



**COURSE: Software Engineering Processes  
(SOEN6011)**

**SUMMER – 2016**

**Tic-Tac-Toe**

**BY:  
Group-8 (LIONS)**

**PROFESSOR**

**NAME: Nicolangelo Piccirilli**

**EMAIL: nicolangelo.piccirilli@concordia.ca**

**DATE OF SUBMISSION**

**06 MAY, 2016**

**Team Members**

Dhruv	Ohri	27149220
Mona	Movahedi	26979327
Heya	Na	27208243
Parth	Patel	40014231
Prashanth	Ramesh	27382006
Sukhjit	Singh	40009670
Supreetha	Mudarajappa	27682417

## Contents

1. Project Charter .....	3
1.1 Project Scope .....	3
1.2 Project Objectives.....	3
1.3 Project Constraints.....	3
1.4 Project Assumptions .....	3
2. Project Plan.....	4
2.1 Team Members and responsibilities: .....	4
<i>Table 1- Roles and responsibilities</i> .....	4
2.2 Hardware and software requirements: .....	5
2.2.1 Hardware requirements: .....	5
2.2.2 Software requirements: .....	5
2.3 Project Schedule .....	5
<i>Fig. 1- Project Schedule</i> .....	5
2.4 Work Breakdown Structure:.....	6
3. Gantt chart based on WBS.....	7
<i>Fig.2- Gantt chart</i> .....	7
4. Communications Management.....	8
<i>Fig.3- Communication matrix</i> .....	8
5. Pert Chart.....	9
<i>Table 2 – Deliverable Schedule</i> .....	9
<i>Fig.4- Pert chart based on table 2</i> .....	9

# 1. Project Charter

## 1.1 Project Scope

The project consists of creating a simple board game called TIC-TAC-TOE that can run both on desktop and android devices. The project will be completed by 3rd June 2016. This game includes a 3X3 matrix board consisting of two players X and O, players marks the space in a 3×3 grid by taking their turns. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game. Players soon discover different ways of winning the game considering all possible permutation and combinations and due to the simplicity of the game it is helpful in teaching the concepts and increases the sportsmanship in players.

## 1.2 Project Objectives

This project is an implementation of Tic-Tac-Toe game, which is very popular and is fairly simple by itself. It is a two player game, it consists a board with n x n squares. The game is 3 x 3 squares. The goal of Tic-Tac-Toe is to be one of the players to get three same symbols in a row - horizontally, vertically or diagonally - on a 3 x 3 grid.

## 1.3 Project Constraints

1. Technical: The technologies to complete this project are readily available to us those of which include:
  - a. A computer system with pre-installed JDK, JRE, Eclipse and Android SDK.
  - b. An android device or a simulator to test the running version of the game on android OS.
  - c. Junit for testing
2. Time: In order to develop the game all the deliverables must meet their deadlines.
3. Resource: Enough resources should be available to design, code and test the project. Based on experience and availability of resources they should be distributed equally to ensure deadlines are met.

## 1.4 Project Assumptions

1. The layout should be 3X3 grid.
2. If one player makes the choice of using symbol X then symbol O should automatically be assigned to the second player
3. In order to win the game, player should get three same symbols either vertically, horizontally or diagonally
4. Devices are java enabled.

## 2. Project Plan

### 2.1 Team Members and responsibilities:

Role	Responsibilities	Members
Project Manager	<ul style="list-style-type: none"><li>● Leads the project's development and testing</li><li>● Elaborates on initial requirement (in meeting with developers)</li><li>● Verifies that all requirements received from the Product Manager are met by the code.</li></ul>	Supreetha
Software Quality Assurance/Tester	<ul style="list-style-type: none"><li>● Lead the team in producing and tracking the quality plan</li><li>● Alerts the team to quality problems</li><li>● Lead the team in defining and documenting its processes and in maintaining the process improvement process</li><li>● Establish and maintain the team's development standards</li><li>● Act as the team's inspection moderator</li><li>● Prepare and document test plan</li><li>● System test</li></ul>	Parth, Heya. Supreetha
Designer	<ul style="list-style-type: none"><li>● High level design</li></ul>	Prashanth Dhruv Mona
Developer	<ul style="list-style-type: none"><li>● Prototypes</li><li>● Detail design for use cases</li><li>● Development</li><li>● Write unit testing</li></ul>	Dhruv, Sukhjit Parth Prashanth
Requirements Engineer	Elicits and generates requirements and dependencies	Mona Supreetha Heya
Customer/Client	<ul style="list-style-type: none"><li>● Defines the requirements</li><li>● Review design</li><li>● Resolve escalated issues</li><li>● Acceptance test planning &amp; testing</li></ul>	Nicolangelo Piccirilli

