

MIS 3720: BUSINESS DATA MANAGEMENT

Project

Group 3

Mohammad Shahbaz Soofi, Georgi Kardzhaliyski, Muyeedul Hoque

10/5/2012

GMS Database Consulting, Inc.

GMS Database Consulting, Inc., a firm located in MA, was hired to create and design a database for WPI's helpdesk. The WPI Helpdesk didn't record all data gathered from their tickets in a centralized database system before they hired our assistance. In fact, that is one of the reasons why they didn't have meaningful data on queues, tickets per month, backlog times, when the tickets entered the system, how long they were present till they were acted upon and many such factors.

For our project we have decided to work with the issues concerning WPI's Technology Helpdesk. Currently, the Helpdesk records all information either manually or on an informal online logging system. WPI's helpdesk wants to gather and record all data in a centralized system. They want us to create a database for recording all their ticketing issues. This will help them to generate more meaningful reports and make informed managerial decisions. Currently, we are aware that the Technology Helpdesk has three major ways in which tickets are incurred – Phone Calls, Emails, and Walk-Ins. The employees sitting at the front desk usually deal with both Walk-Ins and Emails (resolving Email tickets when they are relatively free). Then there is a Call Center located behind the front desk which mainly deals with Phone Calls but also with emails if they aren't fully occupied. Having a database system would keep their data safe, and

will help to connect all pieces of ticketing information in a centralized system from which key analyses can be performed and important managerial decisions can be made.

Our database will comprise of a few tables in order to record all the information necessary. Our current design includes five tables:

- **tblCustomers** – The Customers table includes basic information about the customer for whom the ticket was logged: **CustID** (primary key – unique for every customer), CustName, CustPhone, CustEmail, DeptName, CustStaffOrStudent (a logical field that tells us whether the customer is a staff or student in WPI), and CustAddress.

The table only consists of the list of customers for whom one or more tickets were logged. Only one row will be allocated to each customer even if the customer generated more than one ticket. The purpose of this table is to record a list of all the customers that have incurred tickets in the system.

- **tblTickets** – The Tickets table consists of the primary information about all the tickets that are logged: **TicketID** (primary key), **CustID** (foreign key that links to the 'Customers' table with a 1:N relationship), TimeLogged (the time at which the ticket was logged in the system), TimeResolved (the time at which the issue was resolved completely), TicketImpactedArea (the problematic area in the machine), and TicketDescription (the description of the problem).

The purpose of this table is to record all the tickets that have been registered in the system, and to keep track of the times at which they were initially registered and finally resolved.

- **tblSourceDetails** – The SourceDetails table will comprise of the detailed information for each source of the ticket that the Helpdesk generates. There are times when the same

TicketID may come back to the system as a different source, e.g. a phone call is escalated to a walk-in. That is why this table keeps track of the details of all tickets based on their sources: **SourceType and TicketID** (concatenated primary key with a foreign key that links to the the 'Tickets' table with a 1:N relationship), QueueTime (the duration of time for which the entity, i.e. the customer, was waiting in queue, e.g. in line during walk-ins or on hold during a phone call, before the ticket was received or registered), TimeEntered (the time at which the ticket entered into the system, e.g. an email sent at 2:00 am; this may be the same as TimeLogged in the 'Tickets' table, e.g. in the case of walk-ins), TotalTimeLapsed (the duration of time for which the ticket was present in the technology system, but was not logged in or acted upon, especially in the cases of overnight emails and phone-calls), and OffHours (a logical field which says if the ticket entered the system off-hours or not – if the checkbox is checked, it means the ticket was received off-hours). The purpose of this table is to have data on all the sources for each ticket and to track the duration for which the ticket was in the system, or the entity was waiting for service, before an employee started working on the ticket. Such data will help to analyze efficiency in the system.

