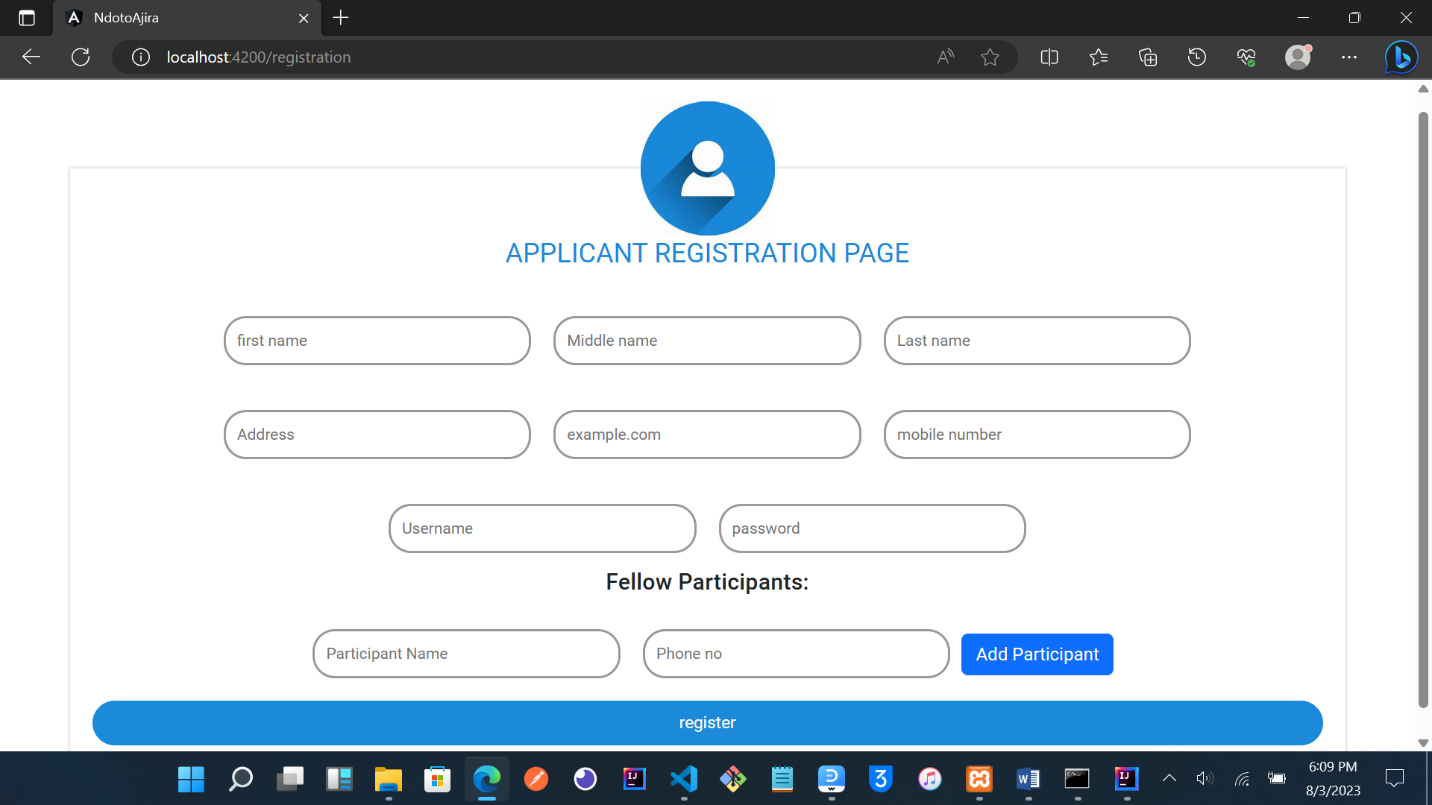


Figure 9 Registration form

Logon Form

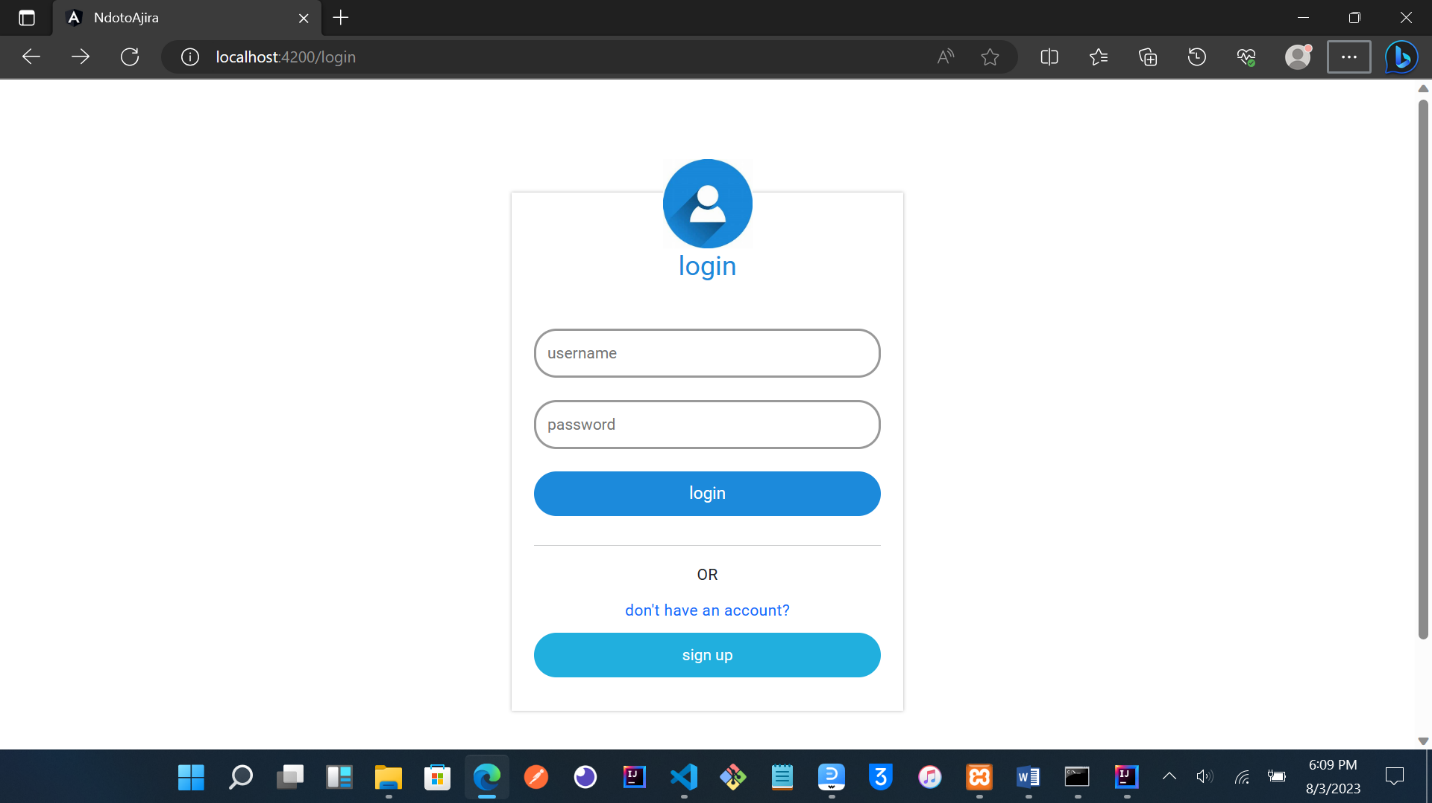


Figure 10 login form

#### **Report**

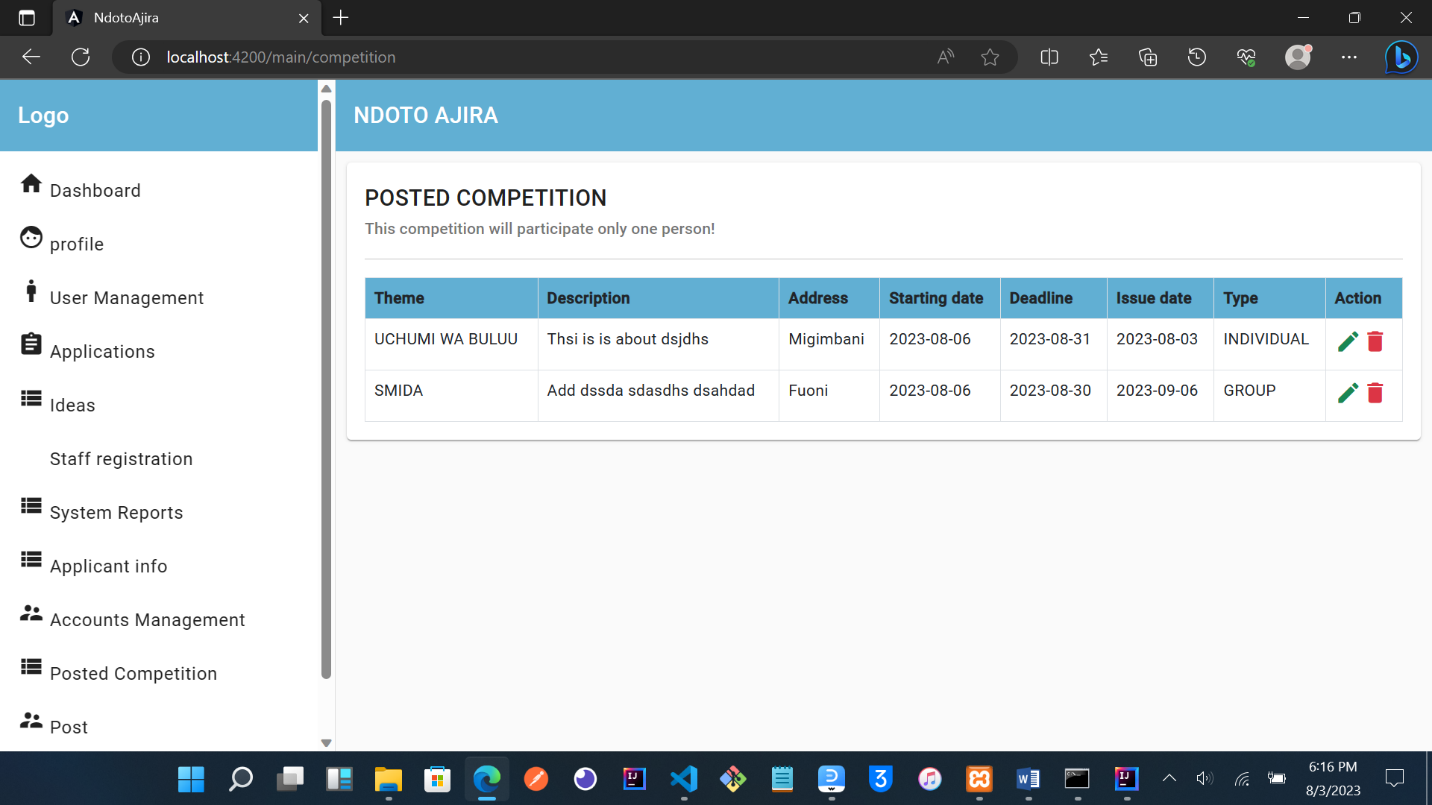


Figure 11 report page

#### 4.8 Interface design sample

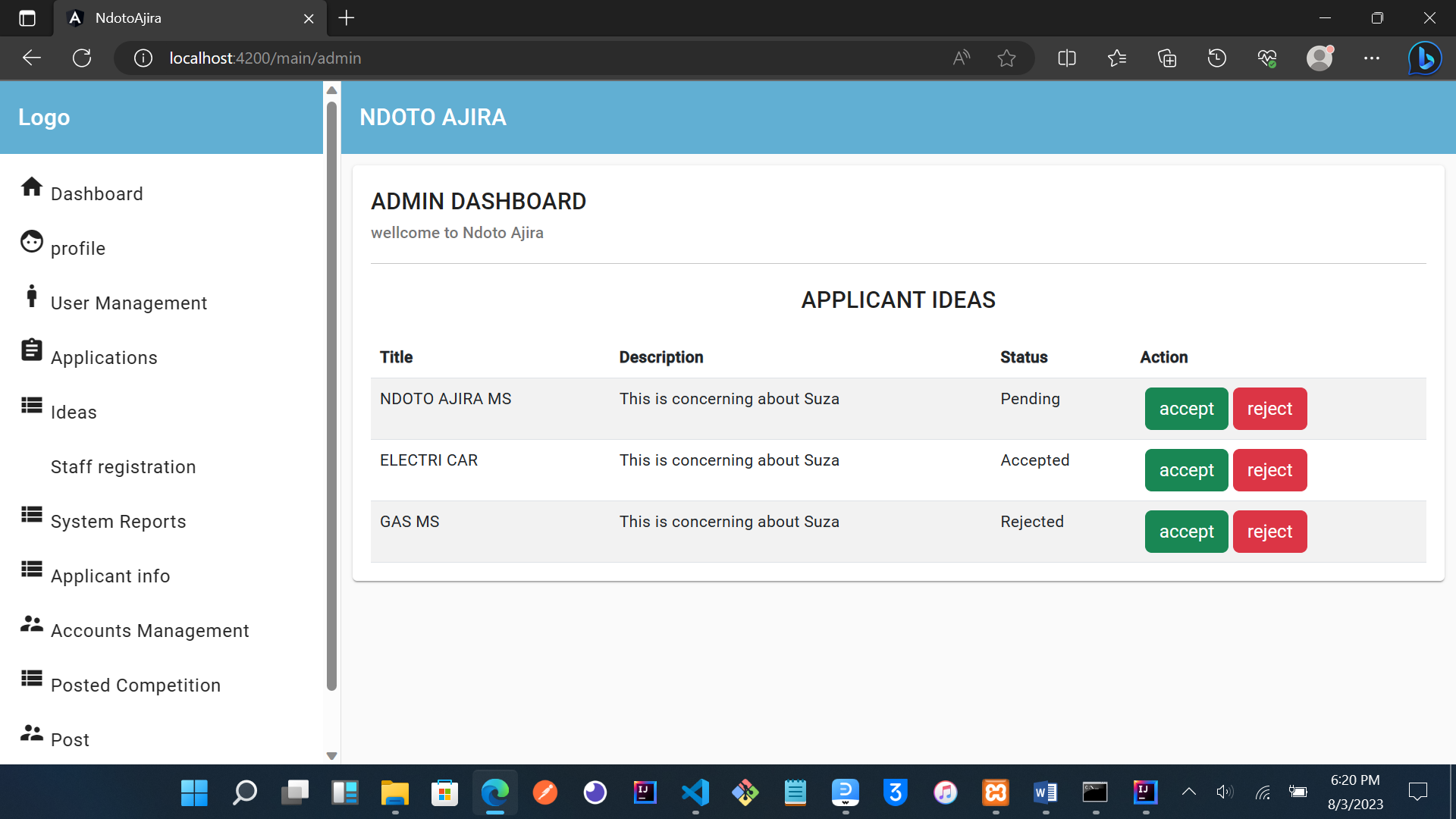


Figure 12 interface design sample

# **CHAPTER FIVE**

## **System implementation and testing**

### 5.1 Technologies

In this project we use various technology such as JAVA(spring boot), HTML, WEBSERVICES, CSS,Tyscript(angular), Android Studio(VScode) and MySQL.

