# Project Management
# Project Report

Chris O'Brien
*0144266*
*Sections 6, 7, 8*

Shane Whelan
*09005763*
*Sections 2, 3*

Brian O'Donoghue
*10104038*
*Sections 4, 5*

May 1, 2014

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

The project chosen for this report is the creation of a web application for the Monaleen Tennis Club. This is a club within the Castletroy area, near to the University. The club uses an existing paper based system, and the goal of this report is to establish the process necessary to manage a project to provide a software solution to the organisation.

The project group consisted of Chris O'Brien, Project Manager, with Shane Whelan and Brian O'Donoghue. We examined the process as detailed by the Project Management Body of Knowledge (PMBOK).

We focused on the following areas:

1. Project Integration
2. Project Scope Management
3. Project Time Management
4. Project Cost Management
5. Project Quality Management
6. Project Risk Management
7. Project Human Resource Management

Microsoft Project, Agena Risk and Cocomo were used to manage project planning, risk, and cost.

# Chapter 2

# Project Integration

Project Integration Management includes the processes and activities to identify, define, combine, unify, and coordinate the various processes and project management activities within the Project Management Process Groups. In the project management context, integration includes characteristics of unification, consolidation, communication, and integrative actions that are crucial to controlled project execution through completion, successfully managing stakeholder expectations, and meeting requirements. (PMBoK 2013)

## 2.1 Develop Project Charter

The project charter is in basic terms similar to a contract for a project, however, unlike a contract there is no financial data in the charter. It is formal acknowledgement of the project allowing an organisation to incorporate the project into their plans. At this point a statement of work has been submitted as well as a business case describing the tennis club project, these are both created by the project sponsor which in this case is the tennis club associated with the project. The project charter is created by the tennis club in collaboration with our project manager, this ensures that the goals set are realistic and can be completed by our company. We have created a formal relationship with the club and now that a charter has been created we can move on to planning the project. Using this project charter we can ensure the product ships on time. We will also use organisational templates from the organisational process assets inputted to the charter to create a template for the project. The project charter is as follows:

1. **Project purpose or justification**
    - The project is to build a tennis club website for a local club.
2. **Measurable project objectives and related success criteria**

- A finished product that is usable and reliable for club members and is better than current paper based system.

3. **High-level requirements**

   - General tennis club website for hosing club details and contact information.
   - Member system for registering and paying fees
   - System to book courts
   - Tournament automation
   - News blog

4. **Assumptions and constraints**

   - The stakeholder assumes that developers understand the domain of club websites and can complete the functionality required. Stakeholder also assumes that the company can provide future contract work to update the server operating system, middle ware and make any updates that are required to the core product.
   - Constraints are a limited budget as this is a community club and not a full time company.

5. **High-level risks**

   - In the project high level risks are security and data protection. If the club members personal data, passwords or payment information was compromised it could risk the integrity of the club.

6. **Summary milestone schedule** Project milestones include:

   - Finishing the planning stage of the project (1 week)
   - Allocating resources to the project (1 day)
   - Development of the project (6 months)
   - Testing (1 month)
   - Deployment (1 week)

7. **Summary budget**

   - The total budget for the project is 50,000

8. **Stakeholder list**

   - Tennis club committee

- Tennis club members
- Project manager
- Developers

9. **Project approval requirements**
    - The project will be deemed a success if the website is operational for one month without issue. In general terms the tennis club makes the decision on if the project is a success and they sign off on project completion.
10. **Assigned project manager, responsibility, and authority level**
    - Joe Blogs will be the project manager who assumes responsibility for this project in total. He will manage the test and development teams and report their progress to stakeholders.
11. **Name and authority of the sponsor or other person(s) authorizing the project charter**
    - John Smith, Club captain.

### 2.1.1 Develop Project Management Plan

The project management plan defines how the project is executed, monitored and controlled, and closed. The project management plans content varies depending upon the application area and complexity of the project (PMBoK 2013).

This overall document serves as a project management plan.

### 2.1.2 Directing and Managing Project Work

Direct and Manage Project Work is the process of leading and performing the work defined in the project management plan and implementing approved changes to achieve the projects objectives (PMBoK 2013). Obviously this process is managed by the project manager.

Once the project plan is complete the concrete work on the project can begin. Directing and managing the project work allows for corrective and preventive action as well as defect repair. The project management information system that will be used in the project is Microsoft Project. Atlassian Jira will also be used to track tasks, defects and blocking issues during project construction.

### 2.1.3 Monitoring and Controlling Project Work

Monitor and Control Project Work is the process of tracking, reviewing, and reporting the progress to meet the performance objectives defined in the project management plan. The key benefit of this process is that it allows stakeholders to understand the current state of the project, the steps taken, and budget, schedule, and scope forecasts (PMBoK 2013). The tools established in section 2.3 will be used to monitor and control the project, as well as the following techniques

1. Regression analysis
2. Causal analysis
3. Root cause analysis
4. Reserve analysis
5. Trend analysis
6. Earned value management
7. Variance analysis

Weekly meetings discussed elsewhere in this project management plan will be held to monitor and control the project.

### 2.1.4 Integrated Change Control

Perform Integrated Change Control is the process of reviewing all change requests; approving changes and managing changes to deliverables, organizational process assets, project documents, and the project management plan; and communicating their disposition. It reviews all requests for changes or modifications to project documents, deliverables, baselines, or the project management plan and approves or rejects the changes. The key benefit of this process is that it allows for documented changes within the project to be considered in an integrated fashion while reducing project risk, which often arises from changes made without consideration to the overall project objectives or plans (PMBoK 2013).

Changes to the project scope or definition can be made through the project manager who will approve the change with the relevant stakeholder in stakeholder meetings, thereby mitigating risk. The changes to the project scope will be documented in change control tools so that there is an audit trail for changes made to the project whether the changes are accepted or not.

### 2.1.5 Close

Close Project or Phase is the process of finalizing all activities across all of the Project Management Process Groups to formally complete the project or phase. The key benefit of this process is that it provides lessons learned, the formal ending of project work, and the release of organization resources to pursue new endeavours (PMBoK 2013).

The project manager will complete a review of the tennis club website and declare it completed or not. A scope baseline review will also be completed. Once all features are completed the project will be handed over to the deployment team who will deploy the web application to a web server and test it for functionality and load balancing.

# Chapter 3

# Project Scope Management

The Tennis Club Web Service (TCWS) project will provide clients and customers with the features that they need in the daily running of the club. As we are dealing with personal information of club members including billing data, security is one of the main non-functional requirements (or quality requirements as Lauesen titled them) of the web site. The project will use Spring Framework, Hibernate, Tomcat and Apache tiles to provide the service. Scope will be validated with the relevant stakeholders when each module and milestone is completed.

Once a scope management plan is defined we can create a WBS for the project. We can also define how formal acceptance of complete project tasks can be verified.

## 3.1 Plan Scope Management

In this phase of the project we will create a scope management plan and a requirements management plan. The scope management plan is a component of the project or program management plan that describes how the scope will be defined, developed, monitored, controlled, and verified (PMBoK 2013). One the project charter is defined and analysed we can begin to develop the scope management plan.

Firstly project scope will be defined with a scope management plan which is defined at project initiation. Project scope will then be controlled by project managers in meetings with development teams, where the scope will be clearly outlined to employees. Meetings with management will be arranged weekly to monitor and control scope of the project. Development leads will meet with team members daily at stand-up meetings at the end of each day. Development leads will then report to middle management at weekly update meetings to update them on development progress. Any changes to the scope of the project will be outlined in these meetings to development leads so they can adjust plans accordingly.

1. The project manager will create a WBS from the project scope statement and will be the person in charge of maintaining and approving the scope.

2. The project manager, requirements analyst and development leads will formally approve and accept completed project deliverables.

3. Requests for change to the project scope statement must be made formally to the project manager.

Secondly a requirements management plan will be created to define how requirements will be analysed, documented and managed (PMBoK 2013). We have decided that a hands-on relationship with the stakeholders and customers is the best way to collect requirements and the most effective way to find out if requirements change.

1. Requirements will be planned, tracked and reported through the requirements analyst, who confers with the project manager to decide if a requirement is included or excluded in scope. Or even part of future scope

2. The customer will be questioned on which requirements are the most important to them

3. Requirements will be traced using Atlassian Jira. This gives the requirements full traceability.

## 3.2 Collect Requirements

The output from planning scope is a requirements management plan. This plan describes how requirements will be analysed documented and managed. Requirements Elicitation is an important step in the development of a software application. There are a number of techniques possible, such as interviewing, protocol analysis, repertory grid, work groups" (Davis et al. 2006).

Structured interviews appear to be one of the most effective elicitation techniques in a wide range of domains and situations (Davis et al. 2006). The hardest part of building a software system is deciding what to build No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later. (Brooks Jr 1956).

In eliciting possible requirements for the site, there was a discussion with three club members with varying backgrounds and experience within the club.

At an early stage of the project, rough storyboards were prepared. These storyboards were used to demonstrate how a page, such as the timetable shown in Figure 3.1, would be displayed by the application.