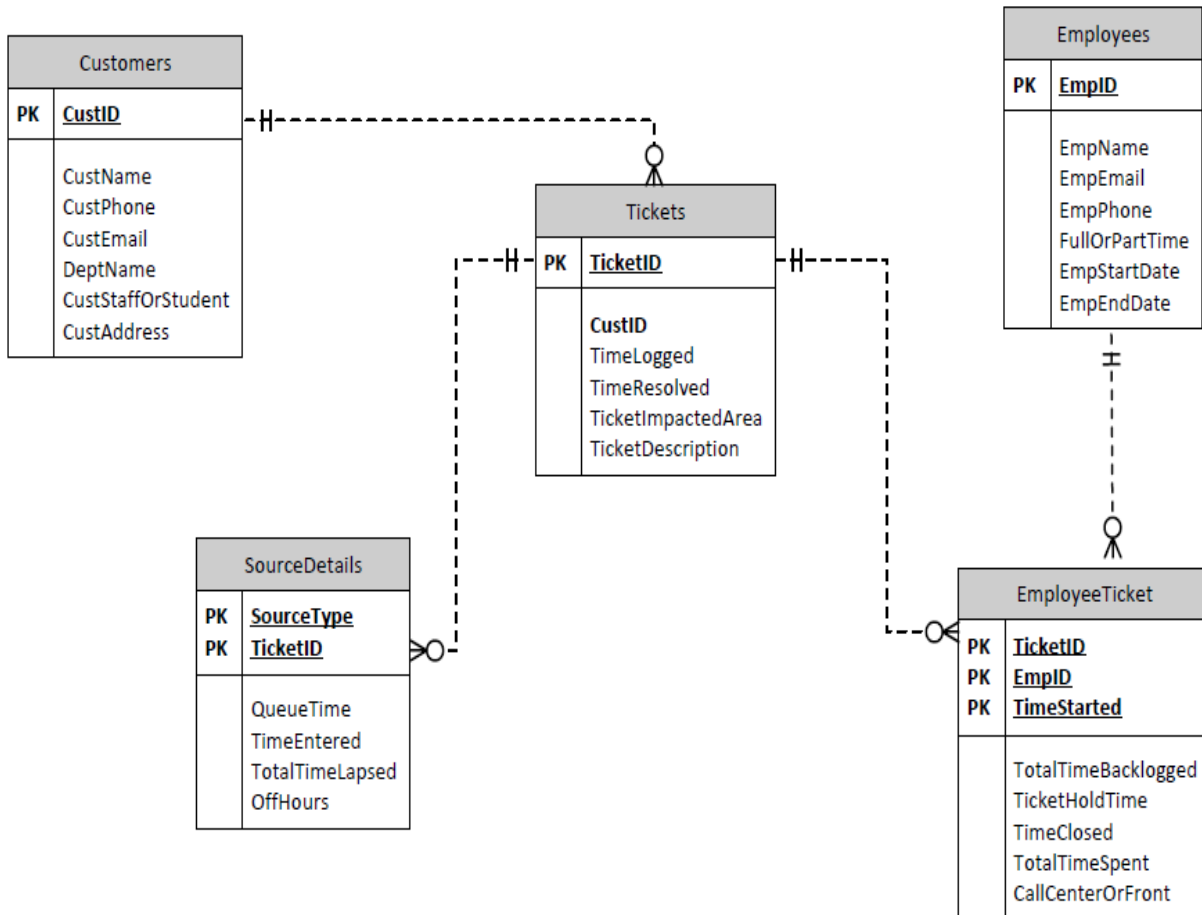
- **tblEmployees** – The Employees table consists of the list of all the employees working at the Helpdesk and their information: **EmpID** (primary key unique to every employee), EmpName, EmpEmail, EmpPhone, FullOrPartTime (a logical field which records whether the employee is a full-time or a part-time employee), EmpStartDate (when the employee started working), EmpEndDate (when the employee no longer worked at the Helpdesk).

The purpose of this table is to keep track of all the employees who have previously worked, or currently work at the helpdesk.

