# SYNOPSIS

## Overview of Population Database

Population is defined as “all the inhabitants of a particular place” or “a community of animals, plants, or humans among whose members interbreeding occurs”. Population database can be defined as the collection of data of a specific population to compute and view statically or detailed information. Here I have chosen Human as the population.

## Requirements

A software that stores the data of a specific population and can be used to find a specific element of that population with a known parameter. The data that needs to be stored is are vital parameters that uniquely identify each element of that population. For this database, we can have parameters like name, date of birth, address, etc. as some of the parameters.

## Design

Tables are designed to keep track of the details of the population. The tables are:

* Identification table
* Detail table
* Address table
* Education table
* Occupation table

## Implementation

The software has a three-layered architecture. The layers are:

1. The database server which has been designed using structured query language (SQL) on Oracle MySQL. The database contains tables that are normalized to forth normal form (4NF).
2. The middle layer is designed using PHP. It’s a connecter between the frontend and the database server. This PHP scripts are hosted on the Apache server.
3. The frontend has been designed using modern web application technology. These include HTML5, CSS3, JavaScript, JQuery, JQuery Mobile. This enables the user to interact with the database server.

## Present Systems

Now days many database systems have emerged. Technology is shifting at a rapid pace. Databases used to be maintained as text only records. Using special querying languages, we can retrieve them. Every database has a frontend build on Visual Basics or Java. This is the strong base set for the modern systems.

## Proposed Systems

With emergent of networking on computers, concept of internet started. Now we are at a time were web applications are the most matured form of web technology. Distributed systems made is even more favourable for web applications to be used everywhere. It’s a software located somewhere on the network. No installation required. Enter the URL and you are ready to use the application even from devices like mobiles and tablets to name a few.

# INTRODUCTION

## Database

A Database may be defined as inter related data stored together without harmful or necessary redundancy to serve multiple applications. The data stored so that they are independent of programs, which use the data, a common and controlled approach is used in adding new database. The data is structured so as to provide a foundation for further application development. One system is said to contain a collection of databases if they are entirely separate in structure.

The main idea of choosing the database is that it provides an enterprise with centralized control of its operational data. So, it replaces difficulties of having each application a different form of storage. Apart from these databases so provide some advantage:

* **Redundancy can be avoided**: In non-database system, each application has its own files. These can often lead to considerable redundancy in stored data which results in wastage of storage space. In database, each of these applications can be integrated into one, thus decreasing the redundancy into nil.
* **Inconsistency can be avoided (to some extent)**: This is corollary of the previous point. Say a data or record is updated and if redundancy is not controlled, there is possible change that the updated entry may not coincide the existing data. Thus, making the database inconsistent.
* **Data can be shared**: In a database, a program can be developed that operates on the existing data in the database. Thus, there will be no need to create new files.
* **Standards can be enforced**: Standardization of stored data formats is describing to data interchanges or communicates between system and thus can achieved in database.
* **Security restriction can be applied**: A database can be used for security of data stored and authorization of data, that there checks to be passed before attempting on sensitive data.

Database is a collection of interrelated data that are to be stored together in a single location. It enables sharing of data among various users as and when required.

A DBMS is a software system with capabilities to organize, manipulate and manage the data.

Defining a database involves specifying the data types, structures and constraints for the data to be stored in the database.

Constructing the database involves specifying the process of storing the data itself on some storage medium that is controlled by the DBMS.

Manipulating the database includes such functions as querying the database to receive specific data, updating the database and generating reports from the data.

The database along with the software to manage is collectively called the Database system.

Commercial data processing involves storing and retrieving voluminous data. This involves developing many application programs. Batch processing system, developed using the traditional management concepts where mechanism and maintenance procedure are gaining importance.

## Advantages of DBMS

* Data definition as a part of program
* Data redundancy reduction
* Data integrity
* Multi-user issues
* Security issues
* Development and maintenance of application systems

## Characteristics of DBMS

* It represents complex relationships between data
* Keeps a tight control over data redundancy
* Enforces user-defined rules to ensure the integrity of table data
* Has a centralized data dictionary for the storage of information pertaining to data and its manipulation
* Ensures that data can be shared across applications
* Has an automatic, intelligent backup and recovery procedure for data
* Have different interfaces via which users can manipulate the data

## Disadvantage of DBMS

* High cost of DBMS software and required hardware
* Lot of time and through has to be put during the design stage, to ensure that the capabilities of a DBMS are fully utilized

## Data Objects

A database object is anything defined and stored in a database. Database objects consist of tables, views, indexes, synonyms, database links, roles, snapshots, users, triggers, and stored program units. Each of these objects define certain characteristics of your database project.

**Table**: The basic unit of data storage in a traditional database management system. Every table has a table name and a set of columns and rows in which the data is stored. Each column is given a column name, a datatype (defining characteristics of the data to be entered in the column), and a width (quantity of space to allocate for data to be entered into the column).

**View**: Customized presentation of data from one or more tables.

**Index**: A database object created to increase the performance of the data retrievals. It provides a fast access path to data in a database. Indexes point directly to the location of the rows containing specified data.

**Relational Database**: A database composed of tables of interconnected information. Data can be reorganised and represented in different ways depending on users’ needs.

## Distributed Database

A single, logical database that is physically located on two or more computers, connected via some form of communications network. An essential feature of a true distributed database is that the user and/or program work as if they had access to the whole database locally. All processing to give this impression is carried out by the database system.

## Local Database

The database that has been installed on your computer. It can be the starter database, which is the database that comes with MySQL, or it can be a database that you have created.

## SQL

SQL (pronounced as ‘sequel’) stands for Structured Query Language. It is used to communicate with the regional database, which are in turn, a set of related information stored in the form of tables.

SQL in a non-procedural language because it processes a set of records rather than just one data at a time and also provides automatic navigation to the data. Here one can manipulate a set of rows than one at a time. SQL commands accept sets or rows as in input and returns set as output. This property of SQL allows the result of one SQL statement as input to another. SQL is utilized as a communication language among database users like DBA, system administrators, security administrators and application programmers.

# SYSTEM REQUIREMENTS

## Hardware Requirements

* A machine with Linux/Windows/MacOS that is capable of running as a web server.
* A powerful system capable to process database.
* Hard disk of storing enough database content
* Minimum screen size of 14-inch to run a web browser

## Software Requirements

* Any OS preferable Ubuntu 16.10
* MySQL server
* Apache2 server
* Firefox or another HTML5 supported browsers
* PHP 7.0 and related modules for MySQL and Apache2

# SOURCE CODE

## SQL

### Creating Database

CREATE DATABASE project;

### Creating Detail Table

CREATE TABLE `project`.`detailtable` (

`UID` INT(3) NOT NULL AUTO\_INCREMENT,

`NAME` VARCHAR(45) NOT NULL,

`GENDER` VARCHAR(6) NOT NULL,

`DOB` VARCHAR(10) NOT NULL,

`RELIGION` VARCHAR(15) NOT NULL,

`MOTHER\_TONGUE` VARCHAR(45) NOT NULL,

PRIMARY KEY (`UID`),

UNIQUE INDEX `UID\_UNIQUE` (`UID` ASC));

### Creating Id Table

CREATE TABLE `project`.`idtable` (

`UID` INT(3) NOT NULL AUTO\_INCREMENT,

`ID\_NUMBER` VARCHAR(30) NOT NULL,

`ID\_TYPE` VARCHAR(15) NOT NULL,

PRIMARY KEY (`UID`, `ID\_TYPE`),

CONSTRAINT `FK1\_UID`

FOREIGN KEY (`UID`)

REFERENCES `project`.`detailtable` (`UID`)

ON DELETE CASCADE

ON UPDATE NO ACTION);

### Creating Address Table

CREATE TABLE `project`.`addresstable` (

`UID` INT(3) NOT NULL AUTO\_INCREMENT,

`PINCODE` INT(6) NOT NULL,

`ADDRESS` VARCHAR(120) NOT NULL,

PRIMARY KEY (`UID`, `ADDRESS`),

CONSTRAINT `FK2\_UID`

FOREIGN KEY (`UID`)

REFERENCES `project`.`detailtable` (`UID`)

ON DELETE CASCADE

ON UPDATE NO ACTION);

### Creating Education Table

CREATE TABLE `project`.`educationtable` (

`UID` INT(3) NOT NULL AUTO\_INCREMENT,

`TEN` INT(1) NOT NULL DEFAULT 0,

`TWE` INT(1) NOT NULL DEFAULT 0,

`UG` INT(1) NOT NULL DEFAULT 0,

`PG` INT(1) NOT NULL DEFAULT 0,

`DR` INT(1) NOT NULL DEFAULT 0,

PRIMARY KEY (`UID`),

CONSTRAINT `FK3\_UID`

FOREIGN KEY (`UID`)

REFERENCES `project`.`detailtable` (`UID`)

ON DELETE CASCADE

ON UPDATE NO ACTION);

### Creating Work Table

CREATE TABLE `project`.`worktable` (

`UID` INT(3) NOT NULL AUTO\_INCREMENT,

`WORK\_TYPE` VARCHAR(60) NOT NULL,

PRIMARY KEY (`UID`, `WORK\_TYPE`),

CONSTRAINT `FK4\_UID`

FOREIGN KEY (`UID`)

REFERENCES `project`.`detailtable` (`UID`)

ON DELETE CASCADE

ON UPDATE NO ACTION);

## PHP

### Connection (connect.php)

<?php

//variables

$servername = "localhost";

$username = "root";

$password = "root";

$db = "project";

// Create connection

$conn = new mysqli($servername, $username, $password,$db);

?>

### Displaying The Tables (select.php)

<?php

//connection

include 'connect.php';

//query

$sql = "SELECT D.UID AS UID,ID\_TYPE,ID\_NUMBER,NAME,GENDER,DOB,RELIGION,MOTHER\_TONGUE,ADDRESS,PINCODE,TEN,TWE,UG,PG,DR,WORK\_TYPE FROM idtable as I, detailtable as D, addresstable as A, educationtable as E,worktable as W WHERE I.UID = D.UID AND D.UID = A.UID AND A.UID = E.UID AND E.UID = W.UID";

$arr = array();

$result = mysqli\_query($conn,$sql);

if(mysqli\_num\_rows($result)>0) {

while($row = mysqli\_fetch\_assoc($result)) {

$arr[] = $row;

}

}

//send output

echo json\_encode($arr);

?>

### Display Count Of Records (count.php)

<?php

//connection

include 'connect.php';

//query

$sql = "SELECT COUNT(\*) AS COUNT FROM idtable";

$arr = array();

$result = mysqli\_query($conn,$sql);

if(mysqli\_num\_rows($result)>0) {

while($row = mysqli\_fetch\_assoc($result)) {

$arr[] = $row;

}

}

//display the count

echo json\_encode($arr);

?>

### Display Tables On Search (search.php)

<?php

//connection

include 'connect.php';

$find = $\_POST['find'];

//search query

$sql = "SELECT D.UID AS UID,ID\_TYPE,ID\_NUMBER,NAME,GENDER,DOB,RELIGION,MOTHER\_TONGUE,ADDRESS,PINCODE,TEN,TWE,UG,PG,DR,WORK\_TYPE FROM idtable AS I, detailtable AS D,addresstable AS A,educationtable AS E,worktable AS W WHERE (I.UID = D.UID AND D.UID = A.UID AND A.UID = E.UID AND E.UID = W.UID) AND (D.UID LIKE '%".$find."%' OR ID\_TYPE LIKE '%".$find."%' OR ID\_NUMBER LIKE '%".$find."%' OR NAME LIKE '%".$find."%' OR GENDER LIKE '%".$find."%' OR DOB LIKE '%".$find."%' OR RELIGION LIKE '%".$find."%' OR MOTHER\_TONGUE LIKE '%".$find."%' OR ADDRESS LIKE '%".$find."%' OR PINCODE LIKE '%".$find."%' OR TEN LIKE '%".$find."%' OR TWE LIKE '%".$find."%' OR UG LIKE '%".$find."%' OR PG LIKE '%".$find."%' OR DR LIKE '%".$find."%' OR WORK\_TYPE LIKE '%".$find."%')";

$arr = array();

$result = mysqli\_query($conn,$sql);

if(mysqli\_num\_rows($result)>0) {

while($row = mysqli\_fetch\_assoc($result)) {

$arr[] = $row;

}

}

//send output

echo json\_encode($arr);

?>

### Display Count Of Records On Search (count\_search.php)

<?php

//connection

include 'connect.php';

$find = $\_POST['find'];

//query

$sql = "SELECT COUNT(\*) AS COUNT FROM idtable AS I, detailtable AS D,addresstable AS A,educationtable AS E,worktable AS W WHERE (I.UID = D.UID AND D.UID = A.UID AND A.UID = E.UID AND E.UID = W.UID) AND (D.UID LIKE '%".$find."%' OR ID\_TYPE LIKE '%".$find."%' OR ID\_NUMBER LIKE '%".$find."%' OR NAME LIKE '%".$find."%' OR GENDER LIKE '%".$find."%' OR DOB LIKE '%".$find."%' OR RELIGION LIKE '%".$find."%' OR MOTHER\_TONGUE LIKE '%".$find."%' OR ADDRESS LIKE '%".$find."%' OR PINCODE LIKE '%".$find."%' OR TEN LIKE '%".$find."%' OR TWE LIKE '%".$find."%' OR UG LIKE '%".$find."%' OR PG LIKE '%".$find."%' OR DR LIKE '%".$find."%' OR WORK\_TYPE LIKE '%".$find."%')";

$arr = array();

$result = mysqli\_query($conn,$sql);

if(mysqli\_num\_rows($result)>0) {

while($row = mysqli\_fetch\_assoc($result)) {

$arr[] = $row;

}

}

//send output

echo json\_encode($arr);

?>

### Insert Record (insert.php)

<?php

//connection

include 'connect.php';

//variables

$idtype = $\_POST["id\_type"];

$idnumber = $\_POST["id\_num"];

$name = $\_POST["name"];

$gender = $\_POST["gender"];

$dob = $\_POST["dob"];

$religion = $\_POST["religion"];

$mothertongue = $\_POST["mother\_tongue"];

$address = $\_POST["address"];

$pincode = $\_POST["pincode"];

$ten = $\_POST["ten"]==1?1:0;

$twe = $\_POST["twe"]==1?1:0;

$ug = $\_POST["ug"]==1?1:0;

$pg = $\_POST["pg"]==1?1:0;

$dr = $\_POST["dr"]==1?1:0;

$work = $\_POST["work"];

$UID;

$mes=0;

//insert detailtable

$sql1 = "INSERT INTO `detailtable`(`NAME`, `GENDER`, `DOB`, `RELIGION`, `MOTHER\_TONGUE`) VALUES('".$name."','".$gender."','".$dob."','".$religion."','".$mothertongue."')";

if (mysqli\_query($conn, $sql1)) {

//$mes=0;

} else {

$mes=1;

}

//insert idtable

$sql2 = "INSERT INTO `idtable`(`ID\_NUMBER`, `ID\_TYPE`) VALUES ('".$idnumber."','".$idtype."')";

if (mysqli\_query($conn, $sql2)) {

//$mes=0;

} else {

$mes=1;

}

//insert addresstable

$sql3 = "INSERT INTO `addresstable`(`PINCODE`, `ADDRESS`) VALUES(".$pincode.",'".$address."')";

if (mysqli\_query($conn, $sql3)) {

//$mes=0;

} else {

$mes=1;

}

//insert educationtable

$sql4 = "INSERT INTO `educationtable`(`TEN`, `TWE`, `UG`, `PG`, `DR`) VALUES(".$ten.",".$twe.",".$ug.",".$pg.",".$dr.")";

if (mysqli\_query($conn, $sql4)) {

//$mes=0;

} else {

$mes=1;

}

//insert worktable

$sql5 = "INSERT INTO `worktable`(`WORK\_TYPE`) VALUES('".$work."')";

if (mysqli\_query($conn, $sql5)) {

//$mes=0;

} else {

$mes=1;

}

//send post status

echo $msg;

?>

### Update Record (update.php)

<?php

//connection

include 'connect.php';

//variables

$uid = $\_POST["uid"];

$sql;

$m;

//update detailtable

if(isset($\_POST["name"])&& $\_POST["name"]!="") {

$name = $\_POST["name"];

$sql = "UPDATE detailtable SET NAME='".$name."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["gender"])&&$\_POST["gender"]!=""){

$gender = $\_POST["gender"];

$sql = "UPDATE detailtable SET GENDER='".$gender."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["dob"])&&$\_POST["dob"]!=""){

$dob = $\_POST["dob"];

$sql = "UPDATE detailtable SET DOB='".$dob."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["religion"])&&$\_POST["religion"]!=""){

$religion = $\_POST["religion"];

$sql = "UPDATE detailtable SET RELIGION='".$religion."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["mother\_tongue"])&&$\_POST["mother\_tongue"]!=""){

$mothertongue = $\_POST["mother\_tongue"];

$sql = "UPDATE detailtable SET MOTHER\_TONGUE='".$mothertongue."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

//update idtable

if(isset($\_POST["id\_type"])&&$\_POST["id\_type"]!="") {

$idtype = $\_POST["id\_type"];

$sql = "UPDATE idtable SET ID\_TYPE='".$idtype."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["id\_num"])&&$\_POST["id\_num"]!="") {

$idnum = $\_POST["id\_num"];

$sql = "UPDATE idtable SET ID\_NUMBER='".$idnum."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

//update addresstable

if(isset($\_POST["address"])&&$\_POST["address"]!="") {

$address = $\_POST["address"];

$sql = "UPDATE addresstable SET ADDRESS='".$address."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["pincode"])&&$\_POST["pincode"]!="") {

$pincode = $\_POST["pincode"];

$sql = "UPDATE addresstable SET PINCODE=".$pincode." WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

//update worktable

if(isset($\_POST["work"])&&$\_POST["work"]!="") {

$work = $\_POST["work"];

$sql = "UPDATE worktable SET WORK\_TYPE='".$work."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

//update educationtable

if(isset($\_POST["ten"])&&$\_POST["ten"]!="") {

$ten = $\_POST["ten"]==1?1:0;

$sql = "UPDATE educationtable SET TEN='".$ten."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["twe"])&&$\_POST["twe"]!="") {

$twe = $\_POST["twe"]==1?1:0;

$sql = "UPDATE educationtable SET TWE='".$twe."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["ug"])&&$\_POST["ug"]!="") {

$ug = $\_POST["ug"]==1?1:0;