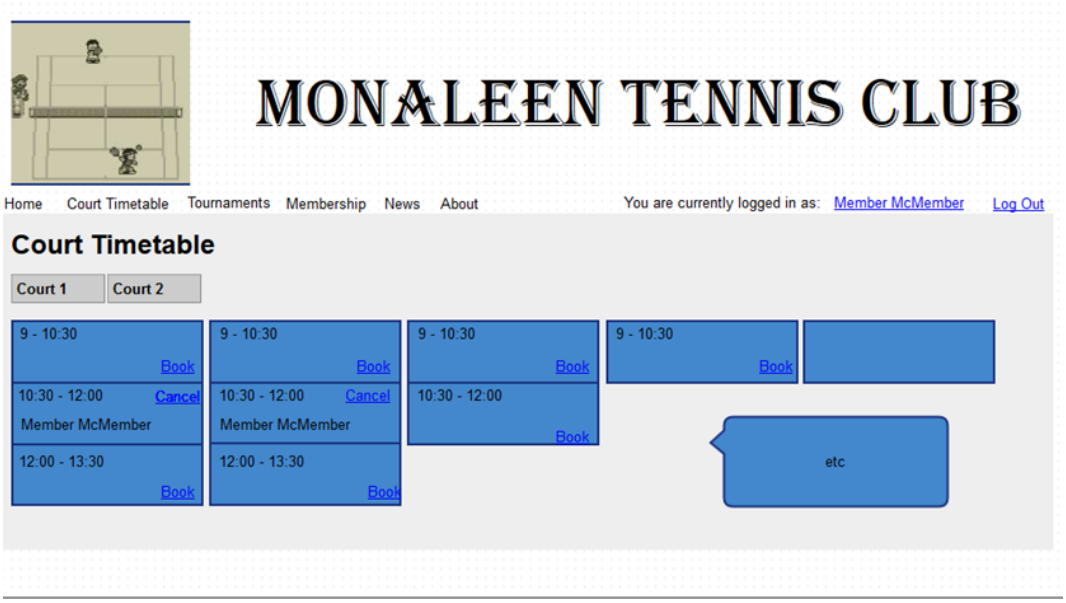
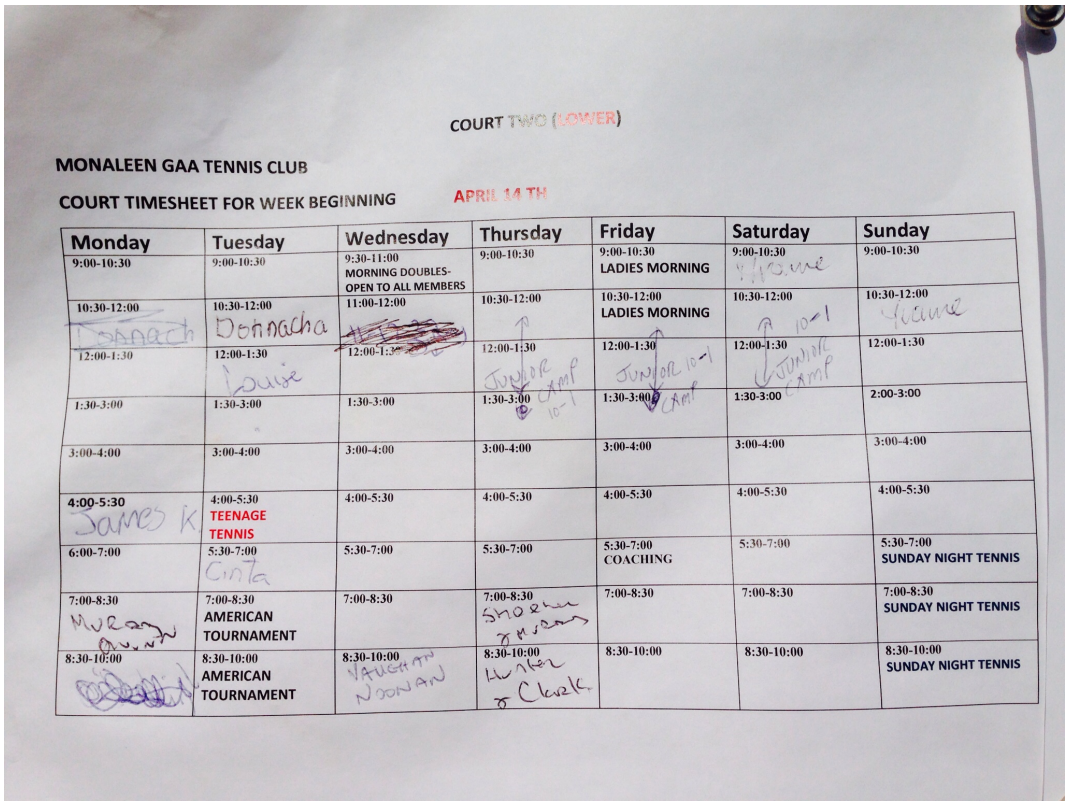



Figure 3.1: Timetable Storyboard, October 2013

The storyboarding visualised aspects of the site, and gave a rough idea of functionality that would be needed within the application. These were shown to interviewees in order to help them visualise requirements.

The style of the timetable was based on the existing paper based solution currently implemented by the club, displayed in Figure 3.2.



COURT TWO (LOWER)

MONALEEN GAA TENNIS CLUB

COURT TIMESHEET FOR WEEK BEGINNING APRIL 14 TH

| Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday |
|---|---|---|---|---|---|---|
| 9:00-10:30 | 9:00-10:30 | 9:30-11:00 MORNING DOUBLES- OPEN TO ALL MEMBERS | 9:00-10:30 | 9:00-10:30 LADIES MORNING | 9:00-10:30 | 9:00-10:30 |
| 10:30-12:00 | 10:30-12:00 | 11:00-12:00 | 10:30-12:00 | 10:30-12:00 LADIES MORNING | 10:30-12:00 | 10:30-12:00 |
| 12:00-1:30 | 12:00-1:30 | 12:00-1:30 | 12:00-1:30 | 12:00-1:30 | 12:00-1:30 | 12:00-1:30 |
| 1:30-3:00 | 1:30-3:00 | 1:30-3:00 | 1:30-3:00 | 1:30-3:00 | 1:30-3:00 | 2:00-3:00 |
| 3:00-4:00 | 3:00-4:00 | 3:00-4:00 | 3:00-4:00 | 3:00-4:00 | 3:00-4:00 | 3:00-4:00 |
| 4:00-5:30 | 4:00-5:30 TEENAGE TENNIS | 4:00-5:30 | 4:00-5:30 | 4:00-5:30 | 4:00-5:30 | 4:00-5:30 |
| 6:00-7:00 | 5:30-7:00 | 5:30-7:00 | 5:30-7:00 | 5:30-7:00 COACHING | 5:30-7:00 | 5:30-7:00 SUNDAY NIGHT TENNIS |
| 7:00-8:30 | 7:00-8:30 AMERICAN TOURNAMENT | 7:00-8:30 | 7:00-8:30 | 7:00-8:30 | 7:00-8:30 | 7:00-8:30 SUNDAY NIGHT TENNIS |
| 8:30-10:00 | 8:30-10:00 AMERICAN TOURNAMENT | 8:30-10:00 | 8:30-10:00 | 8:30-10:00 | 8:30-10:00 | 8:30-10:00 SUNDAY NIGHT TENNIS |

Figure 3.2: Paper Timetable, Monaleen Tennis Club

| Name | Age Bracket | Club Role | Club Membership | Work Background |
|---|---|---|---|---|
| S1 | 35 - 45 | Committee Member | 5 years | Senior Software Engineer |
| S2 | 18 - 25 | New Member | 1 year | Graduate Software Engineer |
| S3 | 55+ | Senior Member | 10+ years | Retired Public Servant |

Table 3.1: Stakeholders for Requirements Elicitation

In order to elicit requirements, a number of interviews were held with stakeholders. A set of questions were prepared as a guideline, and some storyboards were presented. The Tralee Tennis Club website was also presented to the interviewees:

1. Are you aware there is a site for Monaleen Tennis Club?
2. If are, what do you you it for?
3. What are the best features of the site?
4. What features are needed?

5. If you were not aware, why?
6. Would you use a website if you had known it was there?
7. Looking at the examples provided, are there any things you feel this club could benefit from?
8. How do you feel current tournaments are organised?
9. Is it easy to register for tournaments?
10. Have you ever had issues finding information about club events?
11. Do you know when club events are happening?
12. Is it easy to contact club members or find out how to contact them?
13. How easy is it to become a member/renew your membership?
14. How do you contact club committee members?

During the elicitation process a number of areas were highlight as desired features.

1. Online Timetable
    - Allow members to view courts and make/view bookings
2. Tournaments
    - Allow user to register for a tournament, and view tournaments ongoing.
3. Contact Members
    - Easy way to contact all members
4. Member Directory
    - A list of all members, contact details, roles
5. News Section
    - Create new items to display for members
6. Members Area
    - A secure area that only members could access
7. Member Application

- Automated registration, replace old paper form

8. Club Map
    - Directions to the club for new members and non-local visitors
9. Contact Details
    - Information on how to contact within the club for specific needs
10. Statistics
    - Such as games played, Win/Loss ratio

Table 3.2: Requested Features

Table 3.3 refers to each numbered requirement, and whether it was brought up by a stakeholder during the elicitation process.

| *Name* | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| S1 | N | Y | Y | Y | Y | Y | Y | Y | Y | N |
| S2 | Y | Y | Y | N | N | N | N | N | N | Y |
| S3 | Y | N | N | N | N | N | Y | Y | N | N |
| Total | 2 | 2 | 2 | 1 | 1 | 1 | 2 | 2 | 1 | 1 |

Table 3.3: Requested Feature Breakdown

## 3.3 Scope Definition

This section documents the scope of the tennis club website, i.e. what is included and excluded in the scope of the project. Inputs:

1. Scope management plan
2. Project charter
3. Requirements documentation
4. Organisational process assets

Outputs:

1. Project scope statement
2. Project documents updates

### 3.3.1 Included in Scope

The following is an outline of what is included in the project management scope. A major goal of this project is to develop a general tennis club platform that can be applied to many clubs to maximise profit.

Functional Requirements:

1. Timetable for booking courts
    - Flexible
    - Book slot
    - Unbook slot
    - Define a template for timetable (Admin)
    - Reset timetable (Admin)
    - Delete timetable (Admin)
    - Enable and disable timetable
    - Different clubs use different booking systems
    - Timetable Analysis
2. The tournament system is used by both players and administrators
    - Players register an interest in a tournament
    - Admins use system to organise players into teams/matches
    - Create tournament
    - Delete tournament
    - Display tournament fixtures
    - Create singles/doubles/round-robin tournaments
    - Contact users registered for the tournament
    - Display tournament results
    - Register a result of a match (both players have to agree)

3. News Blog
    - Create new post
    - Upload images
    - Rich text formatting
    - Embed videos from third party
    - Delete post
    - Edit post
    - Display Post
4. Members Services
    - Registration
    - Members only access
    - Approve/Block members (Admin)
    - Delete expired members (Admin)
    - Contact all members (Admin)
    - View member list (Admin)
    - Pay membership fee
    - Buy equipment

Also included in the scope of the project is a customer deployment team who can liaise with the customers to deploy the application either on cloud services or the customers dedicated servers. Non-functional requirements (NFR) included in the scope are usability, security and extensibility.

