Online Birth Certificate System



**Project Report By**

# ACKNOWLEDGEMENT

I take this opportunity to express my profound gratitude and deep regards to my teachers Prof. and

Prof . for their exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by them time to time shall carry me a long way in the journey of life on which I am about to embark.

Special gratitude I give to my respected head of the division Mr.XYZ, for allowing me to use the facilities available and also help me to coordinate my project

Furthermore, I would also like to acknowledge with much appreciation the crucial role of faculty members on this occasion.

Lastly, I thank almighty, my parents and friends for their constant encouragement without which this project would not be possible.

.

**----------------------**

# Abstract

Online Birth Certificate System maintains a good record of date of birth of people. This system helps admin to view data of date of birth of people who reside in country.

The main objective of “Online Birth Certificate System” project is to providing easier registration of date of birth and gets certificate of birth online which save lots of time.

# Introduction

**Introduction:-**

In Online Birth Certificate System we use PHP and MySQL Database. This project has two module i.e. admin and user.

**Admin Module**

1. **Home**: In this section, admin can briefly view the total number of new application, total verified application and total rejected application.
2. **Birth Application:** In this section, admin view the application details and they have also right to change application status according to current status.
3. **Reports:** In this section admin can view the application details in a particular period.
4. **Search:** In this section, admin can search application with the help of customer application

Admin can also update his profile, change the password and recover the password.

**User Module**

1. **Home Page**: In this section, user can view welcome page of web application.
2. **Birth Reg Form**: In this section, user can fill the form of birth certificate and see the status of his/her application.
3. **Certificate:** In this section user can take print of verified certificate.

User can also update his profile, change the password and recover the password. User can also update his profile, change the password and recover the password. User can also verify certificate without login.

**Purpose:-**

The purpose of developing birth certificate management system is to computerized the tradition way of birth registration. Another purpose for developing this application is to generate the report automatically.

**Scope:-**

Births registration plays a very important role in planning of various government schemes. All the important information like place of birth date, place of birth and vital particular at the time of Births are required in various places so this project helps to maintain all these records at one place which is useful for government as well as people.

# Requirement Specification

**Hardware Configuration :**

**Client Side:**

|  |  |
| --- | --- |
| **RAM** | 512 MB |
|  |  |
| **Hard disk** | 10 GB |
|  |  |
| **Processor** | 1.0 GHz |
|  |  |

**Server side:**

|  |  |
| --- | --- |
| **RAM** | **1 GB** |
| **Hard disk** | **20 GB** |
| **Processor** | **2.0 GHz** |

**Software Requirement:**

**Client Side:**

|  |  |
| --- | --- |
| **Web Browser** | Google Chrome or any  compatible browser |
| **Operating System** | Windows or any equivalent OS |

**Server Side:**

|  |  |
| --- | --- |
| **Web Server** | APACHE |
| **Server side Language** | PHP5.6 or above version |
| **Database Server** | MYSQL |
| **Web Browser** | Google Chrome or any compatible browser |
| **Operating System** | Windows or any equivalent OS |

## APACHE

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.

The Apache HTTP Server ("httpd") was launched in 1995 and it has been the most popular web server on the Internet since April 1996. It has celebrated its 20th birthday as a project in February 2015.

## PHP

* PHP stands for PHP: Hypertext Preprocessor.
* PHP is a server-side scripting language, like ASP.
* PHP scripts are executed on the server.
* PHP supports many databases (MYSQL, Informix, Oracle, Sybase, Solid, Generic ODBC, etc.).
* PHP is an open source software .
* PHP is free to download and use.

## MYSQL

* MYSQL is a database server
* MYSQL is ideal for both small and large applications
* MYSQL supports standard SQL
* MYSQL compiles on a number of platforms
* MYSQL is free to download and use
* How to access MySQL: <http://localhost/phpmyadmin>

# Analysis and Design

**Analysis:**

In present all birth certificate system work done on the paper. The whole year data is stored in the registers. We can’t generate reports as per our requirements because its take more time to calculate the date of birth report.

**Disadvantage of present system:**

* **Not user friendly:** The present system not user friendly because data is not stored in structure and proper format.
* **Manual Control:** All report calculation is done manually so there is a chance of error.
* **Lots of paper work:** Visitors maintain in the register so lots of paper require storing details.
* **Time consuming**

**Design Introduction:**

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer’s requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data

UML Diagrams:

Actor:

A coherent set of roles that users of use cases play when interacting with the use `cases.

Use case:A description of sequence of actions, including variants, that a system performs that yields an observable result of value of an actor.

UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

## USECASE DIAGRAMS:

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what’s called an actor.

Use case diagram can be useful for getting an overall view of the

system and clarifying who can do and more importantly what they can’t do.

Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

* The purpose is to show the interactions between the use case and actor.
* To represent the system requirements from user’s perspective.
* An actor could be the end-user of the system or an external system.

**USECASE DIAGRAM:** A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object. Primary Actor – Sender, Secondary Actor Receiver.

**Use Case Diagrams:**



**Admin**

Sign In

Dashboard

Manage Application

Search Application

Generate Reports

Update Profile

Change Password

Password Recovery

Visit Website

**User**

Verify Certificate



Sign Up

Sign In

Dashboard

Fill Application

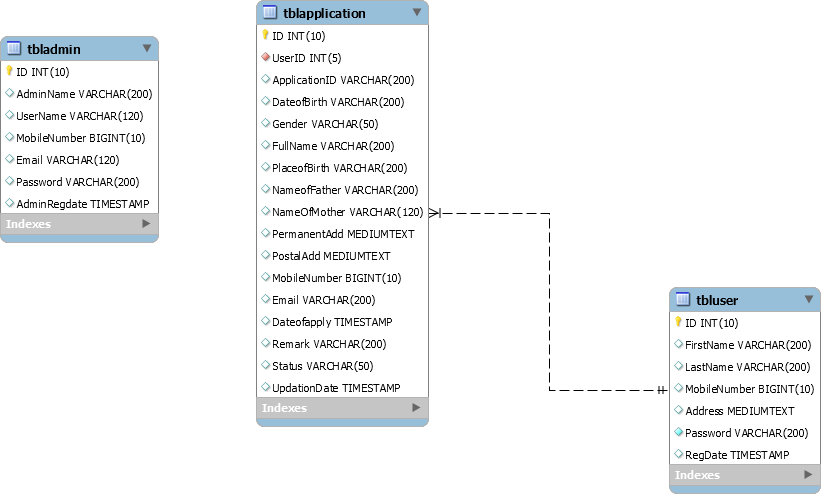
View Application

Update Profile

Change Password

Password Recovery

**Class Diagram:**



A description of set of objects that share the same attributes operations, relationships, and semantics

**ER Diagram:**

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design for the database designer, the utility of the ER model is:

* It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
* It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
* In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

**ER Notation**

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today,

there are a number of notations used; among the more common are Bachman, crow's foot, and IDEFIX.

All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

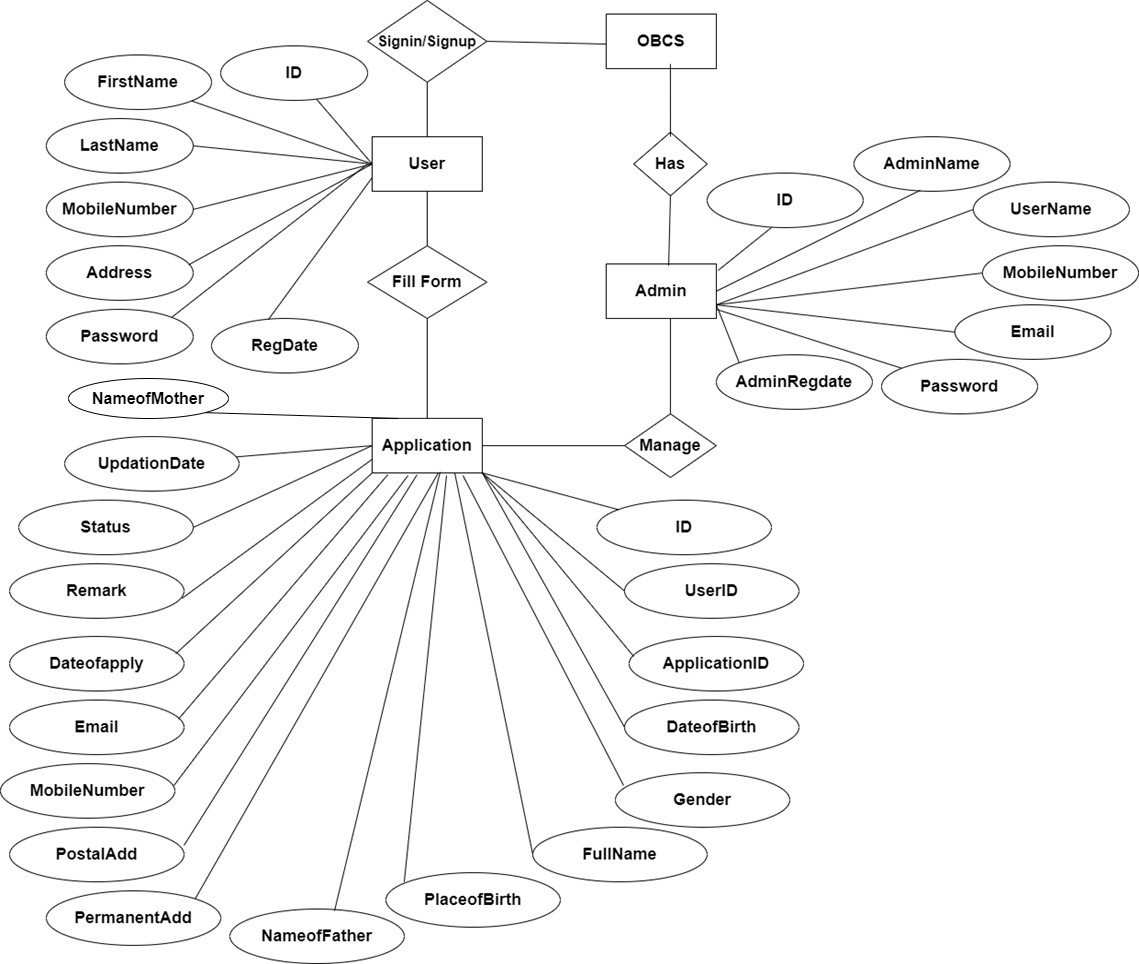
* **Entities** are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.
* **Relationships** are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs
* **Attributes**, when included, are listed inside the entity rectangle.

Attributes which are identifiers are underlined. Attribute names should be singular nouns.