*Table 1- Roles and responsibilities*

## 2.2 Hardware and software requirements:

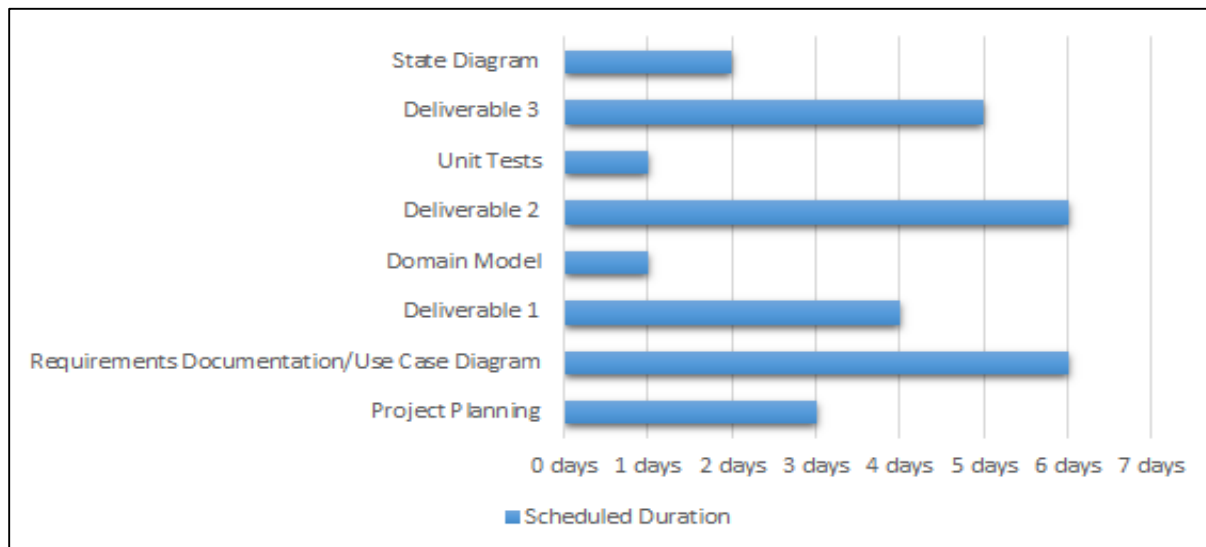
### 2.2.1 Hardware requirements:

- Desktop: RAM: 1GB RAM
- Processor: Intel Pentium 4.
- Hard disk: 40 GB Hard disk.
- Android Mobile: RAM 512 MB
- Processor: 1041 MHZ
- Operating system: Android 2.3 and above

### 2.2.2 Software requirements:

- JRE - Java runtime environment
- JDK - Java development kit
- Android Software development kit
- Android emulator

## 2.3 Project Schedule



*Fig.1- Project Schedule*

## 2.4 Work Breakdown Structure:

1. Create Java Application with Board and ability to draw X and O
  - 1.1 Requirements
  - 1.2 Design
  - 1.3 Implementation
    - 1.3.1 Front End
    - 1.3.2 Back End
  - 1.4 Testing
    - 1.4.1 System Test
  - 1.5 Deliver
2. Full game with two players
  - 2.1 Requirements
  - 2.2 Design
  - 2.3 Implementation
    - 2.3.1 Front End
    - 2.3.2 Back End
  - 2.4 Testing
    - 2.4.1 Test plan
    - 2.4.2 Unit Test
    - 2.4.3 System Test
    - 2.4.4 User Acceptance test
  - 2.5 Deliver
3. Desktop Or mobile app with two computer player against one computer player to beat human player
  - 3.1 Requirements
  - 3.2 Design
  - 3.3 Implementation
    - 3.3.1 Front End
    - 3.3.2 Back End
  - 3.4 Testing
    - 3.4.1 Test plan
    - 3.4.2 Unit Test
    - 3.4.3 System Test
    - 3.4.4 User Acceptance test
  - 3.5 Deliver

### 3. Gantt chart based on WBS



Fig.2- Gantt chart

The above figure explains the work break down structure, the activities and the timelines to complete the tasks. Each deliverable consists of tasks and sub-tasks with their respective deadlines. Each task has dependencies on the prior tasks and the lines indicate the dependencies of tasks.

There are 3 milestones which are scheduled at the end of each sprints which will be followed by requirement gathering, design, and implementation and testing.

## 4. Communications Management

Communication with the stakeholders is an essential element to successful delivery of the Project. The goal of communications is to provide accurate, timely, complete, and transparent communication on issues for which it is responsible. Since our project includes many different moving elements, we strive to provide the stakeholder with accurate and consistent information.

The following table shows the communication management matrix of our project.

Communication Type	Objective of Communication	Medium	Frequency	Audience	Owner	Deliverable
<b>Kick of Meeting</b>	Introduction of the project team and the project Review project objectives and management approach	Face to Face	once	Project Team POD	Project manager	Agenda Meeting minutes
<b>Project Team Meeting</b>	Review project status with the project team	Face to Face Conference Call	weekly	Project team	Project manager	Agenda Meeting minutes
<b>Software Design Meeting</b>	Discuss and develop software design solution for the project	Face to Face	As needed	Project technical staff	Designer	Project status report
<b>Project Status Meeting</b>	Report the status of the project to management	Face to Face Conference Call	monthly	PMO	Project manager	
<b>Project Status Reports</b>	Report the status of the project including activities, progress ,accomplishment cost and issues	Email	weekly	Project Team	Project manager	Project status report
<b>Discussion</b>	Arrange meeting, Communicate about issues	WhatsApp	As needed	Project team	Project manager	
<b>Documentation</b>	Sharing documentation	Google docs	As needed	Project team	Project manager	

*Fig.3- Communication matrix*



## 5. Pert Chart

The following table contains all the details associated with each deliverable

States	Deliverable	Duration	Start Date	Finish Date	Actual Start Date	Actual Finish Date	Constraint	Constraint Date
A	Project Plan	3 days	2016-05-02	2016-05-06	2016-05-03	2016-05-06	As soon as possible	-
B	Requirement Documentation/Use Case Diagram	6 Days	2016-05-06	2016-05-13	2016-05-07	2016-05-13	Must Finish on	2016-05-13
C	Domain Model	1 day	2016-05-13	2016-05-20	2016-05-14	2016-05-15	Start no earlier than	2016-05-14
D	Project Deliverable 1	4 days	2016-05-13	2016-05-20	2016-05-16	2016-05-20	Must Finish on	2016-05-15
E	Project Deliverable 2	6 days	2016-05-18	2016-05-27	2016-05-21	2016-05-16	Start no earlier than	2016-05-15
F	Unit Tests	1 day	2016-05-20	2016-05-27	2016-05-26	2016-05-27	Must Finish on	2016-05-26
G	Deliverable 3	5 days	2016-05-23	2016-06-01	2016-05-27	2016-06-01	Must Finish on	2016-05-27
H	State Diagram	2 days	2016-05-27	2016-06-03	2016-06-01	2016-06-03	Finish no later than	2016-06-02
							Must Finish on	2016-06-03

Table 2 – Deliverable Schedule

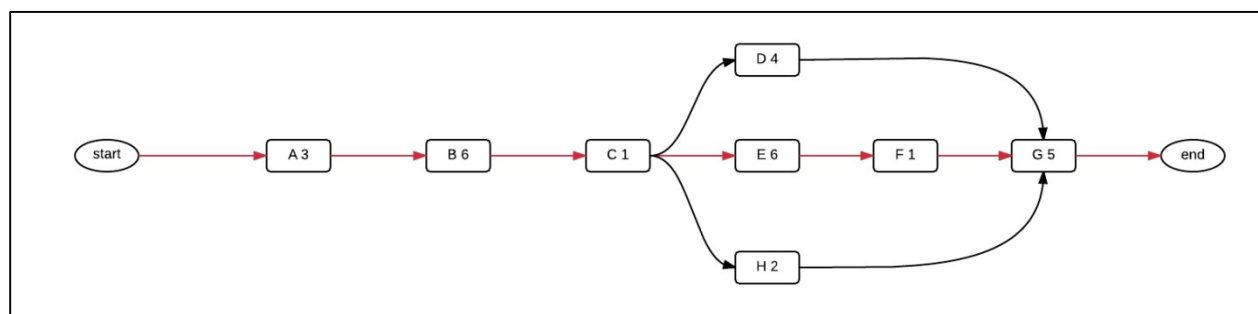


Fig.4- Pert chart based on table 2

### Description:

- The pert chart has been generated using table 2 which lists the number of states and the related deliverables/tasks associated with them. Each task has a start date and a finish date, these dates are the proposed dates that when a task can begin and end.
- Actual start date and finish date determines the real start and end date for a deliverable based on which the duration is calculated.
- Each deliverable has a constraint type and constraint date associated with it and for the project to be completed successfully must abide by these constraints. Name of the constraints are self-explanatory.
- The pert chart illustrates the tasks and their dependencies as we can see some tasks are dependent on other and cannot begin until the previous task has been completed, thus it used to plan and organize the phase of the project.
- The numbers associated with each state shows the execution time in days

**Critical Path Calculation:**

- We can calculate the critical paths of the task based on cumulative elapsed time.
- As one can see there are three paths to the end of the project which are as follows along with their elapsed times:
  - Path 1: A-B-C-E-F-G-E with cumulative elapsed time 21 days
  - Path 2: A-B-C-D-G-E with cumulative elapsed time 19 days
  - Path 3: A-B-C-H-G-E with cumulative elapsed time 17 days

***Thus the path which takes the maximum time is the critical path in the project which is path one.***

**Outcome:** If we can reduce the time of tasks in the critical path and manage them somehow we can reduce the time of the project