## Chapter 4

## Content and Construction

The dbPedigree is a relational database of pedigree associated to a particular genetic defect, developed using the basics of MySQL, a relational database engine, PHP, a server-side scripting language and HTML a scripting language. The whole scheme of the database was driven by a web site which allows the user to query through the content of the site; query can be a disease name or a gene HGNC symbol. The data actually resides in a database created using MySQL, and for that content to be pulled from the database dynamically; we need to connect the genetic information to disease information and pedigree-mutational data (Figure 4.1). Then a web page was created for user to view the output of their query on a regular web browser. The browser gives the link to pedigree.

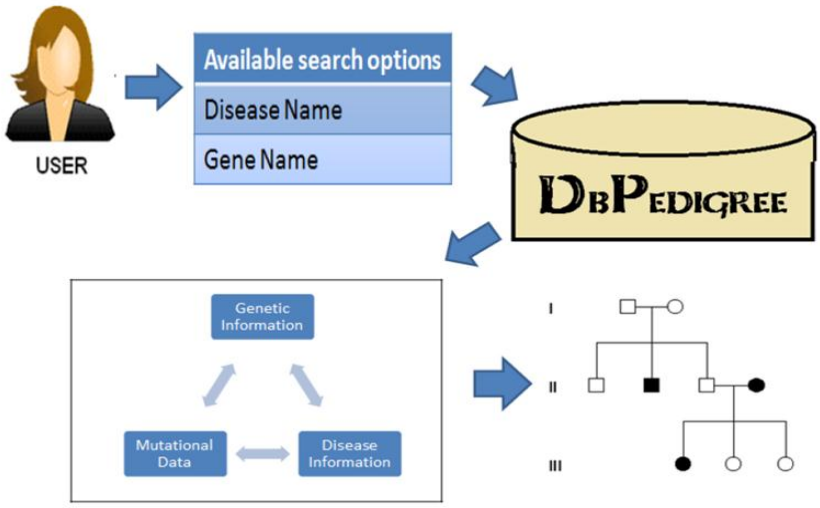



**Figure 4.1 The backend of dbPedigree.**

When a user queries the database driven web site, many process occurs. The process has been discussed below (Figure 4.2)

1. The request is send to the web page using a web address.
2. The web server software (Apache) recognizes and requests file (PHP script), to fires a PHP interpreter to execute the code contained in the file.
3. The PHP commands will then connect to the MySQL database and request the content that belongs in the web page.
4. The MySQL database responds by sending the requested content to the PHP script.

5. The PHP script stores the content into one or more PHP variables, and then uses echo statements to output the content as part of the web page.
6. The PHP interpreter finishes up by handing a copy of the HTML it has created to the web server.
7. The web server sends the HTML to the web browser as it would a plain HTML file, except that instead of coming directly from an HTML file, the page is the output provided by the PHP interpreter.

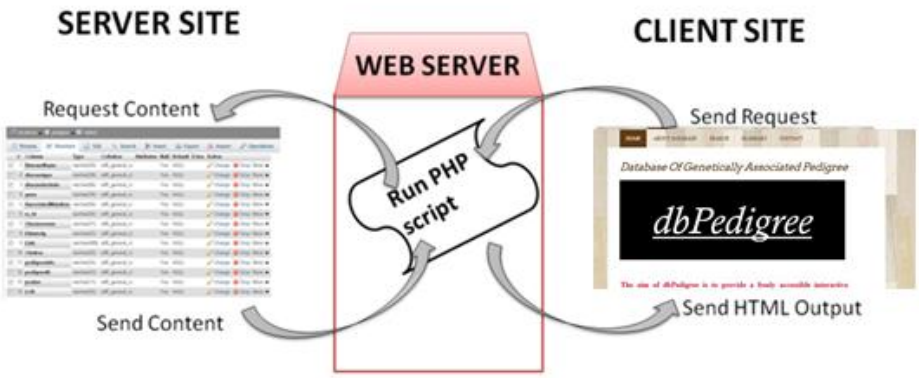



**Figure 4.2 Go-between webpage and database.**

So, at one end of the system user has the site, a web browser to request a page, and expects to receive a standard HTML document in return. This site is known as the **client site**. At the other end is the data of pedigree site, which sits in one or more tables in a MySQL database that understands only how to respond to SQL queries (commands).The database basically sits on the server and is called **server site**. The PHP scripting language is the go-between that speaks both languages. Figure 4.2. shows how the processor sends the page request and fetches the data from the MySQL database, then spits out dynamically as the nicely formatted HTML page that the browser expects.

### 3.1 Database Designing

The dbPedigree was developed keeping in mind the large number of genetic diseases and their pattern of inheritance. This is a relational database that has been curates manually by the authors. Database development was divided into three phases which are styled in detail.

1. Data Collection
2. Data Cleaning
3. Develop a Relational database and data entry

### 3.1.1 Data Collection

The primary disease information such as its disease sub category, gene and associated mutations were obtained from the various research articles. The reliability of the Information was confirmed from Online Mendelian Inheritance in Man (OMIM), which is online database containing vast information for various genetic human diseases. Although a list of genes related to particular genetic diseases were mined from OMIM using the keywords familial aggregation, twin tests, pedigree analysis. Around 87 disease hits were found as shown in Figure 4.2.