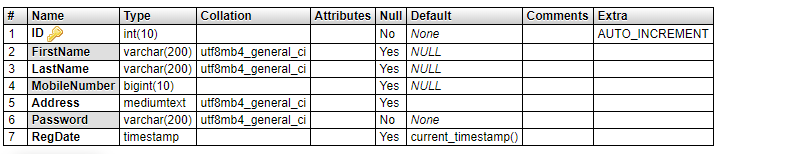
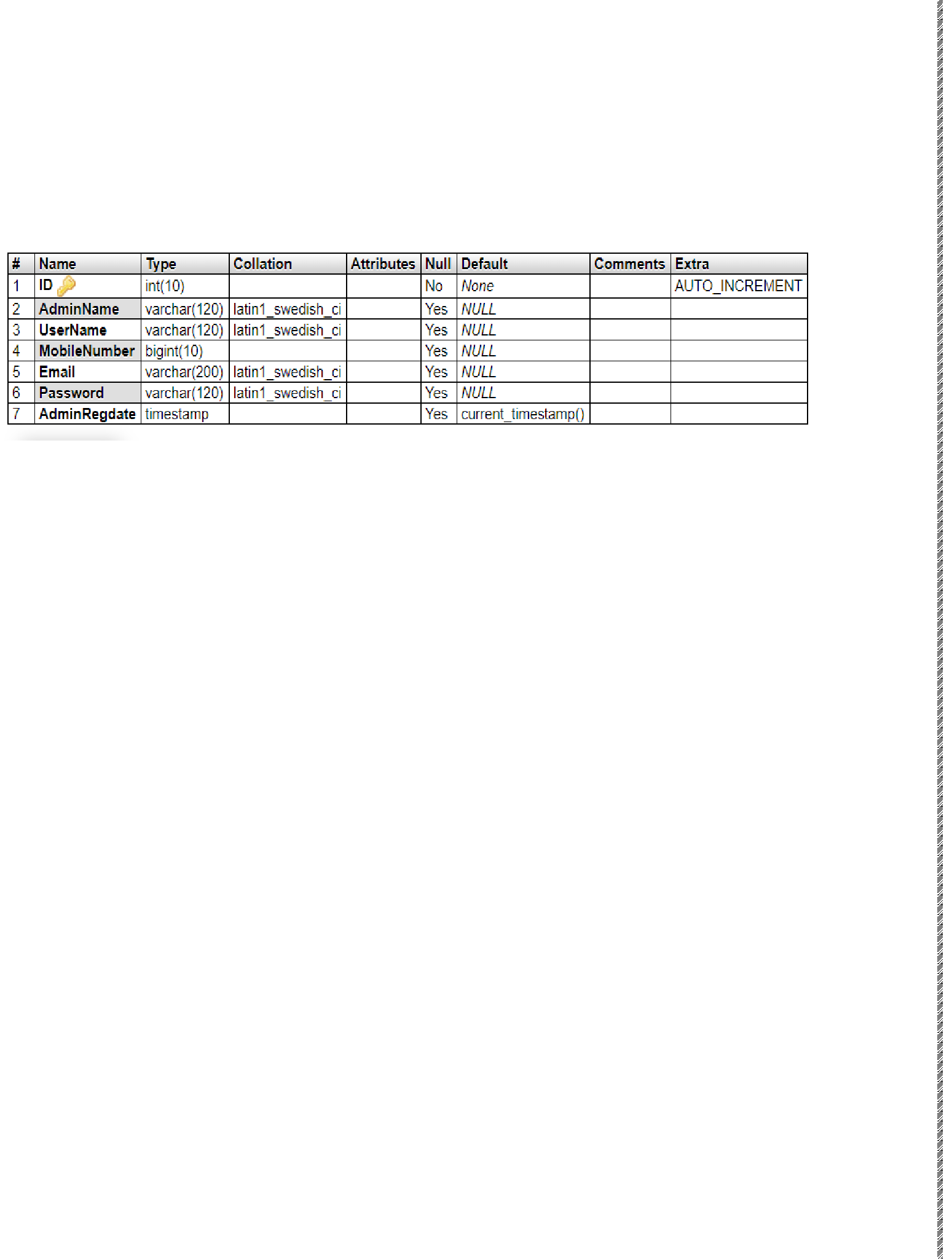
* **Cardinality** of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.

**Existence** is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional.

**ER Diagram**



**MySQL Data Tables:**

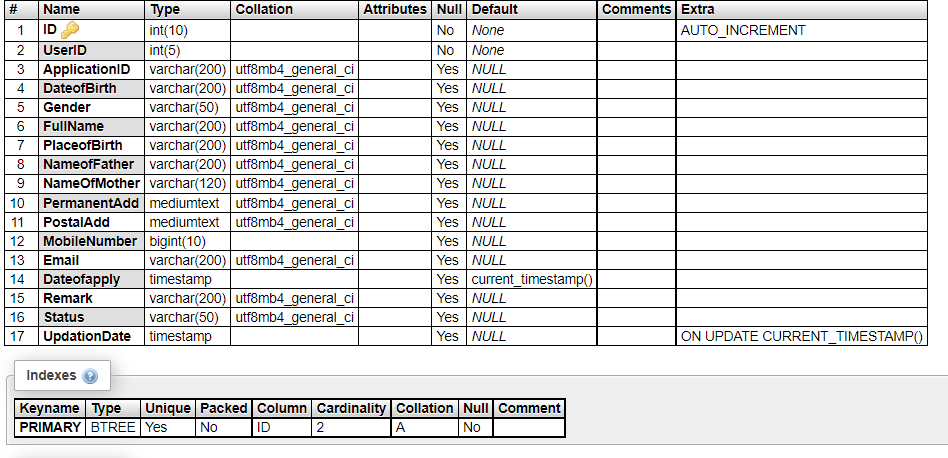


**Admin Table:**(Table name is tbladmin)

This store admin personal and login details.

**User Table(**Table name is tbluser) This stores detail of registered users.

**Application Table:** (Table name is tblapplication)



This table stores the detail of user who applies for birth certificate.

# Implementation and System Testing

After all phase have been perfectly done, the system will be implemented to the server and the system can be used.

**System Testing**

The goal of the system testing process was to determine all faults in our project .The program was subjected to a set of test inputs and many explanations were made and based on these explanations it will be decided whether the program behaves as expected or not. Our Project went through two levels of testing

1. Unit testing
2. Integration testing

## UNIT TESTING

Unit testing is commenced when a unit has been created and effectively reviewed .In order to test a single module we need to

provide a complete environment i.e. besides the section we would require

* + The procedures belonging to other units that the unit under test calls
  + Non local data structures that module accesses
  + A procedure to call the functions of the unit under test with appropriate parameters

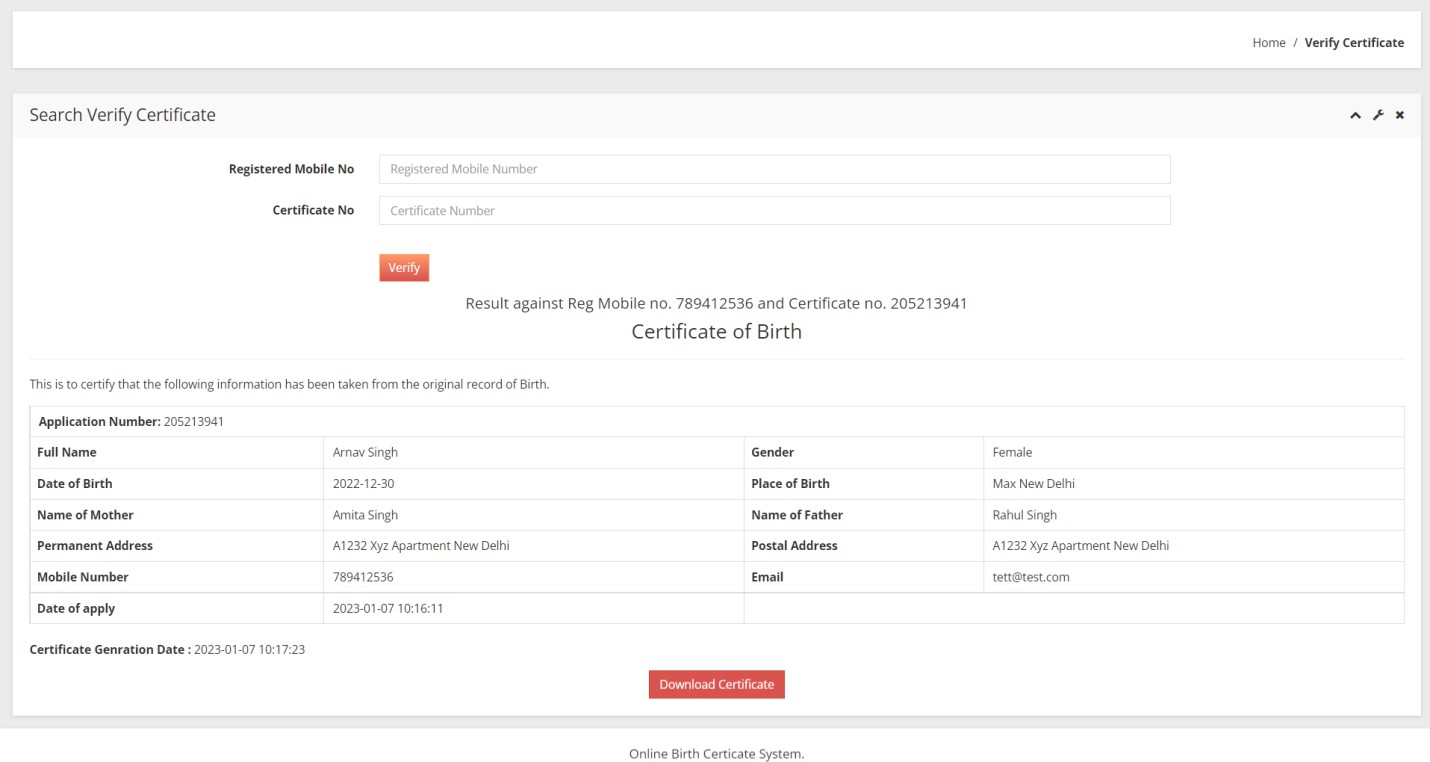
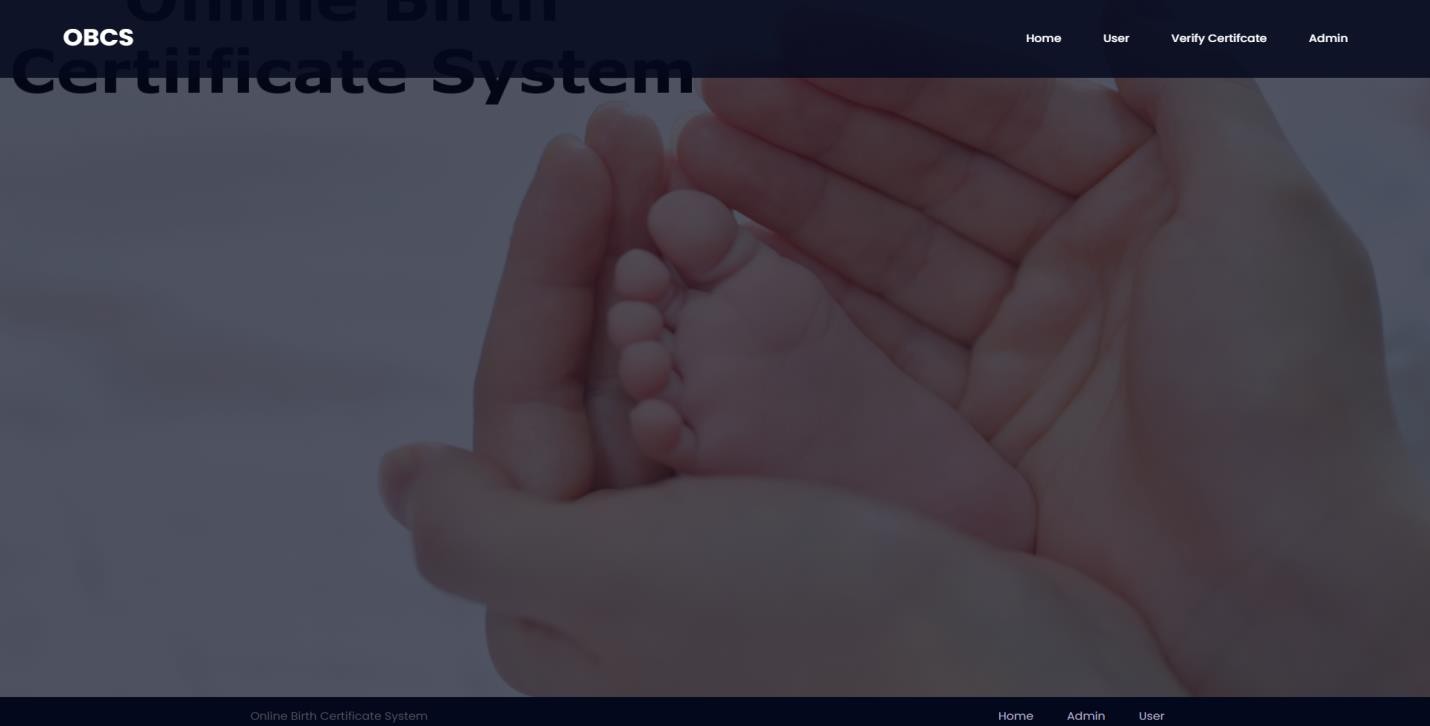
1. **Test for the admin module**
   * **Testing admin login form-**This form is used for log in of administrator of the system. In this form we enter the username and password if both are correct administration page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask the details.
   * **Report Generation:** admin can generate report from the main database.

