

Kusa Vincent Onyango 17-0203

Joab Rogony 16-0798

ACS 362: Advanced Database

Submitted to:

Zipporah Mwololo

Documentation for a Private Blog Newspaper

Department of Computer Science

School of Science Engineering and Health

2nd December 2019

**Documentation for a Private Blog Newspaper**

A Private Blog Newspaper is a newspaper that has been using print as a means of reaching the consumer for a very long time now.

We came up an online blog as a means of reaching the consumer from now on since everything is now going digital and according to us, print media is dying and the newspaper company should not be left behind.

It is a user-friendly website that enables an individual or group to post content at any time and place either on the smartphone or computer/laptop.

**Operating Environment**

This system is a web application hence will run on any web browser of choice; it is also responsive thanks to @media CSS.

Build using HTML5, CSS3 and JavaScript to communicate with the NoSQL (Firebase).

**System Administrator**

They will be responsible of coordinating the team and regulating the blog/article written.

Read and write everyone’s post

**Technical Users**

These are persons with medium level access of the system. They can manipulate data contained in the database. They will after registration be able to sign up to gain the page access, write a blog read and also delete the blog that they are the ones who wrote.

**Users (Novice)**

They only have the read capability and are not able to write into the blog.

**SPECIFIC REQUIREMENTS**

**User interface requirements**

The system user interface will be designed to ease the functionalities of the system. This will be done according to the user specifications and the level or class of the user i.e. the users past experience.

**Hardware interface requirements**

This includes all the computer components used **in input, storage** and **retrieval** of information by the users for example:

**Software interface requirement**

This comprises of the software that enable effective connections between this system and other specific software components. The system is expected to link:

* Network operating system (NOS)
* Network server communication protocol (NSCP)
* Among other operating.

To help in port communication.

**Other software requirements include:**

* Browser
* Firebase Database (Authentication, Database and storage).

**Tier client arrangement (Three Tier Architecture)**

This arrangement is called thin client/ web arrangement. This architecture adds a third tier to the first client that runs the applications performing the business logic and also providing communication between the client and the database server (This tier is web application server).

**Architectural Design**

This is the initial identification of the subsystems control and communication or interface.

Registrar Master consists of several subsystems which are interconnected together to attain a specific goal. The proposed style implemented 3 tier system architecture. The browser makes request to the JavaScript engine, the requests are then sent to the web server and then to the database. The information is processed and the results are sent back to the web browser through the same path.

Client

User Interface

Application

Business Logic

(Application program/ Web program is run here)

Database Server

Database access

**Server Side- Validation**

***Top- Down Functional Decomposition***

In this approach, the system is iteratively broken down into hierarchical subsystems which are, in turn, broken down into components. The design of this system followed this approach.

**A model of how the system operates is documented below: -**

**Entities**

**Sign UP: Users**

Email: [PK] ATTRIBUTE Type: VARCHAR, NOT NULL, UNIQUE

password: ATTRIBUTE Type: VARCHAR, NOT NULL

confirm password: ATTRIBUTE Type: VARCHAR, NOT NULL

**Account Details: Users**

FirstName: ATTRIBUTE Type: VARCHAR, NOT NULL

SecondName: ATTRIBUTE Type: VARCHAR, NOT NULL

Country: ATTRIBUTE Type: VARCHAR, NOT NULL

Gender: ATTRIBUTE Type: VARCHAR, NOT NULL

Address: ATTRIBUTE Type: VARCHAR, NOT NULL

Bio: ATTRIBUTE Type: VARCHAR, NOT NULL

**ForgotPassword**

Email: [PK] ATTRIBUTE Type: VARCHAR, NOT NULL

**Blog**

Description: ATTRIBUTE Type: VARCHAR, NOT NULL

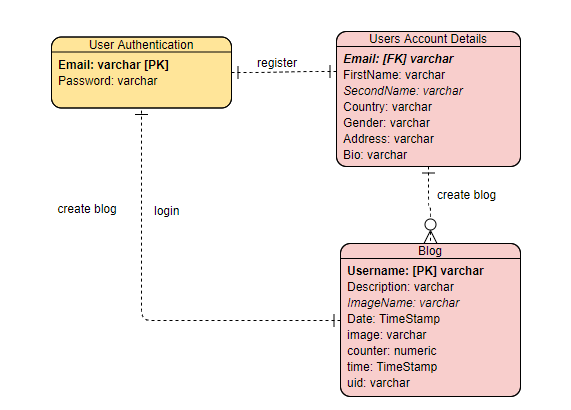
Image: ATTRIBUTE Type: VARCHAR, NOT NULL

