

Software Testing and Quality Assurance

Web-based Application

SUMMARY

Gantt Project

Prepared By:

Shabbab Algamdi 41986247,

Ashwin Chandrasekhar 42900602

1 INTRODUCTION

CURRENT DOCUMENT STATUS	
Version Number	2.5
Project Name	Gantt project
Product Size	3753 LOC
Type	Open Source Product
Description	Software Testing
Language	JAVA

Size of our project:

Due to the short time for the final project, we will be considering only the important features of the project like,

- Creating a Task.
- Creating a Resource.
- Creating Milestones.
- Creating a Gantt chart.
- Moving tasks and milestones.

List of selected testing techniques:

We have used Usage Based Statistical Testing (UBST) with White Box Testing (WBT).

Inside **UBST** we used **MUSA Operational Profile (OP)** for the probability statistics.

White Box Testing: Detailed analysis

- 1) Loop Testing.
- 2) Data Flow Testing.
- 3) Control Flow Testing.

For our project we will use the **Capability Maturity Model (CMM) level 2** for effectively achieving the quality for the software.

‘Capability Maturity Model’, developed by the SEI. It’s a model of 5 levels of organizational ‘maturity’ that determine effectiveness in delivering quality software. It is geared to large organizations such as large U.S. Defense Department contractors. However, many of the QA processes involved are appropriate to any organization, and if reasonably applied can be helpful.

The CMM level two model will consist of tables stating the implementation levels for the following operations: -

- 1) Requirements Management
- 2) Project Planning
- 3) Project monitoring and control
- 4) Measurement and analysis
- 5) Process and Product assurance.

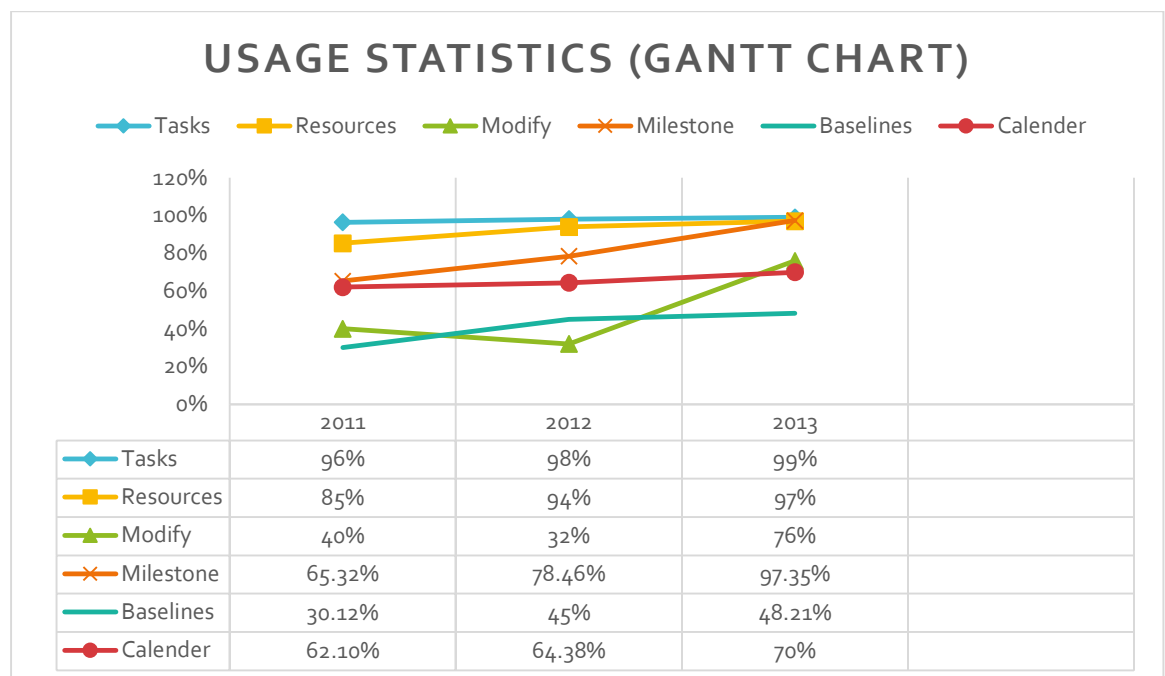
The implement levels will be high, medium and low for our project.

2 TESTING TECHNIQUES

2.1 USAGE BASED STATISTICAL TESTING: -

One important testing technique, the usage-based statistical testing with Musa's operational profiles (OPs) shares the basic model with partition testing techniques, and enhances it to include probabilistic usage information. We next describe Musa 1 OP and their usage in testing.