## INTEGRATION TESTING

In the Integration testing we test various combination of the project module by providing the input.

The primary objective is to test the module interfaces in order to confirm that no errors are occurring when one module invokes the other module.

# EVALUATION

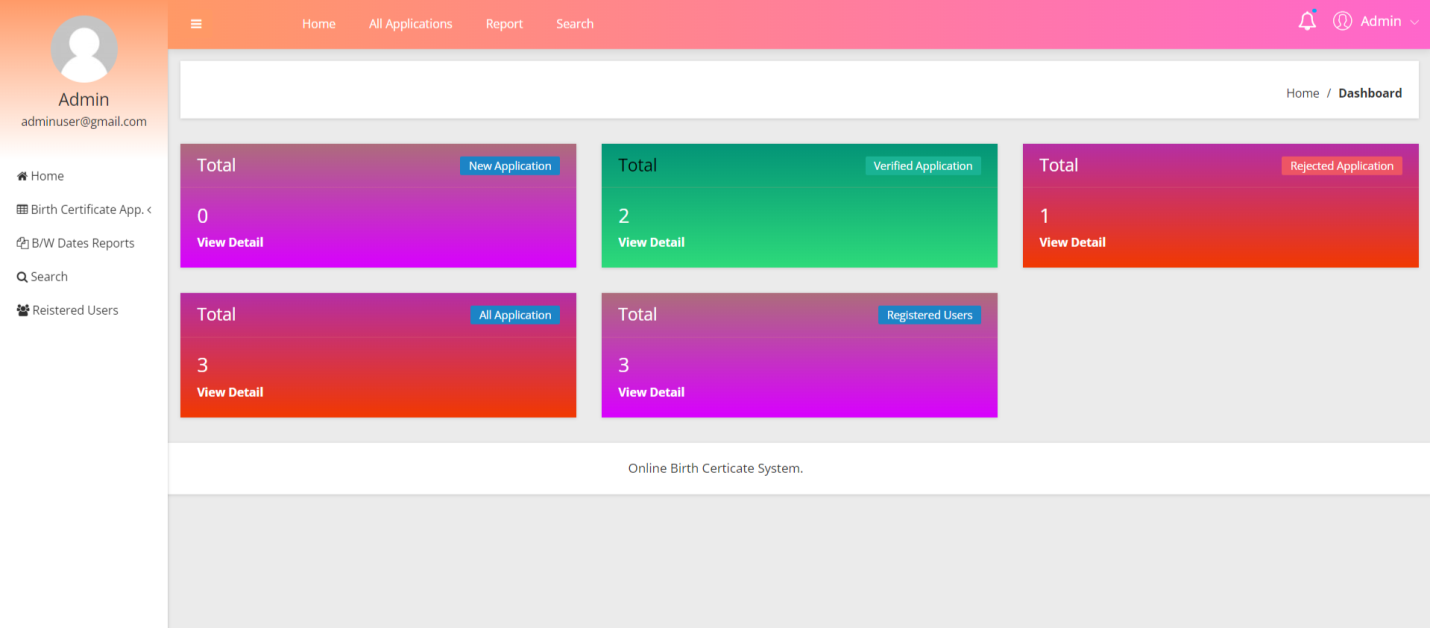
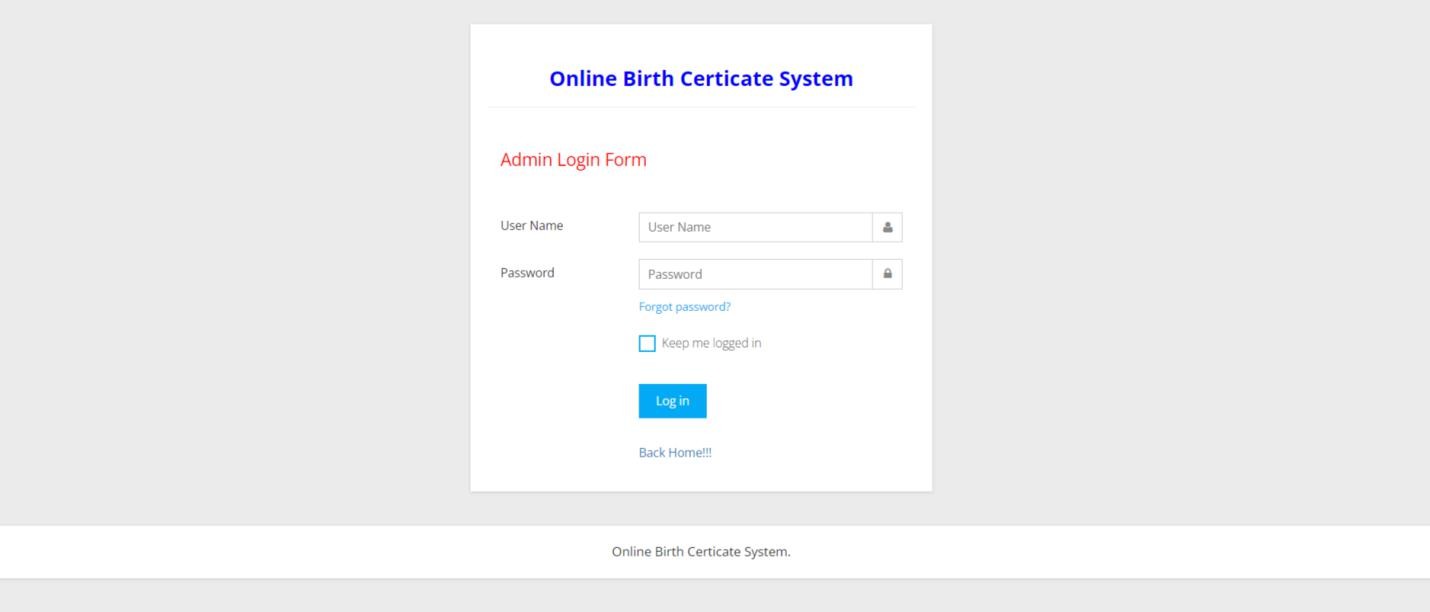


