

# **EMGT 835 FIELD PROJECT:**

## ***Schedule Estimation for Web Application Development***

**By**

***Lake Fernando***

**Master of Science**

**The University of Kansas**

***Spring Semester 2008***

**An EMGT Field Project report submitted to the Engineering Management Program  
and the Faculty of the Graduate School of the University of Kansas in partial  
fulfillment of the requirements for the degree of Master of Science.**

---

<b>Chick Keller</b>	<b>Date</b>
<b>Committee Chair</b>	

---

<b>Annette Tetmeyer</b>	<b>Date</b>
<b>Committee Member</b>	

---

<b>Robert Zerwekh</b>	<b>Date</b>
<b>Committee Member</b>	

## **Copyright**

*All information, data, and software for this particular report are copyrighted by Lilekha (Lake) Fernando, 2008.*

# Table of Contents

<b>TABLE OF FIGURES .....</b>	<b>4</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>5</b>
<b>EXECUTIVE SUMMARY .....</b>	<b>6</b>
<b>ACRONYMS (OR ABBREVIATIONS).....</b>	<b>7</b>
<b>INTRODUCTION .....</b>	<b>10</b>
<b>LITERATURE REVIEW .....</b>	<b>13</b>
<b>PROCEDURE AND METHODOLOGY .....</b>	<b>32</b>
<b>RESULTS.....</b>	<b>36</b>
<b>CONCLUSION AND SUMMARY .....</b>	<b>40</b>
<b>SUGGESTIONS FOR ADDITIONAL WORK .....</b>	<b>42</b>
<b>REFERENCES .....</b>	<b>43</b>
<b>APPENDICES .....</b>	<b>45</b>

## Table of Figures

FIGURE 1 .....	14
FIGURE 2 .....	15
FIGURE 3 .....	16
FIGURE 4 .....	19
FIGURE 5 .....	20
FIGURE 6 .....	21
FIGURE 7 .....	23
FIGURE 8 .....	24
FIGURE 9 .....	30
FIGURE 10 .....	37
FIGURE 11 .....	37
FIGURE 12 .....	37
FIGURE 13 .....	37
FIGURE 14 .....	38
FIGURE 15 .....	45
FIGURE 16 .....	46
FIGURE 17 .....	47
FIGURE 18 .....	48
FIGURE 19 .....	49

## **Acknowledgements**

I want to take a moment to thank the Committee chair, Chick Keller, and committee members, Annette Tetmeyer and Dr. Robert Zerwekh, for the guidance and support that they provided for this particular project.

I also would like to thank everyone who for participating in the web application survey. Special thanks also go to Jason von Nieda who is hosting the web estimation web application for presentations and demonstrations.

Finally I would like to thank Kari Kemper, my fiancée, for the support she provides me and my parents Nelson and Sumithra Fernando for their guidance and support over the years.

## **Executive Summary**

The purpose of the project is to come up with a robust methodology for estimating hours for web application systems development. For the scope of the project, a web application system is an application which is accessed via a web browser and performs numerous tasks via web pages. The project will take a generalized approach for all web application systems.

Estimating the level of effort to complete any application system is a hard task. Web application systems are no different from any other application system when it comes to estimating development time. The main reasons for difficulty in estimating hours are the dynamic nature of the business and the customer's lack of requirements for the web application.

This particular project will generate a robust methodology to provide an accurate estimate for web application system development. The project will dissect a web application and take into account all the other pieces that are integrated into a web application. Then a survey is taken by numerous experienced software engineers who will estimate implementation hours for each task. The average is generated for each task. This average will be used as a bench mark to calculate estimates for a given project.

The final part of this project is the construction of a web application which determines estimates for other web applications. A user will have to know the requirements thoroughly and enter them into the estimation web application. Within a matter of seconds the estimating web application will calculate the estimate for the particular future web application.

## Acronyms (or Abbreviations)

Term/Phrase	Definition
<b>EMGT</b>	<b>Engineering Management</b>
<b>PK</b>	<b>Primary Key of a database table</b>
<b>FK</b>	<b>Foreign key of a database table, which refers to another table's PK</b>
<b>ERP</b>	<b>Enterprise Resource Planning</b>
<b>DAO</b>	<b>Data Access Object, which refers to a programming module which connects to a particular database and retrieves data, inserts rows, and updates rows</b>
<b>IT</b>	<b>Information technology</b>
<b>Project</b>	<b>Information technology project</b>
<b>API</b>	<b>Application Programmer Interface</b>
<b>Database Entity</b>	<b>A particular table in the database, i.e. supplier, employee</b>
<b>KPMG</b>	<b>Klynveld, Peat, Marvick, and Goerdeler</b>

<b>Term/Phrase</b>	<b>Definition</b>
<b>JSP</b>	<b>Java Server Pages API</b>
<b>Servlet</b>	<b>A Java Servlet is a web application component which is deployed in a web server and services web request from a browser.</b>
<b>HTML</b>	<b>HyperText Markup Language</b>
<b>JRE</b>	<b>Java Runtime Environment</b>
<b>Ant</b>	<b>Apache Ant is a software tool for automating software build processes</b>
<b>JUnit</b>	<b>Java unit testing API</b>
<b>CSS</b>	<b>Cascading Style Sheets</b>
<b>MVC</b>	<b>Model-View-Controller. MVC is an architectural pattern used in software development.</b>
<b>MIS</b>	<b>Management Information System</b>
<b>PMBok</b>	<b>Project Management Body of Knowledge</b>



Term/Phrase	Definition
<b>WBS</b>	<b>Work Breakdown Structure</b>
<b>COCOMO</b>	<b>Constructive Cost Model</b>
<b>PERT</b>	<b>Program Evaluation and Review Technique</b>

## Introduction

The three main reasons for IT project failure are over budget spending, not completing within the timeline and not fulfilling the requirements according to the specifications. The main reason for the first two project failure reasons is that the initial planned schedule estimates were incorrect. System Analysis and Design Methodologies 5<sup>th</sup> edition states poor estimating techniques as one of the causes of failed projects (Whitten, 2001). The book states that “Many systems analysts estimate by making a best-calculated estimate (jokingly referred to as a guesstimate) and then doubling that number. This is not a scientific approach” (Whitten, 2001).

Study statistics are staggering when it comes to the failure rate for IT projects. Below are a few examples:

- The Chaos Report (1995) results indicate that 52.7% of IT projects will cost over 189% of their original estimate (Statistics over IT...).
- Robbins-Gioia survey 2001 states that 51% respondents viewed their Enterprise Resource Planning (ERP) implementation project as unsuccessful (Statistics over IT...).
- The KPMG Canada survey (1997) states that 61% of projects were reported failed and over 50% of the projects were over budget by a significant amount (Statistics over IT...).
- The Bull Survey (1998) results indicate that 75% of the IT projects missed their deadlines and 55% of the IT projects exceeded the budget (Project Failure...).

- Carr (1994) states that only 16% of IT projects are completed on time, within budget and with full functionality (Aiken, 2005).

As illustrated by these statistics, better schedule estimation is of utmost importance when it comes to the success of an IT project. Companies and governments around the world spend billions of dollars trying to make up time within IT projects that are behind schedule. The resources spent on behind schedule projects can be utilized elsewhere if the initial estimates were accurate.

Within the IT industry it is extremely difficult to generalize a methodology for schedule estimation for all types of different projects. The scope of this project will focus on the schedule estimation of Web Applications. This will include the web front-end, the middle layer which includes the business logic or business rules, the database layer for persisting data, and the security layer for application security.

The project will dissect each step involving/involved in the development of a web application. Next a survey is built and responded to by numerous experienced software engineers who will estimate hours for all the granular steps. The engineers will estimate hours for low, mid, and high level of complexities for each step. These steps include granular parts in the web front end, the middle layer of business logic, the database layer for persistence, and finally the security framework. The averages are then generated. These averages will be used as a benchmark to calculate estimates for any future projects.

The final and the most important deliverable of this project is the construction of a web application which determines estimates for future web applications. These future web applications need to have well defined user requirements.

A software engineer will analyze these requirements and transform these user requirements into system requirements, and also generate the high level technical design for the project. Afterwards the system requirements and technical design is used to enter data into the estimation web application. Within a matter of seconds the estimation web application will calculate the estimate for the future web application. The future web application has to be a brand new application. This project does not apply to enhancements or maintenance work to existing web applications.