Username (Composite of FirstName ‘||’ SecondName from User, it selects this from user who is online as at that time) ATTRIBUTE Type: VARCHAR

***Technically NoSQL database is not relational hence do not use tables, there for have no Entity Relational Diagram but because of clarity this is just a demonstration of what is going on.***

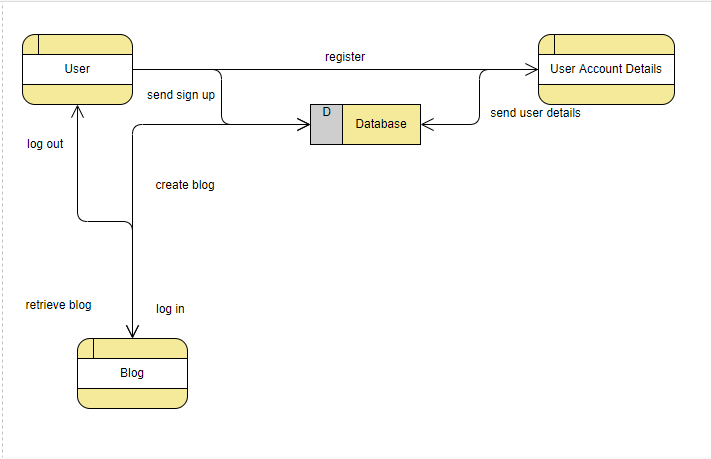
*Schema used is* ***Document***.

