PROJECT ESTIMATION REPORT

For

Movie Recommendation System

By

DASARI JAYANTH
2019BCS-016

VEMULA SATYA PAVAN KALYAN
2019BCS-069

Table of Contents:

1. SIZE ESTIMATION

It’s important to understand that project size estimation is the most fundamental parameter. If this is estimated accurately then all other parameters like effort, duration, cost, etc can be determined easily.
At present two techniques that are used to estimate project size are:
1. Lines of code or LOC
2. Function point
For the purpose of our report, we will be using the Function Point metric as it will be difficult to estimate the LOC at beginning.

1.1 Function Point Metric

Function point metrics proposes that size of the software project is directly dependent on various functionalities it supports. More the features supported the more would be the size. The size of a software product is directly dependent on the number of different high-level functions or features it supports.This assumption is reasonable, since each feature would take additional effort to implement.This technique helps determine size of the project directly from the problem specification so is really helpful to project managers during project planning while determining size.

Step 1:UFP (Unadjusted Function Point) Computation

INPUTS:

1. Personal Details of the User (During Sign Up)
2. Mobile Number and Password (During user Login)
3. User_Id and Password(Admin)
4. Adding new Movie details.
5. Movie Search keywords(Titles,People,Genres)

Total Number of Inputs = 5

OUTPUTS:

1. Confirmation Message: User successfully registered
2. Confirmation Message: User successfully Logged-in
3. Movie Information like cast,plot,genre tags
4. Recommendations based on movie/keywords searched
5. Error Message: Unsuccessful Sign Up
6. Error Message: Unsuccessful Login(for users & Admin)
7. Error Message: Search Unsuccessful
8. Error Message: User Rating Submission Unsuccessful

Total Number Of Outputs: 8

INQUIRIES:

1. Request Movie Data.
2. Search History.
3. Request Movie Recommendations

Total Number Of Inquiries: 3

FILES:

1. Movie Database.
2. User Database.

Total Number of Files: 2

INTERFACES:

1. User Interface
2. Admin Interface

Total Number of Interfaces: 2

UFP = (Number of inputs)*4 + (Number of outputs)*5 + (Number of inquiries)*4 +(Number of files)*10 + (Number of interfaces)*10

UFP = (5*4) + (8*5) + (3*4) + (2*10) + (2*10)
UFP = 112

Step 2: Refine Parameters

TYPE	Simple	Average	Complex
No.of Inputs	3	4	6
No. of Outputs	4	5	7
No. of Inquiries	3	4	6
No. of Files	7	10	15
No. of Interface	5	7	10

Table: Cost of Various Entities for Different Complexities

INPUTS: 3 Simple + 2 Average
OUTPUTS: 6 Simple + 2 Complex
INQUIRIES: 3 Average
FILES: 2 Average
INTERFACES: 1 Average + 1 Complex

Refined UFP =((3*3) + (2*4)) + ((6*4) + (2*7)) + (3*4) + (2*10) + ((1*7) + (1*10))
= (9+8) + (24+14) + 12 + 20 + (7+10)
=104

Step 3: Refine UFP based on complexity of the overall project

Function Point Relative Complexity Adjustment Factors	Score
Requirement for reliable backup and recovery	5
Requirement for data communication	3
Extent of distributed processing	1
Performance requirements	4
Expected operational environment	3
Extent of online data entries	4
Extent of multi-screen or multi-operation online data input	4
Extent of online updating of master files	3
Extent of complex inputs, outputs, online queries and files	5
Extent of complex data processing	4
Extent that currently developed code can be designed for reuse	3
Extent of conversion and installation included in the design	2
Extent of multiple installations in an organisation and variety of customer organisations	4
Extent of change and focus on ease of use	4
Degree of Influence	49

Technical Complexity Factor (TCF) = 0.65 + 0.01 * DI
= 0.65 + 0.49
= 1.14

Function Point = UFP * TCF
= 104 * 1.14
= 118.56

2. EFFORT AND TIME ESTIMATION

2.1 Cost Constructive Model(COCOMO):

The Constructive Cost Model is a procedural software cost estimation model developed by Barry W. Boehm.COCOMO is a regression model based on LOC, i.e number of Lines of Code.Boehm postulated that any software development project can be classified into one of the following three categories based on the development complexity.They are organic, semi detached, and embedded. This project belongs to the organic category the general form of the COCOMO expressions The basic COCOMO model is a single variable heuristic model that gives an approximate estimate of the project parameters. The basic COCOMO estimation model is given by expressions of the following forms:
Effort = a1 × (KLOC)a2 PM
Tdev = b1 × (Effort)b2 months
where,

- KLOC is the estimated size of the software product expressed in Kilo Lines Of Code.
- a1, a2, b1, b2 are constants for each category of software product.
- Tdev is the estimated time to develop the software, expressed in months.
- Effort is the total effort required to develop the software product, expressed in person-months (PMs).