**Project URL:** [**http://localhost/obcs**](http://localhost/obcs)

**Home Page**

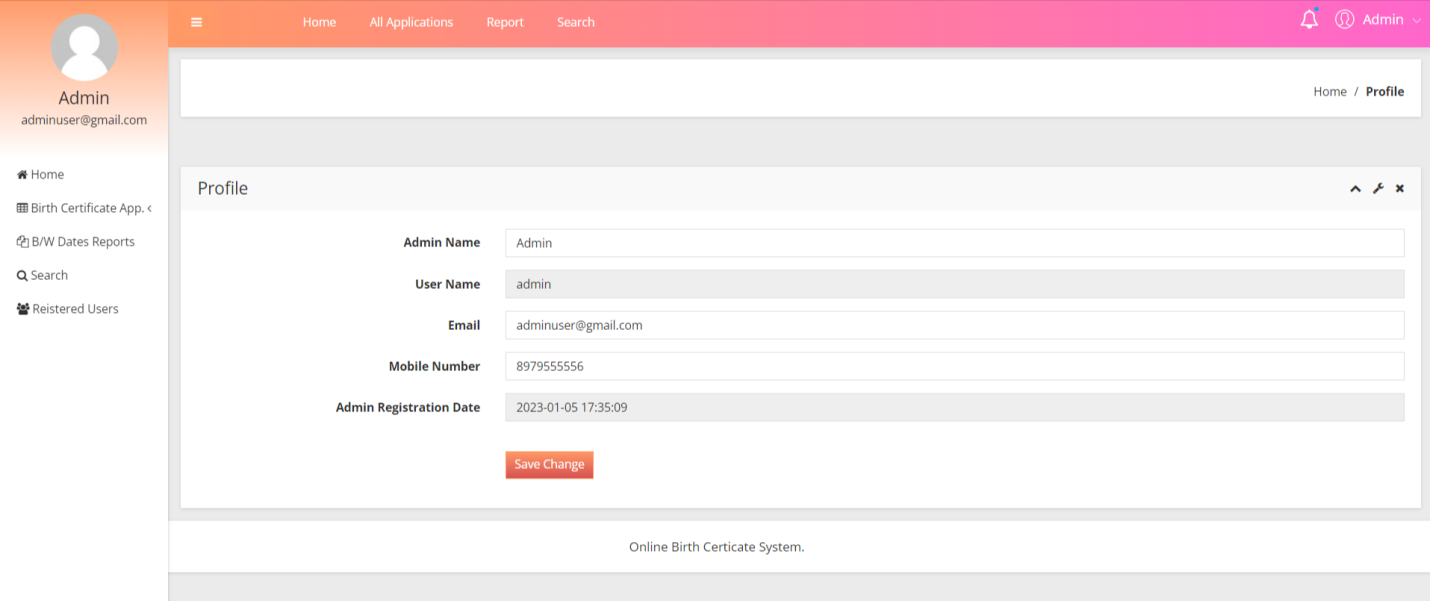
**View Verified Certificate**

**Admin Login Page**



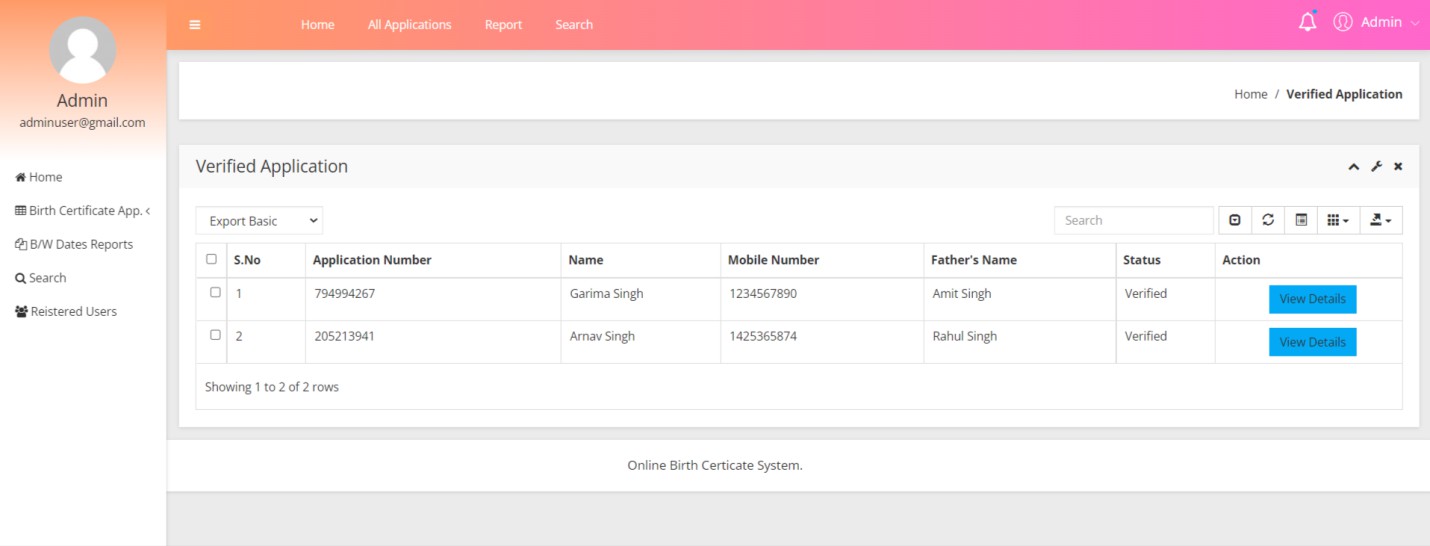
**Dashboard**

**Profile**



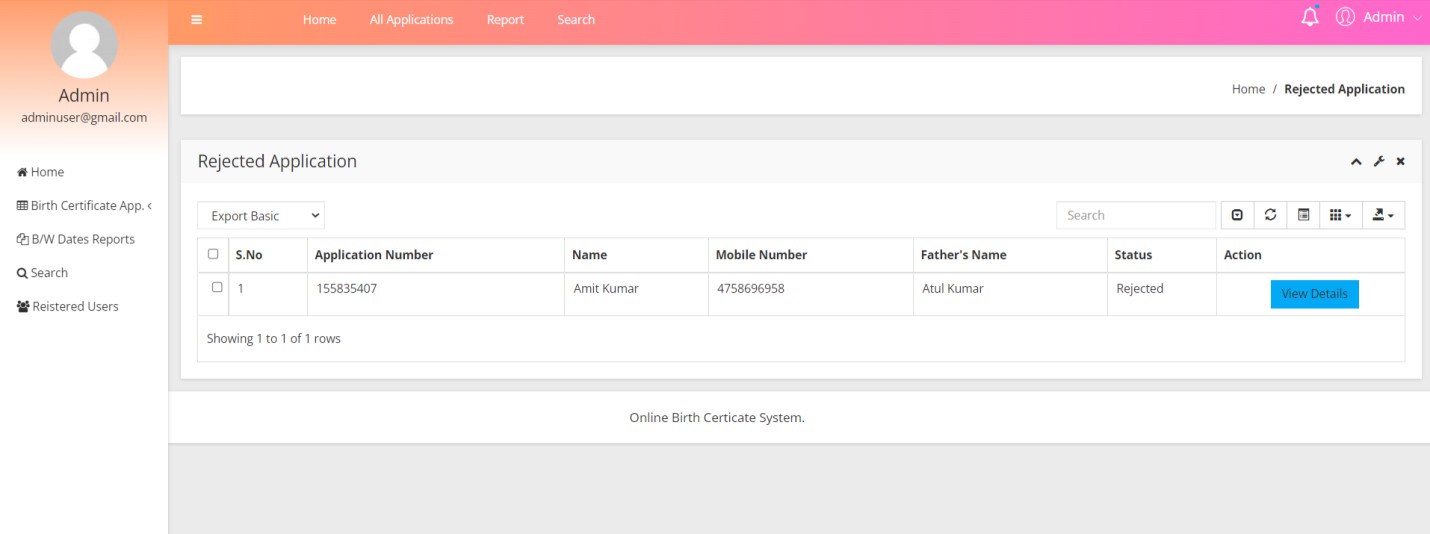
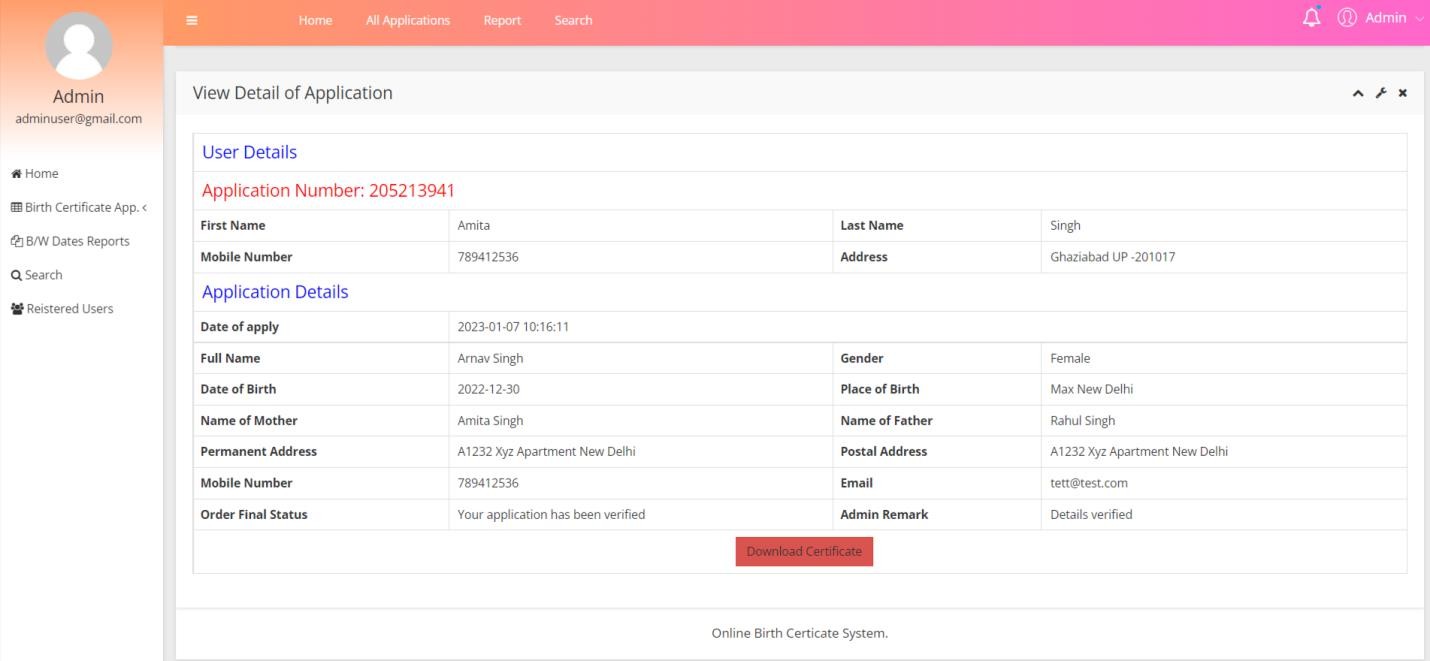
**Change Password**

**New Application**



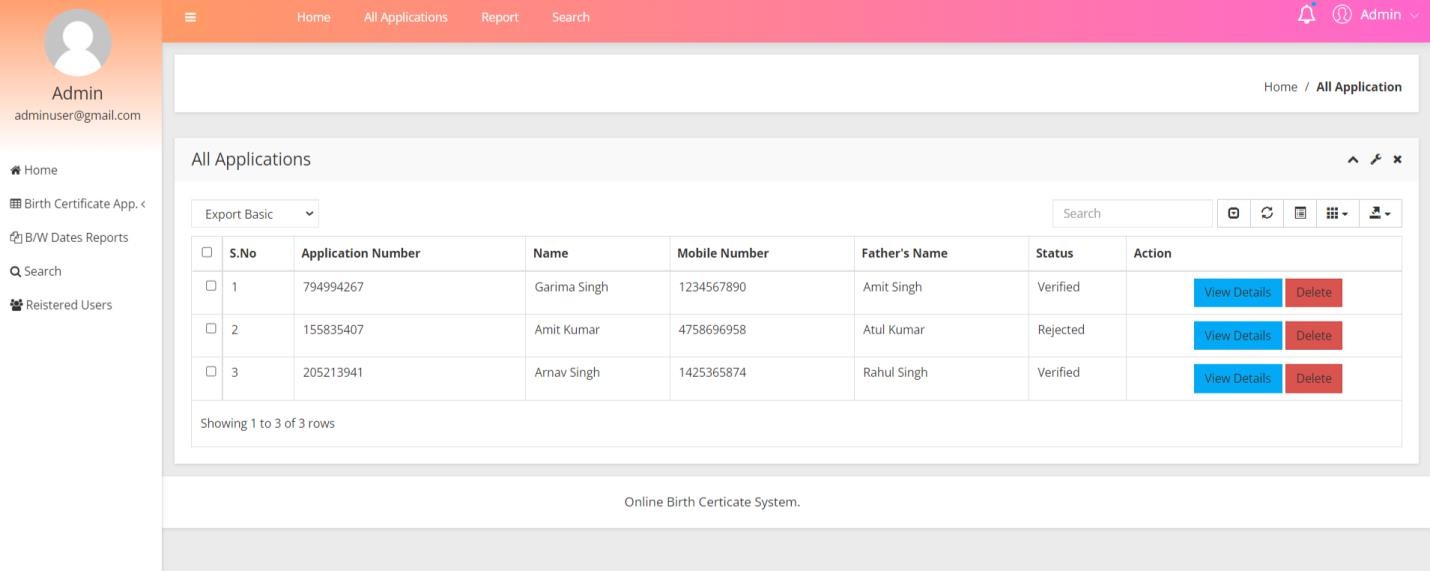
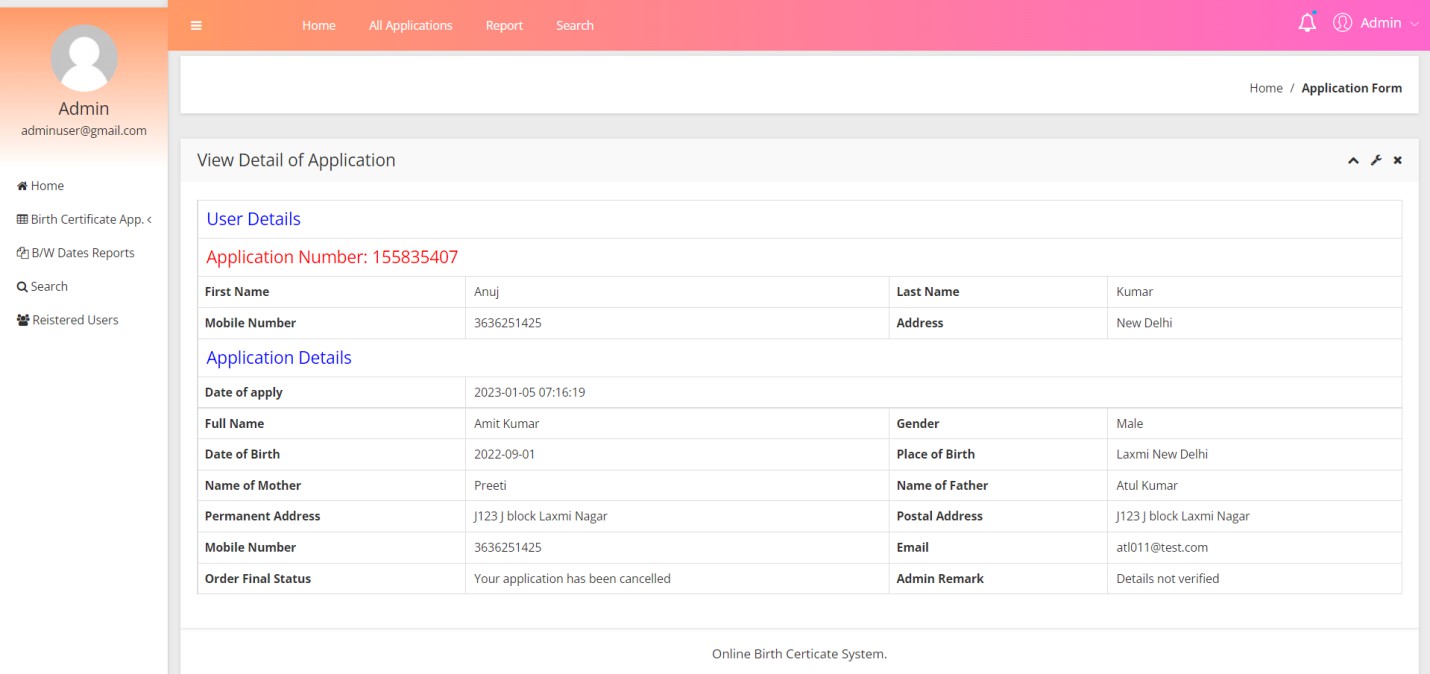
**Verified Application**