**Entity Relationship Diagram (ERD)**



**Data Flow Diagram for the Private Blog Newspaper**

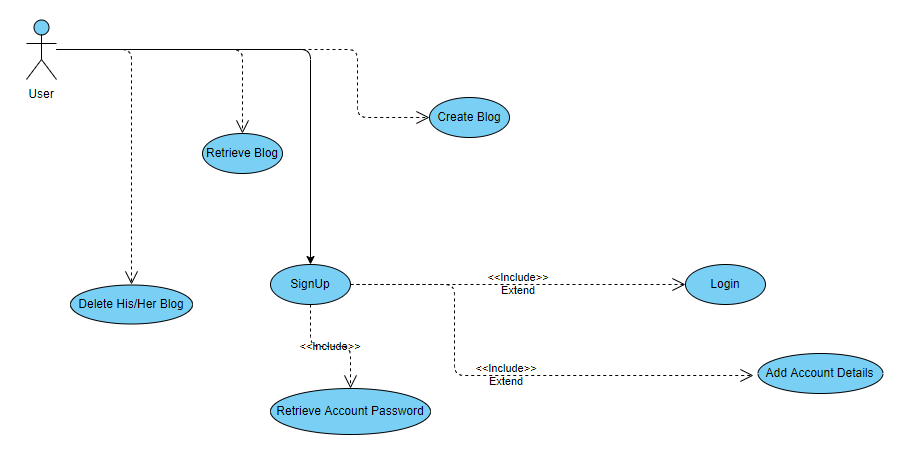
**Level 0 DFD**

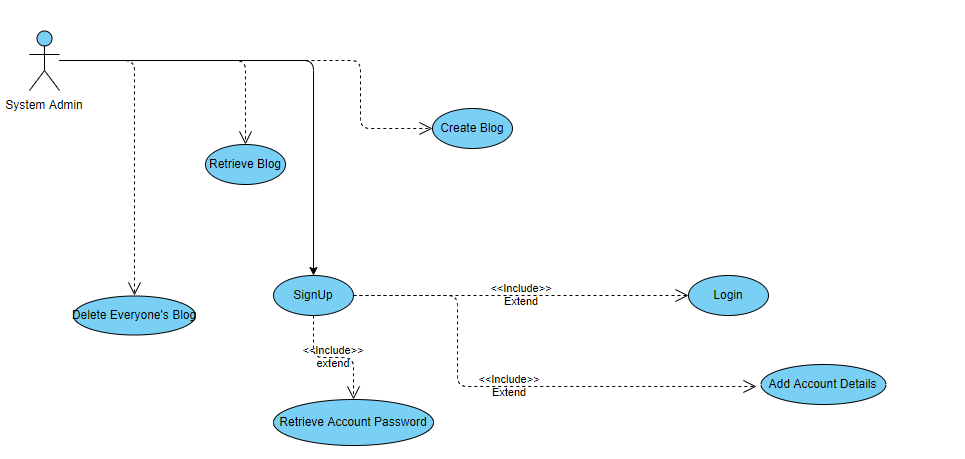


**­­­­­­­­­­­**

**Use case diagram:**

It is a simple representation of user’s interaction with the system that shows the relationship between the user and different use cases (list of actions, events) in which the user is involved.





**Database appearance for the Private Blog Newspaper**

Table: Users Authentication

| **Column** | Type | Comment |
| --- | --- | --- |
| **Email** | varchar [**not null**] |  |
| **password** | varchar [**not null**] |  |

### **Indexes**

|  |  |
| --- | --- |
| **PRIMARY** | *Email* |

## Table: Users Account Details

| **Column** | Type | Comment |
| --- | --- | --- |
| **Email** | varchar [**not null**] |  |
| **password** | varchar [**not null**] |  |
| **Country** | varchar [**not null**] |  |
| **Gender** | varchar [**not null**] |  |
| **Bio** | varchar [**not null**] |  |
| **FirstName** | varchar [**not null**] |  |
| **SecondName** | varchar [**not null**] |  |

### **Indexes**

|  |  |
| --- | --- |
| **FOREIGN KEY** | Users Email |

## Table: Blog

| **Column** | Type | Comment |
| --- | --- | --- |
| **Description** | varchar [**not null**] |  |
| **Image** | varchar [**not null**] |  |
| **UserName** | varchar [**not null**] |  |

### **Indexes**

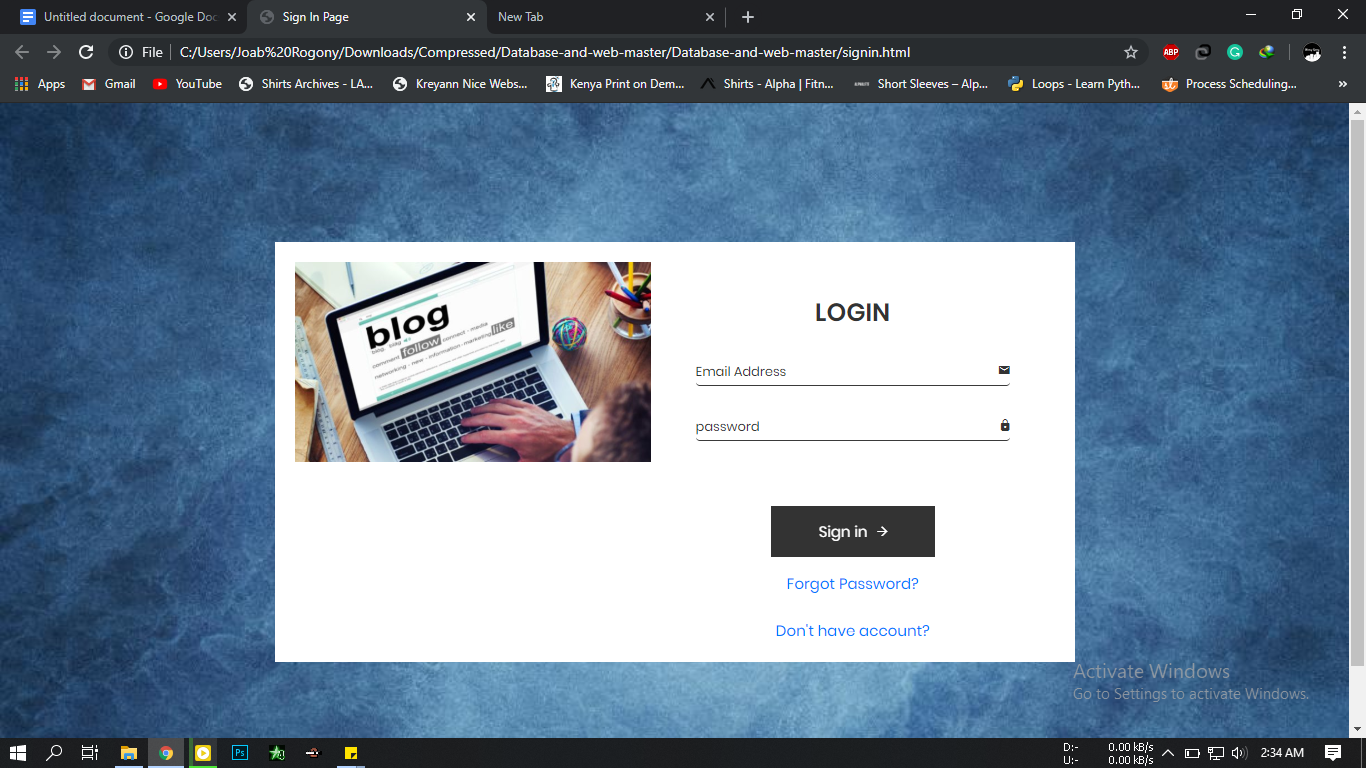
|  |  |
| --- | --- |
| **FOREIGN KEY** | UserName |

