Software Engineering Process
SOEN 6011

Summer 2016

Assignment 1

Group 6

Prepared by:

Vivek Khatri	27292848
Navneet Kaur	27676638
Sarvenaz Khaksarfard	40002880
Pardeep Kaur	40014804
Muhammad Raza Khan	27735987
Neha Kumari	27382073
Ankit Lathia	27378327
Mitesh Kaura	27284756

Project scope

This project is intended to develop a desktop and mobile application for Tic TAC Toe game. We would be developing 3 elementary applications to support desktop and android platforms. These are elementary applications for the game, which would support both, single player (vs. CPU) and multiplayer gaming. The project is intended to increase our understanding of software engineering processes during the whole development process and the final products are not intended to be marketed.

Objectives

Our main objectives for the project is to create 3 deliverables of TIC TAC TOE game, along with proper documentation containing of 5 supporting documents which includes the project plan, Requirements document, Domain Model, Test Report and State Diagram.
Deliverable 1 is a desktop application supporting a multiplayer version of the game.
Whereas, deliverable 2 is the same application as deliverable 1, but with support for Android platform.
Finally, Deliverable 3 would be using different heuristics to make the application support single player gaming against computer with different levels of difficulty.

Project Constraint

We prefer to follow an iterative and incremental process for the development, making it more as a unified process with focus on process adaptability and customer satisfaction; since we have deadlines for each deliverable in gap of 1 week.

The application requires a PC with Java Support or an Android mobile/tablet with a minimum RAM of 128 MB.

The skill set of programmers for Android platform in our team is limited. The exposure to Java swing is yet another domain in which we do not possess expertise.

Project Assumptions

We assume that the user tends to have basic knowledge of using Desktop/Laptop and Android mobiles/tablets.

Although, we would be adding a tab explaining the rules of the game, we assume that the user has prior experience or understating of playing TIC TAC TOE.
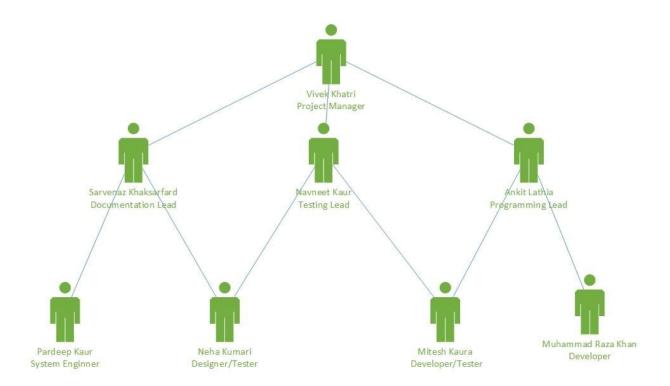
If the first click on the board is "**X**" then the other input on the board should be "**O**" and vice versa.

User should be allowed to reset or close the game window and proper acknowledgement should be received by the player.`
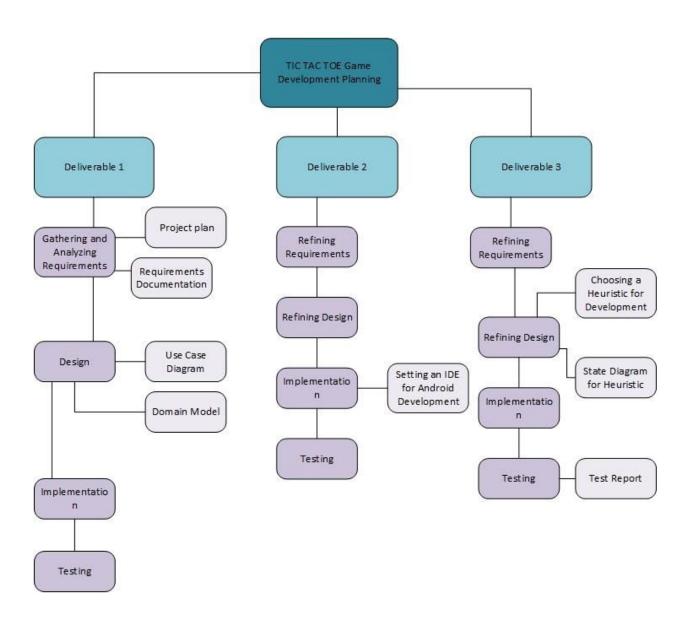
Project Plan

