CHAPTER 4
ANALYSIS AND DESIGN

4.1 Analysis

Determine the critical path using Critical Path Method, through several steps. The table below will explain each steps, with assumption, data already exists and stored.

Data that will be used in this project as follows:

Table 4.1: List of Project Activities

<table>
<thead>
<tr>
<th>CODE</th>
<th>TASK</th>
<th>DURATION</th>
<th>PREDECESSOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Product designing</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Basic research</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Product analysis</td>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>Product model</td>
<td>3</td>
<td>A</td>
</tr>
<tr>
<td>E</td>
<td>Sales brochure</td>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>F</td>
<td>Cost analysis</td>
<td>3</td>
<td>C</td>
</tr>
<tr>
<td>G</td>
<td>Product testing</td>
<td>4</td>
<td>D</td>
</tr>
<tr>
<td>H</td>
<td>Sales training</td>
<td>2</td>
<td>B,E</td>
</tr>
<tr>
<td>I</td>
<td>Price fixing</td>
<td>1</td>
<td>H</td>
</tr>
<tr>
<td>J</td>
<td>Project Reporting</td>
<td>1</td>
<td>F,G,I</td>
</tr>
</tbody>
</table>

From the data above, to search the critical path of the project, required several steps of completion. The steps below:

4.1.1 Iteration

In this process, the step that will be done is drawing all of the activities become two parts that is left (as task) and right (as predecessor). To connect related activities. Do the looping until it's done. Pick which activity that isn’t
connected to the right side. The result of iteration is to determine position of activity on the network diagram.

Illustration 4.1: Iteration 1

From the illustration above, is obtained the results of iteration 1 are activity A and activity B. Because those activities aren’t connected with any other activities.
Illustration 4.2: Iteration 2

From the illustration above, is obtained the results of iteration 2 are activity C, D, and E. Because those activities aren’t connected with any other activities.

Illustration 4.3: Iteration 3

From the illustration above, is obtained the results of iteration 3 are activity F, G, and H. Because those activities aren’t connected with any other activities.
From the illustration above, is obtained the results of iteration 4 is activity J. Because that activity isn’t connected with any other activities.

From the illustration above, is obtained the results of iteration 5 is activity J. Because that activity isn’t connected with any other activities.

From all of the iterations above, is obtained the position of activities on the diagram, as follows

### 4.1.2 Forward Pass

In this process, is to calculate the duration from start to finish. Algorithm in this process is a new activity can be started if the predecessor is already finished, except the initial activity, because the initial activity don’t have
predecessor. The earliest finish time of an activity is equal to the earliest start time plus the duration of the predecessor.

If an activity has two or more past merging activities, then the earliest start time of that activity is equal to the biggest earliest finish time from the past activities.

Below is how to forward pass a project:

4.1.3 Backward Pass

In this process is the opposite of forward pass. Because this process calculates duration from finish to start. From the past processes, is obtained the value from each activities. Algorithm in this process is the latest start time of an activity is equal to the latest finish time of an activity minus the time period of the activities. If an activity split into two activities or more, then the latest finish time of an activity is equal to the latest start time of the smallest next activities.

Below is how to calculate backward pass of a project:
4.1.4 Determine the Critical Path of a Project

To determine an activity which become the critical path is the difference between forward pass and backward pass is equal to zero.

From an activity that chosen to become the critical path, on the diagram will be formed a path that named the critical path of project management.
4.2 Desain Use Case Diagram

From the diagram above, user will input the amount of tasks that will be processed. Then input the task’s name, predecessor, and duration of each task. User can show the data that already input and immediately process the data that already input and will be produce the output in the form of a diagram.