The usage statistics that we got from the users were as follows: -



The above statistics show that most of the usage is related to the tasks, resources, milestones of the Gantt chart. Therefore we perform testing in these critical areas.

Defects Found: 4

Total Number of test cases: 50

Number of people involved: 2

Number of hours spent: 12

2.2 WHITE BOX TESTING:

2.2.1 Loop Testing:

2.2.1.1 Simple Loop: -

In this we test the simple if else and for loops in our code.

Step 1: Skip the loop entirely

Step 2: Only one pass through the loop

Step 3: Two passes through the loop

Step 4: m passes through the loop where $n > m$

Step 5: $n-1$, n , $n+1$ passes through the loops

Where n it's a maximum no. of allowable passes through loops.

No of defects found: 3

No of test cases: 17

Number of people involved: 2

Number of hour spent: 4

Functions	Create Task	Create Resource	Assign Baseline	Modify the Gantt Chart	Dependencies
Test Cases	5	6	2	1	3

2.2.1.2 Nested Loop: -

In this we test the nested if else if loops in our code. All the IF-ELSE-IF loops pass the following steps: -

Step 1: Set all the other loops to minimum value and start at the innermost loop.

Step 2: Conduct simple loop test for the innermost loop and holding the outer loops at their minimum iteration parameter value.

Step 3: Performing test for the next loop and work outward.

Step 4: Continue until all the loops have been tested.

No of defects found: 7

No of test cases: 25

Number of people involved: 2

Number of hour spent: 4

Functions	Create Task	Create Resource	Assign Baseline	Modify the Gantt Chart	Dependencies
Test Cases	8	7	5	1	4

2.2.2 Data Flow Testing:

Data flow testing is another type of white box testing which looks at how data moves within a program. In data flow testing the control flow graph is annotated with the information about how the program variables are defined and used.

No of defects found: 12

No of test cases: 37

Number of people involved: 2

Number of hour spent: 15

Functions	Create Task	Create Resource	Assign Baseline	Modify the Gantt Chart	Dependencies
Test Cases	13	13	3	2	6

- **Control Flow Testing:** -

No of defects found: 6

No of test cases: 20

Number of people involved: 2

Number of hour spent: 10

Functions	Create Task	Create Resource	Assign Baseline	Modify the Gantt Chart	Dependencies
Test Cases	7	6	3	1	3

2.2.3 Defect Comparison:

DF=	0	1	2	3	4 - 9	9 - 15	Greater than 15	All
Module#	15	5	6	7	3	1	0	37
%	40.54	13.51	16.21	18.91	8.108	2.70	0	100
DF sum	0	5	12	21	19	16	0	73
%	6.84	16.43	28.78	26.03	21.92	0	0	100

3 RESULT ANALYSIS:

The two techniques we did above were UBST and WBT. Usage based statistical testing was useful in finding the critical areas affected with most number of defects. By using the information from UBST we were able to perform WBT successfully in those critical areas. WBT was the detailed and in depth testing of the critical modules, like testing the internal code of the functions. The WBT included testing techniques like Loop Testing, Branch Testing etc. as mentioned above. Also the above coverage techniques were used to see to it that all the codes in the critical area was covered successfully (because we were paranoid.)

3.1 EDUCATION BENEFITS:

There was a lot of benefits that we were able to grasp from this project. Again a special thanks to Dr. Jeff Tian for giving a project that was not just a project for the purpose of getting a grade, but also a project that will be very beneficial in the future. We list the benefits as follows: -

- 1) The major testing techniques (like we used UBST and WBT), their implementation, how they work in the actual testing environment.
- 2) The actual environment of doing a group project like strictly following the schedule and submitting the work on time (thanks to the milestones). Doing the specified work assigned to the group member in time. It was a great exposure to that kind of environment.
- 3) Also the workflow that should be followed while performing the testing and QA activities. We learned that in order to perform a successful testing the GOAL for the

testing should be clear in the first place. Also the requirements for performing the testing are all very important.

- 4) We came across a lot of errors while performing the testing and also learned about how to overcome such similar errors in the future while performing WBT or UBST.
- 5) The main thing that we can conclude after performing the testing is that, Simplicity is the key to perform an effective testing. The simpler the test cases the simple will be the workflow.

3.2 MAJOR FOLLOW UP ACTIONS:

The best way to improve the software or remove the defects after product release is through user feedback. Therefore there is a forum for the user who use Gantt in their day to day life. The forum mainly focus on the tricks to use the software, also the users can report if they have encountered any defect so that the developers can rectify those defects in their next release.

3.3 EFFORT COMPARISON:

UBST with Musa OP: 35 %

WBT

- Loop Testing: 20 %
- Data Flow Testing: 30 %
- Control Flow Testing: 30 %