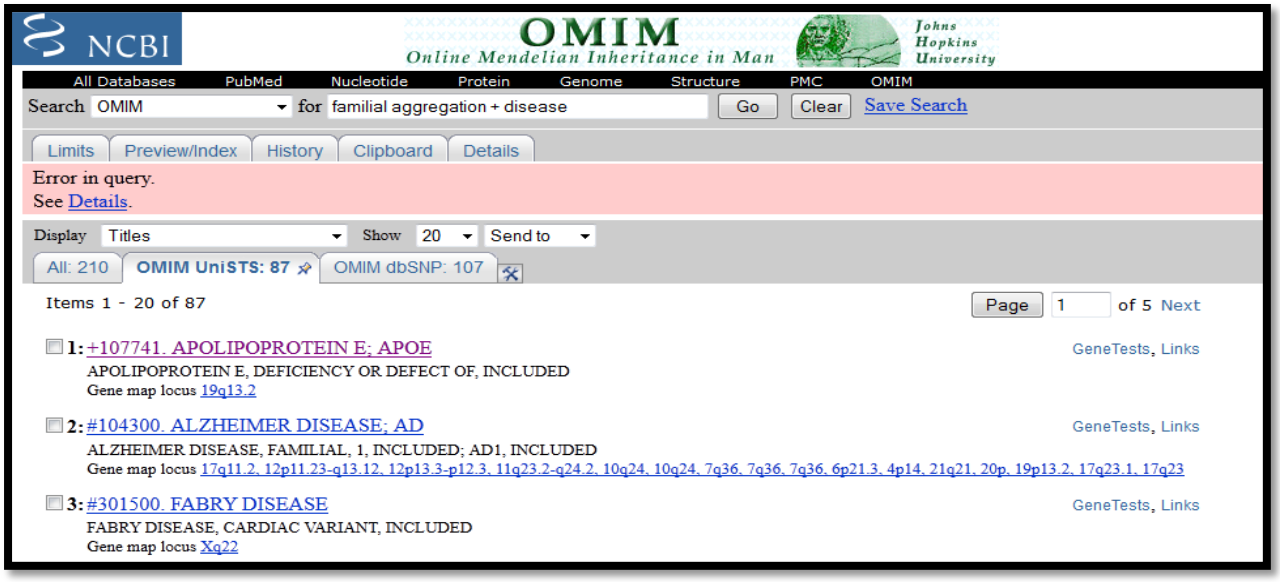



**Figure 4.3 OMIM homepage with search against the keyword "familial aggregation".**

Each gene page was opened and literature relating to familial aggregation was read so as to find if studies were carried out using pedigree. Similarly other databases such as PubMed central, Central Arab Genomic Disease and PubMed articles were also mined for compiling disease, gene mutations studied in a particular family lineage. The PubMed database had 2400 articles(Figure 4.4).

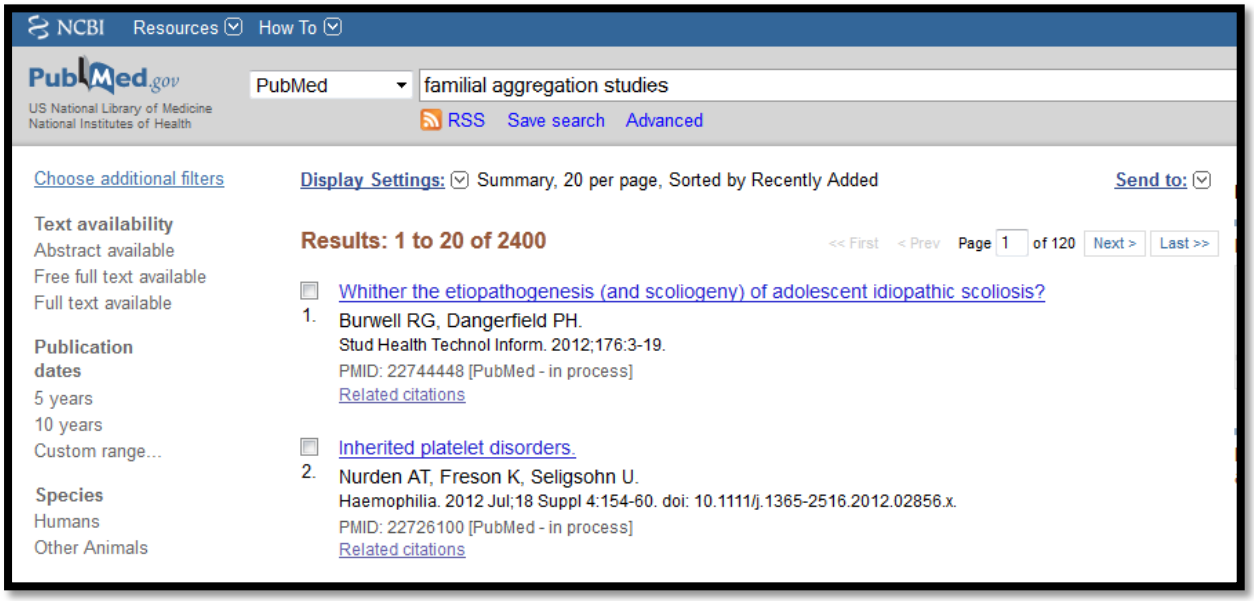



**Figure 4.4 PubMed homepage with search against the keyword "familial aggregation studies".**

PMC is a free full-text archive of biomedical and life sciences journal literature .The search hits were 154 full text article(figure 4.5).

**The search term used for PMC is given below.**

("family"[MeSH Terms] OR Familial[Acknowledgments] OR Familial[Figure/Table Caption] OR Familial[Section Title] OR Familial[Body - All Words] OR Familial[Title] OR Familial[Abstract]) AND aggregation [All Fields] AND studies [All Fields] AND pedigree [Figure/Table Caption])