1. Team Organization

Name	Student ID	Team Roles	Responsibilities
Vivek Khatri	27292848	Project Manager	• Managing the team • Working on creating project plans and schedules • Documenting and designing
Sarvenaz Khaksarfard	40002880	Documentation Lead	• Documenting and designing • Managing people working on documentation • Working on creating final plans and documents
Pardeep Kaur	40014804	Systems Engineer	• Gathering Requirements • Designing • Testing
Muhammad Raza Khan	27735987	Developer	• Working on Eclipse Plugin • Programming • Work on the API
Navneet Kaur	27676638	Testing Lead	• Managing the Testers • Testing • Editing Documents
Neha Kumari	27382073	Designer and Tester	• Documenting • Testing • Designing
Ankit Lathia	27378327	Programming Lead	• Programming • Work On API • Managing the programmers
Mitesh Kaura	27284756	Developer and Tester	• Programming • Testing • Working on API

Work Breakdown Structure and Gantt Chart

The figures below illustrates the time and effort required for the work breakdown structure and Gantt charts shows the commonly used tasks for tracking project schedules. Gantt chart below show additional information about the various tasks or phases of the project, for example how the tasks are related to each other, how far each task has progressed, what resources are being used for each task and so on.

Work Breakdown Structure

Gantt Chart

Gantt Chart represents the tasks to be performed as planned and also shows the dependencies and durations of the tasks. Critical path is the path highlighted in red and defines tasks which are have no slack time. Delaying the critical path tasks leads to a delay in the whole project.

GANTT CHART

#	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1	📌	▲ D1 - JAVA standalone application	14 days	Tue 5/3/16	Fri 5/20/16	
2	📌	▲ Assignment 1	3 days	Wed 5/4/16	Fri 5/6/16	
3	📌	▲ Initiating	2 days	Tue 5/3/16	Wed 5/4/16	
4	📌	Assign team leader	1 day	Tue 5/3/16	Tue 5/3/16	
5	📌	Kick off meeting	1 day	Tue 5/3/16	Tue 5/3/16	
6	📌	Prepare project charter	1 day	Tue 5/3/16	Tue 5/3/16	
7	📌	Prepare project plan	1 day	Wed 5/4/16	Wed 5/4/16	
8	📌	Identify key stakeholders	1 day	Tue 5/3/16	Tue 5/3/16	
9	📌	▲ Planning	3 days	Wed 5/4/16	Fri 5/6/16	
10	📌	Hold project initiation meeting	1 day	Wed 5/4/16	Wed 5/4/16	5
11	📌	Prepare scope statement	1 day	Wed 5/4/16	Wed 5/4/16	6
12	📌	Prepare WBS	1 day	Thu 5/5/16	Thu 5/5/16	6,7,11
13	📌	▲ Assignment 2	5 days	Mon 5/9/16	Fri 5/13/16	
14	📌	▲ Gathering requirements	5 days	Mon 5/9/16	Fri 5/13/16	
15	📌	Document requirements	2 days	Fri 5/6/16	Mon 5/9/16	11,12
16	📌	Draw Use Case diagram	3 days	Fri 5/6/16	Tue 5/10/16	12
17	📌	▲ Assignment 3	2 days	Thu 5/19/16	Fri 5/20/16	
18	📌	Domain/Data model	2 days	Tue 5/10/16	Wed 5/11/16	15
19	📌	▲ Executing	8 days	Wed 5/11/16	Fri 5/20/16	
20	📌	▲ Design	3 days	Wed 5/11/16	Fri 5/13/16	
21	📌	Create design	1 day	Thu 5/12/16	Thu 5/12/16	18,15,16
22	📌	Assemble the design	1 day	Fri 5/13/16	Fri 5/13/16	21
23	📌	▲ Implementation	5 days	Mon 5/16/16	Fri 5/20/16	
24	📌	Software unit coding	3 days	Mon 5/16/16	Wed 5/18/16	22
25	📌	Software unit debugging	1 day	Thu 5/19/16	Thu 5/19/16	24
26	📌	▲ Testing	2 days	Thu 5/19/16	Fri 5/20/16	
27	📌	Unit testing	1 day	Wed 5/18/16	Wed 5/18/16	24
28	📌	System integration testing	1 day	Thu 5/19/16	Thu 5/19/16	27
29	📌	User acceptance testing	1 day	Fri 5/20/16	Fri 5/20/16	28

#	Task Mode	Task Name	Duration	Start	Finish	Predecessors
30	📌	▲ D2 - JAVA mobile application	5 days	Mon 5/23/16	Fri 5/27/16	
31	📌	▲ Executing	5 days	Mon 5/23/16	Fri 5/27/16	
32	🔧	▲ Design	2 days	Fri 5/13/16	Mon 5/16/16	
33	📌	Create design	1 day	Fri 5/13/16	Fri 5/13/16	21
34	📌	Assemble the design	1 day	Mon 5/16/16	Mon 5/16/16	33
35	📌	▲ Implementation	3 days	Tue 5/24/16	Thu 5/26/16	
36	📌	Software unit coding	2 days	Tue 5/17/16	Wed 5/18/16	34
37	📌	Software unit debugging	2 days	Thu 5/19/16	Fri 5/20/16	36
38	🔧	▲ Testing	5 days	Mon 5/23/16	Fri 5/27/16	
39	📌	Unit testing	2 days	Mon 5/23/16	Tue 5/24/16	36,37
40	📌	System integration testing	2 days	Wed 5/25/16	Thu 5/26/16	39
41	📌	User acceptance testing	1 day	Fri 5/27/16	Fri 5/27/16	40
42	📌	▲ Assignment 4	3 days	Wed 5/25/16	Fri 5/27/16	
43	📌	Test Report for 3 main classes	3 days	Wed 5/25/16	Fri 5/27/16	
44	📌	▲ D3 - Two player computer version	5 days	Mon 5/30/16	Fri 6/3/16	
45	📌	▲ Executing	5 days	Mon 5/30/16	Fri 6/3/16	
46	📌	▲ Design	2 days	Mon 5/30/16	Tue 5/31/16	
47	📌	Create design	1 day	Mon 5/16/16	Mon 5/16/16	33
48	📌	Assemble the design	1 day	Tue 5/17/16	Tue 5/17/16	47
49	📌	Heuristic state diagram	1 day	Tue 5/31/16	Tue 5/31/16	
50	📌	▲ Implementation	2 days	Wed 6/1/16	Thu 6/2/16	
51	📌	Software unit coding	2 days	Wed 5/18/16	Thu 5/19/16	48
52	📌	Software unit debugging	1 day	Fri 5/20/16	Fri 5/20/16	51
53	📌	▲ Testing	3 days	Wed 6/1/16	Fri 6/3/16	
54	📌	Unit testing	2 days	Wed 6/1/16	Thu 6/2/16	51
55	📌	System integration testing	1 day	Fri 6/3/16	Fri 6/3/16	54
56	📌	User acceptance testing	1 day	Fri 6/3/16	Fri 6/3/16	
57	📌	▲ Assignment 5	3 days	Wed 6/1/16	Fri 6/3/16	
58	📌	State diagram for heuristic	3 days	Wed 6/1/16	Fri 6/3/16	