2.2 Estimation of development effort:

= 2.4 * (KLOC)^1.05
= 2.4 * (1.1)^1.05
= 2.653 PM

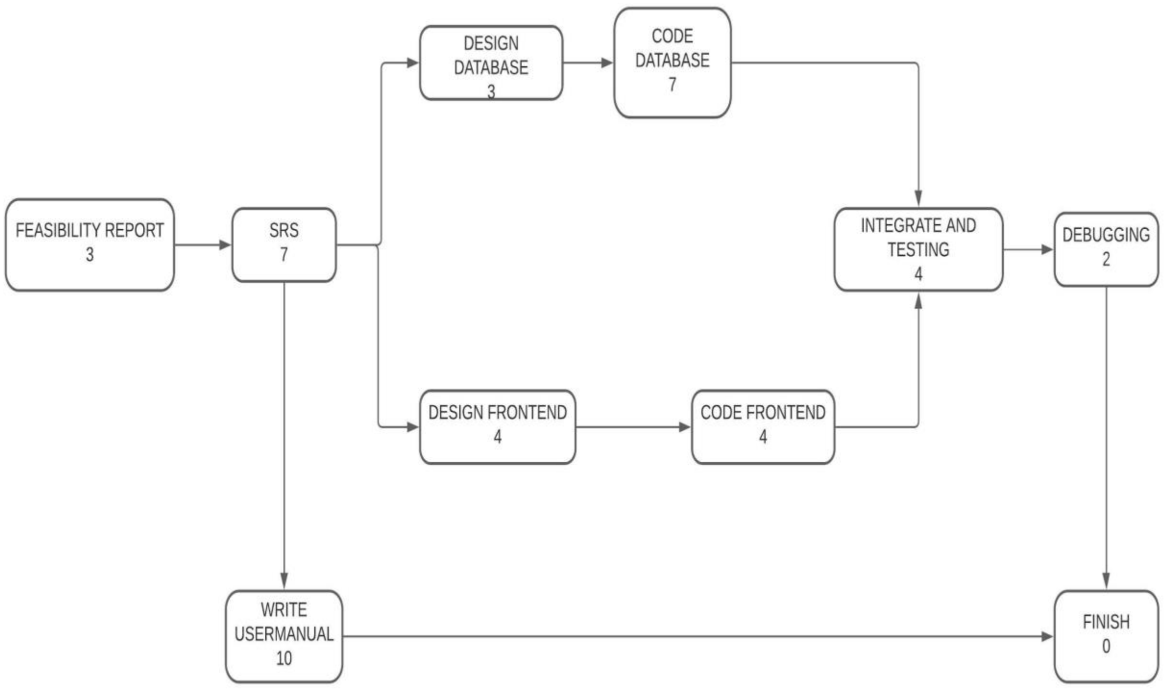
2.3 Estimation of development time:

= 2.5 * (EFFORT)^0.38
= 2.5 * (2.653)^0.38
= 3.622 months

3. PROJECT SCHEDULE BREAKDOWN

3.1 Activity Network

An activity network shows different activities making up a project ,their estimated durations, and their interdependencies.We use Activity On Node representation where each activity is represented by a rectangular (some use circular) node and the duration of the activity is shown alongside each task in the node.The inter-task dependencies are shown using directional edges.