To support the security NFR, user data must be securely sored in our databases. We are building a payment platform for members to pay their fees therefore we need to protect user billing data with the latest security standards. The company can come under legal scrutiny unless user data is handled appropriately.

We plan to support extensibility as a main NFR so that we can deal with changes to existing clubs and supporting different types of clubs with differing requirements. The main customer has two courts for users to book a slot in. The club has plans to expand in the future resulting in the creation of four new courts. The number of courts available for booking therefore needs to be dynamic. Often clubs use different types of tournaments, i.e. there is no one-size-fits-all tournament type that all clubs use. Our application needs to support ladder, bracket and

league tournaments. The club also hosts tournaments with outside clubs; the ability to add non-members as a guest to our tournament system is required.

We are developing the system for a singular customer, however, there is room in the scope of the project to build a general platform that can be customised for each customer with minimal effort on the customer side. Simple settings like number of courts, type of tournaments and different payment plans are examples of features that could be provided.

The age of the club members range from 8 to 80, meaning ease of use of any software solution is important. If a system is going to replace the existing paper based system, a similar level of usability is required so all players can use it. The most common function used by members is the reservation of time slot for a court. A software alternative needs to be intuitive for all members.

### Excluded from Scope

Excluded from the scope of the project is a social network for tennis players of the club. We researched what other solutions offered and while this type of feature would be cutting edge technology for our solution, it is not a feature that users are desperate for or requesting right now.

### Constraints

The project has a tight budget and a low staff head count. The first customer also wants the product to be shipped in 6 months. Further, to justify adding multi club support to the product we need to find a second customer in the first 12 months of release; as this is a global product, this constraint should be low risk.

## 3.4 Create WBS

Creating the WBS is the process of subdividing project deliverables or work into smaller components

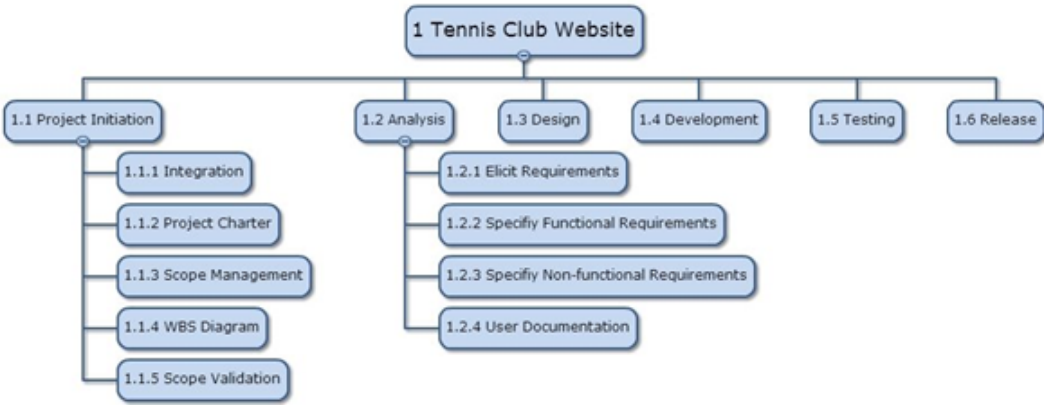



Figure 3.3: Work Breakdown Structure

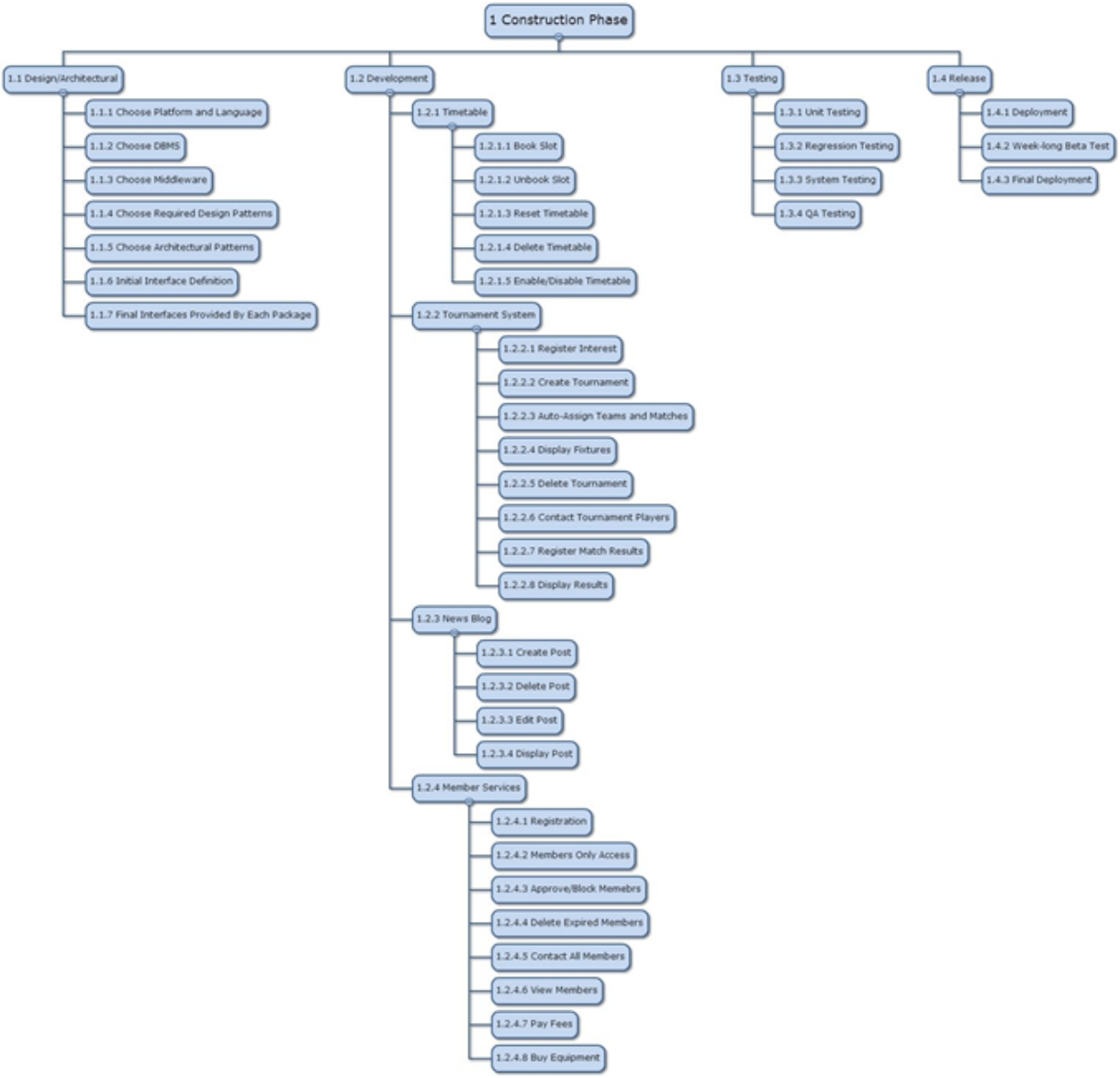



Figure 3.4: Work Breakdown Structure - Construction Phase

## 3.5 Validate Scope

Validating scope is the process of formalizing acceptance of the completed project deliverables.

Validation can be achieved using two techniques: inspection and group decision making techniques. Inspections can be carried out using meeting styles like reviews, audits and walkthroughs. Audits can be executed in a group or individual setting. Walkthroughs are group activities only. Group decision techniques will also be used in this project. As this is a high stakes project, development leads, project managers, test team leads, customers and any other relevant stakeholders will be present at group meetings. The method we will use for reaching group decisions is a majority decision, as this is the fairest to all stakeholders. Group meetings will be held monthly.

### 3.5.1 Outputs

#### Accepted Deliverables

The accepted deliverables of this project are the following:

1. Timetable for booking courts
2. Tournament system to be used by both players and administrators
3. News Blog
4. Members Services

These deliverables have been signed off on the by the project manager and the customer.

## 3.6 Control Scope

Controlling scope is the process of monitoring the status of the project and product scope. It also involves managing changes to the scope baselines and allows the scope baseline to be maintained throughout the project (PMBoK 2013).

Variance analysis is the technique used to measure the difference between the baseline and actual project performance. This type of analysis also measures the cause of the variation. Using this technique we can determine whether we need to change the scope of the project. Variance analysis will be executed bi-weekly.

# Chapter 4

# Project Time Management

Managing time throughout the Tennis Club Web Service is a crucial part of managing this project. Our main aims when dealing with time though out this project is to build processes used by the manager and team in order to complete the project in a timely manner. One process that will be used by the project manager is to look at the project in smaller more manageable pieces. These smaller tasks are sequenced and allocated by the manager to his team. Once the tasks have been allocated and scheduled it is important that the tasks are monitored. Changes may need to be made to future tasks start times as there may be delays in previous task completing by their estimated completion time.

## 4.1 Processes

Project Time Management can be broken up into seven processes. These processes cover three main areas. The first area is the planning process which deals with creating tasks, scheduling tasks and allocation of tasks. We did this through creating a Work Breakdown Structure. Controlling and monitoring process is the next main area. This area is concerned with the progression and monitoring of the tasks which were assigned to the team. Finally, the closing process, an area which is concerned with the end result. A look at challenges and hurdles which were met during the current project will help managers in building better and more accurate plans and processes in future projects.

