CS 411 Project Template

|  |  |
| --- | --- |
| ***Team Name*** | Pioneer |
| ***Members Names (NetIDs)*** | Ramesh Gopisetty (rameshg2), Arun Taneja (arunt2), Sravani Prakki (sprakki2), Mahesh Narasimhan (maheshn2) |
| ***Email*** | rameshg2@illinois.edu, arunt2@illinois.edu, [sprakki2@illinois.edu](mailto:sprakki2@illinois.edu),  maheshn2@illinois.edu |
| ***Captain*** | Ramesh Gopisetty (rameshg2) |
| ***Project Title*** | Hospital Quality Statistics (HQS) |
| ***Project Summary*** | Hospital Quality Statistics provides statistical information and feedback about the Medicare services to the user. The center for Medicare Services has datasets that provide list of hospitals and the structural measures. These measures are developed by Centers for Disease Control and Prevention and collected through the National Healthcare Safety Network. Using these datasets our database application will provide information related to various facilities that are listed in Medicare which are available across multiple locations. It also provides statistics for the quality performance for the measures taken in hospitals. Furthermore, it displays list of Footnotes and summary for each measure taken. |
| ***Project Description*** | * **Description of an application of your choice.** HQS is a health care application which uses health care programs such as Medicare, Medicaid. It collects and analyzes data from various facilities, measures and produces reports related to the quality score. * **Usefulness.** Provide information related to the statistics and feedback of the Medicare and lab equipment to the users.   It helps the users with their search information related to Medicare. The content is indexed well so that the related search results can be found instantly. It is fast and helps in increase of efficiency.  There are similar websites/applications like CMS, Florida Health price finder.  This application provides information score across the entire country. The user can drilldown the medicate services to within state and city.   * **Dataset.**   The main dataset is Healthcare Associated Infections – Hospital. It has 50,000+ records. It is a precompiled dataset from websites like Kaggle, <https://data.medicare.gov/Hospital-Compare/Healthcare-Associated-Infections-Hospital/77hc-ibv8/data>. We also gather information from other datasets such as Footnote- Crosswalk, Hospital General Information, Structural- Measures – Hospital.   * The website provides the information of the hospital measures and quality performance stats across the states. This site helps to compare the data between various hospital measures and choose to help the best one. This also provides to visualize the information graphically and identify the best hospital measures. And also help to analyze the data for improving the measures.   ***Basic Functions***   * We use the basic functions to generate the reports and perform the dml operations for the contributor. The DML operations helps the contributor to perform the Create, Update and Delete the information across various tables to footnotes, measures, facility and health group data.   The application consists of three different kind of users.   1. Administrators   The highest privilege which has access to the user authentication, feedback review, report analysis and contribution.   1. Contributors   Contributors has the privilege to insert/update/delete the records of hospital, measure and footnotes. We will be providing the login information to the National HealthCare Safety Network.   1. Readers/EndUsers   They just have the read only privilege among the data. They review the reports of the hospital measures taken from the start and the end date.   * We will be using several advanced techniques such as partitioning the data to store the measures of various dates with hash-list partition. Several constraints used such as foreign key, unique key, not null constraints to validate the data. To provide the unique key across the facility/score/measure, sequences and triggers used to populate the data. To review the reports created several vies to represent the data.   Advanced Techniques   * Indexing   We have created several indexes across the tables to improve the performance of the reading the information in the measures and facility table.   * Triggers   Created triggers to provide the basic primary key for all the tables using the sequences. The primary key is auto-populated from the sequences that were created. The triggers helps inserting the primary key value during before insert of data.   * Partitioning/Sharding   The Measures table created using the interval partitioning. This helps to run the queries or fetch the information based on intervals.   * View   Views created to access the information required all in single select.   * Stored Procedure   Stored Procedure used to perform validations and run across various tables measures and quality tables.   * Constraints   Several Foreign key, Not Null, Check , Unique Constraints created to maintain the data consistency across the tables.  Advanced Features of the Application   * Uses Google API (JavaScript MAP API) to autocomplete the address and also the latitude and longitude. * Implementing the Social Login either Google/Facebook/Twitter/ Linkedin..etc. for the end user to access the application to save the information and send the emails as per requirement. * Implementing the data visualization of the data to review the statistics. |
| ***ER Design*** | The ER diagram has the following assumptions:   * A Facility performs many measures * A measure has many scores related to performance and quality. * Each measure has the summary or foot notes * There are many facilities for Hospital ownership   Physical Model |
| ***Development Plan*** | The relational schema of your database is as follows:   * FACILITY (FID, FNAME, FADDRESS, FCITY, FSTATE, FZIPCODE, FCOUNTY, FPHONE, FLOCATION, FHTYPE) * FOOTNOTES\_HQS (FNID, MID, F\_TEXT, FDATE) * MEASURES\_TAKEN\_HQS (MID, FID, MSTARTDATE, MENDDATE, MSTART\_QUARTER, MEND\_QUARTER, MNAME, MACRONYM) * QUALITY\_PERFORMANCE\_HQS (QSID, MID, QNATIONAL\_SCORE, QCOMPARE\_TO\_NATIONAL, QSTATE\_SCORE) * HOSPITAL\_DETAILS\_HQS (HID, HNAME, HDESC, HRATING)   The functional dependencies of the database are as follows:   * HID -> FID * FID -> FID, FNAME, FADDRESS, FCITY, FSTATE, FZIPCODE, FCOUNTY, FPHONE, FLOCATION, FHTYPE * FID -> MID, FNID, QSID * FNID -> FNID, MID, F\_TEXT, FDATE * MID -> MID, FID, MSTARTDATE, MENDDATE, MSTART\_QUARTER, MEND\_QUARTER, MNAME, MACRONYM * QSID-> QSID, MID, QNATIONAL\_SCORE, QCOMPARE\_TO\_NATIONAL, QSTATE\_SCORE   **Tools Used**  Database Platform: Oracle  Software: Oracle APEX  The final choice of databases and software platforms/languages that will be used are Oracle Application Express (Oracle APEX) as code development platform, Oracle 19c Database. We will be hosting the application in Oracle Cloud Platform.  **Data**     * We will get the data from precompiled datasets like Kaggle, Medicare.gov   **Project Timeline**  A project timeline with milestones:  Week 7: Create Tables, Create Data and Start planning of UI. Populating the tables with relevant data  Week 8: Start designing static pages of UI, Create SQL queries for basic functionalities.  Week 9: Planning and implementation of constraints, views and triggers. Performing the unit testing. Integrating with Social Login.  Week 10: Planning and implementation of indexing and compound statements, Partitioning\Sharding.  Week 11: Planning and implementation of the stored procedures. And designing the application and Functional testing.  Week 12: Implementation of email notification if user preference has the significant score changes. Documentation update.  Week 13: Implementation of the recommendation engine based upon user search parameters. Preparing the demo video. |
|  | **Division of Work**  ***Ramesh Gopisetty***: Schema designing, Store procedures, UI Planning, SQL Queries, Indexing, Partitioning\Sharding, UI designing and development, Planning and development of the advanced techniques.  ***Sravani Prakki*:** SQL Queries, Store procedures, Compound Statements, Constraints, UI design and development of advanced techniques, documentation, Functional testing of the application.  ***Arun Taneja*:** Triggers, Views, UI developing, Feature planning, UI Development and hosting, development of advanced techniques. Performing Unit testing for the code.  ***Mahesh Narasimhan*:** SQL queries, Indexing, Feature planning, UI Development and hosting. Preparing the video for submission. |
| ***Project URL*** | <https://apex.oracle.com/pls/apex/f?p=89561>  Script Location  <https://github.com/ramesh-edu/HQS>  We are using Oracle APEX. We are working to get this hosted in Oracle Cloud/AWS Cloud. Currently, we are developing the application in the development environment of Oracle APEX Cloud Service. |