Photolithography Management Reports and Engineering Tools Web Application

Atmel Corporation

By

Anthony Melorango
Suu (Mike) Nguyen

June 22, 2007

Executive Summary

The goal of this project was to create a web-based application to help improve the efficiency of Atmel's photolithography department. We accomplished this goal by creating a program which retrieves information from a database. This database is created from tests run within the department to check if computer chips created meet the requirements so they can work properly. The data retrieved is displayed in a two different ways. One way the data is displayed is in a graphical manner, which allows someone that does not know the workings behind the measuring tools to see if any trends are occurring with errors created during the testing process. The other way is showing the raw data from the database, allowing engineers to determine what is causing the errors and fix them.

This goal is accomplished using the Microsoft Visual Web Developer 2005 Express Edition, Microsoft SQL Server 2005 Express Edition and ChartFX for .NET Trial Version. Atmel has purchased the full version of ChartFX already purchased, so they will be able to use this if they need to make any changes to the graphs that are displayed.

After completing the application, we are happy with the results. We feel that we have completed all the specifications that were given us by the client at Atmel. We also learned about how to work with the .NET framework created by Microsoft.

Table of Contents

Abstract

Atmel is a large semiconductor manufacturer with a worldwide distribution of products. Engineers and management are having difficulty diagnosing errors with six machines that create semiconductors for the photolithography group at the company. Prior to this project, engineers were simply reading data from a database and making realizations on possible issues with no set data analysis to help the determine what was causing a problem. With this method, there were many issues with human error in reading the data since trends were hard to determine when looking at massive sets of numbers. Our team was asked to create a graphical user interface for the data stored in the database so that engineers and management can make a more informed decision on where in the photolithography process improvements can be made.

Our team has been asked to create the graphical user interface in Visual Basic.NET and have it connected to a sample data set stored in a Microsoft SQL server. The main goal for this project is to allow management and engineers to see the data from the database without having to process it themselves. With this they can easily see trends that occur when certain recipes and machines are used and will help to limit the human element of data processing.

Introduction to Project

The project that we have taken on for the 2007 MACS Field Session is to help Atmel Corporation improve their efficiency in manufacturing semiconductors. The computer components they produce can be separated into many different categories, such as microcontrollers, radio frequency (RF) components, nonvolatile memory and sensors. These devices are used in a variety of products, varying from components in toys at fast food restaurants to parts in key-less entry controllers used for automobiles. Atmel has centers of operation in many locations around the world. The largest fabrication plant is in Colorado Springs. Here, the chips are sent through many different steps in which the specific design of the chip, called a recipe, is burned onto the chip itself. Between each of these steps the dimensions are checked against the specifications for the recipe's critical dimensions (CDs) with a scanning electron microscope (SEM). The information from these tests is logged locally on the SEM then stored into a database. This record contains hundreds of thousands of entries, and the data must be interpreted by the engineers at Atmel to see what is causing any errors and how they can be fixed. This is where our project comes in.

Requirements and Specifications

Requirements

In order to know what the data output from the CD-SEM represents, the managers at Atmel needed to carefully review the information in the database to find any trends or problems that need to be fixed. Our program will get the information from these resources and display it in an easy-to-read, user-friendly output. The output is broken into two different modes: Management Reports and Engineering Tools. The two different modes are set for the different type of users. Management Reports are easy to understand graphs that are created to show specific trends. These reports are set for management to quickly determine if the tools are working according to specification. Engineering Tools are reports and tables that engineers can view to determine issues with machines and figure out if any fixes can be made.

Specifications

The Management Reports will show the data contained in the database depending on which time frame is specified by the user. This data will be output as three sets of charts. The first set will be two bar graphs that show wafers (chips) and recipes run in the given time period filtered by which shift were running the SEMs at the time. The second set of graphs will be bar graphs that display average run time and average load time per wafer, one separated by tool and the second by shift. The third set of data will be presented as a pie graph based on the percent of errors that occurred in the given time frame.