**HTL codes:-**

    <mat-card>

        <mat-card-header>

            <mat-card-title>

                ADMIN DASHBOARD

            </mat-card-title>

            <mat-card-subtitle>wellcome to Ndoto Ajira</mat-card-subtitle>

        </mat-card-header>

        <mat-card-content>

            <!--inside card--> <hr>

            <h2 class="text-center">APPLICANT IDEAS</h2>

                <table class="table table-striped">

                    <thead>

                    <tr>

                        <th>Title</th><th>Description</th><th>Status</th><th>Action</th>

                    </tr>

                </thead>

                <tbody>

                    <tr \*ngFor="let idea of ideas">

                        <td>{{idea.title}}</td>

                        <td>{{idea.description}}</td>

                        <td>{{idea.status}}</td>

                        <td>

                            <button class="btn btn-success" (click)="acceptedApplication(idea.ideaId)">accept</button>

                            <button class="btn btn-danger" (click)="rejetedApplication(idea.ideaId)">reject</button>

                        </td>

                    </tr>

                </tbody>

                </table>

                </mat-card-content>

**JAVA codes:-**

@RestController  
public class CompetitionController {  
  
 @Autowired  
 private CompetitionService competitionService;  
  
 @Autowired  
 private CompetitionServiceImpl competitionServiceImpl;  
  