Content

This blog contains seven pages which include:-

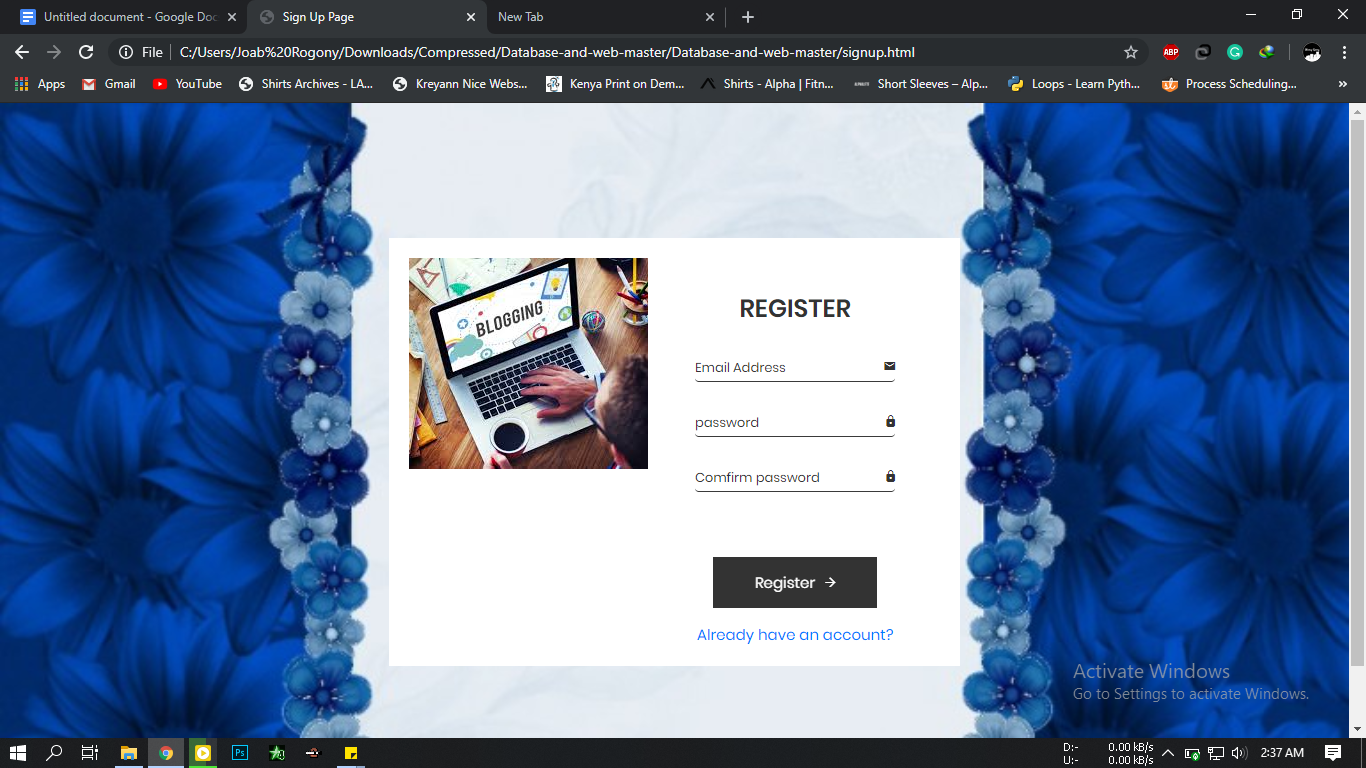
* Main page
* Forgot password page
* Sign-in page
* Myblog page
* Sign-up page
* Home page
* Account settings page