The Engineering Tools will have more of the raw information contained in the database. The main page of this section contains two items. The first segment determines what filters should be placed on the data to be shown in the summary section. There will be three filters to choose from: a time frame filter, a directory filter, and a tool filter. The time frame will be chosen by the user and when set will determine the span of data that will be displayed. The second filter will modify the displayed data to match which machine the user wants to see. The directory filter will allow the user to choose to see production, engineering or qualification data, based on how it is classified in the database. The second section of the main page is where the data summary is presented. The first part of this section is general statistics based on the options chosen by the user previously. These statistics are total recipes, wafers, errors and aborts contained in the data table below this information. The second part is the main piece of the page where all the records from the database are shown according to what was needed.

When a row of the data table is double-clicked, the Engineering Drill Down page for the selected recipe will pop up. The page will contain two sections. The first is a different table from the main Engineering page and it shows all the data contained in the database based on the recipe that was selected. Under this, in the second section, are four bar graphs which allow the user to visualize the data contained in the table. The first graph shows the number of errors split into each run in the previously selected time frame. The next graph contains information based on how many errors occurred when the recipe was run on each SEM. These two graph's bars are split based on what type of error occurred. The third graph shows the duration of each run and the fourth displays the average duration based on which tool was used for the test. These two diagrams have their bars broken into which step of the test was being run at the given time.

Deliverables

The final deliverable for this project is the completed application based on the requirements and specifications listed previously in this report. After talking with our client, they would like to be able to use this program as soon as we turn it in to them. Their senior developer will be working with us to help ensure that this goal can be met.

Development Environment

Since we will be working with our client's main developer, we asked him which language he would like the program to be written in so he can maintain it after we are finished. His choice was Visual Basic, since this is the language that he knows best. Since this will also be a web application, we will need to use the functionality found in the Microsoft Visual Web Developer 2005 Express Edition. This program uses functionality based in ASP pages, with underlying programming in Visual Basic. The database that the information is stored in is a Microsoft SQL database, which is very easy to communicate with Visual Basic programming. This will fulfill one of the requirements set at the beginning of the project of using the .NET framework.

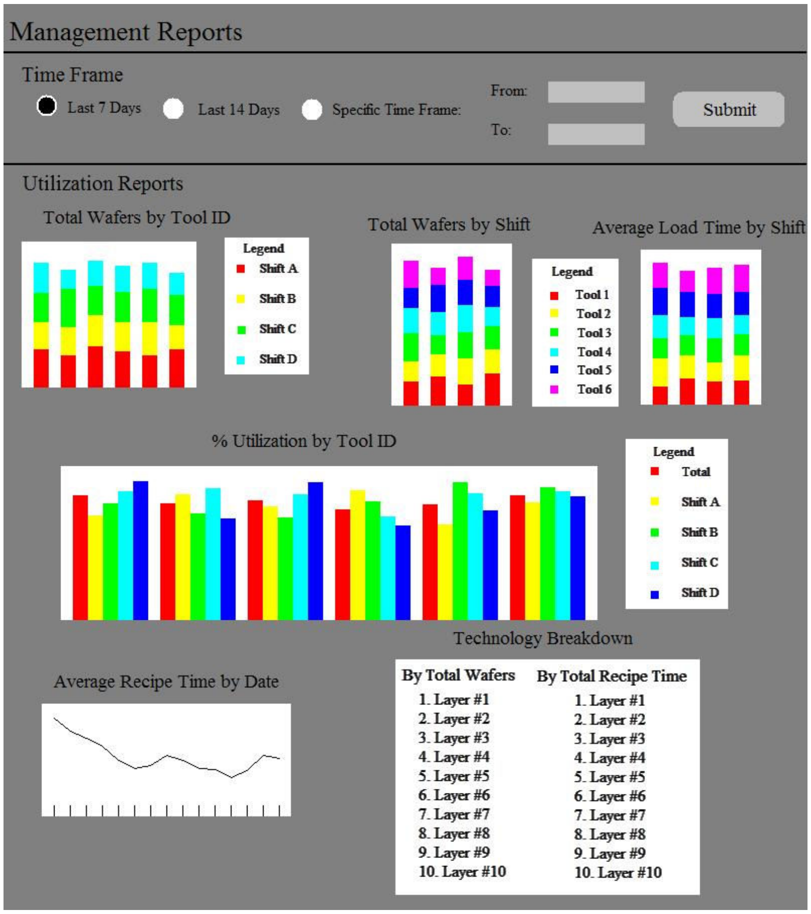
Design

Approach

The approach that our group is taking to attacking the task of completing the application assigned to us is to divide and conquer. After discussing what is expected from us with our clients, we have noticed that we have two main sections in which we can easily divide the application into. The two sections will be the Management Reports and the Engineering Tools. Both of these sections are main pages of the application that will be set on different tabs of the main page. Splitting the work into two main areas will keep us from depending code on each other. This works well since the Express version of Visual Studio does not allow addition of source control tools.

