Design and Implementation of Business Intelligence Systems

Tanvir Ahmad
Abstract

Design and Implementation of Business Intelligence Systems

Tanvir Ahmad

The goal of this project was to investigate the limitations of current Business Intelligence System (BI System) which was needed to re-design its architecture and embed two new modules. These two modules were named as Survey Form and Statistics Report.

The survey form has a very dynamic behavior, for example the administrator can generate different types of surveys instantly by drag drop quick functionality of the interface and he can also send that surveys to the users. The user will give the answers of survey questions and send them back to the management of the company where survey form modules helped them to analyze the answers in the form of graphs. This survey form is also providing five different types of scheduling system where administrator can schedule their surveys in five different ways like Once, weekly, Monthly, Quarterly and Yearly.

Statistics module has different behaviors. It implemented four different types of reports which are inter-related with each other. These four reports were named as financial report, human resource report, project overview and company overview. The finance report was responsible that it will show the revenue amount and source, from where the company generated this revenue. The second report was belonged to human resource department which was responsible for generating the overall situation of the employees and their projects. For example how many are on holidays, idle, working and going to be free soon. The third report was representing the projects overview of all the projects. This report is responsible for checking the status of all projects and show them in bar char who will show the management that how many projects are completed, going to start or delay and going to finish soon. The fourth and final report was responsible for generating report for the overall company status.

This project has been completed successfully. The design, architecture and, implementation of intelligent Survey Form and Statistics modules have been implemented and working successfully.
Acknowledgment

I would like to say thanks to those who helped me in this thesis. My supervisor Charlie Hanssson encouraged me throughout this thesis. He helped me a lot to understand this project and enabled me to complete this project within the time limit.

Secondly, I am indebted to my colleagues Mattias Fliesberg, and Erik Nahkala who has helped me a lot in solving technical issues. This thesis would not have been possible without their help.

Finally, I would like to show my gratitude to everyone who has helped me to complete this project.

Tanvir Ahmad
Contents

1. Introduction ................................................................................................................................. 8
   1.1 Problem Description ............................................................................................................... 8
   1.2 Aim .......................................................................................................................................... 9
      1.2.1 Survey Form Module ...................................................................................................... 9
      1.2.2 Statistics Module ............................................................................................................ 9
   1.3 Demarcations .......................................................................................................................... 10

2. Research Approach ..................................................................................................................... 11

3. Working Tools and Technologies ............................................................................................... 12
   3.1 Development Environment ..................................................................................................... 12
      3.1.1 Programming Language ............................................................................................... 12
      3.1.2 Database ....................................................................................................................... 13
      3.1.3 Web Server .................................................................................................................... 16
      3.1.4 Netbeans IDE 6.9 (Development IDE) .......................................................................... 18
      3.1.5 PhpMyAdmin (MySQL database management tool) ...................................................... 18
      3.1.6 Web Framework ............................................................................................................ 19
      3.1.7 HTML5 (Extensible Hypertext Markup Language) ....................................................... 21
      3.1.8 CSS3 (Cascading Style Sheet) ...................................................................................... 21
      3.1.9 jQuery ............................................................................................................................ 21
      3.1.10 ORM (Object Relation Mapping) .................................................................................. 21

4. Project Background ..................................................................................................................... 23
   4.1 Thumb Expression .................................................................................................................... 24
   4.2 Chat ........................................................................................................................................ 24
   4.3 Consultants ............................................................................................................................ 25
   4.4 Clients and Projects ................................................................................................................ 25
   4.5 Finance ................................................................................................................................... 25
   4.6 Time Reports ........................................................................................................................... 25
   4.7 Print Reports ........................................................................................................................... 25
   4.8 Statistics .................................................................................................................................. 26
   4.9 Overview ................................................................................................................................ 26
5. Design ................................................................................................................................. 27
   5.1 New architecture ........................................................................................................ 27
   5.2 New BI System ........................................................................................................... 28
   5.3 Module Architecture ................................................................................................. 30

6. Implementation .................................................................................................................. 32
   6.1 Survey Form ............................................................................................................... 33
       6.1.1 Administration Section of the Survey Form ....................................................... 33
       6.1.2 Scheduler ............................................................................................................ 40
       6.1.3 User Survey Form ............................................................................................... 41
       6.1.4 Results of the Survey form .................................................................................. 43
   6.2 Statistics ....................................................................................................................... 45
       6.2.1 Financial Report ................................................................................................ 46
       6.2.2 Human Resource (HR) Report .......................................................................... 47
       6.2.3 Project Overview ............................................................................................... 47
       6.2.4 Company Overview ............................................................................................ 48

7. Conclusion ......................................................................................................................... 49

8. Summary of Contribution ................................................................................................. 49

9. Future Suggestions ............................................................................................................ 49
List of Figures

Figure 1 Comparison between PHP, .Net and Java ................................................................. 13
Figure 2: Comparison of Databases ......................................................................................... 15
Figure 3: Comparison of Apache and IIS ................................................................................ 17
Figure 4: Comparison of CodeIgnitor and Symfony .............................................................. 20
Figure 5: Old architecture ........................................................................................................ 23
Figure 6: New System Architecture ........................................................................................ 27
Figure 7: New BI System detailed architecture ....................................................................... 29
Figure 8: Module architecture ................................................................................................. 31
Figure 9: General picture of new modules ............................................................................. 32
Figure 10: Survey Form database schema ............................................................................. 33
Figure 11: User Section of Survey Form ................................................................................ 34
Figure 12: Group List ................................................................................................................ 36
Figure 13: Template categories ............................................................................................... 38
Figure 14: Survey Form layout ............................................................................................... 39
Figure 15: Life cycle of the Survey Form .............................................................................. 40
Figure 16: One year time period frequency ........................................................................... 41
Figure 17: User Survey Form ................................................................................................ 42
Figure 18: Results of the Survey Form .................................................................................. 44
Figure 19: Survey Form results ............................................................................................. 44
Figure 20: Statistics Module Architecture ............................................................................. 45
Figure 21: The financial report ............................................................................................. 46
Figure 22: Human resource report ......................................................................................... 47
Figure 23: The Project Overview ........................................................................................... 47
Figure 24: The Company Overview ....................................................................................... 48
1. Introduction

Business Intelligence Systems (BI) aims to gather in depth information from the company data and to analyze that data using different types of computer techniques and plotting which results in different types of graphs that facilitate the company in the future decisions [1]. The Business intelligence term was introduced by Hand Peter Luhn (Research at international business management, IBM) and he defined it as a ”.... ability to apprehend the interrelationships of presented facts in such a way as to guide action towards a desired goal” [2]. The BI Systems were evolved from the decision support systems that were developed in the 1960s and continued till late 1980s. According to the Howard Dresner the term Business Intelligence was the “.concepts and methods to improve business decision making by using face-based support system“[3]. The BI Systems produced results from large chunks of data that are stored in data warehouses or data mart. On the other hand it is not necessary that every system that gets data from data warehouse is a BI system. Forrester defines the BI system as: “Business Intelligence is a set of methodologies, processes, architectures, and technologies that transforms the raw data into meaningful and useful information. It allows business users to make informed business decisions with real-time data that can put a company ahead of its competitors“[4]. The BI Systems offered different types of reports which provide facility in better decision making. It gathers information from big database and analyzed them and presents in different types of reports (e.g. tabular, charts, graphs etc.). These reports provide different types of statistics such as past and present situation of the company that could help the top management in the better decision making.

1.1 Problem Description

This project was conducted in an Information technology (IT) company (ABC) that provides solutions to different companies in Sweden. The company was using the BI system and they figured out that the current BI system has a lot of problems. First the present architecture of a system did not have a support for integration of new modules because they were not following any authentic pattern like Model View Controller (MVC) model or Service Oriented Architecture (SOA). Second the HTML pages were used to process most of the data on their local machines that slow down the local machine because the resources were already occupied by the already running programs on that machine. These two problems raised number of other issues that are as follows.
- Re-design the architecture of the system
- Minimize Ajax requests
- Remove DOM Objects and use proper HTML templates.

They really need to redesign the architecture of the current BI system because it does not have well structured architecture where they can integrate new modules in the future. Second they want to minimize the Asynchronous JavaScript and XML (AJAX) requests because the HTML pages of system were populating through Ajax requests that slow down the machines performance. Third all Ajax requests response was coming in JavaScript object notation (JSON) objects array which have huge size of data. They were using JavaScript language to manipulate the JSON objects array for creation of Document Object Model (DOM) objects in their HTML pages and that heavy the operations on pages, all these actions were affecting the performance of HTML pages and working machine as well.

1.2 Aim

The objective of my work was to re-design the architecture of old BI system so that it provides the facility to integrate new modules and to handle the number of issues that they had in the old BI system. Furthermore I was liable to build two new modules. Those two modules were Survey Form and Statistics reports.

1.2.1 Survey Form Module

The survey form module was required for consultants and clients of the company. Their consultants often work at customer work place and they want to keep track of all the information and to keep in touch with them. Through the survey form module the company can ask different types of questions about their working environment and stress of the job etc. and vice versa with the clients about consultants and progress of the projects. The survey form module generates the survey forms that will assign to the specific groups of users and each form was available for a specific period of time. The survey form will helps to get useful information from the clients and consultants. Later that information facilitates the higher authority to compare the previously received data with current data. It helps to analyze the satisfaction level of the consultants and clients about the company or the project itself.

1.2.2 Statistics Module

Statistics module had aim to provide four different types of reports which were called financial, human resource, project overview and company overview. Every report will generate the important statistics of the company.
They did not have any specific way to calculate the company revenue, cost, and salaries of the employees within a specific period of time that is why they wanted such as system from where they can generate full report about the company financial situation.

Second report was belongs to human resource department of the company which was responsible to generate a report that show them that how many consultants worked at how many projects, which consultant was difficult to sell in market and which one was the best resource for the company.

Third report was belongs to projects of the company. The management of the company wanted to know that how many projects they had done within a specific period of time and how many they completed within a dead line and after the deadline. This report was responsible to show them that how many projects they have completed before dead line and which project was delayed and which one is going to complete soon. All these results will be representing in the form of pie chart.

Fourth and final report was belongs to company overview. It means that they wanted to see full overview of the company. How the company business is going? Is it growing fast? Or is there any chance to get loss in some projects. Above three reports already generate a financial condition of the company, projects of the company and human resource of the company. By using these three reports I can easily generate a report from where they can see the complete and present situation of the company.

1.3 Demarcations
Typically the BI systems are too complex and have a big domain. So it was difficult to research on all the domains of the BI system. Due to limited resources available at the company that restricted me to only design and implement the survey form module, Statistics module and try to improve the efficiency of the old BI system by re-structuring the previous architecture of the system.
2. Research Approach

Research is defined as “…. a systematic process of collecting, analyzing and interpreting information in order to increase our understanding of the phenomenon we are interested in or concerned of” (Leedy and Ormrod, 2005). Research could be performed in two ways. One way is to enhance the knowledge within the existing knowledge and the other is to generate a new knowledge (Gliner and Morgan, 2000 pp. 5). This thesis follows the first way that is to enhance and improve the functionality within the existing BI systems. In order to enhance and improve the knowledge I followed Ralph Kimball (2008) method which proposed two different way of gathering requirements for the BI systems. He described that “business users and their requirements impact nearly every decision made throughout the design and implementation of a DW/BI system. The business requirements sit at the center of the business core and are related to all aspects of the daily business processes” [5]. There are two ways of gathering information are conducting different types of interviews and facilitated sessions. I had chosen the first way that is to conduct different types of interviews for that I had conducted both formal and informal interviews with five different set of users. Both formal and informal ways were used in order to avoid biases that each method inherent (Interaction Design- Beyond human computer interaction, 2007, p.290). Formal interview contains set of questions that were asked to Chief Executive, Chief Operating Officer, Chief Financial Officer, Sales Manager and Education Manager. Informal interview was also conducted with those users in which different dialogues, chat, meetings, phones and emails were held. In both formal and informal, the design of new system was shown to those users in order to collect their feedbacks.
3. Working Tools and Technologies

The BI systems are unique because every BI system is based on the specific business logic and it creates specific intelligent reports for the company. Therefore I selected some most popular and widely used software from all over the world which helped me to develop this system. I have discussed every software and tool with respect to its needs and functionality in the following:

3.1 Development Environment

I was providing web based solution for the BI System. Therefore in order to choose the best development environment was not an easy task because there were a lot of options available for web. Since my focus was providing cheap, efficient and scalable solution for this project. Therefore I selected the best tool and software after the comparisons of their competitors. In the following section I have different comparisons of the web technologies with nearest competitors where you can see that how I choose my tools for this project and why?

3.1.1 Programming Language

There were many programming languages available for web but three of them were very popular in software industry. PHP, .Net and Java were the most important languages for web and each of them has own properties, advantages and disadvantages according to the specific requirements of the projects. I searched on the Internet “Which language is better for web?” I found hundreds of articles who gave answer about this question but most of the articles were biased because most of the authors were trying to promote their own language over the others. So I am sure all of these languages we can use for big projects but when and where we can use these languages? It is a big question mark today.

PHP and Java are open source languages but .Net is developed by Microsoft. .Net has huge cost for running big projects and small projects. Therefore I had a discussion with the company before starting of this project that how much budget they have for this project. They said to me that they do not want to spend money for purchasing tools. They suggest me to choose tools from open source directory which helped us to minimize the cost of this project. This option did not allow me to choose .Net. Now I have two options PHP and Java.

Both PHP and java were powerful languages for websites but I had found one problem on Internet when I was searching “which language is better?” Java seemed like a web language for corporate sector and PHP used by peoples who were running their own websites.
Let me discuss which language provided me cheap solution because I was looking for cheap solution. I checked that hosting server of Java websites was very expensive and these were belongs to private industries. Other problem with Java was that its project configuration required more time, development process was slow because it takes a lot time during compiling and bug fixing. When I looked at PHP and compare with Java it had a lot of advantages for me. For example its hosting servers were very cheap and it was very popular for medium scale projects. Development process was very fast because very rich IDE available for PHP there developers can manage enterprise level of projects very easily. Finally I choose a PHP as web language for my project.

**Comparison of PHP, .NET and Java**

<table>
<thead>
<tr>
<th>Area</th>
<th>PHP</th>
<th>.NET</th>
<th>Java</th>
</tr>
</thead>
<tbody>
<tr>
<td>License</td>
<td>Free</td>
<td>Not Free</td>
<td>Free</td>
</tr>
<tr>
<td>Support</td>
<td>Free</td>
<td>Free</td>
<td>Free</td>
</tr>
<tr>
<td>Platforms</td>
<td>Multiple</td>
<td>Windows Only</td>
<td>Multiple</td>
</tr>
<tr>
<td>Hosting</td>
<td>Cheap</td>
<td>Expensive</td>
<td>Expensive</td>
</tr>
<tr>
<td>Security</td>
<td>Very Good</td>
<td>Bad</td>
<td>Good</td>
</tr>
<tr>
<td>Performance</td>
<td>Very Good</td>
<td>Require expensive hardware</td>
<td>Require substantial configuration and expensive hardware</td>
</tr>
<tr>
<td>Scalability</td>
<td>Very Well</td>
<td>Can be difficult to scale</td>
<td>Good</td>
</tr>
<tr>
<td>Frameworks</td>
<td>Many good frameworks available</td>
<td>Only one standard Framework</td>
<td>Only one standard Framework</td>
</tr>
<tr>
<td>Configuration</td>
<td>Extremely easy</td>
<td>Not very flexible</td>
<td>Flexible</td>
</tr>
<tr>
<td>Compatibility</td>
<td>Backward Compatibility</td>
<td>Often break functionalities</td>
<td>So many problem between old and new versions</td>
</tr>
</tbody>
</table>

*Figure 1* Comparison between PHP, .Net and Java

### 3.1.2 Database

Database is a type of software where we can store information in an organized form. There are two types of popular database are available one is called relational database and other is called NoSQL database. We can use relational database where we know that our objects have fixed number of properties and we can define their schema very easily in our database. But where we will not be sure about our objects properties then there we shall use a schema free database.

In my case I found that I had fixed number of properties of my objects and I can easily define their schema in relational database. After realizing this I decided that I will use relational database instead of schema free database.

After a little effort I found that there are many popular databases that I can use in my project. MS SQL Database, Oracle Database and MySQL were the most famous ones but I had to
choose one of them to full fill my project requirements. Let me start to dig into one by one and find out that what type of functionalities they had, how easy to use them, How much help was available from community and which one was the most fits with my requirements?

**MS SQL**
MS SQL was a relational database produced and marked by Microsoft. We can interact with database by two query languages T-SQL and ANSI SQL. Microsoft developed this by using three famous computer languages C, C++ and C#. MS SQL only supports Windows platform therefore users were bound to use Windows Server platform. Although it provided all functionalities that we required from a relational database. For example it had strong support of query language, procedures, triggers and views. It also had a strong community help from many online forums. Microsoft also providing huge and well documented help in the form of MSDN library there every developer can find every type of help very easily. Next thing to find out that does it belongs to open source community? Do I need to buy a license? If I shall buy a license then how much money I need to pay? I figured out that Microsoft had its license ownership and if I want to use this database then I need to buy a license which was very costly. The company does not want to spend money for buying software. Finally I skip this option and start searching for Oracle database.

**Oracle**
Oracle database was a relational database produced and marked by Oracle Corporation. Commonly Oracle database called as simply Oracle. Oracle database developed in two famous languages C and C++ which supports on all types of platforms like UNIX, Linux, Windows and MAC. That is why it can also run on all these platforms which were much better choice than MS SQL. SQL was the main language for communication with Oracle which supports all types of CRUD (Create, Read, Update and Delete) operations. It also has support for writing procedures, triggers and views. Oracle was a big database and it was very popular among corporate sector where they had very highly and secured application. For example in banks, ATM machines, telecommunications companies, Airline control database etc. but it was also use in enterprise level applications where companies were running payroll systems, attendance system of employees.

Oracle was also providing support for their customers but their customer needs to buy a premium package for getting different types of support. They were providing online documentation for everyone. Finally let me check out that do I need to buy a license if I want to use in my project. I found that Oracle had huge cost for licensing of this database. I was
really interested to use this database but due to costs and limited amount of help I also skip this database from my list and start searching for MySQL.

**MySQL**

MySQL was a relational database developed by Swedish company MySQL AB but later on it owned by Oracle Corporation. MySQL developed in two famous languages C and C++ which supports on all types of platforms like UNIX, Linux, Windows and MAC. MySQL had also support on all these platforms where these languages have. That is why I can say that it gives me more independence of choosing database for my project. Like the other relational databases MySQL was also using SQL language for CRUD operations on database tables and it had also support for writing procedures, triggers, functions and views. MySQL belongs to open source community and most of the open source software that requires full featured database management system was also using MySQL.

MySQL was very popular choice of database for web applications and 70% of web applications were using this database. Web applications which were using MySQL database: Wikipedia, Facebook, Google, Drupal, Wordpress, Joomla and phpBB. Due to huge social network of MySQL community it had a lot support from users in the form of discussion forums, articles, books, tutorials and web seminars. MySQL was also providing online well-structured documentation which was very helpful and easy to understand. The main advantage for everyone was that it had not license cost and providing freedom to use it under General Public License. This was the only open source database which was competing with all other competitors like MS SQL and Oracle.

**Comparisons of Databases**

<table>
<thead>
<tr>
<th>Feature</th>
<th>MySQL</th>
<th>MS SQL</th>
<th>Oracle database</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platforms</td>
<td>UNIX, Linux, Windows, MAC</td>
<td>Windows</td>
<td>UNIX, Linux, Windows, MAC</td>
</tr>
<tr>
<td>License</td>
<td>Free</td>
<td>Not free</td>
<td>Not free</td>
</tr>
<tr>
<td>Installation</td>
<td>Very easy</td>
<td>Hard</td>
<td>Medium</td>
</tr>
<tr>
<td>Drivers</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Graphical Interface</td>
<td>Yes - not good</td>
<td>Yes – very good</td>
<td>Yes - good</td>
</tr>
<tr>
<td>Triggers</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Procedures</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Scalability</td>
<td>Yes</td>
<td>Yes – not easy</td>
<td>Yes</td>
</tr>
<tr>
<td>Public help</td>
<td>Yes – very good</td>
<td>Yes – very good</td>
<td>Yes – not good</td>
</tr>
<tr>
<td>SQL</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

*Figure 2: Comparison of Databases*
After analyzing of these three databases I found that MySQL was the only database that fits with my project requirements. All databases were very good but the problem was that they were costly and not popular for web application. MySQL provided me open source freedom and all time support as well. MySQL is free with all these features that become factor of minimizing the cost of the ownership. That is why most of the websites are migrating towards this database. It also comes with LAMP package so it is highly configure in this distribution and this is one of the best database that mostly use with PHP and Apache.

3.1.3 Web Server

Web server is software that provides help to get access of contents through Internet. Most of the web servers are use by hosting companies for managing web applications. The web server can deliver HTML pages and other documents like style sheets, JavaScript, images, documents etc. to the client browser. Mostly Web browsers are acting as a client and sending request in the form HTTP request and receive a response as an HTML page. In this world there are many web servers available but I will discuss only two because these two are very popular among web industry. These two web servers are Internet Information Service (IIS) and Apache.

Internet Information Service (IIS)

IIS was a web server which was developed by Microsoft. It worked only on Windows platform that is why it integrates IIS, ASP.NET, Windows communication Foundation, and Window SharePoint Service together. IIS was the second most popular web server after Apache. 20% of websites were running on IIS server. It supports HTTP, HTTPS, FTP, FTPS, SMTP, and NNTP types of requests. So it was very good for web application. I got one problem that it supports only ASP.Net technology but I was using PHP which did not have support with IIS. Finally I skipped this option and moved forward to the Apache.

Apache

Apache server was a computer program that hosts different programs, directories, and files. Apache makes these files accessible to the Internet, intranet and local host users by HTTP requests. Apache renders the entire HTML and PHP files and sends them to the user browser. The Apache server was different from its competitors for example Microsoft IIS (Internet Information Server). Why was it different? Because it was free, open source and its code available for view and modify. The Apache was also coming with different packages for different platforms like WAMP (Windows, Apache, MySQL, PHP ) for windows, LAMP (Linux, Apache, MySQL, PHP ) for Linux and MAMP (Mac, Apache, MySQL, PHP ) for
Mac OS but the most popular package was LAMP and still it was getting more and more popularity.

Apache was a web server and almost 60% of web sites were using it. It was widely used all over the world because of open source and many other interesting features like it had many implemented and compiled modules which extend the functionality of its core from server side programming languages to authentication modules. It supports many languages but most commonly used in PHP, Python, Perl and TCL. The most popular authentication mod was mod_ssl [7].

**Comparison of Apache and IIS**

![Comparison of Apache and IIS](image)

*Figure 3: Comparison of Apache and IIS*

As you can see in figure 5 that Apache and IIS has competition but for the Apache is a win-win situation. As graph shows that in 2007 Apache popularity get down and Microsoft IIS get some popularity but again from 2010 Apache is getting more and more popular. A reason is that Apache is open source and it supports on every types of platforms like UNIX, Linux, Windows and MAC. That is why hosting companies do not need to buy a license and they are providing cheap hosting services to the customers.

**Selection of Web Server**

After analyzing the both web servers I found that Apache was the best web server for my project because it had all functionalities that I needed in my project. For example, it was open
source, very well integration with PHP and MySQL, installation was very easy on all platforms. Apache was available as part of LAMP distribution for Linux. This distribution makes very easy for setting up new web server by using Linux environment which was also open source and free.

Now I have all mandatory components for my project environment where I can start my project development. I setup my web server on Ubuntu which was Linux based operating system. I installed Apache2, MySQL 5.0, and PHP 3.3.2 which provided me full featured web server where I can host my web application and make my all files public.

Next step was to start development but before this I need some more tools which will provide me help during my development period. For example: Netbeans, Mercurial, PhpMyAdmin, ORM (Object Relation Mapping), Ant and LiqueBase. All these tools I selected for project management without any reason.

In the following headings I have short description of all these tools.

3.1.4 **Netbeans IDE 6.9 (Development IDE)**
Netbeans had lot of features that is why I used it in my project. (For example, it had built in support for the PHP advance features, Ant, Mercurial, and other features). It was an open source; free under Common Development and Distribution License (CDDL) and GPL version 2 licenses [22]. It provided help to manage my project folders and files very easily. The PHP plug-in provided an editor for writing well-structured code and highlight the specific block of code. It had also support for intelligently detecting already defined variables, classes and its methods. That is why it was one of the best IDE in the software development industry.

3.1.5 **PhpMyAdmin (MySQL database management tool)**
PhpMyAdmin was a free and open source software tool written in PHP, intended to handle the administration of MySQL over the web. PhpMyAdmin supports a wide range of operations with MySQL. The most frequently used operations are supported by the users interface (managing databases, tables, fields, relations, indexes, users, permissions, etc.). While you still have the ability to directly execute any SQL statement [10]. It provided the graphics interface for MySQL database to perform different types of operation. It makes developer life easy otherwise it was very difficult to run all types of commands from the command prompt.
### 3.1.6 Web Framework

There are many web frameworks available on Internet and each one has own ideology but the common thing is that all are following the MVC (Model, View, Controller) architecture. Basically different frameworks are providing different design of architectures. Most of the web frameworks that belong to PHP technology are open source and give more choice to the developers that they can choose framework according to their project needs. I choose two frameworks that can meet with my project requirements. These frameworks are CodeIgnitor and Symfony. Let me discuss one by one and then I will choose one of them for my project.

**CodeIgnitor**

CodeIgnitor was an open source and powerful PHP framework that was specially design for small web application. It was providing many advantages to the developers who wanted to meet their deadlines and deliver their projects stable and secure to their clients. For example:

- It was providing exceptional speed.
- Very easy to integrate and start new project on pre build architecture.
- It did not force to developer for following specific coding standards.
- It supports template language but does not force to the programmer to use it.
- Very easy to follow the MVC standards.
- There is no extra configuration settings means almost zero configuration settings.

**Symfony**

Symfony was also open source and very popular framework regarding PHP technology. It was designed by Sensio Labs Germany. Symfony specially designed for small and large scale web applications. They had developed many libraries that developers need during development of any type of project. Symfony structure allowed developers to work effectively and efficiently on the most complex types of problems. If developers follow their instructions carefully then they will guarantees the stability, maintainability and upgradeability of the project. Symfony was also providing many advantages to the developers who wanted to deliver their projects on time.

- Its performance very fast and optimization was very easy.
- Very flexible to integrate new application in project.
- Its design provides facility that we can easily expand our project functionality.
- Stable and sustainable.
- It provides fully featured template engine.
- It provides advance way of MVC model.
- It was not easy to configure but they explain step by step in detail.

**Comparison of CodeIgnitor and Symfony**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Symfony</th>
<th>CodeIgnitor</th>
</tr>
</thead>
<tbody>
<tr>
<td>PHP4</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>PHP5</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>MVC</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Modules</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>ORM</td>
<td>Yes</td>
<td>Yes – but very basic</td>
</tr>
<tr>
<td>Authentication</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Cache</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Validator</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ajax</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>License</td>
<td>MIT</td>
<td>Apache/BSD</td>
</tr>
</tbody>
</table>

**Figure 4: Comparison of CodeIgnitor and Symfony**

**Selection of web framework**

After looked into CodeIgnitor I found that it was very good for small applications where we had small number of users, reasonable traffic and we did not need many technologies to integrate in one project. So I skip this because in my case application was an enterprise level and I need to integrate many other technologies as well. If you look at my goal in chapter 1 you will realize that I need a design that helps to the company in order to scale this project in the future. Other reason is that they want template engine that will generate and render HTML pages for this project. Finally I decided that Symfony was providing me all such type of facilities and I can easily use this in my project because it was easy to learn and integrate into the project environment.

After selection of framework I was about to start development but still I need to decide some more things for my HTML pages. Which version I will use for HTML, CSS? Which JavaScript library will I use for calling action of my pages? Why I look on it because there are many web browsers are available in the market and every browser has own standards of rendering HTML, CSS tags that is why it is very important if you want to stable your web pages in every browsers. After a little effort and reading articles on web I learned that most of the famous browsers were following latest standards of HTML and CSS. Finally I decided that I will use HTML5, CSS3 and jQuery for my HTML pages.
3.1.7 HTML5 (Extensible Hypertext Markup Language)
HTML is a basic language for web pages and it is written in the form of elements which are called tags. Every tag has two types of brackets starting and ending (like <html>). These tags usually come with pairs (like <head>, </head>). First tag is called starting and second tag is ending. I figure out that from many years there were no research on these tags but recently World Wide Web Consortium (W3C) did research on it and embedded new features in the HTML language [21] that called major release of the HTML revision.

3.1.8 CSS3 (Cascading Style Sheet)
CSS is primarily designed for separation of layout from the HTML document including font, colors, table layouts and text styles. I used CSS for presenting look and formation of my web pages written in a markup language. It is widely used language for styling HTML pages. It helped me a lot to separate my HTML document presentation, which includes all elements such as layout, colors, and fonts. This improves my content accessibility. It also becomes more flexible and controllable on other pages. It enables me to access my formatting styles on different pages and avoid duplication of my styles.

3.1.9 jQuery
jQuery is a fast and concise JavaScript library that simplifies the HTML documents traversing, event handling, animating, and Ajax interactions for rapid web development [18]. It is open source and had dual license under MIT and GNU General Public License version 2 [19]. It is very rich application because it is very simple and very powerful in that sense that we can get required results after writing a few lines of code. For example, I can put drag and drop functionality on my web pages with it. I found that this library is really useful for interface programming of web pages. Secondly, it provides me a plug-in that can customize my HTML components into attractive one like image works, check box, modification of dropdown interface etc.

3.1.10 ORM (Object Relation Mapping)
Object Relational Mapping (ORM) is a technique used in programming languages when dealing with databases for translating incompatible data types in relational databases. This essentially allows us to have a “virtual object database” that can be used from the programming language [11].

I used PHP Doctrine in my project which was a collection of selected PHP libraries that focused on the persistence functionalities. “Doctrine was an Object Relational Mapper (ORM) for PHP 5.2.3+ that sits on top of the powerful Database Abstraction Layer (DBAL).
One of its key features was the option to write database queries in a proprietary object oriented SQL dialect called Doctrine Query Language (DQL), inspired by Hibernate's HQL. This provided a powerful alternative to developers for SQL maintaining without requiring unnecessary code duplication” [12].
4. Project Background

The BI System was developed by the company in 2006. Till now they had released two versions of the system but now they are planning to add new functionalities to improve the efficiency of the system.

The BI System offers many services to the consultants of the company. These consultants are highly qualified professionals and among of these consultants each of them has different types of expertise in different areas of the computer science. As the time is progressing and everyone is getting more and more experience in his profession. The company wants to manage information of all the consultants in a better way. They were using a system to manage all these information but its architecture was as follows in figure 5:

![Figure 5: Old architecture](image-url)
The old BI System was providing seven different types of services which were as follows:

- Thumb Expression
- Chat
- Consultants
- Clients and Projects
- Finance
- Time Reports
- Statistics
- Overview
- Documents

4.1 Thumb Expression

Thumbs was very interesting feature in the old BI System because whenever any consultant was logged-in, the system instantly prompts the user with six different shapes of thumbs and asked “how do you feel today?” the user then selects a thumb sign according to his current mood and situation of the project. The most important thing was that it was taking only one or two seconds and the user got into his account.

![Thumb Expression]

Figure 6: Thumb Expression

The company obtained a lot of help from this feature. (e.g. they can analyze the consultants health, work load and they can identify the consultants who were not feeling good or maybe they were in work pressure. So the company can easily identify their problems and fix them as soon as possible.

4.2 Chat

In this module all consultants can communicate with each other via text messages. The consultants could only post text messages on the discussion board because it was common for everyone and there were no separate chat system. Due to this problem they were using Skype for internal communication.
4.3 Consultants
There were two types of consultants list: one was “Active” and the other was “Passive”. The active consultants were those who were currently working and the passive were those who were not working currently and the company was trying to sell them in the market. The company can easily add new consultants, view, update, and delete already saved consultant information.

4.4 Clients and Projects
There were two types of clients list one was “Active” and the other was “Passive”. The active clients were those whose projects were currently going on and the passive clients were those whose projects had been completed or will start in near future. This module shows client profiles, their projects and status of each project. It was also showing the complete progress of the projects. The user can also create new client, view already saved client profiles and projects.

4.5 Finance
Finance services were assisting the company in the calculation of the consultant salaries according to per hour rate. It was representing the different properties such as start date, end date, salary/hour, salary/month, tax rate, average, vacation and part time. It also showed the list of all consultants and presented their property values in each column.

4.6 Time Reports
This module was helping the project managers to assign total working hours of the newly created projects to the relative consultants. Each consultant can update their working hours every day in the calendar. The time report helped them to elaborate how many hours had been reported by each consultant regarding their project work and how many hours were remaining. This functionality assisted the consultants to easily report their working hours in the text box and add it into the calendar by using “drag and drop” functionality.

4.7 Print Reports
This module was generating three types of PDF reports stated as Invoice Generation System, Salary Specification, and Project Card. In invoice generation report they were creating the reports against one client by selecting specific period of time. The salary specification type was generating reports against the salary of every consultant within a specific period of time and the project card type was generating reports for a specific project. The system itself saved these reports in server directory because it also displayed the list of all reports so that next time the user of the system can easily download the already generated reports from the server.
4.8 Statistics
This represented the different types of the graph presentation of current year and it also had options for creating graphs according to the specific period of time. It contains weekly work load graph where it can be easily analyzed. How work load was divided in each week? It also contains the graph representation of the thumb expressions. Where the user can see that how many consultants are logged in every day and what was their feelings, either they were tired or fresh.

![Figure 7: Statistics](image)

4.9 Overview
This module represented the full view of the company in just one click and displayed all the information of the projects. It was also a short representation of all the projects and their consultants. (e.g. system can easily calculate status of every project that how many hours were estimated and what was their starting and ending date. While clicking on any particular day of the specific project the system pops up with detailed hours graph as shown in figure 8. This figure presented the three bars: the first bar presents how many hours were estimated at start. The Second bar presented how many hours have been worked out by the consultants and the third bar showed that how many hours were remaining to finish this project.

![Figure 8: Overview of Project Hour](image)
5. Design
The second chapter described that what type of tools I used to build this system and third chapter describe that what kind of old system was and how was its architecture. Now I will explain the design of new system and how I used all my tools to build this new system.

5.1 New architecture
The architecture in the figure 9 shows the general structure of this system which contains two new modules and core of the system. This core will provide the architecture where the company can integrate new modules very easily in the future. Most of the modern architectures are following this classical design for large projects. Since my focus was to develop this system. I divided my solution into three main parts. First I designed core structure of the system which was based on MVC architecture and other many things that I will discuss later. Secondly I designed architecture for its new module which was also based on the MVC architecture.

![Figure 6: New System Architecture](image)

The new BI system core is the heart of the system which will receive all requests from user and forward them to the particular module of the system. For a moment assume that it
forwards all requests successfully to the related modules. How it will forward all requests? That I will describe in next section. The next task was that I have to develop two modules and integrate with my new system. Every module contained two parts one for administrator and second for client. The module architecture is based on MVC model. It has one controller that will handles all request from user and administrator and send them to the related model. The model will communicate with the database and send the feed back to the controller back. After getting feedback from model the controller forwards the request to the template engine which renders the HTML pages and sends them to the browser.

5.2 New BI System
Figure 7 represents the general architecture of the BI system but it is not representing the detail information about the module. How modules are integrated with this system I will describe here but structure of modules I will describe in next section.

The web framework provided me great help in order to handle the HTTP request from web browser. As you can see in figure 7 there are two important part of framework. One is Front Controller and other is Symfony Kernel. Every HTTP request has two types of objects which are called Request and Response Objects. These two objects are very simple itself but the hard part of the application is that what will come in between these two requests. Symfony is specifically designed for solving this type of problems. Otherwise it is very difficult to handle the all types of HTTP requests like sending emails, handling form submissions, saving objects into database, rendering HTML pages and protecting content from hackers. It is also difficult to manage all types of these requests and keep my code organized and maintainable.

Traditionally if you look on the site building process of each web site then you will see that there are many URLs defined manually like index.php and contact.php. These URLs create a huge problem when somebody wants to change its name. Why it has problem? Because it has definition in different places of website for the navigation purpose therefore it needs to change its name everywhere that would be time consuming process. Other problem is that if I will create different PHP libraries then I need to include my all libraries in every file to maintain the consistency of the code. That would be a difficult and time consuming process as well. To overcome this problem I introduced front controller in my architecture which handles all types of request that comes from users. Introducing front controller is much better solution because it is a single PHP file that will handle all types of request that coming to your application. This approach solves all traditional problems like changing URL names,
defining libraries, configuration settings with database. All these definitions should be in one file and everybody can change them very easily and in a very less time.

Front controller has only the definition of all types of requests. It has responsibility to forward these request to the Symfony Kernel which handles these requests and call specific module controller. We will discuss about module controller in the next section. As you can see in figure 7 when Front controller gets a request it simply forward that request to the Kernel. It process the URL filters and send the response back to the kernel. After getting response back. The kernel call the specific module as shown in figure.

Figure 7: New BI System detailed architecture
5.3 Module Architecture
All modules of this project have the same MVC architecture as shown in figure 8. MVC is an architectural pattern and this is used almost in all programming languages. The MVC architecture helps me to separate my domain logics into different layers. It also helps me to control the flow of my application request in a good manner. MVC has different types of flavors but my application has control flow as follows

User makes an interaction with interface through browser and sends a request to the controller. The controller will handle the request and convert into an appropriate form which understandable for model layer. The model will take the action against this request and response back to the controller or directly to the view layer. This view layer will generate a HTML page against this response and forward to the browser.

In my modules when browser will send a request to the web server then web server will forward this request to the Front controller of the system which I discussed in 4.2 sections. Front Controller route this request to the related module of the system where special controller of the module will handle this request and forward to the model layer. I do not have simple model layer as mostly we have in the MVC architecture. I have integrated ORM layer with my model layer which sits between model and database. This ORM layer received a request from model and generates a query for database. After getting query from ORM the database generate a result and response back to the model layer. The model has responsibility to forward this result to the controller. The controller will forward this request to the View part of this module. The view is also not simple as we have in common MVC architecture. As the view will get a response from controller it will interact with HTML template engine. This will generate a HTML page for view and send to the view layer. The view layer just receives the HTML page and flushes to the web browser.
Figure 8: Module architecture
6 Implementation

In the previous section I described how I redesigned all structure of this project. Now in this section I will discuss that how many modules I suppose to implement and how I implemented these modules. There were two modules that I supposed to implement for this system. Figure 9 represents the general idea of these new modules. If you look at the Survey Forms at left side of the BI system then you can see that it has five different types of forms. These types are representing five different types of scheduling of this survey form which I will discuss in coming sections. On the right hand side of the BI System you can see statistics module which is representing four different types of reports. I will discuss in coming sections. What type of these reports are and how I implemented them.

Figure 9: General picture of new modules
After getting the requirements, when I was clear enough about my job I started implementation. I designed a database schema according to my requirements which as follows in figure 10.

<table>
<thead>
<tr>
<th>Objects</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>id, name, formid, categorytype (template, form)</td>
</tr>
<tr>
<td>CosForm</td>
<td>id, name, startdate, enddate, closeable, formtype(once, weekly, monthly, quarterly, yearly), activationdate, status</td>
</tr>
<tr>
<td>FormAverage</td>
<td>id, formid, unweighted, weighted, answerdate</td>
</tr>
<tr>
<td>FormGroups</td>
<td>id, formid, groupid</td>
</tr>
<tr>
<td>SurveyForm</td>
<td>id, formid, categoryid, questionId, answer, answertype, answertext, weight, answerdate</td>
</tr>
<tr>
<td>GlobalForm</td>
<td>id, formid, groupid, userid, usertype(customer, consultant), globalid, viewable, visitdate, answerdate</td>
</tr>
<tr>
<td>UserForm</td>
<td>id, userid, formid, answerdate, formtype, stratdate, enddate, visits, visitdate, activatedate</td>
</tr>
<tr>
<td>UserVisits</td>
<td>id, userid, formid, visits, visitdate</td>
</tr>
<tr>
<td>Group</td>
<td>id, name</td>
</tr>
<tr>
<td>GroupUser</td>
<td>id, groupid, userid, usertype</td>
</tr>
<tr>
<td>Question</td>
<td>id, question, categoryid, questiontype</td>
</tr>
</tbody>
</table>

Figure 10: Survey Form database schema

When I was ready with my schema I start to create my schema within MySQL database. By default MySQL has command line interface. By this interface we can run SQL queries one by one and create a database schema. But Running SQL queries through command line it was difficult and time consuming. So I used one open source tool PhpMyadmin which provided me complete interface. From this tool I can perform all types of operations on database very easily and quickly. Using this tool I create all my tables into database and start implementation of the survey form module.

6.1 Survey Form
The survey form has two types of interfaces. One for the administration and the other section is for the users. From administration section the company can create surveys and send them to the users. From the user section user can answer the survey questions and submit to the management.

6.1.1 Administration Section of the Survey Form
The administration section provides the dynamic feature for this module. I divided survey form layout into three columns as shown in figure 11. Left layout of the survey form shows
the list of all consultants and clients. Center layout represents the list of all groups of the users. These groups will be assigned to the survey forms. Last and most right layout provides the functionality where administrator can create new survey form.

![Survey Form Layout Diagram](image)

**Figure 11: User Section of Survey Form**

Let me discuss one by one how I implemented all these functionalities.

**Group Users**

In the survey form module controller is the responsible of every request that comes from browser. As controller will receive a request to load a HTML page for create the survey form. It will call an index method of the controller that will populate list of users, groups of users, survey forms and list of template categories at bottom section of the survey form.

As controller will receive a request it will call a database service related to particular entity class and create a connection with database. Because it is using persistent database connection so I do not need to worry about connection limitations. As it will establish a connection with database, the controller will send a request to the model for getting list of the users. The model class will pass the request to the ORM layer which is responsible for fetching data from database. I used PhpDoctrine ORM layer which is providing me facility that I can write my queries in the objects form instead of writing traditional SQL queries. The advantage is that I do not need to handle most of the database security issues myself. The most popular issue was database injection which is almost crash the application. After fetching data from database the ORM layer. It sends back the response to the model class. As model class will receive the data then it will forward the request to the controller where controller receives the response data and forward to the particular view. View receives this data and sends to the template engine which will generate a HTML page and forward to the view class. The view class will display this HTML page to the browser.
Form Group

The center layout represents the list of groups. Each group has list of consultants and clients so controller will call the index method which also follow the same request procedure as I describe procedure for user list at left layout. Extra feature is that there is create and delete operations exist for groups. When user will click on create button controller will receive a request and call a create group method which will send the request to the template engine that will create a HTML form and send it to the browser where user will interact this form and write a name of the group. After writing name of the group, user will submit the group form and validate the form field with jQuery form validation plug-in. if form is valid then it will allow the form to send data to the controller. The controller will receive the data and validate the form field at server side as well for secure validation. If validation will failed then error message send to the view which will display the message to the user browser. Otherwise controller will call the database service and establish a persistent connection. After creating a successful connection with database controller will send the HTML form data to the model. The model will forward this data to the ORM layer which insert the group name into database and send response back to the model. If response is true then group name added successfully otherwise there should be an error message. Whatever the response is, the model will send this response to the controller and controller will forward this message to the view which will display the result to the user browser.
After creating a group, now it is time to add consultants and clients in the group from left layout. I used jQuery multi select plug-in which provides me functionality that I can select multiple names by mouse click from user list. At once I have selection of users from list, the next step is just to drag that names and drop into the panel of group that I created before. As I will drop the names $.ajax() method of jQuery will call and create a JavaScript array of all user ids and types. As I discussed before there are two types of users, one is consultant and other is client. After creating the arrays of consultants and clients $.ajax() method will send the HTTP request to the Rest Controller. I introduced the rest controller for separate the Ajax requests. The advantage is that it is easy to separate the types of HTTP requests which provide the flexibility for maintaining the code structure of the controller. After getting the request the rest controller will call the database service and establish a persistent connection. When connection with database will be ready, the controller will grab the arrays information and convert them into PHP arrays. These PHP array has the ids and types of users. The controller will send these arrays to the model where model forward these arrays to the ORM.
layer. Now ORM layer will run the SQL query on Group User table and insert the all user ids and types against this group. After successful insertion into the database of the group users the controller will receive a response from model and re-load the user page. You can see the HTML layout in the group user list figure 12.

**Template Categories**

Template category section is specially designed for those set of questions that mostly use in the survey forms. The advantage of this section is that the management of the company can create a set of questions for future use because it is providing functionality to the management that when they will create a new survey form then they can use these template categories in the form.

In this section we can add, edit and delete categories. All these actions handle by the rest controller of the survey form module. When user click on the add category button then it will call the jQuery method show(). This method will show the HTML form for getting input from user. As user will write a name of the template category and submit the category form. The controller will validate the form and check the category name. Is it already exists or not? If it is already exists then controller will send the message to the user browser that duplicate category does not allow otherwise controller forward the category name to the model of the module. The model will send the value to the ORM layer which is responsible for inserting this value into the category table of database.

You can see in figure 13 that every template category has different type of questions. You can also add, edit and delete the questions from category. Every question will take the answer in the form of radio buttons or text area. I will discuss this functionality in the coming section when I will discuss user interface of the survey form. But in the administration section whenever we will define the answer we will specify the answer type as well. There is a radio button against every question type if it is checked then answer type should be text otherwise it would be choice of radio buttons where user needs to check one radio button while giving the answer.

For update functionality I used an editable plug-in of jQuery. Why I used this plug-in because it provides me functionality that I can easily update my category and question text. When user will click on the category or question edit image button then jQuery calls this plug-in which replace the text with HTML textbox. In this text box user can easily change text and click on the update image again. As user will click on the update image, the editable plug-in
will call the jQuery method $.ajax() and at the same time it will replace the textbox with updated text. But at the backend $.ajax() method will send the Ajax request to the rest controller which will receive the updated text and forward to the model. The model will forward this request to the ORM layer which will update the text into the database.

I have implemented an interactive layout for the template category section. As you can see in figure 13 that there is a list of categories which I have designed using CSS. On this list I have also implemented the jQuery SlideUp() and SlideDown() functionality. When user will click on the heading of the category SlideDown() method will open the category section and list out the questions. The advantage of this functionality is that user can easily manage many categories within small place of page.

![Select, Drag and Drop to Form]

**Figure 13: Template categories**

**Survey Forms**

Until now I have discussed three building blocks that I need for creating one survey. These three building blocks are group users, form groups and template categories. In this section I will describe the section of survey form layout which will use these three blocks. Here Figure 14 represents the original layout of the survey form administration section. Where user can create new surveys and send them to the users. Let me explain how this section will work from top to bottom.

Here we can schedule the survey forms with five different types of time periods. These time periods are defined as Once, Weekly, Monthly, Quarterly and Yearly. The Once type of scheduling represents the survey form which will assign to the users for one time. When all users send the results then system will close the survey form. Other four types of scheduling represents the survey forms which will be assign to the users at once. After assigning the survey form, the user will get the survey at start date of the schedule. According to the schedule type of the survey form the system will automatically calculate the schedule date for next time. When next schedule date will come the system itself send the same survey form
with same set of questions to the same groups of users which have been assigned to the survey forms before. This phenomenon will continue until the end date arrives. At arriving of end date the system will close the survey form.

![Survey Form layout](image)

**Figure 14: Survey Form layout**

**Implementation of Survey Forms**

In this section I will describe in detail that how I implement the above functionality. When user will click on the ADD FORM button at top right hand side of the form section, the show() method of the jQuery will be invoke and show the HTML form. This form consists of four input fields, Form name, start date, end date, and schedule type of the survey form. The form name represents the title of the survey form. The start and end date fields represent the active and closing date of the survey form. The fourth field defines the schedule type of the survey form. The administrator can create a new survey form very easily by using this form. He will just define the four fields and submit the form. The controller is responsible for validation and saving the values into database. The controller of the module will receive the post request and run the validation process on it. If validation process is passed then controller will forward the request to the model. Otherwise an error message will be generate and send back to the user. The model will receive the data and forward to the ORM layer which will run the insert operation on the database and response to the controller. If response is positive then it will refresh the page otherwise send the exception. Once the survey form defines it will provide the HTML panel where administrator can add categories and its questions. There I have provided two choices for them. First they can use the template categories as survey form categories. They can get this template category by drag and drop functionality. Second option is that they can also add categories and questions within a form panel as well. But template categories are providing quick solution for them because
management of the company does not have too much time for creating survey forms therefore they need to use this functionality. When template category will drop on the survey form panel then jQuery method $.ajax() receive the template category id and send to the rest controller which will copy the complete content of the template category and insert all these contents as a new category. Because I want independent categories in the survey forms therefore I copy it instead of saving its reference as foreign key. The advantage is that now they can modify their questions and its categories names. Due to this facility they can use these template categories in other survey forms as well.

**Life cycle of the Survey form**

![Diagram of survey form life cycle](image)

**Figure 15: Life cycle of the Survey Form**

### 6.1.2 Scheduler

In order to control the scheduling periods of each survey form I need to implement one scheduler. The scheduler is the main heart of the survey form. The scheduler divided the survey form duration into different number of time periods according to the schedule type. Figure 16 shows that if survey form has type weekly then scheduler will divide the duration into 52 time periods, similarly monthly survey will be divide into 12 time periods, Quarterly survey into 4 time periods and yearly survey one time in a year. This survey form duration is
calculated from start and end date of the survey form which is specified when survey form created.

![Figure 16: One year time period frequency](image)

I have implemented four cron jobs at apache server and tell to the apache that runs these jobs at the start of every time period. For example the weekly cron job will run on every Sunday night and send the weekly surveys to the users, similarly the monthly cron job will run on every last date of the month, the quarterly cron job will run on every last date of the quarter month of the year and yearly cron job will be run on every last date of the year.

### 6.1.3 User Survey Form

Before this section I have discussed all about the administration section of the survey form. How administrator can design new surveys and send them to the users. I think it is a right time to discuss about user survey form layout that how user will interact with this? How I implemented? Let me discuss short description of page layout that how user will get this survey? How user will give answers to the questions and submit them to the company management.

Figure 17 represents the HTML view of the survey form which has title and list of questions under different category names. As I discussed in template category section that when administrator will define a question. He will also specify the types of question input. Is it a text or choice for user? If you look at figure 17 then you can see that first question has text type and other questions have choice options for user. All these questions are dynamically defined at administration section therefore there is no limit of number of questions per survey. Administrator can define questions as much as he wants. Of course there will be a problem of listing when administrator will define a huge number of questions and it will also
affect the efficiency of the page. So I discussed this issue with the company. They told me that they do not have more than 10 questions in any survey. This answer makes me clear that there will be no problem with number of questions.

![User Survey Form](image)

**Figure 17: User Survey Form**

The question with text area allows only 200 characters to the user for giving answer. This limitation of text bounds the user to write answer short and to the point. The question with radio buttons has special feature. You can see in figure 17 that except first question all other questions have two lines of radio buttons. What does it means? Let me explain this with example. We have an example question “Do you like working environment?” after this question user will have two lines of radio button and each line has 10 options which are starting from 1 and ending at 10. Number 1 means least and 10 means maximum level of satisfaction with this question. First line of radio buttons represents the values of answer. For example user selects the 7 answer so it means that user is quite satisfied. But why second line of radio buttons? Second line is asking to the user about his answer that how much important this answer for him. If he thinks this question is very important for him and he would like that management should take action against his answer then he should choose around 10 because 10 means maximum worth and 1 means least worth of this answer.

After give the answers of all the questions when user submits the results. The controller will receive the results and run the weighted average formula which you can see below under Weighted Average heading.
Weighted Average
Step 1

Multiply answer and weight value and then add.
$answer_{multiply_weight_sum} += answer_{value} \times weight_{value};$

Step 2:

Add all weighted values of all answers.
$weight_{sum} += weight_{value};$

Step 3:

Divide answer from step 1 with answer of step 2.
$weighted_{average} = \frac{answer_{multiply_weight_sum}}{weight_{sum}};$

Why I used this formula because I wanted overall satisfaction level of the users. So this formula has properties that it can calculate the average results of the user answers. So we can compare average results with without average results. This represents to the management that how many users are satisfied with the company. I will show this with line graph in next section. At the backend the controller will send the results to the model which will save these results into the database.

If this form was schedule as a type of monthly then cron job will send this survey every month to the users and they will give the response. Why again because there is a possibility that last month user was not satisfied with working environment of the company and he gave answer and tell the company that he do not like this type of environment. When all users sent the response then company realized that most of the users are not satisfy with company environment so they need some modifications. The company did some modifications last month and asked again same questions. If users liked those modifications then there should be positive results otherwise this process will continue until the company will get the reasonable feedback. When they will feel that it is quite acceptable results then they will close this survey from the administration section.

6.1.4 Results of the Survey form
Now go back to the administration section and analyze the results of the survey form. I will elaborate these results with an example. For example company sent one survey to the employees in January 2011 as shown in figure 18. This survey was scheduled as a monthly type. According to the schedule the system will send this survey every month by cron job. On the other side the user got this form and sends the response back.
<table>
<thead>
<tr>
<th>Time Period</th>
<th>Answer Rate</th>
<th>Overall Results</th>
<th>Overall Results(Weighed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sent 2011-04-01 (April)</td>
<td>20/20</td>
<td>7.4/10 (74%)</td>
<td>9.3/10 (93%)</td>
</tr>
<tr>
<td>Sent 2011-03-01 (March)</td>
<td>20/20</td>
<td>6.5/10 (65%)</td>
<td>7.0/10 (70%)</td>
</tr>
<tr>
<td>Sent 2011-02-01 (February)</td>
<td>20/20</td>
<td>5.9/10 (59%)</td>
<td>6.3/10 (63%)</td>
</tr>
<tr>
<td>Sent 2011-01-01 (January)</td>
<td>20/20</td>
<td>5.4/10 (54%)</td>
<td>6.0/10 (60%)</td>
</tr>
</tbody>
</table>

You can see in figure 18 that first column represents the Time period of sending this form. Second column show that how many users were and how many responded. The third column represents that what was the average result of each user answer. The fourth and important column represents that what was there weighted average results. The System will show these results to the management in the form of line graph as shown in figure 19. This graph shows that as time is progressing and company is taking positive steps for making good environment for the employees. The satisfaction levels of the users are increasing.

Because now users are familiar with each other and with the company environment so everybody use to work here. It means that the company learn that by taking some little but positive steps they can make their employees happy within four months. This will affect the efficiency of the employees. It also increases the chances that the team will complete the project successfully before deadline.
6.2 Statistics

The statistics module implemented the four types of reports which were called financial report, HR report, Project Overview and Company Overview. Figure 20 represents the architecture of this module which represents that Report is an abstract class and other four classes are concrete classes. The abstract class has definition of all common methods. It also has implementation of some common methods which are implementing common functionalities to the reports. The concrete classes are implementing all unique functionalities and inherited the methods from abstract class.

![Figure 20: Statistics Module Architecture](image)

Report class has some common parameters that need every report like Full report, Remember setting, Holidays in line charts and report duration time. The Full report means that when user select this option. The system will generate the report of everything that belongs to this report. The second parameter remembers setting. It is a dropdown box which represents that user can save some sort of quick report setting. This will help to the user that he can generate similar report by selecting the previous saved setting. The third option is holidays in line charts which will show to the user that how many national holidays and weekends are coming during the time of this report. The fourth and last parameter is duration. The duration parameter allows to the user that he can generate specific report by selection of two dates. When user will select the dates the system will generate the report within selected duration of dates.
6.2.1 Financial Report

The financial report deals with the finance section of the company where the company wants to know about finance. The company has more concern about revenues, projected costs, Costs, Salary, Vacations, Employee tax, and Overtime.

There are two types of revenues one is invoice and second is other types of revenues. Which are representing that how much revenue generated in a company and from which sources? Invoice revenue shows that all revenue got from clients and their products but other revenue type shows all unknown sources from where company generates the revenue. The project costs represent the costs of every project that the company has been completed during selected period of dates. The salary parameter represents the salary of all the consultants who worked on the project. So it shows to the company that what was the salary amount and what was the price of the project. With this type of report the company can easily calculate profit and loss of the project. Then next step was that the company wants to know how much employee tax has been paid by the company so I introduced employee tax parameter which will calculate the employee tax according to his salary amount. The final and next step was that the company also wants to know about the overtime of every consultant. If somebody is sitting in office after evening then company should know about him and the company will pay him against his extra hours work.

User can achieve this entire functionality from statistics interface which I showed in figure 21. This designed implemented with the help of CSS, jQuery and jQuery Plug-ins. I modified HTML components like checkbox into image and I customized dropdown box according to my requirements as you can see in figure 21.

![Figure 21: The financial report](image)
6.2.2 Human Resource (HR) Report
The HR report belongs to Human Resource department which has responsibility to take care of all employees and providing new employees to the company. This report is very crucial and generated from the results of four modules of the company. As you can see in the figure 22 that it have first two inputs from form, thumbs modules and overtime of the employee that comes from financial section of report. The fourth parameter will be used as cumulative all these results.

![Figure 22: Human resource report](image)

6.2.3 Project Overview
The project overview reports section represent to the user that how many projects are going on within a company. This report has responsibility that it will show to the user the complete overview of all projects. It has five types of parameters. First one is active which show that currently how many projects are active. Second parameter is margin hours which mean that it will show the estimated remaining time of the projects. The third parameter is demography which represents the demography of all projects estimated time lines. How many projects are going to complete within estimated time line and how many are going to be delay. The fourth parameter represents the summary of all these three results at one place.

![Figure 23: The Project Overview](image)
6.2.4 Company Overview

The company overview report represents the overall status of the company. This report depends upon the results of the other reports like Human Resource and Project Report. The HR points represents that how many cumulative points he got from HR report project points. It means that how many projects have been completed successfully and how many have been delayed from deadlines. This report needs results from all other report so the user needs to generate all other report before generation of this report.

Figure 24: The Company Overview
7 Conclusion

The aim of this project was to re-design its architecture and enhance the functionality of the previous system with two modules namely; one was the “Survey Form” system and the second was “Statistics” module. I have successfully implemented the architecture which was highly scalable and flexible for integrating new modules in the system.

I have successfully implemented the two new modules in the system which were the survey form system and statistics module. The survey form had aim to create new surveys and send to the users. The users will send the response and this response will show to the graphs. The statistics module had aim to generate a four different types of reports which were belongs to the Financial, Human Resource, Project Overview and Company Overview reports.

8 Summary of Contribution

When I was analyzing and designing architecture for this BI system I learned many advance techniques. List of all learned techniques I have mentioned below.

1. Getting requirements from the customer about his problem.
2. Why we need the BI systems in this world and what its role in the company.
3. How to develop the BI systems.
4. Analyzing the complex problems and find a solution.
5. Designing the architecture for projects at professional level.
6. Find a best and economically cheap solution for the customer.
8. How I can work in a team.
9. How to work with the version control software.
10. Follow coding standards and writing clean and simple code.
11. How to work with MVC architecture web frameworks.
12. How to work with SQL language.
13. What is HTML, CSS, jQuery and what its role in front end of website.
14. How important to meet deadlines at professional level projects.

9 Future Suggestions

The current system has lack of Alert Systems and it needs also improvement in statistics module so I will suggest many things for future.

1. It needs fully functional Alert System.
2. Nice Graph Layout API that will present data into graph in a nice way.
3. Survey Form needs proper test cases for testing. We can use PHPUnit library for test cases.
Bibliography

[8] www.mysql.com
[12] NetBeans IDE. Application Development IDE,
[13] PhpMyAdmin. Administration of MySQL,
[16] HTML5. W3C,
[17] CSS. Cascading Style Sheet,
[18] JQuery. JQuery writes less, do more,
[20] ORM. Object Relation Mapping,