Project Plan

For Environment Model Building Tool (EMBT)

Version 1.1

Submitted in partial fulfillment of the requirements of the degree of MSE

Esteban Guillen
CIS 895 – MSE Project
Kansas State University
1. Task Breakdown

1.1. Inception Phase

The inception phase is focused on defining the requirements for the project. A vision document will be developed to provide an overview of the project and to document requirements. A project plan will be developed to provide an estimate of the work load and give a schedule for completing project tasks. A software quality assurance plan will be created to describe the required documentation and the steps taken to ensure a quality product is produced.

The development of a prototype will also take place during the inception phase. The prototype is designed to show the feasibility of the project.

The inception phase is complete when the developer presents all required documentation and the committee members approve.

1.2. Elaboration Phase

The elaboration phase is focused on design issues. Revisions of documents from the inception phase will be completed at the committee members’ request. A formal requirements specification will be developed for a component of the project. An architecture design will be developed to describe the system. A test plan will be developed to describe how testing will be performed and reported. The technical inspectors will review the architecture design and report on their findings. Finally another prototype will be developed to demonstrate some of the more challenging product features.

The elaboration phase is complete when the developer presents all required documentation and the committee members approve.

1.3. Production Phase

The production phase is focused on implementation and testing. The developer will produce a component design to describe the system at a low level. Most of the developer’s time will be spent on developing the code. The code will be well documented and unit testing will be performed. The code will be tested to ensure all the requirements are met. All tests results will be evaluated and documented. The developer will also write a complete user manual which describes how to install and use the software.

The production phase is complete when the developer presents all required documentation, demonstrates the final project, and the committee members approve.

The Gantt chart below provides a schedule for completing the tasks in each phase.
2. Cost Estimate

2.1. COCOMO

The COCOMO model developed by Barry Boehm will be used to estimate project effort and project time. The Environment Model Building Tool will be an independent tool with a fair amount of flexibility and average complexity. The COCOMO model classifies such projects as “organic” and provides the following equations to calculate effort and time:

Effort = 3.2*EAF (Size)^1.05
Time (in months) = 2.5(Effort)^0.38

To calculate effort one needs to estimate the Size and EAF values. The Size is measured in KLOC. The EAF value stands for effort adjustment factor and is the product of 15 adjustment factors. Each adjustment factor is classified as very low, low, normal, high, or very high. The value of each adjustment factor lies within a range and the classification will determine where on the range the value will falls. The table below lists all the adjustment factors and their corresponding ranges.

<table>
<thead>
<tr>
<th>IDENTIFIER</th>
<th>EFFORT ADJUSTMENT FACTOR</th>
<th>RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>RELY</td>
<td>Required reliability</td>
<td>0.75 - 1.40</td>
</tr>
<tr>
<td>DATA</td>
<td>Database size</td>
<td>0.94 – 1.16</td>
</tr>
<tr>
<td>CPLX</td>
<td>Product complexity</td>
<td>0.70 – 1.65</td>
</tr>
<tr>
<td>TIME</td>
<td>Execution time constraint</td>
<td>1.00 – 1.66</td>
</tr>
<tr>
<td>STOR</td>
<td>Main storage constraint</td>
<td>1.00 – 1.56</td>
</tr>
<tr>
<td>VIRT</td>
<td>Virtual machine volatility</td>
<td>0.87 – 1.30</td>
</tr>
<tr>
<td>TURN</td>
<td>Computer turnaround time</td>
<td>0.87 – 1.15</td>
</tr>
<tr>
<td>ACAP</td>
<td>Analyst capability</td>
<td>1.46 – 0.71</td>
</tr>
<tr>
<td>AEXP</td>
<td>Applications experience</td>
<td>1.29 – 0.82</td>
</tr>
</tbody>
</table>
I classified the adjustment factors in the following way:

- RELY as low and a value of 1.0.
- DATA as high and a value of 1.1.
- CPLX as normal and a value of 1.4
- TIME as normal and a value of 1.3
- STOR as low and a value of 1.1.
- VIRT as low and a value of 0.95.
- TURN as low and a value of 0.9.
- ACAO as high and a value of 0.8.
- AEXP as normal and a value of 1.0.
- PCAP as normal and a value of 1.1.
- VEXP as normal and a value of 1.0.
- LEXP as high and a value of 1.0.
- MODP as high and a value of 0.9.
- TOOL as high and a value of 0.9
- SCED as high and a value of 1.15.

The EAF value evaluated to 1.54. I estimated the size to be 2500 LOC based on the current prototype and similar examples. The effort evaluates to:

\[ \text{Effort} = 3.2 \times 1.54 \times 2.5^{1.05} = 12.9 \text{ staff months} \]

The time can now be calculated as:

\[ \text{Time} = 2.5 \times 12.9^{0.38} = 6.6 \text{ months} \]

The 12.9 months for effort seems a little high for a school project, but I believe that figure is a little inflated since the COCOMO model is designed for large project with more than one developer. This project will only have one developer and will not rely on any other applications to function, so I would estimate that 8-9 staff months for Effort would be a more realistic value.
3. **Architecture Elaboration Plan**
   The following items must be complete before the second presentation is made.

3.1. **Revision of Vision Document**
   Changes to the requirements or project scope since the first presentation must get updated in the vision document.

3.2. **Revision of Project Plan**
   Changes to the schedule of the project must be updated in the project plan. The time and cost estimates will be revised using a bottom-up approach based on project progress. There will also be an Implementation Plan section added to the Project Plan document.

3.3. **Architecture Design**
   The developer must have a strong understanding about how to build the system. All interfaces will be defined and UML diagrams will be used.

3.4. **Development of Prototype**
   The prototype will demonstrate the critical requirements (defined in the Vision document) to demonstrate they can be implemented.

3.5. **Test Plan**
   A test plan will document the tests that are to be performed to ensure the requirements are meet.

3.6. **Formal Technical Inspections**
   The architecture design will be inspected by Cem Oguzhan and Kevin Sung. Both inspectors will produce a report on their inspection findings.

3.7. **Formal Requirements Specification**
   At least one part of the project will be formally specified using a methodology such as OCL. I will formally specify some of the laws of physics for objects that are in the environment. For example objects should not be floating in mid air.