

A Major Project Final report on

Database Conversion System With Agents

Submitted in Partial Fulfillment of the Requirements for
the Degree of **Bachelor of Engineering in Information
Technology**

Under Pokhara University

Submitted by:

Bishal Trital, 13410
Nishma Adhikari, 13423
Sanjaya Trital, 13433
Santosh Dahal, 13435
Subas Pokhrel, 13438

Under the Supervision of:

Dr. Roshan Chitrakar

Date: **11 – 18 – 2017**



Department of Science and Technology

**NEPAL COLLEGE OF
INFORMATION TECHNOLOGY**

Balkumari, Lalitpur, Nepal

ACKNOWLEDGEMENT

It is an at-most privilege to express our sincere regards to the project supervisor, **Dr. Roshan Chitrakar** for his valuable guidance and encouragement throughout the duration of the project. We deeply express our sincere thanks to him for encouraging and allowing us to present the project on the topic “**Database Conversion System with Agents**” for the partial fulfillment of the requirements leading to the award of the degree of bachelors of engineering in information technology.

We take this opportunity to thank all the lecturers who have directly or indirectly helped our project accomplishment. We are highly obliged to them for providing us this opportunity to carry out their ideas and work during the project period and helping us to gain the successful completion of the project. We would like to thank **Er. Saroj Shakya and Er. Manjul Bhattarai** for the support we have been provided during this project development period. It would be our privilege to thank **Er. Madan Kadariya**, Head of IT department for his valuable support to the project. We pay the respect and love to our parents and all other family members for their love and encouragement throughout our project development period. Last but not the least, we express our thanks to all our beloved friends for their cooperation and support.

ABSTRACT

Technology is evolving rapidly. Evolution prompts accomplishment and in addition, disappointment. The way we store information is additionally changing with time. Database conversion system with agent is a module that saves the data in the active database (MYSQL) and old existing database (MS Access). The existing system is not preferable nowadays and the several researches have been made in conversion, integration, merging and migration. There are some migration tools available. However, only migrating does not guarantee the data integrity. Therefore, aiming to fill the gap between the evolution and existing old system, it first saves the data into most commonly used database management system i.e. MySQL, extracts the data to CSV file in chunks and finally copy the data into MS-access which is the backup database. The plight of not replacing the old framework entirely but use both the system is to ensure the protection of data, in case of failure of active database. The agent used in this module can also handle interrupt that may occur during the movement of this information from one database system to another system, independent of the operating system used.

Keywords

database, conversion, MySQL, MS-access,

:

LIST OF FIGURES

Figure 1: Incremental Model	17
Figure 2: Integrated View of Increments	18
Figure 3: Pictorial representation of Algorithm.....	25
Figure 4: Time Scheduling.....	26
Figure 6: Context diagram	29
Figure 5: System Architecture	30
Figure 7: Use Case Diagram	31
Figure 8: System Sequence Diagram from MySQL to CSV	32
Figure 9: System Sequence Diagram for CSV to MS Access	33
Figure 10: Class Diagram for formulation of CSV	34
Figure 11: Class Diagram to load data from CSV to My Access	35
Figure 12: tbl_doctor before first scheduler implementation	39
Figure 13: First CSV formation	39
Figure 14: Initial Startpoint and Endpoint	40
Figure 15: Startpoint and endpoint increases.....	40
Figure 16: Second CSV file formation	41
Figure 17: Increment after second scheduler implementation.....	41
Figure 18: Initial tbl_doctor in MYSQL.....	42
Figure 19: Updated tbl_doctor in MYSQL.....	42
Figure 20: The updated data stored in table "tbl_update".....	43
Figure 21: Initial tbl_doctor in MS Access.....	44
Figure 22: Updated tbl_doctor in MS Access.....	44
Figure 23: CSV deleted after converted to MS Access	45
Figure 24: CSV deleted after update of existing data in MS Access.....	45

LIST OF TABLES

Table 1: Team members and divided roles	11
Table 2: Patient details.....	13
Table 3: Doctor details.....	14
Table 4: Employee details.....	15
Table 5: Update details	16
Table 6: Testing table.....	38

TABLE OF CONTENTS

ACKNOWLEDGEMENT	i
ABSTRACT	ii
LIST OF FIGURES	iii
LIST OF TABLES	iv
LIST OF ABBREVIATION	vii
1. INTRODUCTION	1
1.1 PROJECT OVERVIEW	1
1.2 PROJECT SCOPE AND LIMITATIONS	2
1.2.1 SCOPE	2
1.2.2 LIMITATIONS	2
1.3 PROBLEM STATEMENT	3
1.4 SIGNIFICANCE OF STUDY	3
1.5 PROJECT OBJECTIVES	4
2. LITERATURE REVIEW	5
2.1 REVIEW	5
2.2 RELATED TASK	5
2.2.1 DOMAIN	5
2.2.2 EXISTING WORK	5
2.3 COMPARISON WITH EXISTING SYSTEM	6
2.4 DATABASE	7
2.5 DATABASE TRIGGERS	7
2.6 CSV AND JSON	8
2.7 JDBC AND ODBC	8
2.8 TASK SCHEDULER AND CRON TASK	9
2.9 JAVA AND PHP	9
2.10 JAVA THREADS	9
3. TEAM MEMBERS AND DIVIDED ROLES	11
4. REQUIREMENT	12
4.1 REQUIREMENT ANALYSIS	12
5. METHODOLOGY	17
5.1 SOFTWARE DEVELOPMENT LIFE CYCLE	17

5.2 ALGORITHMS	22
5.3 PICTORIAL REPRESENTATION OF ALGORITHM.....	25
6. TIME SCHEDULING	26
7. TOOLS AND TECHNIQUES	27
8. SYSTEM DESIGN	29
8.1 CONTEXT DIAGRAM.....	29
8.2 SYSTEM ARCHITECTURE	30
8.3 USE CASE DIAGRAM.....	31
8.4 SYSTEM SEQUENCE DIAGRAM.....	32
8.4.1 SEQUENCE DIAGRAM FOR MYSQL TO CSV	32
8.4.2 SEQUENCE DIAGRAM FOR CSV TO MS-ACCESS	33
8.5 CLASS DIAGRAM	34
8.5.1 CLASS DIAGRAM FOR CSV FORMATION.....	34
8.5.2 CLASS DIAGRAM TO LOAD DATA FROM CSV TO MS-ACCESS.....	35
9. BUDGET	36
9.1 FUNCTION POINT.....	36
9.2 LINES OF CODE	37
10. TESTING.....	38
10.1 TESTING TABLE	38
10.2 LIST OF TESTING	39
11. RESULT AND DISCUSSION	46
12. FUTURE EXTENSION.....	47
13. REFERENCE.....	48

LIST OF ABBREVIATION

API	Application Programming Interface
BAT	Bacula Administration Tool
CSV	Comma Separated Variables
DCS	Database conversion System
DDL	Database Definition Language
DML	Database Manipulation Language
FP	Function Point
HTTP	Hypertext Transfer Protocol
JAR	Java Archive
JDBC	Java Database Connectivity
KLOC	Kilo Lines of Code
LAMP	Linux, Apache, MySQL, Perl/PHP/Python.
MS	Microsoft
MySQL	My Structured Query Language
ODBC	Open Database Connectivity
PHP	PHP: Hypertext Preprocessor
RDBMS	Relational Database Management System
SQL	Structured Query Language
UML	Unified Modeling Language
VBS	Visual Basic Script
WORA	Write Once Run Anywhere

1. INTRODUCTION

“Database Conversion System with Agents” refers to the collection of processes and procedures for converting the data from one database to another, making the use of agent. Database conversion system with agent includes the following:

- The upgrade process, which upgrades an old existing database to new sophisticated database, and makes use of old existing database as a backup database.
- The Interrupt handling process, which take cares of the hinder during the data conversion process.

As our project is a backend project, we have implemented it in inpatient registration system in hospitals. To accomplish a successful database conversion, we use conversion method that converts data stored in MySQL to MS-Access. In this system, MYSQL is the active database used for day-to-day transaction whereas MS Access is the backup database. MySQL is an open-source relational database management system (RDBMS) [1] whereas MS-Access (Microsoft Access) is a Database Management System (DBMS) from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software-development tools [2]. We use two Databases for making the data available in case of failure of one database. In addition, the user can switch to old existing system whenever he/she wants.

1.1 PROJECT OVERVIEW

The Database Conversion phase includes the construction of new relational database structures designed to provide the same traversal and performance capabilities as the former database. All requirements for table and column naming, date conversion, element grouping, occurs clause handling and data cleansing are handled in this phase. Microsoft Access is a database management system (DBMS) from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software-development tools. Microsoft Access stores data in its own format based on the Access Jet Database Engine. MySQL is an open-source relational database management system (RDBMS). The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks).

Trigger will retrieve the updated data from all tables into one table and transfer the data from that table into csv file. Task scheduler is used to schedule jobs and run at fixed intervals. This creates the CSV file which is then parsed to string and with appropriate conversion, is transferred to MS Access.

1.2 PROJECT SCOPE AND LIMITATIONS

The scope and limitations of our project are:

1.2.1 SCOPE

The scope of this project is to convert the data stored in MySQL Database to MS Access Database through the use of agent. Our Database Conversion System with agent provides the following scope:

- Update the old existing Database to a new sophisticated Database.
- Perfect solution for the organizations which are still using the decade old system.
- Organizations can use the system efficiently without violating the old data.
- Provides the backup database to ensure data protection in case of active database crash.

This project can be useful in following areas:

- Hospital billing system where it requires one database as backup.
- Companies that uses two databases and one database is used as restoring database.

1.2.2 LIMITATIONS

Although our project provides reliability by providing alternative to old existing database system, we have some limitations, which is to be taken into consideration. Some of the limitations of our projects are:

- Our project does not provide the front-end design.
- It does not describe how data will be stored to MySQL Database.
- Use of multiple database may result minimal data integrity.

1.3 PROBLEM STATEMENT

Most of the database in the well-established organizations, hospitals, government agencies are more than a decade old. The backup system has not been able to keep up with the changing technology and offers a very little feasibility in terms of data storage. While technology evolves at a tremendous pace, those system appears to be time inefficient in the current scenario. Hence, most of the organization want to migrate from the old backup to the databases with time friendly features, reliable and which can support better performances.