## Literature Review

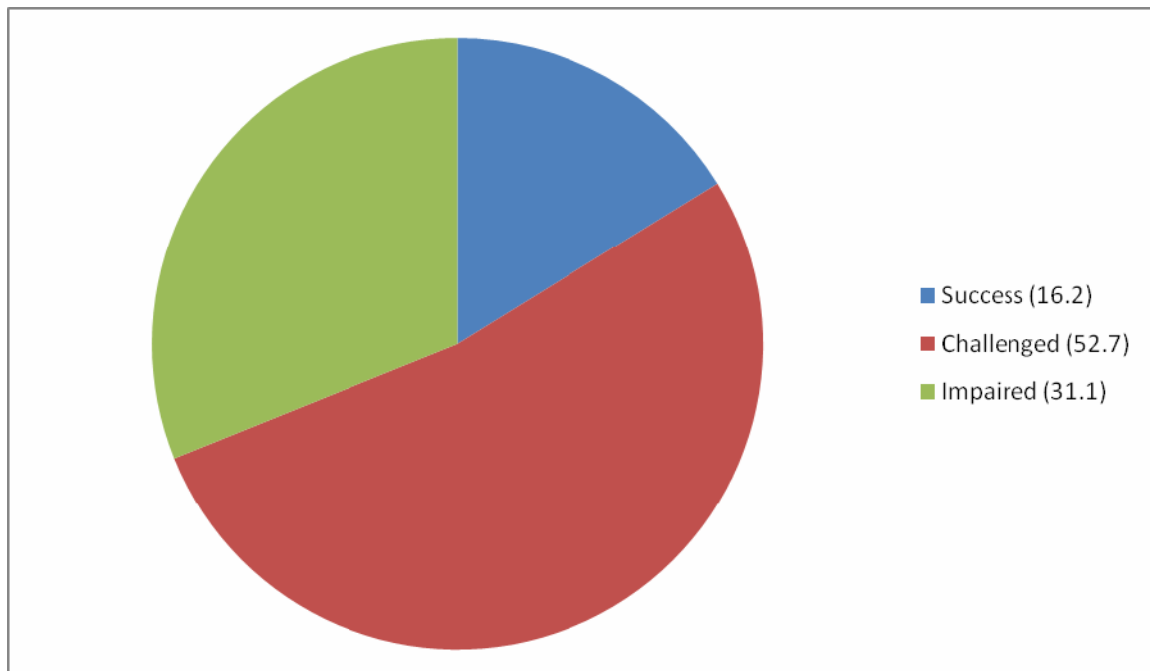
The surveys below were done over the years to verify IT project success rate by various companies around the world.

*The Chaos Report (The Standish Group International Inc. 1995)*

The Standish Group developed a survey to find key information within the IT industry. The survey was distributed among US companies with a Management Information System (MIS). The survey respondents were IT executive managers. The sample included small (US \$100 million to US\$ 200 million in yearly revenue), medium (US\$ 200 million to US\$ 500 million in yearly revenue), and large (greater than US\$ 500 million in yearly revenue) companies. The survey was conducted across industries, such as banking, insurance, manufacturing, health care, and local, state and federal organizations. The study included 365 respondents and 8,380 applications that were managed by the respondents. This particular study categorized each IT project into three types. They are:

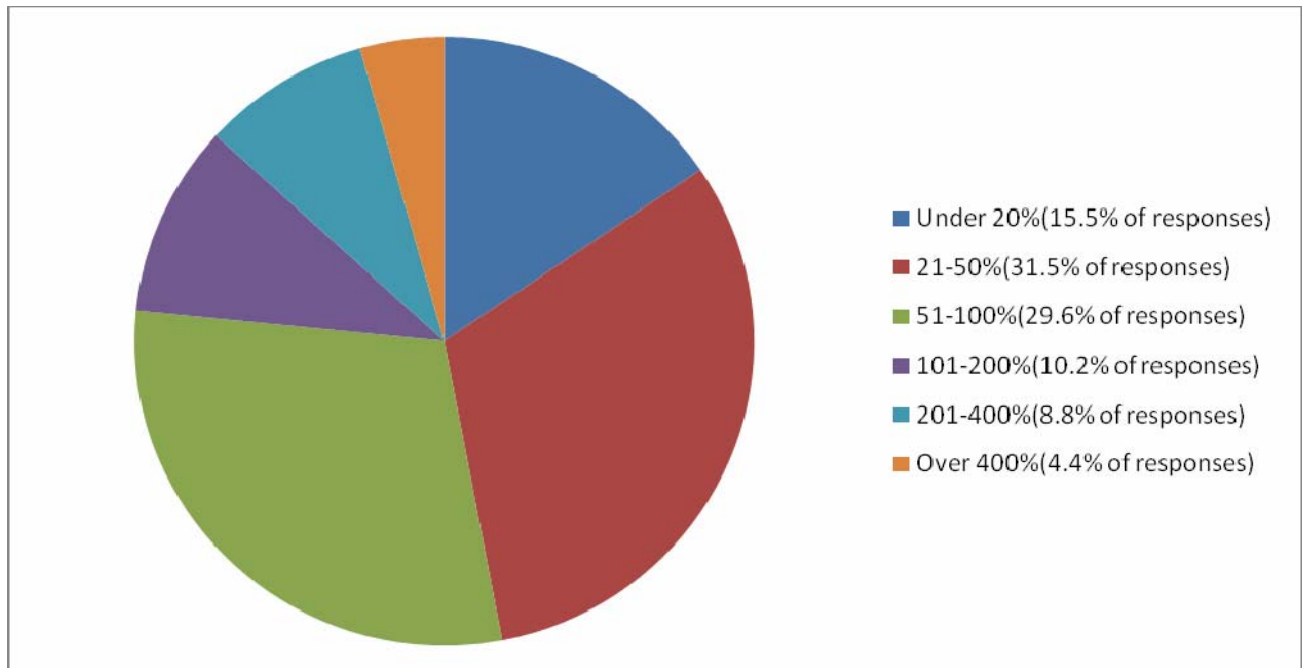
- Resolution Type 1 (project success) – The project was completed within budget, within the timeline and with all the functionality that was specified in the initial specifications.
- Resolution Type 2 (project challenged) – The project was completed; however it was over-budget, it took longer than the original timeline and with minimal functionality which was initially specified.
- Resolution Type 3 (project impaired) – The project was cancelled during the software development lifecycle.

The survey revealed that 16.2% of the projects were successful, 52.7% of the projects were challenged, and 31.1% of the projects were impaired. So 83.8% of the total projects were completed unsuccessfully. Every 4 out of 5 IT projects are failures, and thus only 1 out of 5 is considered a success. Figure 1 illustrates IT project success/failure rate within this survey.



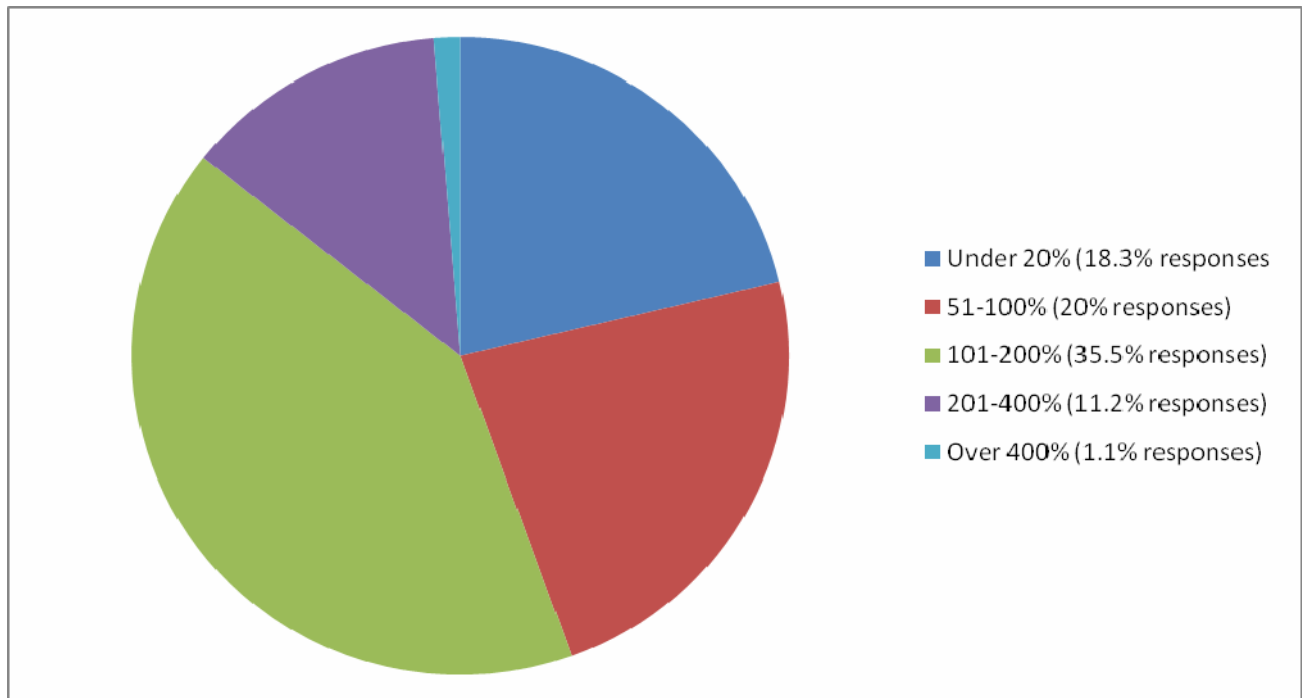
**Figure 1**

The study also revealed that cost overruns within challenged and impaired projects were staggering. The average cost overrun across the board was 189% of the original project cost estimate. The average cost overrun was 178% for large companies, 182% for medium companies, and 214% for small companies. Figure 2 illustrates the project overrun numbers for challenged and impaired projects.