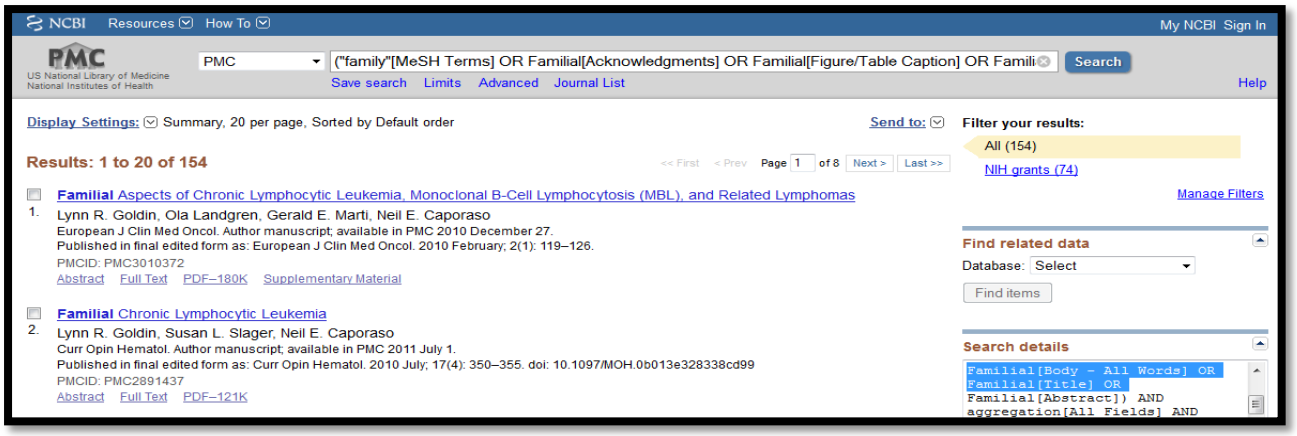



**Figure 4.5 PubMed Central search search results.**

The Centre for Arab Genomic Studies initiated a pilot project to construct the "Catalogue for Transmission Genetics in Arabs" (CTGA) database for genetic disorders in Arab populations. The current version 7.67 of CTGA (release date: 28.6.2012) contains nearly 960 full-text records (figure 4.6), including extensive data from the United Arab Emirates, Bahrain, Oman, Qatar, and Kuwait.

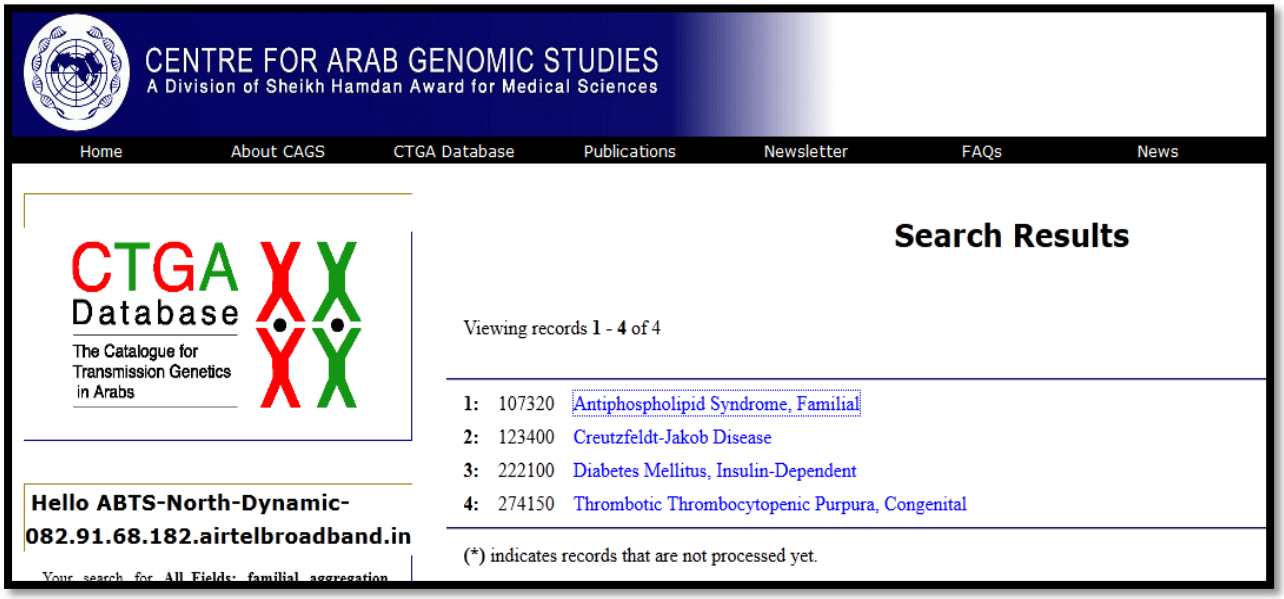



**Figure 4.6 The search results of CTGA Database.**

The gene specific information which includes Gene name and its Chromosomal location were retrieved from HUGO Gene Nomenclature Committee (HGNC) **[61].**The mutational data

such as SNP-ID, Variant Locations (Exonic), Corresponding Amino Acid changes, is recorded from dbSNP. For future reference of those particular articles the PubMed ID or the PubMed Central ID (PMC) of the literature is recorded. The main emphasis is on collecting as many diseases related pedigree as possible. The below listed are the fields of the database.

**DiseaseName**: The disease name is a broad category of disease.

**SubDiseaseName**: Represented as SDName in the maintable of the database contains the different categories of the same disease.

**DiseaseType**: The sample has been mentioned under each category autosomal dominant, autosomal recessive, X-linked dominant, X-linked recessive, Y- linked or mitochondrial mutation as reported in literature.

**OtherDisease**: Represented as ODisease in the maintable of the database shows the name of any other disease which could have been present in the proband's family.

**Gene**: Genes hold the information to build and maintain an organism's cells and pass genetic traits to offspring. Any change in the sequence of the gene may give rise to a defective protein or may silent the gene itself. In both the cases the individual is affected. This has been considered as an important feature for searching. The searching can be carried out using Gene HGNC Symbol. The HGNC symbol is unique and is given only to one gene. It is necessary to provide a unique symbol for each gene so that others can talk about them, and this also facilitates electronic data retrieval from publications and databases.

**GeneLocus**: The specific place on a chromosome where a gene is located is referred as locus.

**Associated Mutation**: The mutation can be insertion, deletion, duplication, single nucleotide change, Translocation etc.