Program Design

The Management Reports page is split into three sections. The first two sections can be seen in Figure 1. The first section is the time frame section, which allows the user to control what information will be summarized in all the graphs on the page. The second section is the utilization reports section, which contains graphs that management can use to determine efficiency of tools and personnel. The utilization graph section will contain four bar charts, a line graph and summary technology/layer data. The two of the four bar charts will graph the total wafers checked during the time frame set in the time frame section. One graph shows the wafers run by Tool ID with each bar split into colors depending on work shifts. The second graph is the total wafers run by shift with the bars split into colors depending on Tool ID. The third graph will be similar to the second, where each bar is of different shifts and the bar is split into colors depending on Tool ID, but the graph will show the average load time. The fourth bar graph will have multiple bars per Tool ID, where the first will graph the total recipe time divided by the total time and the following bars will be total recipe time for each shift divided by the total time per shift. The line graph will be a graph of the average duration with the x-axis being a time increment, depending on the time frame set. The final section will be a summary of the top ten technologies by total wafers and the second summary will be the top ten technologies by duration of the whole testing process.



By Total Wafers	By Total Recipe Time
1. Layer #1	1. Layer #1
2. Layer #2	2. Layer #2
3. Layer #3	3. Layer #3
4. Layer #4	4. Layer #4
5. Layer #5	5. Layer #5
6. Layer #6	6. Layer #6
7. Layer #7	7. Layer #7
8. Layer #8	8. Layer #8
9. Layer #9	9. Layer #9
10. Layer #10	10. Layer #10

Figure 1

The third section of the management report will be a section on error reporting. An example of what this will look like can be seen in Figure 2. The error reports will contain information that management can look at to determine what errors were occurring on which tool or shift. This section does not contain enough information for management to determine the reason the errors were occurring because this information will be contained in the Engineering Tools page of our program. The error reporting will have four graphs, three of which will be bar graphs and the last will be a pie chart. The first two bar charts will be the total errors, both with each bar split by different colors depending on error type. The first graph will have x-axis of Tool ID and the second will be by shift. The final bar chart will have bars in split into several colors, each corresponding to the amount of wafers that were tested for that time period that did or did not contain errors. The x-axis will be the time increment, depending on the time frame set. The final graph will be a pie chart split by wafers with and without errors for the time frame.