- **tblEmployeeTicket** – The EmployeeTicket table consists of the information of all the employees who worked on a ticket for a particular duration of time: **TicketID and EmpID and TimeStarted** (concatenated primary key and foreign keys which link both the ‘Tickets’ and ‘Employees’ tables with a 1:N relationship; TimeStarted is the time at which the employee started working on the ticket), TotalTimeBacklogged (the duration of time for which the ticket has not been worked on even though it has been logged in the database system, e.g. an email ticket was logged but wasn’t worked on for a few hours due to heavy traffic), TicketHoldTime (the duration of time for which a particular employee working on a ticket takes a break, e.g. due to a break in shift, and no other employee started working on this ticket), TimeClosed (the time at which the employee finished working on the ticket either because it was resolved or passed on to another employee), TotalTimeSpent (the total time the employee spent working on that ticket, not considering the hold times), CallCenterOrFront (whether the employee worked in the call center or the front desk while working on the ticket – if the checkbox is checked it means the ticket was operated from the Call Center).

The purpose of this table is to record all information about each employee working on each ticket for a particular duration of time. The table can also track information about how long each employee spent working on a ticket, and how many employees worked on that particular ticket. This will also help for data analysis on possible utilization rates, efficiency and total time spent per ticket calculations.

ER Diagram



List of Queries that the Helpdesk wanted Database Consulting to generate:

Query A

The project manager, Cate, wants to find out a list of employees and how many tickets they have worked on. She has requested to find out the number of employees who worked on each ticket.

Query B

A Business Analyst was hired to work with the employees related to Microsoft Outlook 2010. She needs to generate a query based on the employees who have worked on this issue to get a better idea of which people know how to work with the software.

Query C

The Helpdesk Support works from 8 am - 5 pm on Mondays. However, they receive calls and e-mails during the weekend. These tickets are not assigned a ticket number and are not entered into the system until hours of operation resume. They would like to get the number of tickets received off-hours for the last 6 months, in regard to the ticket issue.

Query D

Because of the high turnover in the Helpdesk, employees can be replaced and new hires would take their assigned ticket. Lisa, the project manager, would like to ensure there is a smooth transition and all customer tickets get resolved by the rest of the available Helpdesk agents. She would like to generate a query about all terminated employees and the tickets they were working on.

Query E

The IT Training division is planning on providing training on using Microsoft SharePoint for team sites next month. The manager would like to find out by department the number of issues on a monthly basis. She would need to generate a query with the number of tickets per department name related to Microsoft SharePoint.

Query F

The Helpdesk Analyst would like to ensure customers receive on-time resolution of their problems in two days. He would like to look at all the tickets generated for the past three months by impacted area, in regard to their logged and resolved times.

Query G

The Helpdesk manager would like to know how long entities wait before registering their issue, and how long the issue stays on the system before being logged. Therefore, he wants a query listing the average queue time and average total time lapsed in the system for each source type in the last six months.

Query H

The Helpdesk manager wants to know the times tickets spent in backlog. So we need to run a query listing the average times for all tickets in backlog for the last 1 year, and whether the Call Center or the Front Desk worked on it. We also need to group it by Ticket Source Type.

Query I

The Helpdesk manager wants to know how many tickets a part-time worker (student) has worked on. So we need to run a query finding out all the part-time students and the count of tickets the student has worked on for the past 1 year.

Query J

The Helpdesk manager wants to know the names of all employees who resolved a ticket completely, or resolved it by working on it last. So we need to run a query naming all the employees who last worked on a ticket and resolved it, including the date, for the past six months.

SQL Code:

Query A

```
SELECT [Tickets].TicketID, count([EmployeeTickets].EmpID) AS [Number of Employees per ticket]
FROM EmployeeTickets, Tickets
WHERE Tickets.TicketID=[EmployeeTickets].TicketID
GROUP BY [Tickets].TicketID;
```

Query B

```
SELECT Employees.EmpName, EmployeeTickets.EmpID, EmployeeTickets.TicketID,  
Tickets.TicketImpactedArea, count (*) AS [Sum of Tickets]  
  
FROM Employees, EmployeeTickets, Tickets  
  
WHERE (((Employees.[Employee ID])=[EmployeeTickets].[EmpID]) AND  
((EmployeeTickets.TicketID)=[Tickets].[TicketID]) AND  
((Tickets.TicketImpactedArea)="Microsoft Outlook 2010") AND  
((Tickets.TimeLogged)>#4/1/2012#))  
  
GROUP BY Employees.EmpName, EmployeeTickets.EmpID, EmployeeTickets.TicketID,  
Tickets.TicketImpactedArea;
```