**Single Nucleotide Polymorphism**: This field contains only those SNP which have been reported in dbSNP.

**Ethnicity**: The field contains the region to which the proband belongs. Most of the proband's of African, Caucasian, French, United States, and Chinese respectively.

**PedigreeNumber**: This field has been introduced so that the problem of a single pedigree involving more than one mutation can be sorted.

**Link**: The pedigree has been reported in the form of hyperlinks to the images in the research article.

**Pedigree Name**: If the pedigree link contains more than 1 pedigree, then this field will report the family name or pedigree reference name.

**p-value**: A measure of how much evidence there is against the null hypothesis. The smaller the p-value, the more evidence exists against. Traditionally, researchers will reject the null hypothesis if the p-value is less than 0.05. A small p-value is evidence against the null hypothesis while a large p-value means little or no evidence against the null hypothesis.

**LOD**: It is the LOD score, logarithm (base 10) of odds, is a statistical test often used for linkage analysis. It compares the likelihood of obtaining the test data if the two loci are indeed linked, to the likelihood of observing the same data purely by chance. Positive LOD scores favor the presence of linkage, whereas negative LOD scores indicate that linkage is less likely.

**Citation**: The PubMed ID or the PubMed Central ID of every literature associated with the pedigree has been reported.

### 3.1.2 Data Cleaning

The data was manually curated to remove redundancy, errors, and incomplete elements. It was then compiled into a single, composite, non-redundant database. After an exhaustive

review of all the literature using different keywords, each entry is confirmed from other sources too.

### 3.1.3 Creation of data model

The data model was created using the free open-source version of the MySQL Workbench. MySQL Workbench is a visual database design application that can be used to efficiently design, manage and document database schematics.

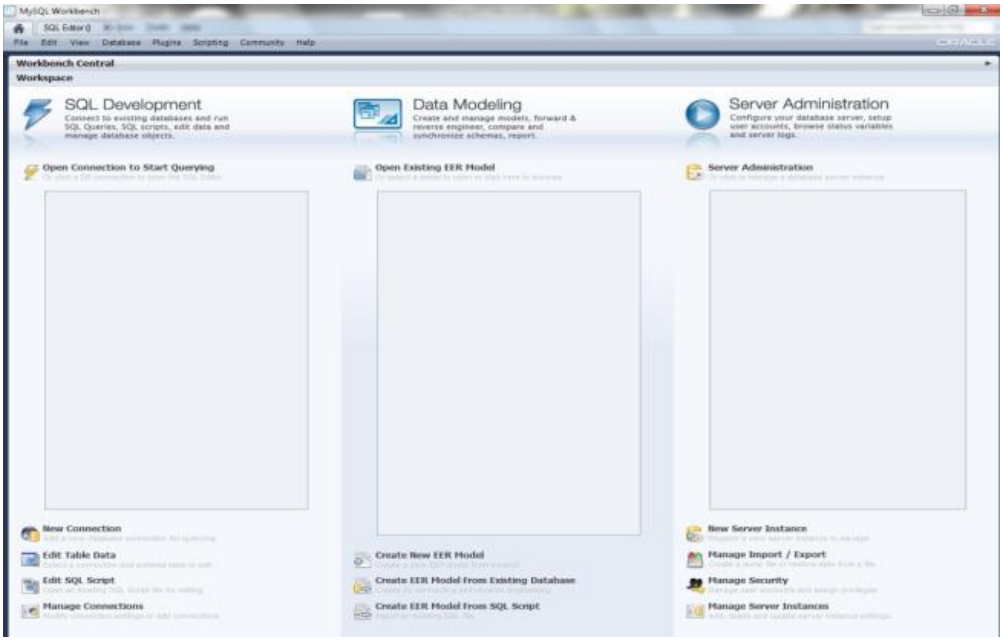


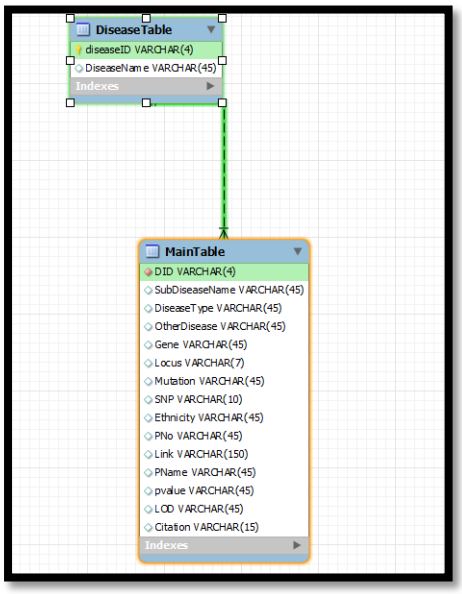

**Figure 4.1 MySQL workbench.**

A data model for the database called dbPedigree was designed using MySQL Workbench version 5.2 (http://www.mysql.com/products/workbench/) to store the information in the disease-pedigree in the form of a relational database. To create a new data model, the following steps were performed:

1. Under the heading “Data Modeling”, click on “Create new data model”.
2. Double-click on the “mydb” tab to enter the name of the database.
3. Double-click on “Add table” to add a new table to the database. It is possible to enter the table name, column names and types as well as declare primary key, foreign keys and other constraints and triggers.

The Physical Schemata of mydb showing 2 tables.