We have implemented this system as a prototype in the Inpatient Registration System. Due to the use of decade old storage system, the work performance of the hospitals has been time consuming. Hence, the conversion agent aims to solve this barrier. The use of old existing database as a backup without replacing it completely ensures the data protection when the active database fails. Therefore, we have made the utilization of both the database for this system.

1.4 SIGNIFICANCE OF STUDY

The study suggests the need of an agent which deals with the conversion of data from one database to another, maintaining the data integrity. This system ensures the storage of data in two databases where one acts as the active storage and another acts as backup. We have implemented this system as a prototype in the inpatient registration system. While the old existing system could be replaced completely, we have taken into consideration the threat of database failure which might occur during switching. The use of Windows Task Scheduler that runs an agent every 1st hour, facilitates the extraction of data from the active database to form the CSV file in chunks and maintains data integrity which is not possible with the use of existing APIs.

The use of this conversion in inpatient billing system is helpful for operation of decade old hospitals where the patient's record from the past can be utilized over time.

1.5 PROJECT OBJECTIVES

The objectives of the project are as follows:

- Convert the data from MySQL database to MS-Access database with the use of agent.
- Interrupt handling while data is transferred from MySQL database to MS-Access database.
- Checks the updated tables in MySQL database using trigger and migrate that updated data to MS-Access database.
- Handles the constraints validity while migrating data from MySQL database to MS-Access database.
- Provides platform independent module. It works on Windows as well as Unix environment.

2. LITERATURE REVIEW

2.1 REVIEW

There are various works going on for database conversion. Many migration tools had been developed in the past by aiming to transfer the data. However, migrating only is not enough. This should ensure proper data integrity. So, for eliminating such problems we have tried to manage both transferring the data and also protecting the data integrity.

2.2 RELATED TASK

2.2.1 DOMAIN

This database conversion system can be used in the organizations working with much old databases which provide very little feasibility in the present context. This can be used in government offices, hospitals and several institutions which have huge data storage from the past.

2.2.2 EXISTING WORK

IMPORTING MYSQL DATA TO ACCESS USING ACCESS 2016

Some mechanism for importing data to MS access are available on Access application. But there was no provision for interrupt handling. Table name is required for every import. This creates complexity on data transmission and unsuitable for the creation of database blocks. This blocks are very important to create for error correction, interrupt handling and retransmission on failure.

DATABASE CONVERSION TOOLS

Several database tools were selected and tested as mentioned on proposal “Towards Flexible Database Conversion with Automatic Restructuring” by SITI Z.Z. ABIDIN, SUZANA AHMAD and WAEL M.S. YAFOOZ to analyze migration and conversion. From the detail observation on this proposal, we lead to our algorithm. Full convert enterprise tool, developed by Spectral Core is to do full conversion of sophisticated database. The old version of database is transformed into a new version. The software has greatly reduced the time and effort required to convert the data. Another tool by the same company, Data Sync, has different approach by finding the differences in database tables and later, helps to synchronize these tables. Other company, SwisQL, developed

a complete data migration tool that helps the migration and transfer of database schemas and data across leading databases such as Oracle, IBM DB2, MS-SQL Server, Sybase, SAP DB, MySQL, PostgreSQL and MS Access. During the migration process, the user can change the table structure and the content of table. However, the migration is limited to a single database as a source. It can be summarized that in database conversion between different database types, the conversion process involves the whole database schema from one database type to another. For database migration, most of the processes are about migrating the current database to a new version. Sometimes, the migration process involves data conversion between data types and database constraints. In all cases, they require the same structure of database to do the migration. In database integration, combining different databases to one-unit database is limited to the same database schema. In other words, the copy and transfer of data is onto one physical unit automatically by the system [3].

2.3 COMPARISON WITH EXISTING SYSTEM

Through our literature review, we studied the existing system and realized that the data integrity is not possible to achieve through the migration tools. The easily available tools could not meet the requirement of our project. These tools are highly optimized for database conversion and migration. But they all work with dialog wizard with limited functionality. These don't provide any interrupt handling mechanism and need to retransmit the whole database again on failure. Also, these tools cannot be optimized with the server to day to day backup process rather they are focused on one-time migration. Our main concern is to develop an agent for the evaluation of database free time to perform database migration and conversion. The objective of our project is to develop a module that can be implemented underneath a system that continuously checks the database and transmit those database entities to another database on regular basis.

2.4 DATABASE

Databases are used to store, retrieve and manipulate data in many organizations. Database technology is used by individual (on personal computer) or group (on network for centralized or distributed system). The word “data” can be defined as facts that could be recorded and stored on a computer. It can be processed to become information which is more convenient and suitable for communication, interpretation or processing by the end users. Moreover, data can also represent facts concerning people, places, events, objects or concepts. When data is grouped together and organized in a formalize manner, it can be called a database. A database management involves both defining structures for storing information and providing mechanisms for manipulating the information. There are many different types of DBMS, ranging from small systems that run on personal computers to large systems that run on mainframes. DBMS can also be divided into three types which include relational, network and hierarchical. The most widely common type of DBMS today is the Relational Database Management Systems (RDBMS) [4]. RDBMS is the basis for SQL, and for all modern database systems like MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access. A Relational database management system (RDBMS) is a database management system (DBMS) that is based on the relational model as introduced by E. F. Codd [5].

2.5 DATABASE TRIGGERS

Triggers are stored programs, which are automatically executed or fired when some events occur. Triggers are, in fact, written to be executed in response to any of the following events –

- A database manipulation (DML) statement (DELETE, INSERT, or UPDATE)
- A database definition (DDL) statement (CREATE, ALTER, or DROP).
- A database operation (SERVERERROR, LOGON, LOGOFF, STARTUP, or SHUTDOWN).

Triggers can be defined on the table, view, schema, or database with which the event is associated. Triggers can be written for generating some derived column values automatically, enforcing referential integrity, event logging and storing information on table access, auditing, synchronous replication of tables, imposing security authorizations, and preventing invalid transactions [6].

2.6 CSV AND JSON

CSV is a simple file format used to store tabular data, such as a spreadsheet or database. Files in the CSV format can be imported to and exported from programs that store data in tables, such as Microsoft Excel or OpenOffice Calc [7].

JavaScript Object Notation (JSON) is a lightweight data-interchange format. It is easy for machine to parse and generate [8].

CSV is preferred for our project over JSON because of following reasons:

1. CSV, it is a lot easier to read the file row by row.
2. Less bandwidth needed because JSON needs more syntax characters to keep the format while csv is as simple as use a character separator
3. Process data is faster because only needs to split by the separator character while JSON needs to interpret the syntax.
4. Big data technology as Hadoop have an integrated parse for CSV format while needs a specific function for parse JSON.

2.7 JDBC AND ODBC

Java Database Connectivity (JDBC) API is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases [9].

Open Database Connectivity (ODBC) is an open standard application programming interface (API) that allows application programmers to access any database. The main proponent and supplier of ODBC programming support is Microsoft, but ODBC is based on and closely aligned with The Open Group Standard Structured Query Language (SQL) Call-Level Interface (CLI) [10].

JDBC is preferred over ODBC due to the following reasons:

1. ODBC is procedural and JDBC is object oriented.
2. ODBC is used to provide connection between front-end application (other than java) and back-end (database like MS-access) and JDBC is used to provide connection between JAVA and database (oracle, Sybase, DB2, MS-access).
3. ODBC is for Microsoft and JDBC is for java applications.
4. ODBC cannot be directly used with Java because it uses a C interface.
5. ODBC makes use of pointers, which have been removed totally from java.

2.8 TASK SCHEDULER AND CRON TASK

The engine is developed such a way that the Cron task or Windows task scheduler triggers database on time basis. But the code is optimized such a way that it makes a single connection and get the value from with that connection. Thanks to java thread for this efficiency. This made less memory consumption and higher output. Once the scheduler calls a script and convert the database entities into csv file, these csv files are available on server memory. The manipulation of these files has no overhead on database. So, a script that saves the csv file to MS access has no impact on database regular operation.

Task Scheduler is a component of Microsoft Windows that provides the ability to schedule the launch of programs or scripts at pre-defined times or after specified time intervals. This is equivalent to Cron job on Unix based system.

2.9 JAVA AND PHP

The transmission of data from MySQL to MS access using php requires ODBC Data Sources (32-bit). During the development of system on windows 64-bit architecture, we found ODBC Data Sources (32-bit) was missing from the system. This required additional need of 32-bit driver download. In addition, the system was less flexible for the creation of chunks and interrupt handling. In addition, the speed and efficiency was comparatively less visible using php.

2.10 JAVA THREADS

Multithreading refers to two or more tasks executing concurrently within a single program. A thread is an independent path of execution within a program. Many threads can run concurrently within a program. Every thread in Java is created and controlled by the java.lang. Thread class. A Java program can have many threads, and these threads can run concurrently, either asynchronously or synchronously.

Multithreading has several advantages over Multiprocessing such as;

- Threads are lightweight compared to processes
- Threads share the same address space and therefore can share both data and code
- Context switching between threads is usually less expensive than between processes

- Cost of thread intercommunication is relatively low that that of process intercommunication
- Threads allow different tasks to be performed concurrently.

There are two ways to create thread in java;

- Implement the Runnable interface (java. lang. Runnable)
- By Extending the Thread class (java. lang. Thread) [11]

3. TEAM MEMBERS AND DIVIDED ROLES

The team members and the respective roles are as shown in table below:

Table 1: Team members and divided roles

Name	Roles	Responsibilities
Bishal Trital	Architecture designer, Developer	Trigger handling, Java thread Implementation, CSV formation
Nishma Adhikari	Project Manager, Requirement analyzer, Integration and documentation	Literature review, Requirement identification, Documentation and Final integration
Sanjaya Trital	Database Administrator	Query management, Data integration and mapping, CSV to MS-access transmission
Santosh Dahal	Developer, Testing & Debugging	CSV to java migration, CSV to MS-access transmission, Scheduler and Cron handling
Subas Pokhrel	Developer, Testing & Debugging	CSV formation, Database connectivity, Thread Implementation

4.REQUIREMENT

4.1 REQUIREMENT ANALYSIS

Requirement analysis is the requirement engineering task that bridges the gap between system level software engineering software design [12]. This encompasses the task that go into determining the need and conditions for the new or altered product taking account for the requirement of the various users. This also encompasses the task for analyzing, documenting, validating and managing system requirements.

SYSTEM REQUIREMENTS SPECIFICATIONS

1. Functional Requirement

1.1 Data Entry method

- Since this project does not involve the design of user interface, the data entered through various windows should be similar to the data types stored in the database. The table in MS Access should be synchronized to the table in MySQL.

1.2 Backup Facility

- This conversion system requires two databases i.e. MySQL and MS Access. MySQL is the active database and MS Access is used as the backup database. The tables in both databases should be synchronized i.e. the table name, field name and respective data types.

2. Input Requirement

This conversion system consists of the backend project which can be implemented for inpatient registration system in the hospital. Though the data entry is done using the user interface, the inputs requirement (in case of inpatient registration for the hospital) are as follows:

Patient details

- This consists of the details of each new patient.

Doctor details

- This consists of the details of the doctors working in the hospital.

Employee details

- This consists of the details of the staffs in the hospital.

Updated details

- This consists the information about the table which is edited and updated along with the modification time and date.

Input list and Validation

The user inputs, for each are categorized on the basis of entry given by the user. They are tabulated below as along with the data description, data length, and validation which would be helpful in the designing phase.

Table 2: Patient details

S.No	Input	Datatype	Length	Description
1	Id	Int	11	This is the id assigned to the patient..
2.	Registration	Varchar	50	It is the unique code give to each patient.
3.	Name	Varchar	50	It is the name of the patient.
4.	Age	Int	11	It includes the age of the patient.
5.	Sex	Tinyint	1	It defines the gender of the patient.
6.	Phone	Bigint	20	It is the contact number of each patient
7.	Ward	Int	11	It is the ward of the hospital to which the patient is assigned to

Table 3: Doctor details

S.No	Input	Datatype	Length	Description
1	Id	Int	11	This is the unique id assigned to each doctor..
2.	NMC no.	Varchar	50	It is the registration number given to the doctor by medical council.
3.	Name	Varchar	50	It is the name of the doctor.
4.	Department	Varchar	50	It includes the department to which the doctor belongs to.
5.	Phone	Bigint	20	It is the contact number of each doctor.
6.	Email	Varchar	11	It is the email address for each doctor.

*Note: The validation for each data type is done using the front end user interface.

Table 4: Employee details

S.No	Input	Datatype	Length	Description
1	Id	Int	11	It is the serial id of each registration.
2.	Name	Varchar	50	It is the name of the staff.
3.	Hospital id	Varchar	50	This is the unique id given to each employee.
4.	Phone	Bigint	20	It is the contact number of each staff.
6.	Address	Varchar	50	It is the address of each staff.
7.	Email	Varchar	50	It is the email address for each staff.

*Note: The validation for each data type is done using the front end user interface.

Table 5: Update details

S.No	Input	Datatype	Length	Description
1	Id	Int	11	It is the table id on which updation is done..
2.	Table Name	Varchar	50	It is the name of the table.
3.	Data Id	Varchar	50	This is the field id which is updated..
4.	Old Value	Varchar	50	It is the old value placed in the table..
6.	New Value	Varchar	50	It is the new value placed in the table.
7.	Modified	Varchar	50	It is the date and time modified.

*Note: The validation for each data type is done using the front end user interface.

3. Output Requirement

Since this project consists of backend, the required output would be

- Formation of .csv file
- Read the data from .csv format
- Data migration to MS Access
- Updated table in MS Access

4. Security Requirement

- The security approach for this system is the requirement of username and password to access the database. This prevents the unauthorized access of the data and ensures data protection.

5. METHODOLOGY

5.1 SOFTWARE DEVELOPMENT LIFE CYCLE

For this project, we have used the Incremental Model of Software Process Model. This model combines linear sequential model with the iterative prototype model. When an incremental model is used, the first increment is a “core product”. The plan addresses the modification of the core product to meet the needs of the customer and the delivery of additional features and functionality. Each linear sequence produces a deliverable “increment” of the software.

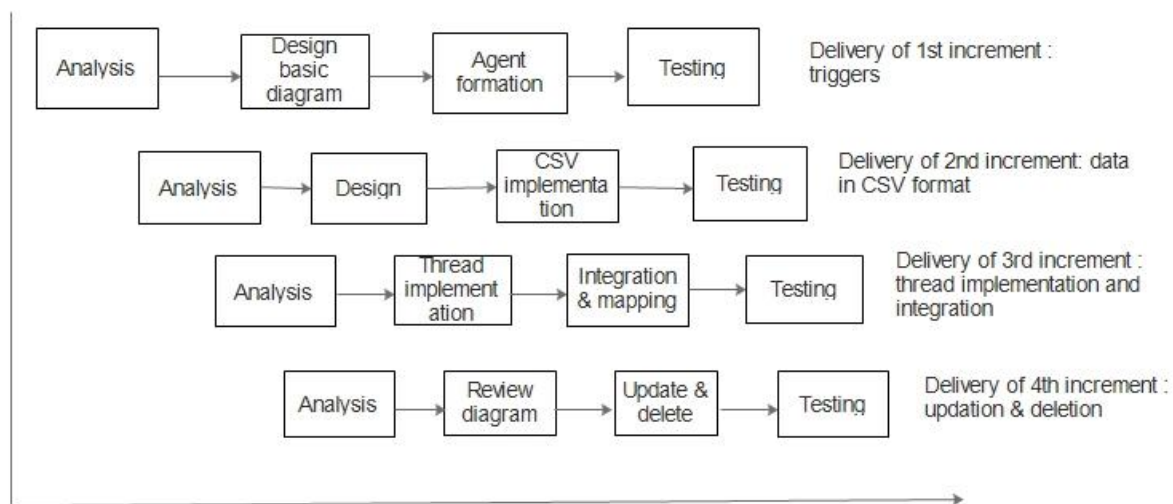


Figure 1: Incremental Model

The process is repeated following the delivery of each increment, until the complete product is produced.

Incremental model includes the following phases:

Analytic Phase- In this phase, the requirements of the software were analyzed which resulted in “Software Requirement Specifications”.

Design Phase- In this phase, analysis was translated into the system’s design. Context Diagram, Use-Case Diagram and Class Diagram were developed.

Coding Phase- This phase involvement the coding as per the design and formation of a working system at the end of the process.

Testing Phase- In this phase, the system was tested. With each testing, certain changes were made as per the suggestion. This was done in an incremental manner until a satisfactory system was made.

The deliverables at the end of each increment are specified below:

1st increment: Trigger and agent formation

2nd increment: CSV Formation

3rd increment: Thread implantation

4th increment: Review update and delete

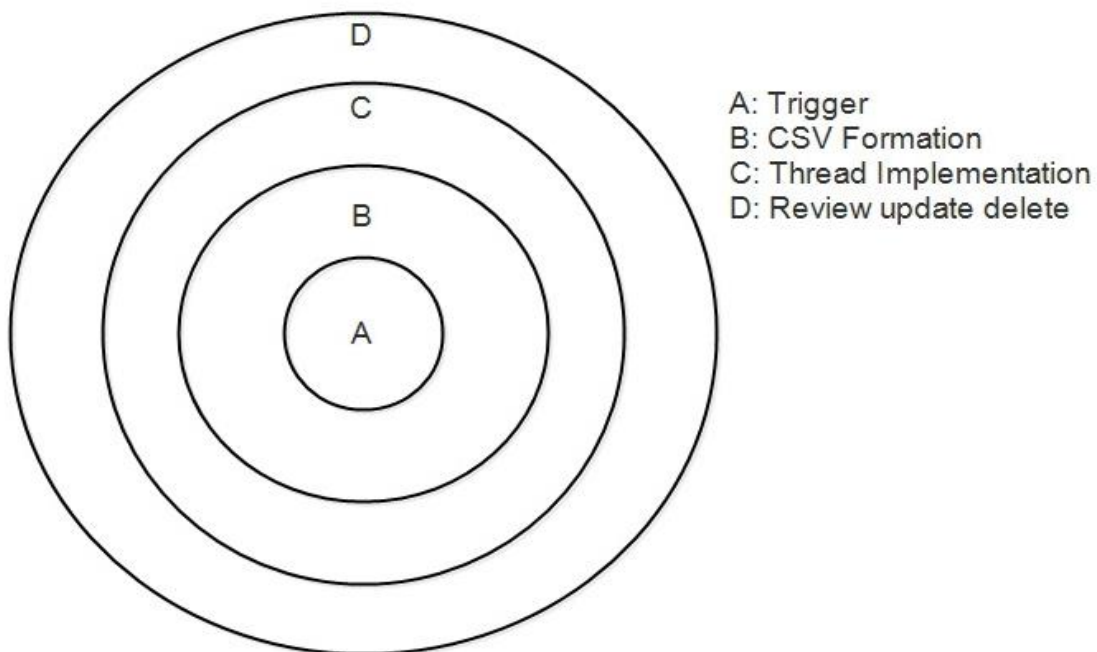


Figure 2: Integrated View of Increments

Each stage of incremental model adds some functionality to the product and passes it on to the next stage. The first increment resulted in creation of a plan for the next increment and so on. It, iteratively enhance the requirements until the final software is implemented.

Our project which implements the Incremental Model, comprises of four increments which are discussed as below:

INCREMENT 1: Agents: Triggers and Windows Task Scheduler

We need to audit the existing 3 to 4 tables in our system without prolonging transactions. For this we have built database triggers as an agent to keep the record of the updated tables and its associated fields.

However, we have not built triggers to keep the record of the insertion operation. This is because every inserted data is kept in a queue and then can be taken into the chunks to form the CSV in a trivial manner. The real problem occurs when the update takes place in the data that has been considered already to form the CSV and then to reload back to MS Access, since the pointer would have gone past the row and the update would not have been considered at all.

To extract the data into CSV files, we have created a VBS script that is scheduled by windows task scheduler. This approach can also be implemented in UNIX system with the help of cron task using suitable script format. On windows platform, the VBS script executes BAT file. The BAT file is connected directly with executable JAR file. The scheduler can also directly schedule the BAT file. But a terminal execution dialog box appears during its execution. To remove this pop up execution terminal we have created an additional VBS script. VBS is an Active Scripting language developed by Microsoft that is modeled on Visual Basic. The task scheduler schedules this VBS file which in turn calls the bat and jar file.

Service Broker can also be used as an alternative to triggers but it is overcomplicated when used mainly to separate DML event from audit event. If not moving the data to another system, it does not seem to offer much benefit since we would still need to persist the INSERTED and DELETED tables in the DML triggers.

So, our preference is the use of triggers for dumping the changes into the table, and then in a separate process that is scheduled to run every X minutes read Y number of rows and process them. This does not matter if we have thousands of single row DML operation or a single thousand row DML operation; this process will just keep chugging along doing what it does [13].

INCREMENT 2: CSV Formation

The Windows Task Scheduler, in fact extracts the data in a chunk of 50. The reason behind extracting the data in chunks, and not extracting the data in one go, is not to create overload and hinder the smooth operation of the active database. In case if any error occurs during the CSV formation, the maximum data that is to be retransmitted is at most 50. This reduces the overhead of retransmitting the data right from the start.

Windows Task Scheduler is scheduled to run every 1st hour. The agent runs the jar file thereby creating the CSV file. During this process, the start point and increment values are read from the text file so as to extract the value from that particular start point from the MySQL database. After the CSV is formed, the start point value in the text file is incremented by the number of rows in the CSV. The increment value is incremented by 1.

When the Task Scheduler runs the agent for the second time, it reads the updated start point and increment values from the particular text file and then extracts the data from the MySQL database starting from the start point. After the CSV has been created successfully, the start point and increment values are updated accordingly.

The process goes on till the data is available in the respective table in MySQL database. The text file is stored in secondary memory.

INCREMENT 3: Thread Implementation, converting data from CSV to MS Access

Thread Implementation:

The existing database has a number of tables. The data is to be extracted from each table so as to form the CSVs respectively. We have used the concept of thread to perform this particular task. Each table in MySQL database is associated to a particular thread. The threads run in parallel to extract the data from associated table. Since the thread only reads the data from respective tables, the execution of one thread do not create hindrance to another though they operate on same database, but on different tables.

Converting data from CSV to MS Access:

We have another agent that the Windows Task Scheduler runs to load the CSV to MS Access database. This agent schedules VBS file which in turn runs a jar file. During this process the CSV file is first loaded and then converted into string. Commas are excluded from those CSV files. The file name is used as a table name with the exclusion of successor number. Once all string of a CSV row are loaded into string, it is now saved into array variable. These variables are parsed according to MS Access data format and sent to MS Access making a database connection. This operation is scheduled twice a day. The reason behind performing the data conversion and verification operation at this phase is that the CSV is created and stored in secondary storage and performing operations at this label does not hinder the operation of active database. If this operation is to be performed at the time of CSV formation, this would take a bit time in handling this and would hinder the smooth operation of the active database.

INCREMENT 4: Update and delete operation

MySQL database is used as primary database i.e. day to day transactions occur in this database in regular manner. Data can change several times. This information should be updated on MS Access database as well. We use the appropriate queries to update the respective tables in MS Access as well so as to synchronize with the MySQL database.

The next operation is the deletion of CSV file from secondary memory. Once copy of data to MS Access acknowledgement is received, the CSV files are deleted to free memory.

5.2 ALGORITHMS

A. ALGORITHM TO CREATE CSV FROM MySQL

1. Windows scheduler runs the jar every 1st hour.
2. Read content of text file as Hash Map and store in reader.
3. while (reader.readLine () != null) {
 - a. Create an ArrayList of Hash Map values.
 - b. int startpoint = array(0)
 - c. int increment = array (1)
 - d. ResultSet rs = execute query to extract data in chunk
 - e. while (rs. Next ()) {
 - i. Count++
 - ii. Write entire row to CSV
 - f. startpoint = startpoint + count
 - g. increment++
 - h. count = 0
 - i. write startpoint and increment to text file.

B. ALGORITHM TO READ CSV AND WRITE TO MS ACCESS

1. Windows scheduler runs the jar every day 12:30 AM and 2:30 AM
2. for (File file = dir. ListFiles ()) {
 1. Scanner inputStream = new Scanner (file)
 2. While (inputStream. hasNext ()) {
 - i. read entire row
 - ii. Extract field values in array
 - iii. Normalize values in accordance to MS Access
 - iv. Execute query to insert normalized values to respective tables in MS access}
3. Verify CSV file insertion
4. delete file

C. ALGORITHM TO STORE UPDATED CSV TO MS ACCESS AND TO UPDATE MS ACCESS TABLES

1. Windows scheduler runs the jar every day 12:30 AM and 2:30 AM
 2. For (File file : dir.ListFiles()){
 - a) Scanner inputStream = new Scanner (file)
 - b) While (inputStream. hasNext ()) {
 - i. read entire row
 - ii. Extract field values in array
 - iii. Normalize values in accordance to MS Access
 - iv. Execute query to insert normalized values to tbl_updated in MS access for backup}}
 3. If all threads are executed {
 - a) Scanner inputStream = new Scanner (file)
 - b) While (inputStream. hasNext ()) {
 - i. read entire row
 - ii. Extract field values in array
 - iii. Normalize values in accordance to MS Access
 - iv. Execute query to insert normalized values to respective tables in MS access for backup}}
 4. Verify update
 5. Delete csv_update
- }

5.3 PICTORIAL REPRESENTATION OF ALGORITHM

The following activity diagram is the pictorial representation of the above algorithm.

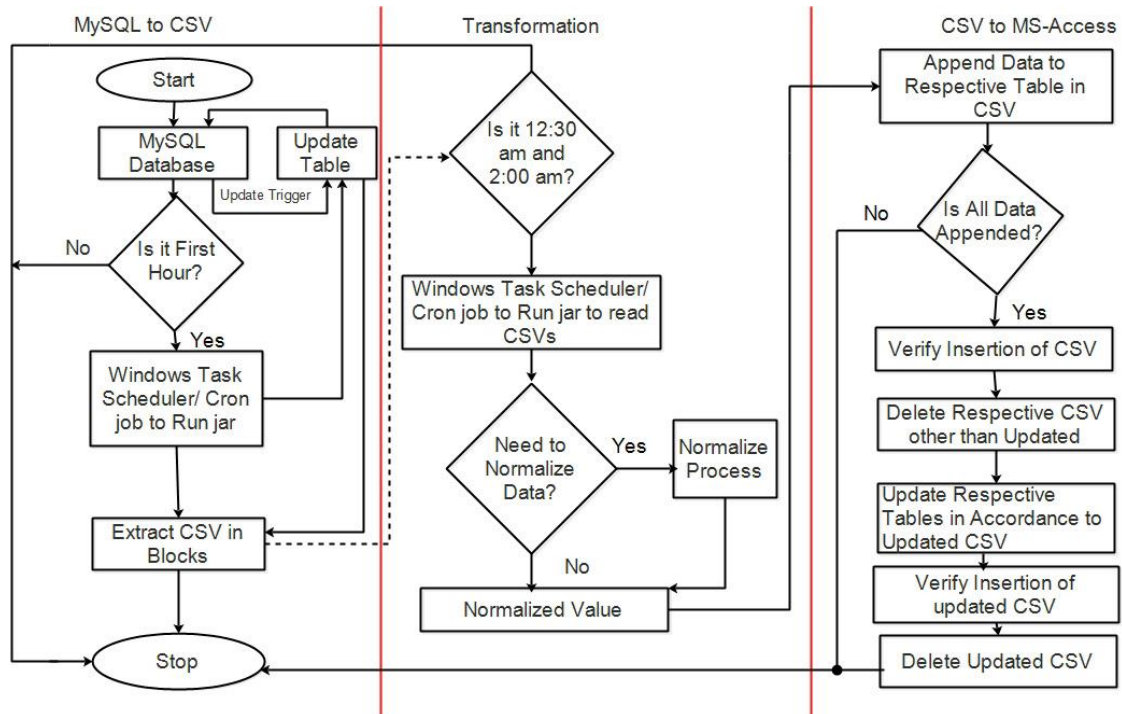


Figure 3: Pictorial representation of Algorithm

6. TIME SCHEDULING

The project schedule has been followed as per requirements and time constraints involved in chart below. The chart does not include numerous informal conversation with the user of the system which had further aided in the development of the system.

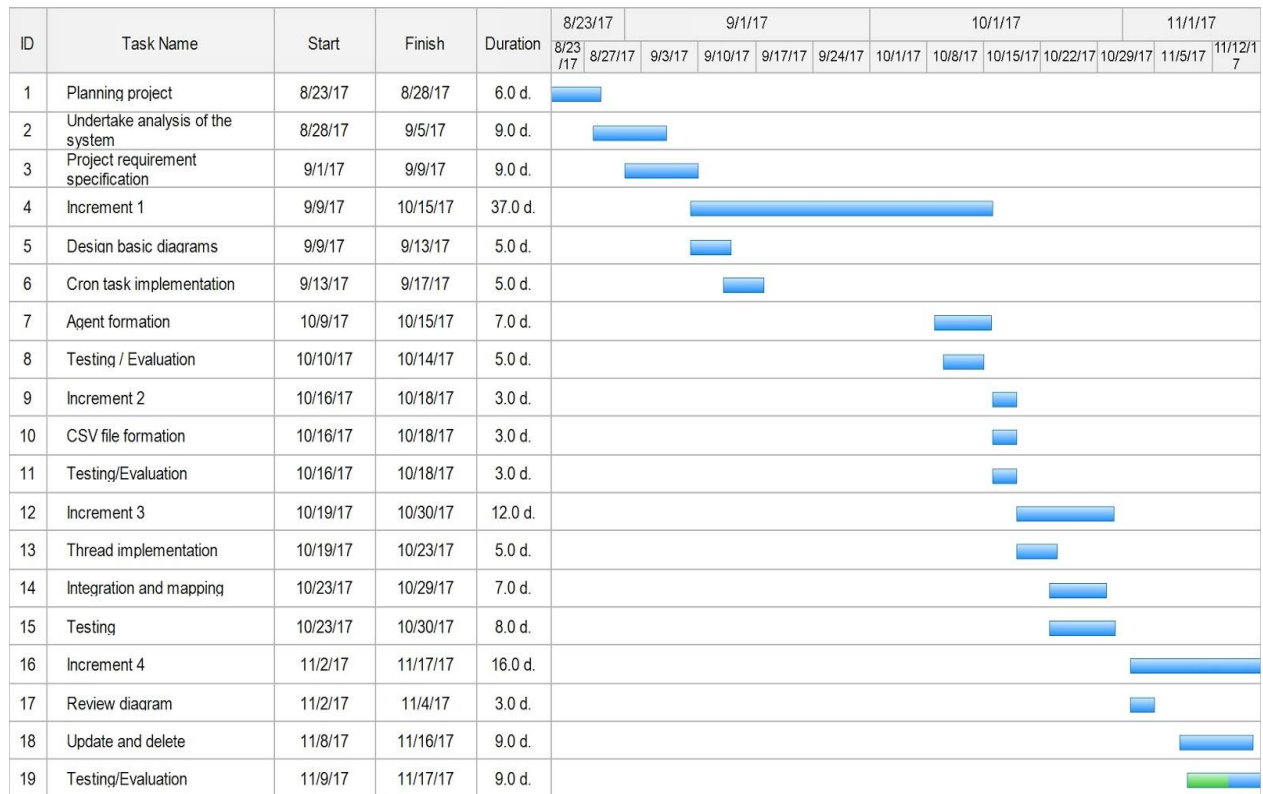


Figure 4: Time Scheduling

7. TOOLS AND TECHNIQUES

The tools and techniques that we have deployed in our system are explained below:

1. MYSQL DATABASE

It is used as transaction Database for doing various activities. It runs as a server and allows multiple users to manage and create numerous databases. MySQL is an open-source Relational Database Management System(RDBMS), which is the basis for SQL, and for all modern database system. The data in RDBMS is stored in database objects called tables. The uses of SQL are to execute queries against a database, to retrieve, insert, update, delete the records on the database, to create new database, to create new tables, to create stored procedures in a database, to create views in database, to set permissions on table, procedures and views.

2. MS-ACCESS DATABASE

Microsoft Access is an information management tool that helps you store information for reference, reporting, and analysis [14]. We will be using MS- Access to backup Database when MySQL database fails or crash. This backup provides an alternative to operate the whole existing system on MS-Access if needed.

3. CRON

CRON is software utility used as time based job scheduler in Unix like operating system or servers. The Cron daemon is a long-running process that executes commands at specific dates and times. We can use this to schedule activities, either as one-time events or as recurring tasks [15]. Scripts used in database conversion and free hour detection can be done using this utility tool.

4. WINDOWS TASK SCHEDULER

Task Scheduler is a component of Windows based systems that provides the ability to schedule the launch of programs or scripts at pre-defined times or after specified time intervals. This can be taken as alternative for CRON on windows based systems.

5. JAVA

Java is used as scripting language that is concurrent, class- based, object-oriented. It is intended to let application developers “write once, run anywhere” (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.

6. XAMPP

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Simple, lightweight Apache distribution is used to create a local web server for testing and deployment purposes.

7. Edraw Max

It is used as a tool for making Gantt charts and UML diagrams.

8. SYSTEM DESIGN

This field contains the detail description of the system, and the associated UML diagrams.

8.1 CONTEXT DIAGRAM

Context diagram describes the overall representation of the system. The diagrammatic representation is given below:

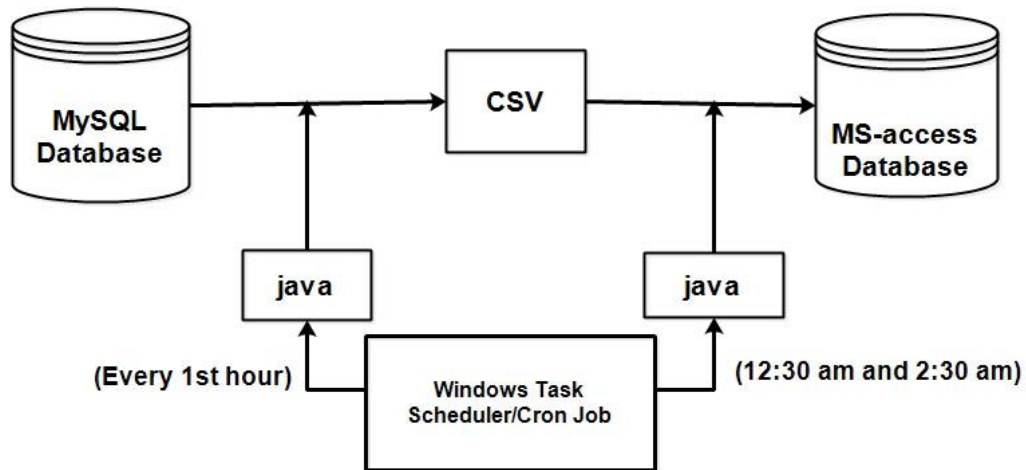


Figure 5: Context diagram

8.2 SYSTEM ARCHITECTURE

This system proposes a design of new prototype architecture on conversion with flexibilities and automatic restructuring.

Figure below illustrates the system architecture:

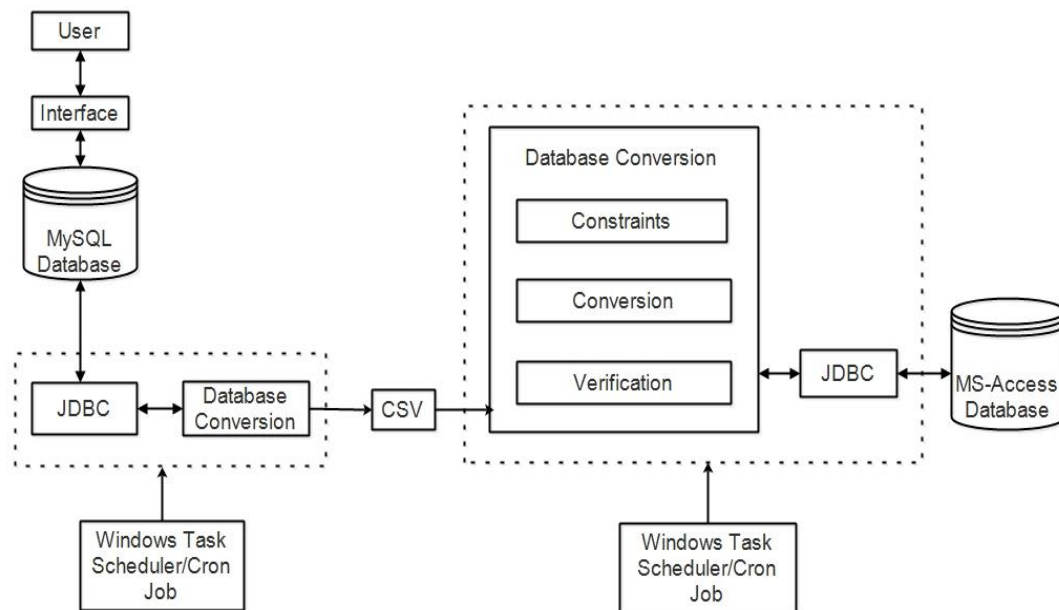


Figure 6: System Architecture

The system architecture is the conceptual design of following important components:

1. **JDBC:** The way to access the database and manipulate data from any platform.
2. **Database Conversion:** First database conversion is run automatically by the system, every first hour, thereby creating the chunks of data from associated table.
Second database conversion is important component of the architecture and is automatically carried out by the system, two times a day. This component is responsible for maintaining constraints and integrity, data conversion and verification.
3. **MySQL database:** Is an online database, the user directly interacts with end-user who can insert and update into the database tables, select tables from the database. There is no provision for deletion of the data.

4. **MS Access database:** It is used for backup purpose and is the secondary database.

The user interface is out of our scope.

8.3 USE CASE DIAGRAM

Use case diagram shows the relation between actors, scenarios and system boundary to reach the user goal.

The agent runs the jar file to migrate the data and to update the data. Firstly, to migrate the data to the destination database, CSV file is formed and secondly, when all the data is migrated, the associated table is updated. Before migrating and updating the data, integrity and constraints, conversion and verification process is carried out.

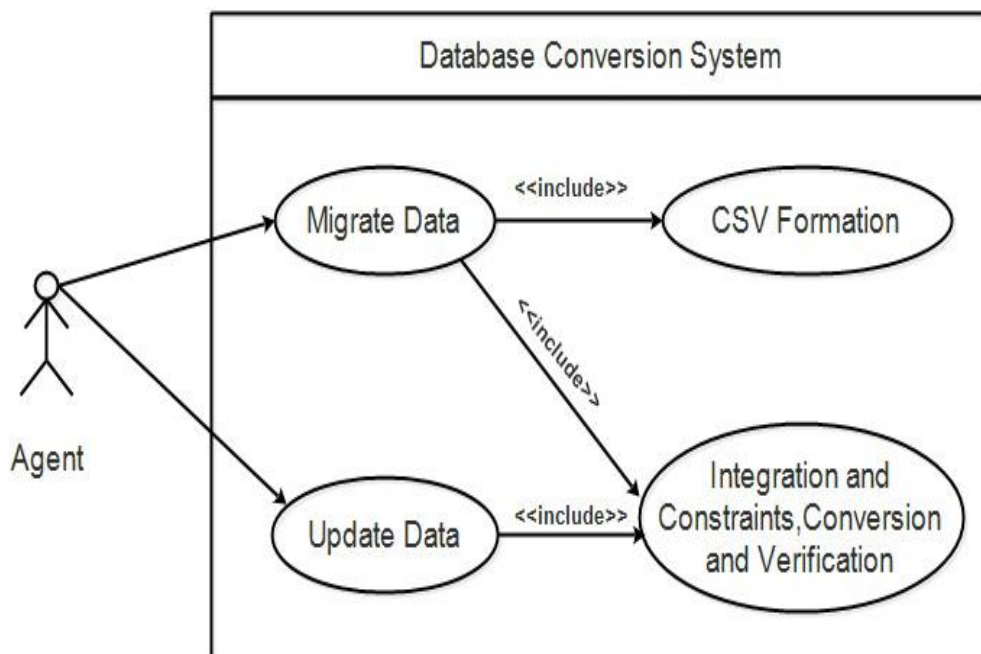


Figure 7: Use Case Diagram

8.4 SYSTEM SEQUENCE DIAGRAM

A system sequence diagram (SSD) is a sequence diagram that shows, for a particular scenario of a use case, the events that external actors generate, their order, and possible inter-system events. System sequence diagrams are the visual summaries of the individual use cases.