**Figure 2**

The study revealed that for the same challenged and impaired projects, over one third also experienced time overruns of 200 to 300%. The average time overrun was 222% of the original estimated timeline. For larger companies, the average was 230% of the original timeline. For medium companies, the average was 202% of the original timeline. For small companies, the average was 239% of the original timeline. Figure 3 illustrates the numbers for time overrun projects for challenged and impaired projects.



**Figure 3**

*Robbins-Gioia survey (Robbins-Gioia Press Release, 2001)*

Robbins-Gioia conducted a survey which involved employee satisfaction regarding their ERP systems. There were 232 respondents spanning across many industries, such as information technology, communications, financial, utility, healthcare and government agencies. A total of 36% of the companies surveyed had or were in the process of implementing an ERP system for their company. 51% of the respondents expressed that their ERP implementation was unsuccessful.

"Looking at the current cost and schedule overruns associated with ERP implementations, as well as the number of implementations that are abandoned mid-stream, it is obvious that the IT world is missing a golden opportunity to deliver business value," said Gioia (2001). He also said that "ERP



implementation success is gauged by its ability to align IT and business management objectives, demanding program management skills and a refined process for success. Failure is not a given" (2001).

One reason for some of these projects to fail is that the initial estimates were incorrect. Down the road when a project is being implemented the budget overruns and the timeline is missed for delivery.

*The KPMG Canada Survey (Project Failure Statistics, 1997)*

KPMG performed a survey in Canada to outline the reasons behind the failure of information technology projects. The survey was issued to 1,450 leading Canadian companies in the private and public sector. Out of the 1,450 that were sent, 176 were analyzed. Down below are the key findings from the survey.

- Over 61 % of the analyzed projects had failed according to the respondents
- More than 75% of the projects went over their schedules by 30% or more
- More than 50% exceeded their budgets by a substantial margin
- An estimated \$25 billion is spent on IT application development in Canada annually
- Unbudgeted IT project expenditures are expected to run into the billions

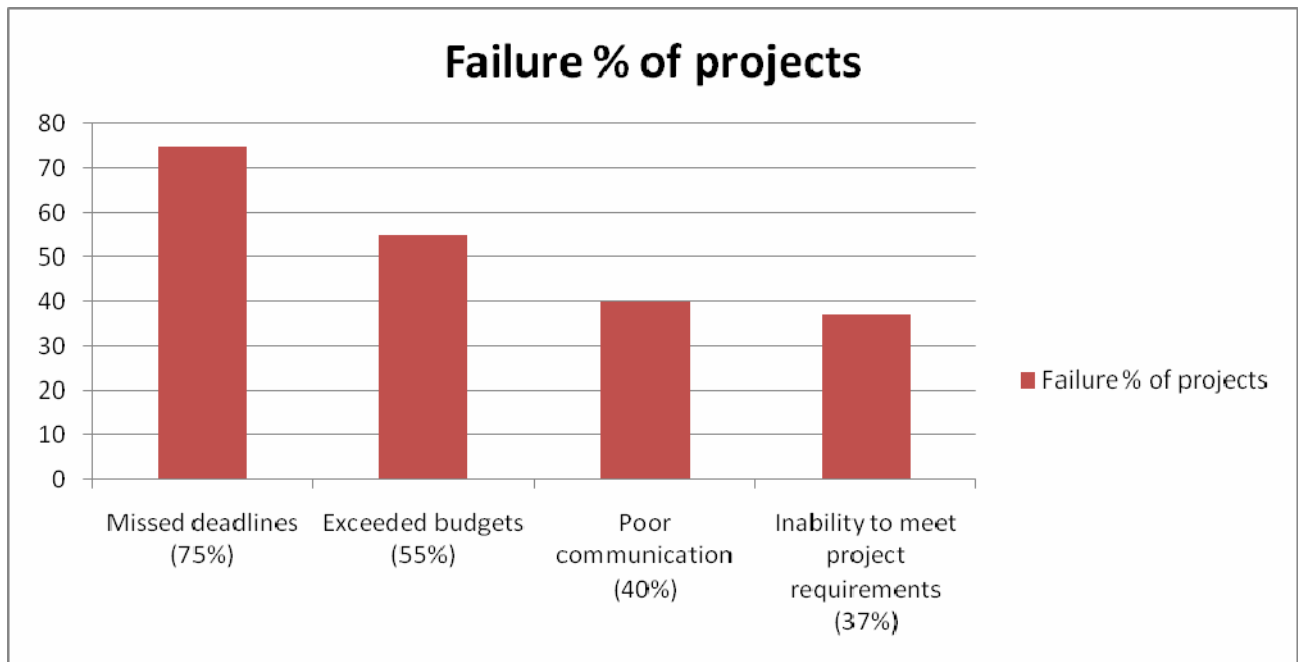
*The Bull Survey (Project Failure Statistics, 1998)*

The French computer manufacturer and systems integrator, BULL, requested an independent research company, Spikes Cavell, to conduct a survey in the UK to identify the major causes of IT project failures.

A total of 203 respondents were interviewed via the telephone to conduct this particular survey. The audience was IT managers and project managers from the finance, utility, manufacturing, business services, telecommunications, and IT services in the United Kingdom. The survey revealed the statistics below.

Main IT project failure criteria: (illustrated in figure 4)

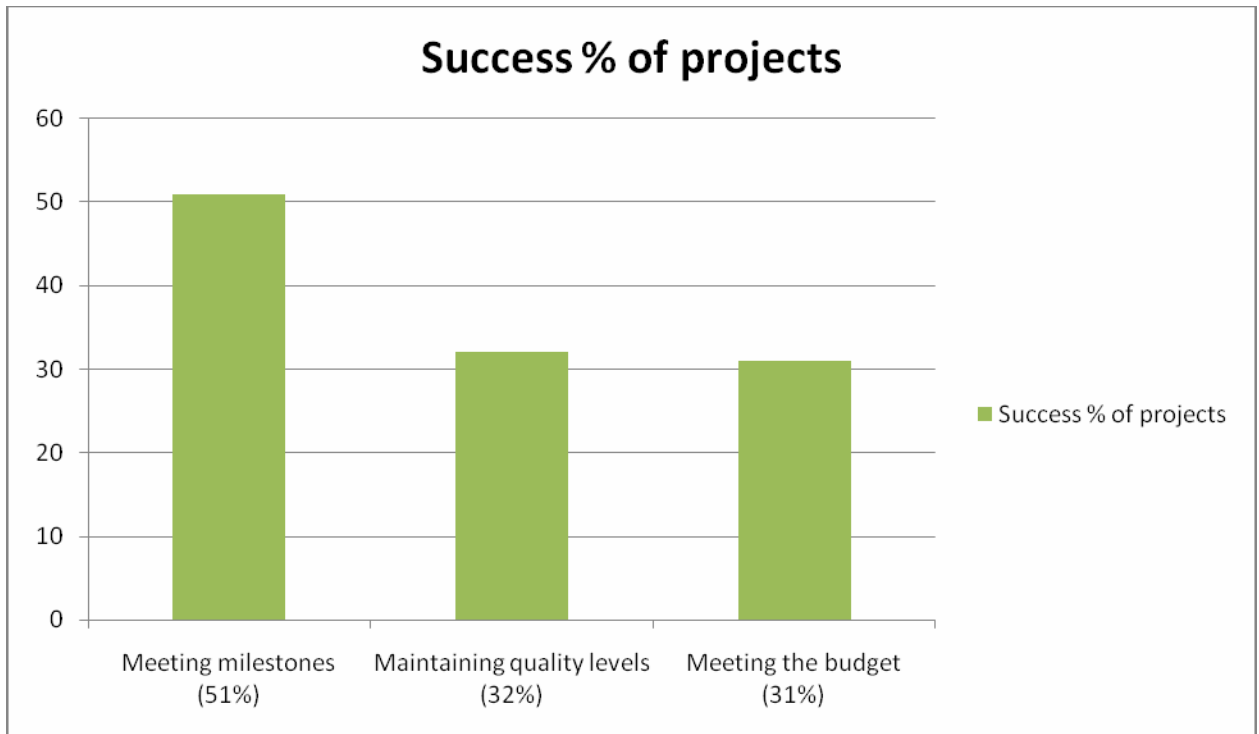
- Missed deadline (75%)
- Exceeded budgets (55%)
- Poor communication (40%)
- Inability to meet project requirements (37%)