$sql = "UPDATE educationtable SET UG='".$ug."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["pg"])&&$\_POST["pg"]!="") {

$pg = $\_POST["pg"]==1?1:0;

$sql = "UPDATE educationtable SET PG='".$pg."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

if(isset($\_POST["dr"])&&$\_POST["dr"]!="") {

$dr = $\_POST["dr"]==1?1:0;

$sql = "UPDATE educationtable SET DR='".$dr."' WHERE UID=".$uid."";

if (mysqli\_query($conn, $sql)) {

$m=0;

} else {

$m=1;

}

}

//send post status

echo $m;

?>

### Delete Record (delete.php)

<?php

//connection

include 'connect.php';

//variables

$UID = $\_POST["uid"];

$mes=0;

//delete detailtable

$sql1 = "DELETE FROM detailtable WHERE UID=".$UID."";

if (mysqli\_query($conn, $sql1)) {

//$mes=0;

} else {

$mes=1;

}

//delete idtable

$sql2 = "DELETE FROM idtable WHERE UID=".$UID."";

if (mysqli\_query($conn, $sql2)) {

//$mes=0;

} else {

$mes=1;

}

//delete addresstable

$sql3 = "DELETE FROM addresstable WHERE UID=".$UID."";

if (mysqli\_query($conn, $sql3)) {

//$mes=0;

} else {

$mes=1;

}

//delete educationtable

$sql4 = "DELETE FROM educationtable WHERE UID=".$UID."";

if (mysqli\_query($conn, $sql4)) {

//$mes=0;

} else {

$mes=1;

}

//delete worktable

$sql5 = "DELETE FROM worktable WHERE UID=".$UID."";

if (mysqli\_query($conn, $sql5)) {

//$mes=0;

} else {

$mes=1;

}

//send post status

echo $mes;

?>

### Generating Report (report.php)

<?php

//connection

include 'connect.php';

$t1=$\_POST['t1'];

$t2=$\_POST['t2'];

$t3=$\_POST['t3'];

$t4=$\_POST['t4'];

$t5=$\_POST['t5'];

$sql="";

//generate report based on table selected

if($t1==0&&$t2==0&&$t3==0&&$t4==0&&$t5==0) {

//$sql="";

} else if($t1==0&&$t2==0&&$t3==0&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM worktable";

} else if($t1==0&&$t2==0&&$t3==0&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM educationtable";

} else if($t1==0&&$t2==0&&$t3==0&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM educationtable as e, worktable as w WHERE e.uid=w.uid";

} else if($t1==0&&$t2==0&&$t3==1&&$t4==0&&$t5==0) {

$sql="SELECT \* FROM addresstable";

} else if($t1==0&&$t2==0&&$t3==1&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM addresstable as a, worktable as w WHERE a.uid=w.uid";

} else if($t1==0&&$t2==0&&$t3==1&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM addresstable as a, educationtable as e WHERE a.uid=e.uid";

} else if($t1==0&&$t2==0&&$t3==1&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM addresstable as a, educationtable as e, worktable as w WHERE a.uid=e.uid AND e.uid=w.uid";

} else if($t1==0&&$t2==1&&$t3==0&&$t4==0&&$t5==0) {

$sql="SELECT \* FROM detailtable";

} else if($t1==0&&$t2==1&&$t3==0&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM detailtable as d, worktable as w WHERE d.uid=w.uid";

} else if($t1==0&&$t2==1&&$t3==0&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM detailtable as d, educationtable as e WHERE d.uid=e.uid";

} else if($t1==0&&$t2==1&&$t3==0&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM detailtable as d, educationtable as e, worktable as w WHERE d.uid=e.uid AND e.uid=w.uid";

} else if($t1==0&&$t2==1&&$t3==1&&$t4==0&&$t5==0) {

$sql="SELECT \* FROM detailtable as d, addresstable as a WHERE d.uid=a.uid";

} else if($t1==0&&$t2==1&&$t3==1&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM detailtable as d, addresstable as a, worktable as w WHERE d.uid=a.uid AND a.uid=w.uid";

} else if($t1==0&&$t2==1&&$t3==1&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM detailtable as d, addresstable as a, educationtable as e WHERE d.uid=a.uid AND a.uid=e.uid";

} else if($t1==0&&$t2==1&&$t3==1&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM detailtable as d, addresstable as a, educationtable as e, worktable as w WHERE d.uid=a.uid AND a.uid=e.uid AND e.uid=w.uid";

} else if($t1==1&&$t2==0&&$t3==0&&$t4==0&&$t5==0) {

$sql="SELECT \* FROM idtable";

} else if($t1==1&&$t2==0&&$t3==0&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM idtable as i, worktable as w WHERE i.uid=w.uid";

} else if($t1==1&&$t2==0&&$t3==0&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM idtable as i, educationtable as e WHERE i.uid=e.uid";

} else if($t1==1&&$t2==0&&$t3==0&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM idtable as i, educationtable as e, worktable as w WHERE i.uid=e.uid AND e.uid=w.uid";

} else if($t1==1&&$t2==0&&$t3==1&&$t4==0&&$t5==0) {

$sql="SELECT \* FROM idtable as i, addresstable as a WHERE i.uid=a.uid";

} else if($t1==1&&$t2==0&&$t3==1&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM idtable as i, addresstable as a, worktable as w WHERE i.uid=a.uid AND a.uid=w.uid";

} else if($t1==1&&$t2==0&&$t3==1&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM idtable as i, addresstable as a, educationtable as e WHERE i.uid=a.uid AND a.uid=e.uid";

} else if($t1==1&&$t2==0&&$t3==1&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM idtable as i, addresstable as a, educationtable as e, worktable as w WHERE i.uid=a.uid AND a.uid=e.uid AND e.uid=w.uid";

} else if($t1==1&&$t2==1&&$t3==0&&$t4==0&&$t5==0) {

$sql="SELECT \* FROM idtable as i, detailtable as d WHERE i.uid=d.uid";

} else if($t1==1&&$t2==1&&$t3==0&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM idtable as i, detailtable as d, worktable as w WHERE i.uid=d.uid AND d.uid=w.uid";

} else if($t1==1&&$t2==1&&$t3==0&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM idtable as i, detailtable as d, educationtable as e WHERE i.uid=d.uid AND d.uid=e.uid";

} else if($t1==1&&$t2==1&&$t3==0&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM idtable as i, detailtable as d, educationtable as e, worktable as w WHERE i.uid=d.uid AND d.uid=e.uid AND e.uid=w.uid";

} else if($t1==1&&$t2==1&&$t3==1&&$t4==0&&$t5==0) {

$sql="SELECT \* FROM idtable as i, detailtable as d, addresstable as a WHERE i.uid=d.uid AND d.uid=a.uid";

} else if($t1==1&&$t2==1&&$t3==1&&$t4==0&&$t5==1) {

$sql="SELECT \* FROM idtable as i, detailtable as d, addresstable as a, worktable as w WHERE i.uid=d.uid AND d.uid=a.uid AND a.uid=w.uid";

} else if($t1==1&&$t2==1&&$t3==1&&$t4==1&&$t5==0) {

$sql="SELECT \* FROM idtable as i, detailtable as d, addresstable as a, educationtable as e WHERE i.uid=d.uid AND d.uid=a.uid AND a.uid=e.uid";

} else if($t1==1&&$t2==1&&$t3==1&&$t4==1&&$t5==1) {

$sql="SELECT \* FROM idtable as i, detailtable as d, addresstable as a, educationtable as e, worktable as w WHERE i.uid=d.uid AND d.uid=a.uid AND a.uid=e.uid AND e.uid=w.uid";

}

$arr = array();

$result = mysqli\_query($conn,$sql);

if(mysqli\_num\_rows($result)>0) {

while($row = mysqli\_fetch\_assoc($result)) {

$arr[] = $row;

}

}

//send output

echo json\_encode($arr);

?>

## HTML5 & JAVASCRIPT

### Frontend Of Project (connect.html)

<!DOCTYPE html>

<html>

<head>

<!-- frameworks, stylesheets and metadata -->

<meta name="viewport" content="width=device-width, initial-scale=1" >

<script type="text/javascript" src="jquery-1.11.3.min.js"></script>

<script type="text/javascript" src="jquery.mobile-1.4.5.min.js"></script>

<link type="text/css" rel="stylesheet" href="jquery.mobile-1.4.5.min.css" />

<link type="text/css" rel="stylesheet" href="font-awesome.min.css" />

<!-- select.php connector -->

<script>

$(document).ready(function() {

$.getJSON("select.php", function(data) {

var tr=[];

for(var i = 0; i < data.length; i++) {

tr.push("<tr>");

tr.push("<td>"+data[i].UID+"</td>");

tr.push("<td>"+data[i].ID\_TYPE+"</td>");

tr.push("<td>"+data[i].ID\_NUMBER+"</td>");

tr.push("<td>"+data[i].NAME+"</td>");

tr.push("<td>"+data[i].GENDER+"</td>");

tr.push("<td>"+data[i].DOB+"</td>");

tr.push("<td>"+data[i].RELIGION+"</td>");

tr.push("<td>"+data[i].MOTHER\_TONGUE+"</td>");

tr.push("<td>"+data[i].ADDRESS+"</td>");

tr.push("<td>"+data[i].PINCODE+"</td>");

tr.push("<td>"+data[i].TEN+" "+data[i].TWE+" "+data[i].UG+" "+data[i].PG+" "+data[i].DR+"</td>");

tr.push("<td>"+data[i].WORK\_TYPE+"</td>");

tr.push("</tr>");

}

$('#tbodyhome').html($(tr.join('')));

});

});

</script>

<!-- count.php connector -->

<script>

$(document).ready(function() {

$.getJSON("count.php", function(data) {

var cnt=[];

for(var i=0; i< data.length; i++) {

cnt.push("<p>");

cnt.push(""+data[i].COUNT+" Records");

cnt.push("</p>");

}

$('#counthome').html($(cnt.join('')));

});

});

</script>

<!-- count\_search.php connector -->

<script>

$(document).ready(function() {

$(document).on('input', '#find', function() {

$.ajax({

url: 'count\_search.php',

data: {find: $('#find').val()},

type: 'post',

cache: 'false',

success: function(result) {

var data = $.parseJSON(result);

var cnt=[];

for(var i=0; i< data.length; i++) {

cnt.push("<p>");

cnt.push(""+data[i].COUNT+" Records");

cnt.push("</p>");

}

$('#counthome').html($(cnt.join('')));

}

});

});

});

</script>

<!-- search.php connector -->

<script>

$(document).ready(function() {

$(document).on('input', '#find', function() {

$.ajax({

url: 'search.php',

data: {find: $('#find').val()},

type: 'post',

cache: 'false',

success: function(result) {

var json\_obj = $.parseJSON(result);

var tr=[];

for(var i = 0; i < json\_obj.length; i++) {

tr.push("<tr>");

tr.push("<td>"+json\_obj[i].UID+"</td>");

tr.push("<td>"+json\_obj[i].ID\_TYPE+"</td>");

tr.push("<td>"+json\_obj[i].ID\_NUMBER+"</td>");

tr.push("<td>"+json\_obj[i].NAME+"</td>");

tr.push("<td>"+json\_obj[i].GENDER+"</td>");

tr.push("<td>"+json\_obj[i].DOB+"</td>");

tr.push("<td>"+json\_obj[i].RELIGION+"</td>");

tr.push("<td>"+json\_obj[i].MOTHER\_TONGUE+"</td>");

tr.push("<td>"+json\_obj[i].ADDRESS+"</td>");

tr.push("<td>"+json\_obj[i].PINCODE+"</td>");

tr.push("<td>"+json\_obj[i].TEN+" "+json\_obj[i].TWE+" "+json\_obj[i].UG+" "+json\_obj[i].PG+" "+json\_obj[i].DR+"</td>");

tr.push("<td>"+json\_obj[i].WORK\_TYPE+"</td>");

tr.push("</tr>");

}

$('#tbodyhome').html($(tr.join('')));

}

});

});

});

</script>

<!-- insert.php connector-->

<script>

var validateInsert = 1;

function removeOldInsertErrors() {

$('#idtypeinserterror').next('i').remove();

$('#idnuminserterror').next('i').remove();

$('#nameinserterror').next('i').remove();

$('#dobinserterror').next('i').remove();

$('#religioninserterror').next('i').remove();

$('#langinserterror').next('i').remove();

$('#addressinserterror').next('i').remove();

$('#pincodeinserterror').next('i').remove();

$('#workinserterror').next('i').remove();

validateInsert = 1;

}

function validatorInsert() {

removeOldInsertErrors();

if($('#id\_type\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#idtypeinserterror');

validateInsert = 0;

}

if($('#id\_num\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#idnuminserterror');

validateInsert = 0;

}

if($('#name\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#nameinserterror');

validateInsert = 0;

}

if($('#dob\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#dobinserterror');

validateInsert = 0;

}

if($('#religion\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#religioninserterror');

validateInsert = 0;

}

if($('#mother\_tongue\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#langinserterror');

validateInsert = 0;

}

if($('#address\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#addressinserterror');

validateInsert = 0;

}

if($('#pincode\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#pincodeinserterror');

validateInsert = 0;

}

if($('#work\_insert').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#workinserterror');

validateInsert = 0;

}

}

$(document).ready(function(){

$(document).on('click','#insertbutton',function(){

validatorInsert();

if (validateInsert == 1) {

$.ajax({

url:'insert.php',

type: 'post',

cache: 'false',

data: { id\_type: $('#id\_type\_insert').val(),

id\_num: $('#id\_num\_insert').val(),

name: $('#name\_insert').val(),

gender: $('input:radio[name=gender\_insert]:checked').val(),

dob: $('#dob\_insert').val(),

religion: $('#religion\_insert').val(),

mother\_tongue: $('#mother\_tongue\_insert').val(),

address: $('#address\_insert').val(),

pincode: $('#pincode\_insert').val(),

ten: $('#ten\_insert').val(),

twe: $('#twe\_insert').val(),

ug: $('#ug\_insert').val(),

pg: $('#pg\_insert').val(),

dr: $('#dr\_insert').val(),

work: $('#work\_insert').val()

},

success: function(resultInsert) {

if(resultInsert==0) { alert('Success'); window.location.href="connect.html"; }

else if(resultInsert==1) { alert('Error'); window.location.href="connect.html"; }

}

});

}

});

});

</script>

<!-- update.php connector -->

<script>

var validateUpdate = 1;

function removeOldUpdateError() {

$('#uidupdateerror').next('i').remove();

validateUpdate = 1;

}

function validatorUpdate() {

removeOldUpdateError();

if($('#uid\_update').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#uidupdateerror');

validateUpdate = 0;

}

}

$(document).ready(function() {

$(document).on('click', '#updatebutton', function() {

validatorUpdate();

if(validateUpdate == 1) {

$.ajax({

url:'update.php',

type: 'post',

cache: 'false',

data: { uid: $('#uid\_update').val(),

id\_type: $('#id\_type\_update').val(),

id\_num: $('#id\_num\_update').val(),

name: $('#name\_update').val(),

gender: $('input:radio[name=gender\_update]:checked').val(),

dob: $('#dob\_update').val(),

religion: $('#religion\_update').val(),

mother\_tongue: $('#mother\_tongue\_update').val(),

address: $('#address\_update').val(),

pincode: $('#pincode\_update').val(),

ten: $('#ten\_update').val(),

twe: $('#twe\_update').val(),

ug: $('#ug\_update').val(),

pg: $('#pg\_update').val(),

dr: $('#dr\_update').val(),

work: $('#work\_update').val()

},

success: function(resultUpdate) {

if(resultUpdate==0) { alert('Success'); window.location.href="connect.html"; }

else if(resultUpdate==1) { alert('Error'); window.location.href="connect.html"; }

}

});

}

});

});

</script>

<!-- delete.php connector -->

<script>

var validateDelete = 1;

function removeOldDeleteError() {

$('#uiddeleteerror').next('i').remove();

validateDelete = 1;

}

function validatorDelete() {

removeOldDeleteError();

if($('#uid\_delete').val() == '') {

$('<i class="fa fa-exclamation-circle" style="color:red;"></i>').insertAfter('#uiddeleteerror');

validateDelete = 0;

}

}

$(document).ready(function() {

$(document).on('click', '#deletebutton', function() {

validatorDelete();

if(validateDelete == 1) {

$.ajax({

url:'delete.php',

type: 'post',

cache: 'false',

data: { uid: $('#uid\_delete').val()

},

success: function(resultDelete) {

if(resultDelete==0) { alert('Success'); window.location.href="connect.html"; }

else if(resultDelete==1) { alert('Error'); window.location.href="connect.html"; }

}

});

}

});

});

</script>

<!-- report.php connector -->

<script>

$(document).ready( function() {

$(document).on('click', '#reportbutton', function() {

var x1 = $('#idtablereport').val()==1?1:0;

var x2 = $('#detailtablereport').val()==1?1:0;

var x3 = $('#addresstablereport').val()==1?1:0;

var x4 = $('#educationtablereport').val()==1?1:0;

var x5 = $('#worktablereport').val()==1?1:0;

$.ajax({

url:'report.php',

type: 'post',

cache: 'false',

data: { t1: $('#idtablereport').val(),

t2: $('#detailtablereport').val(),

t3: $('#addresstablereport').val(),

t4: $('#educationtablereport').val(),

t5: $('#worktablereport').val()

},

success: function(result) {

var th=[];

th.push("<tr>");

if(x1==1||x2==1||x3==1||x4==1||x5==1) {

th.push("<th>UID</th>");

}

if(x1==1) {

th.push("<th>ID\_TYPE</th>");

th.push("<th>ID\_NUMBER</th>");

}

if(x2==1) {

th.push("<th>NAME</th>");

th.push("<th>GENDER</th>");

th.push("<th>DOB</th>");

th.push("<th>RELIGION</th>");

th.push("<th>MOTHER\_TONGUE</th>");

}

if(x3==1) {

th.push("<th>ADDRESS</th>");

th.push("<th>PINCODE</th>");

}

if(x4==1) {

th.push("<th>TEN</th>");

th.push("<th>TWE</th>");

th.push("<th>UG</th>");

th.push("<th>PG</th>");

th.push("<th>DR</th>");

}

if(x5==1) {

th.push("<th>WORK\_TYPE</th>");

}

th.push("</tr");

$('#theadreport').html($(th.join('')));

var json\_obj = $.parseJSON(result);

var tr=[];

for(var i = 0; i < json\_obj.length; i++) {

tr.push("<tr>");

if(x1==1||x2==1||x3==1||x4==1||x5==1) {

tr.push("<td>"+json\_obj[i].UID+"</td>");