1. Plan Schedule Management
2. Define Activities
3. Sequence Activities
4. Estimate Activity Resources

5. Estimate Activity Durations
6. Develop Schedule
7. Control Schedule

## 4.2 Plan Schedule Management and Define Activities

Project Schedule development uses the outputs from the processes to define activities, sequence activities, estimate activity resources, and estimate activity durations in combination with the scheduling tool to produce the schedule model. The finalized and approved schedule is the baseline that will be used in the Control Schedule process (PMBoK 2013).

Understanding and defining a clear scope is crucial as all listed actions must be within the boundaries of scope and measurable.

Some tools and techniques we used when identifying and managing these actions are:

1. Decomposition
    - We divided the project scope and deliverables into smaller actions. It is much easier to identify and control these action in smaller pieces rather than looking at the project as a whole.
2. Rolling Wave Planning
    - This technique involves planning the work to be done in the near future in more detail then work which is not scheduled until later in the project.

## 4.3 Sequence Activities

PDM is a process used to identify the relationships between activities. This relationship is then used when creating the schedule plan and the sequence of the tasks which must be completed. These tasks or activities are represented by nodes which ar linked together to graphically represent the critical path and sequence of tasks.

Types of dependences:

1. Finish-to-Start
    - A logical relationship which a successor activity cannot start until a predecessor activity has finished.

2. Finish-to-Finish

    - A logical relationship in which a successor activity cannot finish until a predecessor activity has finished.

3. Start-to-Start

    - A logical relationship which a successor activity cannot start until a predecessor activity has started.

4. Start-to-Finish

    - A logical relationship in which a successor activity cannot finish until a predecessor activity has started.

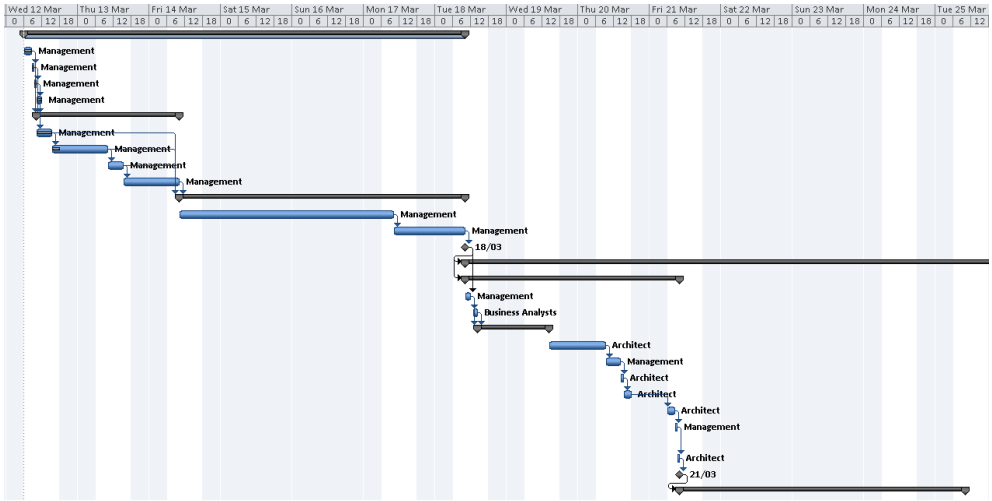



Figure 4.1: Gantt Chart

## 4.4 Estimate Activities

This is a key step in project time management where the resources for the activities and the time taken to complete these activates are done. The resources are allocated to the tasks and Work Breakdown Structures are created. The critical path is now identified and the total project duration is estimated. Expert judgement and experience from previous projects is crucial at this point.

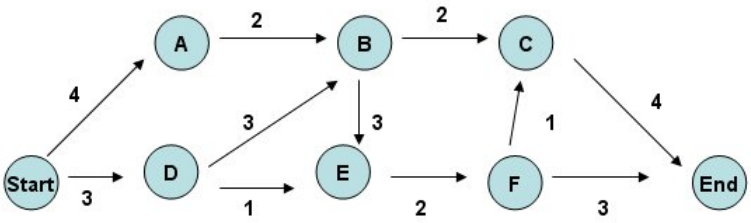



Figure 4.2: Critical Path

## 4.5 Develop and Control Schedule

All information and outputs from the previous processes are now used to create a accurate schedule. Tools such as Microsoft Project are used for creating and monitoring the schedule.

Schedule Network Analysis  This is a detailed report of how and when each stage or task should be executed. Gantt charts and network diagrams are used to graphically represent the activities and milestones which have been completed and what is left to do.

1. Critical Path Method
    - A process used to identify the critical path of a project, any delays or changes will result on your project being late.
2. Resource Levelling
    - This process is used to identify the need for reallocation of team members in order to stay on the critical path and achieve the targeted project completion time.

Schedule Compression is used n the event that tasks get delayed. The following two techniques are used to change the current schedule and achieve a minimum impact on the remaining project.

1. Crashing
    - Allocating more people to complete a task or assigning team to work overtime.
2. Fast tracking
    - Starting a task before a previous task has completed

Schedule Control is important as if any changes to the schedule are needed, they must be re-evaluated and the project schedule must be updated.

# Chapter 5

# Project Cost Management

Cost management is concerned with estimating, job control and data collection. The following are some techniques which are used to generate outputs such as cost management plans and cost requirements are as follows.

1. Plan Cost Management
2. Estimate Costs
3. Determine Budget
4. Control Costs

## 5.1 Planning Cost Management

Planning for cost management is a important part of cost management. Planning cost from the beginning will ensure accurate time and cost estimates. Expert judgement and experience is essential when managing and controlling project costs through a Cost Management Plan and Project Charters.

A Cost Management Plan includes:

1. Units of measure
2. level of precision
3. level of accuracy
4. Organisational procedures link
5. Control thresholds

6. Rule of performance measurement
7. Reporting formats
8. Process descriptions
9. Additional details

## 5.2 Budgeting

When estimating cost in a project all risks and environmental and organizational process must be considered. A scope baseline is used to determine and prevent scope creep. An accurate and complete project scope statement, work breakdown structure and WBS dictionary are used to determine the project baseline.

We developed a project cost estimate through an application called COCOMO. The following is an image of the estimate using recent typical salary statistics found.

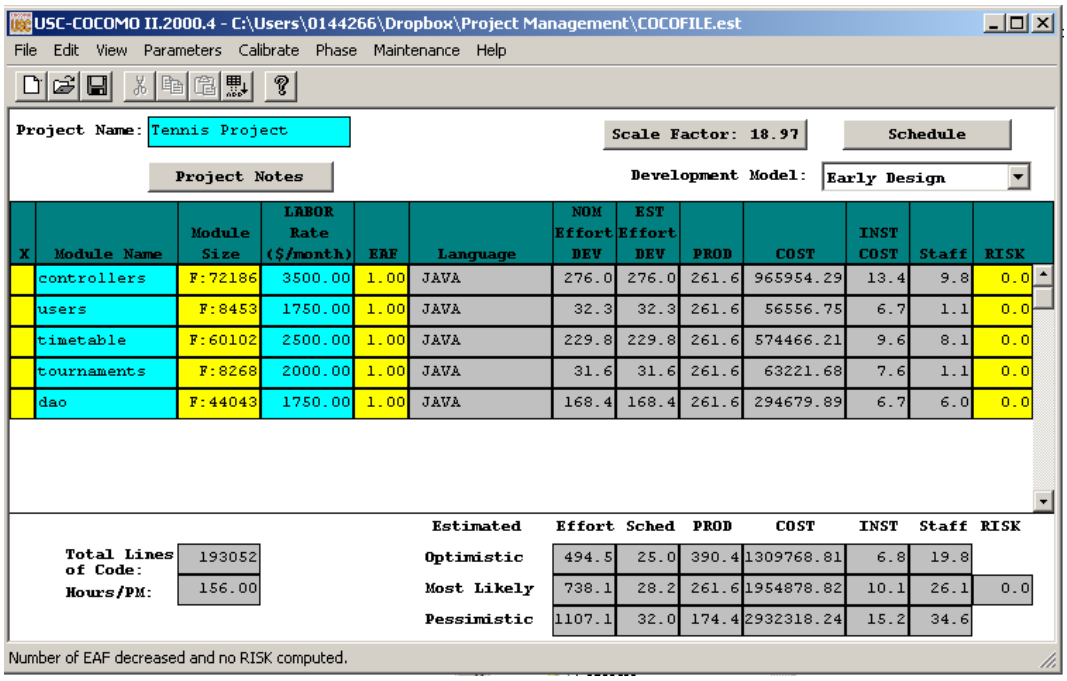



Figure 5.1: COCOMO Early Design

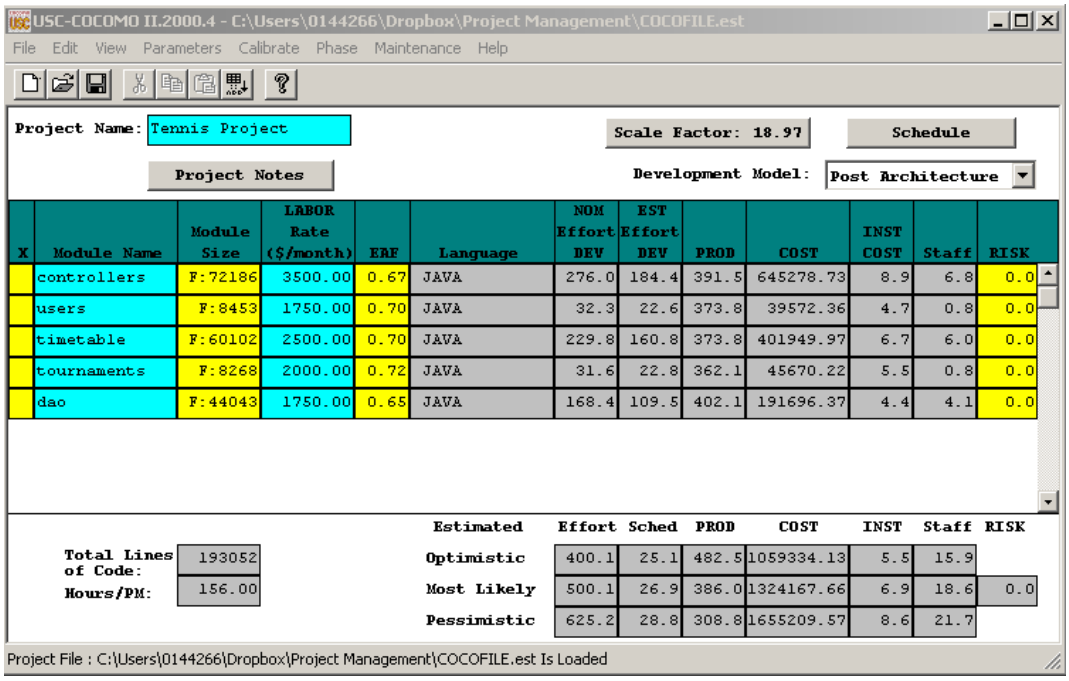



Figure 5.2: COCOMO Post Architecture

The key attribute that we focused on as a cost factor was re-usability. This is concerned with creating a more generic design of software, more elaborate documentation, and more extensive testing to ensure components are ready for use in other applications.