**Figure 4**

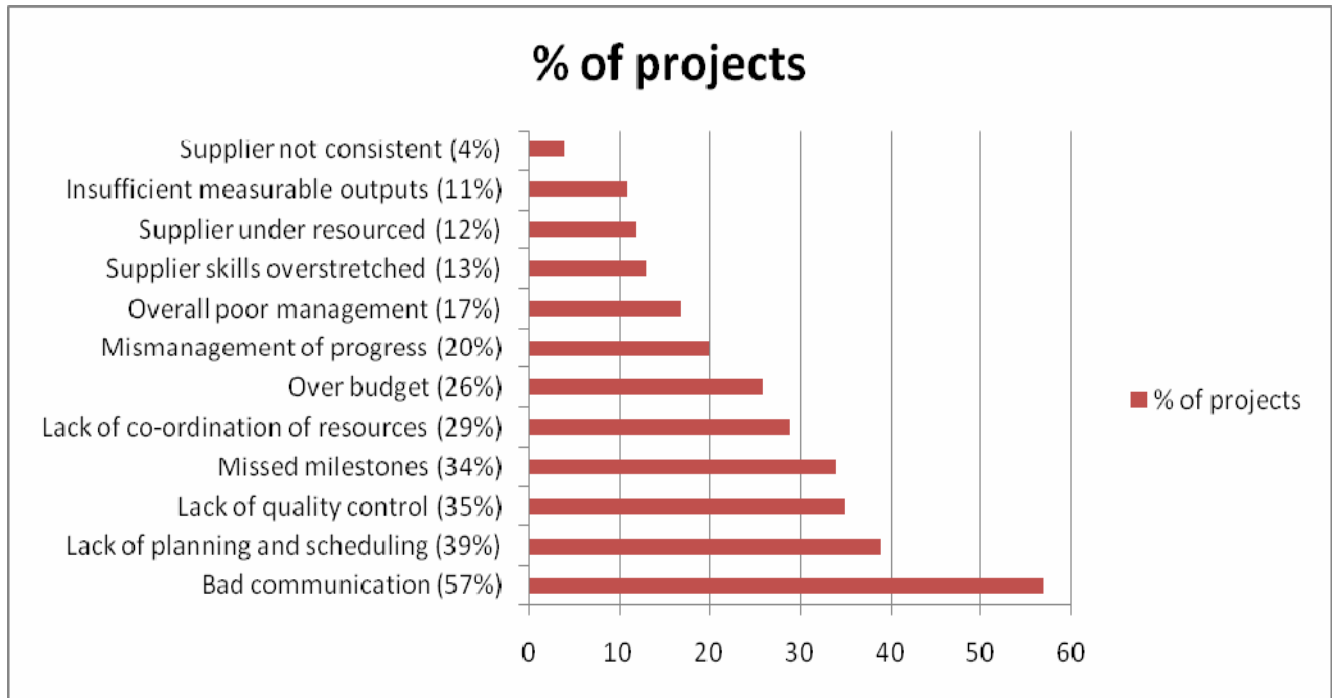
Main IT project success criteria: (illustrated in figure 5)

- Meeting milestones (51%)
- Maintaining the required quality levels (32%)
- Meeting the budget (31%)



**Figure 5**

Figure 6 illustrates the major causes of IT project failure during software development life cycle, as revealed from the Bull survey.



**Figure 6**

According to the Bull survey 39% of projects failed because of the lack of planning and scheduling. The root cause for the lack of planning and scheduling is the lack of calculating estimates at the beginning of the project before budget and timeline are defined. If there was a system which would calculate the estimates for a defined project (project with good requirements) 39% of the projects could have been successful given there were no other complications. Moreover, the 34% of the missed milestone projects and the 26% of over budget projects could have been successful if there were accurate estimates from the beginning of the project.

All the surveys above suggest that most IT projects are more likely to fail. There are more failures than successful projects according to all of the surveys above. The main reason for failing projects is the inability to estimate hours for

tasks by analyzing the requirements for applications. The estimates need to be calculated before the budget and the timelines are set for a particular project. As a matter of fact, budgets and timelines are driven by the estimates that are provided by the appropriate individuals. Accurate estimates are of utmost importance when it comes to software development. An IT project's success is dependent on accurate estimates upfront.

*Project Cost Management (Duncan, 1996)*

The Chapter 7 of the Project Management Body of Knowledge (PMBok) discusses the Project Cost Management. There are 4 components which are essential for the successful cost management of any project, and an IT project is no different. They are:

1. Resource Planning
2. Cost Estimating
3. Cost Budgeting
4. Cost Control

Figure 7 (Duncan, 1996) below provides the key components of the Project Cost Management and inputs, tools and techniques, and output for each component.

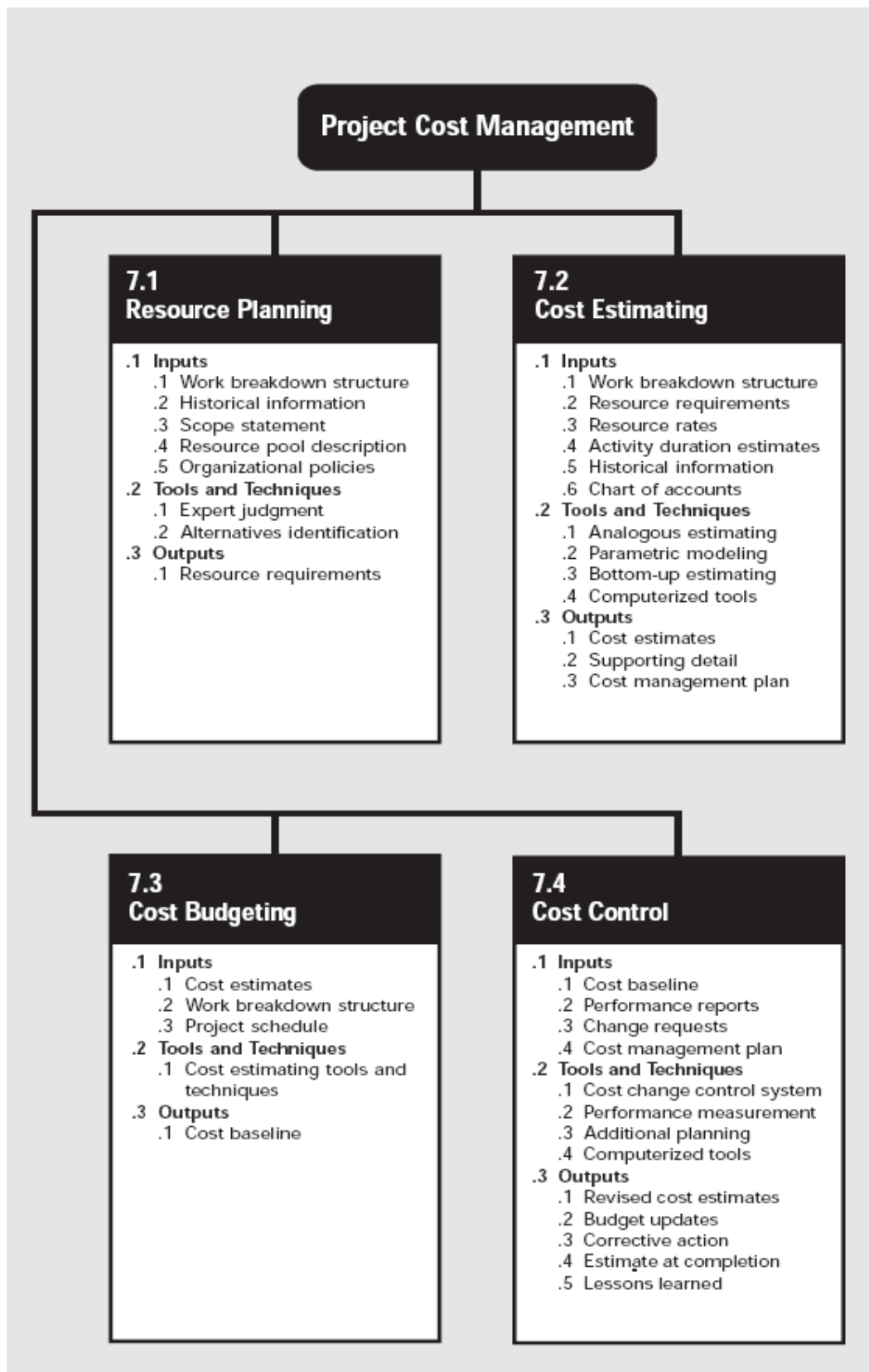
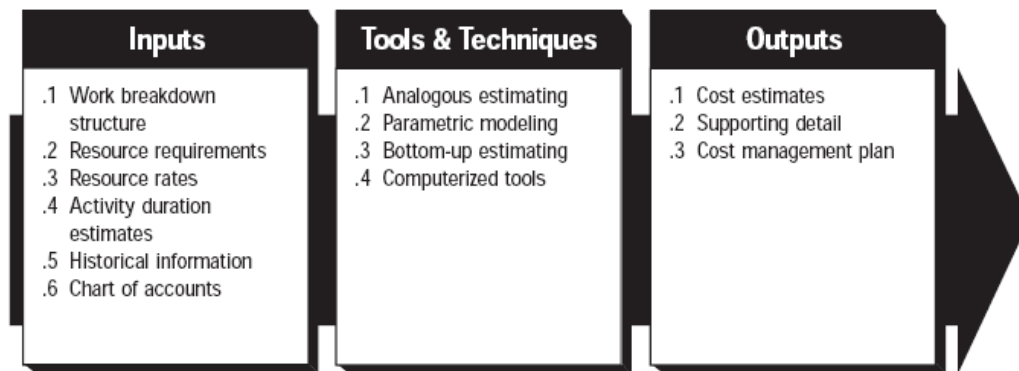