**View Verified Application**



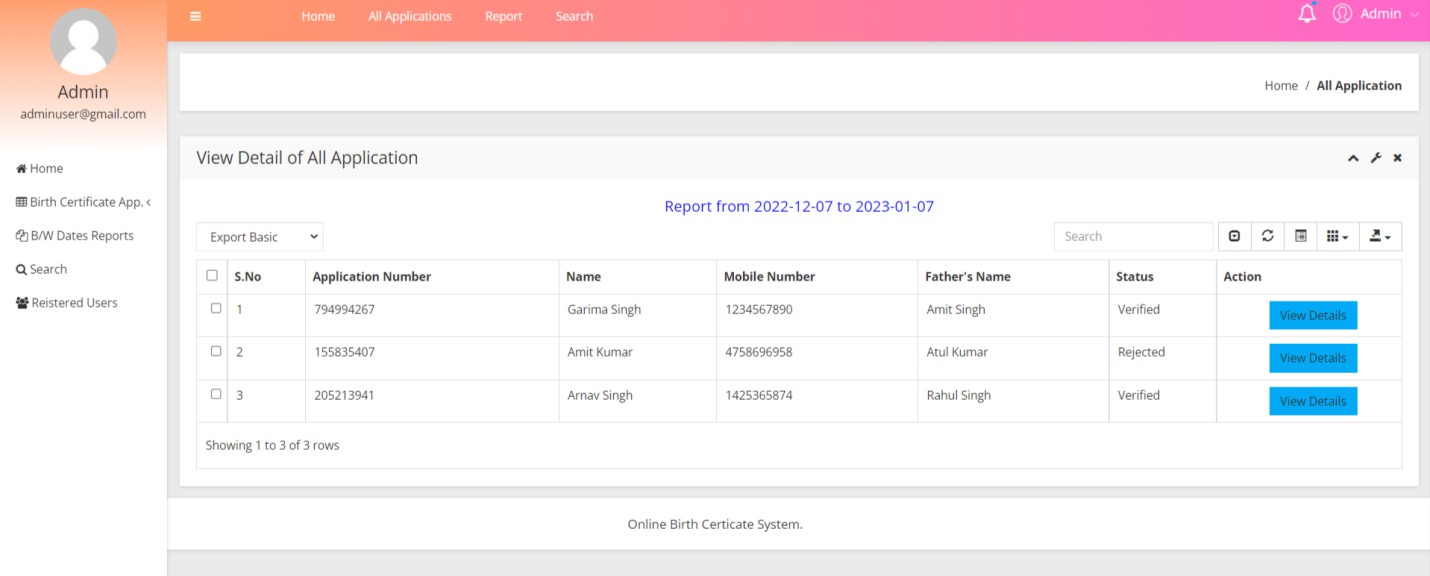
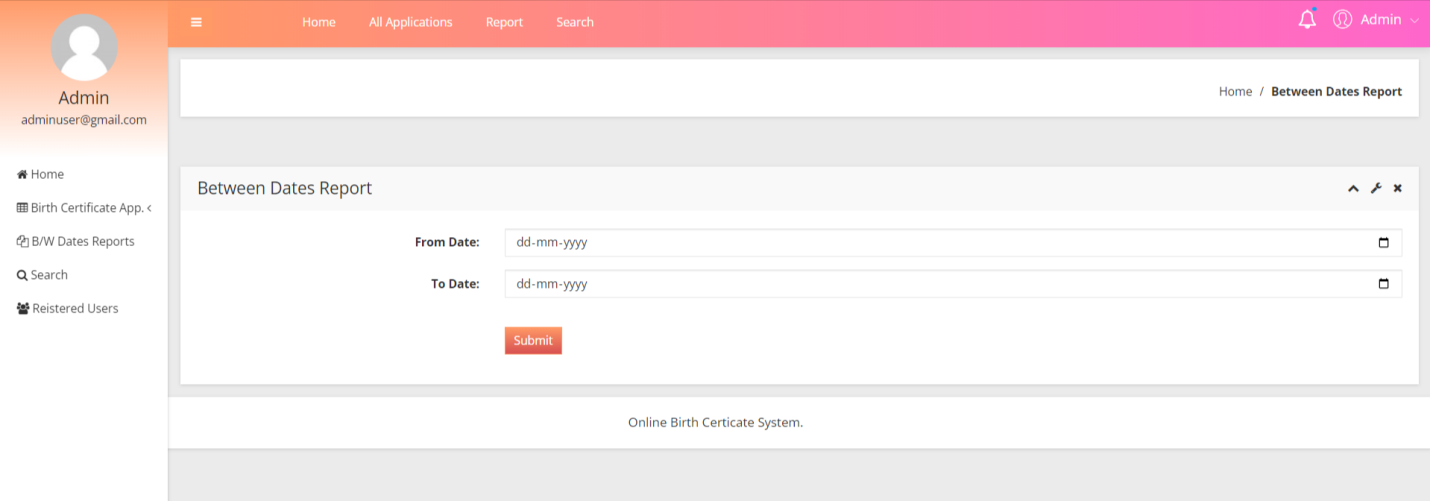
**Rejected Application**