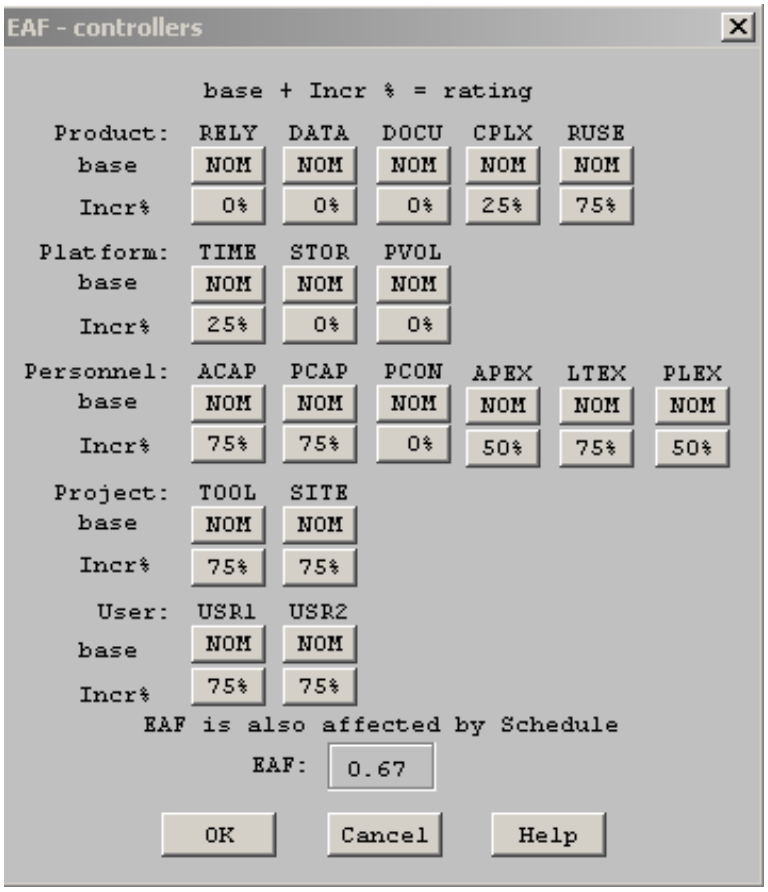



Figure 5.3: COCOMO Cost Weighting

1. RELY

   - "This is the measure of the extent to which the software must perform its intended function over a period of time. If the effect of a software failure is only slight inconvenience then RELY is low. If a failure would risk human life then RELY is very high." (*COCOMO II Cost Driver and Scale Driver Help* 1998). Within this project, reliability was not especially important and was given a low value.

2. DATA

   - "This measure attempts to capture the affect large data requirements have on product development. The rating is determined by calculating D/P. The reason the size of the database is important to consider it because of the effort required to generate the test data that will be used to exercise the program." (*COCOMO II Cost Driver and Scale Driver Help* 1998). The database requirements for this project were not massive, so this was given a low ranking.

3. DOCU

- "The rating scale for the DOCU cost driver is evaluated in terms of the suitability of the project's documentation to its life-cycle needs. The rating scale goes from Very Low (many life-cycle needs uncovered) to Very High (very excessive for life-cycle needs)." This is slow as a lot of the factors, such as security of the application, are supported by the frameworks chosen.

4. CPLX

   - "Complexity is divided into five areas: control operations, computational operations, device-dependent operations, data management operations, and user interface management operations. Select the area or combination of areas that characterize the product or a sub-system of the product. The complexity rating is the subjective weighted average of these areas." (*COCOMO II Cost Driver and Scale Driver Help* 1998). Complexity was adjusted upwards due to the interactions between modules. This was helped by the use of a service layer.

5. RUSE

   - "This cost driver accounts for the additional effort needed to construct components intended for reuse on the current or future projects. This effort is consumed with creating more generic design of software, more elaborate documentation, and more extensive testing to ensure components are ready for use in other applications." (*COCOMO II Cost Driver and Scale Driver Help* 1998). This was a risk within the project, and the attempt to create a generic solution that could be reused instead of a specific site affected the project costs significantly.

6. TIME

   - "This is a measure of the execution time constraint imposed upon a software system. The rating is expressed in terms of the percentage of available execution time expected to be used by the system or subsystem consuming the execution time resource. The rating ranges from nominal, less than 50% of the execution time resource used, to extra high, 95% of the execution time resource is consumed." (*COCOMO II Cost Driver and Scale Driver Help* 1998).

7. STOR

   - "This rating represents the degree of main storage constraint imposed on a software system or subsystem" (*COCOMO II Cost Driver and Scale Driver Help* 1998). This was not deemed to be important due the the cost of increasing memory, storage and processor speed for an application. While optimization is important, this attribute was determined at a much different time in the development industry.

8. PVOL

    - ""Platform" is used here to mean the complex of hardware and software (OS, DBMS, etc.) the software product calls on to perform its tasks. This rating ranges from low, where there is a major change every 12 months, to very high, where there is a major change every two weeks." (*COCOMO II Cost Driver and Scale Driver Help* 1998). This is not as important as the application should run with the current version of the framework for a significant amount of time. Updating the frameworks may actually increase development time due to changes introduced.

9. ACAP

    - This is the ability of the analyst to design the application. " The major attributes that should be considered in this rating are Analysis and Design ability, efficiency and thoroughness, and the ability to communicate and cooperate" (*COCOMO II Cost Driver and Scale Driver Help* 1998)

10. PCAP

    - This is the capability of the programming ability within the project. "Evaluation should be based on the capability of the programmers as a team rather than as individuals. Major factors which should be considered in the rating are ability, efficiency and thoroughness, and the ability to communicate and cooperate. The experience of the programmer should not be considered here; it is rated with AEXP. A very low rated programmer team is in the 15th percentile and a very high rated programmer team is in the 95th percentile." (*COCOMO II Cost Driver and Scale Driver Help* 1998)

11. PCON

    - This is the turnover within the development team. "The rating scale for PCON is in terms of the project's annual personnel turnover: from 3%, very high, to 48%, very low." (*COCOMO II Cost Driver and Scale Driver Help* 1998)

12. AEPX

    - "This rating is dependent on the level of applications experience of the project team developing the software system or subsystem" (*COCOMO II Cost Driver and Scale Driver Help* 1998). This is how much experience that the chosen development team would have with the frameworks (Spring MVC, Hibernate, Tiles) chosen for this project.

13. LTEX

- The is Language and Tool Experience. "This is a measure of the level of programming language and software tool experience of the project team developing the software system or subsystem." (*COCOMO II Cost Driver and Scale Driver Help* 1998)

14. TOOL

    - This is governed by the use of software tools to aid development. "The tool rating ranges from simple edit and code, very low, to integrated lifecycle management tools, very high." (*COCOMO II Cost Driver and Scale Driver Help* 1998). Examples are the use of an IDE like Eclipse, and a source control system, like git.

15. SITE

    - This is the cost created by the dispersion of staff within multiple sites. "Determining its cost driver rating involves the assessment and averaging of two factors: site collocation (from fully collocated to international distribution) and communication support (from surface mail and some phone access to full interactive multimedia)." (*COCOMO II Cost Driver and Scale Driver Help* 1998)

## 5.3 Realistic Targets

Tools and techniques used to generate the work breakdown structures include:

1. Cost Aggregation

    - WBS are generated by grouping cost estimates into packages and those packages are then grouped into higher levels of the WBS.

2. Reserve Analysis

    - It involves both the contingency reserves and the management reserves.

3. Expert Judgement

    - Mathematical models to predict total project cost are used to reconcile with with any funding limitations. A cost baseline,funding requirements and determined expenditures are now generated form the previous processes.

## 5.4 Cost Control

There must be constant updates made to the project cost plans and any changes to the cost baseline must be updated. These changes are done through Cost forecasts and Change Requests and constantly documented and updated. Another technique used to measure project performance is Earned Value Management. This technique integrates scope time and cost date which is determined form the baseline. The following information is input into the EVM on a regular basis.

1. Planned Value(PV)
2. Actual Cost (AC)
3. Earned Value(EV)
4. Schedule Variances (SV)
5. Cost Variance (CV)
6. Schedule Performance Index(SPI)
7. Cost Performance Index(CPI)

| Term | Formula |
|---|---|
| Cost Variance | CV = EV — AC |
| Schedule Variance | SV = EV — PV |
| Cost Performance Index | CPI = EV/AC |
| Schedule Performance Index | SPI = EV/PV |

Figure 5.4: EVM

# Chapter 6

# Project Quality Management

The aim of Quality Management is to manage the quality of the final software product, and the quality of the development process used to create it. Another measure of quality is ensuring that the final product meets the requirements as defined by the stakeholders.

