

A Project Report  
on  
E-SETU  
( E-Governance )

*by*

Suthar Lalit Hajarimal -62  
Hemlata Khandagale -31  
Vinayak Jagdish Walhe -67  
Yash Narayan Bhatia -05

*Under the guidance of*

Prof. Jayshree Mahajan



Department of Information Technology

Smt. Kashibai Navale Sinhgad Institute of Technology & Science, Lonavala

**UNIVERSITY OF PUNE**

**2015-2016**

Sinhgad Technical Education Society,  
Department of Information Technology  
Smt. Kashibai Navale Sinhgad Institute of Technology &  
Science, Lonavala.



---

Date:05/10/2015

## **CERTIFICATE**

This is to certify that,

Suthar Lalit Hajarimal -62

Hemlata Khandagale -31

Vinayak Jagdish Walhe -67

Yash Narayan Bhatia -05

of class T.E IT; have successfully completed their project work on “ *E-Setu* ” at Smt. Kashibai Navale Sinhgad Institute of Technology & Sciences, Lonavala in the partial fulfillment of the Graduate Degree course in T.E at the **Department of Information Technology**, in the academic Year 2015-2016 Semester – I as prescribed by the University of Pune.

Prof. Jayshree Mahajan  
Guide

Prof. A.C. Manpeatil  
Project Co-ordinator

Prof. G.M. Kadam  
Head of the Department

Dr. M.S.Rohokale  
Principal

(Department of Information Technology)

## **Acknowledgements**

“We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. we would like to extend my sincere thanks to all of them.

We are highly indebted to Prof. Jayshree Mahajan for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express Our gratitude towards Our Parents , Teaching and Non-teaching staff for their kind co-operation and encouragement which help me in completion of this project.

We would like to express Our special gratitude and thanks to Prof. Parag Achaliya for giving me such attention and time.

Our thanks and appreciations also go to our colleague in developing the project and people who have willingly helped me out with their abilities.”

Suthar Lalit Hajarimal -62

Hemlata Khandagale -31

Vinayak Jagdish Walhe -67

Yash Narayan Bhatia -05

# **Contents**

<b>Sr. No.</b>	<b>Topic</b>	<b>Page No.</b>
	<b>Acknowledgement</b>	i
	<b>Contents</b>	ii
	<b>List of Tables</b>	iii
	<b>List of Forms</b>	iv
	<b>Acronyms</b>	v
	<b>Abstract</b>	vi
<b>Chapter-1</b>	<b><u>Introduction</u></b>	
1.1	Motivation	1
1.2	Problem Statement	2
1.3	Framework of the proposed work in project	2
1.3.1	Objective	3
<b>Chapter-2</b>	<b><u>Scope</u></b>	
2.1	Scope Introduction	4
2.2	System Description.	4
<b>Chapter-3</b>	<b><u>Software Requirement Specification</u></b>	
3.1	Hardware Requirements	5
3.2	Software Requirements	5
<b>Chapter-4</b>	<b><u>Data Modeling Features</u></b>	
4.1	Data Dictionary	6
<b>Chapter-5</b>	<b><u>Relational Database Design</u></b>	
5.1	Entity-Relationship Diagram	7
5.2.	Architecture Diagram	9
<b>Chapter-6</b>	<b><u>Database Normalization</u></b>	10
<b>Chapter-7</b>	<b><u>Graphical User Interface</u></b>	11
<b>Chapter-8</b>	<b><u>Source Code</u></b>	17
<b>Chapter-9</b>	<b><u>Testing Documents</u></b>	25
<b>Chapter-10</b>	<b><u>Conclusion</u></b>	26

## List of Tables

Table No.	Title	Page No.
5.1	ER diagram symbol	6
5.2	Database Architecture	8

## List of Forms

Figure No.	Title	Page No.
7.1	Home.	11
7.2	Validation	11
7.3	User login.	12
7.4	Admin login.	12
7.5	SignUp	13
7.6	Admin main page.	13
7.7	Caste .	14
7.8	Income	14
7.9	Domicile	15
7.10	Senior	15

## **Acronyms**

CERT	Unique Certificate No.
IDNO	Unique ID Card No.
SETU	<i>“Sevetun Samadhhan”</i> .
FAQ	Frequently Asked Question.

## **Abstract**

The purpose of the project is to make the entire process transparent so that any layman can easily follow the step-by-step instructions to get the desired certificate hassle-free and on time. SETU ensures the process is convenient, faster and fully beneficial to the common man. The SETU website is an initiative by this Collectorate to be more responsive to the people's need using today's Information Technology (IT). SETU has built the bridge. Use it. And help us to help you better. The motto of SETU says it perfectly - 'Sevetun Samadhaan' which means Satisfaction through Service.

The interface should provide the user with all the regular Facilities of Government.