Query C

```
SELECT TicketImpactedArea, count (*) AS [Number of Tickets Received Off Hours]  
  
FROM SourceDetails, Tickets  
  
WHERE SourceDetails.TicketID=Tickets.TicketID  
  
And TimeEntered > #3/31/2012#  
  
And OffHours = True  
  
Group By TicketImpactedArea;
```

Query D

```
Select EmpName, TimeStarted, Tickets.TicketID  
  
From Employees, EmployeeTickets, Tickets  
  
Where Employees.EmpID = EmployeeTickets.EmpID  
  
And EmployeeTickets.TicketID = Tickets.TicketID  
  
And EmpEndDate IS Null  
  
And TimeResolved IS Null;
```


Query E

```
Select DeptName, TicketImpactedArea, count (*) AS [Microsoft Sharepoint Tickets]
From Customers, Tickets
Where Customers.CustID=Tickets.CustID
And TicketImpactedArea = "Microsoft Sharepoint"
Group By DeptName, TicketImpactedArea;
```

Query F

```
Select Tickets.TicketID, TicketImpactedArea, TimeResolved, TimeLogged
From Tickets
Where TimeLogged > #6/30/2012#;
```

Query G

```
SELECT SourceType, avg(QueueTime) AS [Average Queue Time], avg(TotalTimeLapsed) AS
[Average Time Lapsed]
FROM SourceDetails
Where TimeEntered > #3/31/2012#
Group by SourceType;
```

Query H

```
SELECT avg(EmployeeTickets.TotalTimeBacklogged) AS [Average Time Backlogged],
CallCenterorFront, SourceType
FROM EmployeeTickets, Tickets, SourceDetails
WHERE Tickets.TicketID = EmployeeTickets.TicketID
AND Tickets.TicketID = SourceDetails.TicketID
AND TotalTimeBackLogged IS Not Null
Group By SourceType, CallCenterorFront
Order By CallCenterorFront;
```

Query I

```
Select Employees.EmpName, count (*) AS [Count of Tickets]
From Employees, EmployeeTickets
Where Employees.EmpID=EmployeeTickets.EmpID
And TimeStarted>#09/01/2011#
And FullorPartTime = False
Group By Employees.EmpName;
```

Query J

```
Select EmpName, TimeResolved, EmployeeTickets.TicketID
From Employees, EmployeeTickets, Tickets
Where Employees.EmpID=EmployeeTickets.EmpID
AND EmployeeTickets.TicketID=Tickets.TicketID
AND TimeResolved=TimeClosed;
```

Space Estimates

Customers:

Estimated Length of Each Row: $15(\text{CustomerID}) + 30(\text{CustomerName}) + 15(\text{CustomerPhone}) + 30(\text{CustomerEmail}) + 25(\text{DeptName}) + 2(\text{CustStaffOrStudent}) + 30(\text{CustomerAddress}) = 147$

Estimated Number of Rows: 4,500

Explanation: Seeing as WPI is an institution with between 4,000 and 5,000 students it makes sense that there would be quite a large number of rows. Of course not every student will use the helpdesk, so 4500 seems reasonable. This number seems large enough to encompass the number staff that would be in the helpdesk's database as well.

Estimated Number of Rows in 5 Years: 30,000

Explanation: As a university helpdesk it would be unrealistic for them to keep ticket data for anything more than 5 years. As a result after 5 years, the database would be purged of customers associated with 5 year old tickets. Realistically then, the database would never have more than about 30,000 rows. Having this many rows is reasonable as the student body grows a little bit each year and it also accounts for staff. It is reasonable to retain this large number of rows because it will be relevant for up to 5 years later as students attend WPI for 4 to 5 years.

