3.9 Introduction to project management

Introduction
All IT development requires a management method. Knowledge and understanding of the product development life cycle (PDLC) should be used as a framework to develop an IT solution for the internal assessment. It is recommended that this topic is covered before students start work on the project.

IT concepts to address in this topic

Theoretical fundamentals
- Client, end-user, developer
- Data collection techniques for content and product design, citing of sources
- Role of testing and processes used
- Technical and end-user documentation (manuals)
- End-user training

The product development life cycle (PDLC)
- Investigation of existing system(s)
- Feasibility study
- Requirements specification
- Project schedule
- Product design
- Product development and technical documentation
- Client and end-user evaluation

Practical techniques
- Appropriate design techniques
- Data capture
- Product testing and debugging

What is PDLC?

Product development life cycle (PDLC) is the complete process of creating and bringing a new product into use. It includes the following 7 steps:

1. Investigation of existing system(s)
2. Feasibility study
3. Requirements specification
4. Project schedule
5. Product design
6. Product development and technical documentation
7. Client and end-user evaluation

1. Investigation of existing system(s)
Here we need to identify the owner or client of the existing product. The stage includes the information attainment using interviews, observation, questionnaires, sampling, discussion and reports so that designer, (or the product developer), can understand how the current system(s) works. This is like a fact finding mission. In so doing, the developer or designer will understand what the problem is and what will is/are required to solve it.

2. Feasibility study
Feasibility study looks at the present system, the requirement that it was intended to meet, problems in meeting these requirements, new requirements that have come since it was implemented and briefly investigates alternative solutions. The aim of this stage is to make sure that if product development is to continue, then it must be feasible (possible)
- Legally
- Organizationally and socially
- Technically
- Economically
A recommended solution is normally recommended at the end of this stage with an outline of functional specifications: this information is normally given to the management in forma of an oral presentation who then decide whether to proceed or not.

3. Requirements specification

If it is decided that the product development goes on, then formal requirements of the proposed product are written down as requirement specification. The first thing here is to formally define the extended product: defining diagram helps the programmer to see the components. Decisions of what the program should do are also made. The client is the main contributor here because it is the client who approves and gives what he or she requires the product to do to him or her.

Example of definition diagram:

If you were asked to write a program which would compute the cost per square meter of living space for a house, given the dimensions of the house, the number of stories, the size of the non-living space, and the total cost of the land, you would know that any noun or adjective is input or output and any verb is process.

**Definition Diagram**

<table>
<thead>
<tr>
<th>Input</th>
<th>Processing</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Width of the house</td>
<td>Calculate cost per metre</td>
<td>Gross footage</td>
</tr>
<tr>
<td>Length of the house</td>
<td>Calculate living area</td>
<td></td>
</tr>
<tr>
<td>Number of stories</td>
<td>Calculate gross footage</td>
<td></td>
</tr>
<tr>
<td>Size of nonliving space</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Selling price, less land</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4. Project schedule

1. *Timeline* can be drawn to schedule a project
2. A *calendar of events* can also be put in place to schedule a project
3. *CHECKLIST* or check sheet to follow through the project requirement criteria. It can also be taken as To Do List.
4. A *Gantt* chart can be used to display both the estimated and actual durations for activities. A Gantt chart is an integral part of a project management tool that may also provide guidance on how to put right any deviation from schedule.

A Gantt chart is a graphical representation of the *duration of tasks against the progression of time*. A Gantt chart is a useful tool for planning and scheduling projects.
EXAMPLES OF CALENDAR OF EVENTS:

Fig. 1

MAR 28 - Friday @ Martinez, CA... Gary Villalba’s Final Day of Work for the Contra Costa County Service Office after 20 Years of Service!
APR 1 - Tuesday @ 1830 in Danville... East Bay Stand Down Public Community Information & Planning Meeting - Veteran's Building
APR 3 - Thursday @ 1700 in Danville... General Membership Dinner Meeting @ Crow Canyon Country Club in Danville
APR 8 - Tuesday @ 1730 in Danville... Veteran's Day Advance Planning Meeting - Danville Veteran's Memorial Building - Lounge - 1 hour
APR 9 - Wednesday @ 1200 in Danville... Exchange Club Lunch Meeting at FAZ Restaurant - Guests & VNVDV WELCOME - Lunch $18
APR 14 - Monday @ 1730 in Pleasanton... WCF Kentucky Derby Advance Planning Meeting at Sandee Wiedmann's home (see flyer)
APR 17 - Thursday @ 1800 in Danville... VNVDV "Board of Directors' Business Meeting @ Veteran's Memorial Building in Danville
APR 18 - Friday @ 1800 in Diablo... Danville / Sycamore Valley Rotary "Charter Night" at Diablo Country Club - Black Tie Event
APR 19 - Saturday @ 1000 in Dublin... Camp Parks Combat Support Training Center - "Army Reserve Centennial Celebrations (10:30pm)
APR 24 - Sunday @ 0900 in Danville... AHF Foundation "Board of Directors' Meeting @ Estabrook's Shed in Danville