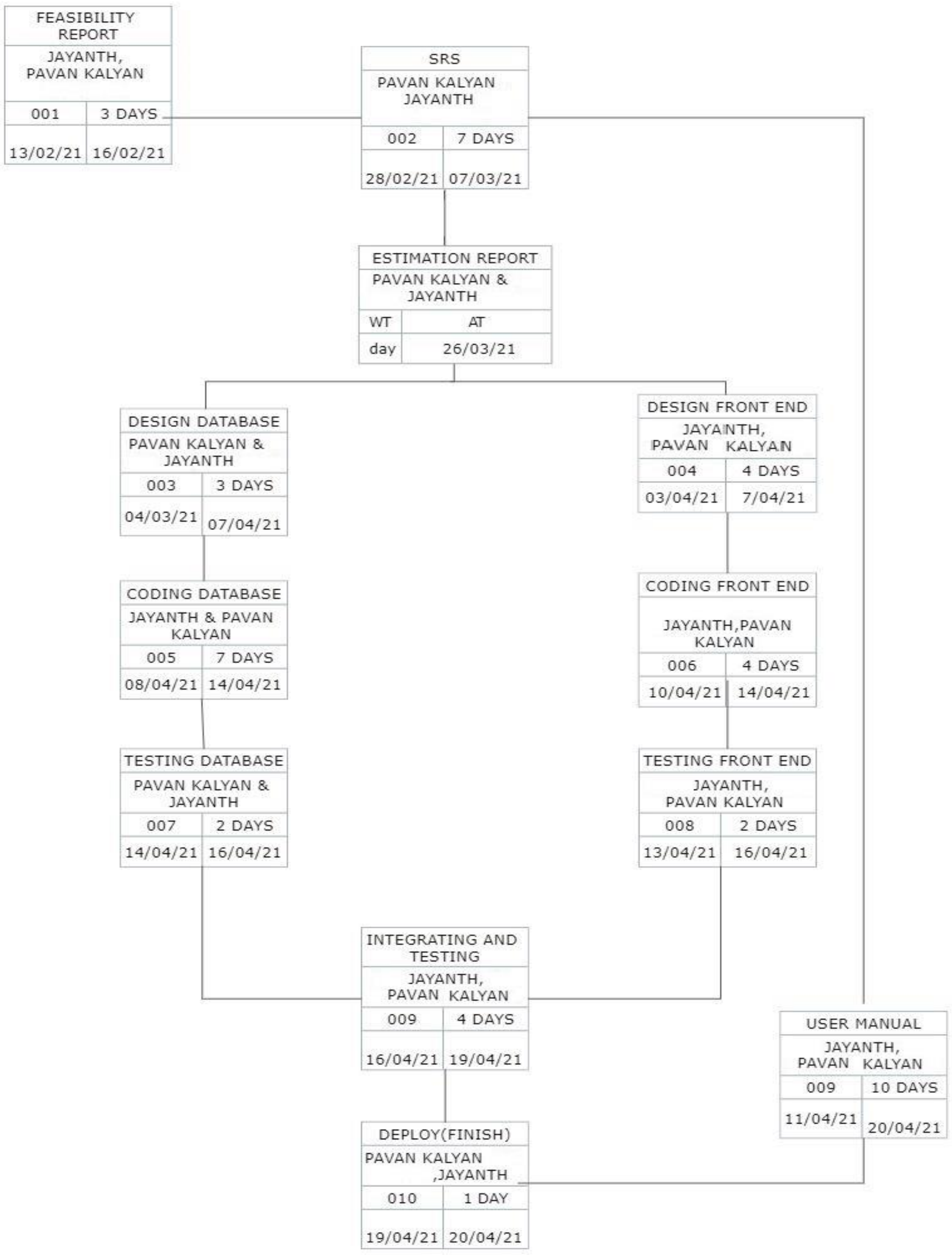
Activity On Node

The activity durations computed using an activity network are only estimated duration.
It is therefore not possible to estimate the worst case (pessimistic) and best case (optimistic) estimations using an activity diagram.Since, the actual durations might vary from the estimated durations, the utility of the activity network diagrams are limited.

3.2 Pert Chart

Program Evaluation and Review Technique (PERT) is a method used to examine the tasks that are in a schedule and determine a variation of the Critical Path Method (CPM). It analyzes the time required to complete each task and its associated dependencies to determine the minimum time to complete a project.Project evaluation and review technique (PERT) charts are a more sophisticated form of activity chart. PERT charts like activity networks consist of a network of boxes and arrows. The boxes represent activities and the arrows represent task dependencies.

A PERT chart represents the statistical variations in the project estimates assuming these to be normal distribution.PERT allows for some randomness in task completion times, and therefore provides the capability to determine the probability for achieving project milestones based on the probability of completing each task along the path to that milestone.
Each task is annotated with three estimates:
Optimistic (O): The best possible case task completion time.
Most likely estimate (M): Most likely task completion time.
Worst case (W): The worst possible case task completion time.


FEASIBILITY REPORT
JAYANTH, PAVAN KALYAN
001
3 DAYS
13/02/21
16/02/21
SRS
PAVAN KALYAN JAYANTH
002
7 DAYS
28/02/21
07/03/21
ESTIMATION REPORT
PAVAN KALYAN & JAYANTH
WT
AT
day
26/03/21
DESIGN DATABASE
PAVAN KALYAN & JAYANTH
003
3 DAYS
04/03/21
07/04/21
DESIGN FRONT END
JAYAINTH, PAVAN KALYAIN
004
4 DAYS
03/04/21
7/04/21
CODING DATABASE
JAYANTH & PAVAN KALYAN
005
7 DAYS
08/04/21
14/04/21
CODING FRONT END
JAYANTH,PAVAN KALYAN
006
4 DAYS
10/04/21
14/04/21
TESTING DATABASE
PAVAN KALYAN & JAYANTH
007
2 DAYS
14/04/21
16/04/21
TESTING FRONT END
JAYANTH, PAVAN KALYAN
008
2 DAYS
13/04/21
16/04/21
INTEGRATING AND TESTING
JAYANTH, PAVAN KALYAN
009
4 DAYS
16/04/21
19/04/21
USER MANUAL
JAYANTH, PAVAN KALYAN
009
10 DAYS
11/04/21
20/04/21
DEPLOY(FINISH)
PAVAN KALYAN ,JAYANTH
010
1 DAY
19/04/21
20/04/21