Figure 7

Cost estimating is the focus of this particular report. Cost estimating is also known as estimation of hours. Figure 8 (Duncan, 1996) illustrates how inputs are entered into certain tools and techniques and how outputs are generated from those tools and techniques in order to generate total estimates for the project.



**Figure 8**

Hours to complete a particular task is directly associated with the cost to complete that particular task. If it takes  $h$ , hours to complete a task, where the hourly cost is  $c$ , the total cost,  $tc$  can be calculated for this particular task like such,

$$tc = h * c$$

*Inputs to Cost Estimating (Duncan, 1996)*

1. Work Breakdown Structure (WBS) – Technique for defining and organizing the total scope of a project, using a hierarchical tree structure.
2. Resource Requirements – What types of resources are required and in what quantities for each element of the WBS.



3. Resource Rates – Unit rates, within an IT project hourly rate for a developer, database administrator etc.
4. Activity Duration Estimates – Involves assessing the number of work periods likely to be required to complete each identified activity.
5. Historical Information – Previous project team files (lessons learned), commercial cost estimating database of historical projects, and project team knowledge of previous projects.
6. Chart of Accounts – Coding structure used by the organization to report financial information.

#### *Tools and Techniques for Cost Estimating*

1. Analogous estimating (Duncan, 1996) – Analogous estimating is using estimations from a previous, similar project as the basis for the estimating for the current project. Analogous technique is used when limited information is known about the current project. The advantages of analogous estimating are it's less costly than other methods and its relatively simple. The disadvantage of analogous estimating is it is less accurate, because it is next to impossible to find two similar or identical projects. The definition of a project clearly defines that projects are unique.
2. Bottom-up estimating (Leung) – Bottom-up estimating is estimating the cost of individual work items, or granular tasks. Afterwards these individual estimates are summed up for the total estimate for the project. Bottom-up estimating is used when a project has well defined requirements. Bottom-up estimating also drives business owners of the project to generate well

- defined requirements. The main advantages of bottom-up are the accuracy and the ability to track individual work items explicitly. The main disadvantages are it is time consuming, expensive, and the inability to use on a project where less information is known regarding the project.
3. Top-down estimating (Leung) – This is the opposite of the bottom-up estimating. The cost estimates are generated by global properties. This methodology is best used at an early stage of a project. The main advantages are the ease of use, simplicity, cost effectiveness and quick response. The main disadvantage of this method is it is highly inaccurate.
  4. Parametric Modeling (Duncan, 1996) – Parametric modeling, also known as the algorithmic method using project parameters in a mathematical model to predict the costs. There are two types, simple and complex. Software development falls under the complex category, which has 13 separate adjustment factors each of which has 5-7 points on it. The costs are accurate when the historical information used to develop the model is accurate. The parameters used in the model are readily quantifiable, and the model is scalable. The main disadvantage is the varying accuracy of costs. Any parametric model has the form:  
$$\text{Effort} = f(x_1, x_2, \dots, x_n)$$
 where  $(x_1, x_2, \dots, x_n)$  denotes the cost factors (Leung). Constructive Cost Model (COCOMO) is a form of parametric modeling, which is widely, accepted in the industry however it can be extremely complicated to use with IT projects.

5. Computerized Tools (Duncan, 1996) – Computerized tools such as project management software and spread sheets are used to assist with cost estimating. The main advantages are the ease of use and cost effectiveness. The main disadvantage is the inaccuracy of the estimates. An example of this methodology is the Program Evaluation and Review Technique (PERT) analysis. Here is an example estimation methodology taken by System Analysis and Design Methods (Whitten, 2001). Assume that one is estimating effort for task T1.

- Estimate the minimum amount of time it would take to complete task T1, and call this the Optimistic Duration. (OD) This estimate assumes that everything goes according to plan and no interruptions are going to affect the duration.
- Estimate the maximum amount of time it would take to complete task T1, and call this the Pessimistic Duration. (PD) This estimate assumes that everything can go wrong and everything will affect the duration.
- Estimate the expected duration (ED) that will take to complete the task T1. This estimate reflects more the realistic approach.
- Calculate the Duration (D) like such,

$$D = (OD + (4 * ED) + PD) / 6$$

6. Expert Judgment – Expert judgment uses previous project lessons learned along with experiences of the individuals who were a part of previous projects. Expert judgment is the most widely used estimating methodology in the world today for software development. The advantages of this

methodology are the accurateness of the estimates when the correct resources are used, and the cost effective nature. The disadvantages are the company needs historical information and subject matter experts in the industry with the appropriate experience generate estimates using this methodology.

7. Parkinson (Leung) – Parkinson's principle was "work expands to fill the available volume", applied to a software project, the cost is determined by the resources available for that particular project. The project requirements are not considered as a part of the estimate. The main advantages are the speed and ease of creating the estimate. The main disadvantage is the inaccurate nature of the estimates.
8. Price-to-win (Leung) – The cost is estimated to win the project for the company. The estimate is based on the customer's budget and not the functionality of the project. The only advantage is that the company wins the project. The disadvantages are the estimates are inaccurate and the project most likely will end up being late and unsuccessful. A project which was initiated with a price-to-win methodology will often require the project team to work overtime.
9. "Guestimates" – The cost estimates are based on guesses by software engineers, business analysts, or any other project members. The only advantage is that it is quick and simple. The main disadvantage is that it is highly inaccurate. Unfortunately this method is used extensively in the IT

industry resulting in many unsuccessful projects. This method should never be used in estimating costs for IT projects.