**Figure 4.2 The Physical Schemata of mydb.**



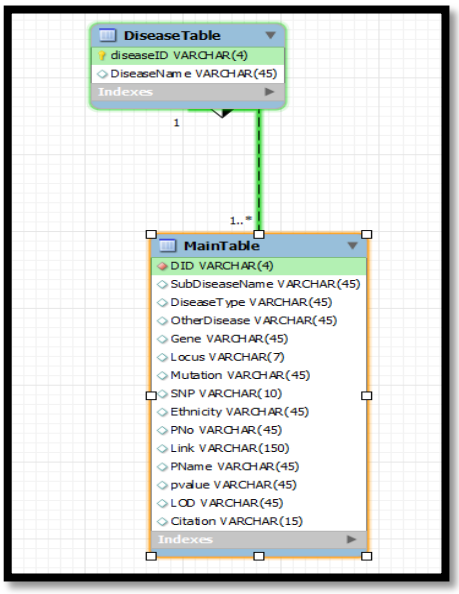




**Figure 4.3 The crow foot model of mydb.** **Figure 4.4 The classic view of mydb.**

The fields were divided into two tables. Each disease table is assigned one primary key. Foreign keys were used to reference the other tables in the database. To create a new table, the following steps were performed:

1. Double-click on “Add table” under the heading “Tables”.
2. Type the table name in the text box.

3. Double-click under the heading Column Name and type the name of the column that is the primary key. Ensure that the PK (Primary Key) and NN (Not Null) checkboxes are selected.
4. Select the data-type of the column.
5. Create the other fields in the same way.
6. To declare a column as a foreign key, click on the "Foreign Keys" tab at the bottom. Type the foreign key name and the referenced table and column. Also select Cascade option in the case of both Update and Delete. This will ensure that the changed value of the referenced column is reflected accurately in this table as well.

### 3.1.4 Development of a database and importing tables

The WAMP Server Version 2.2(Version Française) was downloaded and executed. The Server Configuration was found to be having Apache Version: 2.2.22, PHP Version: 5.3.13 and MySQL Version: 5.5.24.


**Figure 4.5 WAMP Server Version 2.2.**

It also contains a module called phpMyAdmin (version 3.5.1) that makes creation and maintenance of MySQL databases simple. This can be accessed from

http://localhost/phpmyadmin. Some of the functions of phpMyAdmin are browse and drop databases, tables, views, columns and indexes, create, copy, drop, rename and alter databases, tables, columns and indexes, maintenance server, databases and tables, with proposals on server configuration, load text files into tables, import data and MySQL structures from OpenDocument spreadsheets, as well as XML, CSV, and SQL files, manage MySQL users and privileges, using Query-by-example (QBE), create complex queries automatically connecting required tables, transform stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link etc.