Estimated Size of Current Table: $147 \times 4,500 = 661,500$

Estimated Size of Table 5 Years Later: $147 \times 30,000 = 4,410,000$

Employees:

Estimated Length of Each Row: $10(\text{EmployeeID}) + 30(\text{EmployeeName}) + 30(\text{EmployeeEmail}) + 15(\text{EmployeePhone}) + 2(\text{FullOrPartTime}) + 4(\text{EmpStartDate}) + 4(\text{EmpEndDate}) = 95$

Estimated Number of Rows: 30

Explanation: WPI is an institution with lots of work-study students that work for the helpdesk, so 30 rows seems realistic and encompasses the part-time student employees, as well as full-time employees.

Estimated Number of Rows in 5 Years: 120

Explanation: 120 is the estimate in 5 years because with each year, some students will leave the helpdesk job and some new ones will begin working. The full-time employees will for the most part remain constant, so the 120 accounts primarily for the new part-time student employees that will join over the 5 year period.

Estimated Size of Current Table: $95 \times 30 = 2,850$

Estimated Size of Table 5 Years Later: $95 \times 120 = 11,400$

Tickets:

Estimated Length of Each Row: $10(\text{TicketID}) + 10(\text{CustomerID}) + 4(\text{TimeLogged}) + 4(\text{TimeResolved}) + 30(\text{TicketImpactedArea}) + 255(\text{TicketDescription}) = 313$

Estimated Number of Rows: 20,000

Explanation: With up to 4,500 rows of customers, it is reasonable to anticipate an average of around 4 to 5 visits per customer over the course of the first year.

Estimated Number of Rows in 5 Years: 100,000

Explanation: Similar to the customers table after 5 years this table would be purged of any tickets from 5 years ago, so that the table would never exceed about 100,000 rows. It seems plausible that there will be a consistent number of tickets over the course of the 5 years. It is reasonable to retain this large number of rows because it will be relevant for up to 5 years later as students attend WPI for 4 to 5 and their past issues could still be relevant within this time frame.

Estimated Size of Current Table: $313 \times 20,000 = 6,260,000$

Estimated Size of Table 5 Years Later: $313 \times 100,000 = 31,300,000$

EmployeeTickets:

Estimated Length of Each Row: $10(\text{TicketID}) + 10(\text{EmployeeID}) + 4(\text{TimeStarted}) + 2(\text{TotalTimeBacklogged}) + 2(\text{TicketHoldTime}) + 4(\text{TimeClosed}) + 2(\text{TotalTimeSpent}) + 2(\text{CallCenterOrFront}) = 36$

Estimated Number of Rows: 80,000

Explanation: There will be about 80,000 rows in this table as multiple employees may work on one ticket, so there will be multiple employee ticket rows that pertain to one ticket. Given that there are about 20,000 tickets from the Tickets table, it is reasonable to think that on average one

ticket may be worked on by about 4 employees or worked on four different times by the same employee.

Estimated Number of Rows in 5 Years: 400,000

Explanation: After 5 years, the number of issues that an employee works on will probably be quite consistent from year to year. The helpdesk needs to be able to still have a record of what employees worked on, so this large number of rows will be important for retaining such information. Again, each ticket can be worked on multiple times, hence there will be more employee tickets than tickets, since the employee tickets table tracks the different periods of time that a ticket was worked on and by whom.

Estimated Size of Current Table: $36 \times 80,000 = 2,880,000$

Estimated Size of Table 5 Years Later: $36 \times 400,000 = 14,400,000$

Source Details:

Estimated Length of Each Row: $10(\text{SourceType}) + 10(\text{TicketID}) + 2(\text{QueueTime}) + 4(\text{TimeEntered}) + 2(\text{TotalTimeLapsed}) + 2(\text{OffHours}) = 30$

Estimated Number of Rows: 50,000

Explanation: There will be 50,000 rows because any particular ticket could end up having multiple sources, for example if a customer calls and gets a ticket for their issue, and then later on does a walk-in regarding the same issue. Seeing as there are 20,000 rows in the tickets table, 50,000 rows in the source details table is reasonable as it accounts for some of the tickets having up to all three source types (phone, e-mail, walk-ins), but not all, as most tickets will not go through every source type.

Estimated Number of Rows in 5 Years: 250,000

Explanation: After 5 years this data will surely have amounted to about 250,000 rows, as there are no shortages of technological issues and people always need technological assistance of some form or another.

Estimated Size of Current Table: $30 \times 50,000 = 1,500,000$

Estimated Size of Table 5 Years Later: $30 \times 250,000 = 7,500,000$

Summary of Space Estimates

	Customers	Employees	Tickets	Employee Tickets	Source Details
Length of Each Row	147	95	313	36	30
Number of Rows	4,500	30	20,000	80,000	50,000
Number of Rows in 5 Years	30,000	120	100,000	400,000	250,000
Size of Current Table	661,500	2,850	6,260,000	2,880,000	1,500,00
Size of Table in 5 Years	4,410,000	11,400	31,300,000	14,400,000	7,500,000

WPI Technology Helpdesk Database Design

The initial database is designed for managers of WPI helpdesk only, which is why it is currently not password protected. When a user opens the database, it automatically opens to the front page of the database. Our database relies on simplicity so that managers can have easy access and useful information in the database. The front page of our database is broken down into several segments.

Image – Home Page



- Database Maintenance – This hold the buttons to all the tables. There are two tabs, separating viewing and editing purposes. Each command button opens up the form for the records on the table on a new tab. This will ensure a user can always access the home page. We also have a label describing what the buttons represent. The ‘View’ buttons lead to forms through which the user cannot update any data. However, there are list boxes available for ease of use. There is also a sub-table (wherever appropriate) which connects the record to its foreign key in another table. There are several buttons at the bottom for convenience – Go To First Record, Go To Last Record, Previous and Next. The ‘Modify’ buttons lead to table records in a form which can be modified. Apart from the usual commands, these forms also include New Record, Undo, Print, Delete and Save buttons. The New Record button is to add new data in the database. The Delete button attempts to delete a record in the database but will only be successful if that record is not being referenced anywhere else in the database. For example, if we try to delete a record in the Employees table, we’ll get a few error messages which display to the user that the record cannot be deleted as the information on the tickets the employee has worked on still exists in the database. This is a form of error-checking methods so that useful information isn’t deleted when trying to delete a particular record. But if we do delete an entire record successfully, it deletes it from the particular table in the database. The Print button opens up the Print dialog box for printing purposes, and the undo button resets the system to its previous position.

Image – ‘View’

WPI Technology Helpdesk Database View Employees Assigned per Ticket

View Employees Assigned per Ticket

Select Ticket

Ticket ID

Employee ID

Time Started

Total Time Backlogged

Ticket Hold Time

Time Closed

Total Time Spent

Call Center or Front ☒

Go To First Record Go To Last Record

Previous Next

Image – ‘Modify’

WPI Technology Helpdesk Database Modify Customers

Modify Customers

Select Customer

Customer ID

Customer Name

Customer Phone

Customer E-mail

Department Name

Staff or Student ☐

Customer Address

09 Aladdin Agrabah
15 Hayden Summer
151 Bob Builder
17 Robert Hershey
19 Eugene Iver
29 Watson Monderir
33 Tom Cruise
46 Alexander Tyson
57 Lisa Mars
59 Charlie Brown
75 Tumida Koper
79 Hyena Misters
87 Tim Mayers
93 Paul Downey

Ticket ID	Time Logged	Time Resolved	Ticket Impacted Area	Ticket Description
25	3/18/2012 8:00:00 AM	3/18/2012 8:23:00 AM	Microsoft Sharepoint	Randomly loses connection

Record: 14 1 of 1 No Filter Search

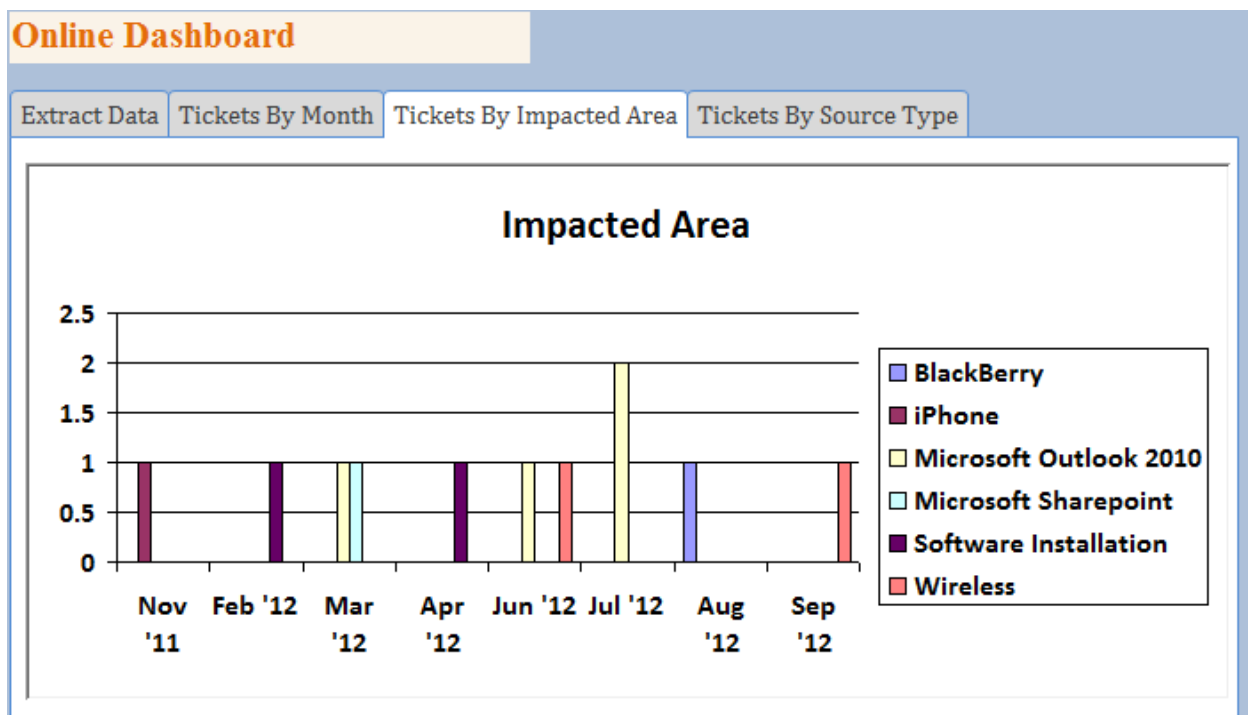
Go To First Record Delete Save Go To Last Record

Previous Undo Print Next

New Record

- **Online Dashboard** – The dashboard is an interactive experience present for managers to analyze the data and have a visual understanding of trends and patterns. The first tab allows the manager to export several groups of data to an excel file and save it anywhere. The other tabs displays up-to-date charts and graphs about useful data that the manager would like to analyze on a regular basis – Tickets By Month, Tickets By Impacted Area, and Tickets By Source Type. These charts will update automatically as new data enters the system. The Online Dashboard is user-friendly and provides simplistic visualization of key factors to managers, which can be readily changed according to managerial needs. This will also help managers to make fast and informed decisions.

Image – Online Dashboard Features



- **Reporting** – This part of the home page provides different tabs full of reports that the manager wanted to generate. He can update and edit this box to his liking as time goes

by, where he includes the reports that he finds most useful. Currently we have 10 reports listed, 5 on each tab. These reports are based on the 10 queries the manager wanted us to generate. We provide them in the form of reports so that a user cannot manually edit the data.

Image – Reports

Ticket ID	Ticket Impacted Area	Time Resolved	Time Logged
45	Microsoft Outlook 2010	7/10/2012 12:44:00 PM	7/10/2012 9:44:00 AM
75	BlackBerry	8/17/2012 2:17:00 PM	8/14/2012 8:06:00 AM
78	Microsoft Outlook 2010	7/13/2012 4:37:00 PM	7/13/2012 4:01:00 PM
9	Wireless		9/25/2012 8:00:00 AM

Page 1 of 1

- Other Aspects – The home page has a logo at the top which describes the name of the school and the database. There are also two other buttons at the right bottom corner – About Us and Exit Database. Clicking the ‘About Us’ button displays a message box describing who created the database. The ‘Exit Database’ button allows the user to exit the database. The top-right corner also shows the date and time at which the database was opened.

Image – ‘About Us’