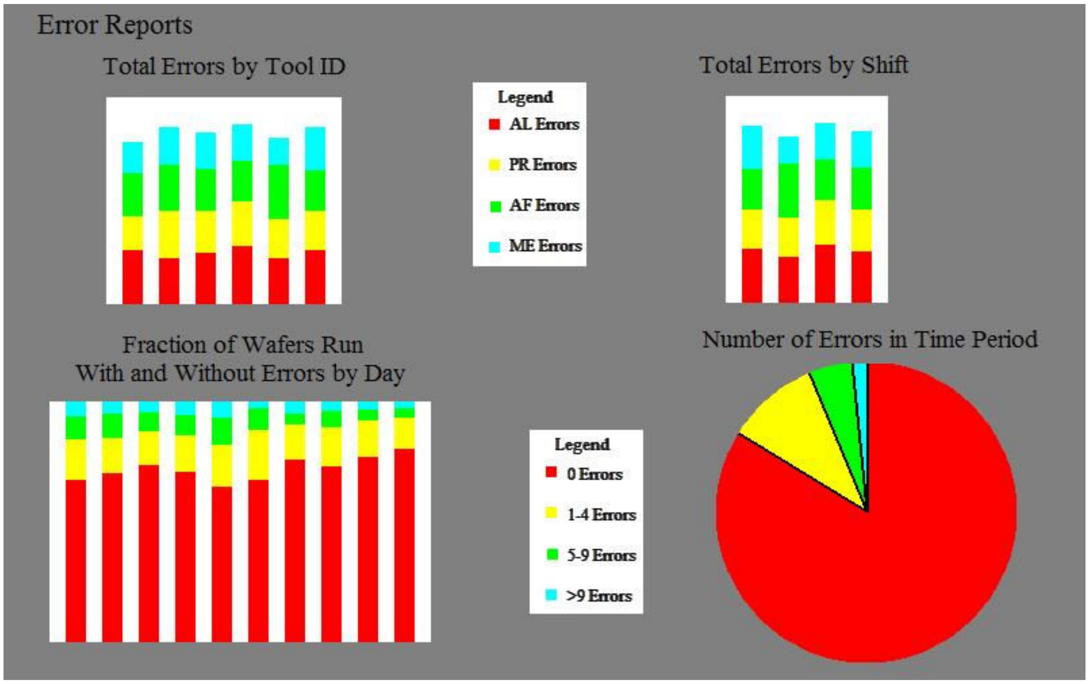
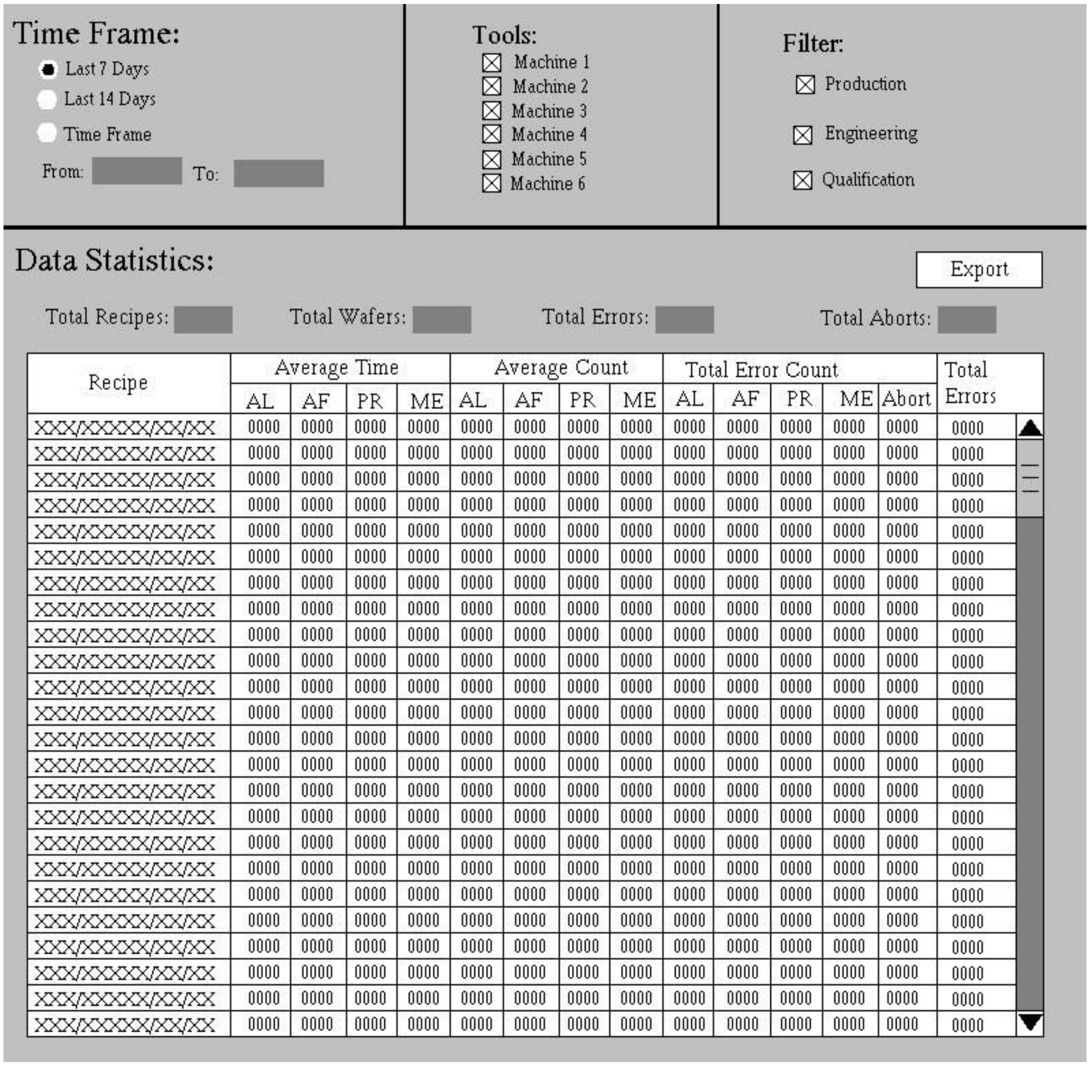


Figure 2

The Engineering Reports page will have two distinct parts. The first is the main screen that will show up when the Engineering section is selected to be displayed. A screen shot of what this will look like can be found in Figure 3. The top section of this page will have options to configure data to be displayed in the table below. Above the table are summaries of all the data contained from the options chosen above. Also on this page is the button labeled "Export" which will take the data contained in the table and exports it to a comma separated value file, or .csv for short. This will let the engineers look at the data they received for their queries without needing to remember the precise options they selected to create the table.



Time Frame:

● Last 7 Days
○ Last 14 Days
○ Time Frame
From: To:

Tools:

☒ Machine 1
☒ Machine 2
☒ Machine 3
☒ Machine 4
☒ Machine 5
☒ Machine 6

Filter:

☒ Production
☒ Engineering
☒ Qualification

Data Statistics:

Total Recipes: Total Wafers: Total Errors: Total Aborts:

Recipe	Average Time				Average Count				Total Error Count					Total Errors
	AL	AF	PR	ME	AL	AF	PR	ME	AL	AF	PR	ME	Abort	
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
XXX/XXXXX/XX/XX	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

Figure 3

The second part of the Engineering Reports is the Drill Down section. This page comes up when the user double-clicks a specific recipe from the table in the initial page. An example of this screen is shown in Figure 4. The top part of this page contains the raw data directly from the database. It changes based on the recipe that was chosen to be examined further and the time period in which the data table from the previous page was based on. The "Export" button here has the same functionality as before, allowing this data to be seen without having to use our program specifically. Under this information are four graphs which will help the engineer to see how the data can be interpreted. The first two graphs break the total errors that occur when the recipe is tested into the four different types of errors. The first graph lists the bars broken into which run contained in the time frame selected previously. The second graph has the error broken into the Tool ID that was used when the test was run. The third and fourth graphs have their bars broken into which category of time that was spent on each step of the test's process. The third graph shows the duration of the test broken apart based on which run. The forth graph shows the average duration of the total test type based on which tool was used.