  
 @PostMapping("/competitions")  
 public Competition saveCompetition(@RequestBody Competition competition) {  
 return competitionService.saveCompetition(competition);  
 }  
  
 @GetMapping("/competitions")  
 public List<Competition> fetchCompetitionList() {  
 return competitionService.fetchCompetitionList();

**Tyscript code:-**

@Component({

  selector: 'app-admin',

  templateUrl: './admin.component.html',

  styleUrls: ['./admin.component.css']

})

export class AdminComponent implements OnInit{

  ideas!: Idea[];

  constructor(private ideaService: IdeaService){}

  ngOnInit(): void {

    this.getIdeas();

  }

  private getIdeas(){

    this.ideaService.getIdeaList().subscribe(data =>{

      this.ideas! = data;

    });

  }

  acceptedApplication(id: number){

    this.ideaService.acceptApplication(id).subscribe(

      data=>{

        console.log(data);

      //   this.snack.open('Application Accepted','',{

      //     duration:3000,

      //   });

        this.getIdeas();

      },

      (error) => {

        console.log(error);

      }

    );

  }

  rejetedApplication(id: number){

    this.ideaService.rejectApplication(id).subscribe(

**CSS code:-**

.idea\_card{

    width: 400px;

    float: left;

}

.btn{

    margin-left: 4px;

