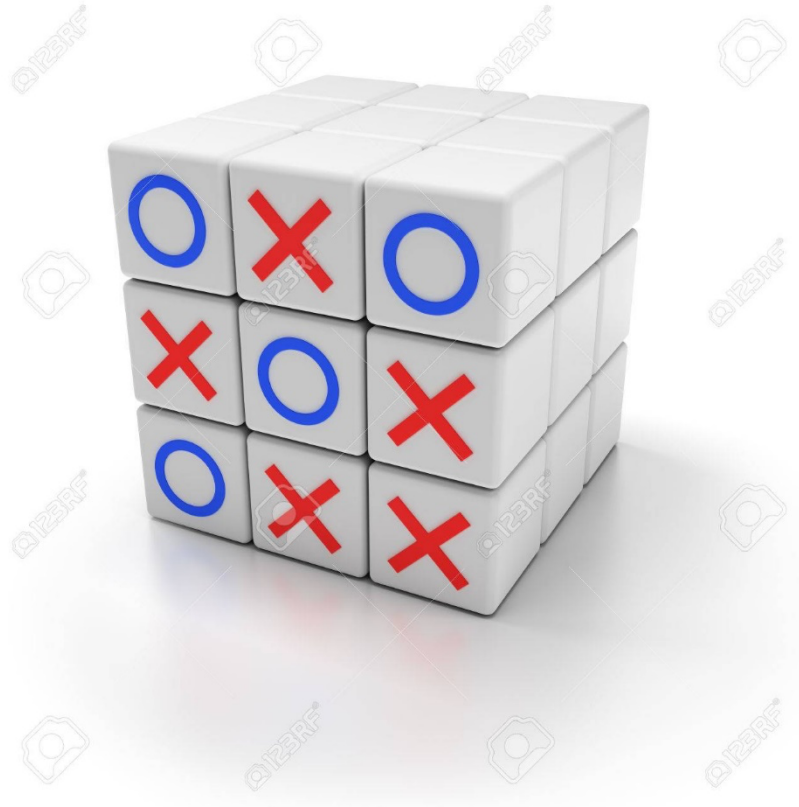


Score 82/100

# 3D Tic-Tac-Toe Project Report



**Prepared by  
Nick Toledo, Omar Flores,  
Stephen Lambert, Yash Shah  
at the  
University of Illinois Chicago**

**October 2019**

## Table of Contents

	List of Figures.....	6
	List of Tables.....	7
I	Project Description .....	8
1	Project Overview .....	8
2	The Purpose of the Project .....	8
2a	The User Business or Background of the Project Effort.....	8
2b	Goals of the Project.....	8
2c	Measurement.....	8
3	The Scope of the Work.....	9
3a	The Current Situation.....	9
3b	The Context of the Work.....	9
3c	Work Partitioning.....	10
3d	Competing Products .....	10
4	The Scope of the Product .....	10
4a	Scenario Diagram(s) .....	11
4b	Product Scenario List .....	11
4c	Individual Product Scenarios .....	12
5	Stakeholders .....	12
5a	The Client.....	12
5b	The Customer .....	12
5c	Hands-On Users of the Product .....	12
5d	Maintenance Users and Service Technicians.....	12
5e	Other Stakeholders.....	12
5f	User Participation.....	13
5g	Priorities Assigned to Users .....	13
6	Mandated Constraints.....	14
6a	Solution Constraints.....	14
6b	Implementation Environment of the Current System .....	14
6c	Partner or Collaborative Applications .....	14
6d	Off-the-Shelf Software.....	14
6e	Anticipated Workplace Environment .....	15
6f	Schedule Constraints.....	15
6g	Budget Constraints .....	15
7	Naming Conventions and Definitions .....	15
7a	Definitions of Key Terms .....	15
7b	UML and Other Notation Used in This Document.....	15
7c	Data Dictionary for Any Included Models .....	15

8	Relevant Facts and Assumptions.....	16
8a	Facts .....	16
8b	Assumptions .....	16
II	Requirements.....	16
9	Product Use Cases .....	16
9a	Use Case Diagrams .....	17
9b	Product Use Case List .....	18
9c	Individual Product Use Cases .....	18
10	Functional Requirements.....	20
11	Data Requirements .....	20
12	Performance Requirements .....	21
12a	Speed and Latency Requirements .....	21
12b	Precision or Accuracy Requirements .....	21
12c	Capacity Requirements .....	21
13	Dependability Requirements .....	22
13a	Reliability Requirements.....	22
13b	Availability Requirements.....	22
13c	Robustness or Fault-Tolerance Requirements.....	22
13d	Safety-Critical Requirements .....	23
14	Maintainability and Supportability Requirements.....	23
14a	Maintenance Requirements .....	23
14b	Supportability Requirements.....	23
14c	Adaptability Requirements.....	24
14d	Scalability or Extensibility Requirements.....	24
14e	Longevity Requirements .....	24
15	Security Requirements.....	24
15a	Access Requirements .....	25
15b	Integrity Requirements.....	25
15c	Privacy Requirements .....	25
15d	Audit Requirements.....	25
15e	Immunity Requirements.....	26
16	Usability and Humanity Requirements .....	26
16a	Ease of Use Requirements.....	26
16b	Personalization and Internationalization Requirements.....	26
16c	Learning Requirements .....	27
16d	Understandability and Politeness Requirements .....	27
16e	Accessibility Requirements.....	27

16f	User Documentation Requirements .....	28
16g	Training Requirements .....	28
17	Look and Feel Requirements .....	28
17a	Appearance Requirements .....	28
17b	Style Requirements .....	29
18	Operational and Environmental Requirements .....	29
18a	Expected Physical Environment .....	29
18b	Requirements for Interfacing with Adjacent Systems .....	29
18c	Productization Requirements .....	30
18d	Release Requirements .....	30
19	Cultural and Political Requirements .....	30
19a	Cultural Requirements .....	30
19b	Political Requirements .....	31
20	Legal Requirements .....	31
20a	Compliance Requirements .....	31
20b	Standards Requirements .....	31
21	Requirements Acceptance Tests .....	32
21a	Requirements – Test Correspondence Summary .....	32
21b	Acceptance Test Descriptions .....	32
III	Design .....	33
22	Design Goals .....	33
23	Current System Design .....	33
24	Proposed System Design .....	33
24a	Initial System Analysis and Class Identification .....	33
24b	Dynamic Modelling of Use-Cases .....	33
24c	Proposed System Architecture .....	33
24d	Initial Subsystem Decomposition .....	34
25	Additional Design Considerations .....	34
25a	Hardware / Software Mapping .....	34
25b	Persistent Data Management .....	34
25c	Access Control and Security .....	34
25d	Global Software Control .....	34
25e	Boundary Conditions .....	35
25f	User Interface .....	35
25g	Application of Design Patterns .....	35

26	Final System Design .....	35
27	Object Design .....	35
27a	Packages .....	36
27b	Subsystem I .....	36
27c	Subsystem II .....	36
27d	etc. ....	36
IV	Project Issues .....	36
28	Open Issues .....	36
29	Off-the-Shelf Solutions .....	36
29a	Ready-Made Products .....	36
29b	Reusable Components .....	36
29c	Products That Can Be Copied .....	37
30	New Problems .....	37
30a	Effects on the Current Environment.....	37
30b	Effects on the Installed Systems.....	37
30c	Potential User Problems .....	37
30d	Limitations in the Anticipated Implementation Environment That May Inhibit the New Product .....	37
30e	Follow-Up Problems .....	37
31	Migration to the New Product .....	38
31a	Requirements for Migration to the New Product.....	38
31b	Data That Has to Be Modified or Translated for the New System .....	38
32	Risks .....	38
33	Costs .....	38
34	Waiting Room .....	38
35	Ideas for Solutions .....	38
36	Project Retrospective.....	39
V	Glossary .....	39
VI	References / Bibliography .....	39
VII	Index .....	39

**List of Figures**

Figure 1- Work Context Diagram ..... 9

Figure 2 – Scenario Digram..... 11

## List of Tables

Figure 1 – Work Partitioning .....	10
------------------------------------	----

# **I Project Description**

## **1 Project Overview**

3D Tic Tac Toe is a multiplayer game available on a PC and a mobile device, similar to the original tic tac toe game. The game board is designed on a 3x3 cube, each player can rotate the board in any direction, place their characters, and see a live update of the game. The goal of the game is to connect three characters on any side of the board. The game is designed to be fun for every player and create income for the developers.

## **2 The Purpose of the Project**

The purpose of this project is to extend a single player game into a multiplayer game that has multiple dimensions. We plan to create a user-friendly application that gives live updates of the gameboard and detects the top three players that connect their characters.

### **2a The User Business or Background of the Project Effort**

Our main client's business is the gaming industry. The gaming industry allows for a lot of creativity and new development projects. This allows us to turn a two-dimensional game into a three-dimensional game that we are interested in developing but could also be interesting to the gaming community. Our plan is to produce a new multi-dimensional tic tac toe game that is significantly different from the original, with live updates of the game board. We were inspired to create this game, because we all enjoy playing video games, and thought it would be challenging to create a new game, that is significantly different from the original.

### **2b Goals of the Project**

The goal of this project is to make a new multiplayer and multi-dimensional game that is significantly different from the original, and that is profitable. The game aims to be fun and enjoyable for anyone over the age 3 and must be able to attract new customers and keep current customers. The game must be profitable, covering the cost of development, and future improvements.

### **2c Measurement**

The enjoyment of this game will be measured using different methods, for example, customer reviews, number of downloads, and user activity. It is important to gather this information, because then the developers can determine the success, or failure of the product, and find ways to improve it. Customer reviews will help improve the game, because it shows the developers how the users feel about the product, and if they are happy with the game. This will help the developers determine whether or not users will keep playing the game, and help developers determine other games that users would enjoy playing. The user activity measurement will determine how many users are playing the game weekly or monthly, and if they are most likely to continue playing the game. This will help developers find ways to keep new and existing users, because not every player will submit reviews.



### 3 The Scope of the Work

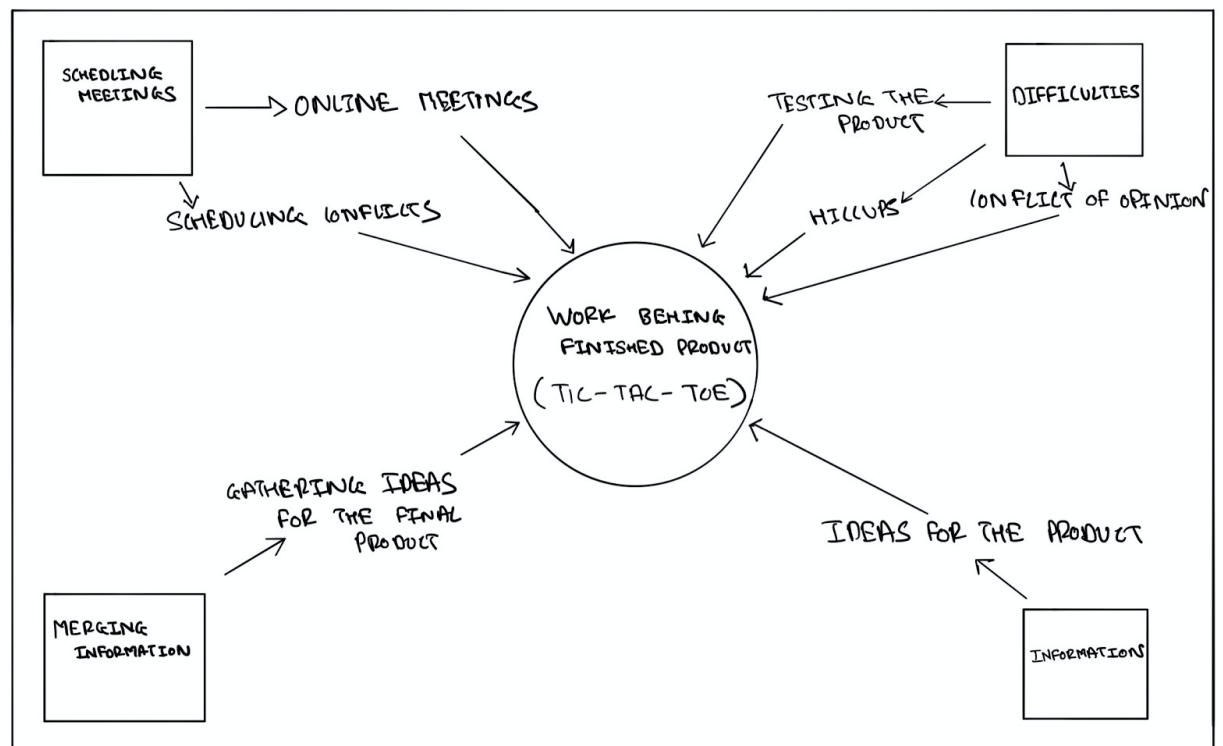
The main purpose of this game would be to entertain people on a regular basis and make an income from this game.

#### 3a The Current Situation

Tic-Tac-Toe is a classic game that can be traced back to ancient Egypt. This simple yet timeless game has managed to survive the test of time thanks to its easy to setup and play nature. All somebody needs to play the original game is pen, paper, and a friend. However, now that we are entering a more advanced age many people have moved away from traditional forms of entertainment in favor of digital entertainment. One of which is mobile games, according to an article by Medium, almost a third of the world's population will play a mobile game by the end of 2019.

As our entertainment interests grow more advanced, so should our entertainment. For this reason, we want to take advantage of the growing mobile game industry and bring back the classic game Tic-Tac-Toe as a more modern update, 3D Tic-Tac-Toe. Although the original required only pen, paper, and a friend to play, 3D Tic-Tac-Toe will only require a mobile phone or pc running our product which will find other players to play with. Most, if not a majority of the population has a smart phone, so we will be hitting a very large demographic. No longer will people need to scrounge for pen and paper to play Tic-Tac-Toe with their friends.

#### 3b The Context of the Work



### 3c Work Partitioning

Event Name	Input /Output	Summary
Ideas	Getting Ideas of peers (input)	Gathering all the ideas
Working on product	Applying all the ideas for the product (output)	Testing all ideas for the development of the final product
Problems in the development phase	dealing with opinion conflicts and issues (input)	Solving all the issues before the release of the final product

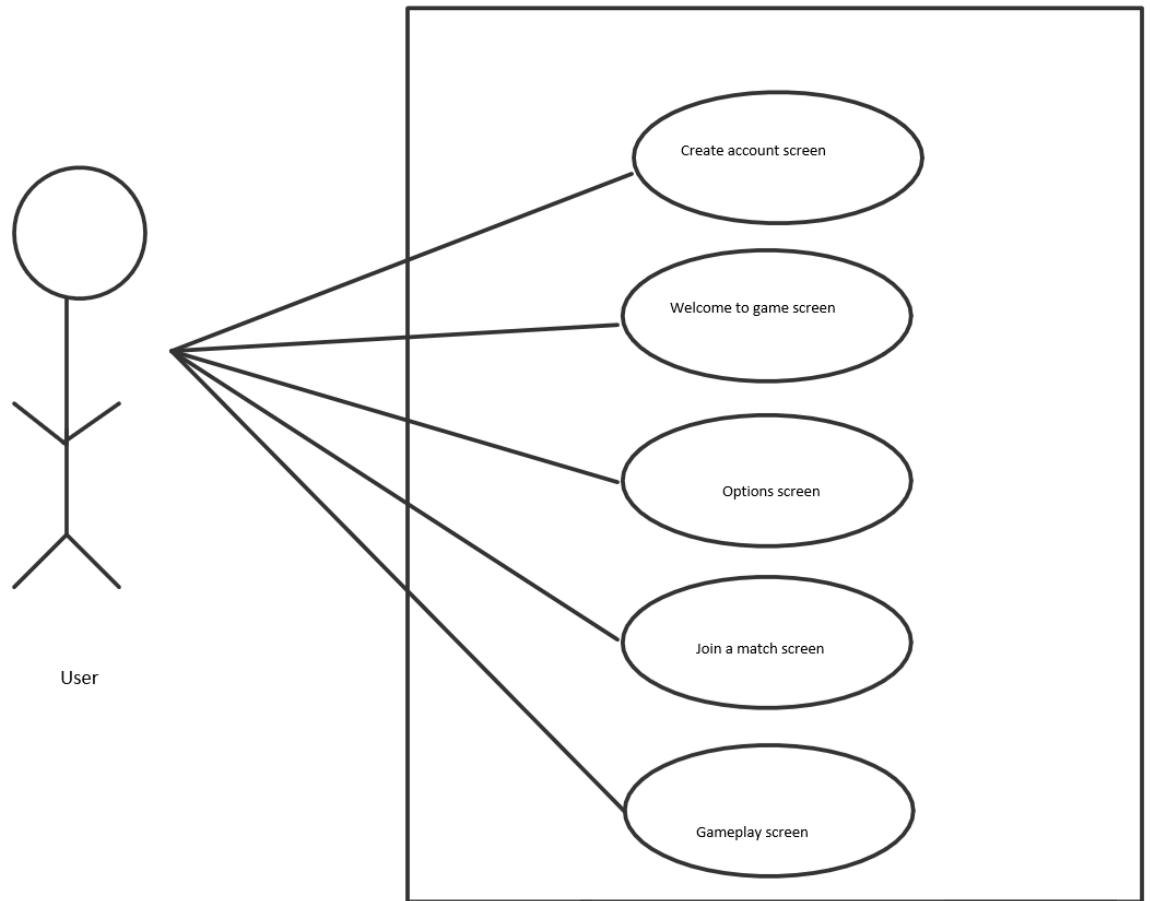
### 3d Competing Products

When we think of the direct competitors of the current product, there are no other products like this. But the original Tic tac Toe will be the direct competitor to this product. The only shortcoming of this product will be to understand the entire game due to its complicated working methodology.

## 4 The Scope of the Product

User's will be able to find and install our product either through the App/Play store for both Android and IOS as well as on Steam for PCs. The product will allow for account creation as well as linking pre-existing accounts from Steam, Play Games, or Game Center. Upon logging in users will be able to choose from one of the pre-existing game pieces to use in game. Afterwards user's will be able to start the game by choosing which game mode they would like to play, which will put the user in a queue to find an opponent. As user's continue playing, they will level up and gain experience points which can be used to unlock in game rewards.

#### 4a Scenario Diagram(s)



#### 4b Product Scenario List

##### PC Game

1. Create/Login account screen.
2. Welcome screen.
3. Options screen.
4. Join a match screen.
5. Gameplay screen.

##### Mobile Game

6. Create/Login account screen.
7. Welcome screen.
8. Options screen.
9. Join a match screen.
10. Gameplay screen.

#### **4c Individual Product Scenarios**

Users will start the application on the respective device. Once the application starts, the user will be prompted to create an account if the user doesn't have one already or sign in using a pre-existing account. When the user signs in, their online profile will display on the screen. Users will have the ability to mess with the game settings such as audio, key configurations, and other options that lead to a better user experience. Users will have the opportunity to join an online match or play with a friend. Both options will lead to the game populating.

### **5 Stakeholders**

#### **5a The Client**

We are the client. The goal is to make a product that will sell. Stakeholders/investors are clients too.

#### **5b The Customer**

The customer will be anyone who is into strategic games and those seeking a tougher challenge than the traditional Tic-Tac-Toe game.

#### **5c Hands-On Users of the Product**

Our main hands-on users for the product are going to be gamers. Specifically gamers that like strategic games/games in general. However, it doesn't necessarily mean that it's just limited to gamers as the product will be accessible by anyone who can operate a computer. There will be no discrimination within the users. Users can be any gender with the age range being open.

#### **5d Maintenance Users and Service Technicians**

Server administrators will be needed to make sure the servers are working properly so that there are no issues when it comes to users playing. Game developers will be needed to improve the user experience such as the visuals, game animations, etc. Software developers will be needed to help keep the game up to date and performing with no issues/bugs.

#### **5e Other Stakeholders**

**Tester** - Testers will have a small stake in the whole development of the game. Testing is important for companies who are on the verge of releasing a new product. As a creator of a product, proper testing must be done so there's no bugs or issues come release date. The knowledge needed by the testers will be the logical aspect of the game. Testers must know the rules of the game and the proper outputs of different moves. Aside from logic, testers must also inspect the visual aspect of the game. Do all components on the screen give a proper response, the transition from screens, etc. Any conflict between stakeholders who have an interest in the same knowledge will be handled by a meeting with other stakeholders and the clients.

**Marketing Experts** - Marketing experts will have a small stake in the whole development of the game. Marketing is important for companies who are on the verge of releasing a new product. Exposure will lead to success for products. The knowledge needed by marketing experts is our target audience. Based on our target audience, the marketing experts will offer suggestions for improvements, plan and implement social media or other marketing campaigns to help lure that audience. Aside from our target audience, marketing experts will be able to gather information on certain groups of people that aren't being attracted by the product and offer suggestions on how to attract those groups. Any conflict between stakeholders who have an interest in the same knowledge will be handled by a meeting with other stakeholders and the clients.

**Business Analysts** - Business analysts will have a small stake. Business analysts will help facilitate solutions for stakeholders. They are responsible for bridging the gap between IT and the business using data analytics to assess processes, determine requirements and deliver data-driven recommendations and reports to executives and stakeholders. Growth is what every company looks for and business analysts will be there to advance the company's future. Any conflict between stakeholders who have an interest in the same knowledge will be handled by a meeting with other stakeholders and the clients.

## **5f User Participation**

A couple of months before the product releases, there will be a beta. During this time, 100 users will be selected to participate in the beta. Selected users will be allowed to interact with the product and provide user feedback. Through this, we hope to address any issues or suggestions to enhance the user experience.

## **5g Priorities Assigned to Users**

**Key Users** - Gamers/Younger Audience

Gamers will be our key users. Preferably the younger gamers. Typically, this audience is already exposed to a variety of different games and won't mind adding another game to the collection. After marketing the product, it should be easier to attract this group of users.

**Secondary Users** - Strategic enthusiasts/Older Adults

Given that the product is based on the traditional Tic-Tac-Toe game, we expect that anybody who loves strategic games will come to appreciate a challenge. Doesn't necessarily have to be a hard core gamer. Older adults tend to have more time and are known to play more strategic games in their free time.

**Unimportant Users** - Children

Given that the product is a more complex game than the traditional Tic-Tac-Toe game, we don't expect infant children to grasp a full understanding of the product. Therefore, infants won't be much of a focus compared to our key users and secondary users.

## **6 Mandated Constraints**

### **6a Solution Constraints**

**Description:** Product shall run on average mobile specs.

**Rationale:** The client will not pay for a new mobile device to play 3D-Tic Tac Toe.

**Fit Criterion:** Mobile version of the game shall be playable on most if not all modern phones.

**Description:** Product shall run on both Android and IOS.

**Rationale:** The client will not switch phone OS just to play 3D Tic-Tac-Toe

**Fit Criterion:** Game will be available to download from both the Apple App Store, and the Android Play Store.

### **6b Implementation Environment of the Current System**

Our product will be available for both Android and IOS devices through their respective app stores as well as for PCs through Steam. The 3D graphics of the game would be designed using the cross-platform game engine Unity with C# as the scripting language. Using a Unit will allow us to design our product for both mobile and PC operating systems using most if not all of the same code base. This will help reduce the cost of development by avoiding having to develop our product for IOS, Android, and PCs natively.

### **6c Partner or Collaborative Applications**

The mobile version of our application will allow for users to log and save their data using both Google and Apple accounts. Likewise, PC users will also be able to log in using corresponding online retailers such as Steam.

### **6d Off-the-Shelf Software**

The PC version of the application will be taking advantage of the already existing online network for online distributors such as steam. Doing so users will be able to log in with their online distributor credentials and access their already existing friends list as well as gain achievements towards their account while using our application.

Likewise, the mobile version of the game will take advantage of Play Games on Android, and Game Center on IOS to provide user's access to their friends lists and achievements while using the application

## **6e Anticipated Workplace Environment**

The mobile version could be used in both quiet and loud environments; it must provide both audio and visual cues.

The application will be aimed towards all age demographics, it must be easy enough to be casually played by an older audience as well as children

## **6f Schedule Constraints**

Game must be released prior to the next major Android and IOS updates to avoid the application from becoming obsolete during production

## **6g Budget Constraints**

For the development of our product, we expect our budget to stay within our budget of \$400,000. Most of our budget will go towards the initial development of our product, as well as future product support. However, we will also need to budget for app testing to ensure that we are not releasing a buggy and faulty product into the market which could give our company a bad reputation. We will also need to pay for marketing to ensure we gain a big enough audience for our product.

# **7 Naming Conventions and Definitions**

## **7a Definitions of Key Terms**

PC: Personal computer, referring to both laptop and desktop computers.

IOS: iPhone Operating System used for mobile Apple devices.

Android: An open source mobile operating system, used by most mobile manufacturers.

Steam: A cloud boused game library which allows consumers to buy and download pc games using their steam account.

Android Play Store: An Android app equivalent to the Apple app store for Iphones.

## **7b UML and Other Notation Used in This Document**

This report adheres to the Version 2.0 OMG UML standard which is detailed by Fowler in [1].

## **7c Data Dictionary for Any Included Models**

User account info will consist of

- Username
- Email
- Password
- Possible linked third party account (Steam/Apple/Google)

Games will have the following data

- Record of icons for each player.
- State of the board.
- Score for each player.

## **8 Relevant Facts and Assumptions**

### **8a Facts**

Possible rating for our product would be E for everyone.

### **8b Assumptions**

User's should be familiar with using product platforms such as phones, tablets, and Pcs.

## **II Requirements**

*SV: Sections 9 and 10 deal with functional requirements. Sections 11 to 20 are a very thorough list of possible non-functional requirements, not all of which apply to every project. You should think carefully about each of these, form requirements if applicable, or write "Not Applicable" otherwise. See section 10 for the format of individual requirements. Section 21 documents the acceptance tests planned to verify the requirements – See that section for further details, and be aware that every requirement needs at least one verifying acceptance test ( though some tests may verify more than one requirement. )*

### **9 Product Use Cases**

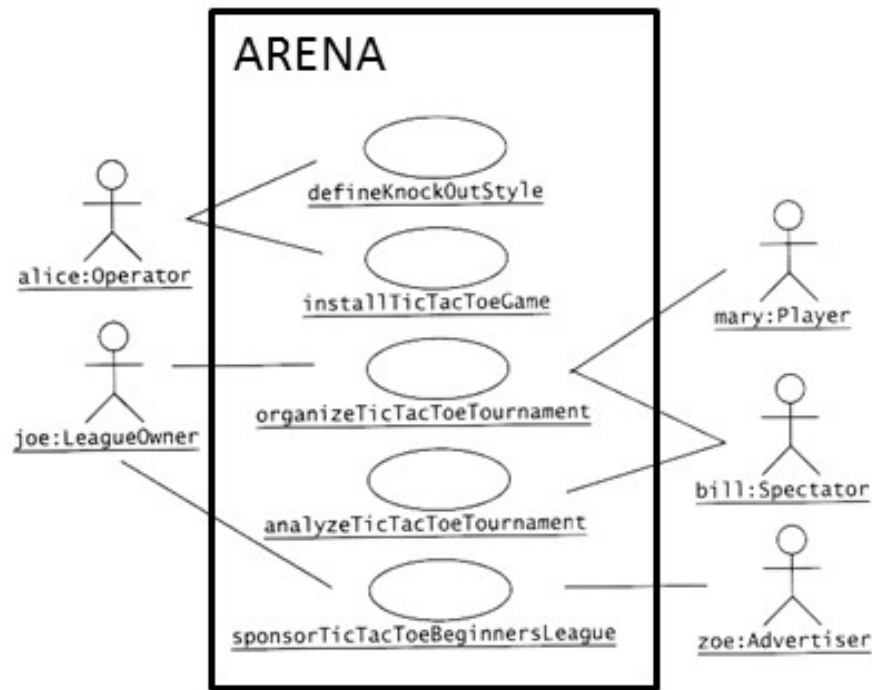
*SV: Product Use Cases are very similar to Product Scenarios, but in more formal detail. They serve as a first step towards developing functional requirements, and can aid in organizing requirements according to the use case(s) from which they were developed. See the CS 440 web site for a sample use-case form, with instructions.*



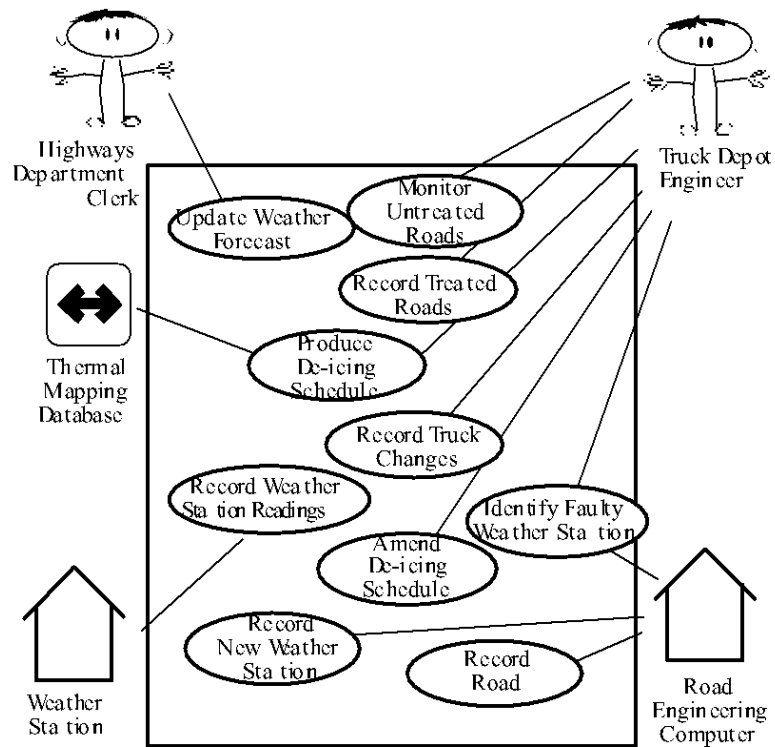
## 9a Use Case Diagrams

*SV: Use case diagrams list the use cases developed for a system, mark the boundary of what is internal or external to the system to be developed, and indicate which external entities (actors) are associated with each use case.*

### Examples



**Figure 1 - Sample Use Case Diagram from Bruegge & DuToit ( modified )**



**Figure 2 - Sample Use Case Diagram from Robertson and Robertson**

## 9b Product Use Case List

*SV: A list ( table ) of use cases is an alternative to the use case diagram, particularly when there are many use cases. There may be additional information in the table not found in the diagram, such as cross referencing to other sections or materials.*

## 9c Individual Product Use Cases

*SV: The following example was copied from “useCaseFormWithInstructions.docx”, available on the CS 440 web site. ( There is also a blank version available. )*



- *notes*
- *Alternatives and Exceptions may be listed either as separate use cases or as notes to a base case, depending on their significance and similarity.*
- *For regularly occurring periodic events, "time" can be listed as the initiating actor.*

## 10 Functional Requirements

*SV: Each requirement listed needs to have a unique identifier, a short name, a one- or two-sentence description, a rationale, a fit criteria, and reference to one or more acceptance tests to be used to confirm the completion of this particular requirement. The acceptance tests themselves are documented in section 0- See that section for further details. It is recommended to number the requirements according to their type, such as F-4 for the fourth functional requirement or U-2 for the second usability requirement. Functional requirements specifically deal with the functionality the system must have, and are generally derived directly from the steps the system takes during use cases.*

### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 11 Data Requirements

*SV: Data requirements deal with requirements that are somehow related to data, such as the definition of what is included in a "student record" or the acceptable form of an e-mail address or allowable range of certain data items.*

### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 12 Performance Requirements

### 12a Speed and Latency Requirements

*SV: Requirements specifying how fast ( or slow ) the product must operate or how much lag is allowable between stimulus and either initial response or task completion. Other timing-related requirements could go in this section.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 12b Precision or Accuracy Requirements

*SV: Self-explanatory. How accurate or precise must the system be.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 12c Capacity Requirements

*SV: Requirements regarding the largest “thing” the system must be able to handle, or perhaps how many things it can handle ( at once. ) Note: Requirements regarding how many things it can handle in a given time period would be a speed requirement, covered in section 12a above.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 13 Dependability Requirements

### 13a Reliability Requirements

*SV: Reliability relates to how frequently the system fails, ( either by shutting down or by delivering erroneous results ), and the consequences of those failures. These requirements may also address the conditions under which it is allowed to fail ( or not. ), See also availability and robustness in the following sections.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 13b Availability Requirements

*SV: Availability addresses the amount of time the system is running and available for use. It is affected by how often the system goes down ( reliability ), but also by the time required to bring the system back up again, the availability lost due to regularly scheduled maintenance down times, and the ability of the system to offer at least partial functionality in the face of failures or resource shortages. See also reliability and robustness.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 13c Robustness or Fault-Tolerance Requirements

*SV: This section deals with the system's ability to provide at least partial functionality in the face of failures or resource shortages, such as operating in offline mode when network connectivity is unavailable. See also reliability and availability.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### **13d Safety-Critical Requirements**

*SV: These requirements address potential harm to health, safety, or property, and may refer to relevant standards such as OSHA compliance.*

#### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## **14 Maintainability and Supportability Requirements**

### **14a Maintenance Requirements**

*SV: This section deals with the ease with which the system can be maintained, and possibly who will perform system maintenance and under what conditions. The ease of evolving the system into future versions may also be addressed here, or in a separate section ( not included in this template ) if that is a major concern.*

#### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### **14b Supportability Requirements**

*SV: What ongoing support is to be provided, e.g. through a help desk. See also training requirements in section 16g below.*

#### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

#### **14c Adaptability Requirements**

*SV: Description of other platforms or environments to which the product must be ported.*

##### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

#### **14d Scalability or Extensibility Requirements**

*SV: The ease of expanding the system to a larger capacity as the business grows.*

##### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

#### **14e Longevity Requirements**

*SV: This specifies the expected lifetime of the product.*

##### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### **15 Security Requirements**

*SV: Security requirements address who is allowed what type of access to the system, and what areas require special protection or diligence. In practice security requirements must often be written by security experts, and may refer to standards.*



### 15a Access Requirements

*SV: These requirements address who has access to what ( data or functionality ) and under what conditions or restrictions.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 15b Integrity Requirements

*SV: These requirements address the protection of data(bases) from intentional or accidental corruption, loss, or theft.*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 15c Privacy Requirements

*SV: These requirements address data that must remain confidential, such as medical records or other personally identifiable data. Laws often apply. (See also section 20.)*

#### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 15d Audit Requirements

*SV: This section applies when a system must provide support for transaction auditing, such as some financial or medical systems.*

**ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

**15e Immunity Requirements**

*SV: This section addresses the system's ability to resist viruses, worms, Trojan Horses, etc.*

**ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

**16 Usability and Humanity Requirements**

*SV: This section is concerned with requirements that make the product usable and ergonomically acceptable to its hands-on users.*

**16a Ease of Use Requirements**

*SV: This section addresses the ease with which the intended audience can use the system properly, and conversely the difficulty with which they can use it improperly.*

**ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

**16b Personalization and Internationalization Requirements**

*SV: This section addresses the ease with which the system can be configured for personal preferences, and for things such as language, currency, units, symbols, etc.*

**ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

**16c Learning Requirements**

*SV: Requirements related to how easy it is for the intended audience to learn to use the product.*

**ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

**16d Understandability and Politeness Requirements**

*SV: These requirements relate to how intuitively the intended audience understands what the program does, what its messages mean, and how to use it. Definitely related to ease of use, ( section 16a ), but more specifically addressing comprehension of the program output, instructions, and other messages.*

**ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

**16e Accessibility Requirements**

*SV: Requirements related to use of the product by individuals with disabilities.*

**ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## **16f User Documentation Requirements**

*SV: List of the user documentation to be supplied as part of the product.*

### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## **16g Training Requirements**

*SV: A description of the training needed by users of the product.*

### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## **17 Look and Feel Requirements**

### **17a Appearance Requirements**

*SV: These requirements address things such as the colors, fonts, and logos used, often to reflect corporate branding or similarity to related products. See also style in the next section.*

### **ID# - Name**

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 17b Style Requirements

*SV: Style requirements address the impression the product makes upon users, such as professionalism for a tax accounting package, friendliness for a children's game, or how "cool" it is for a teenage audience. Product packaging may also be addressed here, and/or appearance in the previous section.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 18 Operational and Environmental Requirements

### 18a Expected Physical Environment

*SV: These requirements relate to the physical environment in which the product will operate.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 18b Requirements for Interfacing with Adjacent Systems

*SV: This section describes the requirements to interface with partner applications and/or devices that the product needs to successfully operate.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 18c Productization Requirements

*SV: Requirements related to the distribution and/or installation of the product.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 18d Release Requirements

*SV: Specification of the intended release cycle for the product and the form that the release shall take.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 19 Cultural and Political Requirements

### 19a Cultural Requirements

*SV: This section contains requirements that are specific to the sociological factors that affect the acceptability of the product. If you are developing a product for foreign markets, then these requirements are particularly relevant. Bear in mind that “cultural groups” may also apply to population subgroups such as teenagers, the elderly, or ironworkers.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 19b Political Requirements

*SV: Requirements included strictly to make “the boss” happy, either internally to the development company, or internally to the client company, or possibly an external third party.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 20 Legal Requirements

### 20a Compliance Requirements

*SV: A statement specifying the legal requirements for this system, often referring to relevant laws and/or requiring approval by the legal department.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

### 20b Standards Requirements

*SV: These requirements specify documented standards to which the product must conform, as opposed to legal regulations.*

### ID# - Name

**Description:** Your description here . . .

**Rationale:** Your rationale here . . .

**Fit Criterion:** Your fit criteria here . . .

**Acceptance Tests:** List ID# and/or names here . . .

## 21 Requirements Acceptance Tests

*SV: Every requirement must have one or more acceptance tests associated with it, to confirm that the requirement has been met. At this point these tests are not yet completely specified – A one- or two-sentence description of each test will suffice. Note that some tests may verify more than one requirement, and that some requirements may require multiple tests for their confirmation.*

### 21a Requirements – Test Correspondence Summary

*SV: The following sample table is available from the CS 440 web site as “Sample Requirement Test Correspondence Table.xlsx” It is recommended that you work with the table in Excel, and then drag it into the document when it is completed. Depending on the number of requirements and/or tests included, it may be necessary to use multiple tables, and/or use landscape mode. Every row and every column of the table should include at least one X. Below the table list the ID #, name, and short description of each individual acceptance test.*

Test	Requirements																			
	Req 1	Req 2	Req 3	Req 4	Req 5	Req 6	Req 7	Req 8	Req 9	Req 10	Req 11	Req 12	Req 13	Req 14	Req 15	Req 16	Req 17	Req 18	Req 19	Req 20
Test 1	X																			
Test 2		X				X														
Test 3			X	X																
Test 4					X	X														
Test 5																				
Test 6																				
Test 7																				
Test 8																				
Test 9																				
Test 10																				
Test 11																				
Test 12																				
Test 13																				
Test 14																				
Test 15																				

*Table 1 - Requirements - Acceptance Tests Correspondence*

### 21b Acceptance Test Descriptions

*SV: Provide a brief description of each acceptance test. Detailed test specifications will appear in a separate document, which may be referenced here when available.*

**ID # - Name**



**Description:** Your description here . . .

### III Design

#### 22 Design Goals

*SV: Identify the important design goals that are to be optimized in the proposed design.*

Your text goes here . . .

#### 23 Current System Design

*SV: **IF** the proposed new system is to replace an existing system, then the current system should be described here. Otherwise insert a brief statement that there is no pre-existing system.*

Your text goes here . . .

#### 24 Proposed System Design

*This section will make heavy use of class diagrams, and also sequence and deployment diagrams where noted. However don't overlook finite state, activity, communication, or other diagram types as needed for effective communication.*

##### 24a Initial System Analysis and Class Identification

*SV: Perform grammatical and similar analyses to identify the most important and obviously needed classes, and to organize them into an initial class structure. An initial class diagram is appropriate, containing few if any internal details.*

Your text goes here . . .

##### 24b Dynamic Modelling of Use-Cases

*SV: Insert sequence diagrams of ( at least the most important ) use-cases, as a means of identifying other needed classes.*

Your text goes here . . .

##### 24c Proposed System Architecture

*SV: Identify the Software Architecture to be applied to this project, such as Client-Server, Repository, MVC, etc., along with justification for the choice.*

Your text goes here . . .

## 24d Initial Subsystem Decomposition

*SV: A slightly more detailed class diagram, showing the classes identified in sections 24a, 24b, and 0 above, partitioned into subsystems. For each subsystem provide a brief description of the subsystem, including its key responsibilities. There should still be few if any internal details.*

Your text goes here . . .

## 25 Additional Design Considerations

*SV: The sections listed here do not need to be presented in the order given, and may not all be relevant for any particular project. Those that are relevant can help identify additional classes that are needed as a result.*

### 25a Hardware / Software Mapping

*SV: This is particularly important for distributed systems, such as those employing a client-server architecture. Use a deployment diagram to indicate which subsystems are mapped onto which piece(s) of hardware, and what communication subsystems need to be added to the system as a result.*

Your text goes here . . .

### 25b Persistent Data Management

*SV: Document the classes and perhaps subsystems necessary to store persistent data when the system shuts down, and to restore that data when the system starts back up again.*

*Reiterate key data structures and information as necessary for the understanding of this design phase. Refer the reader back to the data dictionary in section **Error! Reference source not found.** to avoid undue repetition, while reviewing only the most relevant items here.*

Your text goes here . . .

### 25c Access Control and Security

*SV: Identify the access control and security concerns for this system, and the new classes and/or subsystems that must be added to handle those concerns.*

Your text goes here . . .

### 25d Global Software Control

*SV: Identify the global software control concerns for this system, and the new classes and/or subsystems that must be added to handle those concerns.*

Your text goes here . . .

## **25e Boundary Conditions**

*SV: Identify the boundary condition concerns for this system, and the new classes and/or subsystems that must be added to handle those concerns. In particular consider startup, shutdown ( normal or abnormal ), and the creation and/or maintenance of any configuration files, databases, or similar supporting data files.*

Your text goes here . . .

## **25f User Interface**

*SV: Include a preliminary user interface design here, possibly as a rough sketch or other mockup, in order to identify additional classes needed to implement the interface.*

Your text goes here . . .

## **25g Application of Design Patterns**

*SV: Any design patterns applied as a result of previous sections should have been addressed there, and identified as such at the time. Use this section to document only the additional design patterns that were not previously covered elsewhere. ( If any. )*

Your text goes here . . .

## **26 Final System Design**

*SV: Include here the final version of the overall system design, incorporating all the subsystems and classes added as a result of additional design considerations. Multiple diagrams may be needed, possibly starting with an overall package diagram showing all the different subsystems and the ( important ) classes contained within each one. Still not a lot of internal details.*

Your text goes here . . .

## **27 Object Design**

*This section documents the internal details of each class, to the extent that they can be designed at this time. Included should be the class interfaces ( public method signatures and responsibilities ) and constraints. It is probably best to break this section up into subsections corresponding to subsystems as documented above, and/or by ( Java ) packages if those are designed. It may also be appropriate to address additional design pattern considerations here, but not to the point of being redundant of previous documentation.*

*Certain methods, such as simple getters, setters, and constructors are not always documented, unless there is something special about them such as in the Singleton or Factory Method design patterns.*

## **27a Packages**

*SV: If the design involves assigning classes to packages ( .e.g Java packages ), then the packages to be created should be documented here.*

Your text goes here . . .

## **27b Subsystem I**

Your text goes here . . .

## **27c Subsystem II**

Your text goes here . . .

## **27d etc.**

Your text goes here . . .

# **IV Project Issues**

## **28 Open Issues**

*SV: Issues that have been raised and do not yet have a conclusion.*

Your text goes here . . .

## **29 Off-the-Shelf Solutions**

*SV: Discussion of products or components currently available that could either be incorporated into the new solution or simply used instead of developing ( parts of ) the new solution. The distinction between sections 35 a, b, and c is subtle, and not very important.*

Your text goes here . . .

## **29a Ready-Made Products**

*SV: Products available for purchase that could be used either as part of a solution or instead of ( a part of ) a solution.*

Your text goes here . . .

## **29b Reusable Components**

*SV: Similar to 35a, but for components such as libraries or toolkits instead of fully blown products.*

Your text goes here . . .

### **29c Products That Can Be Copied**

*SV: Products that could legally be copied would typically be past projects developed by the same development group, provided there were no restrictions that would prevent their reuse.*

Your text goes here . . .

## **30 New Problems**

*SV: The proposed new system certainly has its benefits, but it could also raise new problems. It is a good idea to identify any such potential problems early on, rather than being surprised by them later.*

### **30a Effects on the Current Environment**

*SV: Could the new system have any adverse effects on the working environment, e.g. the way people do their jobs?*

Your text goes here . . .

### **30b Effects on the Installed Systems**

*SV: Could the new system have any adverse effects on other hardware or software systems?*

Your text goes here . . .

### **30c Potential User Problems**

*SV: Could the new system have any adverse effects on the users of the software? Could users possibly have a negative response to the new system?*

Your text goes here . . .

### **30d Limitations in the Anticipated Implementation Environment That May Inhibit the New Product**

*SV: Are there any ( physical ) limitations in the expected environment that could inhibit the proposed product? ( e.g. weather, electrical interference, radiation, lack of reliable power, etc. )*

Your text goes here . . .

### **30e Follow-Up Problems**

*SV: Basically any other possible problems that could occur.*

Your text goes here . . .

## 31 Migration to the New Product

*SV: This section only applies when there is an existing system that is being replaced by a new system, particularly when data must be preserved and possibly translated / reformatted. Otherwise just write "Not Applicable" under section 38 and remove sections 38a and 38b.*

### 31a Requirements for Migration to the New Product

*SV: These are a list of requirements relevant to the migration procedures. For example a requirement that the two systems be run in parallel for a time until the client is satisfied with the new system and the users know how to use it.*

Your text goes here . . .

### 31b Data That Has to Be Modified or Translated for the New System

*SV: This section specifically addresses data that must be preserved and/or translated / reformatted during the migration process.*

Your text goes here . . .

## 32 Risks

*SV: Consideration of the potential risks that could cause the project to fail / underperform.*

Your text goes here . . .

## 33 Costs

*SV: An estimate of what it will cost to complete this project. Think not only in terms of dollars, but also time, resources, lost opportunities, etc.*

Your text goes here . . .

## 34 Waiting Room

*SV: This is a place to record ideas or wishes that will not be included in the current release of the product, but which might be worth reconsidering at a later date.*

Your text goes here . . .

## 35 Ideas for Solutions

*SV: When developing requirements only, it is not the role of the business analyst to dictate the implementation of the solution. However they can pass along any ideas they have here as suggestions to the developers. For CS 440 this report includes system and object design, so this section would make suggestions for implementation and*

*testing that would come after design, such as the use of a particular language, IDE, library, or other tools.*

Your text goes here . . .

## **36 Project Retrospective**

*SV: At the conclusion of the ( CS 440 ) project, reflect back on what worked well and what didn't, and how the process could be improved in the future.*

Your text goes here . . .

## **V Glossary**

*SV: The glossary is a more complete and inclusive dictionary of defined terms than that found in section I.7.a, the latter of which only covered the most important key terms needed to understand the report.*

Your text goes here . . .

## **VI References / Bibliography**

*This section describes the documents and other sources from which information was gathered. This sample bibliography was generated using the “Insert Citation” and “Bibliography” buttons in the “Citations & Bibliography” section under the “References” tab of MS Word. Creating new citations will not update this list unless you click on it and select “Update Field”. You may need to reset the style for this paragraph to “normal” after updating.*

- [1] Robertson and Robertson, Mastering the Requirements Process.
- [2] A. Silberschatz, P. B. Galvin and G. Gagne, Operating System Concepts, Ninth ed., Wiley, 2013.
- [3] J. Bell, "Underwater Archaeological Survey Report Template: A Sample Document for Generating Consistent Professional Reports," Underwater Archaeological Society of Chicago, Chicago, 2012.
- [4] M. Fowler, UML Distilled, Third Edition, Boston: Pearson Education, 2004.

## **VII Index**

*This section provides an index to the report. The sample below was generated using the “Mark Entry” and “Insert Index” items from the “Index” section on the “References” tab, and can be automatically updated by right clicking on the table below and selecting “Update Field”. To*

*remove marked entries from the document, toggle the display of hidden paragraph marks ( the paragraph button on the “Home” tab ), and remove the tags shown with XE in { curly braces. }*

Design ..... 61, 63  
Requirements ..... 35, 51, 58

Test..... 64, 65