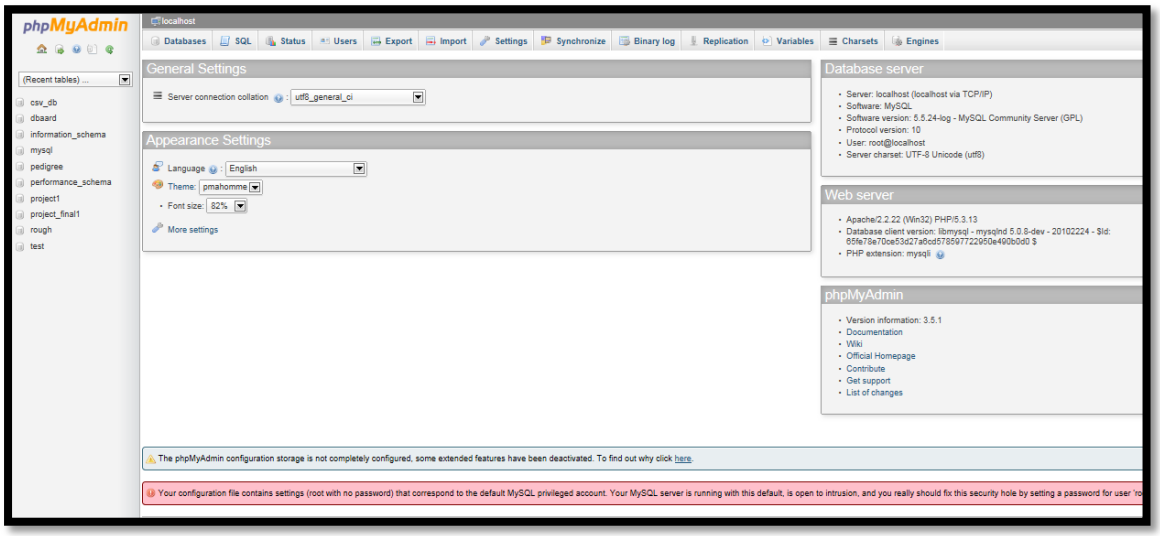



**Figure 4.6 The phpMyAdmin homepage.**

For creating a database the steps to be performed are

1. Click on Database tab.In the Create database text box, enter the name of the database, and then click on create.

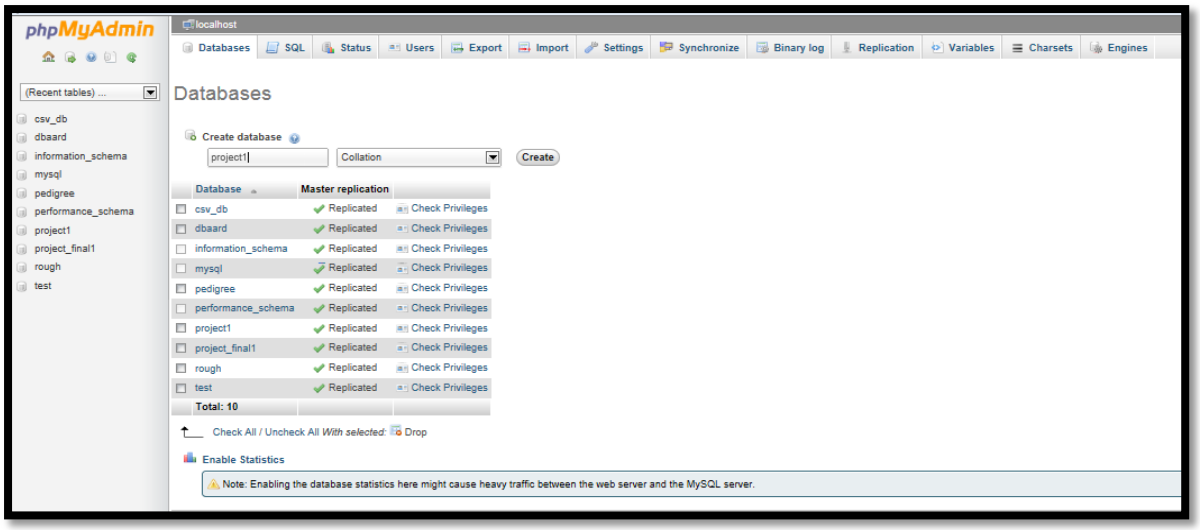



**Figure 4.7 Database Creation in phpMyAdmin.**

2. As soon as the database is created, its name will appear along with the existing databases.

After the creating of the database, tables need to be created. The raw data is divided into different Comma Separated Variable (CSV) files such that one CSV file corresponded to one table in the data model. The data is loaded into the tables using the Import tab in phpMyAdmin.

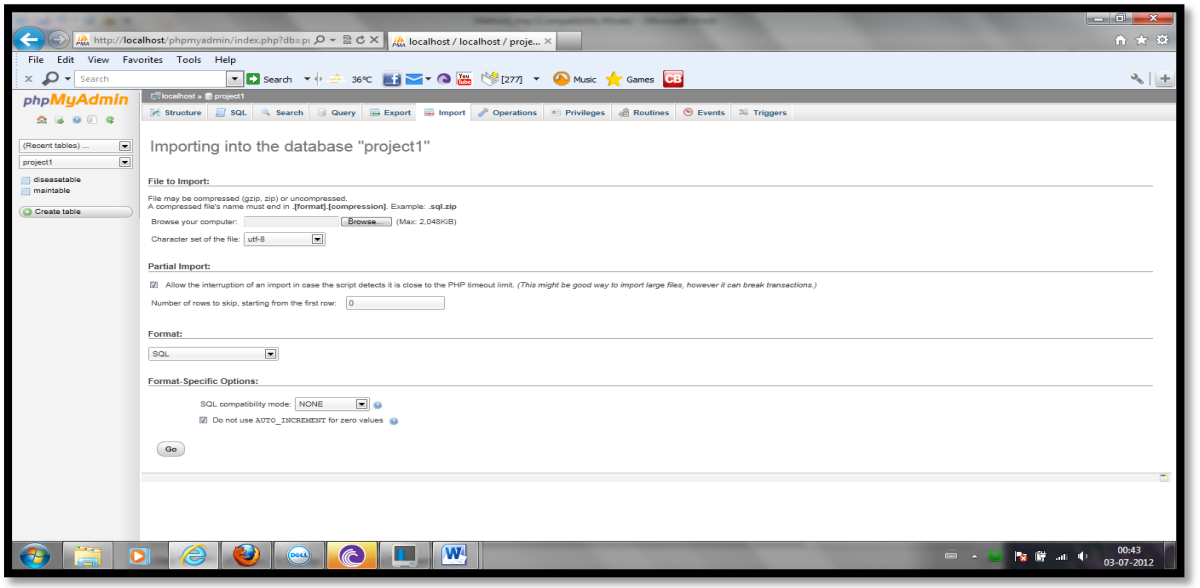



**Figure 4.8 Importing a CSV file into the database.**

With every genetic defect reported, the database also provides a family history data reported in literature. The database contains extensive information on each heritable disorder .The database contains 2 tables: DiseaseTable and MainTable. Both the tables are linked by a key field. The DID is foreign key of main table referring to DiseaseID which is primary key of disease table. The fields that are present in the database are showed in Table 4.1.

| DISEASE TABLE | MAIN TABLE |
|---|---|
| DiseaseID,<br>DiseaseName | DID (Foreign key) referring to DISEASE TABLE. DiseaseID, SubDiseaseName,DiseaseType,Other Disease,Gene,GeneLocus,Associated Mutation, SNP, Ethnicity, PedigreeNumber, Link, Pedigree Name, Pvalue, LOD Score, Citation |

**Table 4.1 Shows the fields present in the tables of dbPedigree.**

After importing the tables, structure of the table has to be defined. By default column 1 is named COL1, changes can be made by clicking change tab and then clicking on Save.

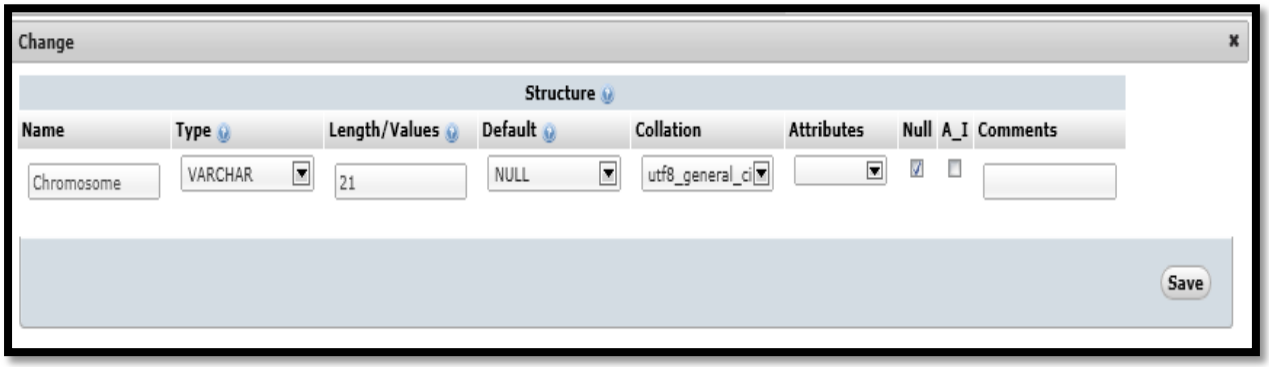


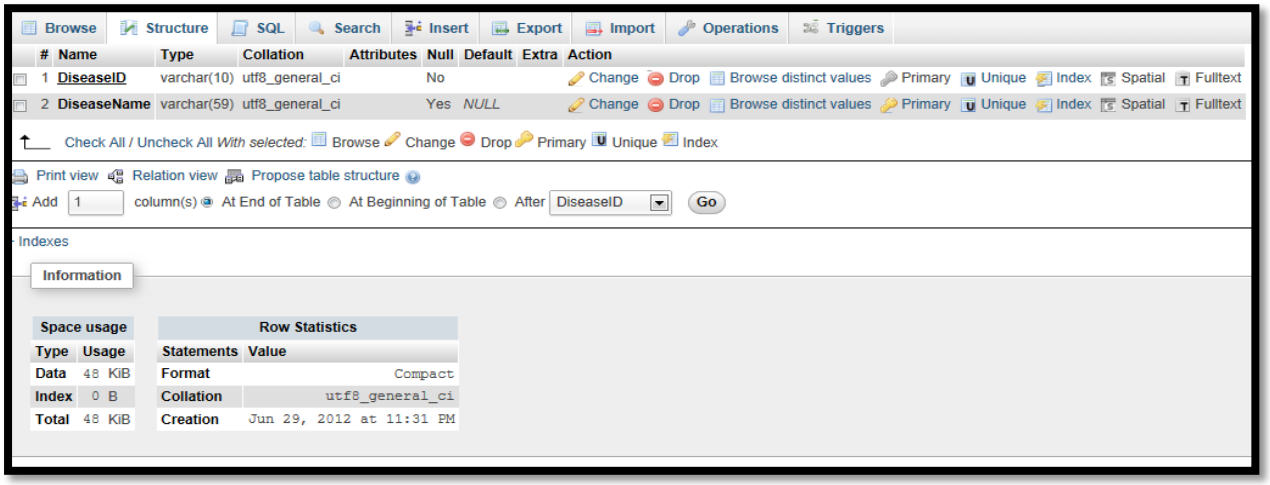

**Figure 4.9 Renaming a column.**

Similarly whole structure of each table can be defined. The structures of both the tables have been given in figures below .The table structure can be viewed by structure tab and browse tab allows to browse the data present in the table.

| # | Name | Type | Collation | Attributes | Null | Default | Extra | Action |
|---|---|---|---|---|---|---|---|---|
| 1 | DID | varchar(10) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 2 | SDName | varchar(59) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 3 | DType | varchar(28) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 4 | ODisease | varchar(66) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 5 | Gene | varchar(39) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 6 | Locus | varchar(21) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 7 | Mutation | varchar(94) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 8 | SNP | varchar(24) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 9 | Ethnicity | varchar(43) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 10 | PNo | varchar(15) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 11 | Link | varchar(488) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 12 | PName | varchar(28) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 13 | pvalue | varchar(11) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 14 | LOD | varchar(25) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 15 | Citation | varchar(22) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |

Check All / Uncheck All *With selected:* Browse Change Drop Primary Unique Index

**Figure 4.10 The SCHEMA of the main table.**



| # | Name | Type | Collation | Attributes | Null | Default | Extra | Action |
|---|---|---|---|---|---|---|---|---|
| 1 | DiseaseID | varchar(10) | utf8_general_ci | | No | | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |
| 2 | DiseaseName | varchar(59) | utf8_general_ci | | Yes | NULL | | Change Drop Browse distinct values Primary Unique Index Spatial Fulltext |

Check All / Uncheck All *With selected:* Browse Change Drop Primary Unique Index

Print view Relation view Propose table structure

Add 1 column(s) At End of Table At Beginning of Table After DiseaseID Go

Indexes

Information

| Space usage | |
|---|---|
| Type | Usage |
| Data | 48 KiB |
| Index | 0 B |
| Total | 48 KiB |

| Row Statistics | |
|---|---|
| Statements | Value |
| Format | Compact |
| Collation | utf8_general_ci |
| Creation | Jun 29, 2012 at 11:31 PM |

**Figure 4.17 The SCHEMA of the disease table.**

### 3.2 WEB DESIGNING:

The user friendly webpages of dbPedigree were designed using the Macromedia Dreamweaver (version8). Macromedia Dreamweaver 8 is a professional HTML editor for designing, coding, and developing websites, web pages, and web applications. The control coding HTML or visual editing environment both are present. The steps to install Dreamweaver are as follows.

1. Insert the Dreamweaver CD into your computer's CD-ROM drive.
2. In Windows, the Dreamweaver installation program starts automatically.
3. Follow the onscreen instructions. The installation program prompts you to enter the required information.
4. If prompted to do so, restart your computer.


**Figure 4.11 Macromedia Dreamweaver 8.**

Each webpage is designed in HTML. The code mode can be used for adding content to the webpage.

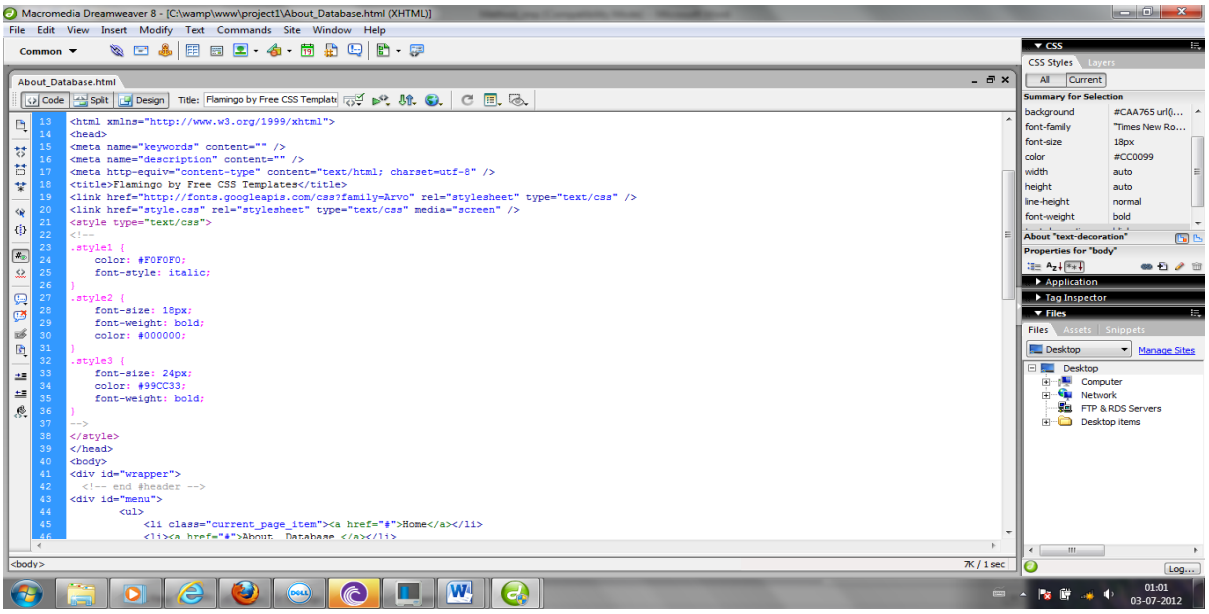


**Figure 4.12 Edit window of Macromedia Dreamweaver.**

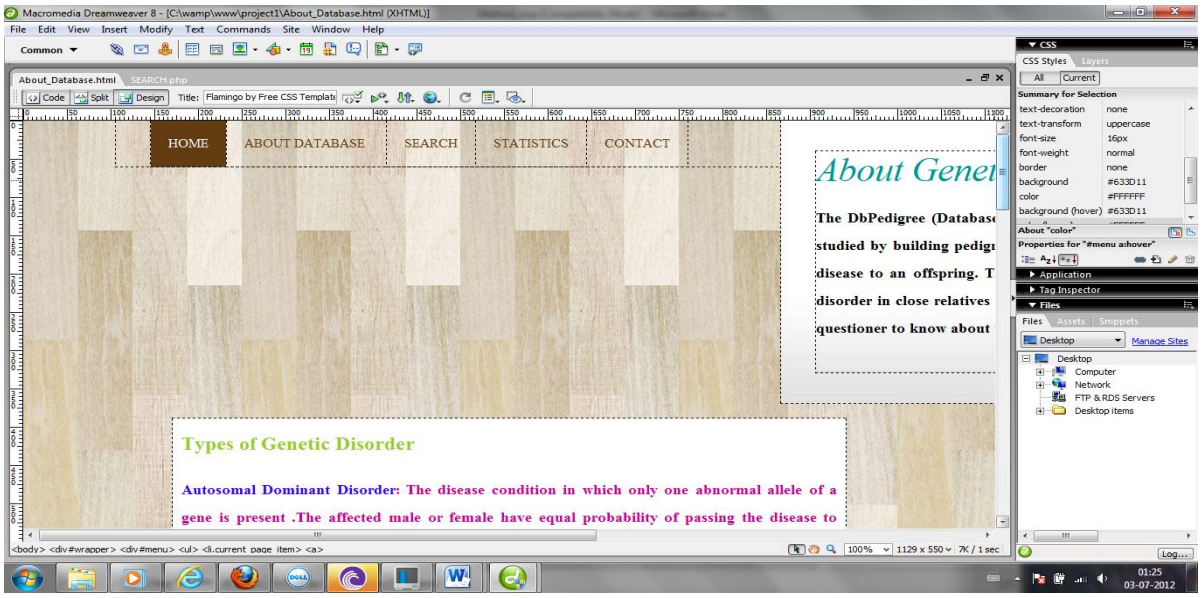



**Figure 4.13 Design view of page.**

After creation of each page in HTML, all pages were saved with extension PHP.As the data in HTML page remains static, this step is performed.

### 3.3 Connecting Webpage And Database

The connection between the webpage and the SQL database is done using PHP scripting language. PHP is a server-side scripting language designed for Web development to produce

dynamic Web pages. It is one of the first developed server-side scripting languages to be embedded into an HTML source document rather than calling an external file to process data. The code is interpreted by a Web server with a PHP processor module which generates the resulting Web page. PHP is already present in Dreamweaver software. The PHP variables were connected to database. SQL commands were written to extract data from database. The result is stored in other PHP variables .These variables were then displayed on the output window. Therefore the process is as follows:

1. The browser sends the query to the server. This may be done using the GET or POST methods.
2. The server executes the PHP script which connects to the database and gets the information required.
3. This information is passed to the server.
4. The server sends the information back to the browser where it is displayed to the user.

The Searching process for dbPedigree would be disease wise or gene wise. Disease search would be carried out by selection disease radio button and then specifying the disease of interest from the enlisted diseases in the search option. After applying all the filters the list of all the entries pertaining to that particular disease will appear on the screen. User can copy the link to pedigree and paste it in address bar .This links would direct him/her to the pedigree. The link is provided so that user can also refer to the literature. The other related information would be regarding associated mutation and ethnicity of the family has also been provided. Another option is to search by gene name. This can be done by the similar process .The database only includes HGNC gene symbol. After selecting the gene, the user is presented with a list of associated diseases from which the disease of interest can be selected for further information. The POST method is used to submit the form, so the form data is sent inside a message to the server, which makes it easier to extract and parse. The output is shown in the form of a table. For any further queries or updates the Email ID of the author has been provided. This would help in keeping the database updated with the most recently identified pedigrees.