}

### 5.2 Database implementation

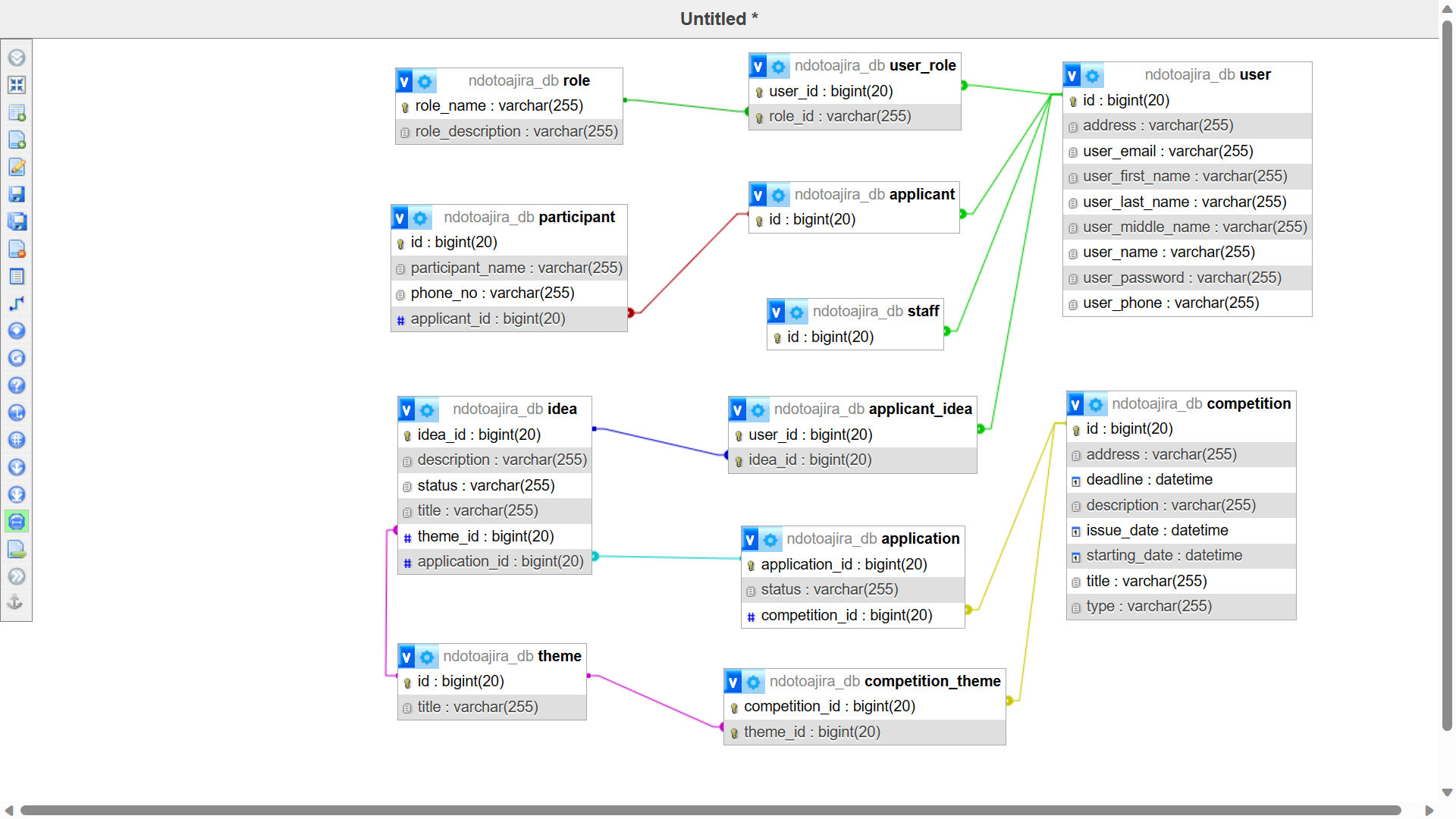
****

Figure 13 database implementation

### 5.3 Internal Schema of database (database schema)

@Entity  
@Data  
@NoArgsConstructor  
@AllArgsConstructor  
public class Competition {  
  
 @Id  
 @GeneratedValue(strategy = GenerationType.*IDENTITY*)  
 private Long id;  
 private String title;  
 @JsonFormat(pattern = "yyyy-MM-dd")  
 private Date issueDate;  
 private String description;  
 private String address;  
 private String type;  
 @JsonFormat(pattern = "yyyy-MM-dd")  
 private Date deadline;  
 @JsonFormat(pattern = "yyyy-MM-dd")  
 private Date startingDate;  
   
  
 @EqualsAndHashCode.Exclude  
 @ManyToMany(cascade = CascadeType.*ALL*)  
 @JoinTable(name = "competition\_theme",  
 joinColumns = {  
 @JoinColumn(name = "competitionId")  
 },  
 inverseJoinColumns = {  
 @JoinColumn(name = "themeId")  
 })  
 private Set<Theme> themes = new HashSet<>();  
 //--------------------------------------  
  
 @OneToMany(cascade = CascadeType.*ALL*)  
 @JoinColumn(name = "competitionId", referencedColumnName = "id")  
 private Set<Application> applications;  
  
}

### 5.4 Data Dictionary

### Testing

In this project testing provided is like unit testing, component testing and system testing.

UNIT TESTING/ COMPONENT TESTING

Unit testing, also known as Component testing is a level of software testing where individual units / components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output.

Therefore, here we test by using login credential of Username and password , by putting wrong Username and password that does make us login. And using correct Username and password which result to login in the system.

### 5.5 User Interfaces

Dashboard(home page)

This is page that enable ndoto ajira completion to accept or reject the idea provided by applicant when making application.

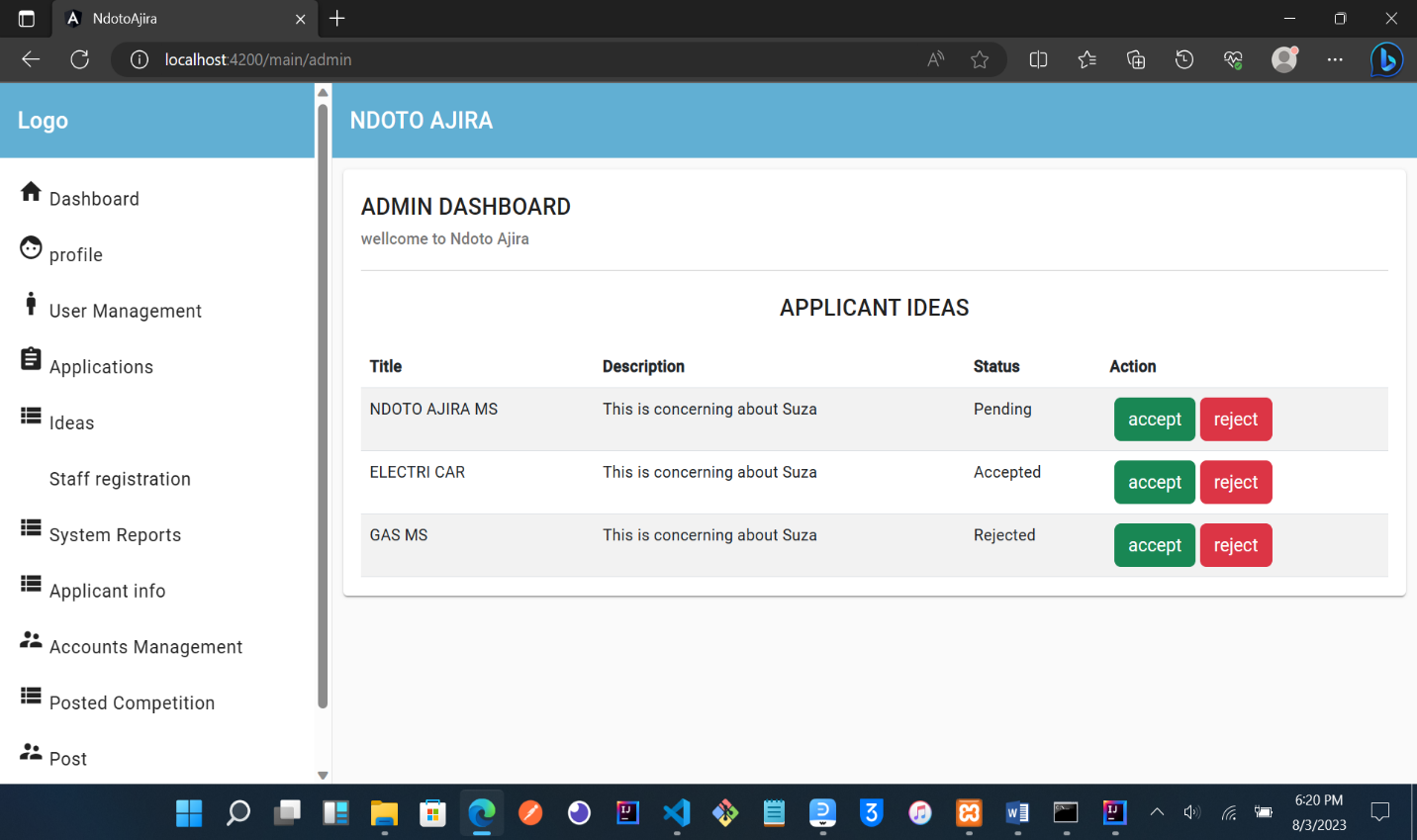


Figure 14 dashboard

### 5.6 Strength ad Limitation of the system

### 5.7 What is covered from requirements

In this project we cover most of the requirement as mentioned above include functional requirements and non-functional requirements.

### 5.8 What is not covered

I just not cover some of validation in the few forms in my peroject.

**CHAPTER SIX**

## Conclusion, Recommendations Challenges and References

### **Conclusion**

Ndoto Ajia Management System (NAMS) is the web based system that will help the all people who are touched (stackeholder) by Ndoto Ajira’s operations, this system will help in time consuming, cost saving, data retrieval and so on, without forgetting in confirmation issue.

## **Recommendation**

## My recommendation is for University and Stundents as well:-

* The university should programme well the modules and final year project because it become hard in the last semester we have a lot of modules to study and we have final year together.
* And for the student, I advice them to start early doing a project because as time goes cause you do not complete things in your project.

## **Challenges**

### **Cost**

I have financial problems across the project development for Learning a lot of things concerning with implantation such as looking at how are some JAVA method created so as to complete the implementation because I use emerging technology (Spring boot & angular) so is hard to get the guidance from the people.

Not only that but also cost in delivery of System which is required to be in hard copy and soft copy within the CD so it cost a lot in terms of printing system documentation.

### **Error debugging**

## Some time I face a lot of error from the backend (Spring boot) and frontend (angular) so that make me stacking until I solve error so this issue loose my time so much

## **References**

[www.google.com](http://www.google.com)

https://[angular](https://angular.io/).io

start.spring.io

[Spring Boot Tutorial | Full Course [2023] [NEW] - YouTube](https://www.youtube.com/watch?v=9SGDpanrc8U)

Pardigm, V. (2017, September 25). *Conceptual, Logical and Physical Data Model*.

Visual Paradgm. [https://cs.visual-paradigm.com/.](https://cs.visual-paradigm.com/)