**View Rejected Application**



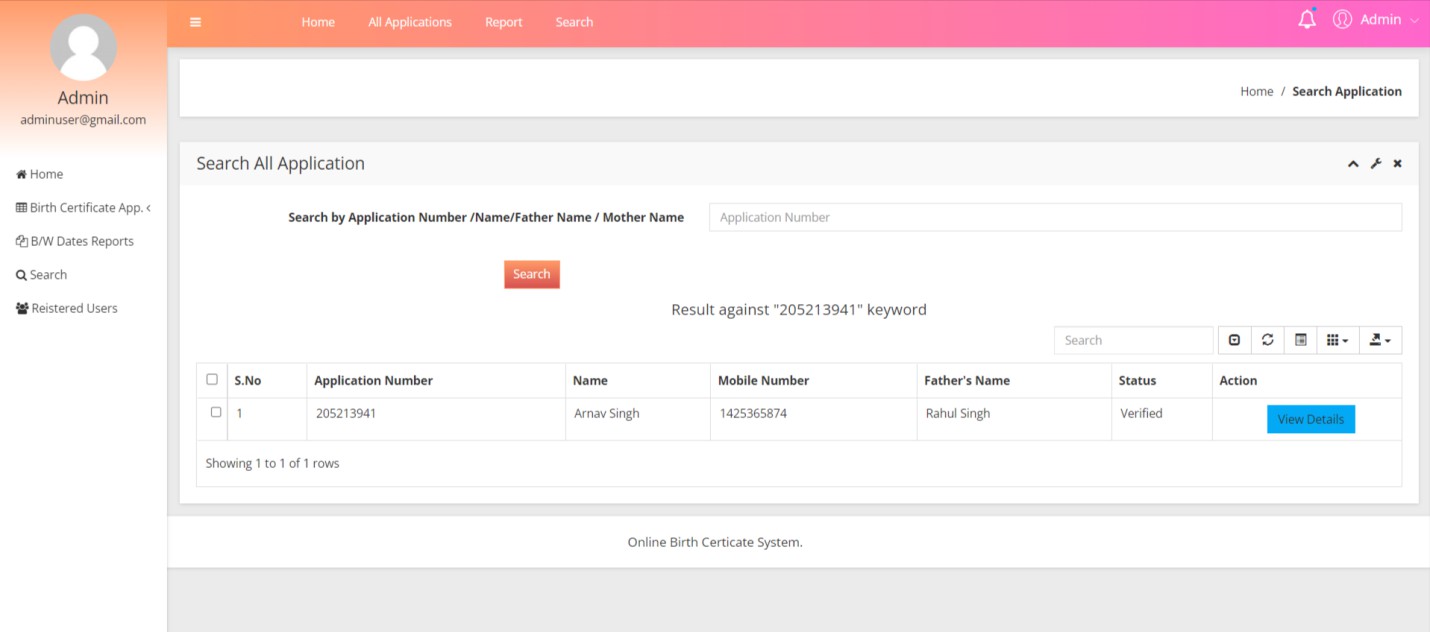
**All Application**

**Report of Application**



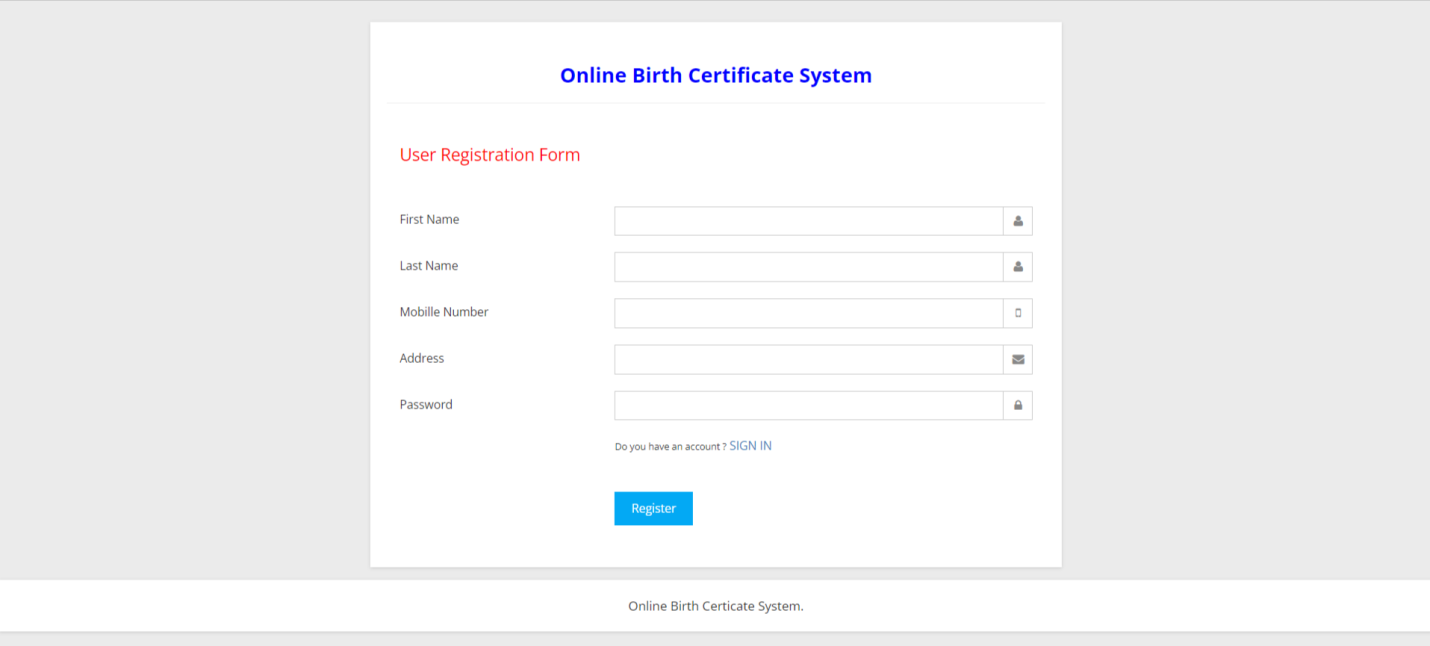
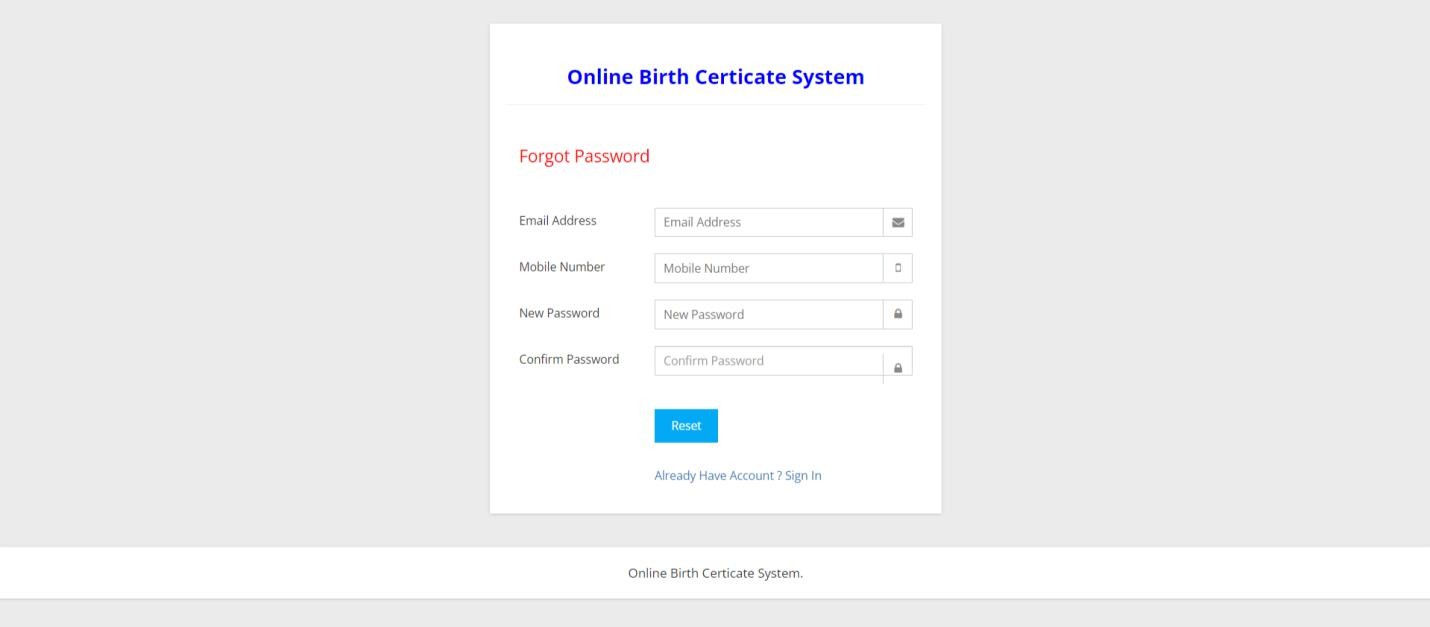
**View Report of Application**

**Search Application**



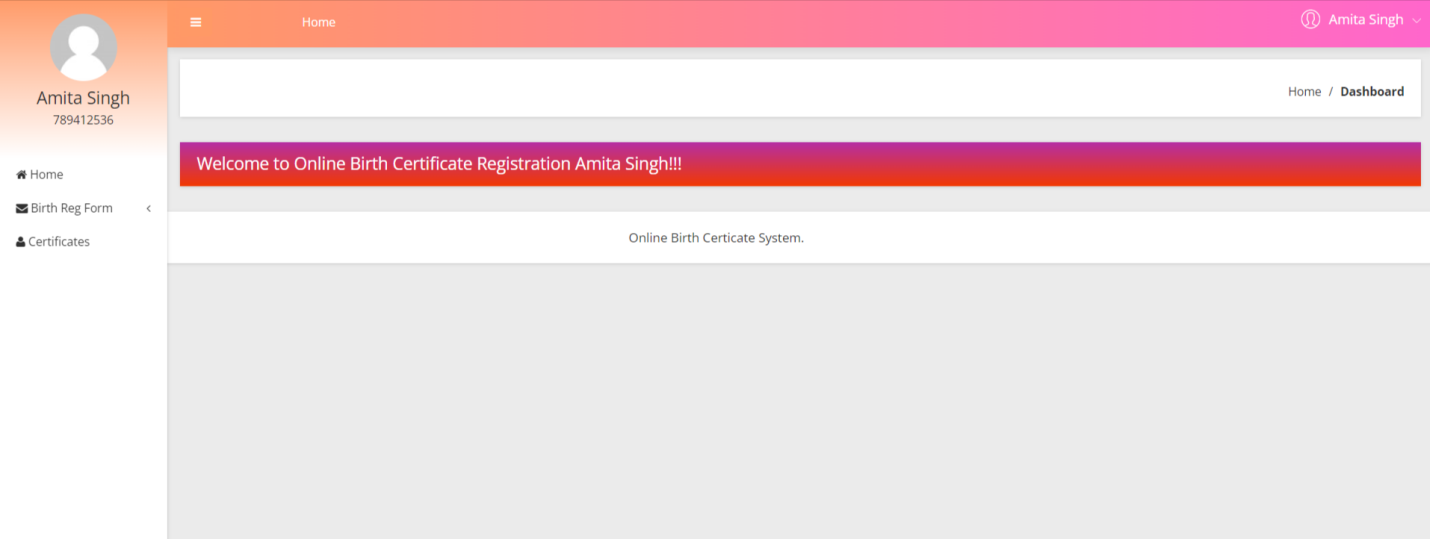
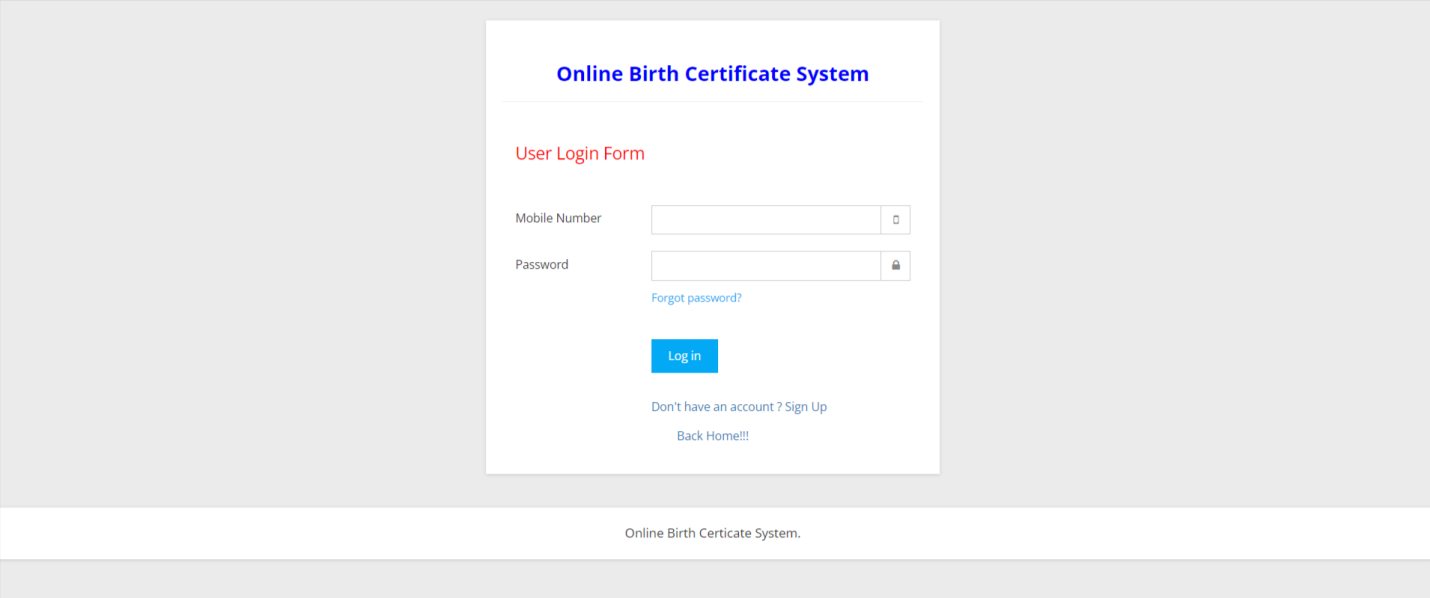
**View Registered Users**