}

if(x1==1) {

tr.push("<td>"+json\_obj[i].ID\_TYPE+"</td>");

tr.push("<td>"+json\_obj[i].ID\_NUMBER+"</td>");

}

if(x2==1) {

tr.push("<td>"+json\_obj[i].NAME+"</td>");

tr.push("<td>"+json\_obj[i].GENDER+"</td>");

tr.push("<td>"+json\_obj[i].DOB+"</td>");

tr.push("<td>"+json\_obj[i].RELIGION+"</td>");

tr.push("<td>"+json\_obj[i].MOTHER\_TONGUE+"</td>");

}

if(x3==1) {

tr.push("<td>"+json\_obj[i].ADDRESS+"</td>");

tr.push("<td>"+json\_obj[i].PINCODE+"</td>");

}

if(x4==1) {

tr.push("<td>"+json\_obj[i].TEN+"</td>");

tr.push("<td>"+json\_obj[i].TWE+"</td>");

tr.push("<td>"+json\_obj[i].UG+"</td>");

tr.push("<td>"+json\_obj[i].PG+"</td>");

tr.push("<td>"+json\_obj[i].DR+"</td>");

}

if(x5==1) {

tr.push("<td>"+json\_obj[i].WORK\_TYPE+"</td>");

}

tr.push("</tr>");

}

$('#tbodyreport').html($(tr.join('')));

}

});

});

});

</script>

</head>

<body onLoad="document.getElementById('searchbar').reset();document.getElementById('insertform').reset();document.getElementById('updateform').reset();document.getElementById('deleteform').reset();document.getElementById('reportform').reset();">

<!-- home page -->

<div data-role="page" id="homepage">

<div data-role="header" data-position="fixed">

<a href="#helppage" data-role="button" data-icon="alert" data-mini="true" data-transition="pop">Help</a>

<h1>Population Database</h1>

<a href="#aboutpage" data-role="button" data-icon="info" data-iconpos="right" data-mini="true" data-transition="pop">About</a>

</div>

<div data-role="main" class="ui-content">

<div class="ui-field-contain">

<form id="searchbar">

<input type="search" name="find" id="find" data-mini="true" placeholder="Search....">

</form>

</div>

<h7 id="counthome" style="text-align: center;/\*! \*/font-weight: bold;"></h7>

<table data-role="table" class="ui-responsive">

<thead><tr>

<th>UID</th>

<th>ID TYPE</th>

<th>ID NUMBER</th>

<th>NAME</th>

<th>GENDER</th>

<th>DOB</th>

<th>RELIGION</th>

<th>MOTHER TONGUE</th>

<th>ADDRESS</th>

<th>PINCODE</th>

<th>EDUCATION</th>

<th>WORK</th>

</tr></thead>

<tbody id="tbodyhome"></tbody>

</table>

</div>

<div data-role="footer" data-position="fixed">

<div data-role="navbar">

<ul>

<li><a href="#insertpage" data-role="button" data-icon="plus" data-mini="true" data-iconpos="top" data-transition="flip">Insert</a></li>

<li><a href="#updatepage" data-role="button" data-icon="edit" data-mini="true" data-iconpos="top" data-transition="flip">Update</a></li>

<li><a href="#deletepage" data-role="button" data-icon="delete" data-mini="true" data-iconpos="top" data-transition="flip">Delete</a></li>

<li><a href="#reportpage" data-role="button" data-icon="grid" data-mini="true" data-iconpos="top" data-transition="flip">Report</a>

</ul>

</div>

</div>

</div>

<!-- about page -->

<div data-role="page" data-dialog="true" id="aboutpage">

<div data-role="header">

<h3>About</h3>

</div>

<div data-role="main" class="ui-content">

<p>Author: Mayank Metha D<br>

USN: 1MV14CS054</p>

</div>

</div>

<!-- help page -->

<div data-role="page" data-dialog="true" id="helppage">

<div data-role="header">

<h3>Help</h3>

</div>

<div data-role="main" class="ui-content">

<ul>

<li>UID is used to update and delete a record, so dont forget it.</li>

<li>While using search box, don't press enter key.</li>

<li>While updating record Gender radio button and Education checkbox could change the original value.Please enter its original value to avoid change of original value after update</li>

<li>The Education Column has 5 bits that stand for Tenth,+2,Undergraduate,Postgraduate,Doctrate respectively.If the value is 1 then that field is checked else if 0 that field is unchecked.</li>

<li>Required input boxed if left blank will have <i class="fa fa-exclamation-circle" style="color:red;"></i> symbol before the input field label on submit.</li>

<li>Any error that occurs during add, update or delete of record will have a generalized error message.</li>

</ul>

</div>

</div>

<!-- insert page -->

<div data-role="page" id="insertpage">

<div data-role="header">

<a href="#homepage" id="back" data-role="button" data-mini="true" data-icon="carat-l" data-transition="flip">Back</a>

<h2>Insert</h2>

</div>

<div data-role="main" class="ui-content">

<form id="insertform">

<div data-role="fieldcontain">

<label for="id\_type\_insert"><div id="idtypeinserterror"></div> Id Type</label>

<input type="text" data-mini="true" id="id\_type\_insert" name="id\_type\_insert">

</div>

<div data-role="fieldcontain">

<label for="id\_num\_insert"><div id="idnuminserterror"></div> Id Number</label>

<input type="text" data-mini="true" id="id\_num\_insert" name="id\_num\_insert">

</div>

<div data-role="fieldcontain">

<label for="name\_insert"><div id="nameinserterror"></div> Name</label>

<input type="text" data-mini="true" id="name\_insert" name="name\_insert">

</div>

<div data-role="fieldcontain">

<fieldset data-role="controlgroup" data-mini="true" data-type="horizontal">

<legend> Gender</legend>

<input type="radio" id="male" name="gender\_insert" value="Male" checked="check">

<label for="male">Male</label>

<input type="radio" id="female" name="gender\_insert" value="Female">

<label for="female">Female</label>

<input type="radio" id="others" name="gender\_insert" value="Others">

<label for="others">Others</label>

</fieldset>

</div>

<div data-role="fieldcontain">

<label for="dob\_insert"><div id="dobinserterror"></div> Date Of Birth</label>

<input type="date" name="dob\_insert" id="dob\_insert" data-mini="true" placeholder="dd-mm-yyyy">

</div>

<div data-role="fieldcontain">

<label for="religion\_insert"><div id="religioninserterror"></div> Religion</label>

<input type="text" data-mini="true" id="religion\_insert" name="religion\_insert">

</div>

<div data-role="fieldcontain">

<label for="mother\_tongue\_insert"><div id="langinserterror"></div> Mother Tongue</label>

<input type="text" data-mini="true" id="mother\_tongue\_insert" name="mother\_tongue\_insert">

</div>

<div data-role="fieldcontain">

<label for="address\_insert"><div id="addressinserterror"></div> Address</label>

<input type="text" data-mini="true" id="address\_insert" name="address\_insert">

</div>

<div data-role="fieldcontain">

<label for="pincode\_insert"><div id="pincodeinserterror"></div> Pincode</label>

<input type="text" data-mini="true" id="pincode\_insert" name="pincode\_insert">

</div>

<div data-role="fieldcontain">

<fieldset data-role="controlgroup" data-mini="true" data-type="horizontal">

<legend> Education</legend>

<input type="checkbox" id="ten\_insert" name="ten\_insert" onChange="$(this).val(this.checked? '1': '0');">

<label for="ten\_insert">Tenth</label>

<input type="checkbox" id="twe\_insert" name="twe\_insert" onChange="$(this).val(this.checked? '1': '0');">

<label for="twe\_insert">+2</label>

<input type="checkbox" id="ug\_insert" name="ug\_insert" onChange="$(this).val(this.checked? '1': '0');">

<label for="ug\_insert">Under Graduate</label>

<input type="checkbox" id="pg\_insert" name="pg\_insert" onChange="$(this).val(this.checked? '1': '0');">

<label for="pg\_insert">Post Graduate</label>

<input type="checkbox" id="dr\_insert" name="dr\_insert" onChange="$(this).val(this.checked? '1': '0');">

<label for="dr\_insert">Doctrate</label>

</fieldset>

</div>

<div data-role="fieldcontain">

<label for="work\_insert"><div id="workinserterror"></div> Work Type</label>

<input type="text" data-mini="true" id="work\_insert" name="work\_insert">

</div>

<input type="button" id="insertbutton" name="insertbutton" value="Submit" data-mini="true" data-icon="check">

</form>

</div>

</div>

<!-- update page -->

<div data-role="page" id="updatepage">

<div data-role="header">

<a href="#homepage" id="back" data-role="button" data-icon="carat-l" data-mini="true" data-transition="flip">Back</a>

<h2>Update</h2>

</div>

<div data-role="main" class="ui-content">

<form id="updateform">

<div data-role="fieldcontain">

<label for="uid\_update"><div id="uidupdateerror"></div> UID</label>

<input type="text" data-mini="true" id="uid\_update" name="uid\_update">

</div>

<div data-role="fieldcontain">

<label for="id\_type\_update">Id Type</label>

<input type="text" data-mini="true" id="id\_type\_update" name="id\_type\_update">

</div>

<div data-role="fieldcontain">

<label for="id\_num\_update">Id Number</label>

<input type="text" data-mini="true" id="id\_num\_update" name="id\_num\_update">

</div>

<div data-role="fieldcontain">

<label for="name\_update">Name</label>

<input type="text" data-mini="true" id="name\_update" name="name\_update">

</div>

<div data-role="fieldcontain">

<fieldset data-role="controlgroup" data-mini="true" data-type="horizontal">

<legend>Gender</legend>

<input type="radio" id="male" name="gender\_update" value="Male" checked="check">

<label for="male">Male</label>

<input type="radio" id="female" name="gender\_update" value="Female">

<label for="female">Female</label>

<input type="radio" id="others" name="gender\_update" value="Others">

<label for="others">Others</label>

</fieldset>

</div>

<div data-role="fieldcontain">

<label for="dob\_update">Date Of Birth</label>

<input type="date" name="dob\_update" id="dob\_update" data-mini="true" placeholder="dd-mm-yyyy">

</div>

<div data-role="fieldcontain">

<label for="religion\_update">Religion</label>

<input type="text" data-mini="true" id="religion\_update" name="religion\_update">

</div>

<div data-role="fieldcontain">

<label for="mother\_tongue\_update">Mother Tongue</label>

<input type="text" data-mini="true" id="mother\_tongue\_update" name="mother\_tongue\_update">

</div>

<div data-role="fieldcontain">

<label for="address\_update">Address</label>

<input type="text" data-mini="true" id="address\_update" name="address\_update">

</div>

<div data-role="fieldcontain">

<label for="pincode\_update">Pincode</label>

<input type="text" data-mini="true" id="pincode\_update" name="pincode\_update">

</div>

<div data-role="fieldcontain">

<fieldset data-role="controlgroup" data-mini="true" data-type="horizontal">

<legend>Education</legend>

<input type="checkbox" id="ten\_update" name="ten\_update" onChange="$(this).val(this.checked? '1': '0');">

<label for="ten\_update">Tenth</label>

<input type="checkbox" id="twe\_update" name="twe\_update" onChange="$(this).val(this.checked? '1': '0');">

<label for="twe\_update">+2</label>

<input type="checkbox" id="ug\_update" name="ug\_update" onChange="$(this).val(this.checked? '1': '0');">

<label for="ug\_update">Under Graduate</label>

<input type="checkbox" id="pg\_update" name="pg\_update" onChange="$(this).val(this.checked? '1': '0');">

<label for="pg\_update">Post Graduate</label>

<input type="checkbox" id="dr\_update" name="dr\_update" onChange="$(this).val(this.checked? '1': '0');">

<label for="dr\_update">Doctrate</label>

</fieldset>

</div>

<div data-role="fieldcontain">

<label for="work\_update">Work Type</label>

<input type="text" data-mini="true" id="work\_update" name="work\_update">

</div>

<input type="button" id="updatebutton" name="updatebutton" value="Submit" data-mini="true" data-icon="check">

</form>

</div>

</div>

<!-- delete page -->

<div data-role="page" id="deletepage">

<div data-role="header">

<a href="#homepage" id="back" data-role="button" data-icon="carat-l" data-mini="true" data-transition="flip">Back</a>

<h2>Delete</h2>

</div>

<div data-role="main" class="ui-content">

<form id="deleteform">

<div data-role="fieldcontain">

<label for="uid\_delete"><div id="uiddeleteerror"></div> UID</label>

<input type="text" data-mini="true" id="uid\_delete" name="uid\_delete">

</div>

<input type="button" id="deletebutton" name="deletebutton" value="Submit" data-mini="true" data-icon="check">

</form>

</div>

</div>

<!-- report page -->

<div data-role="page" id="reportpage">

<div data-role="header">

<a href="#homepage" id="back" data-role="button" data-icon="carat-l" data-mini="true" data-transition="flip">Back</a>

<h2>Report</h2>

</div>

<div data-role="main" class="ui-content">

<form id="reportform">

<div data-role="fieldcontain">

<fieldset data-role="controlgroup" data-mini="true" data-type="horizontal">

<legend>Select Entities</legend>

<input type="checkbox" id="idtablereport" name="idtablereport" onChange="$(this).val(this.checked? '1': '0');">

<label for="idtablereport">Id Table</label>

<input type="checkbox" id="detailtablereport" name="detailtablereport" onChange="$(this).val (this.checked? '1': '0');">

<label for="detailtablereport">Detail Table</label>

<input type="checkbox" id="addresstablereport" name="addresstablereport" onChange="$(this).val(this.checked? '1': '0');">

<label for="addresstablereport">Address Table</label>

<input type="checkbox" id="educationtablereport" name="worktablereport" onChange="$(this).val(this.checked? '1': '0');">

<label for="educationtablereport">Education Table</label>

<input type="checkbox" id="worktablereport" name="worktablereport" onChange="$(this).val(this.checked? '1': '0');">

<label for="worktablereport">Work Table</label>

</fieldset>

</div>

<input type="button" id="reportbutton" name="reportbutton" value="Submit" data-mini="true" data-icon="check">