*Outputs from Cost Estimating (Duncan, 1996)*

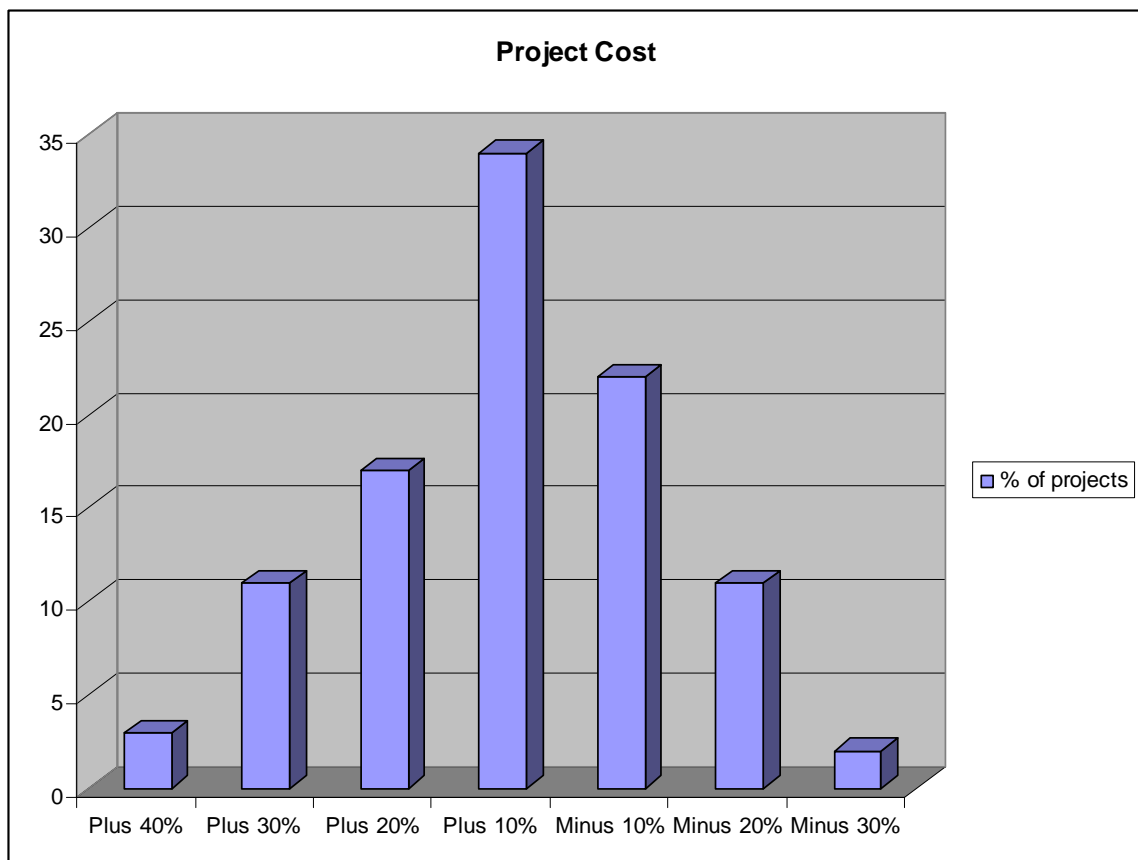
1. Cost estimates – Likely costs of the resources required to complete all the tasks within the WBS. Cost estimates could be in summary or in detail.
2. Supporting Details – Should include the WBS, documentation of the methodology used for the basis of the estimate, documentation of any assumptions made, and an indication of range of results, a plus, minus range. E.g. \$100,000 + or – 10% or \$10,000. So the cost range for this example project is \$90,000 - \$110,000.
3. Cost Management Plan – Describes how variances will be managed within the duration of the project. A cost management plan could be formal or informal, detailed or generalized depending on the requirement of the stakeholders.

*Cost estimating at xxxx.*

xxxx is an independent subsidiary of xxxx Software. xxxx provides end-to-end xxx consulting solutions and services for communication, financial, technology, healthcare and government industries. The development team at xxxx is about 20 developers and the implementation team is about six application consultants.

xxxx mainly uses the expert judgment methodology to estimate costs for client CRM solutions. This methodology has worked for xxxx because the company has the subject matter experts when it comes to xxx system solutions.

The experts at xxxx have the experience with previous customizations and installations with xxx systems. However within the last year, no estimate has been 100% accurate for all projects. Within the last year, all estimates provided by xxxx have fallen within the + 40% and – 30% range which is extremely efficient considering the inaccurate estimates within the IT industry. The figure below provides project cost estimates within the last year at xxxx.



**Figure 9**

However a more efficient cost estimating system is required for xxxx to be a market leader. Let's look at how cost estimates that are too high or too low impact a business such as xxxx. If the estimates are too low (plus % projects),

xxxx absorbs the additional costs. If the estimates are too high (minus % projects), the current resources are not utilized efficiently and projects are turned down because of the unavailability of resources. Turning projects down translates to lost revenue. So a business gets penalized for low cost estimates as well as high cost estimates.

## Procedure and Methodology

### *Survey Procedure - Web Application Estimation Survey*

A typical web application was dissected into each step involving in the development of a web application. This information was used to compile a survey which collected implementation estimates for each granular step. The survey consists of four parts. They are web frontend, middle layer (business logic), database layer, and security layer. The survey requires a software engineer to estimate the total time to complete a particular task, this includes development, unit test and integration test efforts. For each the survey question low, mid and high complexity level estimates were collected.

Survey questions for the web frontend

Implement a static field in a particular web page?
Implement a dynamic field in a particular web page? (dynamic e.g. drop down menu from the db)
Implement a validation rule in a static field?
Implement a validation rule in a dynamic field? (dynamic e.g. need to go to the db to verify an input)
Implement a submit action from one page to another?
If there was more than one locale (e.g. English and Spanish) what % of time would you add to each of the tasks above? (Take more of a general



approach, there is no low, mid, high, just
--

Survey questions for the middle layer (business logic)

Implement a static business rule?
-----------------------------------

Implement a dynamic business rule?
------------------------------------

Implement a manager/session bean for a group of entities? (consider an average, say 5 entities)
---

Survey questions for the database layer

How much time will it take to design a database with 10 tables?
---

How much time will it take to design a database with less than 30 tables but greater than 10 tables?
--

How much time will it take to design a database with 30 plus tables?
--

Implement a DAO for a table with insert, delete, update for all fields except PK? (low = 5 fields no FK, mid = 10 fields 1 FK, high = 10+ fields with 2+ FK)
--

Survey questions for the security framework

Implement a general security framework? (this includes analysis and design work as well)
--

Implement a new role into the security framework?
---

How much time is it going to take to integrate this security framework into a standalone web application?
---

### *Computational Methods*

The average estimate time for each granular task is then calculated from all the respondents. For an example if there were n responses for task xyz with estimates e1 through en, the average (avg) is calculated like such,

$$\text{avg} = (e1 + e2 + e3 + ..... + en)/n$$

For a concrete example if the estimates were 2, 4, 5, 3 hours for 4 responses, the average estimate would be,

$$(2 + 4 + 5 + 3)/4 = 3.5 \text{ hrs}$$

Afterwards these average estimates are used as a baseline in the implementation of web application estimation application.

### *Ultimate Cost Estimating Methodology*

The methodology that's used within this report is a hybrid of the expert judgment and the bottom-up methodologies. By breaking down a web application into its granular tasks fulfills two important aspects of estimating. They are:

- Requirements for each screen is well defined
- Makes it easy to perform the bottom-up estimate

The expert judgment dimension comes with the subject matter experts that responded to the survey. The expert judgment aspect can never be put on paper, it comes with experience, and it comes with years and years of implementing similar programming tasks. There is no methodology in paper or a fancy equation for the expert judgment; it all lies within the expertise of an experienced, professional software engineer. Down below are the advantages of using the proposed methodology for cost estimating.

- Don't require subject matter experts to perform the estimate. A business analyst and a novice software engineer can generate the cost estimates.
- Drives the business to deliver well defined requirements for each screen within the web application
- Simple and considerably cost effective than purchasing software estimating tool.
- Since expert judgment and bottom-up methodologies both are extremely accurate, this methodology is safely assumed to be accurate.

## Results

### *Survey Results*

The survey had 12 respondents whose average web application development experience was seven years. The respondents were extremely experienced and had worked on every industry from financial, health care, telecommunication, entertainment, insurance, banking, non-profit, sporting, legal, agricultural, transportation, manufacturing, construction, utilities, architectural, and government agencies. The bottom line is these software engineers had experienced and had worked on pretty much every type of web application relating to all industries. So the estimates which they have provided are very accurate to the actual time which it will take to implement each granular task.

The survey results were used to generate the averages for each granular task. The averages are down below for the four main sections of a web application; web frontend, business logic layer, security framework and the database layer. Note that the DAO is a part of the database layer.

### *Survey result averages for the web front end*

	Complexity low hours	Complexity medium hours	Complexity high hours
static field	x	x	x
dynamic field	x	x	x
static validation rule	x	x	x
dynamic validation rule	x	x	x
submit action	x	x	x

**Figure 10**

*Survey result averages for business logic layer*

**Figure 11**

*Survey result averages for security framework*

Task	Complexity low hours	Complexity medium hours	Complexity high hours
security framework	x	x	x
add role to framework	x	x	x
integrate to existing app	x	x	x

**Figure 12**

*Survey result averages for DAO development*

**Figure 13**

*Survey result averages for database layer*

**Figure 14**

*Estimation Web Application*

These estimates were used to generate the estimation web application. Here are the steps within the estimation application in order to get an accurate estimate for a future web application.

1. Enter the number of screens and locales for the application. Figure 12
2. For each screen define the input fields, validation rules, navigation actions etc. Figure 13
3. Define business rules behind each screen. Figure 14
4. Define the security requirements of the application. Figure 15
5. Define the database structure and requirements. Figure 15
6. The total estimate will be displayed in the following screen. Figure 16

The following technologies and software was used in the development of the estimation web application. The application also followed Model-View-Controller (MVC) design pattern.

- Java 5 Standard Edition – Java SE
- Java 5 Enterprise Edition – Java EE
- JRE version 5
- JSP version 2.0
- Servlet version 2.5
- Struts version 1.2.9
- Apache Tomcat Web Server
- HTML

- JavaScript
- Ant
- JUnit
- CSS

## Conclusion and Summary

More than 50% of the IT projects end up being failures in every industry worldwide. One of the main reasons for these failures is poor estimation methods up front. One could make an educated assumption that accurate estimates leads to successful projects, thus making estimation an utmost important activity for any IT project.