**Forgot Password**



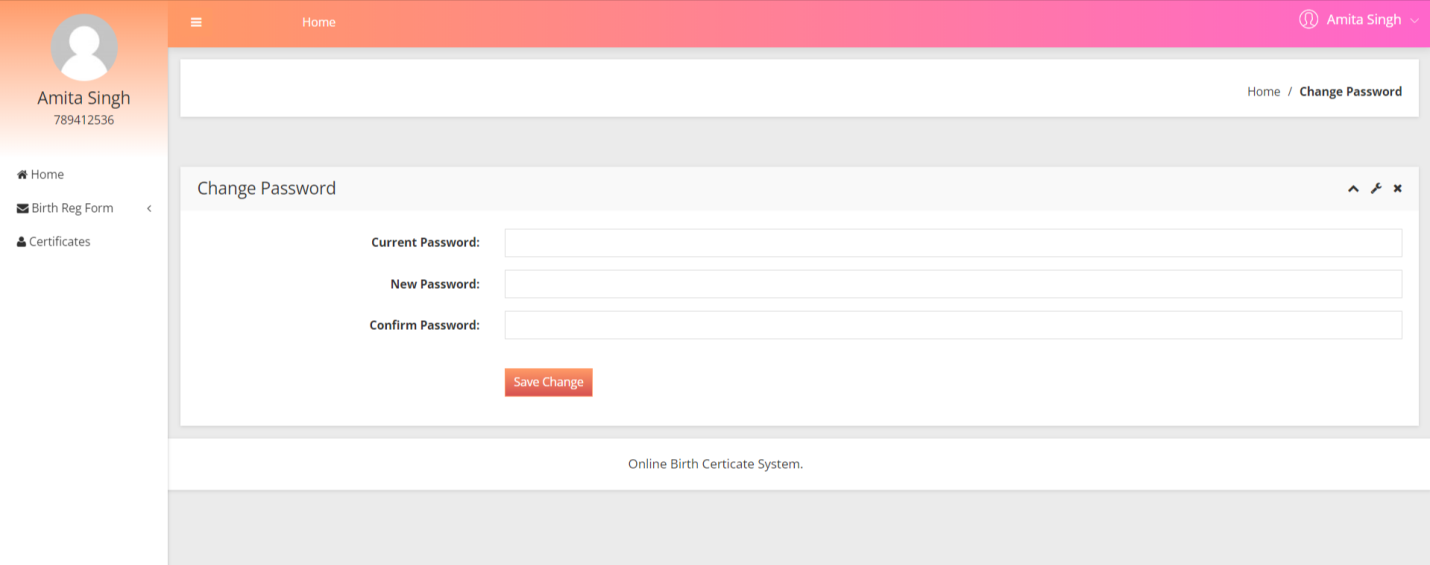
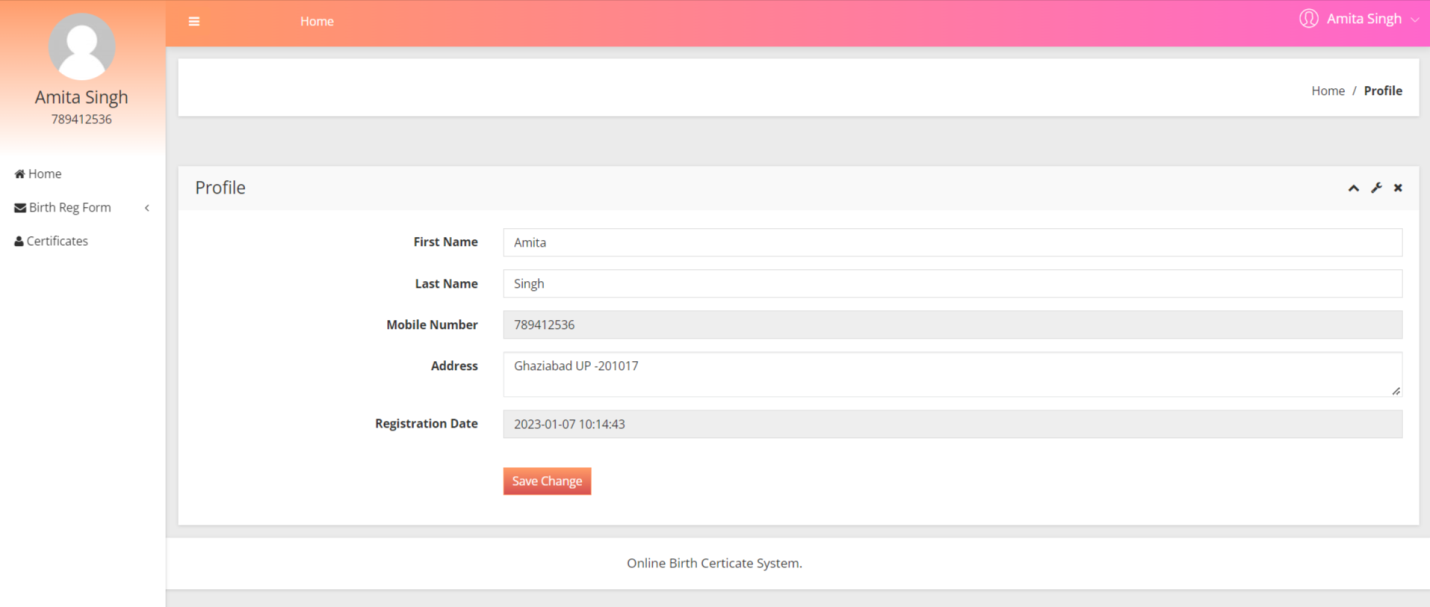
**User Signup Page**

**Signin**



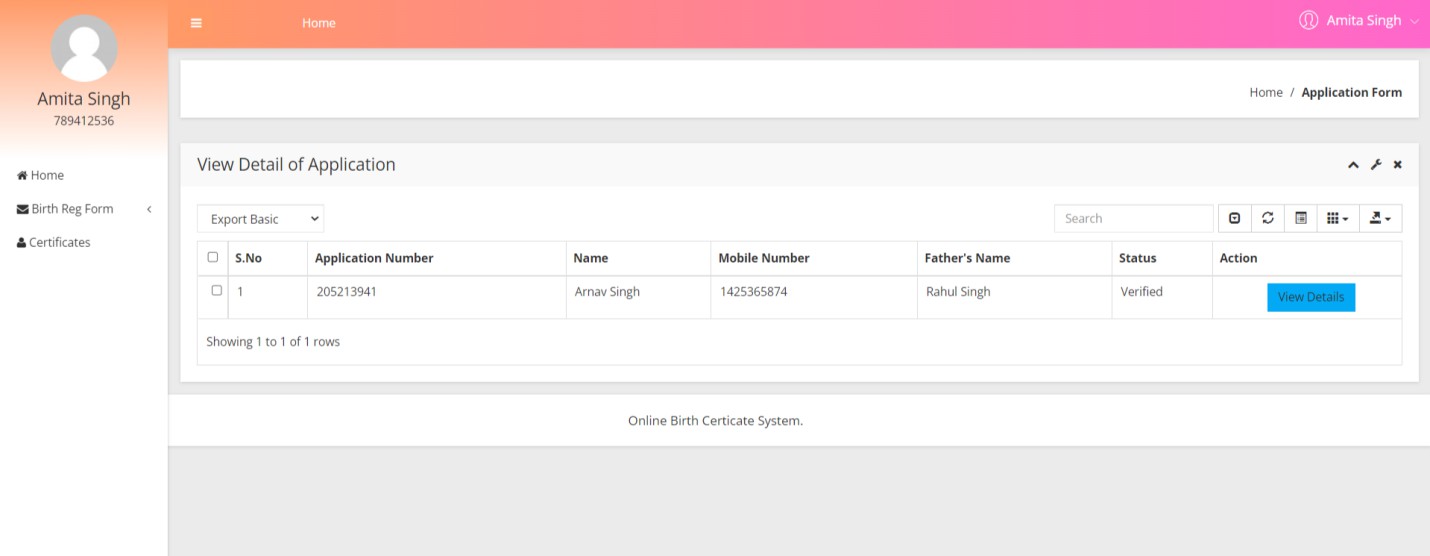
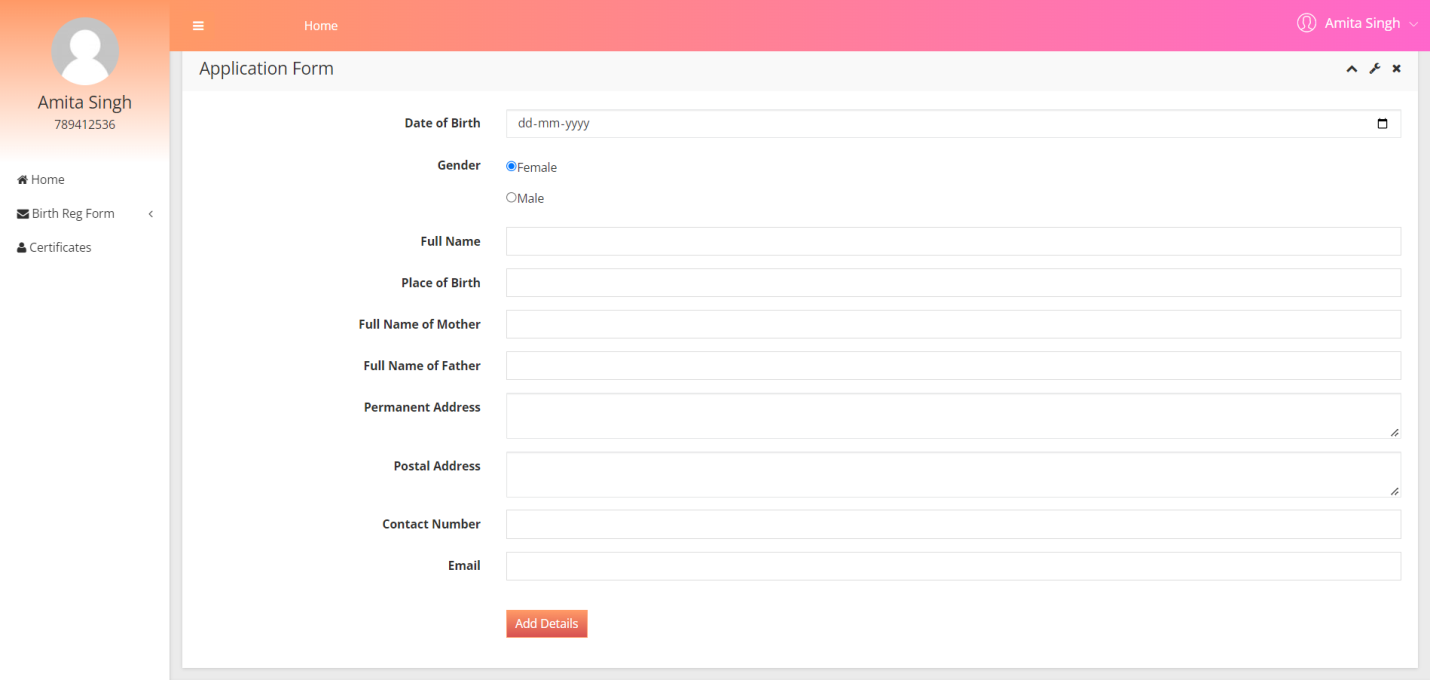
**Dashboard**