</form>

<table data-role="table" class="ui-responsive ui-table">

<thead id="theadreport"></thead>

<tbody id="tbodyreport"></tbody>

</table>

</div>

</div>

</body>

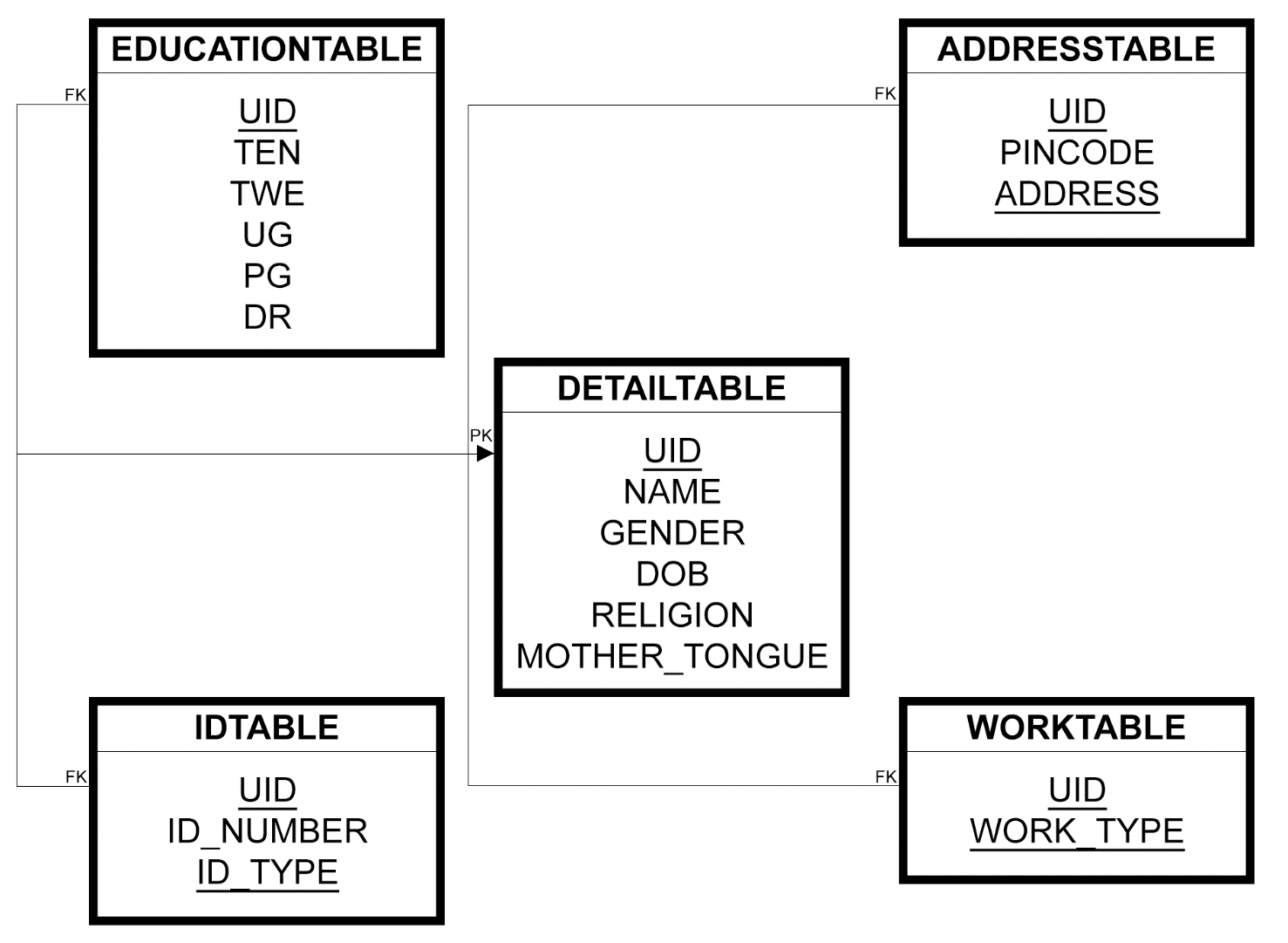
</html>

# PROJECT IMPLEMENTATION

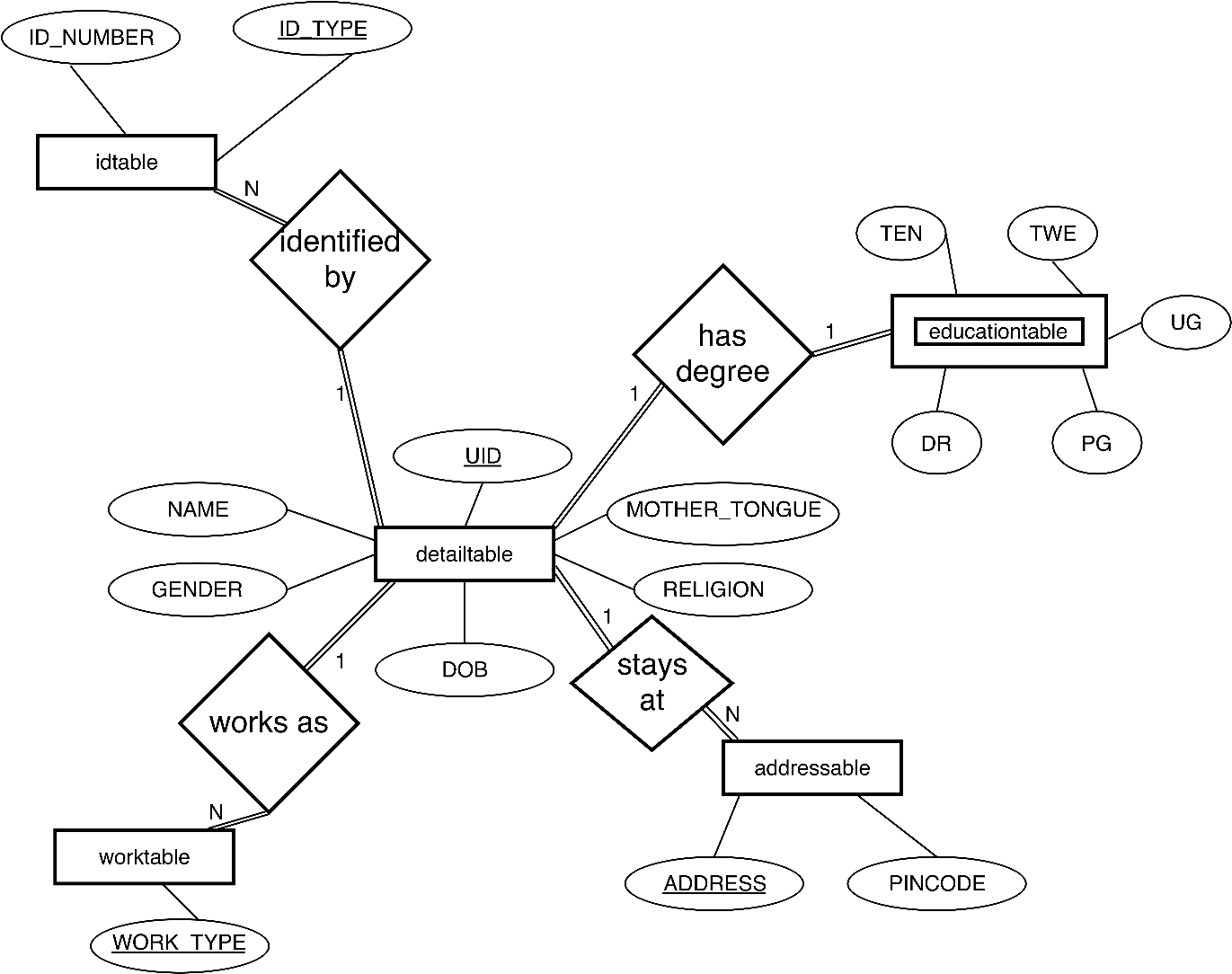
## Database Implementation

The tables are normalized to 4NF to reduce redundancy of data with multi valued dependencies. The tables are joined to display the values such that there are no spurious records. The frontend limits the data being entered as it only allows single value dependencies.