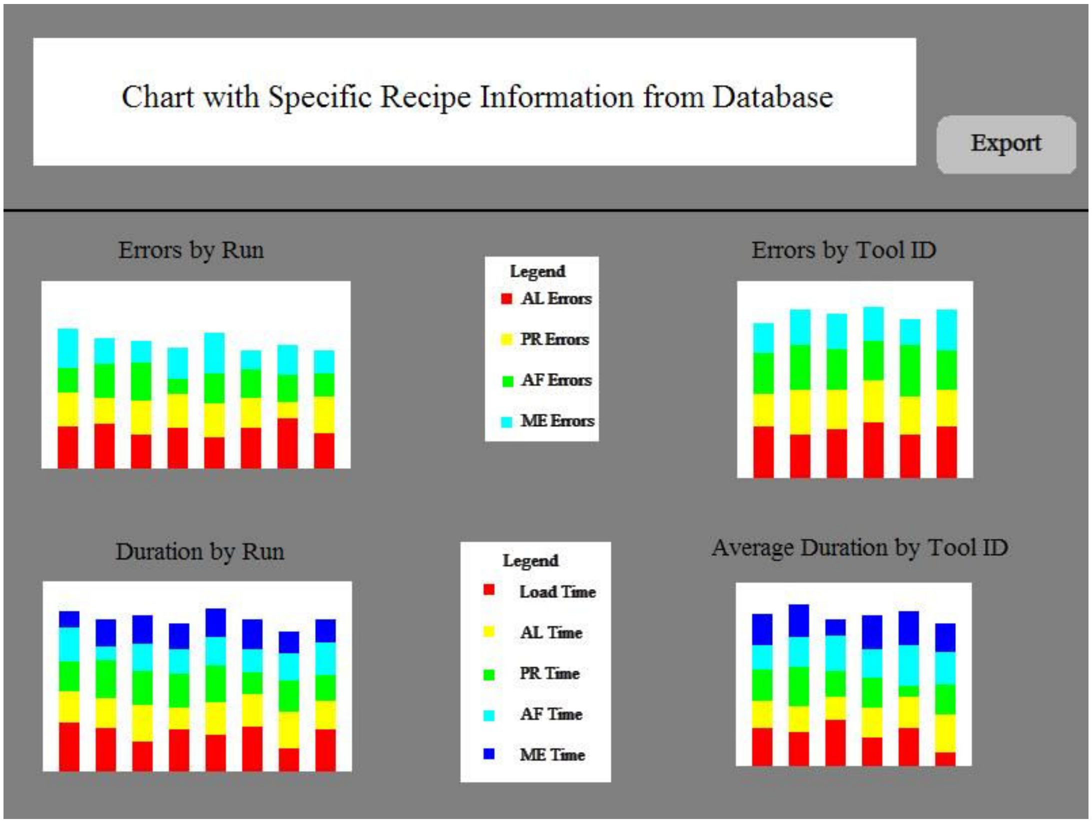


Figure 4

Schedule

After looking at how the program should be designed and how we are going to split the work up, we have come up with a timeline of how our project will be completed. The first week of our time was spent meeting with the client to find out what the requirements were needed for our project. Also during this time, we began researching what new resources we would need to use to implement our design and started to learn them. The second week was spent completing the design on which our project will be based on. We met with the client via teleconference to ensure that this design was what they were looking for from us. Also during this time, we received a sample database from the client to be used during our implementation phase. At the end of this week, we presented to the class our design and wrote out our plan in this paper.

The next two weeks are when we began implementing of our project. Since we split the project into the two parts, we created each of our sections separately and then met together near the beginning of the fourth week to combine the parts to ensure they worked properly together. At the end of the fourth week, we delivered a nearly completed version of our program to the client to be looked over for any problems or changes that they would like to be made. In our meeting at the beginning of the fifth week, we communicated with the client to get these changes. We completed this part of the project, and turned in a fully functional version of the product to the client. Then the sixth week was devoted to writing and revising our final presentation and report. This time line showed how this project was finished within the given time for this session.

Implementation Details

During the implementation stage of our project, we encountered more difficulties than originally estimated. The Engineering Tools section of application went fairly well, since the functionality of displaying tables is available easily through the SQL Database and Visual Web Developer Studio development environment. Also, Mike was in charge of these utilities and he has been working with SQL statements for nearly three years, so he was able to write queries easily for the correct information.

The Management Reports page did not go as smoothly when Anthony attempted to implement the needed functionality. In order to display the results from the database in a graphical way, we were using ChartFX. However, after attempting to follow the tutorials provided for this program, we were unable to run the application without errors. When we researched the error we were finding, we found out that there was an additional step that needed to be taken to enable the functionality needed to display the graphs. We needed to create a different server from what was initialized when the program was compiled in the development environment. This required the use of Windows XP Professional, which Anthony did not have installed on his computer. Because of this, Mike took over the development of this page and was able to find out the way to enable the use of graphs. After these details were worked out, the actual implementation went fairly smoothly.

Results and Conclusion

After overcoming the slight setbacks listed above, we were able to complete the applications according to the client's specifications. The Management Reports page has all the graphs that were talked about with the client. Some of them required more work than others, but we were able to complete them correctly. The Engineering Tools page has both the main screen and the Drill Down functionality that was needed to meet the requests of the client for this application. Since we were using a small sample database provided by the client, when we deliver the product, the main computer programmer there can simply attach their main database to the application to allow for full functionality with all their data that has been collected. Since we followed the client in what language should be used to create the program, this step will not be very difficult.