## 6.1 Quality Process

The process following for quality would focus on three areas.

1. Plan Quality Management
2. Perform Quality Assurance
3. Control Quality

### 6.1.1 Quality Management

Quality Management is the process of "identifying quality requirements and standards for the project" (PMBoK 2013). It is also important that the project demonstrates "compliance with relevant quality requirements and standards" (PMBoK 2013). This project focuses on a number of quality attributes during the development process.

1. Performance

    - The ability of the project to respond to requests with clearly defined time constraints. For this project, the reaction time the the web application to user demanded is important.

2. Security

    - The protection, integrity and non-repudiation of stored data. User information, such as payment information, will be stored. The security of this data is integral to the application.

3. Usabilty

    - The system must be easy to use, and easy to learn. Usability studies need to be performed on the application to ensure ease of use.

4. Maintainability

    - It must not be cost prohibitive to perform corrective, perfective and adaptive maintenance on the product. The project must be designed in such a way that any changes that need to be made can be done at minimal cost.

5. Re-usability

    - Components of the system must be design modularly in order to promote reuse. A reusable system, if the risk is managed correctly, will save costs if components are reusable, and may also allow further product lines.

6. Portability

    - The project application must work across a range of operating systems and browsers.

The project aims to conform to the ISO 25010 standard detailed later in this report.

### 6.1.2 Quality Assurance

Quality Assurance is governed by the user of standards, such as the ISO 25010, within the project. In a big organisation, the idea of Total Quality Management is used to ensure that a high quality product is delivered. This draws on a number of different quality methods illustrated in Figure 6.1.

- Controls

- Job Management
- Well defined processes
- Employee competence
- Organisation culture
- Team spirit

Figure 6.1: Sample of methods to ensure quality

## 6.2 Quality Control

Quality Control is a monitoring and control process. It inspects every project deliverable, and is measured in some way. Results are checked to ensure their conformity to quality standards. This process covers the project, and the products produced by the project. Corrective measures needs to be applied to any defects discovered throughout this process.

Quality Assurance can often be confused with Quality Control. The PMBoK gives an example to clarify the difference.

"Quality assurance could be calibrating a machine or training the operator, while quality control is inspecting or testing the products that are being made by the machine" (PMBoK 2013).

## 6.3 Quality Support

Quality Support throughout the development process was given through the use of support tools, namely inFusion, that examined code focusing on a number of quantifiable attributes. These included Encapsulation, Coupling, Cohesion, Complexity and Inheritance. This allowed developers to be aware of design flaws throughout the development process, and allow the identification of bugs and discrepancies at development time, rather than expose them during testing.

### 6.3.1 Tools and Techniques

#### Techniques

A number of defined techniques are available to use within the project in order to control quality.

**Cost Benefit Analysis** offsets the costs of supporting a quality attribute versus the benefit of implementing it. An example may be having a very strong security component, when little to no data is stored in the application.

**Cost of Quality** "includes all costs incurred over the life of the product by investment in preventing non-conformance to requirements" (PMBoK 2013). The types of failures are internal and external: those found by the project, and those found the the customer. These need to be factored into overall project cost, and decisions made regarding quality can greatly impact the project long term.

A number of diagrams can be used to map quality within a project, shown in Figure **??**.

- Cause-and-effect Diagrams
- Flowcharts
- Check sheets
- Pareto Diagrams
- Histograms
- Control Charts
- Scatter Diagrams

### Tool - Infusion

Infusion is a tool that highlights flaws in a number of areas. It breaks them now, and identifies 'bad code smells' (Fowler 2006). Figure 6.2 shows a breakdown of a prototype application, and shows possible quality deficits within the application. This is an overall breakdown on the entire system. This can be further broken down to class level.

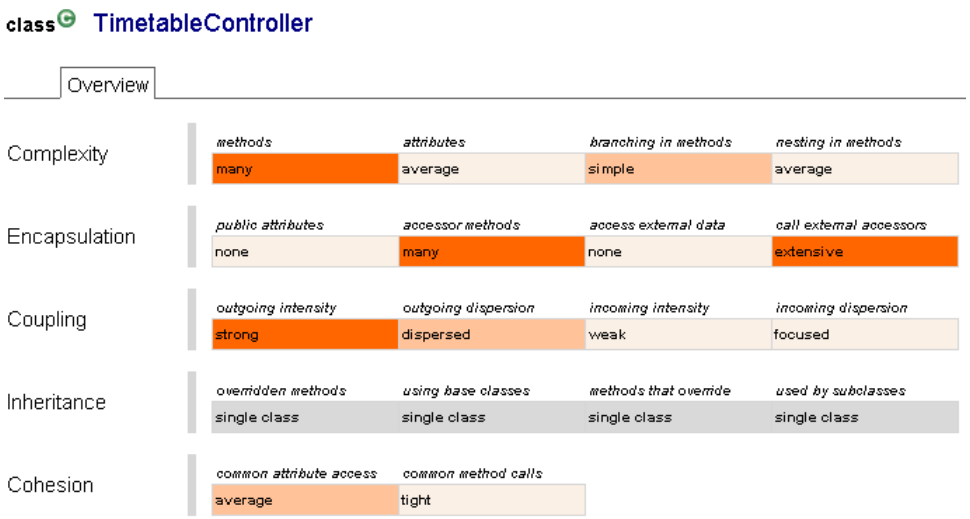



Figure 6.2: Example of Class 'Bad Code Smell' Breakdown using Infusion

An overall quality result is given to the application. Changes can then me made, in order to achieve a benchmark score. It is also worth noting that the use of frameworks, if that decision is made, made have a negative effect on these scores and a cognisance of this is paramount to ensure that time is not spend on corrective action for 'flaws' introduced by design choices.

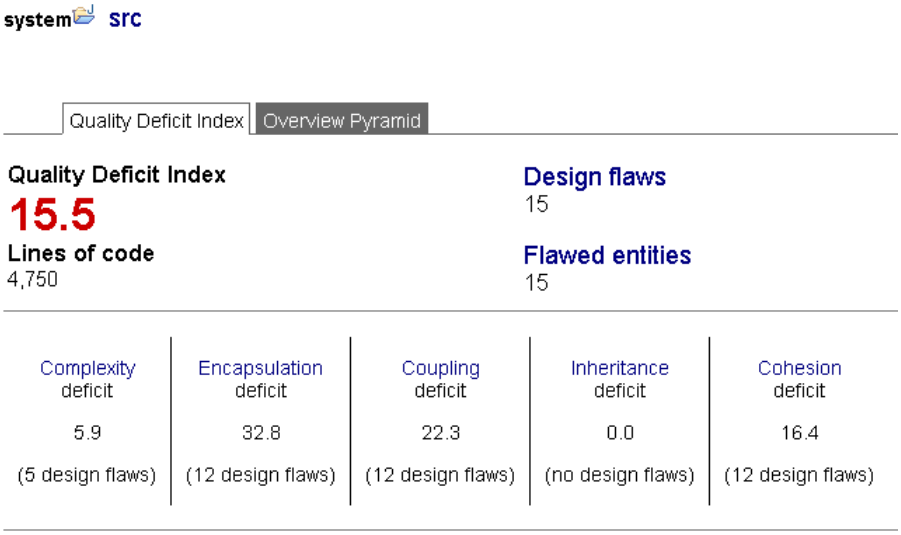



Figure 6.3: Quality Measurement using Infusion

## 6.4 Quality Standards

One standard looked at in this project was *ISO/IEC 25010*. This replaced the old model, *ISO 9126*.

In this model, there is a focus on a number of different attributes. In order to help a project succeed, there attributes must be examined, and acted upon, through the project management process. This is to ensure that the final product is a high quality product, and one that is what the customer wanted. Figure 6.4 shows a summary of the areas that this standard focuses on.

1. Functional Stability
    - This refers to the 'appropriateness, accuracy and compliance" of the final product. (ISO 2011)
2. Reliability
    - This section focuses on the "availability, fault tolerance and recoverability compliance" of the final product. (ISO 2011)
3. Performance Efficiency
    - This is concerned with measurements such as "time behaviour and resource utilisation compliance". (ISO 2011)
4. Operability
    - This encompasses usability. and focuses on "ease of use, helpfulness, attractiveness, technical accessibility and learn-ability" of the created system. (ISO 2011)
5. Security
    - Within this application, the focus of security is defined by the user requirements. This area will focus on the "confidentiality, integrity, non repudiation, and authenticity" of information within the system.
6. Compatibility
    - This area focuses on the effect that different configurations may have on the system. This include "interoperability and replace-ability". (ISO 2011)
7. Maintainability
    - This attribute focuses on the design of an application. Key attributes are the "modularity, re-usability, testability and changeability" of the application (ISO 2011). These attributes normally increase the development time of the application, and thus the risk and cost.
8. Transferability

- This refers to the "portability and adaptability" of the application to run on different systems, operating systems and configurations. (ISO 2011)

Figure 6.4: Quality Areas Defined by ISO 25010

# Chapter 7

# Project Risk Management

Project Risk Management is defined by PMBoK as "the process of conducting risk management planning, identification, analysis, response planning and monitoring and control on a project" (PMBoK 2013). The objectives of this project management area are to increase the probability and impacts of positive events, while at the same time decreases the same of negative events within the scope of the project.

## 7.1 Processes

There are a number of processes involved in the Risk Management aspect of a project plan, illustrated in Figure 7.1.

1. Plan Risk Management
    - The process of "defining how to conduct risk management activities for a project" (PMBoK 2013).
2. Identify Risks
    - Determining "which risks may affect the project and documenting their characteristics" (PMBoK 2013).
3. Perform Qualitative Risk Analysis
    - This is the process of prioritising risks. These risks can then be sent "for further analysis or action by assessing and combining their probability of occurrence and impact" (PMBoK 2013).
4. Perform Quantitative Risk Analysis

- This is the process of "numerically analysing the effect of identified risks on the overall project objectives" (PMBoK 2013).

5. Plan Risk Responses

   - This builds on the results of the previous processes. This process plans for how each risk should be managed, and which person is responsible for the management of that particular risk.

6. Control Risks

   - This is the process of "implementing the risk response plans, tracking identified risks, identifying new risks, and evaluating risk process effectiveness throughout the project" (PMBoK 2013).

Figure 7.1: Risk Management Processes

One of the main risks in a software project is the choice of the architecture. A qualified software architect, with the experience necessary to properly evaluate and choose a software solution, would play a major role in negating this risk. The choice of a proper architecture lays the groundwork for the future development, and is a mistake that is very hard to rectify further on in the project lifecycle.

## 7.2 Techniques

## 7.3 Risk Management Plans

The main risks exposed by this project are design choices and requirements elicitation. A big risk in the project would be a choice to develop a more generic solution, and tailor it towards the client, rather than develop specifically for the client. This approach would cost more over the lifetime of the project, but it would also allow the project to reach a larger potential customer base. The design of the system as a more modular, separate unit, composed of many parts, would increase development time and complexity, but long term, it would allow for reduced costs with regards towards the various kinds of maintenance: corrective, adaptive and perfective.

1. Use of Commercial Off The Shelf (COTS) software within the application

- Is there a cost to the user to get this software? What happens if it's updated/discontinued?

2. Staff Experience with Frameworks
   - Are the staff experienced with the chosen frameworks? Will it affect development?
3. Choice of frameworks
   - Risks associated with choosing more modern frameworks over those that developers have experience with, for reasons such as extensibility, performance (e.g. Spring MVC Vs Enterprise Java Beans; Tomcat 7 Vs Glassfish).
4. Changes to project modules
   - What modules would require the most work to change? How much work? How will it affect other modules?

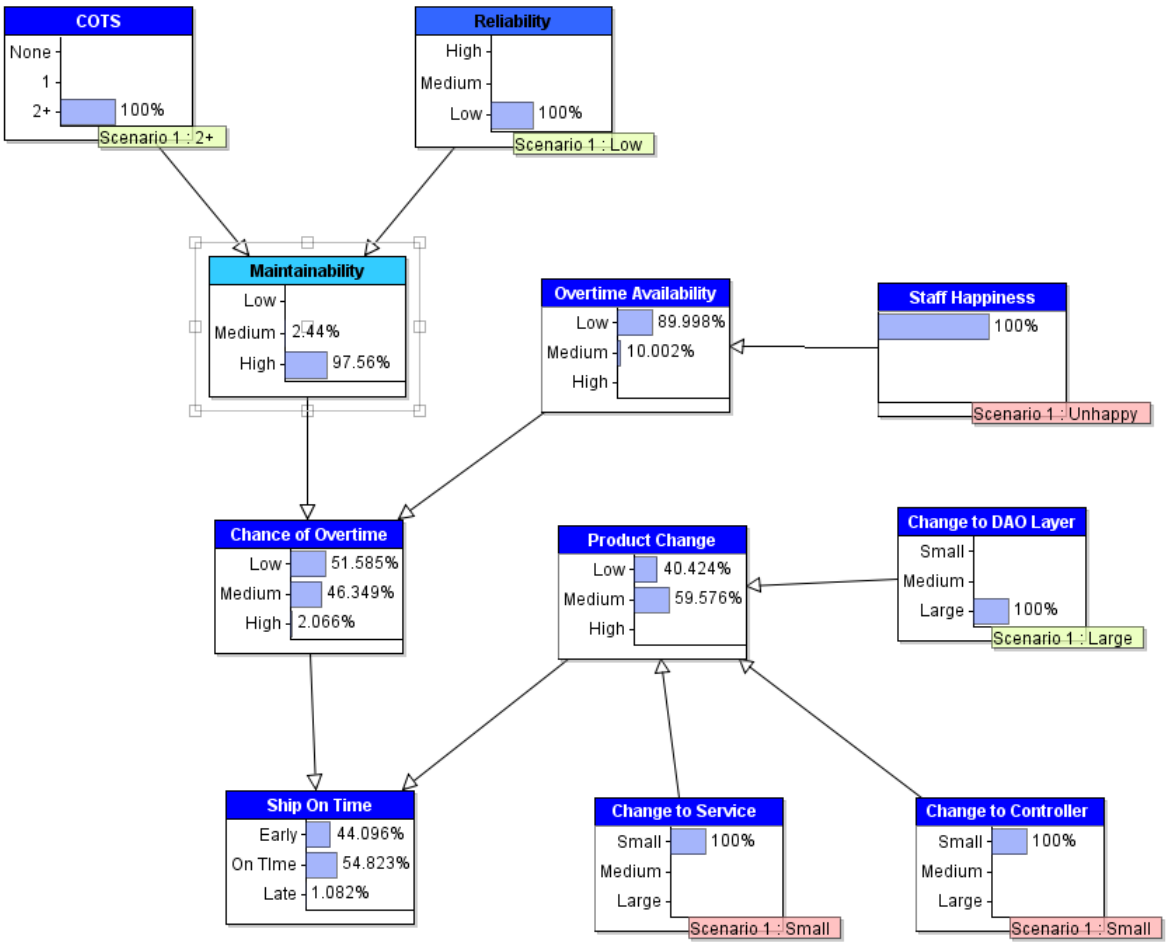



Figure 7.2: Agena Risk Model

# Chapter 8

# Project Human Resource Management

Human Resource Management is the proper coordination and organisation of the people within a project. It affects the "way the organisation acquires and uses human resources. and how employees experience the employment relationship" (Huemann, Keegan, and Turner 2007)

## 8.1 Plan Human Resource Management

The key output of this stage is a human resource management plan. In order to ensure that a proper plan is developed, it is necessary to be cognisant of a number of areas.

### 8.1.1 Roles

There are a number of roles needs for the development of this project. The project manager needs to ensure that the right people are in the right roles.

1. Project Manager
    - Responsible for managing as aspects of the project, such as ensuring the right people are in the right roles.
2. Software Architect
    - Responsible for choosing the architecture and language required to implement the project. They are also responsible for the design of the system.
3. Designers
    - Responsible for creating the design of the product, such as appearance, art etc.

4. Developers

    - Responsible for creating the actual product according the the specification given by the project manager

5. Testers

    - Responsible for testing the product produced by the developers and finding errors within it.

### 8.1.2 Relationships

The project manager needs to be aware of previously existing relationships within the project dynamics. These can be both beneficial and detrimental to the project result, and need to be managed to ensure that the project in improved by human interaction, rather than hurt.

## 8.2 Authority and Organisation

There needs to be a clear breakdown of who is responsible for what. An example of how to do this is using a responsibility assignment matrix (RAM). Figure 8.1 is an example of a RACI (Responsible, Accountable, Consult, and Inform) chart, which shows with role is responsible for what actions within the scope of this project.

| **Activity** | **Project Manager** | **Architect** | **Designer** | **Developer** | **Tester** |
|---|---|---|---|---|---|
| **Create charter** | R | I | I | I | I |
| **Collect Requirements** | I | A | R | I | I |
| **Submit Change Request** | I | A | R | I | I |
| **Develop Test Plan** | A | I | C | I | R |

A - Accountable, R - Responsible, I - Informed, C - Consult

Table 8.1: RACI Matrix, adapted from (PMBoK 2013)

## 8.3 Identify Competencies

This is the use of "expert judgement" (PMBoK 2013) by the project manager. The PMBOK provides a number of areas for the project manager to focus on:

1. List the preliminary requirements for the required skills;

2. Assess the roles required for the project based on standardized role descriptions within the organization;

3. Determine the preliminary effort level and number of resources needed to meet project objectives;

4. Determine reporting relationships needed based on the organizational culture;

5. Provide guidelines on lead time required for staffing, based on lessons learned and market conditions;

6. Identify risks associated with staff acquisition, retention, and release plans; and

7. Identify and recommend programs for complying with applicable government and union contracts.

    - (PMBoK 2013)

## 8.4 Organisation and Planning

In the present day, the concept of Global Software Development is widely explored as a method for software development in many companies. As such, project managers need to react to this change in the working environment. There is a "need to capitalize on the global resource pool to successfully and cost competitively use scarce resources" (Herbsleb and Moitra 2001). With the diversity introduced by this development, "close cooperation of individuals with different cultural backgrounds" (Herbsleb and Moitra 2001) is required by a project manager.

The PMBOK technique, Virtual Teams, would be beneficial in a project such as this. A Virtual Team is a group of people with a shared goal who fulfil their roles within the project, without having to actually meet face to face. Communication is facilitated through email, telephone calls, with work being managed by a source control system.

# References


Brooks Jr, Frederick P. (1956). "NO SILVER BULLET - ESSENCE AND ACCIDENTS OF SOFTWARE ENCINEERING". In:

*COCOMO II Cost Driver and Scale Driver Help* (1998). URL: `http://sunset.usc.edu/research/COCOMOII/expert_cocomo/drivers.html`.

Davis, Alan et al. (2006). "Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review". In: *Requirements Engineering, 14th IEEE International Conference*. IEEE, pp. 179–188.

Fowler, Martin (2006). *Code Smell*. URL: `http://martinfowler.com/bliki/CodeSmell.html`.

Herbsleb, James D and Deependra Moitra (2001). "Global software development". In: *Software, IEEE* 18.2, pp. 16–20.

Huemann, Martina, Anne Keegan, and J Rodney Turner (2007). "Human resource management in the project-oriented company: A review". In: *International Journal of Project Management* 25.3, pp. 315–323.

ISO, ISO (2011). "IEC 25010: 2011: Systems and software engineering–Systems and software Quality Requirements and Evaluation (SQuaRE)–System and software quality models". In: *International Organization for Standardization*.

PMBoK, A (2013). "Guide to the project Management body of knowledge". In: *Project Management Institute, Pennsylvania USA*.