Fig. 2

Fig. 3
EXAMPLES OF GANTT CHARTS:

Fig. 1

**Gantt Chart: Reverse Engineering Project**

<table>
<thead>
<tr>
<th>Tasks</th>
<th>February</th>
<th>March</th>
<th>April</th>
<th>May</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assign Teams</td>
<td>19</td>
<td>0</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Select Reverse Engr. Project</td>
<td>22</td>
<td></td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>Write Proposal</td>
<td></td>
<td>9</td>
<td>23</td>
<td>30</td>
</tr>
<tr>
<td>Make Charts &amp; Diagrams</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mechanical Dissection</td>
<td>4</td>
<td>9</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Component Sketching</td>
<td></td>
<td>12</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Computer Modeling</td>
<td></td>
<td>9</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>Materials Analysis</td>
<td></td>
<td>11</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>Writing Final Report</td>
<td></td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
</tbody>
</table>

- Intermediate Report Section Due
- Final Report Due

Fig. 2

![Gantt Diagram](image)
5. Product design

Product design phase focuses on **how** the product requirements, identified in the requirement specification stage will be met.

In design, decide about the input requirements, variables as well as control structures by keeping in mind the output required. Develop a detailed logic plan using the following tools to group the program's activities into sections and also devise a method of solution or algorithm for each section. Finally come up with a test plan.

In this phase the focus is on four main areas:
- User interface design
- Data input design
- Data Output design
- Data processing design

To assist in the design phase we use the following design tools (also called CASE tools)

1. **ER diagrams**
2. **Data flow diagram**, 
3. **Structure charts**, 
4. **Decision Trees**, 
5. **Decision tables**, 
6. **Pseudocode** 
7. **flowcharts** 
8. **PERT charts**

1. **ER diagrams**

**Entity-relationship Diagram (ERD)** illustrates the logical structure of a product. It is usually used in database design.

Generally E-R Diagrams require the use of the following symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity</td>
<td>An entity is an object or concept about which you want to store information. <em>(Noun)</em></td>
</tr>
<tr>
<td>Relationship</td>
<td>Relationships illustrate how two entities share information in the database structure. <em>(Verb)</em></td>
</tr>
<tr>
<td>Attribute</td>
<td>An attribute is used to explain the entity <em>(Adjective</em> i.e. describes)*</td>
</tr>
</tbody>
</table>

**EXAMPLES**
2. Data flow diagram (DFD)

Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs. A data flow diagram (DFD) is a graphical representation of the flow of data between processes. In other words, it shows:

- What goes in
- How it is changed
• What comes out
• How it is stored

EXAMPLES of DFDs

3. Structure charts (Also called a decomposition chart)
Structure charts is a chart which shows the breakdown of a system to its lowest manageable levels. An organization Chart is an example of this *(Example Sandford Organization chart?)*

4. **Decision Trees**

A *decision tree* is a decision support tool that uses a tree-like model of decisions and their possible consequences. They are used for classification (categorization) and prediction.
5. **Decision tables**

- A decision table lists causes and effects in a matrix. Each column represents a unique combination.
- Purpose is to structure logic

<table>
<thead>
<tr>
<th>The four quadrants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conditions</td>
</tr>
<tr>
<td>Actions</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Entrance charge</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>40 or over?</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Employed?</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>$5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$</td>
<td>$</td>
</tr>
<tr>
<td>$2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25 years</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

6. **Product development and technical documentation**

Here we translate the design into an application using a programming language - if the product is a computer program, or use an application software's development tool - if the product is an application software. We create the user interface and write code. As we develop the product, we include internal documentation - comments and remarks within the code, if the product is a computer program that explains the purpose of code statements. The documentation which is written in technical, complex terms and which is meant to assist the product developer in maintaining the product is called **technical documentation**.

**External documentation** includes user manuals and anything that is not the actual code or is part of the listing. This should also include materials that are placed on a website such as FAQs (frequently asked questions) and help areas. User manuals, FAQs and any documentation that is written in simple, non technical terms which is meant to assist the end user in running the product is called **user documentation**.
7. Client and end-user evaluation
This is the last stage in product development. Apart of evaluation, the product is also tested here, just before evaluation. There are different types and methods of testing product. Generally a product will undergo:
- Technical testing – to check if the product is technically functional
- End user testing – to test if the product can be used effectively and to the specifications by the end user. Sometimes we also have client testing where the client checks if the product works as desired.

After each test, it is expected that the product will undergo some changes to reflect the corrections which occur during testing.

Evaluation occurs as the last stage of product development. Evaluation involves the end user and the client reviewing the product by checking if the product met the specifications which were earlier stated and whether it is actually solving the problems that it was supposed to solve. Later on the product may need some modifications and could be reviewed and maintained as an on-going task.

Reference:

