Engineering and Maintenance Tracking System (EMTS) Software Design

Orlando Marrero Luna
Master of Engineering in Computer Engineering
Advisor: Alfredo Cruz, Ph.D.
Electrical & Computer Engineering and Computer Science Department
Polytechnic University of Puerto Rico

Abstract *— In today's complex and technical world, good management of the data processing and technical information in the manufacturing environment of medical device companies are vital to the success of these organizations. The Engineering and Maintenance Tracking System is a software designed for the Haemonetics Medical Device Company with the purpose of enhancing their technical data tracking activities within its Engineering department by interconnecting the five most critical operational areas. This software written in Java code consist of a Graphic User Interface that interconnect the company data among those critical operational areas, by connecting each other by means of an SQL database located in a virtual server. The need for maintaining centralize data related to the daily process activities is crucial for the decision-making process of these organization, since they constantly manage auditable information subject to review by Federal Agencies.*

Key Terms *— Cloud Platform, Java Classes, JFrame, JTree.*

INTRODUCTION

The Haemonetics Fajardo branch is a company whose purpose is to develop medical devices for the blood filtering technology. The company is a leader in blood solutions for clients. As part of its daily activities in the manufacturing process, the automation department of the company which belongs to the engineering unit of the corporation constantly experiences problems in the daily technical activities of the engineering and manufacturing departments. These problems produce constant errors in the manufacturing process, and a high downtime reports because the technical data among the engineering department is not centralize. This inefficiency results in high operational costs for the company.

The correct centralization of data is an important factor to solve major business issues in today's manufacturing environments [1]. Recent studies demonstrate that there is a direct relationship between the machine efficiency and the data collection process. To achieve this efficiency, it is important that the data is collected and monitored constantly. Another concept that is revolutionizing the manufacturing industry is what is known as Internet of things. This concept reveals the importance of the collection of data in real time to improve the efficiency of the production lines [2].

The idea of this project came from the need of the medical device company Haemonetics to work with a real day-to-day problem of data handling in the operations of this organization. This leads us to propose the development of this software that can offer a real alternative to improve the company critical data management systems, making their process more accessible for users, resulting in a more efficient and robust system.

The paper explains with technical detail the development process of the Engineering and Maintenance Tracking System software design, its business model, software structure, functional design, and purpose.

METHODOLOGY & DESIGN

In any software design, it is necessary to know the requirements and needs of the affected people or stakeholders. In this project a software was

designed and developed to enhance the technical management process within the Engineering and Maintenance Department of the Haemonetics Company. The company current data collection processing of different critical areas within the company are being manage manually and is very inefficient because all key users do not have access to vital information needed for the execution of their respective tasks. The following process areas are very critical in the company daily operations:

- Machine Information Management
- Drawing Management
- Software Back-up Management
- Work Order Process
- Machine Component Replacement System

The goal with the development of this software design "Engineering and Maintenance Tracking System", is to centralize the data entry during the manufacturing process in the areas of preventive maintenance, software back-up storage, control of technical drawings, data management of manufacturing machines, and control of replacement components in a more robust database process. The overall design concept of this project was broken down in the following phases:

- Database Design
- Application and Programming
- Hardware

Database Design

For the data management a confine database in the application was designed. This application was done under the Microsoft SQL Server 2008 R2 platform. The software is linked to a series of tables to store vital information such as: the user's information with levels of permission and authentication, work order data tracking, machine downtime information, inventory of components, machine drawings, and software back-up's. The database used is Microsoft SQL which resides in a virtual server, and the users can access the database through the software applications with the proper credentials. Only the administrator will have privilege to access and modify the database.

An entity relationship diagram (ERD) was designed to better understand the relationship between each of the tables. Figure 1 shows the Entity Relationship Diagram for the Engineering and Maintenance Tracking System Software where the graphical relationship between the tables (entities) can be observed.

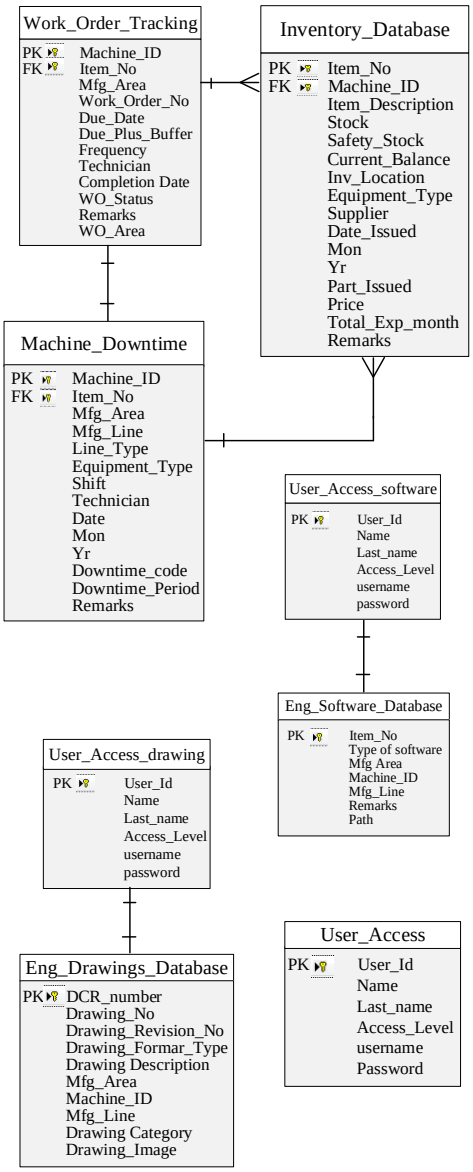


Figure 1
Database ERD

The database is composed of eight (8) tables that are described as follows:

1. Eng_Drawing_Database – This table stores the company drawings that are tied to a particular machine identification number.

2. Eng_Software_Database – This table stores the company software back-ups files that are tied to the machine identification number.
3. Inventory_Database – This table stores the spare parts inventory information. From this table the user has access to the spare parts inventory information data.
4. Machine_Downtime – This table stores the machine downtime information. From this table the user has access to the machine information and downtime information data
5. User_Access – This table stores basic user information such as user identification, first name and last names, access level, username, password. From this table the user has access to the software main screen.
6. User_Access_drawing – This table stores basic user information such as user identification, first name and last names, access level, username, password. From this table the user has access to the drawing module main screen and data.
7. User_Access_software – This table stores basic user information such as user identification, first name and last names, access level, username, password. From this table the user has access to the software module main screen and data.
8. Work_Order_Tracking – This table stores the work order information. From this table the user has access to the work order main screen and data.

Application and Programming

The development of the application was the main phase of the project and the most time consuming. It was the center of the system and ties all components together. The software was written using Java Programming Language Code. In this stage the programing to link the database with the different software modules was finalized.

A virtual server is the repository of the SQL database. This virtual server is accessible for the user through the execution of the software applications with the proper credential granted. The credentials include the level of access such as Administrator, Engineer, etc. Figure 2 shows the project network layout overview design concept.

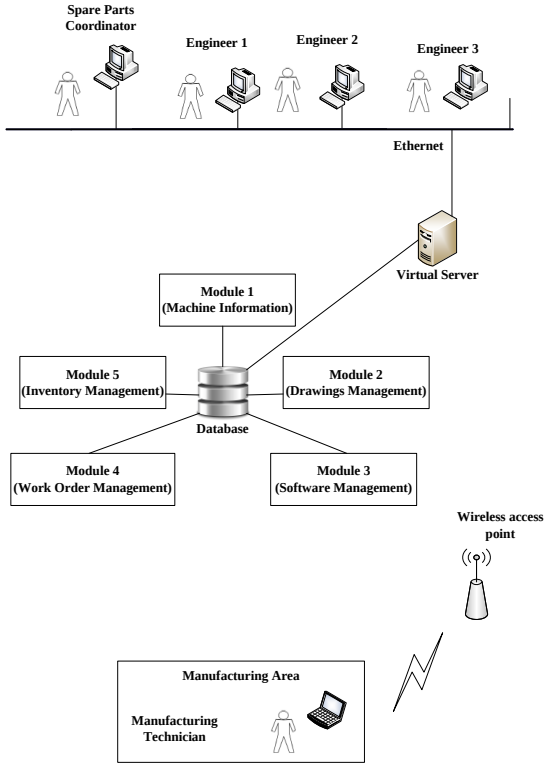


Figure 2
Network Layout Design Concept

In this phase a web page was designed and developed to provide users with access to the executable version of the software. The web page has a new user account registration section, and a login page. The web access link was created using the cloud platform Heroku that supports several programming languages. The database used to acquire user's information for accessing the web page was developed utilizing Mongo Lab which provides safe and accurate database services. With the proper access granted the users download the software executable version and install it in their respective computer stations. Figure 3 below shows a case of use for the software executable file download.

This design of the Graphic User Interface started with the development of a JFrame display login page. The JFrame is a window that has decorations such as a border, a title, and supports

button components that the user may or may not interact [3]. To access the software main screen the user, need to create a new account. In this registration access page, users create their accounts according to the levels of access of administrator, engineer, manager, and technician that the administrator provides. With the proper access granted this login JFrame screen take the user to access the data from the different five software modules. The modules consist of:

- Engineering Drawings
- Software Database
- Machine Information
- Inventory Information
- Work Order Tracking

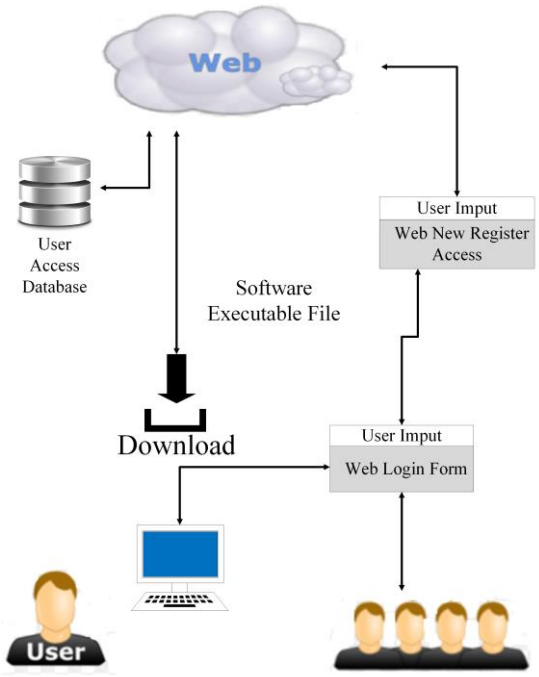


Figure 3
Software Executable Download

Some modules have a login privilege JFrame screen granted by the administrator. The granted username and password are stored and linked to the SQL database access table. Once the admittance is granted, the user can access all modules. The software was developed using NetBeans IDE version 8.2 platform using different tools and method. The java application used in the design of the program was based on the creation and declaration of thirteen (13) java classes, methods and JFrames. The software Classes and JFrames are the following:

1. About_us.java – This java class present the information about the program.
2. Eng_Drawings.java – This JFrame contains all codes related to the drawing management module.
3. Eng_Software.java – This JFrame contains all codes related to the software back-up management.
4. FileSystemModel.java – This java class contains all codes related to the JTree application used in the machine downtime module.
5. Inventory_Database1.java – This JFrame contains all codes for the inventory information.
6. Login_Jframe1.java – This java class contains all codes related to the program main login page.
7. Login_Jframe2.java – This JFrame contains all codes related to the drawing module login page.
8. Login_Jframe3.java – This JFrame contains all codes for the software back-up module login page.
9. Mach_Downtime1.java – This JFrame contains all codes for the machine information module.
10. Main_Menu_Page.java – This java JFrame contains all code related to the main program page.
11. RegisterForm.java – This java JFrame contains all code related to the program registration form.
12. WO_Tracking1.java – This java JFrame contains all code related to the work order module.
13. Javaconnect.java – This java class contains all code related to the database connection.

Hardware

The program was designed to run on any computer window base. The Virtual Server-Side Configuration is a Window Server 2012 R2. An Internet connection is needed for the download of the software executable version. The access to the virtual server is also critical because the SQL database is depository on the server, and the database Uniform Resource Locator (URL) link the software code connection with the database through the server. Without the server the database would not be available. During the development, one laptop was used to test the server-side connection with the computer. The laptop had Microsoft

Windows 10 Operating System and an Intel® Core i7-7820HQ CPU @ 2.90GHz.

RESULTS

EMTS software system consists of six (6) main components:

- Software Access and Registration
- Software Back-up Module
- Drawing Module
- Work Order Module
- Machine Information Module
- Inventory Module

Software Access and Registration

The initial part of the project started with the development of a web page to download the software executable version file. To get admission to the software file the user needs to visit http://engmainttrackingsystem. herokuapp.com/ and enter the valid credentials at the page. The web page design includes a section for user registration.

Once the software executable file is downloaded and installed into their respective user working station, the user must create a new account to get admission to the software applications. The credential data entered is depository in the SQL database user access table. Figure 4 shows the software main login screen.

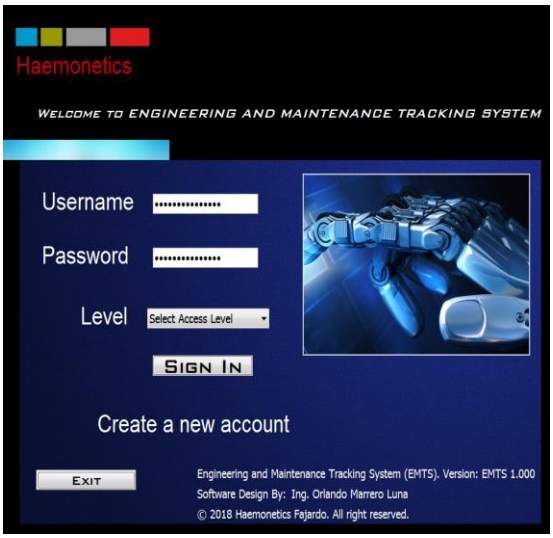


Figure 4
Software Main Login Screen

To create a new account the user must register by pressing the "*create a new account*" on the main login screen. During registration the user must select the level of access. Figure 5 shows the software register form screen.

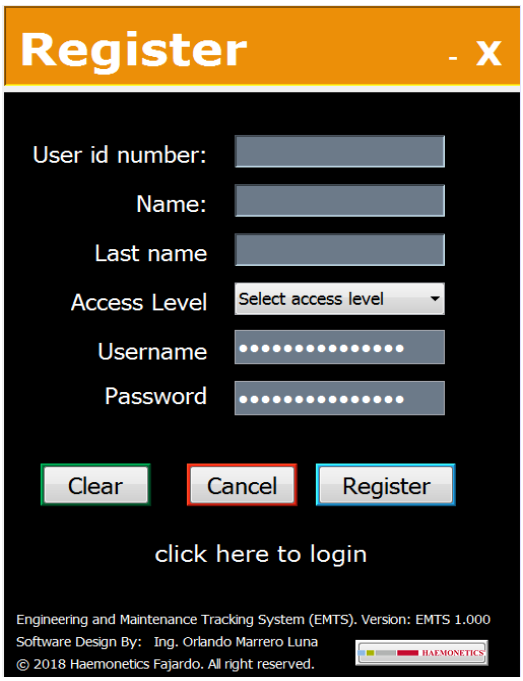


Figure 5
Software Register Form Screen

With the authorize credentials the user would have access to the software main menu screen where the different modules are located. Figure 6 shows the software Main Menu Screen.

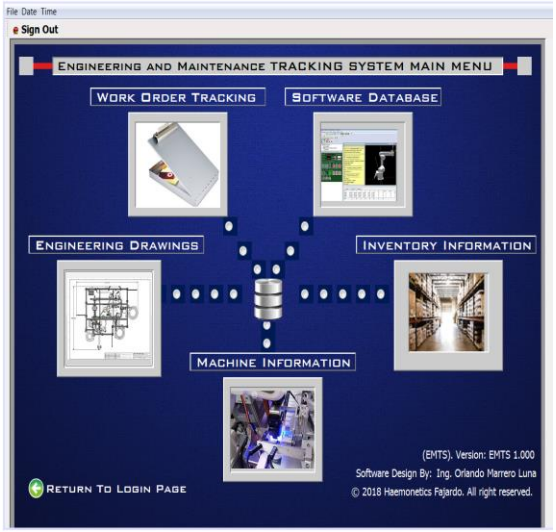


Figure 6
Software Main Menu Screen

The class diagram for the "Software Access and Registration" section is seen in Figure 7. The access starts with the connection to the database through the *java.connect class*. Upon connection to the database the *Login_jframe1 method* is called to access the Login Page. The method that call the registration form is *void RegisterForm* which is called from the *Register Form class*. The Main Menu Screen opens through the *Main_Menu_Page method* after the authorize credentials are granted.

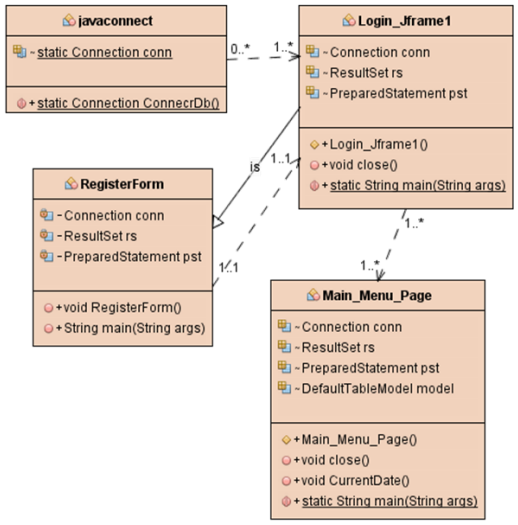
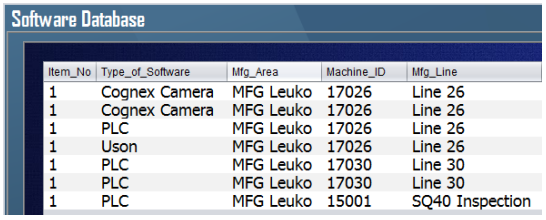


Figure 7
Class Diagram of Software Access and Registration

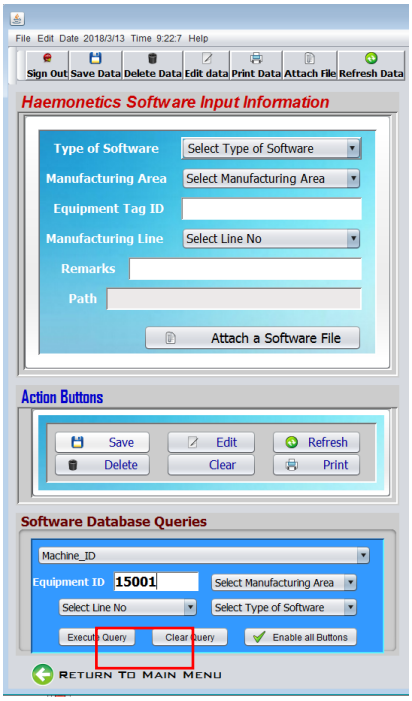
Software Back-up Module

In this screen the user will have access to manage the software backups. From the main menu screen the user must have credentials granted by the administrator to have access to the software database module. This module is capable to add, delete, update, and search data for software back-ups in the database. Also, the module has the capability of execute queries to search for a specific software in the database.

The application was design to have the company machine software's back-up stored in the database. To access the software back-up file, the user must query the search by Machine ID, Line no, Manufacturing Area, and Type of Software. Once the selection is done, the user must press the query button. Figure 8 shows the work-flow example of the software query by Machine ID 15001. Figure 8a shows the user screen database table. In Figure 8b the user select the equipment ID. Figure 8c shows the results of the query on the screen table database. In Figure 8d the user click the selected query on the table to access the software back-up.

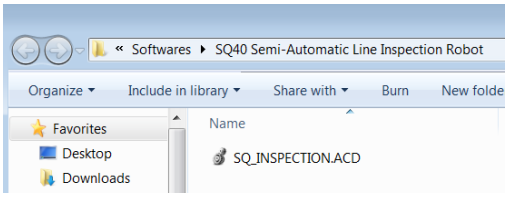


(a) User Screen Database Table



(b) Machine ID Selection

(c) Query Executed by Machine ID 15001



(d) Software Back-up File for Machine ID 15001

Figure 8
Software Back-up Query Work-flow

The class diagram for the Software Module section is seen in Figure 9. The access starts with the connection to the database through the *java.connect class*. Upon connection to the database the user access the *Main_Menu_Page Class* to access the software main screen. The *Login_jframe3 method* is called to access the Login Page for the software database. With the authorize credentials the system access the *Eng_Software class* to get admission to the module fixtures.

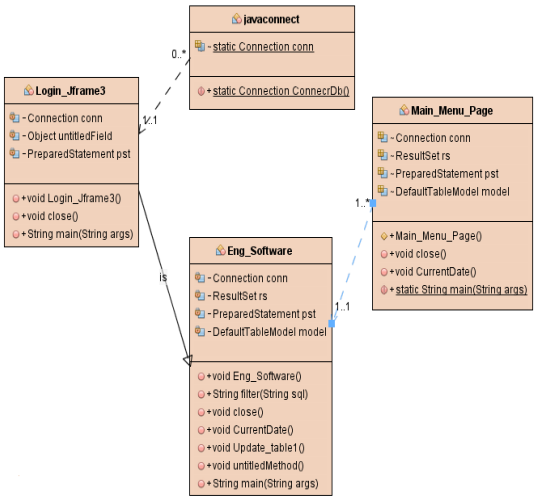
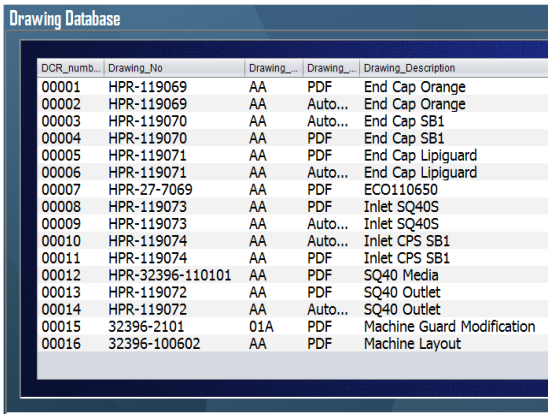


Figure 9
Class Diagram of Software Module

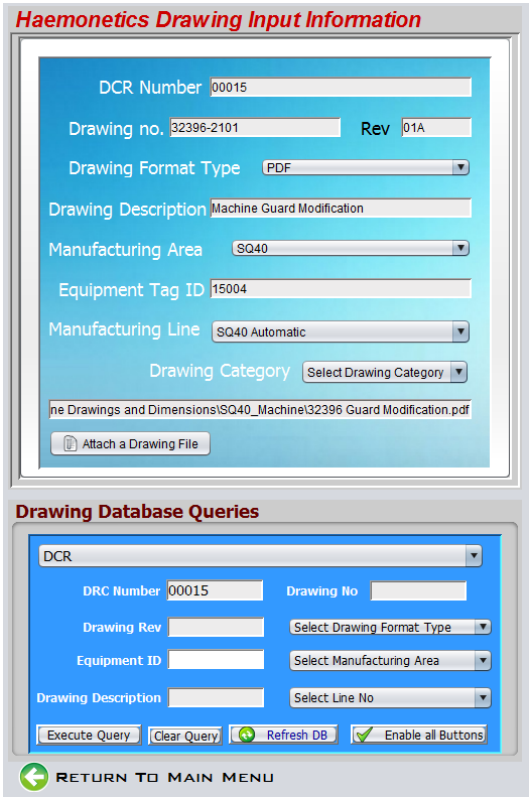
DRAWING MODULE

In this screen the user will have access to manage the drawings files. From the main menu screen the user must have credentials granted by the administrator. The main purpose of the module is to store the company machine drawings in the database. This module can add, delete, update, and execute queries to search for a specific drawing in the database.

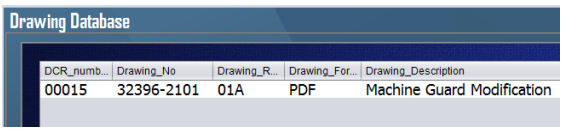
The application was design to have the company machine drawings stored in the database. To access the drawings, the user can execute a query search by Drawing No., Machine ID, DCR Number, among others. Once the selection is done the user must press the query button. Figure 10 shows the work-flow example of the drawing query by DCR No 00015. Figure 10a shows the user screen database table. In Figure 10b the user select the DCR number 00015. Figure 10c shows the results of the query on the screen table database. From there the user must click on the selected query found in the user table to access the machine drawing located in the database. Figure 10d shows the Drawings File stored in the database.



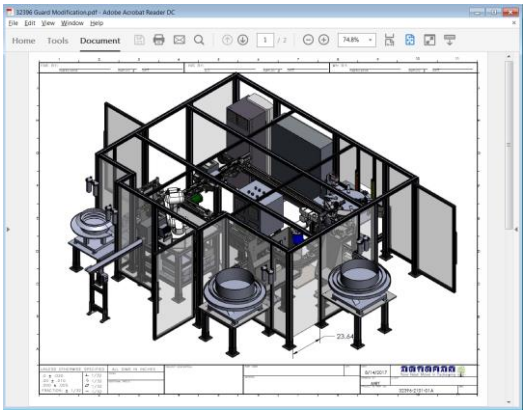
(a) User Screen Database Table



(b) DCR Number Selection



(c) Query Executed by DCR Number 00015



(d) Drawing File for DCR Number 00015

Figure 10
Drawing File Query by DCR Number Work-flow

The class diagram for the Drawing Module section is seen in Figure 11. The access starts with the connection to the database through the *java.connect class*. Upon connection to the database the user access the *Main_Menu_Page Class* to access the software main screen. The *Login_jframe2 method* is called to access the Login Page for the drawing database. With the authorize credentials the system access the *Eng_Drawings class* to access the module fixtures.

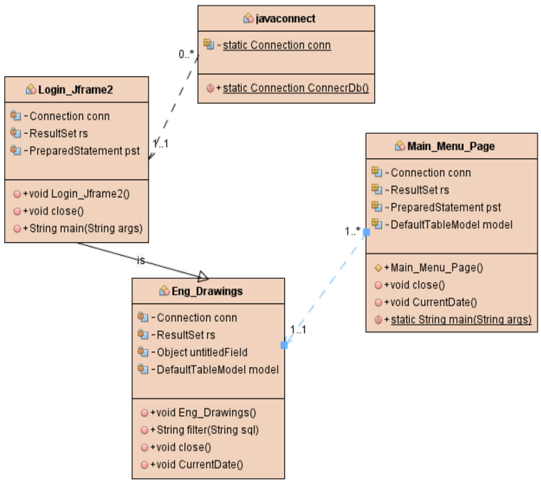
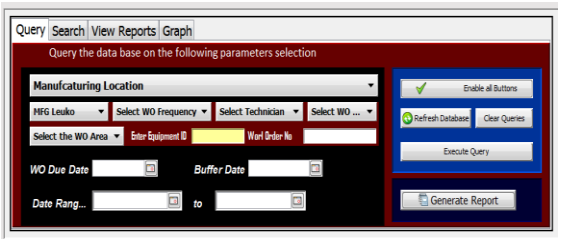


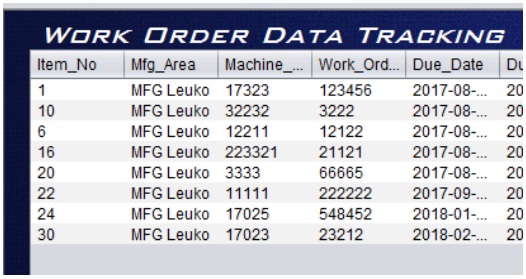
Figure 11
Class Diagram of Drawing Module

Work Order Module

In this module the user will have access to manage the company work order system. Depending on the access granted by the administrator the user can add, delete, update, and execute searching queries of the work order data. Also the module was designed to run status and graph reports of work order information and the results are based on the users query selection. The module offered different alternatives of searching by equipment ID, work order number, due date, completion date, technician who execute the work order, among others. Figure 12 shows the work-flow example of a work order report query by manufacturing location "MFG Leuko". Figure 12a shows the user screen in the software to select the desired query. In this case the selection was "MFG Leuko". In Figure 12b shows the user screen database table after the execution of the query. Figure 12c shows the generated report after the pressing the "Generate Report" push button for the selected query by "MFG Leuko'.



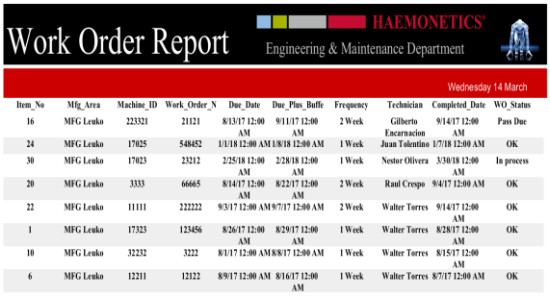
(a) GUI Query Screen for Work Order



WORK ORDER DATA TRACKING

Item_No	Mfg_Area	Machine_...	Work_Ord...	Due_Date	Du
1	MFG Leuko	17323	123456	2017-08-...	20
10	MFG Leuko	32232	3222	2017-08-...	20
6	MFG Leuko	12211	12122	2017-08-...	20
16	MFG Leuko	223321	21121	2017-08-...	20
20	MFG Leuko	3333	66665	2017-08-...	20
22	MFG Leuko	11111	222222	2017-09-...	20
24	MFG Leuko	17025	548452	2018-01-...	20
30	MFG Leuko	17023	23212	2018-02-...	20

(b) User Screen Database Table Query Screen



Work Order Report — HAEMONETICS — Engineering & Maintenance Department

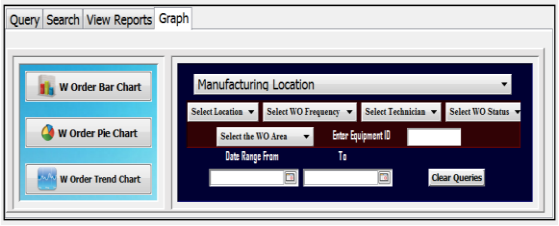
Wednesday 14 March

Item_No	Mfg_Area	Machine_ID	Work_Order_N	Due_Date	Due_Plus_Buffe	Frequency	Technician	Completed_Date	WO_Status
16	MFG Leuko	223321	21121	8/13/17 12:00 AM	9/11/17 12:00 AM	2 Week	Gilberto Encarnacion	9/14/17 12:00 AM	Pass Due
24	MFG Leuko	17025	548452	1/1/18 12:00 AM	1/8/18 12:00 AM	1 Week	Juan Tolentino	1/7/18 12:00 AM	OK
30	MFG Leuko	17023	23212	2/25/18 12:00 AM	2/28/18 12:00 AM	1 Week	Nestor Olivera	3/30/18 12:00 AM	In process
20	MFG Leuko	3333	66665	8/14/17 12:00 AM	8/22/17 12:00 AM	2 Week	Raul Crespo	9/4/17 12:00 AM	OK
22	MFG Leuko	11111	222222	9/3/17 12:00 AM	9/7/17 12:00 AM	2 Week	Walter Torres	9/14/17 12:00 AM	OK
1	MFG Leuko	17323	123456	8/26/17 12:00 AM	8/29/17 12:00 AM	1 Week	Walter Torres	8/28/17 12:00 AM	OK
10	MFG Leuko	32232	3222	8/1/17 12:00 AM	8/8/17 12:00 AM	1 Week	Walter Torres	8/15/17 12:00 AM	OK
6	MFG Leuko	12211	12122	8/9/17 12:00 AM	8/16/17 12:00 AM	1 Week	Walter Torres	8/7/17 12:00 AM	OK

(c) Generated Report by Manufacturing Location

Figure 12
Work Order Query by Manufacturing Location Work-flow

From this module the user can generate accurate chart reports. Figure 13 shows the work-flow example of a work order Bar Chart Report query by manufacturing location. Figure 13a shows the user screen in the software to select the desired Chart Query Report. To open the report the user has to select the query selection and press the desire type of chart report (Pie or Bar). In this case the chart report is showing the quantity of work orders done by Manufacturing Locations. Figure 13b shows the generated Bar Chart Report after execution of the query.



(a) GUI Query Screen for Chart Reports

(b) Bar Chart Report by Manufacturing Location

Figure 13
Chart Report Query by Manufacturing Location Work-flow

The class diagram for the Work Order Module section is seen in Figure 14. The access starts with the connection to the database through the *java.connect class*. Upon connection to the database the user access the *Main_Menu_Page Class* to access the software main screen. The *WO_tracking1 method* is called to access the *WO_Tracking1 class* that contains all module fixtures. The *About_us class* is called through the method *About_us* and contains all the information related to the sofware rights. This class is called from each of the software modules.

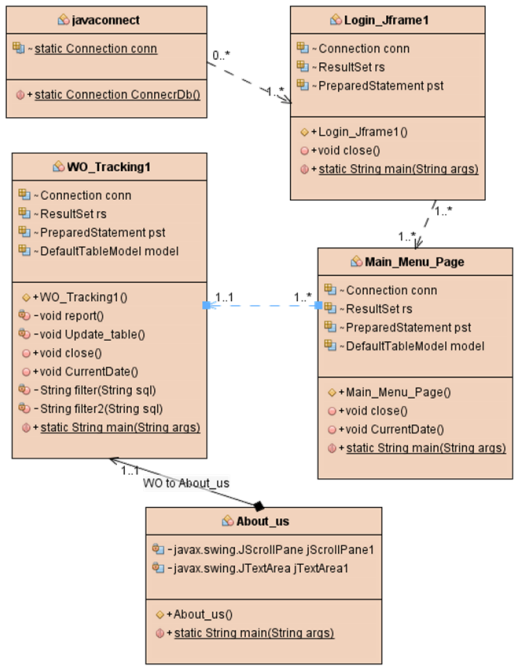


Figure 14
Class Diagram of Work Order Module

Machine Information and Inventory Module

The Machine Information and Inventory Modules manage the company machine downtime and spare parts information respectively. Like the other modules previously discussed, the users can add, delete, update, execute machine information and components inventory queries. In addition, in both modules the users can generate written and chart trending reports. The following are the class diagrams for both modules:

The class diagram for the Machine and Inventory Information Modules are shown in Figure 15. The access starts with the connection to the database through the *java.connect class*. Upon connection to the database the user access the *Main_Menu_Page Class* to access the software main screen. To access the Machine Information Module the *Mach_Downtime1 method* is called to access the *Mach_Downtime1 class.* The *FileSystemModel class* contains the FileSystemModel method that is called when the

user request the Jtree option for technical file management.

In the case of the Inventory Information Module the class that contains all information about this module is the *Inventory_Database1 Class*. The class is called from the *Main_Menu_Page Class* through the *Inventory_Database1 method*. The *About_us class* is called through the method About_us and contains all the information related to the sofware rights. This class is called from each of the software modules.

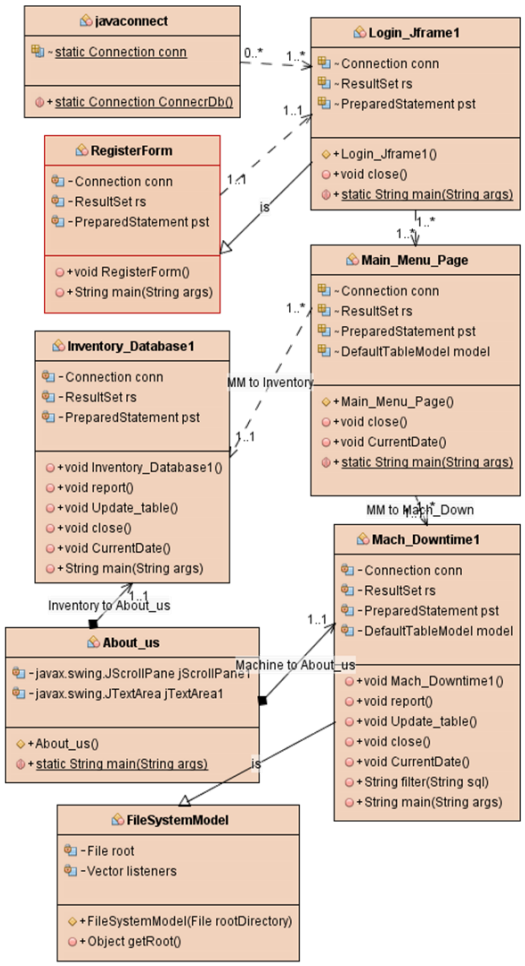


Figure 15
Machine and Inventory Class Diagram

FUTURE WORK

The entire project was meant to be use as prototype concept for the company data collection management. Certainly, the implementation of this project is the basis for future expansions as far as data collection at Haemonetics Corporation. In the future it is expected to integrate the communication and control systems of the machines in the manufacturing area to collect real data in real time automatically. Also, the quality, as well as the manufacturing systems, will be synchronized together to monitor and control the statistical processes of the behavior of the machines and personnel by means of the Statistic Process Control (SPC) tool. To this end, the developed software will be revised to serve as a tool to manipulate and evaluate the data automatically collected from the process, and turn it into useful reports that can provide greater visibility to the company's management staff in the decision-making process.

CONCLUSION

The main goal of this project was to develop and design a software with the ability to integrate and centralize the data between the different critical and technical areas of the medical devices company Haemonetics. Based on the results of the design, and after observing the software's capability to re-organize and collect data, it is concluded that the main purpose of the design was achieved. With the implementation of this software the Engineering Department within the Haemonetics company, is one more capable of dealing with the daily problem of data collection since now with the new software the department has more flexibility to keep the processes in control. Based on the results after the implementation and development of this software, the company technical resources and engineers, as well as the manager staff of the company, are very confident that the new software will help them to have a more robust process and better control of the technical activities within their Engineering Department.

The Engineering and Maintenance Tracking System is a prototype design that certainly has room for improvements, and there is still much to be done with the development of the design, but at least it provides the basis for future works at the Haemonetics Corporation.

REFERENCES

[1] C. Boulton (2017, Jun. 26). *The big-time benefits of a centralized data analytics team* [Online]. Available: https://www.cio.com/article/3203364/analytics/the-big-time-benefits-of-a-centralized-data-analytics-team.html. [Accessed: Dec. 14, 2017].

[2] A. C. Uzialko. (2017, Nov. 28). "The Internet of Things is changing the Entire Production and Supply Chain," in *Businessnewsdaily.com, B2B Staff Writer* [Online]. Available: https://www.businessnewsdaily.com/10395-iot-manufacturing-production-supply-chains.html.

[3] Oracle. (n. d.). *How to Make Frames (Main Windows)* [Online]. Available: https://docs.oracle.com/javase/tutorial/components/frame.html.