The Interface should provide the user his certificate i.e. Caste ,Domicile . Income ,Senior citizen ID card, etc. The Problem of the users should be appropriately addressed by providing Government services through Internet. Save Time of user and Create a fully Transparency government System.



# Chapter 1

## Introduction

---

**E-SETU** is an integrated web application that handles various E-governance activities. The system can access by every user/employees of the setu through internet connected computers with the aid of his user name and password. Every employee will have a customized home page with his/her profile management facilities. Through links that displays in the home page the user can access different options of the website.

### 1.1 Motivation

This project aims to develop Government Scheme and services Online, which is more helpful for common citizen as this web-application provide services to user without any timewaste And Without extra charges. This Web-application make the government system With transparency and integrated with Right to Information Act.

**E-Governance:** Use of internet by the government to provide its services at the door step of customers, business and other stakeholder.

In E-Governance, government makes best possible use of internet technology to communicate and provide information to common peoples and businessman. Today, electricity, water, phone and all kinds of bills can be paid over the internet. All this is what government and citizens is using and doing. All are dependent on internet and when citizens depends on government internet services all that come is E-Governance.

There are four pillars of E-Governance:-

1. **CONNECTIVITY**:-Connectivity is required to connect the people to the services of the government. There should be a strong connectivity for an effective e-governance.
2. **KNOWLEDGE**: - Here knowledge refers to IT knowledge. Government should employ skill full engineers who can handle the e-governance in an efficient way. These engineers also handle all kind of fault that may occur during the working of e-governance.
3. **DATA CONTENT**: - To share any kind of knowledge or information over the internet, there should be its database. This database should have the data content which is related to government services.
4. **CAPITAL**:-Capital can be on public or private partnership. It refers to money used by government to provide their services or to that sector of the economy based on its operation

## **1.2 Problem Statement**

This project aims to develop an E-Governance system. This system has two user levels Administrator and User. The user means the customer who has the access to the services provided by system. This system will provide a web-based interface to the users, where they can use the system/services offered by Government Authorities. This system enables the users to easily access data from the website.

## **1.3 Framework of the proposed work in project**

Here we aimed to design online web application for issuing the Certificate of various Documents. As to get benefit of various government schemes .So we call it as E-governance Management which delivers via the user-employee interfaces as online system which acting as a Service Provider.

By using this technology we can make fast interaction between the employee of setu and user on any time ( 24x7 ) as they referred in online system. As compared to the manual system, online system is very simple to use and also easy to understand.

### **1.3.1 Objective**

The objectives of our system are as follows:

- To save the time and reduce the paper work.
- To improve the quality of service.
- To provide accuracy in customer data.
- To provide user information.
- To access the system by user (USER/EMPLOYEE) from staying anywhere.

## Scope

---

### 2.1 Scope Introduction

- If we do some changes in designing then this system is also useful for Government Setu Offices.
- If we do some changes in architecture then this system is used for some giant organizations , Government Offices & companies.

### 2.2 System Description

The purpose of this system is to gather information / data of users as well as fill the data of user and provide them an access. Users after login into the system, then only is possible to see their Schemes / document which user can get easily, Notifications of upcoming schemes / policies etc. User / Employee has to login with his Username & Password.

User will get Auto-generated PDF of document after successful submitting form like caste certificate , income certificate , domicile certificate , senior citizen Id card , etc.

Auto-generated PDF of document contain All the information provided by user Based on Document Submitted online.

Pdf of document also contain CERT no which is nothing but Unique certificate no. of each certificate and contain Qrcode which will easily readable form for electronic scanning purpose and for check validity of Certificate.

# System Requirements

---

### 3.1 Software Requirements:

- Windows (32bit & 64bit)
- Apache Tomcat 6.0 or above
- Web Browser
- XAMP
- Diagram Designer
- MongoDB

### 3.2 Hardware Requirements:

- 50 MB HDD Free Space (200 MB free space Recommended)
- 256 MB RAM (1GB or more Recommended)
- Pentium IV Processor (or above)

### Data Modeling Features

---

#### 4.1 Data Dictionary.

Data dictionary is a repository of information describing the data in the database, that is the metadata or the '*data about the data*'.

Data dictionary can be classified as active and passive. Active is always consistent with database structure, maintained automatically and passive system may not be consistent with database structure as changes are initiated by the users.

Data dictionary for E-setu project is :

- Cert no : which indicate unique certificate no.
- Fname : full name of user / customer.
- Uname : username of user / customer / admin.
- Pass : password of user / customer / admin.
- Caste : caste of user / customer for caste certificate.
- Incm : annual income of user / customer for income certificate.
- Doc : Document that of user / customer have for further application.
- Mob : contact no of user / customer / admin.
- Email : email id of user / customer / admin.
- Type : type of certificate .
- Valid : validity information of certificate.