If the requirements of an IT project are static and don't change within the duration of the project, it has a better chance of succeeding with accurate initial estimates. On the other hand, if the requirements are dynamic and changes within the duration of the IT project, it becomes harder to follow the original estimates and there is a higher probability of the project failing.

One could use the estimation web application to estimate hours for future web applications. It would be interesting to see how it would vary from the actual hours.

When a company initiates the process of starting a web application, the requirements need to be finalized and documented. Then a software engineer needs to analyze the requirements and generate the system requirements and the functional design for the web application. Then the estimates need to be generated for each task. This is the most important aspect of the project, and the project success depends on this activity. Afterwards the budget and timeline needs to be set according to the estimates and the availability of resources for the project. With accurate estimates, accurate budget and timelines will be set.



Thus the project will have a very high probability to succeed in a marketplace where more than 50% of the projects fail.

## Suggestions for Additional Work

It would be interesting to investigate the estimates if development was done using a web application framework such as [Struts](#), [JSF](#), or [Spring](#). Also the estimates would be somewhat different if a persistent framework such as [Hibernate](#) was used to develop the database layer. Web application frameworks and persistent frameworks cuts down the development efforts of an IT project significantly.

It would be an interesting to compare the actual hours of IT projects to estimates that were generated by the estimation web application in the real world.

## References

- Whitten, Jeffery L., Bentley, Lonnie D., Dittman, Kevin C., *System analysis and design methods 5<sup>th</sup> edition*, McGraw-Hill, 2001,  
[www.mhhe.com/whitten](http://www.mhhe.com/whitten)
- Project Failure Statistics –  
[http://www.it-cortex.com/Stat\\_Failure\\_Cause.htm](http://www.it-cortex.com/Stat_Failure_Cause.htm)
- Statistics over IT Failure Rate –  
[http://www.it-cortex.com/Stat\\_Failure\\_Rate.htm](http://www.it-cortex.com/Stat_Failure_Rate.htm)
- Chaos Report -  
[http://www3.uta.edu/faculty/reyes/teaching/general\\_presentations/chaos1994.pdf](http://www3.uta.edu/faculty/reyes/teaching/general_presentations/chaos1994.pdf)
- Robbins-Gioia Press Release -  
[http://www.robbinsgioia.com/news\\_events/012802\\_erp.aspx](http://www.robbinsgioia.com/news_events/012802_erp.aspx)
- <http://dandriani.wordpress.com/2007/12/06/statistics-on-it-projects-failure-rate/>
- Longstreet, David, <http://www.softwaremetrics.com>
- <http://en.wikipedia.org/wiki/KPMG>
- [http://en.wikipedia.org/wiki/Apache\\_Ant](http://en.wikipedia.org/wiki/Apache_Ant)
- <http://en.wikipedia.org/wiki/Model-view-controller>
- Duncan, William, A Guide to the Project Management Body of Knowledge, 1996 - <http://www.di.unipi.it/~macri/pmbok.pdf>
- Leung, Hareton, Fan, Zhang, Software Cost Estimation,  
<http://www.st.cs.uni-sb.de/edu/empirical-se/2006/PDFs/leung.pdf>

- Aiken, Peter, 2005 -

<http://www.dama-ncr.org/Library/2006-02-15PeterAikinXML/x1.pdf>

# Appendices

## Estimation Web Application Screen Shots

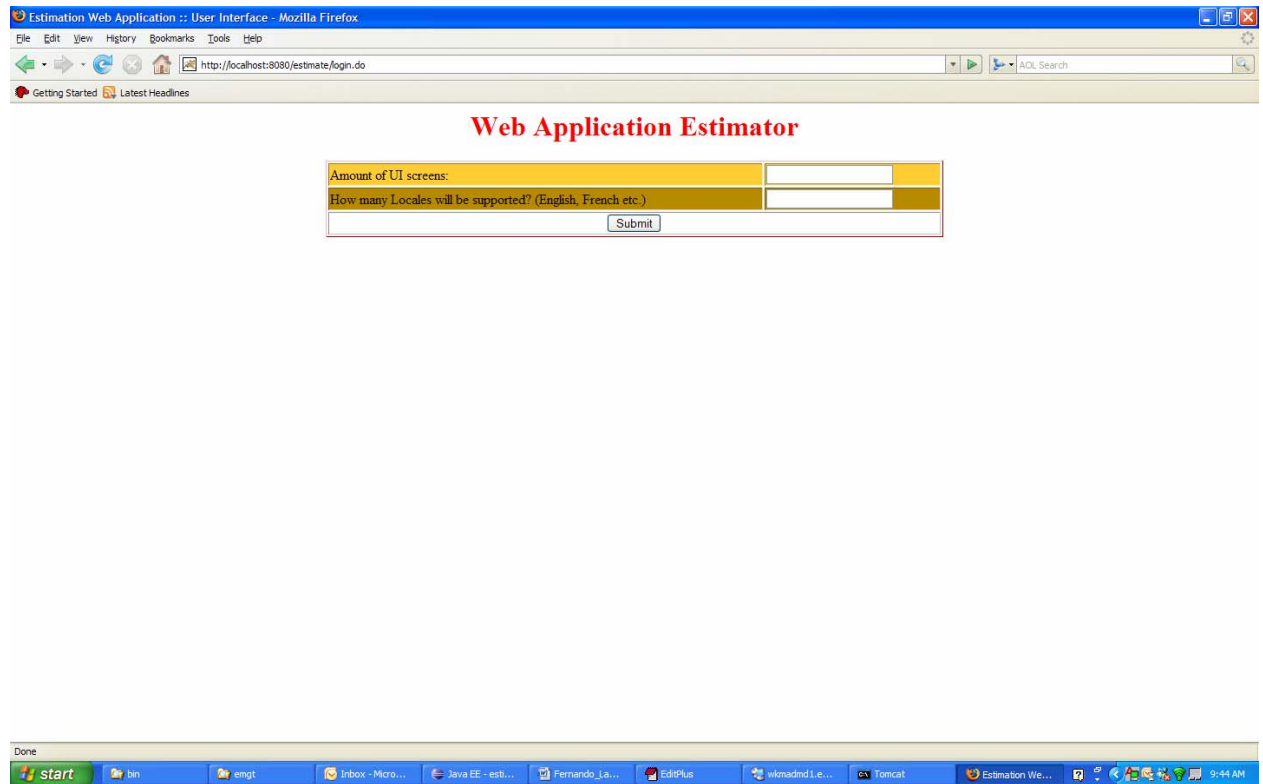


Figure 15

Estimation Web Application :: Frontend - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://localhost:8080/estimate/ui.do?upages=2&locale=en

Getting Started Latest Headlines

## Web Application Estimator

**For each UI screen please enter the following data  
Complexity (low, mid, high)**

**Enter appropriate inputs for screen 1**

Amount of static fields:	1	1	1
Amount of dynamic fields:	1	1	1
Amount of static validation rules for all fields:	1	1	1
Amount of dynamic validation rules for all fields:	1	1	1
Amount of navigation actions for this particular screen:	1	1	1

**Enter appropriate inputs for screen 2**

Amount of static fields:	1	1	1
Amount of dynamic fields:	1	1	1
Amount of static validation rules for all fields:	1	1	1
Amount of dynamic validation rules for all fields:	1	1	1
Amount of navigation actions for this particular screen:	1	1	1

Done

start Java EE - estimate/w... bin Estimation Web Appl... Tomcat Document1 - Microsof...