The UML diagram of the database is



The ER diagram for the database is



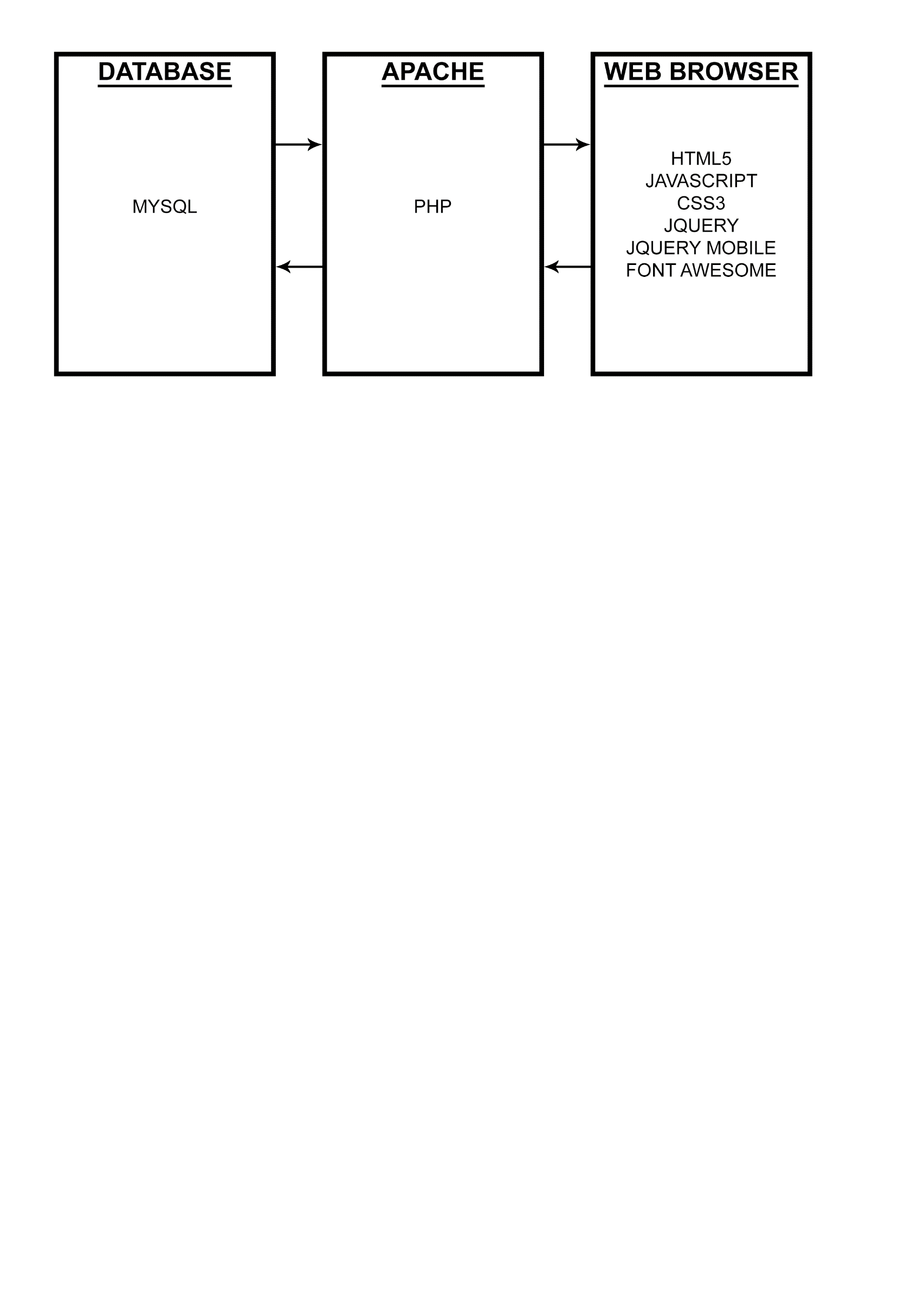
## Project Implementation

The project is implemented as a 3-tier layer. The layers are

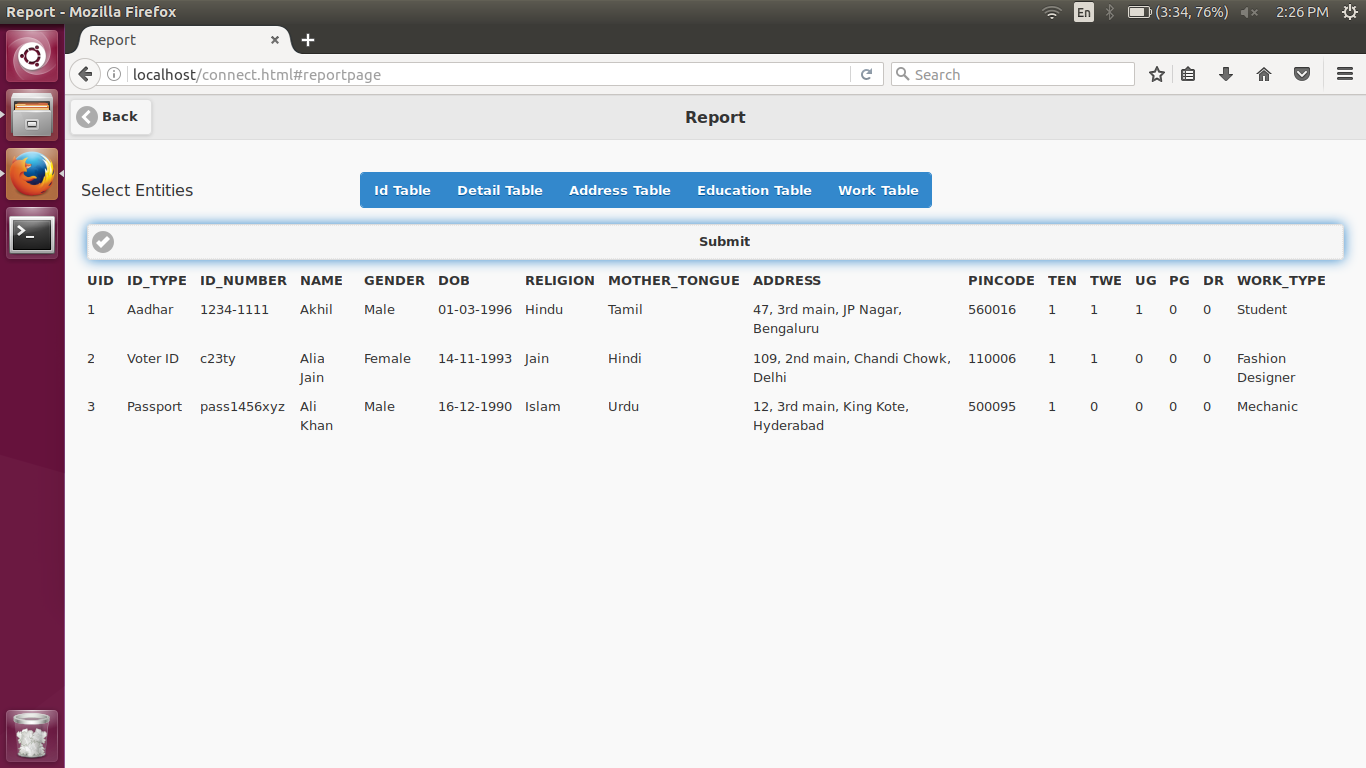
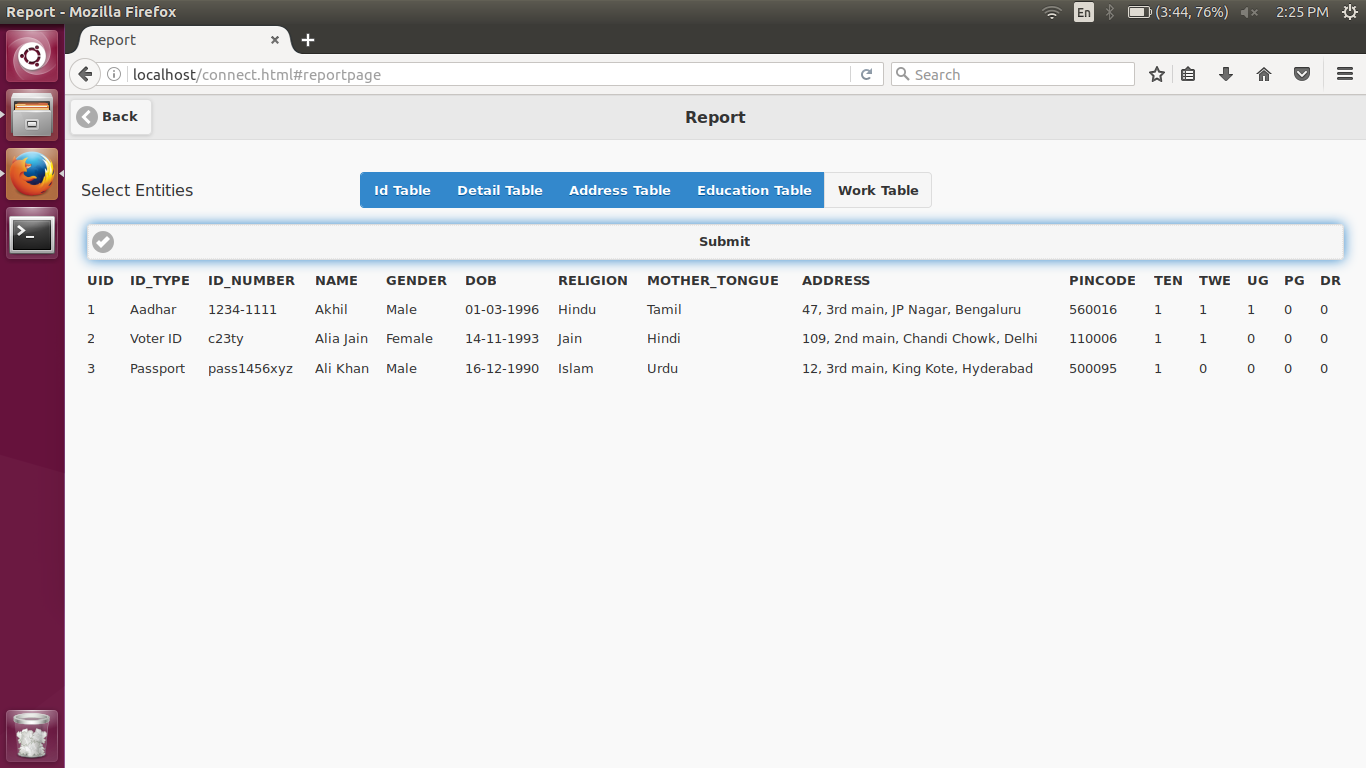
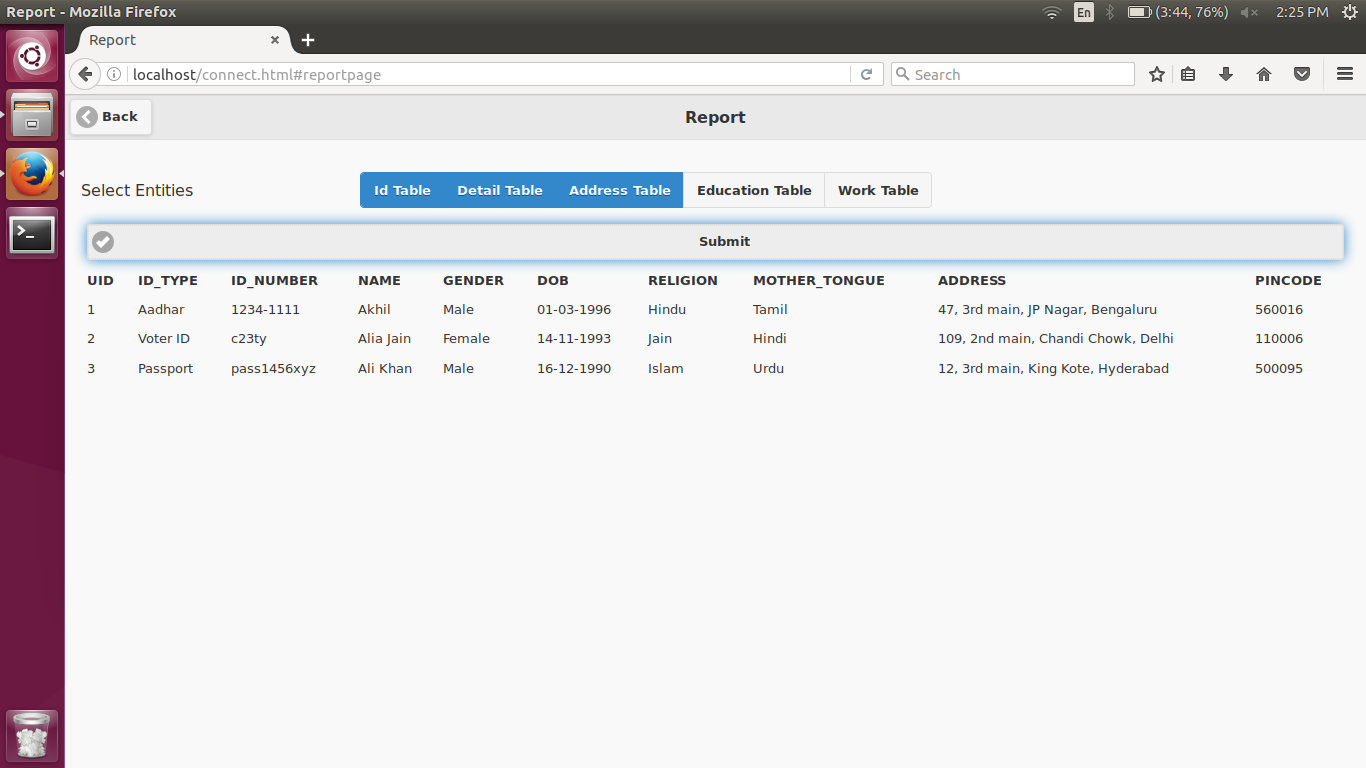
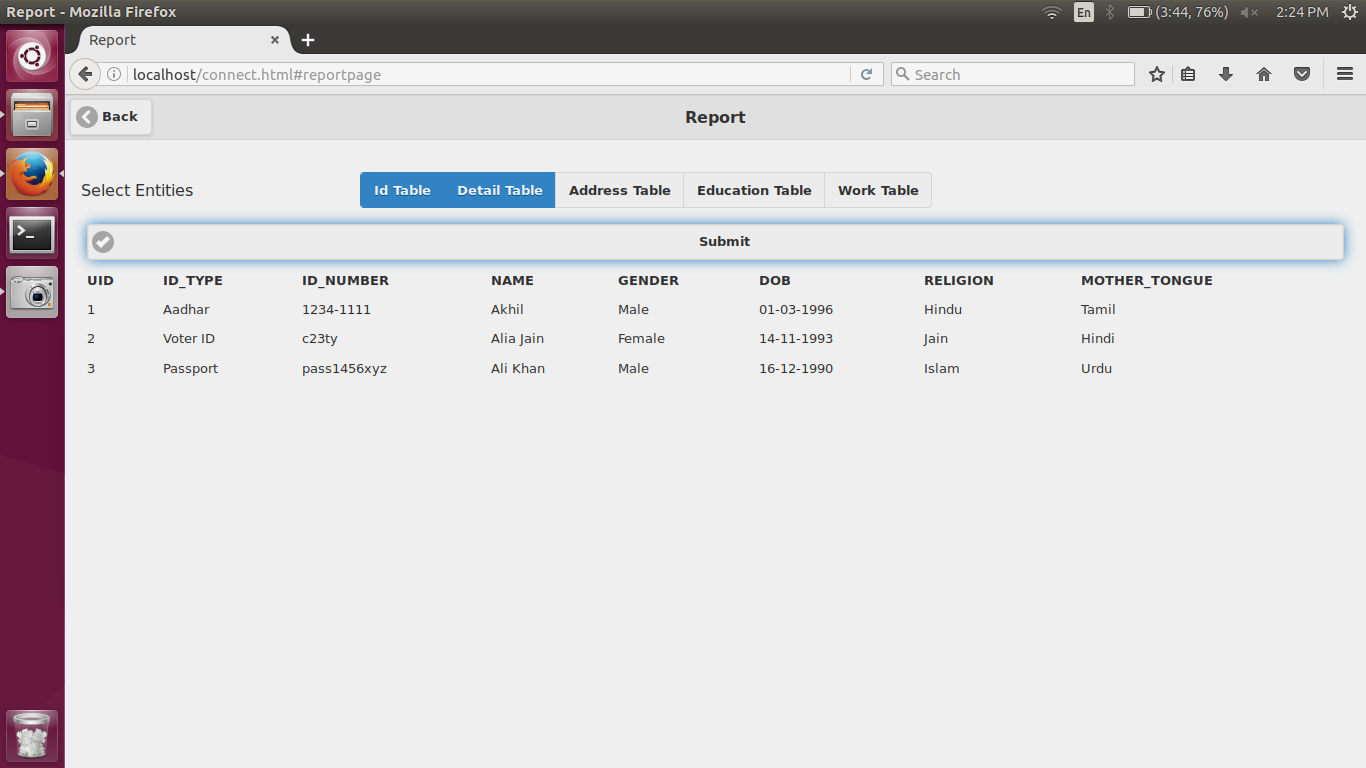
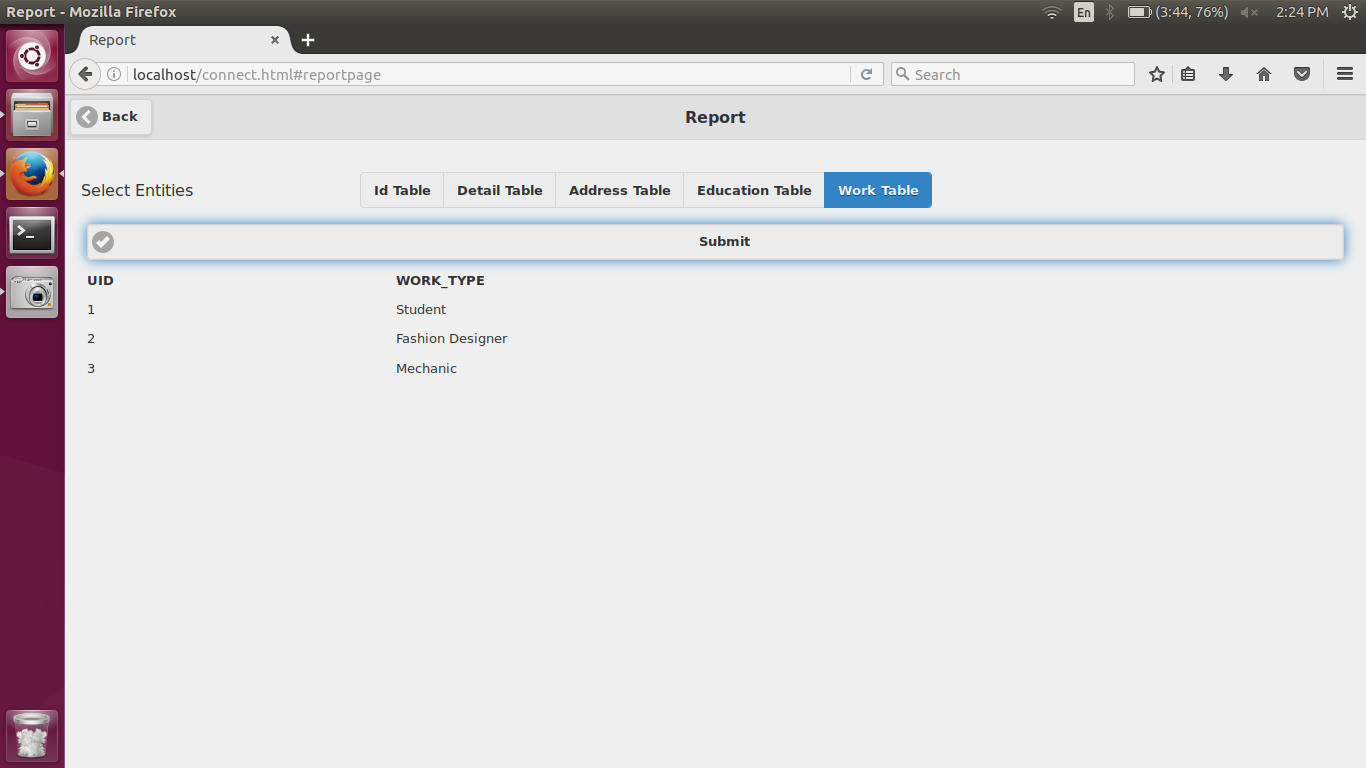
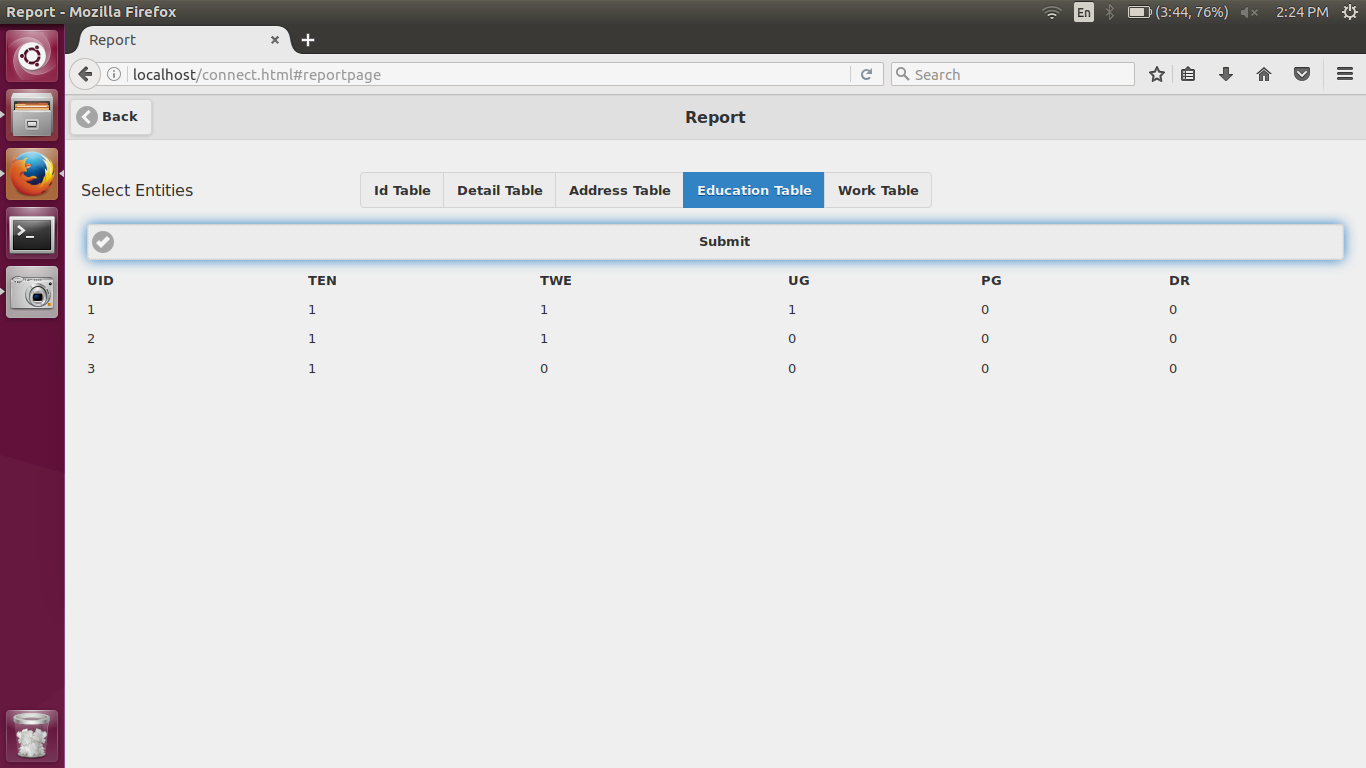
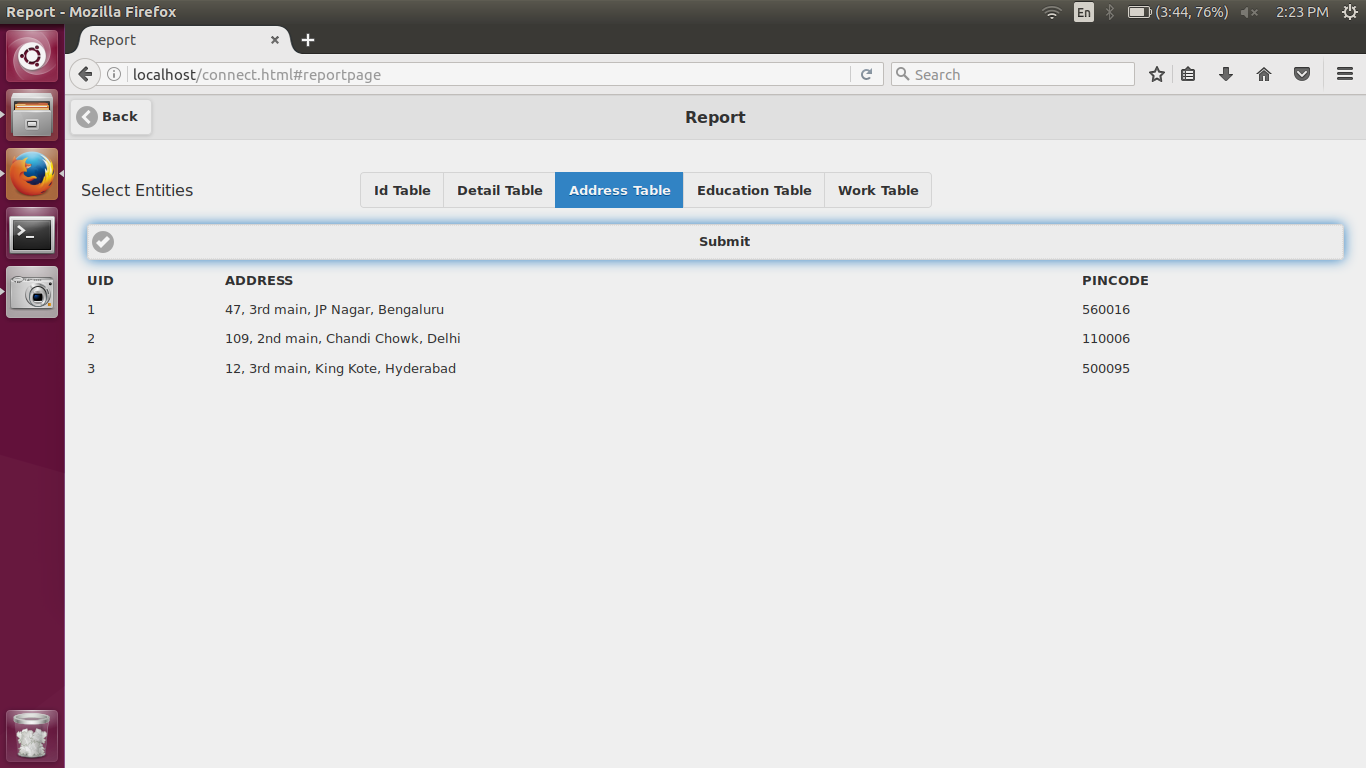
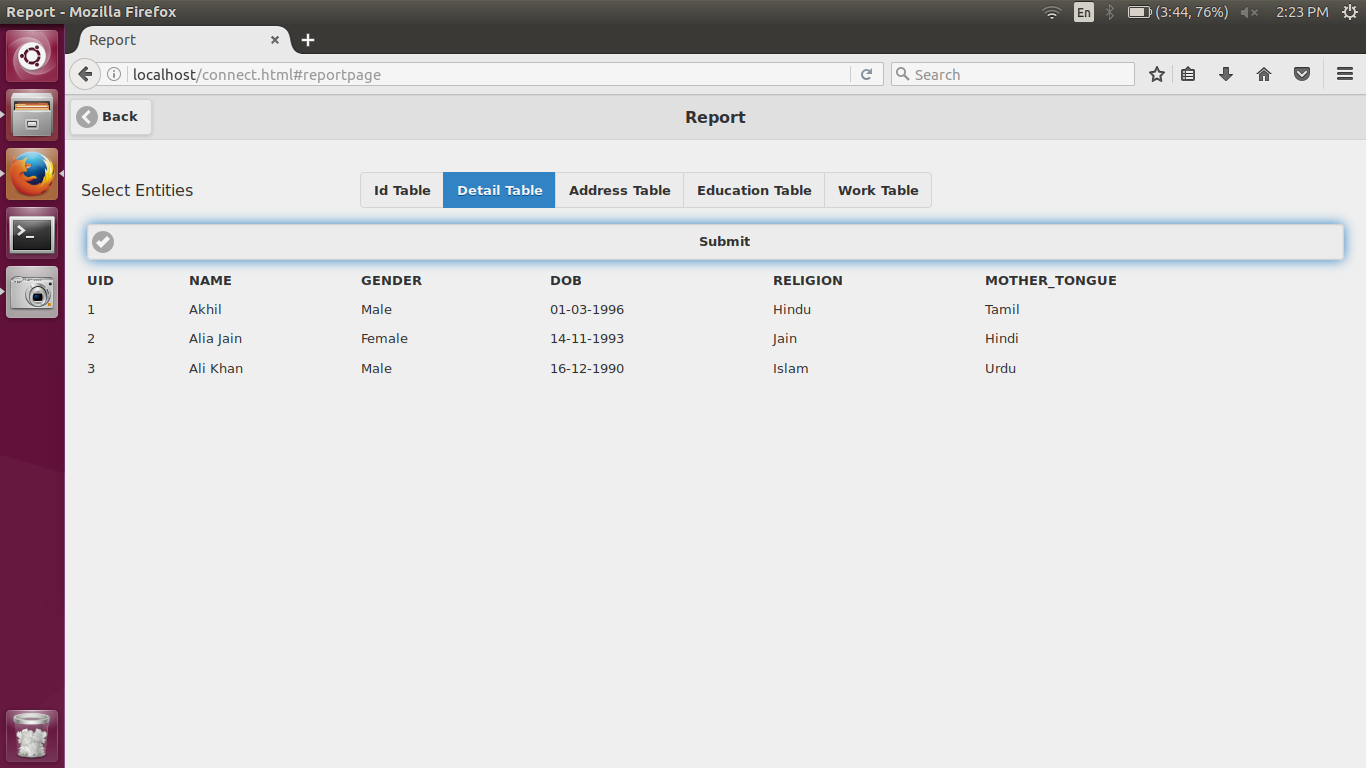