## Relational Database Design

### 5.1 Entity-Relational Diagram

#### 5.1.1 Definition

- A *database* can be modeled as: a collection of entities, relationship among entities.
- An *entity* is an object that exists and is distinguishable from other objects.
- Entities have *attributes*.
- A relationship is an association among several entities.

#### 5.1.2 Symbol








Component name	Symbol	Description
Rectangle		Represent entity set
Ellipses		Represents attributes
Diamonds		Represents relationship set
Lines		Represents link
Double Ellipses		Represents multivalued attributes
Dashed Ellipses		Represents derived attributes
Double Rectangle		Represent weak entity set

Table 5.1 E-R Diagram Symbol.

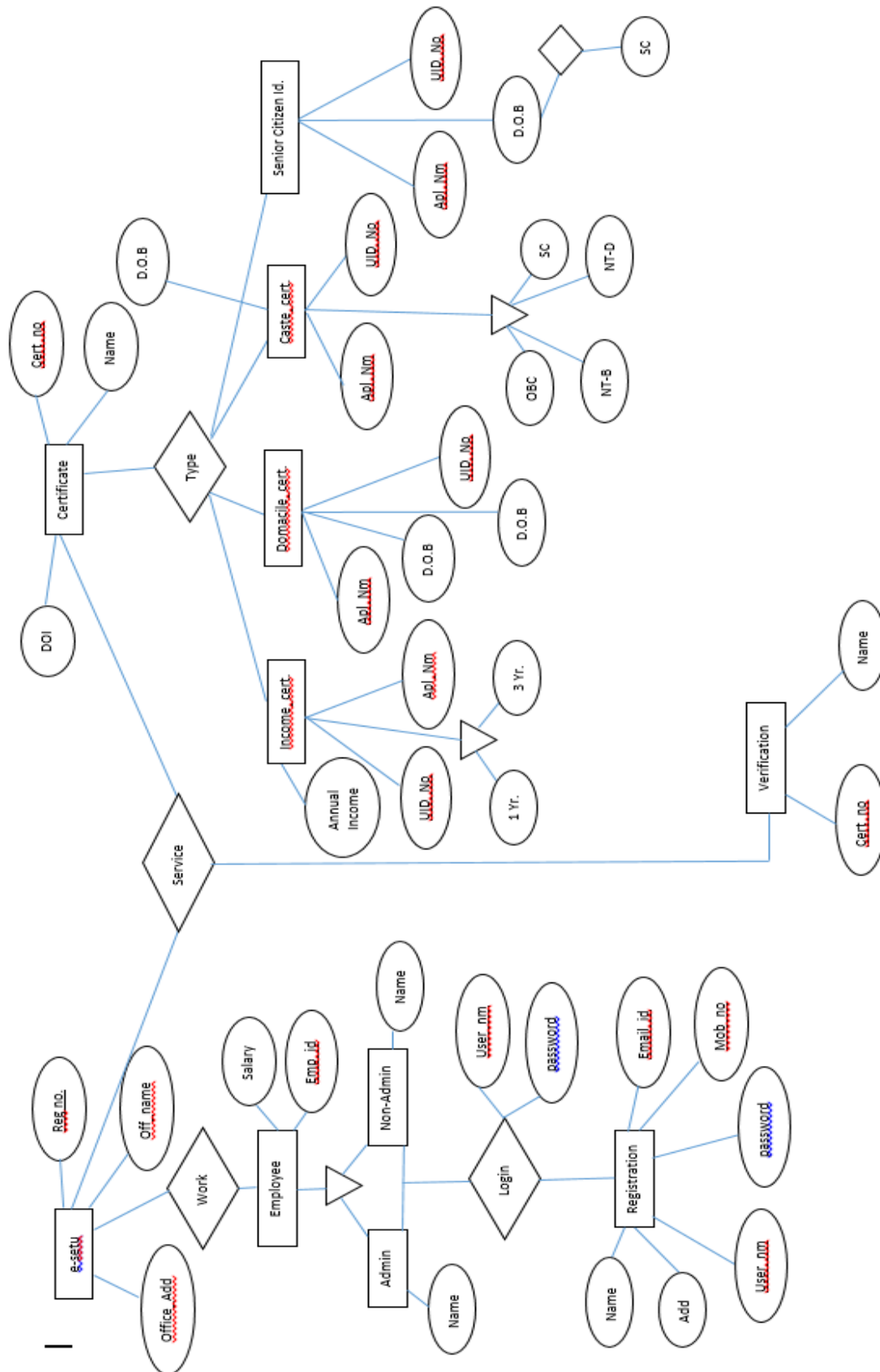


Fig 5.1 E-Setu ER Diagram



## 5.2 Architecture Diagram

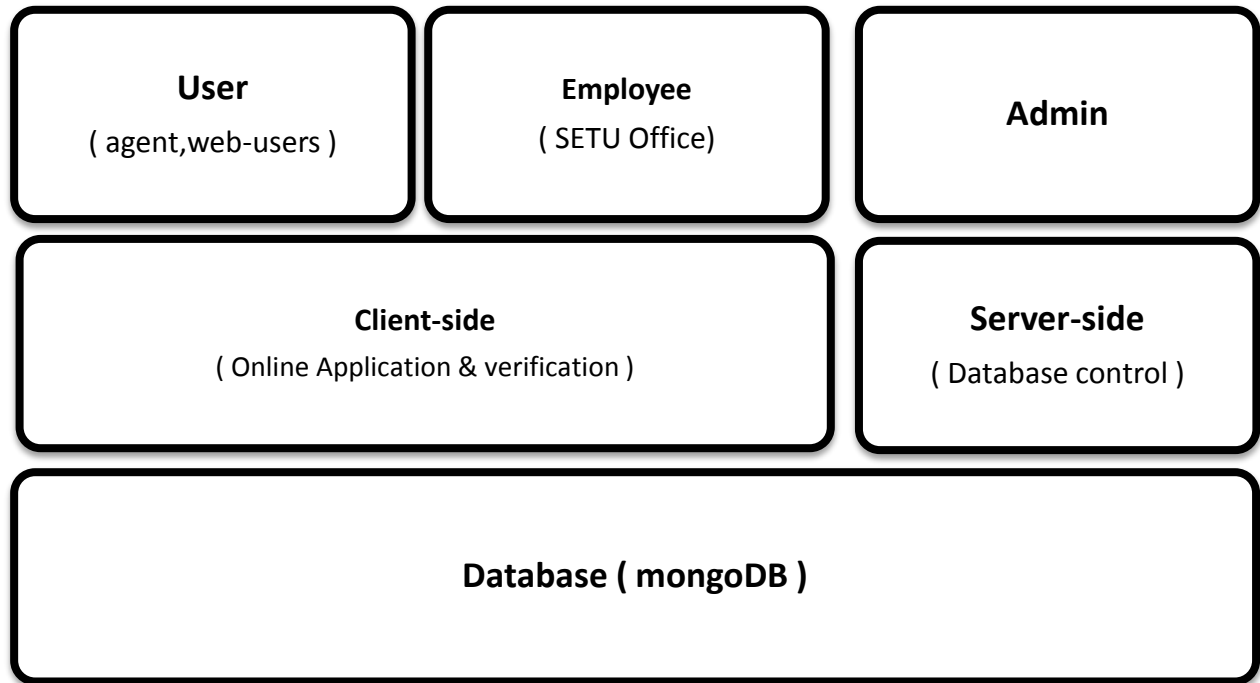


Fig 5.2 Architecture Diagram

# Database Normalization

---

### 6.1 Purpose

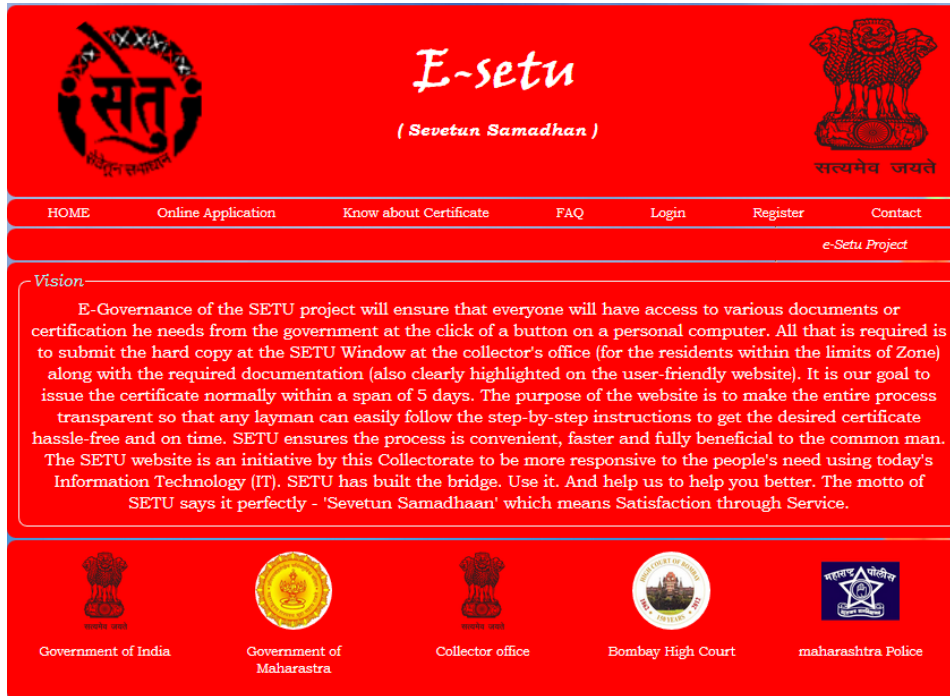
- To minimize redundancy in data.
- Remove insert , delete and update anomaly during database activities.
- Reduce the need to reorganize the data when it is modified or enhanced.
- Normalization reduces a complex process user view to set of small and stable subgroups of fields/relation.
- This process helps to design a logical data model known as conceptual data model.

### 6.2 Type

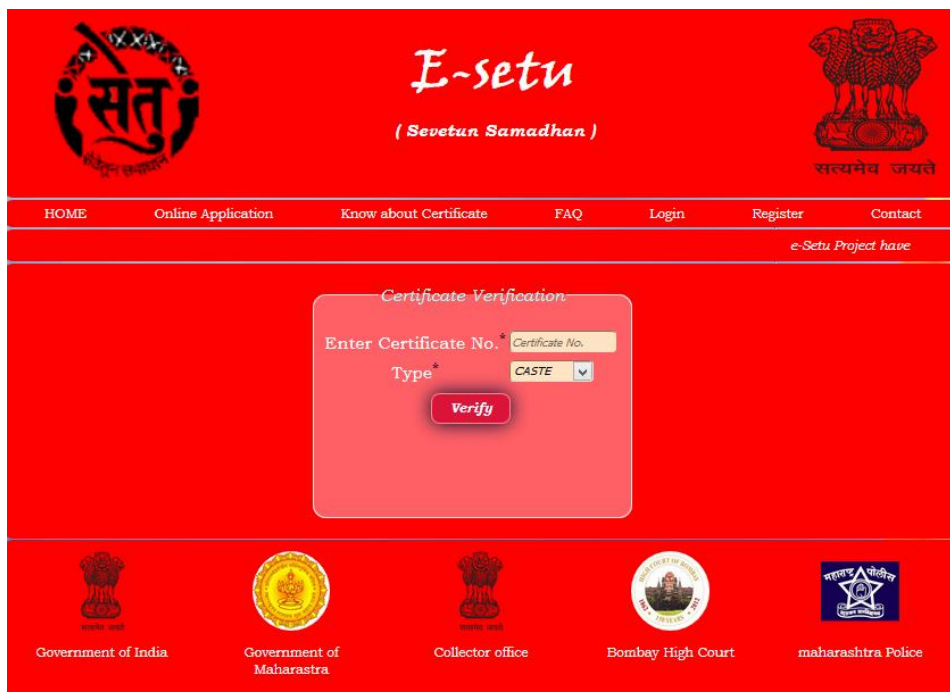
- 1NF ( No repeating group )
- 2NF ( no partial dependencies )
- 3NF ( no transitive dependencies )
- BCNF ( if determinant is candidate key )
- 4NF (no multivalued dependencies )
- 5NF(no join dependencies )

## Graphical User Interface

### 7.1 Home



### 7.2 Validation



## 7.3 Login

### 7.3.1 User login

**E-setu**  
( Sevetun Samadhan )

HOME Online Application Know about Certificate FAQ Login Register Contact

e-Setu Project have received 'SI'

**User Login**

Username \* :

Password \* :

\* is Mandatory

Login Signup

Government of India Government of Maharashtra Collector office Bombay High Court maharashtra Police

### 7.3.2 Admin login

**E-setu**  
( Sevetun Samadhan )

HOME Online Application Know about Certificate FAQ Login Register Contact

e-Setu Project have

**Admin Login**

Username \* :

Password \* :

\* is Mandatory

Login Signup

Government of India Government of Maharashtra Collector office Bombay High Court maharashtra Police

## 7.4 SignUp

**E-setu**  
(Sevetun Samadhan)

HOME Online Application Know about Certificate FAQ Login Register Contact

e-Setu Project have received 'SKOCH AWARD' as India's Best Governance Project - 2014.

**Signup**

Full Name \* :

username \* :

Password \* :

Mobile No :

Email Id :

Gender : ☐ Male ☐ Female

\* is Mandatory

[Submit](#) [Reset](#) [Login](#)

## 7.5 User main page

**E-setu**  
(Sevetun Samadhan)

HOME Online Application Know about Certificate FAQ Login Register Contact

e-Setu Project have received 'SKOCH AWARD' as India's E

Welcome, LALIT HAJARIMAL SUTHAR

**Online Application Form**

[caste certificate.](#)

[Income certificate.](#)

[Senior citizen Id Card.](#)

[Domacile certicate.](#)

[logout](#)

## 7.6 Admin main page



**E-setu**  
(Sevetun Samadhan)

HOME Online Application Know about Certificate FAQ Login Register Contact

e-Setu Project have received 'SKOCH AWARD' as India's Best

Welcome ,slalit360!

**UPDATE**

**DELETE**

**DISPLAY**

**LOGOUT**

**Update Customer Details.**

username \* :  \* is Mandatory

**Update** **Clear**

**Delete Customer Details.**

username \* :  \* is Mandatory

**Delete** **Clear**

## 7.7 Caste form



**Caste Certificate**

Full Name \* :

Address :

Mobile No \* :

Email Id:

Gender : ☒ Male. ☐ Female

Document \* : ☐ Ration card ☐ Income Proof/Receipt ☐ Adhaar card

Caste : ☒ OBC ☐ SC ☐ ST ☐ NT-A ☐ NT-B \* is Mandatory

**Submit** **Reset**

**Home**

## 7.8 Income form

### Income Certificate

Full Name\* :

Address :

Mobile No\* :

Email Id:

Gender : ☒ Male. ☐ Female

Document\* : ☐ Ration card  
☐ Income Proof/Receipt  
☐ Birth Certificate  
☐ Adhaar card

Annual income\* :

Purpose\* :

\* is Mandatory

## 7.9 Domicile form

### Domicile Certificate

Full Name\* :

Address :

Mobile No\* :

Email Id:

Gender : ☒ Male. ☐ Female

Document\* : ☐ Ration card  
☐ Income Proof/Receipt  
☐ Birth Certificate  
☐ Adhaar card

Birth Date\* :

Birth place\* :

\* is Mandatory

## 7.10 Senior form

Senior Citizen ID Application Form

Full Name \* :

Address :

Mobile No :

Email Id:

Gender : ☐ Male ☐ Female

Date Of Birth \* :  -  -

Document \* : ☐ Birth Certificate ☐ Addhar card

\* is Mandatory

## 7.11 Database GUI ( mongoChef )

MongoChef - 3T Software Labs - Non-Commercial License

File Edit Database Collection Index Document GridFS View Help

Connect Aggregate IntelliShell Export Import Users Roles Feedback

(localhost:27017)

- admin
- customer
- local
- setu
  - admin
  - caste
  - customer
  - dom
  - income
  - senior
  - system.indexes

localhost:27017 setu caste

Query {}

Projection {} Sort {}

Table View

_id	cert_no	name	mob	email	gender	add	caste	doc	date
56125f5a7ff36...	02K07X	HEMLATA A	8446569018	hemlata7509@...	male	idsdjhghajafajskq	NT-A	[ 3 elements ]	[ 11 fields ]
5612ace87ff36...	668312	LALIT HAJARIMAL SUTHAR	9403240013	slalit360@gma...	male	lonavala,pune	OBC	[ 3 elements ]	[ 11 fields ]

Operations 1 item selected

01:57 AM 06/10/2015



# Source Code

---

## 8.1 Language Used

### 8.1.1 Php

#### What is PHP?

- PHP stands for PHP: Hypertext Pre-processor
- PHP is a widely-used, open source scripting language.
- PHP scripts are executed on the server.
- PHP is free to download and use.

#### What is PHP File?

- PHP files can contain text, HTML, JavaScript code, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have a default file extension of ".php"

#### What Can PHP Do?

- PHP can generate dynamic page content.
- PHP can create, open, read, write, and close files on the server.
- PHP can collect form data.
- PHP can send and receive cookies.
- PHP can add, delete, and modify data in your database.
- PHP can restrict users to access some pages on your website.
- PHP can encrypt data.

#### Why PHP?

- PHP runs on different platforms (Windows, Linux, UNIX, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP has support for a wide range of databases
- PHP is free.
- PHP is easy to learn and runs efficiently on the server side

### 8.1.2 Javascript

#### What is JavaScript ?

- JavaScript was designed to add interactivity to HTML pages.
- JavaScript is a scripting language.
- A scripting language is a lightweight programming language.
- JavaScript is usually embedded directly into HTML pages JavaScript is an interpreted language (means that scripts execute without preliminary compilation).
- Everyone can use JavaScript without purchasing a license.

#### What can a JavaScript do ?

- JavaScript can put dynamic text into an HTML page - A JavaScript statement like this: document. Write ("

# " + name + "</h1>") can write a variable text into an HTML page.
- JavaScript can react to events - A JavaScript can be set to execute when something happens, like when a page has finished loading or when a user clicks on an HTML element.
- JavaScript can read and write HTML elements - A JavaScript can read and change the content of an HTML element.
- JavaScript can be used to validate data - A JavaScript can be used to validate form data before it is submitted to a server. This saves the server from extra processing.
- JavaScript can be used to detect the visitor's browser - A JavaScript can be used to detect the visitor's browser, and - depending on the browser - load another page specifically designed for that browser.
- JavaScript can be used to create cookies - A JavaScript can be used to store and retrieve information on the visitor's computer.

### 8.1.3 Database ( mongoDB )

**MongoDB** is a cross-platform, document oriented database that provides, high performance, high availability, And easy scalability. MongoDB works on concept of collection and document.

**Database** is a physical container for collections. Each database gets its own set of files on the file system. A single MongoDB server typically has multiple databases.

**Collection** is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database. Collections do not enforce a schema. Documents within a collection can have different fields.

### **Why should use MongoDB?**

- Document Oriented Storage : Data is stored in the form of JSON style documents
- Index on any attribute
- Replication & High Availability
- Auto-Sharding
- Rich Queries
- Fast In-Place Updates
- Professional Support By MongoDB

### **Where should use MongoDB ?**

- Big Data
- Content Management and Delivery
- Mobile and Social Infrastructure
- User Data Management
- Data Hub

### **MongoDB Advantages:**

Any relational database has a typical schema design that shows number of tables and the relationship between These tables. While in MongoDB there is no concept of relationship.

### **Advantages of MongoDB over RDBMS:**

- Schema less: MongoDB is document database in which one collection holds different documents.
- Number of fields, content and size of the document can be differing from one document to another.
- Structure of a single object is clear
- No complex joins
- Deep query-ability. MongoDB supports dynamic queries on documents using a document-based query language
- that's nearly as powerful as SQL
- Tuning
- Ease of scale-out: MongoDB is easy to scale
- Conversion / mapping of application objects to database objects not needed
- Uses internal memory for storing the (windowed) working set, enabling faster access of data

### **MongoDB Data types:**

- Integer: This type is used to store a numerical value. Integer can be 32 bit or 64 bit depending upon your server.
- String: This is most commonly used data type to store the data. String in mongodb must be UTF-8 valid.
- Boolean: This type is used to store a Boolean (true/ false) value.
- Double: This type is used to store floating point values.
- Min/ Max keys: This type is used to compare a value against the lowest and highest BSON elements.
- Arrays: This type is used to store arrays or list or multiple values into one key.
- Timestamp: c time stamp. This can be handy for recording when a document has been modified or added.
- Object: This data type is used for embedded documents.
- Null: This type is used to store a Null value.
- Symbol: This data type is used identically to a string however, it's generally reserved for languages that use a specific symbol type.
- Date: This data type is used to store the current date or time in UNIX time format. You can specify your own date time by creating object of Date and passing day, month, year into it.
- Object ID: This data type is used to store the document's ID.
- Binary data: This data type is used to store binary data.
- Code: This data type is used to store java script code into document.
- Regular expression : This data type is used to store regular expression

## **8.2 Connectivity code**

```
<?php
    $con=new MongoClient();
    $db=$con->setu;
    $collection=$db->customer;
?>
```

## 8.3 Insert code

```
<?php if(isset($_POST['submitform']))
{ $con=new MongoClient();
  $db=$con->setu;
  $collection=$db->customer;
  $uname=null;
  $pass =null;
  $mob =null;
  $name =null;
  $email =null;
  $gender =null;
  $uname=$_POST['uname'];
  $pass=$_POST['pname'];
  $mob=$_POST['mob'];
  $name=strtoupper($_POST['fname']);
  $email=$_POST['email'];
  $gender=$_POST['sex'];
  $qry =
array("uname"=>$uname,"pass"=>$pass,"name"=>$name,"mob"=>$mob,"email"=>$em
ail,"gender"=>$gender);
  $qry1 = array("uname"=>$uname);
  $yes=$collection->findOne($qry1);
  if(!$yes)
  {
    $result=$collection->insert($qry);
    if(isset($result))
      { echo "<script type='text/javascript'>alert('You successfully
Registered !')</script>"; }
    else
      { echo "<script type='text/javascript'>alert('! Sorry You Not
Registered !')</script>"; }
  }
  else
    { echo "<script type='text/javascript'>alert('Username Already
exist!')</script>"; }
  $con->close();
}
?>
```

## 8.4 Update code

```
<?php
if(isset($_POST['submitform']))
{ $con=new MongoClient();
  $db=$con->setu;
  $collection=$db->customer;
  $uname=$_POST['uname'];
  $pass=$_POST['pname'];
  $mob=$_POST['mob'];
  $name=strtoupper($_POST['name']);
  $email=$_POST['email'];
  $gender=$_POST['sex'];
  $qry1=array("uname"=>$name);
  $yes1=$collection->findOne($qry1);

  if(!$yes1)
  {
    $result=$collection->update
(array("uname"=>$yes),array('$set'=>array("uname"=>$uname,"pass"=>$pass,"name"=>
$name,"mob"=>$mob,"email"=>$email,"gender"=>$gender)));

    if($result)
      { echo "<script type='text/javascript'>alert('Data Updated
!')</script>"; }

    else
      { echo "<script type='text/javascript'>alert('! Update
failed!')</script>"; }
  }

  else
    { echo "<script type='text/javascript'>alert('Username Already
exist!')</script>"; }

  $con->close();
}

?>
```

## 8.5 Delete code

```
<?php
$result2=null;
$y=null;
if(isset($_POST["delete"]))
{
    $uname=$_POST["uname"];
    $con=new MongoClient();
    $db=$con->setu;
    $collection=$db->customer;
    $qry2 = array("uname"=>$uname);
    $y=$collection->findOne($qry2);
    if($y)
    {
        $result2=$collection->remove($qry2);
        if($result2)
        {
            echo "<script>window.alert('Data removed');</script>";
        }
        else
        {
            echo "<script>window.alert('Not removed. '); </script>";
        }
    }
    else
    { echo "<script>window.alert('Data Doesn't Exist.');"; }
}
?>
```

## 8.6 Display

```
<?php
$data = "<table style='border:1px solid red;";
$data .= "border-collapse:collapse' border='1px'>";
$data .= "<thead>";
$data .= "<tr>";
$data .= "<th>Name</th>";
```

```

$data .= "<th>username</th>";
$data .= "<th>Password</th>";
$data .= "<th>mobile no.</th>";
$data .= "<th>Email</th>";
$data .= "<th></th>";
$data .= "<th>Additional Comments</th>";
$data .= "</tr>";
$data .= "</thead>"; $yname = $staff = $cleanliness = $value = $bagain = $rfriend = $acomm = "";
$data .= "<tbody>";

```

```

try{
    $db = $conn->setu;
    $collection = $db->customer;
    $cursor = $collection->find();
    foreach($cursor as $document){
        $data .= "<tr>";
        $data .= "<td>" . $document["fname"] . "</td>";
        $data .= "<td>" . $document["uname"] . "</td>";
        $data .= "<td>" . $document["pass"] . "</td>";
        $data .= "<td>" . $document["mob"] . "</td>";
        $data .= "<td>" . $document["email"] . "</td>";
        $data .= "<td>" . $document["gender"] . "</td>";
        $data .= "</tr>";
    }
    $data .= "</tbody>";
    $data .= "</table>";
    echo $data;

} catch(MongoException $mongoException){
    print $mongoException;
    exit;
}
?>

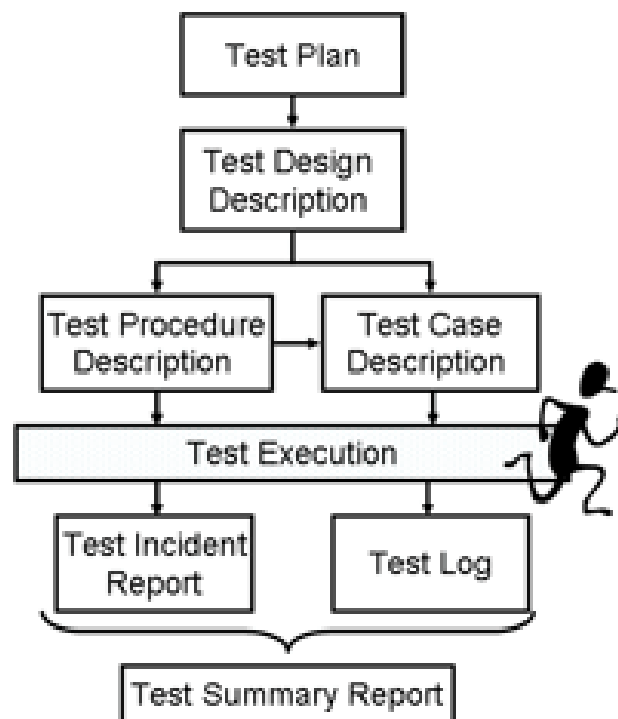
```



# Testing Document

---

Test documentation is the complete suite of artifacts that describe test planning, test design, test execution, test results and conclusions drawn from the testing activity. As testing activities typically consume 30% to 50% of project effort, testing represents a project within a project. Testing activities must therefore be fully documented to support resource allocation, monitoring and control. This page identifies the types of documents you need to set up and run your test program and summarises their content .



Source: [IEEE 829]

Fig 9.1. Testing Documentation.

### Conclusion

---

In this project , we have given a framework and application of E-Governance along with a list E-Governance facility run by state and central governments. We have also proposed future technology for E-Governance with pictorial representation of working of E-Governance with new technology. We have also proposed benefits of clouds with a graph showing how clouds reduce labor cost. Implementing E-Governance without cloud computing and open source is an old technology. Cloud computing and open source is a hottest buzzword in IT sector and we should make best possible use of these emerging technology. There are number of reasons which make cloud and open source technology so famous in E-Governance. These technologies not only provide organization, technical benefits but also provide economical benefits. NIC is providing the network backbone and a wide range of ICT ( Information and Communication Technologies ) services to government organizations throughout India. Several drafts have also been passed by IT department to implement E-Governance with these technologies so that working, efficiency, transparency and security can be increased in E-Governance

## References

---

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

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

[www.PHP.net](http://www.PHP.net)

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

[www.w3school.com](http://www.w3school.com)