# Appendix A

# Appendices

## A.1 Microsoft Project Images

| | Task Name | Baseline Dur. | Baseline Start | Baseline Finish | Baseline Work | Baseline Cost | Cost |
|---|---|---|---|---|---|---|---|
| 1 | **Project Initialisation** | **4.3 days** | **Wed 12/03/14** | **Tue 18/03/14** | **136.8 hrs** | **£0.00** | **£2,224.00** |
| 2 | Identify Stakeholders | 0.1 days | Wed 12/03/14 | Wed 12/03/14 | 0.8 hrs | £0.00 | £12.00 |
| 3 | Examine Existing Website for Core Functionality | 0.1 days | Wed 12/03/14 | Wed 12/03/14 | 0.8 hrs | £0.00 | £12.00 |
| 4 | Preparation of Questions for Interview | 0.1 days | Wed 12/03/14 | Wed 12/03/14 | 0.8 hrs | £0.00 | £12.00 |
| 5 | Meeting with team to discuss findings | 0.2 days | Wed 12/03/14 | Wed 12/03/14 | 1.6 hrs | £0.00 | £24.00 |
| 6 | **Review Inputs to Initiation** | **2 days** | **Wed 12/03/14** | **Fri 14/03/14** | **32 hrs** | **£0.00** | **£480.00** |
| 7 | Review Contract | 0.5 days | Wed 12/03/14 | Wed 12/03/14 | 4 hrs | £0.00 | £60.00 |
| 8 | Review Project Statement of Work | 0.5 days | Wed 12/03/14 | Thu 13/03/14 | 4 hrs | £0.00 | £60.00 |
| 9 | Review Enterprise Environmental Factors | 0.5 days | Thu 13/03/14 | Thu 13/03/14 | 4 hrs | £0.00 | £60.00 |
| 10 | Review Organizational Process Assets | 0.5 days | Thu 13/03/14 | Fri 14/03/14 | 4 hrs | £0.00 | £60.00 |
| 11 | **Produce Outputs from Initiation** | **2 days** | **Fri 14/03/14** | **Tue 18/03/14** | **32 hrs** | **£0.00** | **£480.00** |
| 12 | Develop Project Charter | 1 day | Fri 14/03/14 | Mon 17/03/14 | 8 hrs | £0.00 | £120.00 |
| 13 | Develop Peliminary Project Scope Statement | 1 day | Mon 17/03/14 | Tue 18/03/14 | 8 hrs | £0.00 | £120.00 |
| 14 | Initiation Processes Complete | 0 days | Tue 18/03/14 | Tue 18/03/14 | 0 hrs | £0.00 | £0.00 |
| 15 | **Project Planning** | **19 days** | **Tue 18/03/14** | **Mon 14/04/14** | **448 hrs** | **£0.00** | **£8,164.00** |
| 16 | **Analysis of Software Requirements** | **3 days** | **Tue 18/03/14** | **Fri 21/03/14** | **56 hrs** | **£0.00** | **£1,344.00** |
| 17 | Conduct needs analysis | 0.2 days | Tue 18/03/14 | Tue 18/03/14 | 1.6 hrs | £0.00 | £24.00 |
| 18 | Interview with Club Captains for Requirements Elicitatic | 0.2 days | Tue 18/03/14 | Tue 18/03/14 | 1.6 hrs | £0.00 | £48.00 |
| 19 | **Draft requirements as a result of interview** | **1 day** | **Tue 18/03/14** | **Wed 19/03/14** | **16 hrs** | **£0.00** | **£480.00** |
| 22 | Draft preliminary software specifications | 0.5 days | Wed 19/03/14 | Thu 20/03/14 | 4 hrs | £0.00 | £180.00 |
| 23 | Develop preliminary budget | 0.5 days | Thu 20/03/14 | Thu 20/03/14 | 4 hrs | £0.00 | £60.00 |
| 24 | Review software requirements/budget with team | 0.1 days | Thu 20/03/14 | Thu 20/03/14 | 0.8 hrs | £0.00 | £36.00 |
| 25 | Incorporate feedback on software specifications | 0.2 days | Thu 20/03/14 | Thu 20/03/14 | 1.6 hrs | £0.00 | £72.00 |
| 26 | Develop delivery timeline | 0.1 days | Fri 21/03/14 | Fri 21/03/14 | 0.8 hrs | £0.00 | £36.00 |
| 27 | Obtain approval to proceed with stakeholders | 0.1 days | Fri 21/03/14 | Fri 21/03/14 | 0.8 hrs | £0.00 | £12.00 |
| 28 | Secure required resources | 0.1 days | Fri 21/03/14 | Fri 21/03/14 | 0.8 hrs | £0.00 | £36.00 |
| 29 | Analysis complete | 0 days | Fri 21/03/14 | Fri 21/03/14 | 0 hrs | £0.00 | £0.00 |
| 30 | **Scope Management Process** | **2 days** | **Fri 21/03/14** | **Tue 25/03/14** | **32 hrs** | **£0.00** | **£480.00** |
| 39 | **Activity Planning** | **2 days** | **Tue 25/03/14** | **Thu 27/03/14** | **32 hrs** | **£0.00** | **£480.00** |
| 44 | **Cost Planning** | **1 day** | **Thu 27/03/14** | **Fri 28/03/14** | **16 hrs** | **£0.00** | **£240.00** |
| 47 | Complete Quality Plan | 0.5 days | Fri 28/03/14 | Fri 28/03/14 | 4 hrs | £0.00 | £60.00 |
| 48 | Complete Human Resource Plan | 1 day | Fri 28/03/14 | Mon 31/03/14 | 8 hrs | £0.00 | £120.00 |
| 49 | Complete Communication Plan | 0.5 days | Mon 31/03/14 | Tue 01/04/14 | 4 hrs | £0.00 | £60.00 |
| 50 | **Risk Management Processes** | **2 days** | **Tue 01/04/14** | **Thu 03/04/14** | **32 hrs** | **£0.00** | **£480.00** |
| 55 | **Procurement Management Processes** | **2 days** | **Thu 03/04/14** | **Mon 07/04/14** | **32 hrs** | **£0.00** | **£480.00** |

Figure A.1: Work Breakdown Structure Task List

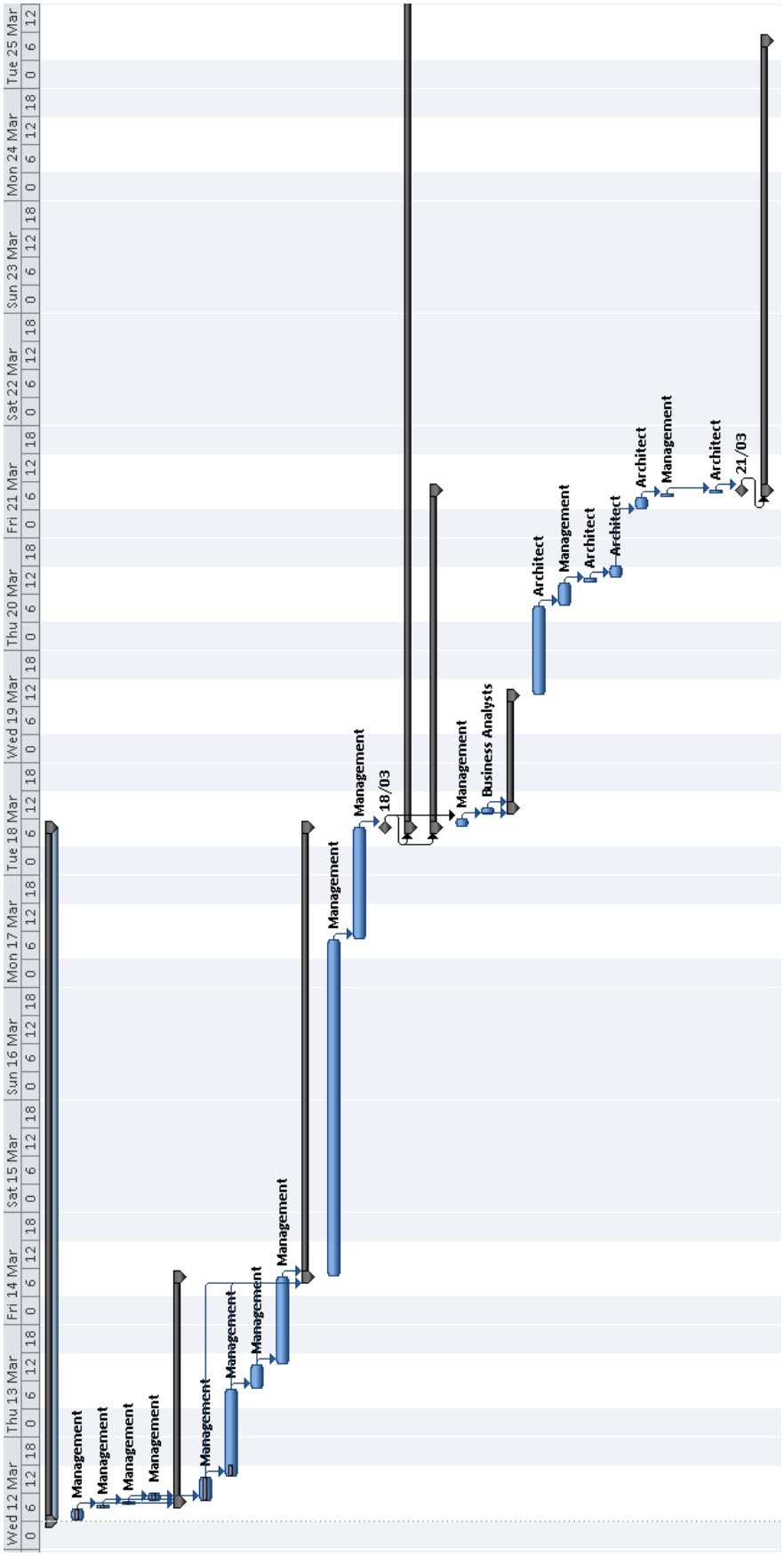



Figure A.2: Gantt Chart