**User Profile**



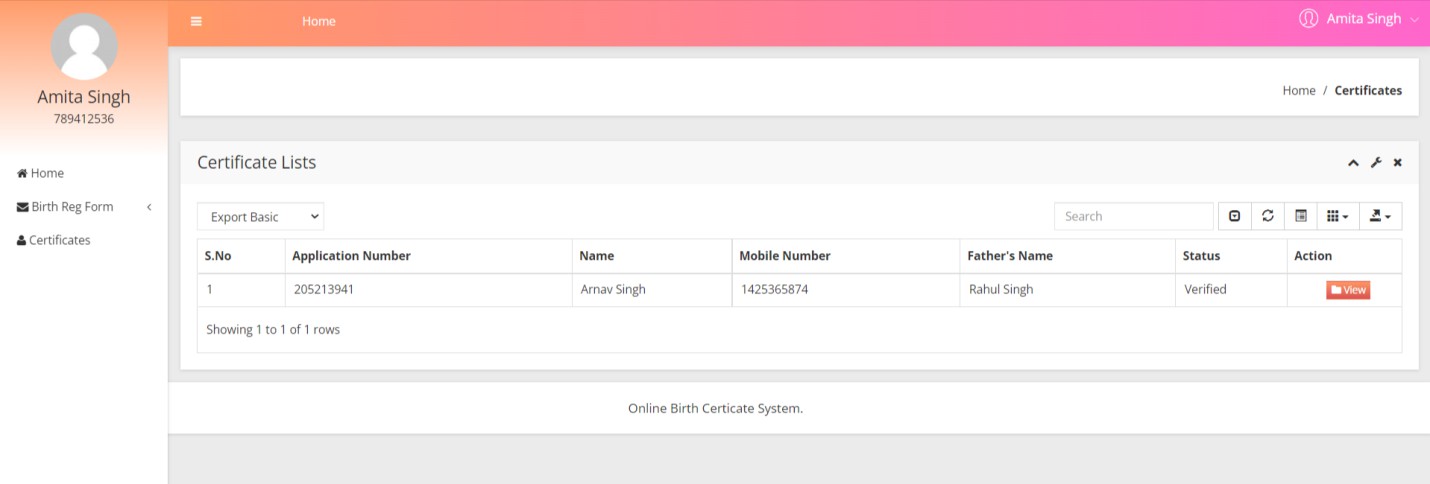
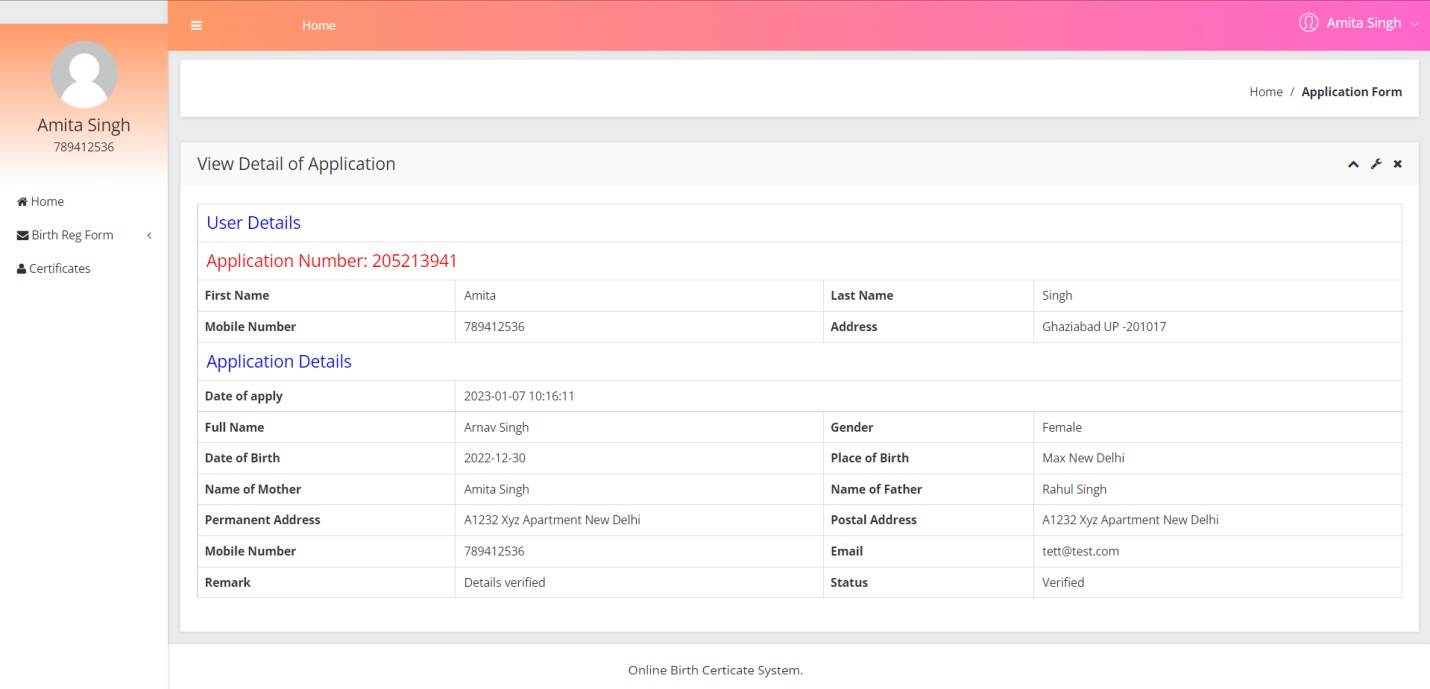
**Change Password**

**Add Application**



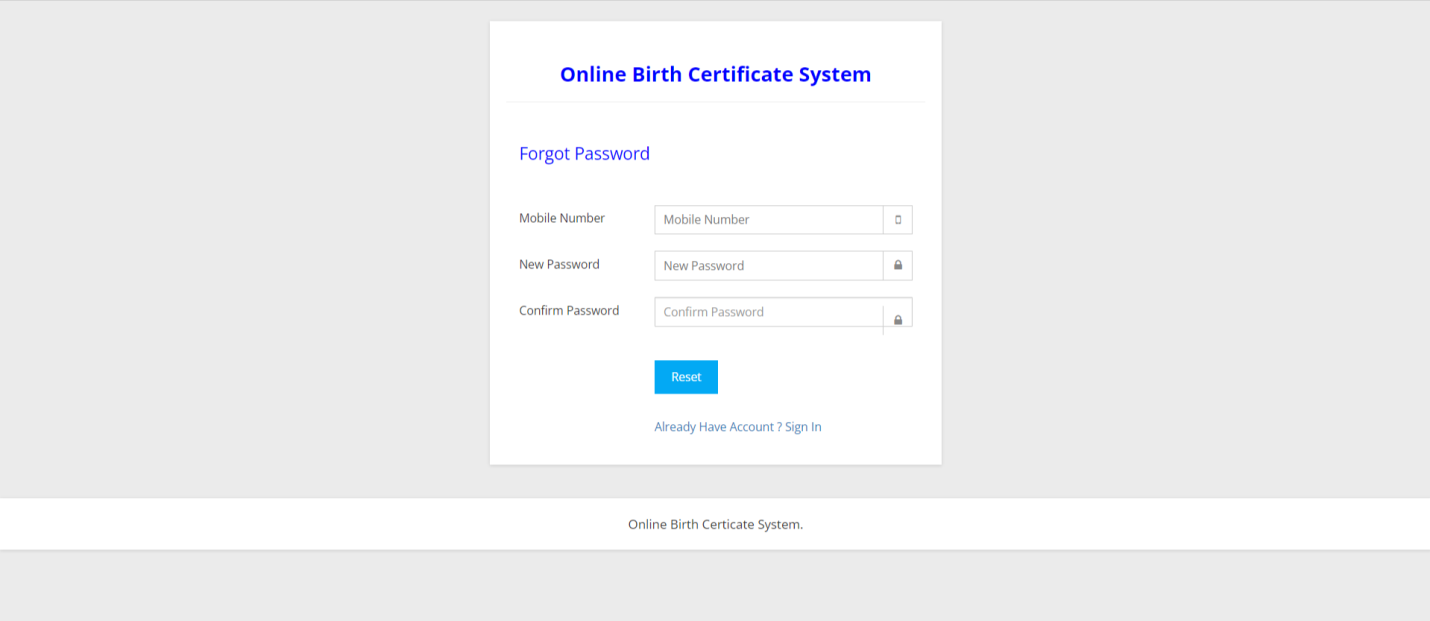
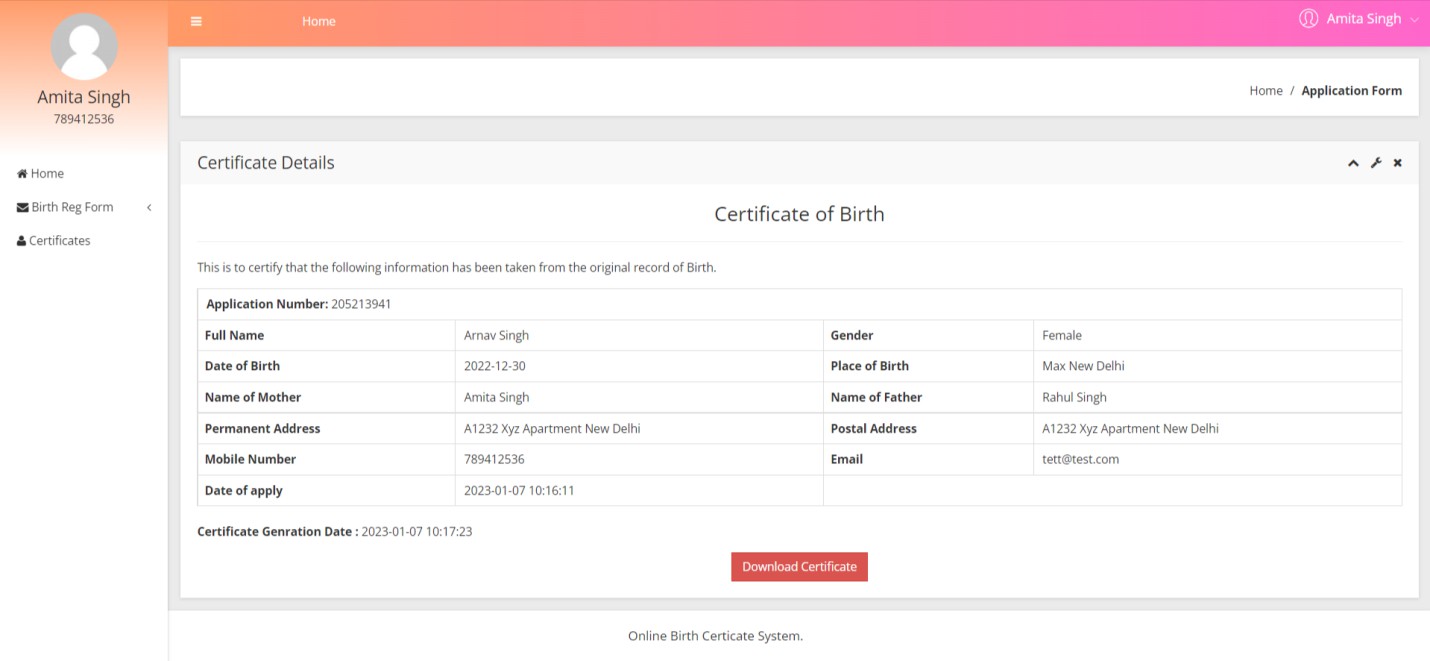
**Manage Applications**

**View Applications Detail**



**Certificate List**

**View Certificate**



**Forgot Password**

# Conclusion:



This Application provides a computerized version of Birth Certificate which helps admin to view data of date of birth of people who reside in country.

It makes entire process online and can generate reports. It has a facility of user’s login where user can fill the application details and send to admin.

The Application was designed in such a way that future changes can be done easily. The following conclusions can be deduced from the development of the project.

* Automation of the entire system improves the productivity.
* It provides a friendly graphical user interface which proves to be better when compared to the existing system.
* It gives appropriate access to the authorized users depending on their permissions.
* It effectively overcomes the delay in communications.
* Updating of information becomes so easier.
* System security, data security and reliability are the striking features.
* The System has adequate scope for modification in future if it is necessary.



# References

**For PHP**

* <https://www.w3schools.com/php/default.asp>
* <https://www.sitepoint.com/php/>
* <https://www.php.net/>



**For MySQL**

* <https://www.mysql.com/>
* [http://www.mysqltutorial.org](http://www.mysqltutorial.org/)

**For XAMPP**

* <https://www.apachefriends.org/download.html>