8.4.1 SEQUENCE DIAGRAM FOR MYSQL TO CSV

The following SSD explains visually, the sequence of steps, in generating the associated CSV file.

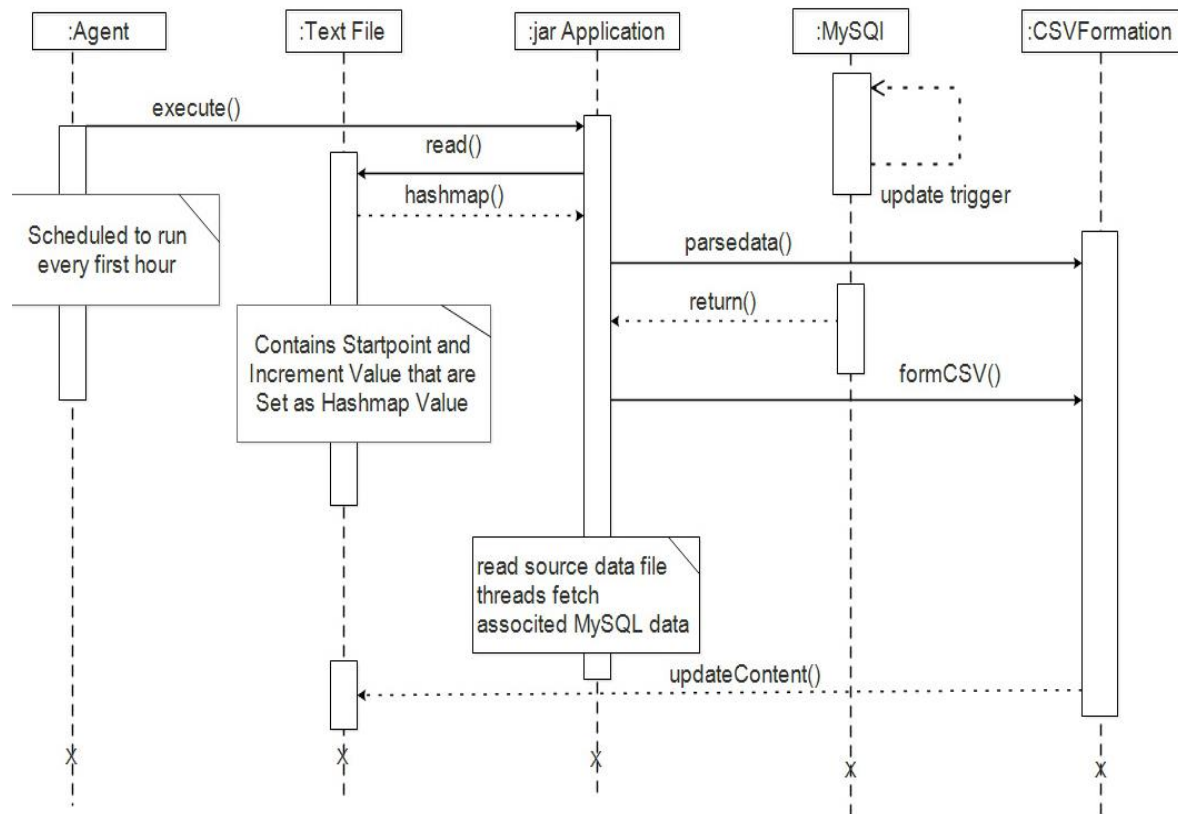


Figure 8: System Sequence Diagram from MySQL to CSV

8.4.2 SEQUENCE DIAGRAM FOR CSV TO MS-ACCESS

The following SSD explains visually, the sequence of steps, in mapping the CSV file to the associated table and also updating the associated tables, once all the mapping operation is completed.

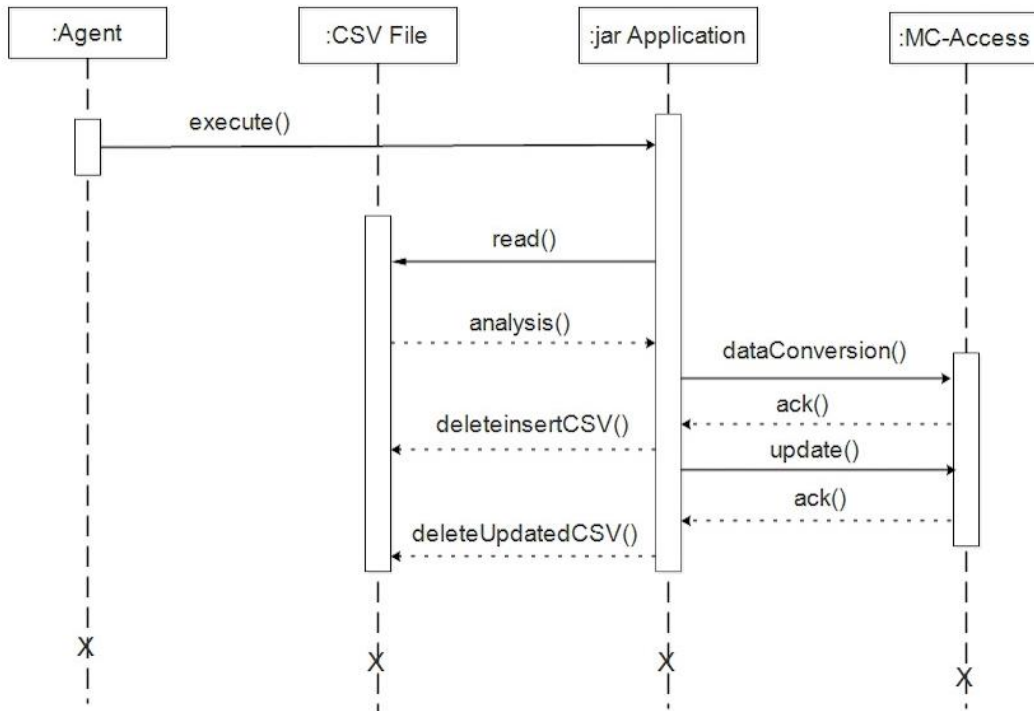


Figure 9: System Sequence Diagram for CSV to MS Access

8.5 CLASS DIAGRAM

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

8.5.1 CLASS DIAGRAM FOR CSV FORMATION

The associated classes, their attributes and methods, and the relationship between them in CSV formation is as shown:

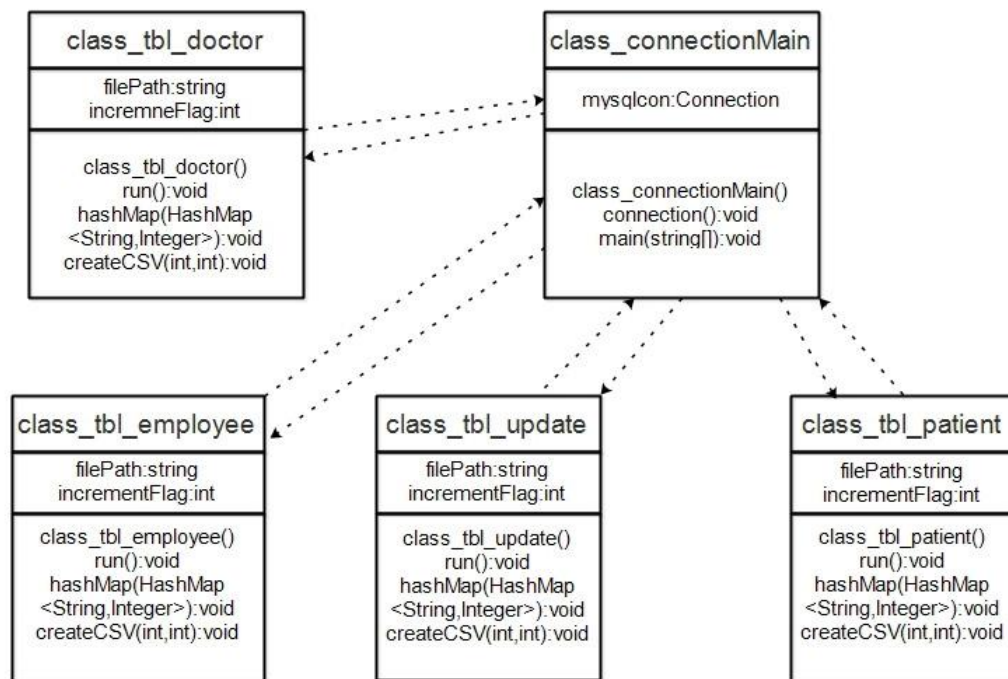


Figure 10: Class Diagram for formulation of CSV

8.5.2 CLASS DIAGRAM TO LOAD DATA FROM CSV TO MS-ACCESS

The associated classes, their attributes and methods, and the relationship between them in mapping the CSV to the associated table, and performing the update operation is as shown:

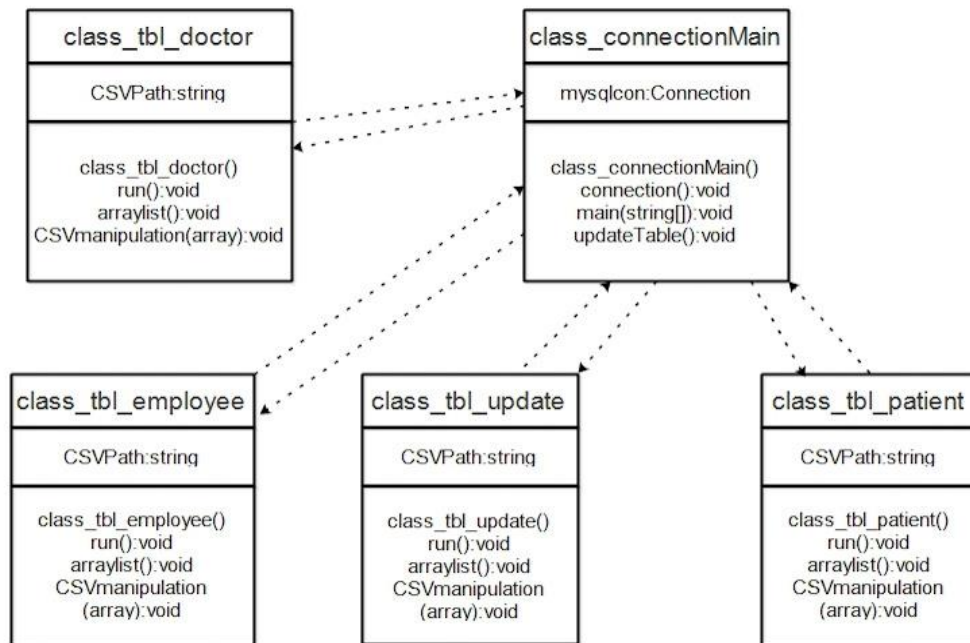


Figure 11: Class Diagram to load data from CSV to My Access

9. BUDGET

9.1 FUNCTION POINT

Information Domain	Value	Weight Count	Total
No. of user input	24	4	96
No. of user output	24	5	120
No. of user inquires	4	4	16
No. of logical files	8	10	80
No. of external interface	0	7	0
Count			312

Count Total= 312

Function point (FP) = Count total * Complexity Multiplier

$$= 312 * [0.65 + 0.01 * \sum 1^4 F_i]$$

$$= 312 * 1.07$$

$$= 333.84$$

Average Productivity = 10 FP/pm

Labor Rate= Rs. 20,000 per month

Effort = FP/(Average Productivity)

$$= 333.84 / 10$$

$$= 33.3$$

Total Project Cost = FP* Labor rate/(productivity)

$$= 333.84 * (20000) / 10$$

$$= \text{Rs. } 667680$$

9.2 LINES OF CODE

Estimated LOC = 900

Average Productivity = 30 LOC/pm

Labor Rate = Rs. 20000

Now,

Estimated Project Cost = Estimated LOC * Cost/LOC

= $900 * \text{Labor rate} / (\text{Average Productivity})$

= $900 * 20000 / (30)$

= Rs. 600000

Estimated Efforts = $\text{KLOC} / (\text{Average Productivity})$

= $900 / 90$

= 10

10. TESTING

Testing is very important as it is necessary to determine whether our work is correct or not. So, we have created a test plan in which validation will be tested for our backend project. The system is tested for the normal condition.

10.1 TESTING TABLE

Table 6: Testing table

Test No.	Table	Test	Expected Result	Actual Result	Evidence	Status
1	tbl_doctor	First Scheduler implementation	First CSV formation	First CSV formed	Test 1.1	Success
2	tbl_doctor	First Scheduler implementation	Startpoint & endpoint increment	Startpoint & endpoint increases	Test 2.1	Success
3	tbl_doctor	Second Scheduler implementation	Second CSV formation and increment occurs	Second CSV formed and increment occurs	Test 3.1	Success
4	tbl_doctor, tbl_update	Update in the existing data	Table is updated in MYSQL and updated data is stored in different table by the use of trigger.	Table is updated in MYSQL and the updated data is stored in tbl_update	Test 4.1	Success
5	tbl_doctor, tbl_update	Update in the MS Access	Table is updated in MS Access and existing CSV file is deleted	Existing CSV file gets deleted after the update in MS Access	Test 5.1	Success

10.2 LIST OF TESTING

All testings are given below:

1. Table: tbl_doctor

Test 1.1

Purpose: To know what happens if the first scheduler is implemented.

Input: First scheduler is implemented.

Expected Output: First CSV file is formed.

Status: Success

	id	nmc_number	name	department	phone	email
<input type="checkbox"/> Edit Copy Delete	1	DOC-2013-76-420	Hari Bahadur	Radiologist	9842042020	i.am.hari.and.you@nmc.org
<input type="checkbox"/> Edit Copy Delete	2	DOC-2013-79-515	Madan Bahadur	Sergion	9852052020	heromadn@nmc.org
<input type="checkbox"/> Edit Copy Delete	3	DOC-2016-75-429	Jitu Nepal G	Cardiologist	9812657890	nepaljitu@yahoo..com
<input type="checkbox"/> Edit Copy Delete	4	DOC-2010-72-205	Sundari G	Cardiology	9876234156	sun.dari@bcd.com
<input type="checkbox"/> Edit Copy Delete	5	DOC-2013-34-33	Kancha Bahadur	pathology	9836382834	k.bahadur@jdsfh.com
<input type="checkbox"/> Edit Copy Delete	6	Doc-1993-378-5872	Rama Kumari	Gynocology	9834643345	gynorama@fdk.com
<input type="checkbox"/> Edit Copy Delete	7	DOC-2007-34-987	Harka Prasad Pahari	Oncology	9873457253	pahariharka@ghj.com
<input type="checkbox"/> Edit Copy Delete	8	DOC-2015-398-793	Santosh Lal Pandit	Pedesterian	9987326534	i.am.pandit@sdfh.com

Figure 12: tbl_doctor before first scheduler implementation


Name	Date modified	Type	Size
 tbl_doctor1	11/16/2017 12:07 ...	Microsoft Excel C...	1 KB

Figure 13: First CSV formation

2. Table: tbl_doctor

Test 2.1

Purpose: To know what happens if the first scheduler is implemented.

Input: First scheduler is implemented.

Expected Output: Startpoint and endpoint increases.

Status: Success

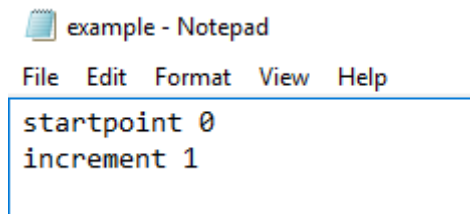


Figure 14: Initial Startpoint and Endpoint

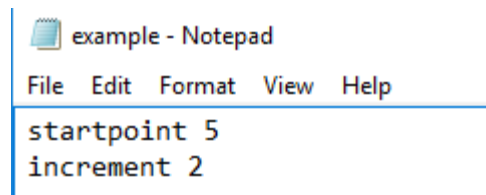


Figure 15: Startpoint and endpoint increases

3. Table: tbl_doctor

Test 3.1

Purpose: To know what happens if the second scheduler is implemented.

Input: Second scheduler is implemented.

Expected Output: - Second CSV formation
- Startpoint and endpoint increases.

Status: Success



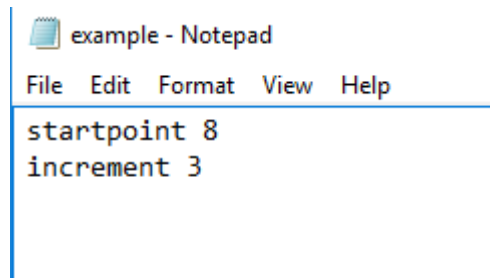
Name	Date modified	Type	Size
 tbl_doctor1	11/16/2017 12:07 ...	Microsoft Excel C...	1 KB
 tbl_doctor2	11/16/2017 12:09 ...	Microsoft Excel C...	1 KB

Figure 16: Second CSV file formation



```
startpoint 8  
increment 3
```

Figure 17: Increment after second scheduler implementation

4. Table: tbl_doctor, tbl_update

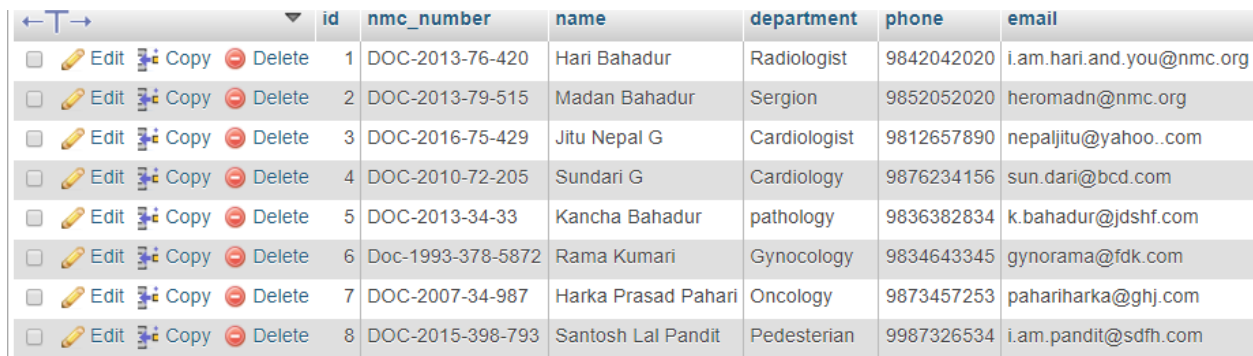
Test 4.1

Purpose: To know what happens when data is updated in MYSQL. We have updated the field name “Hari Bahadur” to “Hari B. Rana Magar”

Input: Data is updated in MYSQL.

Expected Output: - The updated data is stored in separate tbl_update by the use of trigger.