8:05 PM

**Figure 16**

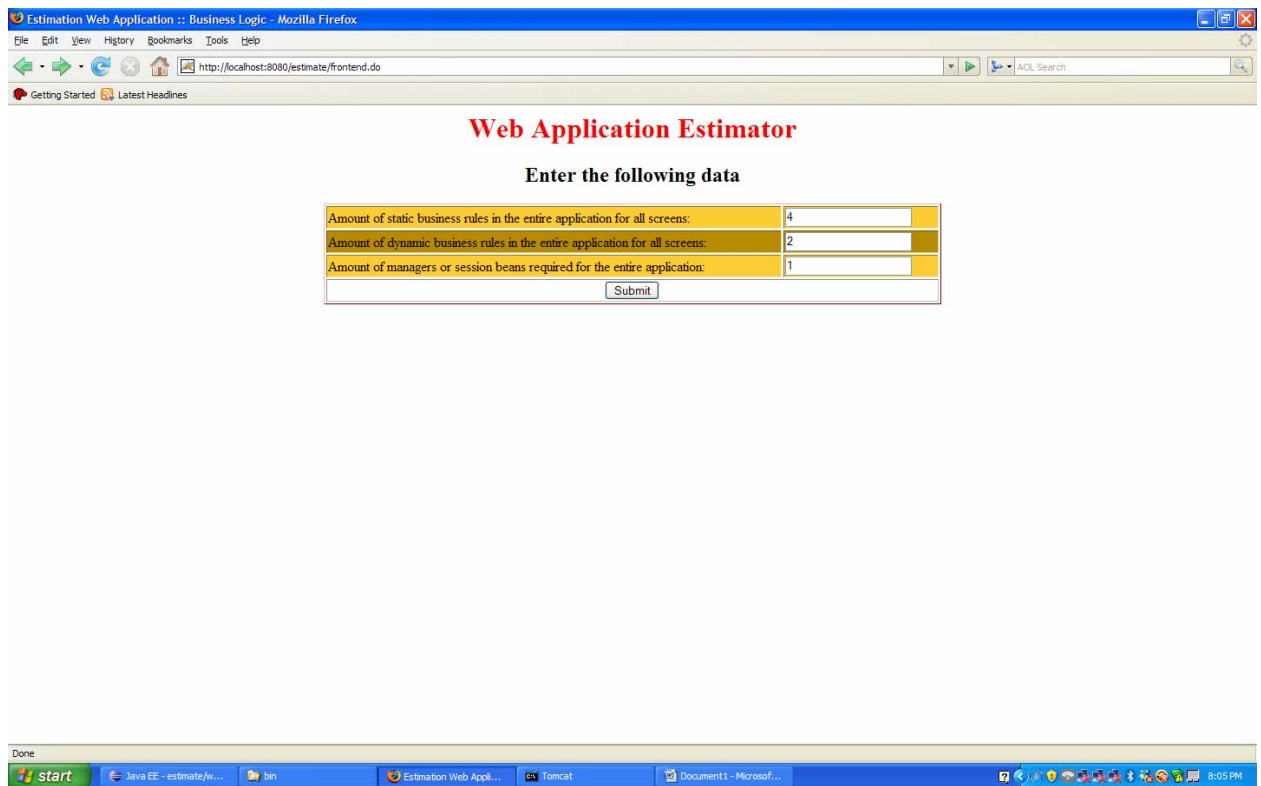


Figure 17

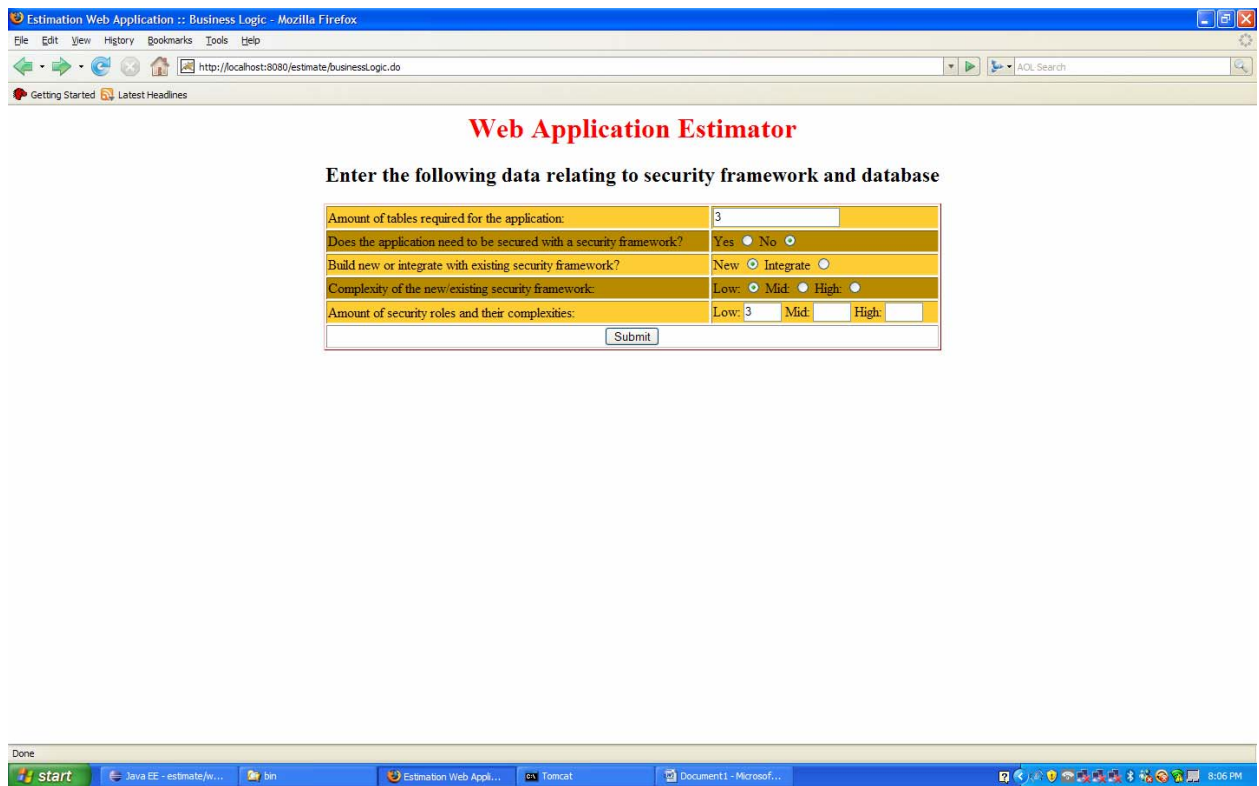
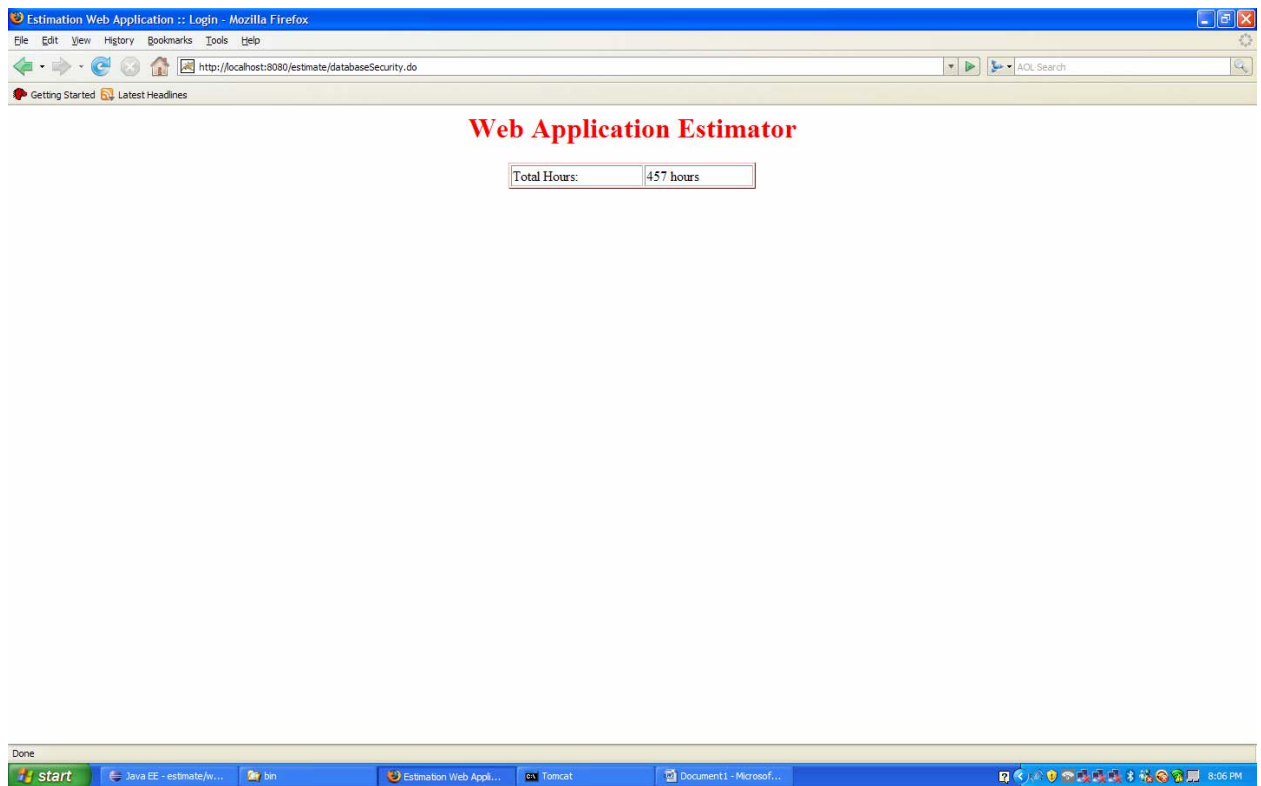


Figure 18





**Figure 19**