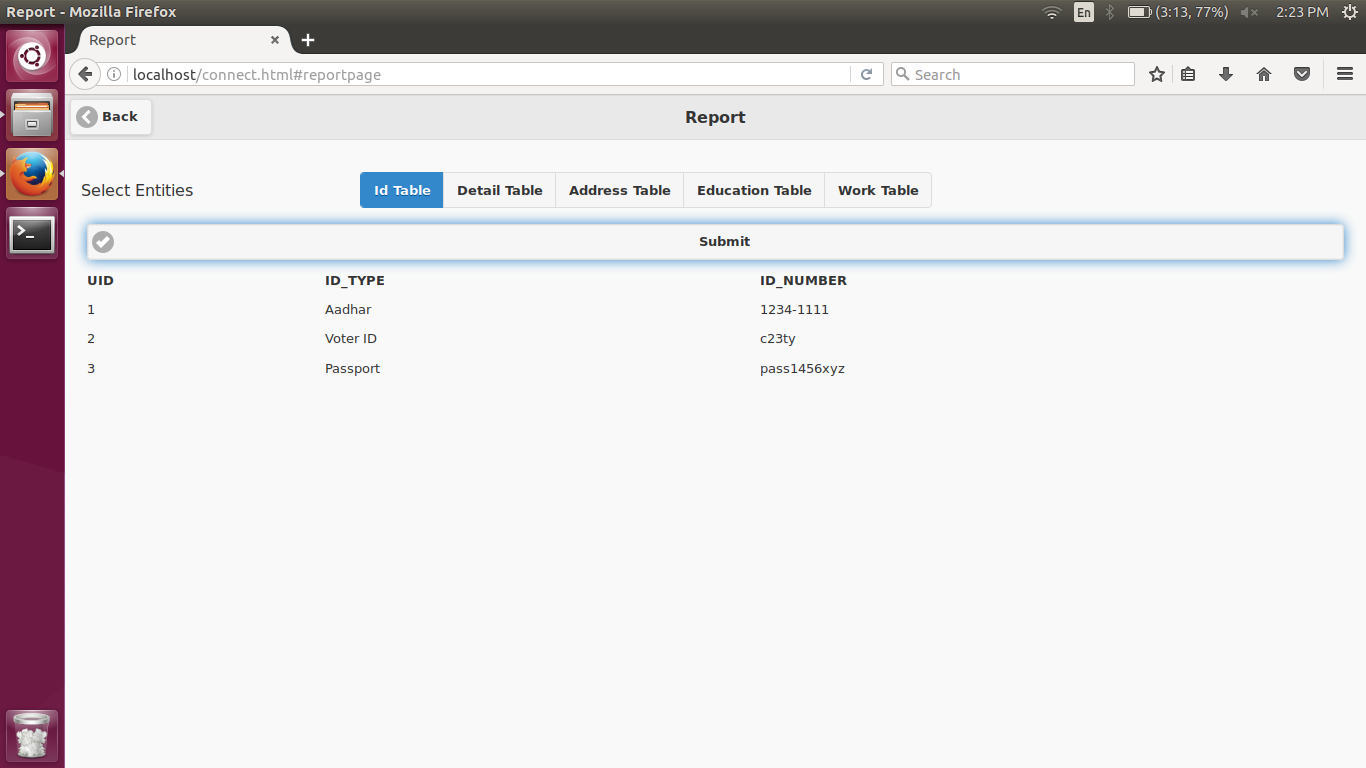
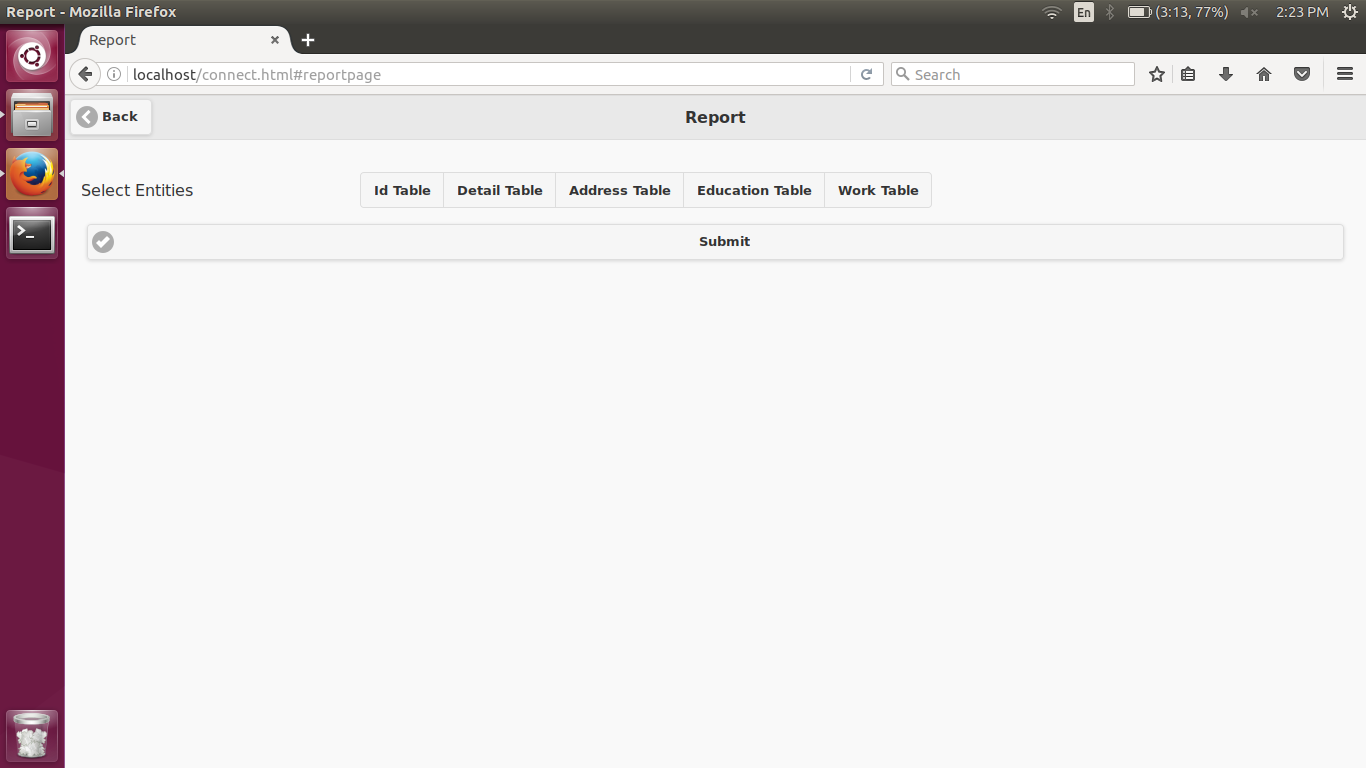
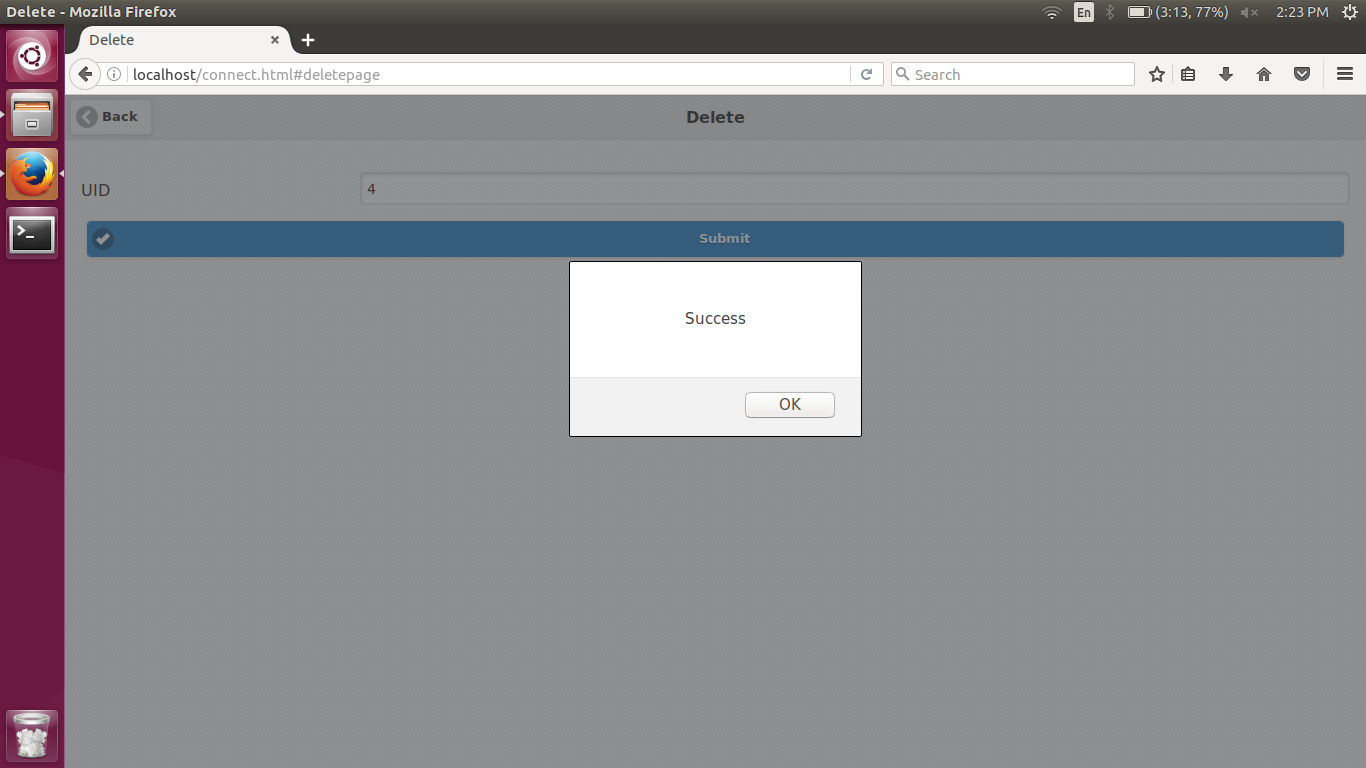
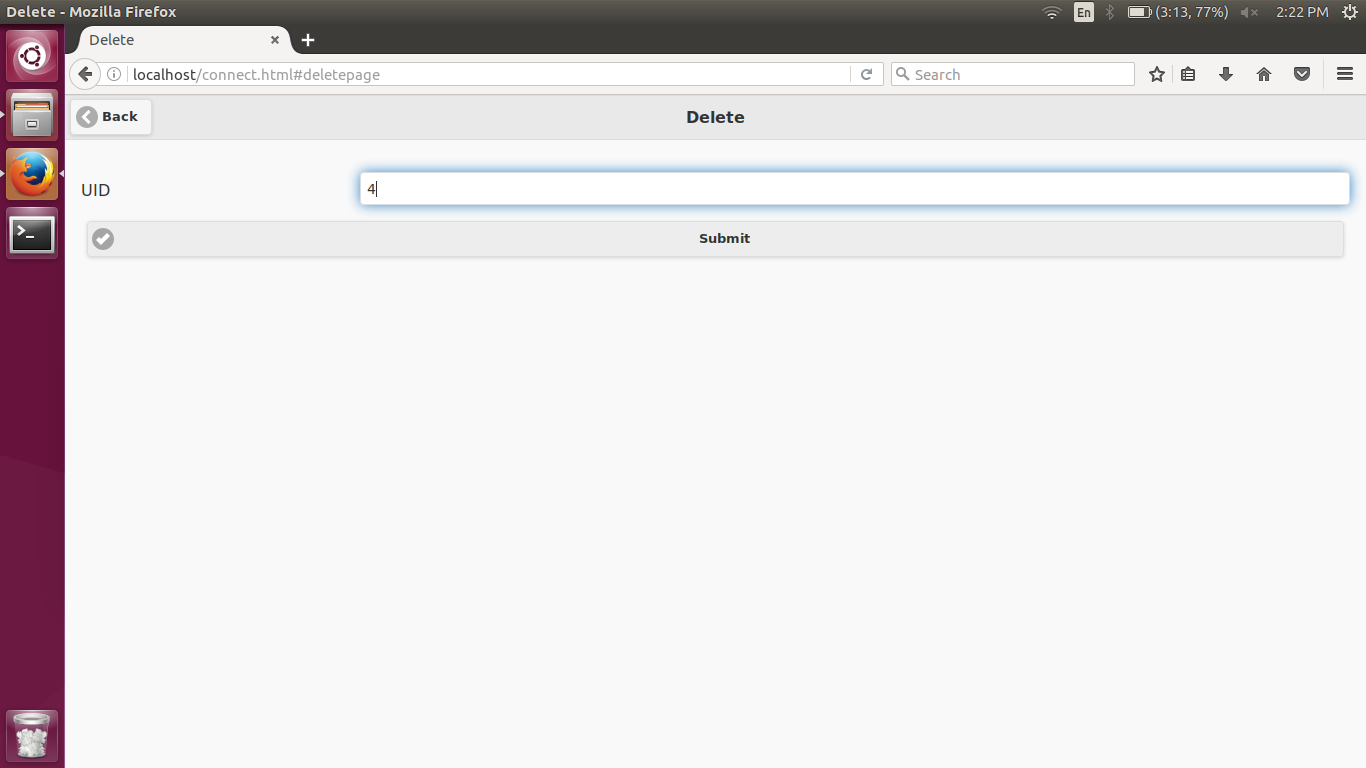
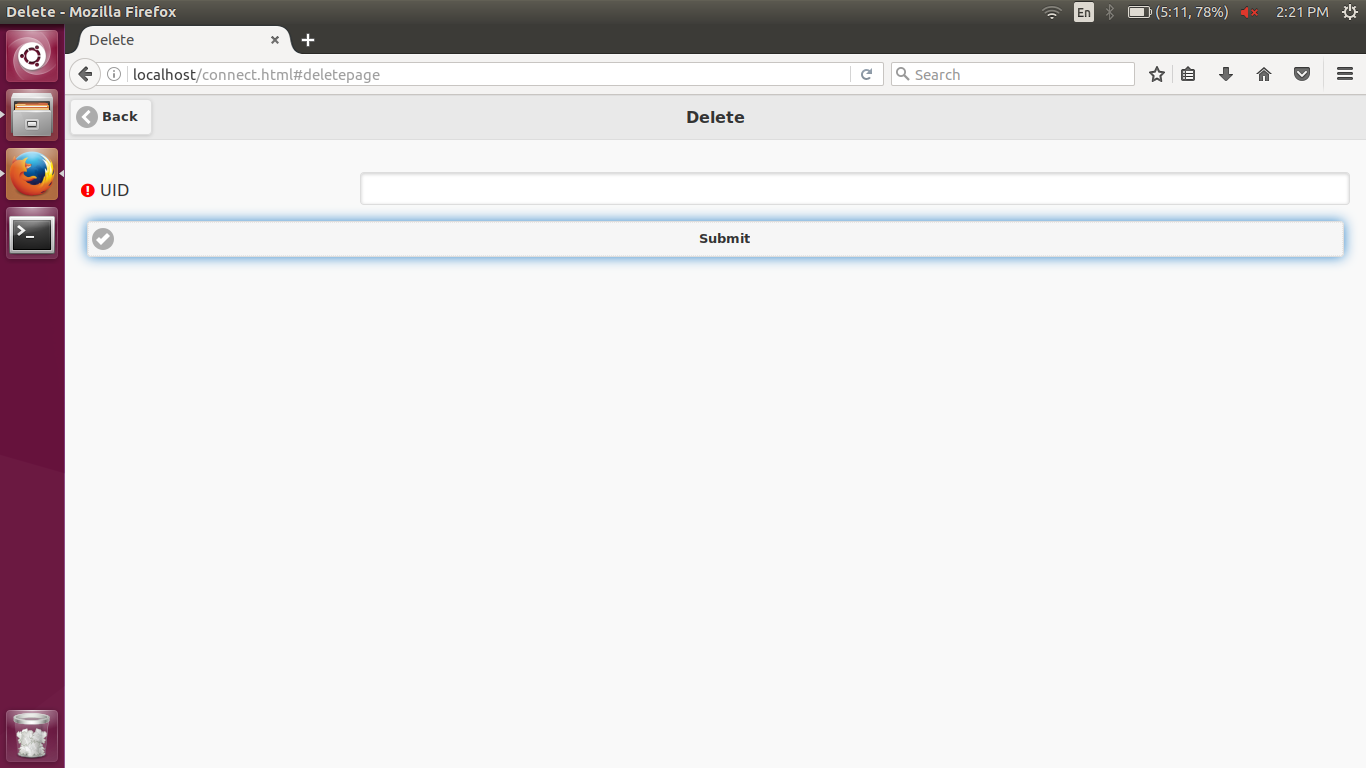
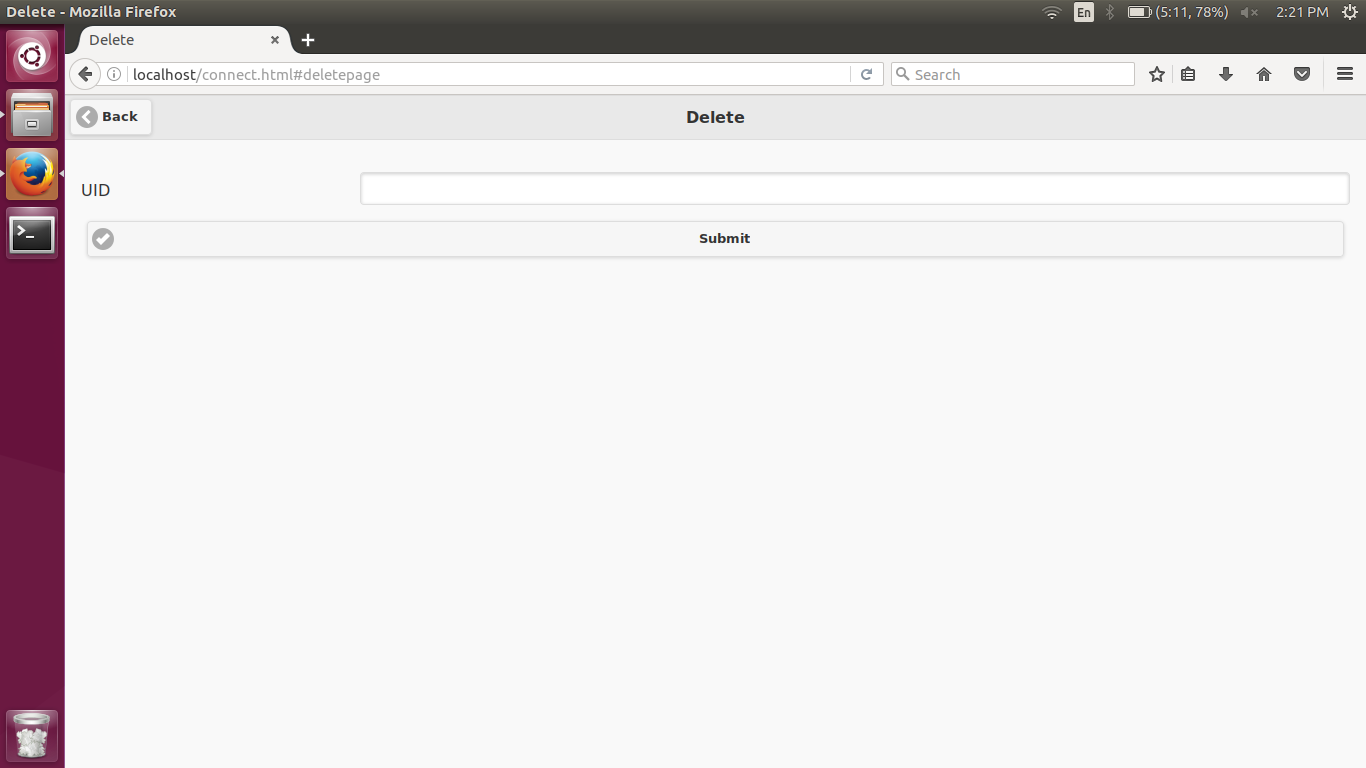
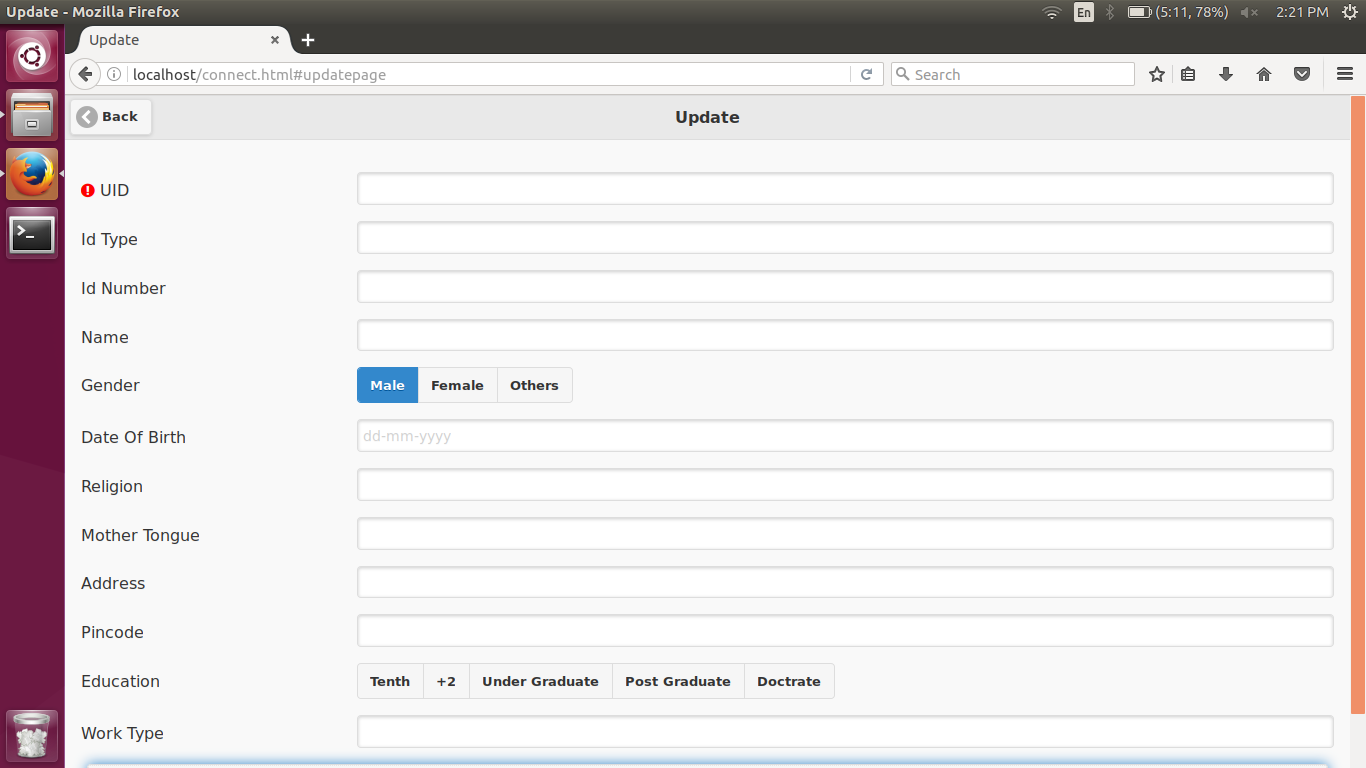
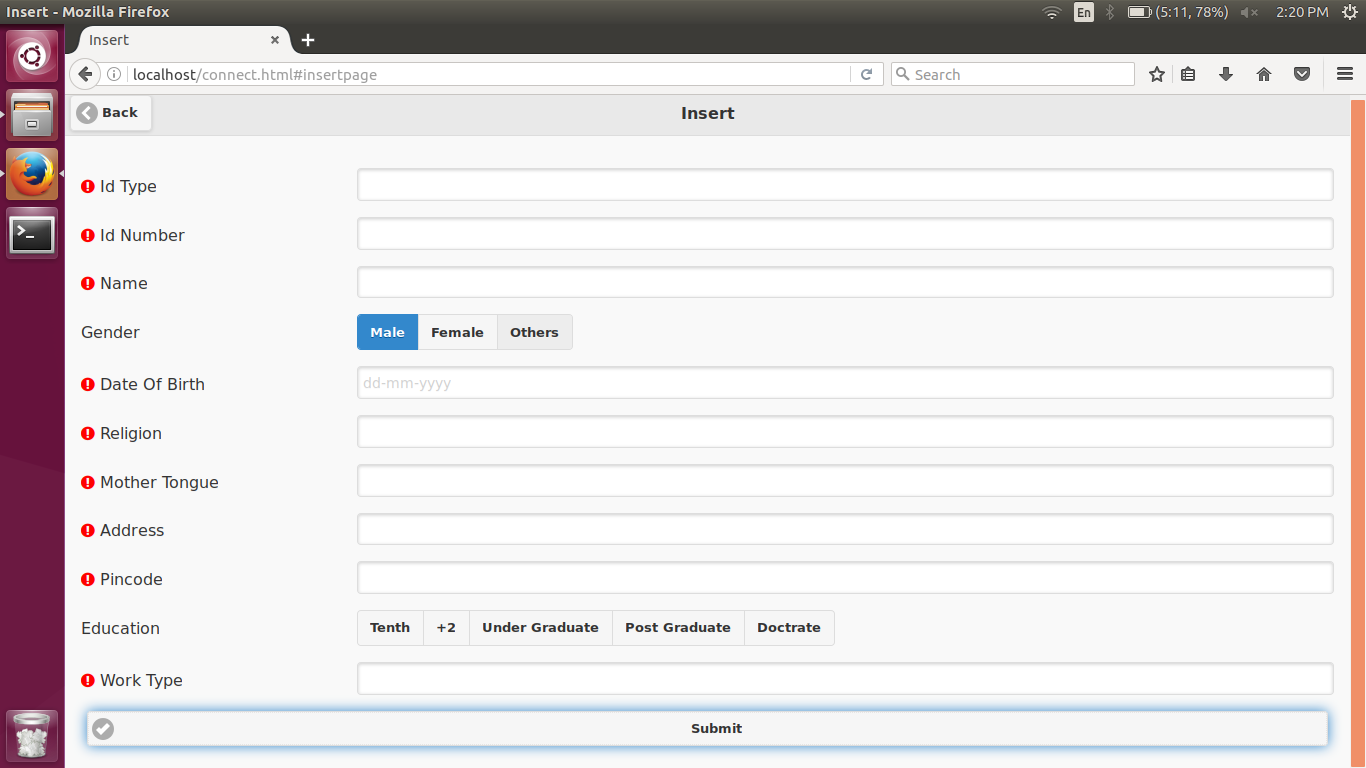
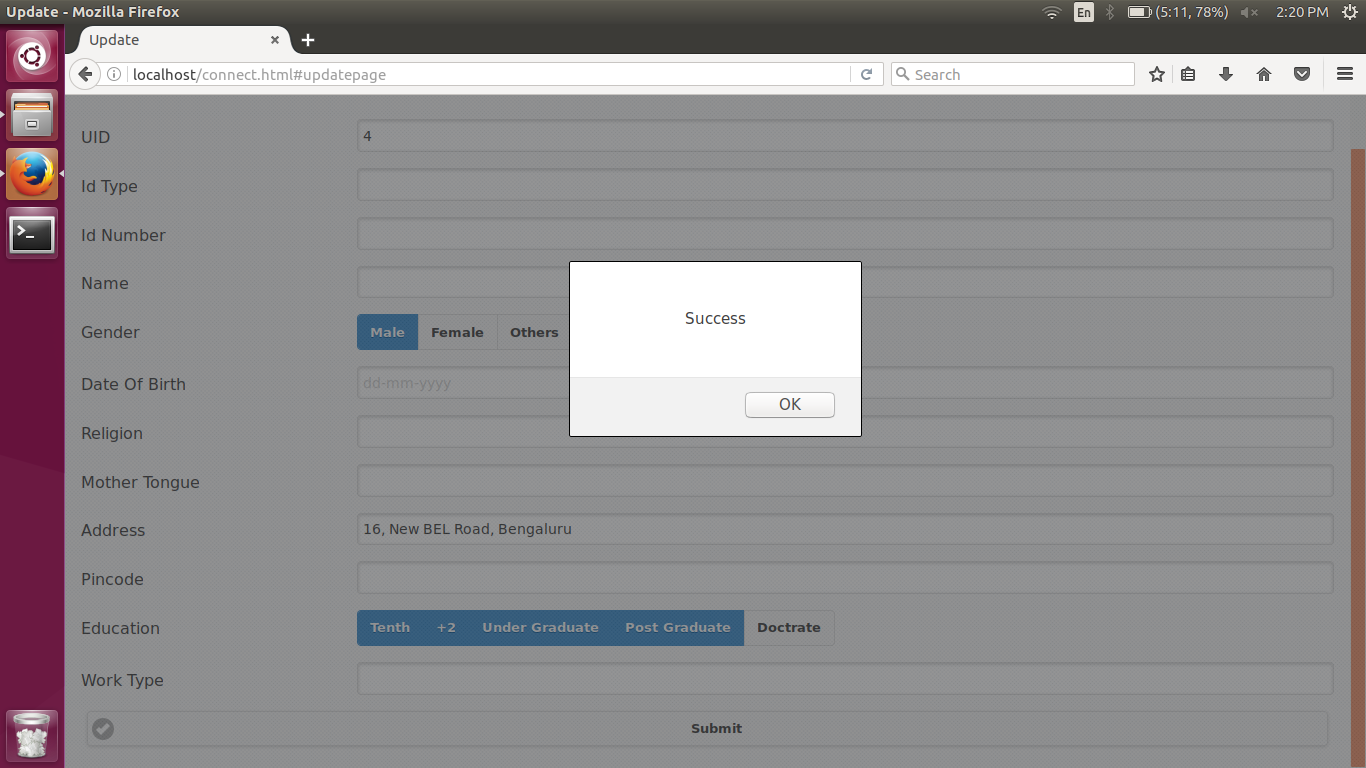
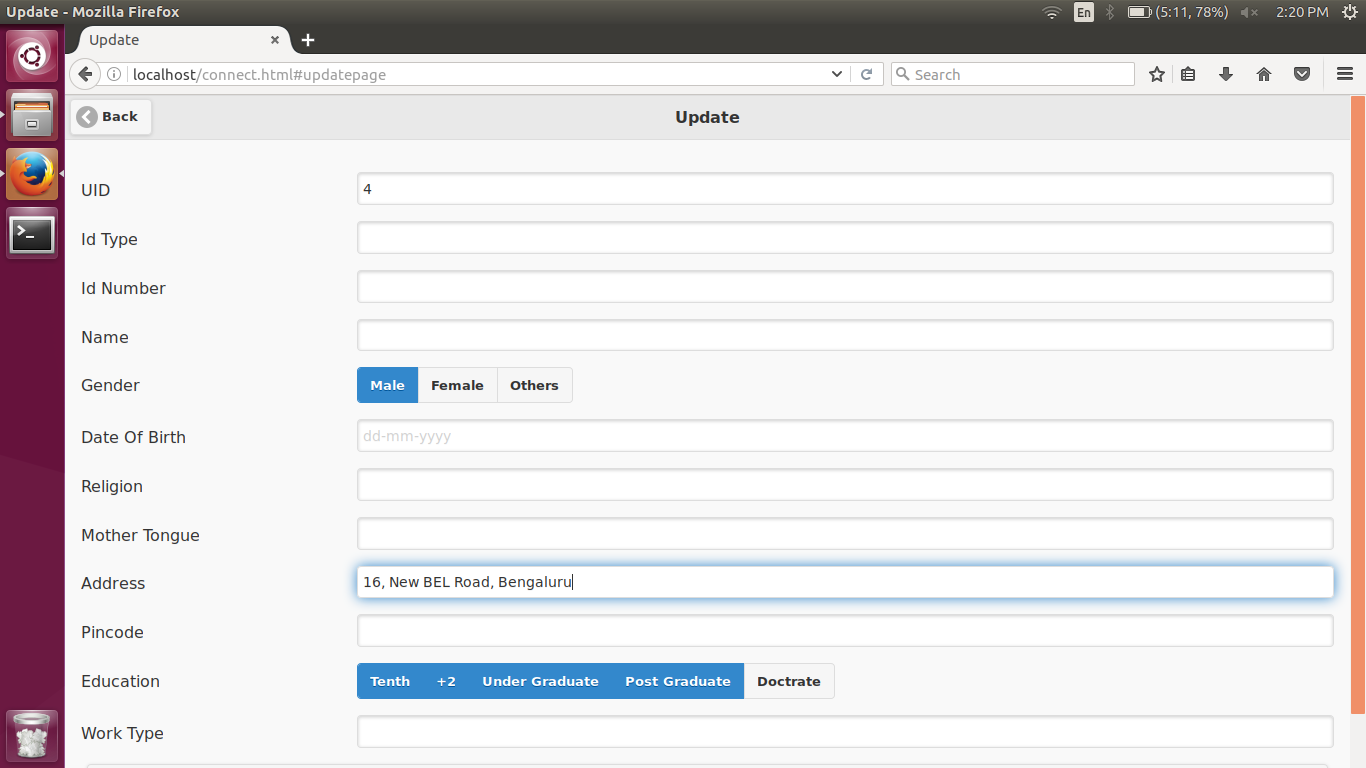
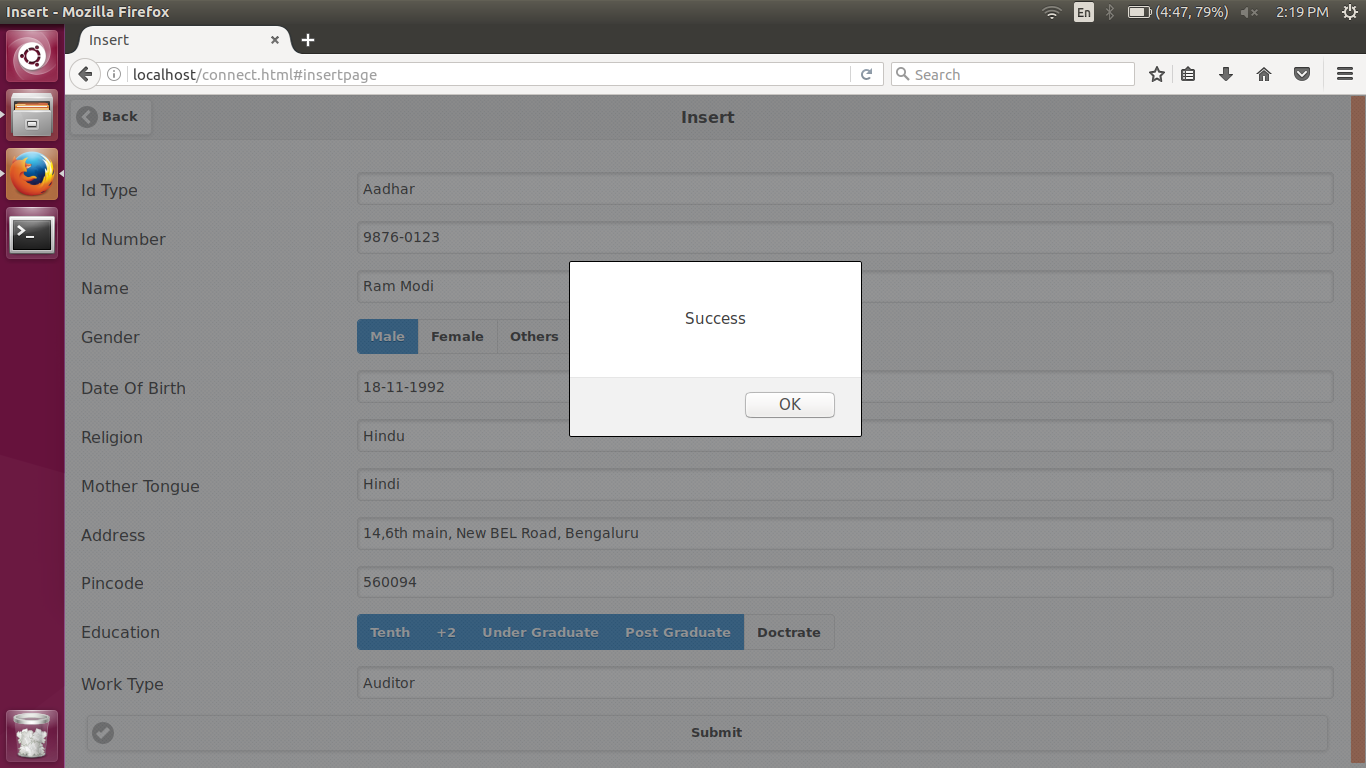
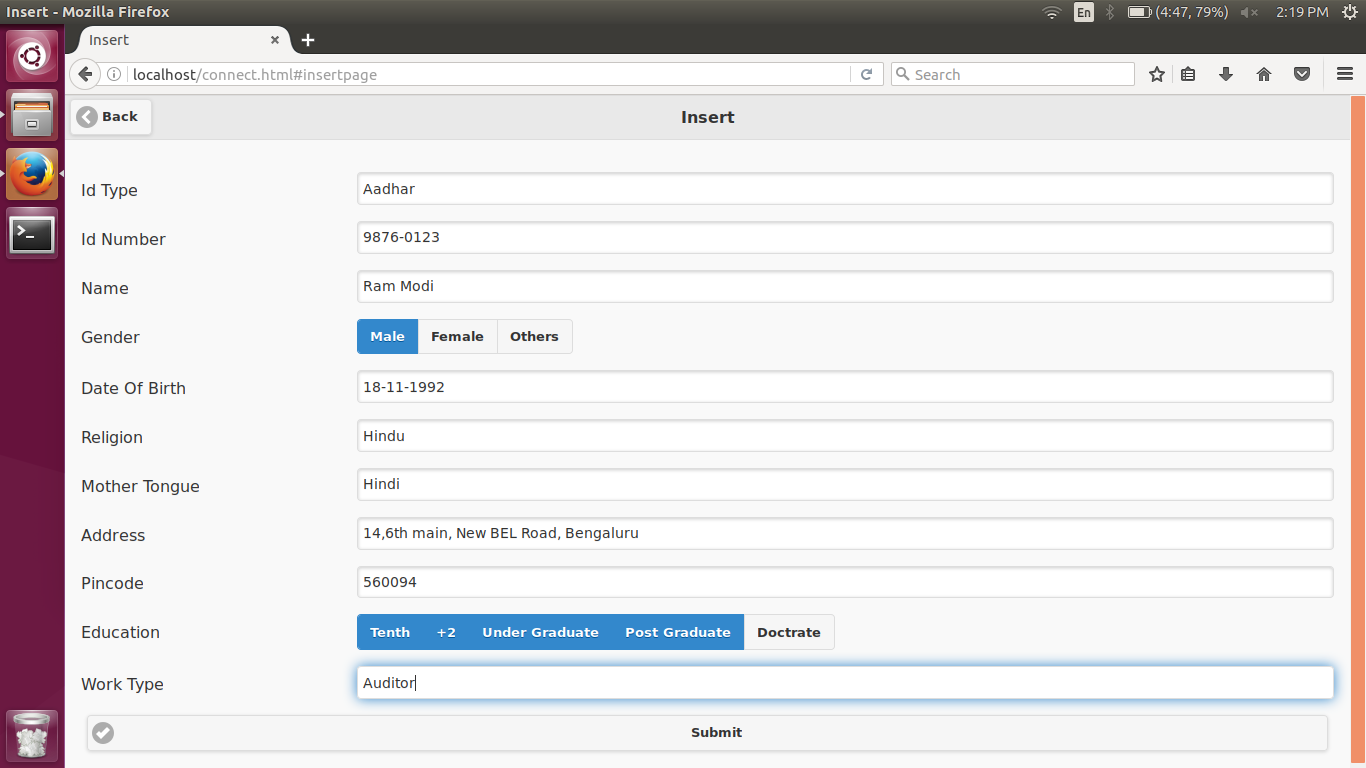
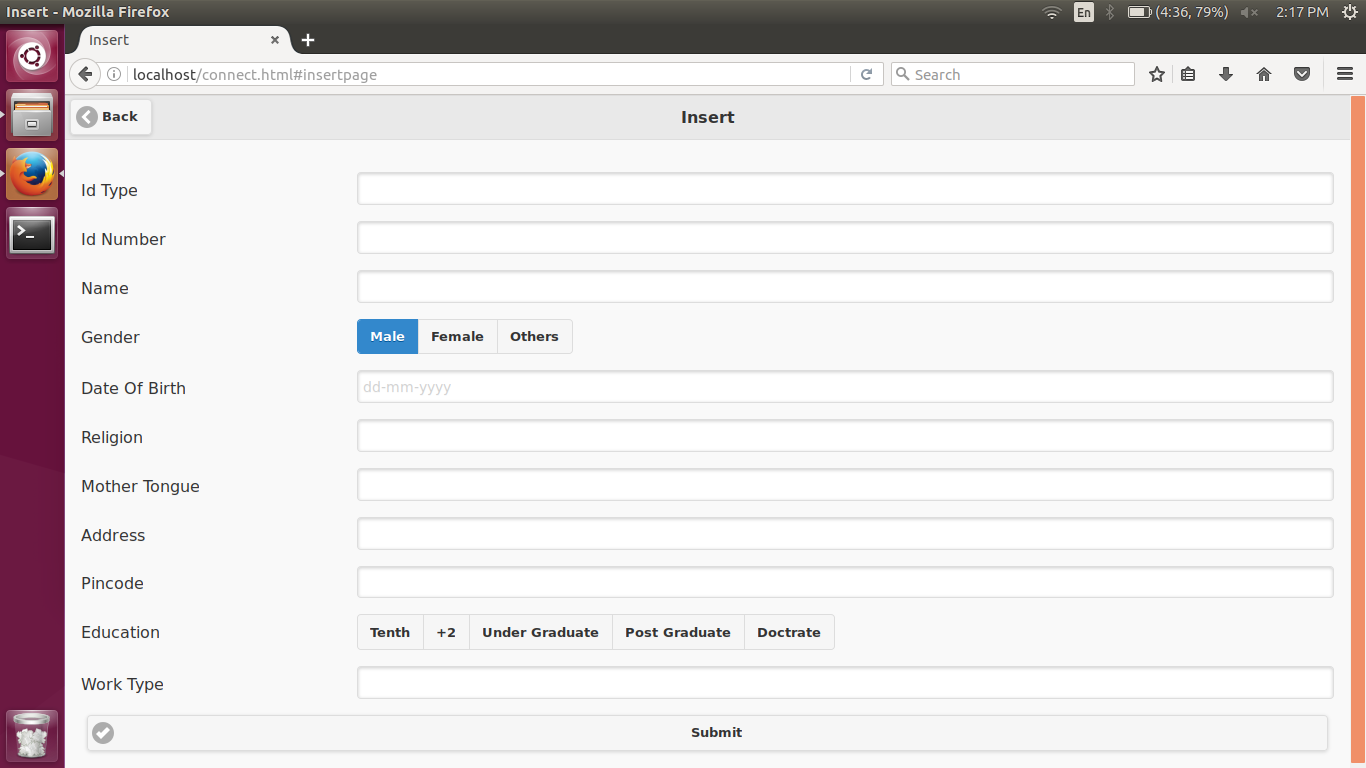