Status: Success



	id	nmc_number	name	department	phone	email
<input type="checkbox"/> Edit Copy Delete	1	DOC-2013-76-420	Hari Bahadur	Radiologist	9842042020	i.am.hari.and.you@nmc.org
<input type="checkbox"/> Edit Copy Delete	2	DOC-2013-79-515	Madan Bahadur	Sergion	9852052020	heromadn@nmc.org
<input type="checkbox"/> Edit Copy Delete	3	DOC-2016-75-429	Jitu Nepal G	Cardiologist	9812657890	nepaljitu@yahoo..com
<input type="checkbox"/> Edit Copy Delete	4	DOC-2010-72-205	Sundari G	Cardiology	9876234156	sun.dari@bcd.com
<input type="checkbox"/> Edit Copy Delete	5	DOC-2013-34-33	Kancha Bahadur	pathology	9836382834	k.bahadur@jdsfh.com
<input type="checkbox"/> Edit Copy Delete	6	Doc-1993-378-5872	Rama Kumari	Gynecology	9834643345	gynorama@fdk.com
<input type="checkbox"/> Edit Copy Delete	7	DOC-2007-34-987	Harka Prasad Pahari	Oncology	9873457253	pahariharka@ghj.com
<input type="checkbox"/> Edit Copy Delete	8	DOC-2015-398-793	Santosh Lal Pandit	Pedesterian	9987326534	i.am.pandit@sdfh.com

Figure 18: Initial tbl_doctor in MYSQL



	id	nmc_number	name	department	phone	email
<input type="checkbox"/> Edit Copy Delete	1	DOC-2013-76-420	Hari B. Rana Magar	Radiologist	9842042020	i.am.hari.and.you@nmc.org
<input type="checkbox"/> Edit Copy Delete	2	DOC-2013-79-515	Madan Bahadur	Sergion	9852052020	heromadn@nmc.org
<input type="checkbox"/> Edit Copy Delete	3	DOC-2016-75-429	Jitu Nepal G	Cardiologist	9812657890	nepaljitu@yahoo..com
<input type="checkbox"/> Edit Copy Delete	4	DOC-2010-72-205	Sundari G	Cardiology	9876234156	sun.dari@bcd.com
<input type="checkbox"/> Edit Copy Delete	5	DOC-2013-34-33	Kancha Bahadur	pathology	9836382834	k.bahadur@jdsfh.com
<input type="checkbox"/> Edit Copy Delete	6	Doc-1993-378-5872	Rama Kumari	Gynecology	9834643345	gynorama@fdk.com
<input type="checkbox"/> Edit Copy Delete	7	DOC-2007-34-987	Harka Prasad Pahari	Oncology	9873457253	pahariharka@ghj.com
<input type="checkbox"/> Edit Copy Delete	8	DOC-2015-398-793	Santosh Lal Pandit	Pedesterian	9987326534	i.am.pandit@sdfh.com

Figure 19: Updated tbl_doctor in MYSQL

Database Conversion System With Agents

<div><div>← T →</div></div>				id	table_name	data_id	field	old_value	new_value	modified
<div><div><div></div></div></div>	<div>Edit</div>	<div>Copy</div>	<div>Delete</div>	3	tbl_doctor	2	department	sergion	sergionary	2017-10-31 09:23:25
<div><div><div></div></div></div>	<div>Edit</div>	<div>Copy</div>	<div>Delete</div>	4	tbl_employee	4	address	Kalanki	Putalisadak	2017-10-31 09:46:57
<div><div><div></div></div></div>	<div>Edit</div>	<div>Copy</div>	<div>Delete</div>	5	tbl_patient	5	sex	0	1	2017-10-31 09:56:13
<div><div><div></div></div></div>	<div>Edit</div>	<div>Copy</div>	<div>Delete</div>	6	tbl_doctor	3	name	Jitu Nepal	Jitu Nepal G	2017-11-14 17:13:53
<div><div><div></div></div></div>	<div>Edit</div>	<div>Copy</div>	<div>Delete</div>	7	tbl_doctor	1	name	Hari Bahadur	Hari B. Rana Magar	2017-11-16 12:13:24

Figure 20: The updated data stored in table "tbl_update"

5. Table: tbl_doctor, tbl_update

Test 5.1

Purpose: To know what happens to the data in MS access if existing data is updated in MYSQL.

Input: Data is updated in MYSQL.

Expected Output: - The data is updated in MS Access and existing CSV file gets deleted.

Status: Success

ID	table_id	nmc_numbe	name	department	phone	email	Click to Add
351	1	DOC-2013-76-4	Hari Bahadur	Radiologist	9842042020	i.am.hari.and.y	
352	2	DOC-2013-79-5	Madan Bahadu	sergionary	9852052020	heromadn@nn	
353	3	DOC-2016-75-4	Jitu Nepal G	Cardiologist	9812657890	nepaljitu@yah	
354	4	DOC-2010-72-2	Sundari G	Cardiology	9876234156	sun.dari@bcd.i	
355	5	DOC-2013-34-3	Kancha Bahadu	pathology	9836382834	k.bahadur@jds	
356	6	Doc-1993-378-5	Rama Kumari	Gynocology	9834643345	gynorama@fdk	
357	7	DOC-2007-34-9	Harka Prasad P	Oncology	9873457253	pahariharka@g	
358	8	DOC-2015-398-	Santosh Lal Par	Pedesterian	9987326534	i.am.pandit@s	
*(New)							

Figure 21: Initial tbl_doctor in MS Access

ID	table_id	nmc_numbe	name	department	phone	email	Click to Add
351	1	DOC-2013-76-4	Hari B. Rana Magar	Radiologist	9842042020	i.am.hari.and.you@nmc.org	
352	2	DOC-2013-79-5	Madan Bahadur	sergionary	9852052020	heromadn@nmc.org	
353	3	DOC-2016-75-4	Jitu Nepal G	Cardiologist	9812657890	nepaljitu@yahoo..com	
354	4	DOC-2010-72-2	Sundari G	Cardiology	9876234156	sun.dari@bcd.com	
355	5	DOC-2013-34-3	Kancha Bahadur	pathology	9836382834	k.bahadur@jdshf.com	
356	6	Doc-1993-378-5	Rama Kumari	Gynocology	9834643345	gynorama@fdk.com	
357	7	DOC-2007-34-9	Harka Prasad Pahar	Oncology	9873457253	pahariharka@ghj.com	
358	8	DOC-2015-398-	Santosh Lal Pandit	Pedesterian	9987326534	i.am.pandit@sdfh.com	
*(New)							

Figure 22: Updated tbl_doctor in MS Access

MySQLtoCSV > folder_tbl_doctor > csv_doctor

I	Type	Size
This folder is empty.		

Figure 23: CSV deleted after converted to MS Access

MySQLtoCSV > folder_tbl_updated > csv_updated

Type	Size
This folder is empty.	

Figure 24: CSV deleted after update of existing data in MS Access

11. RESULT AND DISCUSSION

The inserted and updated data entities on MySQL database can be converted into CSV file format. The script is triggered and the system converts the data into CSV file format in the block of 50 entities. These CSV blocks are stored on secondary memory. Another script is triggered which parse these CSV files into strings. With the conversion, verification and integrity checking processes, data are sent to associated table in MS access database. After the insertion operation is finished, associated CSV file is deleted and the update operation for associated table is carried out. Once the update operation is completed, the CSV file containing the updated values is deleted. The CSV file is deleted so as to free up the memory as the CSV file grows in passage of time.

12. FUTURE EXTENSION

As this project is only a backend project, this can be further extended to make it a runnable software and can be implemented. Our future extension includes:

- The user interface can be integrated with this backend project which makes this complete.
- We have used MySQL at present. But the databases like SQLite, PostgreSQL could also be used.
- We have worked on the few tables for this project. However, this can be used for large number of data with more tables.

13. REFERENCE

- [1] Title: “MySQL Tutorial” [online]. Available, URL: “<https://www.tutorialspoint.com/mysql/>” [Accessed: 16-july-2017 at 4:12 pm].
- [2] Title: “MS Access Tutorial”, [online]. Available, URL: “<https://www.tutorialspoint.com/mysql/>” [Accessed: 16-july-2017 at 4:12 pm].
- [3] Siti Z.Z. Abilin, Suzana Ahmed, Whael M.S. Yafooz, Volume I, Towards Flexible Database Conversion with Automatic Restructuring, 2010
- [4] Siti Z.Z. Abidin, Suzana Ahmed, Whael M.S. Yafooz, ISSUE 11, Volume 9, A New System Architecture for Flexible Database Conversion, November 2010
- [5] Title: “SQL-RDBMSConcepts”. [online]. Available, URL: “<https://www.tutorialspoint.com/sql/sql-rdbmsconcepts.htm>.” [Accessed: 18-Oct-2017 at 11:05 am].
- [6] Title: “PL/SQL-Triggers”, [online]. Available, URL: “https://www.tutorialspoint.com/plsql/plsql_triggers.htm.” [Accessed: 2-Nov-2017 at 1:29 pm].
- [7] Title: “How to create a CSV File”, [online]. Available, URL: “<https://www.computerhope.com/issues/ch001356.htm>.” [Accessed: 5-Nov-2017 at 11:20 pm].
- [8] Title: ‘Introducing JSON’,[online].Available, URL: “<http://www.json.org/> .” [Accessed: 5-Nov-2017 at 1:50 pm].
- [9] Title: “JAVA SE Technologies- Database” [online]. Available, URL: “<http://www.oracle.com/technetwork/java/javase/jdbc/index.html>.” [Accessed: 5-Nov-2017 at 3:15 pm].
- [10] Title: “Open Database Connectivity(ODBC)” [online]. Available, URL: “<http://searchoracle.techtarget.com/definition/Open-Database-Connectivity>.” [Accessed: 2- Nov-2017 at 7:30 pm].
- [11] Title: “Introduction to JAVA Threads” [online]. Available, URL: “<http://www.wideskills.com/java-tutorial/java-threads-tutorial>” [Accessed: 11-Nov-2017at 1:21 pm]
- [12] Roger S. PressMan, Seventh Edition, Software Engineering, 2010

[13] Title: “Is Service Broker the best choice for auditing data changes on SQL Server Express” [online]. Available, URL: “<https://dba.stackexchange.com/questions/120248/is-service-broker-the-best-choice-for-auditing-data-changes-on-sql-server-express>” [Accessed:17-November-2017 at 2:09 pm].

[14] Title: “What is Microsoft Access used for?” [online]. Available, URL: “<http://www.opengatesw.net/ms-access-tutorials/What-Is-Microsoft-Access-Used-For.htm>” [Accessed: 18-july-2017 at 8:29 pm].

[15] Title: “ARCHIVE: What are Cron and Crontab, and how do I use them” [online]. Available, URL: “<https://kb.iu.edu/d/afiz>” [Accessed: 18-july-2017 at 9:29 pm].