Main Page



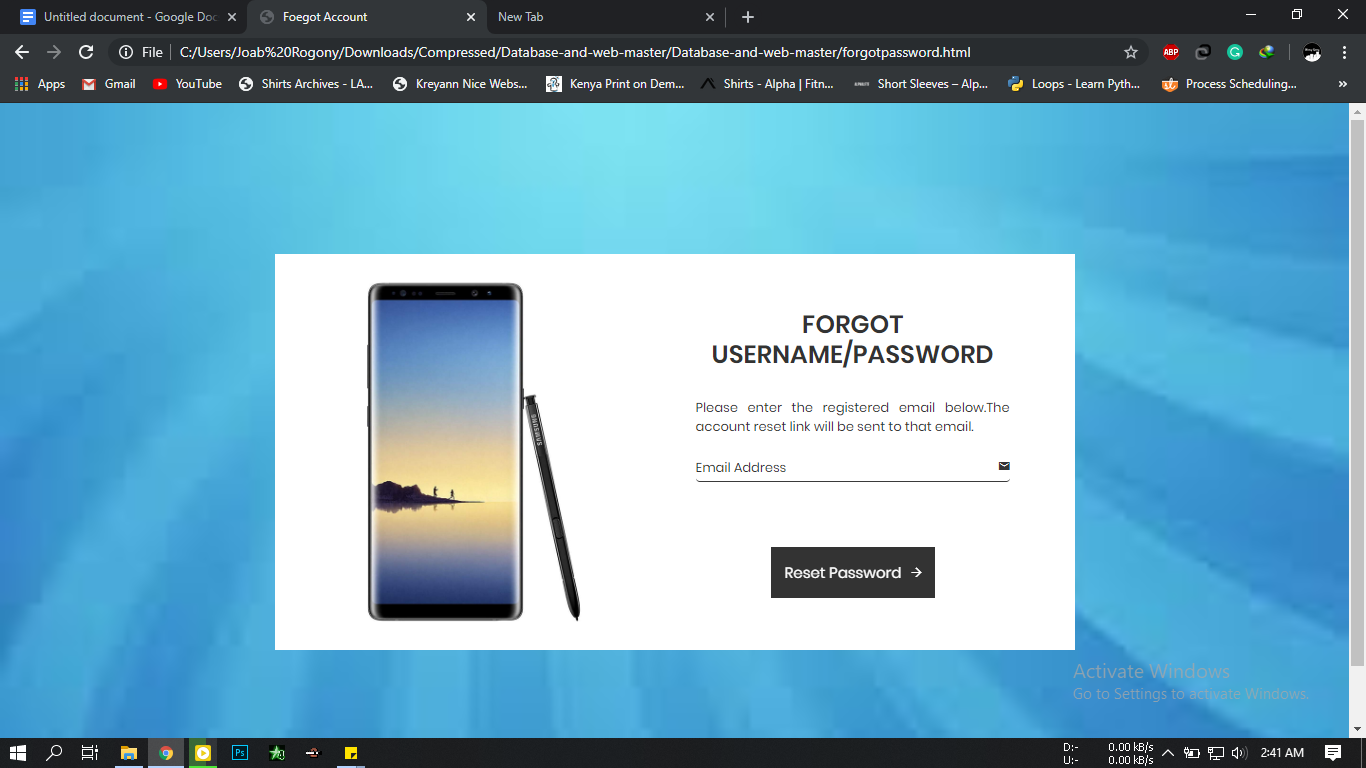
* This is the login/Main page where the user logins to his/her account.
* It comes with the option of recovering the users password and creating a new account.
* Forgot password takes you to the forgot password page
* “Do you have an account” takes the user to the register page.

Sign-up Page



* The user is able to register here and get an account and get started with his/her blogging.
* The “Already have an account” takes the user back to the main login page.

Forgot password page



Here the user is able to recover his/her password by accessing the database connected to this website and give the user his/her password after a verification.