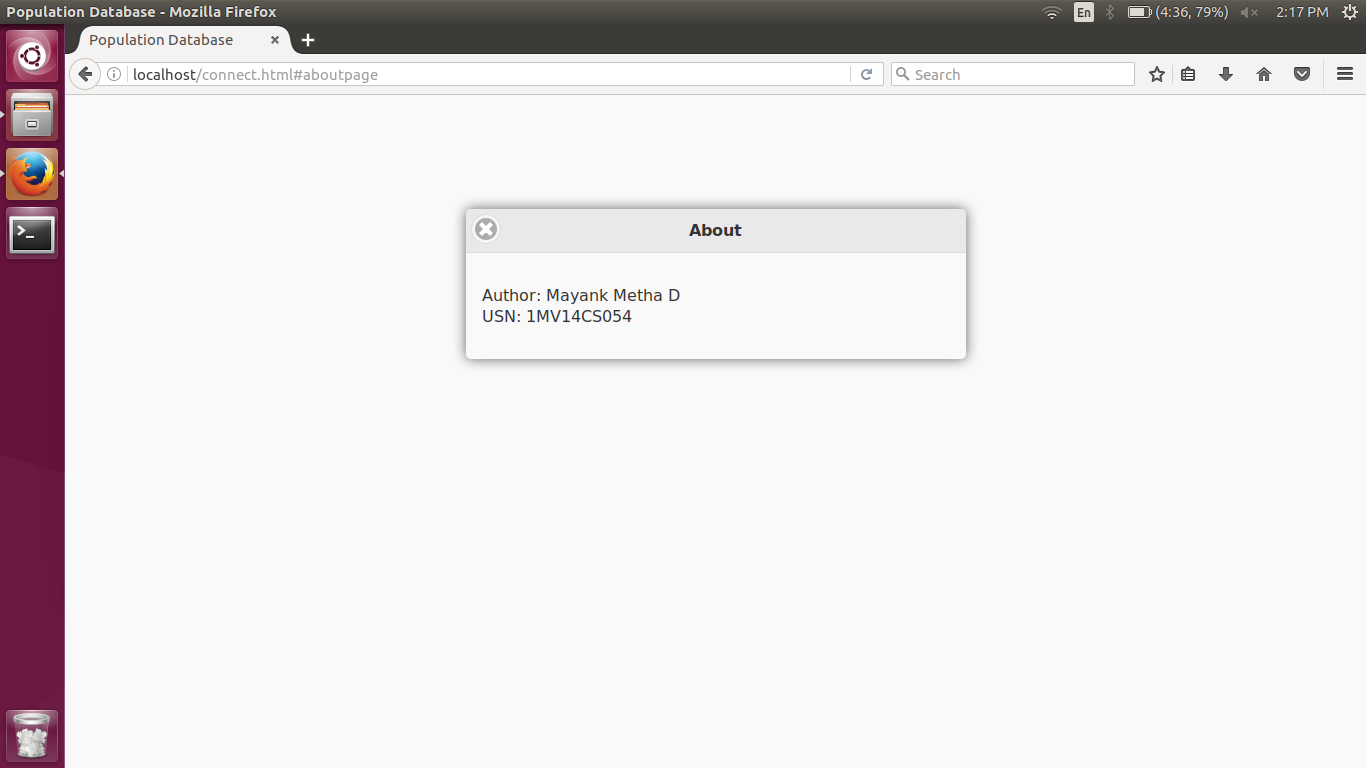
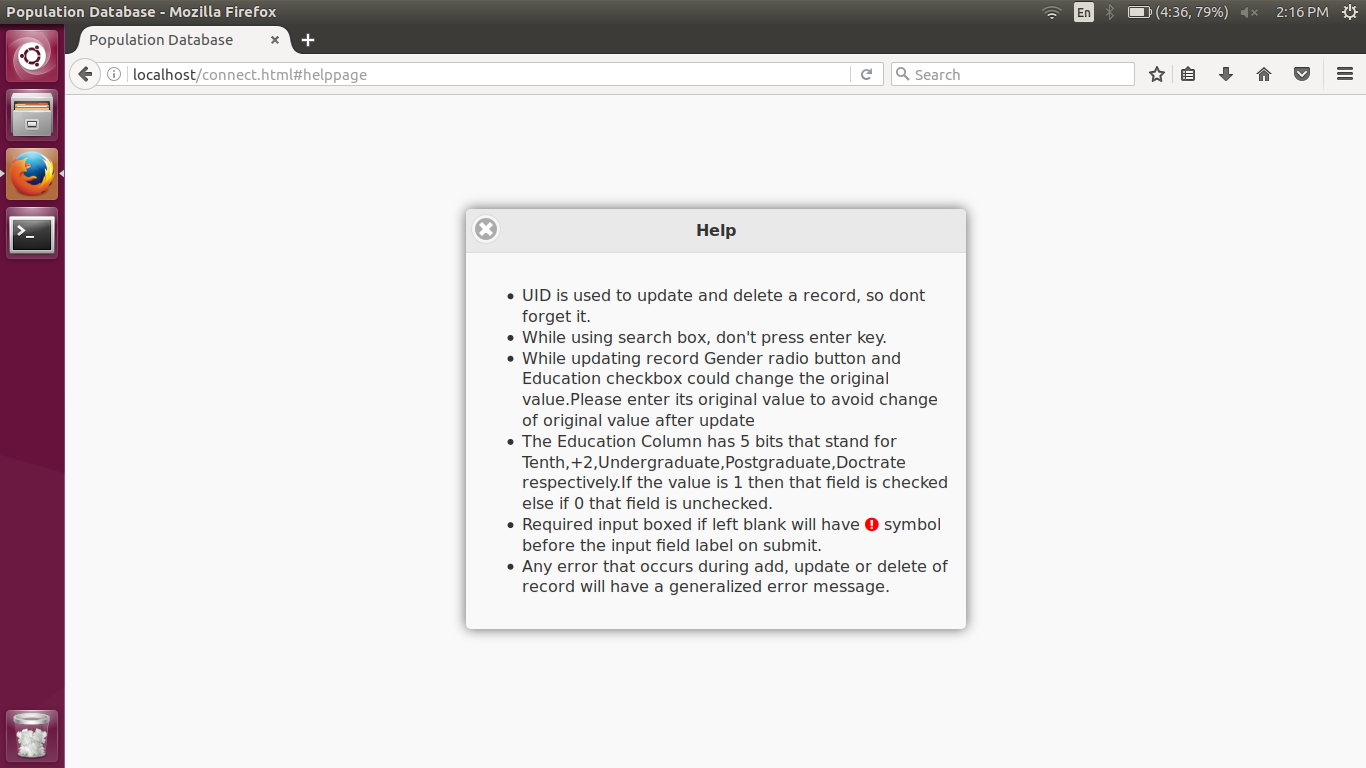
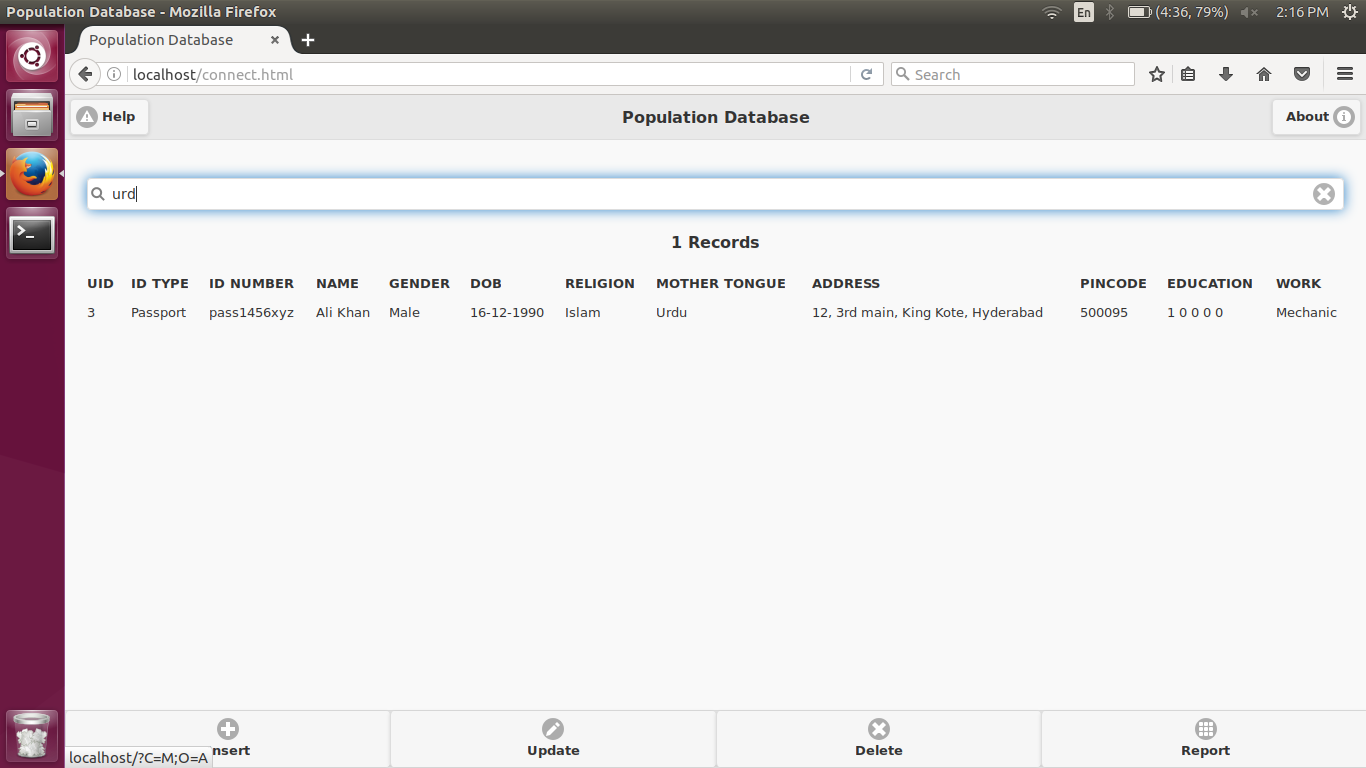
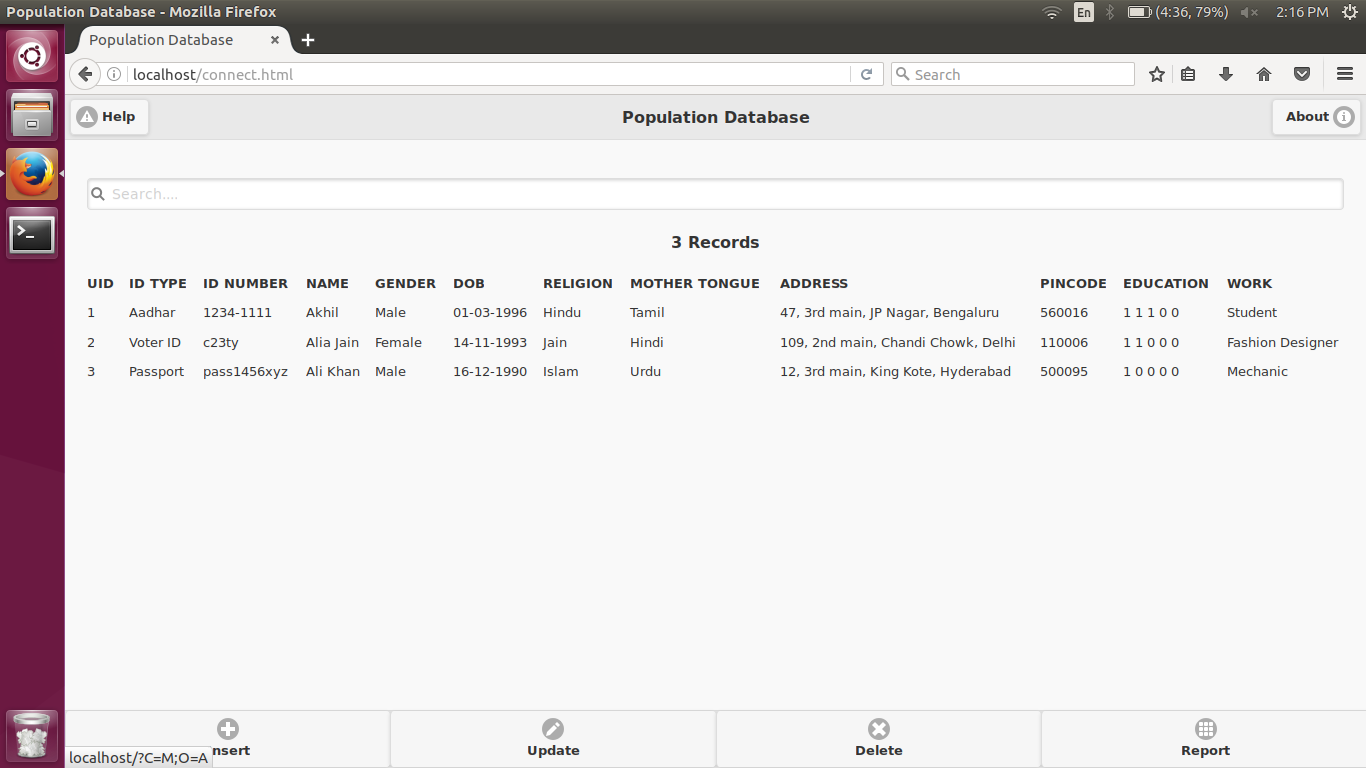
1. Database layer
2. Apache layer
3. Web browser layer

The Database layer contains the tables where data is stores. The Apache layer contains the PHP scripts that are emulated as website server. The Web browser layer contains the HTML5 code and JavaScript files along with 3rd party API like the JQuery, JQuery Mobile, FontAwesome in the form of minified JavaScript, CSS3, Icons (png & svg) and font.

The project implementation diagram is



# SCREENSHOTS



# BIBILIOGRAPHY

## Books

* Database Systems 6th Edition by Ramez Elmasri, Shamkant B. Navathe
* JQuery Mobile Up and Running by Maximiliano Firtman
* Mobile JavaScript Application Development by Adrian Kosmaczewski

## Websites

* http://www.w3schools.com/
* http://demos.jquerymobile.com/1.4.5/
* http://php.net/docs.php
* https://httpd.apache.org/docs/2.4/
* http://dev.mysql.com/doc/
* https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu