***Ikon Wars Project Report***

**

***Prepared by***

***Nemil Shah, Wishy Parikh, Ruchit Patel, David Qiao***

***for use in* CS 440**

**at the**

**University of Illinois Chicago**

**Table of Contents**

[*REMOVE OR REPLACE ALL TEXT IN RED ITALICS BEFORE SUBMITTING REPORT* 2](#_30j0zll)

[*How to Use This Document* 2](#_3z7bk57)

[List of Figures 9](#_1fob9te)

[List of Tables 10](#_3znysh7)

[I](#_3znysh7) Project Description 11

[1](#_2et92p0) Project Overview 11

[2](#_tyjcwt) The Purpose of the Project 11

[2a](#_3dy6vkm) The User Business or Background of the Project Effort 11

[2b](#_1t3h5sf) Goals of the Project 11

[2c](#_4d34og8) Measurement 12

[3](#_17dp8vu) The Scope of the Work 13

[3a](#_3rdcrjn) The Current Situation 13

[3b](#_26in1rg) The Context of the Work 13

[3c](#_lnxbz9) Work Partitioning 16

[3d](#_35nkun2) Competing Products 18

[4](#_1ksv4uv) The Scope of the Product 18

[4a](#_44sinio) Scenario Diagram(s) 19

[4b](#_2jxsxqh) Product Scenario List 19

[4c](#_z337ya) Individual Product Scenarios 19

[5](#_3j2qqm3) Stakeholders 20

[5a](#_1y810tw) The Client 20

[5b](#_4i7ojhp) The Customer 20

[5c](#_2xcytpi) Hands-On Users of the Product 21

[5d](#_1ci93xb) Maintenance Users and Service Technicians 22

[5e](#_3whwml4) Other Stakeholders 22

[5f](#_2bn6wsx) User Participation 23

[5g](#_qsh70q) Priorities Assigned to Users 24

[6](#_3as4poj) Mandated Constraints 25

[6a](#_1pxezwc) Solution Constraints 25

[6b](#_49x2ik5) Implementation Environment of the Current System 26

[6c](#_2p2csry) Partner or Collaborative Applications 27

[6d](#_3o7alnk) Off-the-Shelf Software 28

[6e](#_23ckvvd) Anticipated Workplace Environment 29

[6f](#_ihv636) Schedule Constraints 30

[6g](#_32hioqz) Budget Constraints 31

[7](#_1hmsyys) Naming Conventions and Definitions 31

[7a](#_41mghml) Definitions of Key Terms 31

[7b](#_2grqrue) UML and Other Notation Used in This Document 32

[7c](#_vx1227) Data Dictionary for Any Included Models 33

[8](#_3fwokq0) Relevant Facts and Assumptions 34

[8a](#_1v1yuxt) Facts 34

[8b](#_4f1mdlm) Assumptions 34

[II](#_2u6wntf) Requirements 36

[9](#_19c6y18) Product Use Cases 36

[9a](#_3tbugp1) Use Case Diagrams 36

[9b](#_37m2jsg) Product Use Case List 38

[9c](#_1mrcu09) Individual Product Use Cases 38

[10](#_46r0co2) Functional Requirements 40

[11](#_2lwamvv) Data Requirements 41

[12](#_111kx3o) Performance Requirements 42

[12a](#_3l18frh) Speed and Latency Requirements 42

[12b](#_206ipza) Precision or Accuracy Requirements 43

[12c](#_4k668n3) Capacity Requirements 44

[13](#_2zbgiuw) Dependability Requirements 45

[13a](#_1egqt2p) Reliability Requirements 45

[13b](#_3ygebqi) Availability Requirements 46

[13c](#_2dlolyb) Robustness or Fault-Tolerance Requirements 47

[13d](#_sqyw64) Safety-Critical Requirements 48

[14](#_3cqmetx) Maintainability and Supportability Requirements 49

[14a](#_1rvwp1q) Maintenance Requirements 49

[14b](#_4bvk7pj) Supportability Requirements 50

[14c](#_2r0uhxc) Adaptability Requirements 51

[14d](#_1664s55) Scalability or Extensibility Requirements 52

[14e](#_3q5sasy) Longevity Requirements 52

[15](#_25b2l0r) Security Requirements 53

[15a](#_kgcv8k) Access Requirements 53

[15b](#_34g0dwd) Integrity Requirements 54

[15c](#_1jlao46) Privacy Requirements 55

[15d](#_43ky6rz) Audit Requirements 56

[15e](#_2iq8gzs) Immunity Requirements 57

[16](#_xvir7l) Usability and Humanity Requirements 57

[16a](#_3hv69ve) Ease of Use Requirements 57

[16b](#_1x0gk37) Personalization and Internationalization Requirements 59

[16c](#_4h042r0) Learning Requirements 60

[16d](#_2w5ecyt) Understandability and Politeness Requirements 61

[16e](#_1baon6m) Accessibility Requirements 62

[16f](#_3vac5uf) User Documentation Requirements 63

[16g](#_2afmg28) Training Requirements 64

[17](#_pkwqa1) Look and Feel Requirements 65

[17a](#_39kk8xu) Appearance Requirements 65

[17b](#_1opuj5n) Style Requirements 66

[18](#_48pi1tg) Operational and Environmental Requirements 67

[18a](#_2nusc19) Expected Physical Environment 67

[18b](#_1302m92) Requirements for Interfacing with Adjacent Systems 68

[18c](#_3mzq4wv) Productization Requirements 69

[18d](#_2250f4o) Release Requirements 70

[19](#_haapch) Cultural and Political Requirements 71

[19a](#_319y80a) Cultural Requirements 71

[19b](#_1gf8i83) Political Requirements 72

[20](#_40ew0vw) Legal Requirements 73

[20a](#_2fk6b3p) Compliance Requirements 73

[20b](#_upglbi) Standards Requirements 74

[21](#_1tuee74) Requirements Acceptance Tests 75

[21a](#_4du1wux) Requirements – Test Correspondence Summary 75

[21b](#_184mhaj) Acceptance Test Descriptions 76

[III](#_3s49zyc) Design 76

[22](#_279ka65) Design Goals 76

[23](#_meukdy) Current System Design 76

[24](#_36ei31r) Proposed System Design 76

[24a](#_1ljsd9k) Initial System Analysis and Class Identification 77

[24b](#_45jfvxd) Dynamic Modelling of Use-Cases 77

[24c](#_zu0gcz) Proposed System Architecture 77

[24d](#_3jtnz0s) Initial Subsystem Decomposition 77

[25](#_1yyy98l) Additional Design Considerations 77

[25a](#_4iylrwe) Hardware / Software Mapping 77

[25b](#_2y3w247) Persistent Data Management 78

[25c](#_1d96cc0) Access Control and Security 78

[25d](#_3x8tuzt) Global Software Control 78

[25e](#_2ce457m) Boundary Conditions 78

[25f](#_rjefff) User Interface 78

[25g](#_3bj1y38) Application of Design Patterns 79

[26](#_1qoc8b1) Final System Design 79

[27](#_4anzqyu) Object Design 79

[27a](#_2pta16n) Packages 79

[27b](#_14ykbeg) Subsystem I 79

[27c](#_3oy7u29) Subsystem II 79

[27d](#_243i4a2) etc. 79

[IV](#_j8sehv) Project Issues 80

[28](#_338fx5o) Open Issues 80

[29](#_1idq7dh) Off-the-Shelf Solutions 80

[29a](#_42ddq1a) Ready-Made Products 81

[29b](#_2hio093) Reusable Components 81

[29c](#_wnyagw) Products That Can Be Copied 81

[30](#_3gnlt4p) New Problems 82

[30a](#_1vsw3ci) Effects on the Current Environment 82

[30b](#_4fsjm0b) Effects on the Installed Systems 83

[30c](#_2uxtw84) Potential User Problems 83

[30d](#_1a346fx) Limitations in the Anticipated Implementation Environment That May Inhibit the New Product 84

[30e](#_3u2rp3q) Follow-Up Problems 84

[31](#_2981zbj) Migration to the New Product 85

[31a](#_odc9jc) Requirements for Migration to the New Product 85

[31b](#_38czs75) Data That Has to Be Modified or Translated for the New System 86

[32](#_1nia2ey) Risks 86

[33](#_47hxl2r) Costs 87

[34](#_2mn7vak) Waiting Room 88

[35](#_11si5id) Ideas for Solutions 89

[36](#_3ls5o66) Project Retrospective 90

[V](#_4kx3h1s) Glossary 90

[VI](#_302dr9l) References / Bibliography 91

[VII](#_1f7o1he) Index 91

### 

### Project Description

## Project Overview

Game named ‘Ikon Wars’ is to be developed two team competes with different avatars aka as characters. The main purpose of this game development is to combine the different qualities of players. For example, the best player is who has education skills like critical and analytical thinking, problem solving skills, interpersonal skills like teamwork, communication, has sarcastic values with how the character is designed etc. The game will enable the player to enhance various skills that he might use in various situations of real life.

The target audience for this game will be for teenagers with age range of 15- 30 years. However, there is no age restriction used. The game is played on online server. Design for this game should contain artistic and creativity for designing characters and making features look more interesting.

Further, we may need to add some problems, puzzles, hints which involves educational values to the game. The title of the game is ‘Ilkon Wars’, but doesn’t have need to have aggressive violence to defeat the player. A player can’t challenge other players with his guns, arms etc, but also with other strategies and powers that can defeat other players without involving violence.

Overall, the project is designed with various features like character build, different levels of the game, real world maps etc. In this game, character designing all will be built before the start of the game. Once the game starts, player gets certain frame time to solve puzzles, riddles and other skills to get the artifacts in their possession. After that, players with similar skill level will be matched in the teams. After that, two teams will be competing.

## The Purpose of the Project

Purpose of this project is to promote the individual growth in all dimensions which includes educational, artistic, and other interpersonal skills.

### The User Business or Background of the Project Effort

There are many online games that have been criticized for the use of violence and brainwashing the minds of teenagers. Because of that many schools in some countries ban that game. Our game should not follow in that category. Whenever any kid or teenager plays the game, no parents or teacher should feel regret about that. The target audience should be very concrete and specific. When all development values are included, audience is wide with less restriction.

Many games are banned with excessive violence which damages the mental health of client to some extend. Our deliver product should enable the player to gain some skills which he/she never had before. Once we reached that, the business will be enabled from all the corners.

Can a student build his aptitude by solving only math problems? Participating in academics is not the only way to increase your talent. Many are struggling in leadership and communication skills. Without solving above problems and visions, the project won’t be started.

Games normally tends to become over additive. Addiction in studying for hours is also harmless. So we are considering certain ways to restrict the user to play the game for only certain hours. Hence, we want the universal acceptance of the game.

### Goals of the Project

Project is carried with vision to sharpen the skills of client in all aspects. The main goal of this game is to combine entertainment values with educational values. Getting artifacts will required the user to solves clues, puzzles and logical reasonings. Further, creating avator will build the creativity side by side. Moreover, during the war, players will require strong teamwork, communication skills and leadership quality. Hence, product is meant for overall personality development of the client.

Special Consideration: This product should also encourage the educational department and social organizations to keep an eye on this.

### Measurement

Several statistics will be made for measuring the success rate for this product. The key measure at the initial stages is to look for how many customers are buying this product in certain time frame. We want to keep track of the graph and see how it is increasing. Next, we want to see that whether the product is accepted universally or not. For example, our age range is from 15-30, but we want to see whether the older people will be able to play or not. Not only this, we also want to check and see whether the game is attracted by other class of people like teachers, elder people, and other critics. We need research team to carry out this process. Other innovative measure we will do is to take short aptitude test for the clients at different stages of the game played. If we somehow able to show that clients are becoming smarter and more intelligent, many doors will be able to market this product in better way.

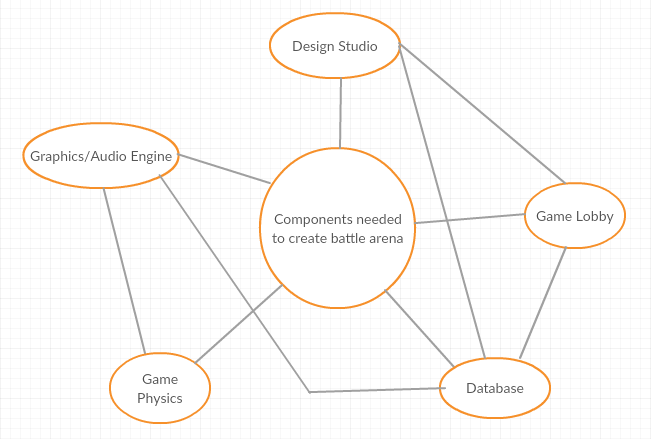
## The Scope of the Work

There are many different maps that users can play on. Each game provides a different experience as there are multiple maps and many different kinds of players can join and play together.

### The Current Situation

This can be published on Steam so; it can be used on Mac and windows. For enthusiasts, this can be costly as it requires playing and winning the games to increase the matchmaking rank to reach on top. These locations can be very large or small. For large maps time limit can be added that will force the avatars to fight with each other. Not only is this, organizing groups of people who can play a virtual game on a real life map near to impossible. In addition, the game will allow users to host an Ikon War server anytime of the day from the comfort of their own homes. Players can save replays of their matches so they can watch it later and learn what they did right or wrong in the match.

### The Context of the Work



### Work Partitioning

|  |  |  |
| --- | --- | --- |
| Event Name | Input and Output | Summary |
| Create Account | User will enter their account info into the database | Allows a player to create and account with his unique ID that will be used in game play |
| Select Map | User will select the map that he wants to play in. | Allows players to download a map from the map database. |
| Find Match | User selects the region that he is in. | Allows players to play in their own region and find players similar to their rank. |
| Start the game | Input: game info from the database.  Output: render information and create game using the game and media engines | When a game is started, information about to the players and the specific scenario being played are sent to the game engines to be rendered and allows the users to play the game. The engines will aid in creating the life-like feel and immersive experience we are aiming for |
| Select Avatar | User will select an Avatar from the database for one game | Allows player to select and apply cosmetics linked to their account. |
| View Profile | Select profile from database | Allows player to check their stats/current progress and watch their old recorded games |
| Watch a game | Server | Allows player to spectator a game that their friends are playing. |

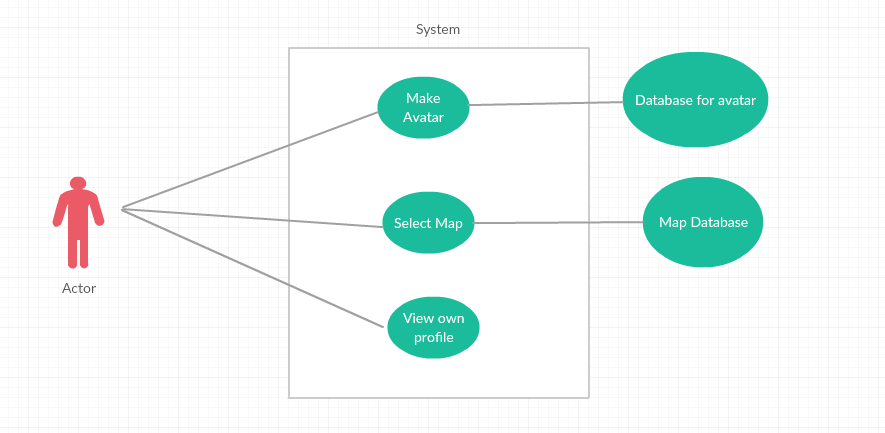
### Competing Products

After doing much research, we found there are other products in MOBA(multiplayer online battle arena) but, we are introducing change in maps and different avatars that are not in the market yet. So it will help players to interact with the real world maps that they live in. Users can create their own virtual world inside their real world maps.

## The Scope of the Product

The main focus of this game is to develop a virtual world inside the real world maps.

### Scenario Diagram(s)



### Product Scenario List

* + - 1. Single Player
      2. Party Game
      3. Bots

### Individual Product Scenarios

Single Player: - In single player game user is matched with other 4 players in the team. Other players will have similar match making rank as the user. Server will find teammates as well as 5 opponents and place them on the map to play.

Party Game: - In party matches user can play with their friends it can be 2, 3, 4 or 5 players. Server will find other players accordingly.

Bots: - In this user will play with the predesigned algorithm which will have 3 levels crusader which will be an easy level. Legend which will be medium level and Immortal will be the hard level.

## Stakeholders

### The Client

The client is the company that are really interest to spread ideas of the map-based game. and motivate how is this interesting while playing. They could start finding target with startup or big company like PS4 or XBox because they have plenty of experience about making games and marketing.

### The Customer

The game customer are the more than average people that love to fight. This game are multiplayer because of that one player connect with other player and make a team while playing game. In addition, Two teams are make challenges and get rank based on who won or lose.

### Hands-On Users of the Product

● User category: Teen age or older.

● User role: Role is play the game

● Subject matter experience: No specific knowledge required to know how to play game, just follow instruction if user is new.

● Technological experience: User need to know how to use computer, but there will also be instruction to guide the user.

● Other user characteristics: The User should enjoy the game with different maps, and different Avatar. Also, some maps are little changing to play. After all, user feels enjoyable and challenging game.

### Maintenance Users and Service Technicians

The people who will work with product maintenance they should know if they find any bug then how to troubleshoot. Also, they know about how to update game.

### Other Stakeholders

Company name: Mixer

Mixer is video game live streaming platform. This company serve large amount of users who enjoy watching other people’s content live. This platform provide good example for streams, who are really interested to playing game. Also, how enjoyable this game and how challenging. In addition, viewers are provide their reviews regarding game.

### User Participation

The user must have computer and some knowledge about how to use computer. There is no additional requirement need accept follow instruction in game.

### Priorities Assigned to Users

● Key users: Simple computer knowledge of use.

● Secondary users: Player will recommend to other players and playing in team is very enjoyable.

● Unimportant users: N/A

## Mandated Constraints

### Solution Constraints

*Description:* The product shall allow to play Ikon game in Single player, Party Games and Bots.

*Rationale:* The client can select avatar and map, this are available with game. There are not other requirements to pay for any extra maps.

*Fit criterion:* Starting menu show display about Single player, Party Game or Bots. User can select anyone. Also, select which map user want to play game.

*Description:* This product work with Window operating system and Mac operating system.

*Rationale:* The client does not have to change their operating system.

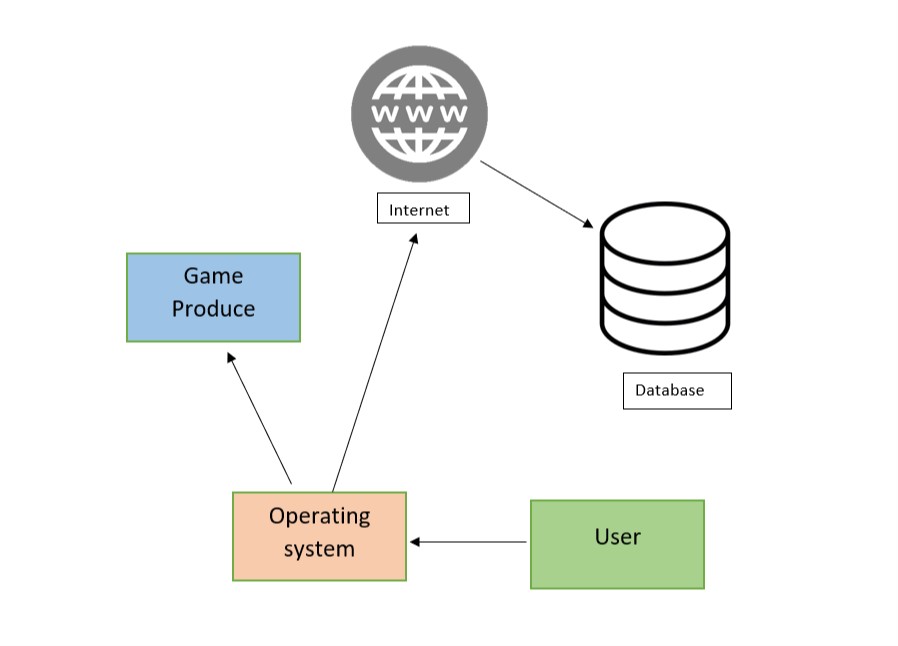
*Fit criterion:* This product connects between Window to Mac or Mac to Window OS. There is not require any particular OS.

*Description:* The product shall allow users to quit in middle of the game, even quit when playing in Party Game, and later resume game. Progress will be saved.

*Rationale:* If Party Game player may not have enough time to play, still they can keep on side their progress and resume later. Between those times, user can play single player game or start Party Game player with different user.

*Fit criterion:* User does not lose any progress with any player.

### implementation Environment of the Current System



### Partner or Collaborative Applications

*Game Engine:* We need connect game engine with Google map to render 3D objects to display map and avatar. We get from another company.

*Game Server:* Product need server. User can save progress and game per player connect with each other and make teams for game, for all this product need server. We can get from another company.

### Off-the-Shelf Software

Content:

Does not need any off-the-shelf software is required for this product.

Motivation:

The product shall handle all requirements such as single player, party game, bots and save progress without depending on any off-the-shelf software.

### Anticipated Workplace Environment

The workplace environment for this product is anywhere. User can access computer/operating system with product install. Internet connection does not require but user want more enjoy with party game than Internet connection is recommended.

### Schedule Constraints

The product requirement will met on its schedule release, as well as any post -release update the product may have.

### Budget Constraints

The budget would depend on project developer team for this product. In Addition rent a database and maintaining after post launch.

## Naming Conventions and Definitions

### Definitions of Key Terms

Player: Human controlled entity, interacts with other players or bots within the map.

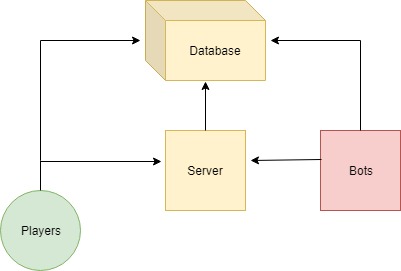
Map: The environment where the match takes place.

Match: A period when entities are allowed to interact with each other. Max players varies by match.

Spectator: Human controlled, allows users to watch matches.

Bot: AI controlled entity, interacts with other players or bots within the map.

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



### 7c Data Dictionary for Any Included Models

Database: Holds information of all entities, bots, players, and the match.

Bot Difficulty: Ranges from Crusader to Immortal. (Easy -> Hard)

## Relevant Facts and Assumptions

### 8a Facts

Players and bots alike will aim to bring their team to victory each match.

### 8b Assumptions

Players will connect to online servers during online ranked matches.

Players will host their own local server if they choose to play solo with bots.

# 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. )*

## 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.*

*This section begins to describe in more specific and precise detail exactly what steps the system takes in the course of its performance. Use cases serve not only to more specifically define the system ( and its boundaries ), but also to identify functional requirements, to identify initial objects / classes, and to organize the work.*

### 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.*

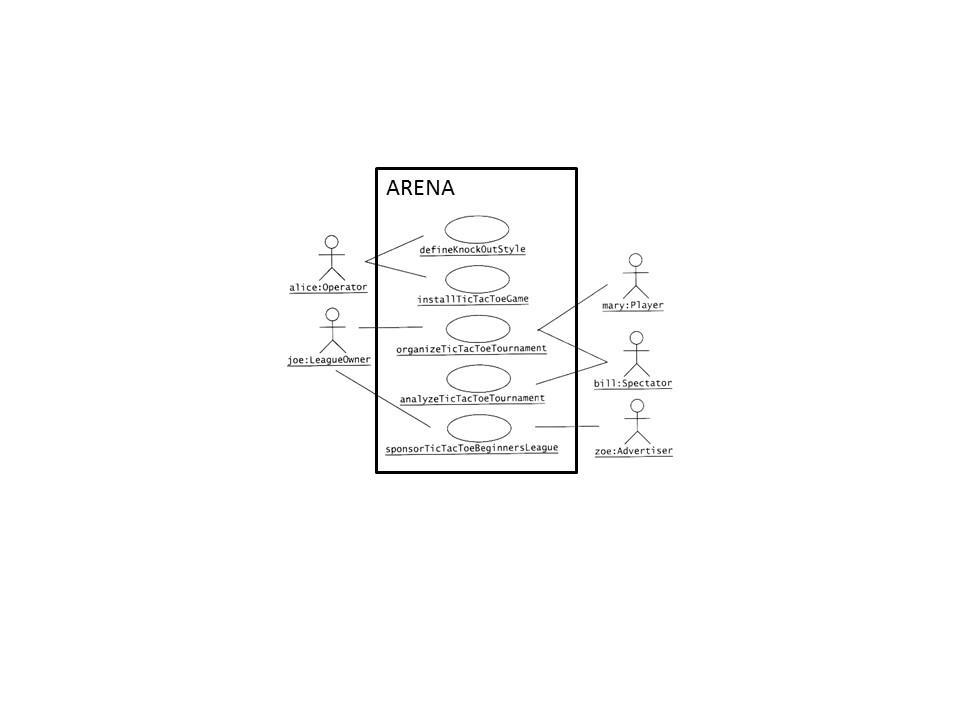
*Use Case diagrams serve two purposes: As a form of graphical table of contents listing the individual use-cases, and also to define the boundary of what is included as part of the proposed system and what is not included.*

*A use case diagram identifies the boundaries between the users (actors) and the product. You arrive at the product boundary by inspecting each business use case and determining, in conjunction with the appropriate stakeholders, which part of the business use case should be automated (or satisfied by some sort of product) and what part should be done by the user. This task must take into account the abilities of the actors (section 3), the constraints (section 4), the goals of the project (section 1), and your knowledge of both the work and the technology that can make the best contribution to the work.*

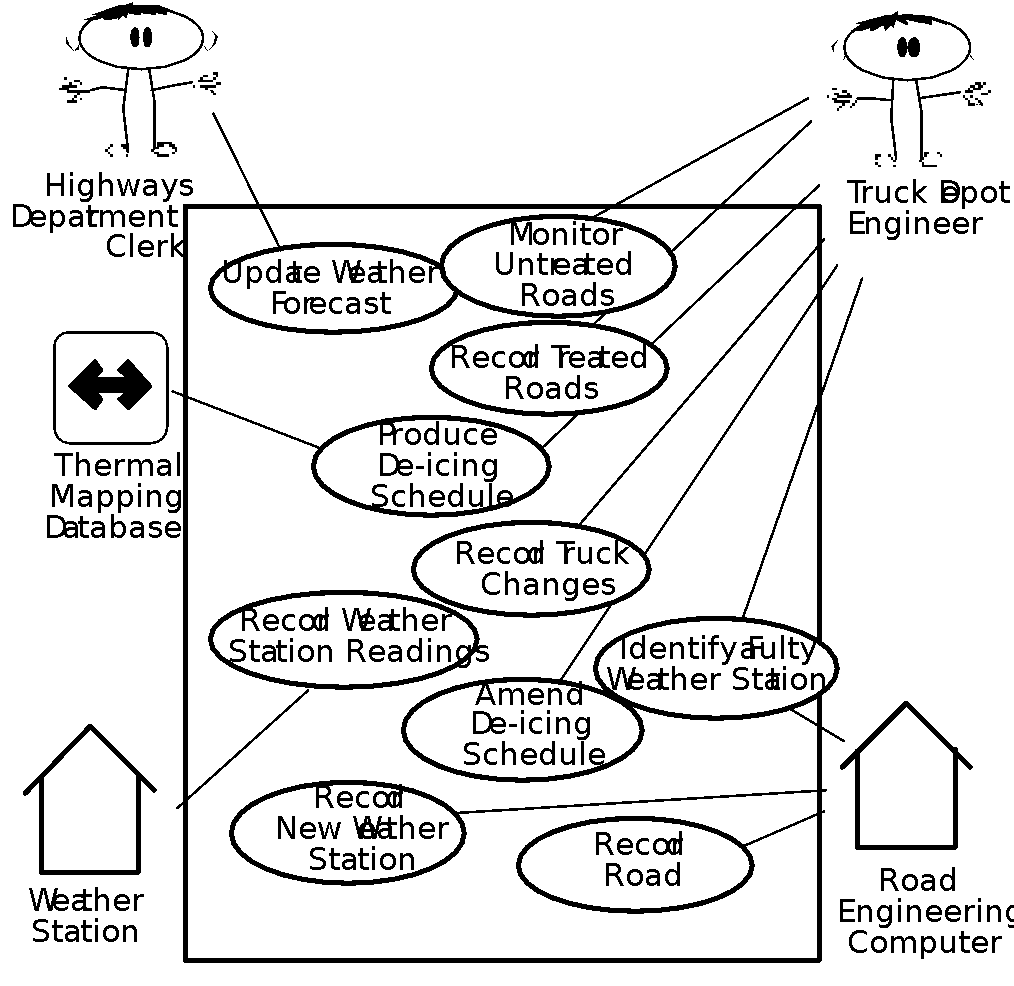
*The use case diagram shows the actors outside the product boundary (the rectangle). The product use cases are the ellipses inside the boundary. The lines denote usage. Note that actors can be either automated or human.*

*Depending on the complexity of the product it may be necessary to use more than one diagram to list all of the use cases. When more than one diagram is required the use-cases can be divided up several ways: Normal operations versus exceptional cases, or daily tasks versus monthly tasks, or user tasks versus administration tasks, etc.*

*Examples*



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

**

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

*Derive the product use cases by deciding where the product boundary should be for each business use case. These decisions are based on your knowledge of the work and the requirements constraints.*

### 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.*

*The use case diagram is a graphical way of summarizing the product use cases relevant to the product. If you have a large number of product use cases (we find 15–20 is a good limit), then it is better to make a list of the product use cases and model or describe each one individually.*

### Individual Product Use Cases

*Use cases are similar to scenarios, in that both tell the story of how the system interacts with the user(s) in response to some business event or while conducting some business task. The difference is that use-cases are much more formal, with certain pre-determined sections for each use-case, and that use-cases indicate clearly what action the system takes in response to what actions taken by the user.*

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

|  |
| --- |
| Use case ID: Name:  pre-conditions:  post-conditions:  Initiated by:  Triggering Event:  Additional Actors: |
| Sequence of Events:   1. Initiating event or action should be step 1, taken by initiating actor. 2. System response follows, indented right. 3. All external action steps are aligned with step 1. ( "stimulus" style ) 4. All system responses are indented right, aligned with step 2. ( "response" style ) 5. All steps should be expressed in the active voice, clearly indicating **who** performs each action 6. The sequence of events should show a back-and-forth stimulus-response relationship. |
| Alternatives: These would be normal and expected variations from the base case.  Exceptions: These would be unusual variations from the base case, often caused by problems. |

* *For all of the above, list as NA if not applicable.*
* *The following may be added if relevant, or omitted otherwise:*
  + *related use cases or scenarios*
  + *associated tests, systems, classes, etc.*
  + *revision history*
  + *references to other documents*
  + *author(s) / originator( s )*
  + *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.*

*You may also want to view Figure 4.7 from "Object Oriented Software Engineering" by Bruegge and DuToit*

## 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.*

*Content*

*A specification for each functional requirement. A full explanation is included in this template’s introductory material.*

*Motivation*

*To specify the detailed functional requirements for the activity of the product.*

*Fit Criterion*

*Each functional requirement should have a fit criterion or a test case. In any event, the fit criterion is the benchmark to allow the tester to determine whether the implemented product has met the requirement.*

*Considerations*

*If you have produced an event/use case list (see sections 7b and 8a), then you can use it to help you trigger the functional requirements for each event/use case. If you have not produced an event/use case list, give each functional requirement a unique number and, to help with traceability, partition these requirements into event/use case–related groups later in the development process.*

**ID# - Name**

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

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

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

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

## 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.*

*Content*

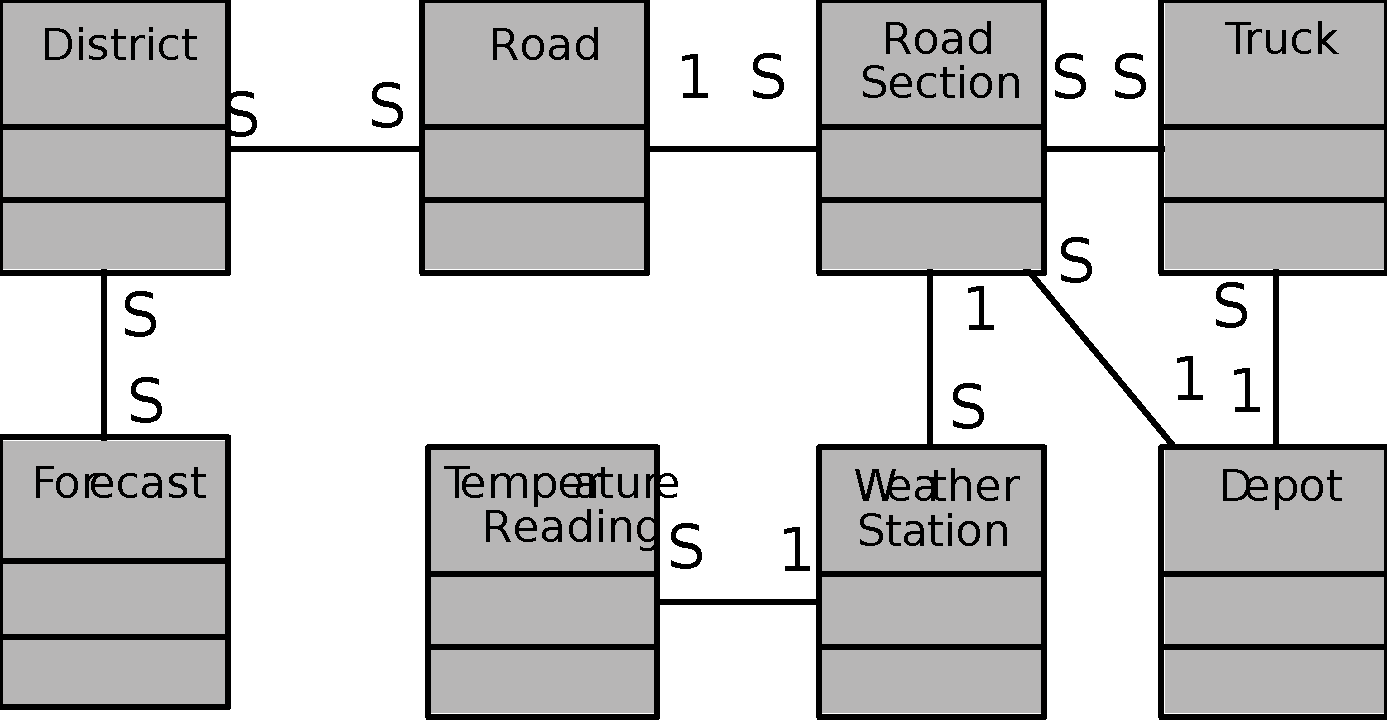
*A specification of the essential subject matter, business objects, entities, and classes that are germane to the product. It might take the form of a first-cut class model, an object model, or a domain model. Alternatively, these requirements might be described by defining the terms in the dictionary described in section 5.*

*Motivation*

*To clarify the system’s subject matter, thereby triggering recognition of requirements not yet considered.*

*Example*

*This is a model of the system’s business subject matter using the Unified Modeling Language (UML) class model notation.*

**

*You can use any type of data or object model to capture this knowledge. The issue is to capture the meaning of the business subject matter and the connections between the individual parts, and to show that you are consistent within your project. If you have an established company standard notation, use that, as it will help you to reuse knowledge between projects.*

*Considerations*

*Are there any data or object models for similar or overlapping systems that might be a useful starting point? Is there a domain model for the subject matter dealt with by this system?*

**ID# - Name**

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

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

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

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

## Performance Requirements

### 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.*

*Content*

*Specifies the amount of time available to complete specified tasks. These requirements often refer to response times. They can also refer to the product’s ability to operate at a speed suitable for the intended environment.*

*Motivation*

*Some products—usually real-time products—must be able to perform some of their functionality within a given time slot. Failure to do so may mean catastrophic failure (e.g., a ground-sensing radar in an airplane fails to detect an upcoming mountain) or the product will not cope with the required volume of use (e.g., an automated ticket-selling machine).*

*Examples*

*Any interface between a user and the automated system shall have a maximum response time of 2 seconds.*

*The response shall be fast enough to avoid interrupting the user’s flow of thought.*

*The product shall poll the sensor every 10 seconds.*

*The product shall download the new status parameters within 5 minutes of a change.*

*Fit Criterion*

*Fit criteria are needed when the description of the requirement is not quantified. However, we find that most performance requirements are stated in quantified terms. The exception is the second requirement shown above, for which the suggested fit criterion is*

*The product shall respond in less than 1 second for 90 percent of the interrogations. No response shall take longer than 2.5 seconds.*

*Considerations*

*There is a wide variation in the importance of different types of speed requirements. If you are working on a missile guidance system, then speed is extremely important. By contrast, an inventory control report that is run once every six months has very little need for a lightning-fast response time.*

*Customize this section of the template to give examples of the speed requirements that are important within your environment.*

**ID# - Name**

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

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

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

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

### Precision or Accuracy Requirements

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

*Content*

*Quantification of the desired accuracy of the results produced by the product.*

*Motivation*

*To set the client’s and users’ expectations for the precision of the product.*

*Examples*

*All monetary amounts shall be accurate to two decimal places.*

*Accuracy of road temperature readings shall be within ±2°C.*

*Considerations*

*If you have done any detailed work on definitions, then some precision requirements might be adequately defined by definitions in section 5.*

*You might consider which units the product is intended to use. Readers will recall the spacecraft that crashed on Mars when coordinates were sent as metric data rather than imperial data.*

*The product might also need to keep accurate time, be synchronized with a time server, or work in UTC.*

*Also, be aware that some currencies have no decimal places, such as the Japanese yen.*

**ID# - Name**

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

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

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

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

### 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.*

*Content*

*This section specifies the volumes that the product must be able to deal with and the amount of data stored by the product.*

*Motivation*

*To ensure that the product is capable of processing the expected volumes.*

*Examples*

*The product shall cater for 300 simultaneous users within the period from 9:00 a.m. to 11:00 a.m. Maximum loading at other periods will be 150 simultaneous users.*

*During a launch period, the product shall cater for a maximum of 20 people to be in the inner chamber.*

*Fit Criterion*

*In this case, the requirement description is quantified, and thus can be tested.*

**ID# - Name**

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

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

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

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

## Dependability Requirements

### 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.*

*Content*

*This section quantifies the necessary reliability of the product. The reliability is usually expressed as the allowable time between failures, or the total allowable failure rate.*

*Motivation*

*It is critical for some products not to fail too often. This section allows you to explore the possibility of failure and to specify realistic levels of service. It also gives you the opportunity to set the client’s and users’ expectations about the expected frequency and significance of potential failures.*

*Examples*

*The product shall not fail more than once per day.*

*No data shall be lost or damaged in the event of a failure. ( This is an example of a* ***fail-safe*** *requirement, which states that the product is allowed to fail, but it must do so safely. )*

*Considerations*

*Consider carefully whether the real requirement for your product is that it is available for use or that it does not fail at any time.*

*Consider also the cost of reliability and availability, and whether it is justified for your 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 . . .

### 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.*

*Content*

*This section quantifies the necessary availability of the product. The availability is usually expressed as the fraction of total time that the system is up and available for use.*

*Availability is a function of the mean time between failures, the mean time required to bring the system back up after a failure, and the mean time the system is expected to be down for routine maintenance.*

*Motivation*

*There is a subtle distinction between how often a system goes down ( reliability ) and how much total time it spends being down ( availability ). This section allows you to specify realistic expectations about the amount of time that the product will be available for use.*

*Examples*

*The product shall be available for use 24 hours per day, 365 days per year.*

*The product shall be available for use between the hours of 8:00 a.m. and 5:30 p.m.*

*The escalator shall run from 6 a.m. until 10 p.m. or the last flight arrives.*

*The product shall achieve 99 percent uptime.*

*Considerations*

*Consider carefully whether the real requirement for your product is that it is available for use or that it does not fail at any time.*

*Consider also the cost of reliability and availability, and whether it is justified for your product.*

*The sections on reliability and availability can sometimes be combined.*

**ID# - Name**

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

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

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

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

### 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.*

*Content*

*Robustness specifies the ability of the product to continue to function under abnormal circumstances.*

*Motivation*

*To ensure that the product is able to provide some or all of its services after or during some abnormal happening in its environment.*

*Examples*

*The product shall continue to operate in local mode whenever it loses its link to the central server.*

*The product shall provide 10 minutes of emergency operation should it become disconnected from the electricity source.*

*Considerations*

*Abnormal happenings can almost be considered normal. Today’s products are so large and complex that there is a good chance that at any given time, one component will not be functioning correctly. Robustness requirements are intended to prevent total failure of the product.*

*You could also consider disaster recovery in this section. This plan describes the ability of the product to reestablish acceptable performance after faults or abnormal happenings.*

**ID# - Name**

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

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

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

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

### Safety-Critical Requirements

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

*Content*

*Quantification of the perceived risk of damage to people, property, and environment. Different countries have different standards, so the fit criteria must specify precisely which standards the product must meet.*

*Motivation*

*To understand and highlight the damage that could potentially occur when using the product within the expected operational environment.*

*Examples*

*The product shall not emit noxious gases that damage people’s health.*

*The heat exchanger shall be shielded from human contact.*

*Fit Criterion*

*The product shall be certified to comply with the Health Department’s standard E110-98. It is to be certified by qualified testing engineers.*

*No member of a test panel of [specified size] shall be able to touch the heat exchanger. The heat exchanger must also comply with safety standard [specify which one].*

*Considerations*

*The example requirements given here apply to some, but not all, products. It is not possible to give examples of every variation of safety-critical requirement. To make the template work in your environment, you should customize it by adding examples that are specific to your products.*

*Also, be aware that different countries have different safety standards and laws relating to safety. If you plan to sell your product internationally, you must be aware of these laws. A colleague has suggested that for electrical products, if you follow the German standards, the largest number of countries will be supported.*

*If you are building safety-critical systems, then the relevant safety-critical standards are already well specified. You will likely have safety experts on your staff. These experts are the best source of the relevant safety-critical requirements for your type of product. They will almost certainly have copious information that you can use.*

*Consult your legal department. Members of this department will be aware of the kinds of lawsuits that have resulted from product safety failure. This is probably the best starting place for generating relevant safety requirements.*

**ID# - Name**

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

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

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

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

## Maintainability and Supportability Requirements

### 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.*

*Content*

*A quantification of the time necessary to make specified changes to the product.*

*Motivation*

*To make everyone aware of the maintenance needs of the product.*

*Examples*

*New MIS reports must be available within one working week of the date when the requirements are agreed upon.*

*A new weather station must be able to be added to the system overnight.*

*Considerations*

*There may be special requirements for maintainability, such as that the product must be able to be maintained by its end users or by developers who are not the original developers. These requirements have an effect on the way that the product is developed. In addition, there may be requirements for documentation or training.*

*You might also consider writing testability requirements 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 . . .

### Supportability Requirements

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

*Content*

*This specifies the level of support that the product requires. Support is often provided via a help desk. If people will provide support for the product, that service is considered part of the product: Are there any requirements for that support? You might also build support into the product itself, in which case this section is the place to write those requirements.*

*Motivation*

*To ensure that the support aspect of the product is adequately specified.*

*Considerations*

*Consider the anticipated level of support, and what forms it might take. For example, a constraint might state that there is to be no printed manual. Alternatively, the product might need to be entirely self-supporting.*

**ID# - Name**

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

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

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

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

### Adaptability Requirements

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

*Content*

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

*Motivation*

*To quantify the client’s and users’ expectations about the platforms on which the product will be able to run.*

*Examples*

*The product is expected to run under Windows XP and Linux.*

*The product might eventually be sold in the Japanese market.*

*The product is designed to run in offices, but we intend to have a version running in restaurant kitchens.*

*Fit Criterion*

*Specification of system software on which the product must operate.*

*Specification of future environments in which the product is expected to operate.*

*Time allowed to make the transition.*

*Considerations*

*Question your marketing department to discover unstated assumptions that have been made about the portability 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 . . .

### Scalability or Extensibility Requirements

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

*Content*

*This specifies the expected increases in size that the product must be able to handle. As a business grows (or is expected to grow), our software products must increase their capacities to cope with the new volumes.*

*Motivation*

*To ensure that the designers allow for future capacities.*

*Examples*

*The product shall be capable of processing the existing 100,000 customers. This number is expected to grow to 500,000 customers within three years.*

*The product shall be able to process 50,000 transactions per hour within two years of its launch.*

**ID# - Name**

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

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

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

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

### Longevity Requirements

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

*Content*

*This specifies the expected lifetime of the product.*

*Motivation*

*To ensure that the product is built based on an understanding of expected return on investment.*

*Examples*

*The product shall be expected to operate within the maximum maintenance budget for a minimum of five years.*

**ID# - Name**

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

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

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

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

## 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.*

### Access Requirements

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

*Content*

*Specification of who has authorized access to the product (both functionality and data), under what circumstances that access is granted, and to which parts of the product access is allowed.*

*Motivation*

*To understand the expectations for confidentiality aspects of the system.*

*Examples*

*Only direct managers can see the personnel records of their staff.*

*Only holders of current security clearance can enter the building.*

*Fit Criterion*

*System function name or system data name.*

*User roles and/or names of people who have clearance.*

*Considerations*

*Is there any data that management considers to be sensitive? Is there any data that low-level users do not want management to have access to? Are there any processes that might cause damage or might be used for personal gain? Are there any people who should not have access to the system?*

*Avoid stating how you will design a solution to the security requirements. For instance, don’t “design a password system.” Your aim here is to identify the security requirement; the design will then come from this description.*

*Consider asking for help. Computer security is a highly specialized field, and one where improperly qualified people have no business. If your product has need of more than average security, we advise you to make use of a security consultant. Such consultants are not cheap, but the results of inadequate security can be even more expensive.*

**ID# - Name**

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

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

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

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

### Integrity Requirements

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

*Content*

*Specification of the required integrity of databases and other files, and of the product itself.*

*Motivation*

*To understand the expectations for the integrity of the product’s data. To specify what the product will do to ensure its integrity in the case of an unwanted happening such as attack from the outside or unintentional misuse by an authorized user.*

*Examples*

*The product shall prevent incorrect data from being introduced.*

*The product shall protect itself from intentional abuse.*

*Considerations*

*Organizations are relying more and more on their stored data. If this data should be come corrupt or incorrect—or disappear—then it could be a fatal blow to the organization. For example, almost half of small businesses go bankrupt after a fire destroys their computer systems. Integrity requirements are aimed at preventing complete loss, as well as corruption, of data and processes.*

**ID# - Name**

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

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

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

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

### 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.)*

*Content*

*Specification of what the product has to do to ensure the privacy of individuals about whom it stores information. The product must also ensure that all laws related to privacy of an individual’s data are observed.*

*Motivation*

*To ensure that the product complies with the law, and to protect the individual privacy of your customers. Few people today look kindly on organizations that do not observe their privacy.*

*Examples*

*The product shall make its users aware of its information practices before collecting data from them.*

*The product shall notify customers of changes to its information policy.*

*The product shall reveal private information only in compliance with the organization’s information policy.*

*The product shall protect private information in accordance with the relevant privacy laws and the organization’s information policy.*

*Considerations*

*Privacy issues may well have legal implications, and you are advised to consult with your organization’s legal department about the requirements to be written in this section.*

*Consider what notices you must issue to your customers before collecting their personal information. A notice might go so far as to warn customers that you intend to put a cookie in their computer. Also, do you have to do anything to keep customers aware that you hold their personal information?*

*Customers must always be in a position to give or withhold consent when their private data is collected or stored. Similarly, customers should be able to view any private data and, where appropriate, ask for correction of the data.*

*Also consider the integrity and security of private data—for example, when you are storing credit card information.*

**ID# - Name**

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

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

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

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

### Audit Requirements

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

*Content*

*Specification of what the product has to do (usually retain records) to permit the required audit checks.*

*Motivation*

*To build a system that complies with the appropriate audit rules.*

*Considerations*

*This section may have legal implications. You are advised to seek the approval of your organization’s auditors regarding what you write here.*

*You should also consider whether the product should retain information on who has used it. The intention is to provide security such that a user may not later deny having used the product or participated in some form of transaction using 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 . . .

### Immunity Requirements

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

*Content*

*The requirements for what the product has to do to protect itself from infection by unauthorized or undesirable software programs, such as viruses, worms, and Trojan horses, among others.*

*Motivation*

*To build a product that is as secure as possible from malicious interference.*

*Considerations*

*Each day brings more malevolence from the unknown, outside world. People buying software, or any other kind of product, expect that it can protect itself from outside interference.*

**ID# - Name**

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

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

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

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

## Usability and Humanity Requirements

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

### 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.*

*Content*

*This section describes your client’s aspirations for how easy it is for the intended users of the product to operate it. The product’s usability is derived from the abilities of the expected users of the product and the complexity of its functionality.*

*The usability requirements should cover properties such as these:*

*● Efficiency of use: How quickly or accurately the user can use the product.*

*● Ease of remembering: How much the casual user is expected to remember about using the product.*

*● Error rates: For some products it is crucial that the user commits very few, or no, errors.*

*● Overall satisfaction in using the product: This is especially important for commercial, interactive products that face a lot of competition. Web sites are a good example.*

*● Feedback: How much feedback the user needs to feel confident that the product is actually accurately doing what the user expects. The necessary degree of feedback will be higher for some products (e.g., safety-critical products) than for others.*

*Motivation*

*To guide the product’s designers toward building a product that meets the expectations of its eventual users.*

*Examples*

*The product shall be easy for 11-year-old children to use.*

*The product shall help the user to avoid making mistakes.*

*The product shall make the users want to use it.*

*The product shall be used by people with no training, and possibly no understanding of English.*

*Fit Criterion*

*These examples may seem simplistic, but they do express the intention of the client. To completely specify what is meant by the requirement, you must add a measurement against which it can be tested—that is, a fit criterion. Here are the fit criteria for the preceding examples:*

*Eighty percent of a test panel of 11-year-old children shall be able to successfully complete [list of tasks] within [specified time].*

*One month’s use of the product shall result in a total error rate of less than 1 percent.*

*An anonymous survey shall show that 75 percent of the intended users are regularly using the product after a three-week familiarization period.*

*Considerations*

*Refer to section 3, Users of the Product, to ensure that you have considered the usability requirements from the perspective of all the different types of users.*

*It may be necessary to have special consulting sessions with your users and your client to determine whether any special usability considerations must be built into the product.*

*You could also consider consulting a usability laboratory experienced in testing the usability of products that have a project situation (sections 1–7 of this template) similar to yours.*

**ID# - Name**

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

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

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

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

### 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.*

*Content*

*This section describes the way in which the product can be altered or configured to take into account the user’s personal preferences or choice of language.*

*The personalization requirements should cover issues such as the following:*

*● Languages, spelling preferences, and language idioms*

*● Currencies, including the symbols and decimal conventions*

*● Personal configuration options*

*Motivation*

*To ensure that the product’s users do not have to struggle with, or meekly accept, the builder’s cultural conventions.*

*Examples*

*The product shall retain the buyer’s buying preferences.*

*The product shall allow the user to select a chosen language.*

*Considerations*

*Consider the country and culture of the potential customers and users of your product. Any out-of-country users will welcome the opportunity to convert to their home spelling and expressions.*

*By allowing users to customize the way in which they use the product, you give them the opportunity to participate more closely with your organization as well as enjoy their own personal user experience.*

*You might also consider the configurability of the product. Configurability allows different users to have different functional variations 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 . . .

### Learning Requirements

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

*Content*

*Requirements specifying how easy it should be to learn to use the product. This learning curve ranges from zero time for products intended for placement in the public domain (e.g., a parking meter or a web site) to a considerable amount of time for complex, highly technical products. (We know of one product where it was necessary for graduate engineers to spend 18 months in a training program before being qualified to use the product.)*

*Motivation*

*To quantify the amount of time that your client feels is allowable before a user can successfully use the product. This requirement guides designers to understand how users will learn the product. For example, designers may build elaborate interactive help facilities into the product, or the product may be packaged with a tutorial. Alternatively, the product may have to be constructed so that all of its functionality is apparent upon first encountering it.*

*Examples*

*The product shall be easy for an engineer to learn.*

*A clerk shall be able to be productive within a short time.*

*The product shall be able to be used by members of the public who will receive no training before using it.*

*The product shall be used by engineers who will attend five weeks of training before using the product.*

*Fit Criterion*

*An engineer shall produce a [specified result] within [specified time] of beginning to use the product, without needing to use the manual.*

*After receiving [number of hours] training a clerk shall be able to produce [quantity of specified outputs] per [unit of time].*

*[Agreed percentage] of a test panel shall successfully complete [specified task] within [specified time limit].*

*The engineers shall achieve [agreed percentage] pass rate from the final examination of the training.*

*Considerations*

*Refer to section 3, Users of the Product, to ensure that you have considered the ease of learning requirements from the perspective of all the different types of users.*

**ID# - Name**

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

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

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

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

### 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.*

*This section is concerned with discovering requirements related to concepts and metaphors that are familiar to the intended end users.*

*Content*

*This specifies the requirement for the product to be understood by its users. While “usability” refers to ease of use, efficiency, and similar characteristics, “understandability” determines whether the users instinctively know what the product will do for them and how it fits into their view of the world. You can think of understandability as the product being polite to its users and not expecting them to know or learn things that have nothing to do with their business problem.*

*Motivation*

*To avoid forcing users to learn terms and concepts that are part of the product’s internal construction and are not relevant to the users’ world. To make the product more comprehensible and thus more likely to be adopted by its intended users.*

*Examples*

*The product shall use symbols and words that are naturally understandable by the user community.*

*The product shall hide the details of its construction from the user.*

*Considerations*

*Refer to section 3, Users of the Product, and consider the world from the point of view of each of the different types of users.*

**ID# - Name**

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

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

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

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

### Accessibility Requirements

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

*Content*

*The requirements for how easy it should be for people with common disabilities to access the product. These disabilities might be related to physical disability or visual, hearing, cognitive, or other abilities.*

*Motivation*

*In many countries it is required that some products be made available to the disabled. In any event, it is self-defeating to exclude this sizable community of potential customers.*

*Examples*

*The product shall be usable by partially sighted users.*

*The product shall conform to the Americans with Disabilities Act.*

*Considerations*

*Some users have disabilities other than the commonly described ones. In addition, some partial disabilities are fairly common. A simple, and not very consequential, example is that approximately 20 percent of males are red-green colorblind.*

**ID# - Name**

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

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

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

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

### User Documentation Requirements

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

*Content*

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

*Motivation*

*To set expectations for the documentation and to identify who will be responsible for creating it.*

*Examples*

*Technical specifications to accompany the product.*

*User manuals.*

*Service manuals (if not covered by the technical specification).*

*Emergency procedure manuals (e.g., the card found in airplanes).*

*Installation manuals.*

*Considerations*

*Which documents do you need to deliver, and to whom? Bear in mind that the answer to this questions depends on your organizational procedures and roles.*

*For each document, consider these issues:*

*● The purpose of the document*

*● The people who will use the document*

*● Maintenance of the document*

*What level of documentation is expected? Will the users be involved in the production of the documentation? Who will be responsible for keeping the documentation up-to-date? What form will the documentation 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 . . .

### Training Requirements

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

*Content*

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

*Motivation*

*To set expectations for the training. To identify who is responsible for creating and providing that training.*

*Considerations*

*What training will be necessary? Who will design the training? Who will provide the training?*

**ID# - Name**

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

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

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

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

## Look and Feel Requirements

### 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.*

*Content*

*The section contains requirements relating to the spirit of the product. Your client may have made particular demands for the product, such as corporate branding, colors to be used, and so on. This section captures the requirements for the appearance. Do not attempt to design it until the appearance requirements are known.*

*Motivation*

*To ensure that the appearance of the product conforms to the organization’s expectations.*

*Examples*

*The product shall be attractive to a teenage audience.*

*The product shall comply with corporate branding standards.*

*Fit Criterion*

*A sampling of representative teenagers shall, without prompting or enticement, start using the product within four minutes of their first encounter with it.*

*The office of branding shall certify the product complies with the current standards.*

*Considerations*

*Even if you are using prototypes, it is important to understand the requirements for the appearance. The prototype is used to help elicit requirements; it should not be thought of as a substitute for the requirements.*

**ID# - Name**

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

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

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

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

### 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.*

*Content*

*Requirements that specify the mood, style, or feeling of the product, which influences the way a potential customer will see the product. Also, the stakeholders’ intentions for the amount of interaction the user is to have with the product.*

*In this section, you would also describe the appearance of the package if this is to be a manufactured product. The package may have some requirements as to its size, style, and consistency with other packages put out by your organization. Keep in mind the European laws on packaging, which require that the package not be significantly larger than the product it encloses.*

*The style requirements that you record here will guide the designers to create a product as envisioned by your client.*

*Motivation*

*Given the state of today’s market and people’s expectations, we cannot afford to build products that have the wrong style. Once the functional requirements are satisfied, it is often the appearance and style of products that determine whether they are successful. Your task in this section is to determine precisely how the product shall appear to its intended consumer.*

*Example*

*The product shall appear authoritative.*

*Fit Criterion*

*After their first encounter with the product, 70 percent of representative potential customers shall agree they feel they can trust the product.*

*Considerations*

*The look and feel requirements specify your client’s vision of the product’s appearance. The requirements may at first seem to be rather vague (e.g., “conservative and professional appearance”), but these will be quantified by their fit criteria. The fit criteria give you the opportunity to extract from your client precisely what is meant, and give the designer precise instructions on what he is to accomplish.*

**ID# - Name**

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

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

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

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

## Operational and Environmental Requirements

### Expected Physical Environment

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

*Content*

*This section specifies the physical environment in which the product will operate.*

*Motivation*

*To highlight conditions that might need special requirements, preparations, or training. These requirements ensure that the product is fit to be used in its intended environment.*

*Examples*

*The product shall be used by a worker, standing up, outside in cold, rainy conditions.*

*The product shall be used in noisy conditions with a lot of dust.*

*The product shall be able to fit in a pocket or purse.*

*The product shall be usable in dim light.*

*The product shall not be louder than the existing noise level in the environment.*

*Considerations*

*The work environment: Is the product to operate in some unusual environment? Does this lead to special requirements? Also see section 11, Usability and Humanity Requirements.*

**ID# - Name**

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

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

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

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

### 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.*

*Content*

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

*Motivation*

*Requirements for the interfaces to other applications often remain undiscovered until implementation time. Avoid a high degree of rework by discovering these requirements early.*

*Examples*

*The products shall work on the last four releases of the five most popular browsers.*

*The new version of the spreadsheet must be able to access data from the previous two versions.*

*Our product must interface with the applications that run on the remote weather stations.*

*Fit Criterion*

*For each inter-application interface, specify the following elements:*

*● The data content*

*● The physical material content*

*● The medium that carries the interface*

*● The frequency*

*● The volume*

**ID# - Name**

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

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

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

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

### Productization Requirements

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

*Content*

*Any requirements that are necessary to make the product into a distributable or salable item. It is also appropriate to describe here the operations needed to install a software product successfully.*

*Motivation*

*To ensure that if work must be done to get the product out the door, then that work becomes part of the requirements. Also, to quantify the client’s and users’ expectations about the amount of time, money, and resources they will need to allocate to install the product.*

*Examples*

*The product shall be distributed as a ZIP file.*

*The product shall be able to be installed by an untrained user without recourse to separately printed instructions.*

*The product shall be of a size such that it can fit on one CD.*

*Considerations*

*Some products have special needs to turn them into a salable or usable product. You might consider that the product has to be protected such that only paid-up customers can access it.*

*Ask questions of your marketing department to discover unstated assumptions that have been made about the specified environment and the customers’ expectations of how long installation will take and how much it will cost.*

*Most commercial products have some needs in this area.*

**ID# - Name**

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

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

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

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

### Release Requirements

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

*Content*

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

*Motivation*

*To make everyone aware of how often you intend to produce new releases of the product.*

*Examples*

*The maintenance releases will be offered to end users once a year.*

*Each release shall not cause previous features to fail.*

*Fit Criterion*

*Description of the type of maintenance plus the amount of effort budgeted for it.*

*Considerations*

*Do you have any existing contractual commitments or maintenance agreements that might be affected by the new 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 . . .

## Cultural and Political Requirements

### 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.*

*Content*

*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.*

*Motivation*

*To bring out in the open requirements that are difficult to discover because they are outside the cultural experience of the developers.*

*Examples*

*The product shall not be offensive to religious or ethnic groups.*

*The product shall be able to distinguish between French, Italian, and British road-numbering systems.*

*The product shall keep a record of public holidays for all countries in the European Union and for all states in the United States.*

*Considerations*

*Question whether the product is intended for a culture other than the one with which you are familiar. Ask whether people in other countries or in other types of organizations will use the product. Do these people have different habits, holidays, superstitions, or cultural norms that do not apply to your own culture? Are there colors, icons, or words that have different meanings in another cultural environment?*

**ID# - Name**

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

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

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

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

### 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.*

*Content*

*This section contains requirements that are specific to the political factors that affect the acceptability of the product.*

*Motivation*

*To understand requirements that sometimes appear irrational.*

*Examples*

*The product shall be installed using only American-made components.*

*The product shall make all functionality available to the CEO.*

*Considerations*

*Did you intend to develop the product on a Macintosh, when the office manager has laid down an edict that only Windows machines are permitted?*

*Is a director also on the board of a company that manufactures products similar to the one that you intend to build?*

*Whether you agree with these political requirements has little bearing on the outcome. The reality is that the system has to comply with political requirements even if you can find a better, more efficient, or more economical solution. A few probing questions here may save some heartache later.*

*The political requirements might be purely concerned with the politics inside your organization. However, in other situations you may need to consider the politics inside your customers’ organizations or the national politics of the country.*

**ID# - Name**

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

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

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

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

## Legal Requirements

### 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.*

*Content*

*A statement specifying the legal requirements for this system.*

*Motivation*

*To comply with the law so as to avoid later delays, lawsuits, and legal fees.*

*Examples*

*Personal information shall be implemented so as to comply with the Data Protection Act.*

*Fit Criterion*

*Lawyers’ opinion that the product does not break any laws.*

*Considerations*

*Consider consulting lawyers to help identify the legal requirements.*

*Are there any copyrights or other intellectual property that must be protected? Conversely, do any competitors have copyrights on which you might be in danger of infringing?*

*Is it a requirement that developers have not seen competitors’ code or even have worked for competitors?*

*The Sarbanes-Oxley (SOX) Act, the Health Insurance Portability and Accountability Act (HIPAA) and the Gramm-Leach-Bliley Act may have implications for you. Check with your company lawyer.*

*Might any pending legislation affect the development of this system?*

*Are there any aspects of criminal law you should consider?*

*Have you considered the tax laws that affect your product?*

*Are there any labor laws (e.g., working hours) relevant to your 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 . . .

### Standards Requirements

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

*Content*

*A statement specifying applicable standards and referencing detailed standards descriptions. This does not refer to the law of the land—think of it as an internal law imposed by your company.*

*Motivation*

*To comply with standards so as to avoid later delays.*

*Example*

*The product shall comply with MilSpec standards.*

*The product shall comply with insurance industry standards.*

*The product shall be developed according to SSADM standard development steps.*

*Fit Criterion*

*The appropriate standard-keeper certifies that the standard has been adhered to.*

*Considerations*

*It is not always apparent that there are applicable standards because their existence is often taken for granted. Consider the following:*

*● Do any industry bodies have applicable standards?*

*● Does the industry have a code of practice, watchdog, or ombudsman?*

*● Are there any special development steps for this type of product?*

**ID# - Name**

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

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

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

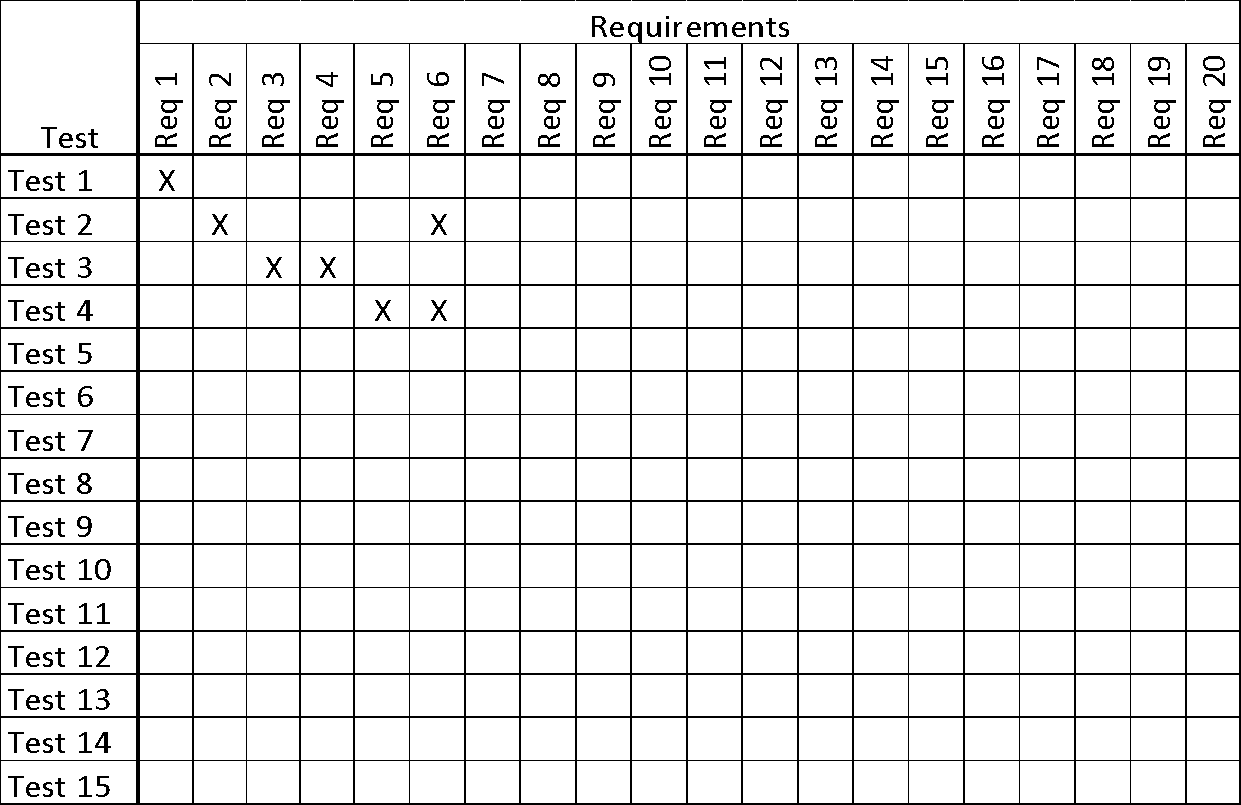
**Acceptance Tests:** List ID# and/or names here . . .

## 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.*

### 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.*

**

***Table 1 - Requirements - Acceptance Tests Correspondence***

### 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 . . .

# Design

## Design Goals

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

*Content*

*Design goals are important properties of the system to be optimized, and which may affect the overall design of the system. For example computer games place a higher priority on speed than accuracy, and so the physics engine for a computer game may make some rough approximations and assumptions that allow it to run as fast as possible while sacrificing accuracy, whereas the physics calculations performed by NASA must be much more rigorously correct, even at the expense of speed.*

*Note an important difference between design goals and requirements: Requirements include specific values that must be met in order for the product to be acceptable to the client, whereas design goals are properties that the designers strive to make "as good as possible", without specific criteria for acceptability. ( Note also that the same property may appear in both a requirement and a design goal, so a design goal may be to make the system run as fast as possible, with a requirement that says any speed below a certain specified threshold is unacceptable. )*

Your text goes here . . .

## 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 . . .

## 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.*

### Initial System Analysis and Class Identification

*SV: Perform grammatical and similar analyses to identify the most import 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 . . .

### 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.*

*Content*

*Include sequence diagrams of each important use-case here. This is a first step towards identifying preliminary objects. ( If the sequence diagram would be too big to fit, then it can either be broken down into pieces or a communication diagram can be used in its place. )*

Your text goes here . . .

### 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 . . .

### 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 . . .

## 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.*

### 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 . . .

### 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 I7c above to avoid undue repetition, while reviewing only the most relevant items here.*

Your text goes here . . .

### 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 . . .

### 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 . . .

### 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 . . .

### 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.*

*The final user interface design will normally be developed by appropriate experts in that area. However it is appropriate to include an initial design here, including possibly a low- or high- fidelity sketch/mockup, in order to identify key classes necessary to implement the user interface, such as forms and dialog windows. It may also go towards addressing usability and/or look-and-feel requirements, and/or identifying other overlooked components.*

Your text goes here . . .

### 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 . . .

## 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 . . .

## 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.*

### 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 . . .

### Subsystem I

Your text goes here . . .

### Subsystem II

Your text goes here . . .

### etc.

Your text goes here . . .

# Project Issues

## Open Issues

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

*Content*

*A statement of factors that are uncertain and might make significant difference to the product.*

*Motivation*

*To bring uncertainty out in the open and provide objective input to risk analysis.*

*Examples*

*Our investigation into whether the new version of the processor will be suitable for our application is not yet complete.*

*The government is planning to change the rules about who is responsible for gritting the motorways, but we do not know what those changes might be.*

*Considerations*

*Are there any issues that have come up from the requirements gathering that have not yet been resolved? Have you heard of any changes that might occur in the other organizations or systems on your context diagram? Are there any legislative changes that might affect your system? Are there any rumors about your hardware or software suppliers that might have an impact?*

Your text goes here . . .

## 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 . . .

### 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.*

*Content*

*List of existing products that should be investigated as potential solutions. Reference any surveys that have been done on these products.*

*Motivation*

*To give consideration to whether a solution can be bought.*

*Considerations*

*Could you buy something that already exists or is about to become available? It may not be possible at this stage to make this determination with a lot of confidence, but any likely products should be listed here.*

*Also consider whether some products must not be used.*

Your text goes here . . .

### Reusable Components

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

*Content*

*Description of the candidate components, either bought from outside or built by your company, that could be used by this project. List libraries that could be a source of components.*

*Motivation*

*Reuse rather than reinvention.*

Your text goes here . . .

### 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.*

*Content*

*List of other similar products or parts of products that you can legally copy or easily modify.*

*Motivation*

*Reuse rather than reinvention.*

*Examples*

*Another electricity company has built a customer service system. Its hardware is different from ours, but we could buy its specification and cut our analysis effort by approximately 60 percent.*

*Considerations*

*While a ready-made solution may not exist, perhaps something, in its essence, is similar enough that you could copy, and possibly modify, it to better effect than starting from scratch. This approach is potentially dangerous because it relies on the base system being of good quality.*

*This question should always be answered. The act of answering it will force you to look at other existing solutions to similar problems.*

Your text goes here . . .

## 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.*

### 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?*

*Content*

*A description of how the new product will affect the current implementation environment. This section should also cover things that the new product should not do.*

*Motivation*

*The intention is to discover early any potential conflicts that might otherwise not be realized until implementation time.*

*Examples*

*Any change to the scheduling system will affect the work of the engineers in the divisions and the truck drivers.*

*Considerations*

*Is it possible that the new system might damage some existing system? Can people be displaced or otherwise affected by the new system?*

*These issues require a study of the current environment. A model highlighting the effects of the change is a good way to make this information widely understandable.*

Your text goes here . . .

### Effects on the Installed Systems

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

*Content*

*Specification of the interfaces between new and existing systems.*

*Motivation*

*Very rarely is a new development intended to stand completely alone. Usually the new system must coexist with some older system. This question forces you to look carefully at the existing system, examining it for potential conflicts with the new development.*

Your text goes here . . .

### 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?*

*Content*

*Details of any adverse reaction that might be suffered by existing users.*

*Motivation*

*Sometimes existing users are using a product in such a way that they will suffer ill effects from the new system or feature. Identify any likely adverse user reactions, and determine whether we care about those reactions and what precautions we will take.*

Your text goes here . . .

### 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. )*

*Content*

*Statement of any potential problems with the new automated technology or new ways of structuring the organization.*

*Motivation*

*The intention is to make early discovery of any potential conflicts that might otherwise not be realized until implementation time.*

*Examples*

*The planned new server is not powerful enough to cope with our projected growth pattern.*

*The size and weight of the new product do not fit into the physical environment.*

*The power capabilities will not satisfy the new product’s projected consumption.*

*Considerations*

*This requires a study of the intended implementation environment.*

Your text goes here . . .

### Follow-Up Problems

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

*Content*

*Identification of situations that we might not be able to cope with.*

*Motivation*

*To guard against situations where the product might fail.*

*Considerations*

*Will we create a demand for our product that we are not able to service? Will the new system cause us to run afoul of laws that do not currently apply? Will the existing hardware cope?*

*There are potentially hundreds of unwanted effects. It pays to answer this question very carefully.*

Your text goes here . . .

## 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.*

### 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.*

*Content*

*A list of the conversion activities. Timetable for implementation.*

*Motivation*

*To identify conversion tasks as input to the project planning process.*

*Considerations*

*Will you use a phased implementation to install the new system? If so, describe which requirements will be implemented by each of the major phases.*

*What kind of data conversion is necessary? Must special programs be written to transport data from an existing system to the new one? If so, describe the requirements for these programs here.*

*What kind of manual backup is needed while the new system is installed?*

*When are each of the major components to be put in place? When are the phases of the implementation to be released?*

*Is there a need to run the new product in parallel with the existing product?*

*Will we need additional or different staff?*

*Is any special effort needed to decommission the old product?*

*This section is the timetable for implementation of the new system.*

Your text goes here . . .

### 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.*

*Content*

*List of data translation tasks.*

*Motivation*

*To discover missing tasks that will affect the size and boundaries of the project.*

*Fit Criterion*

*Description of the current technology that holds the data.*

*Description of the new technology that will hold the data.*

*Description of the data translation tasks.*

*Foreseeable problems.*

*Considerations*

*Every time you make an addition to your dictionary (see section 5), ask this question: Where is this data currently held, and will the new system affect that implementation?*

Your text goes here . . .

## Risks

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

*All projects involve risk—namely, the risk that something will go wrong. Risk is not necessarily a bad thing, as no progress is made without taking some risk. However, there is a difference between unmanaged risk—say, shooting dice at a craps table—and managed risk, where the probabilities are well understood and contingency plans are made. Risk is only a bad thing if the risks are ignored and they become problems. Risk management entails assessing which risks are most likely to apply to the project, deciding a course of action if they become problems, and monitoring projects to give early warnings of risks becoming problems.*

*This section of your specification should contain a list of the most likely risks and the most serious risks for your project. For each risk, include the probability of that risk becoming a problem. Capers Jones’s Assessment and Control of Software Risks (Prentice-Hall, Englewood Cliffs, N.J., 1994) gives comprehensive lists of risks and their probabilities; you can use these lists as a starting point. For example, Jones cites the following risks as being the most serious:*

*• Inaccurate metrics*

*• Inadequate measurement*

*• Excessive schedule pressure*

*• Management malpractice*

*• Inaccurate cost estimating*

*• Silver bullet syndrome*

*• Creeping user requirements*

*• Low quality*

*• Low productivity*

*• Cancelled projects*

*Use your knowledge of the requirements as input to discover which risks are most relevant to your project.*

*It is also useful input to project management if you include the impact on the schedule, or the cost, if the risk does become a problem.*

Your text goes here . . .

## 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.*

*For details on how to estimate requirements effort and costs, refer to Appendix C Function Point Counting: A Simplified Introduction*

*The other cost of requirements is the amount of money or effort that you have to spend building them into a product. Once the requirements specification is complete, you can use one of the estimating methods to assess the cost, expressing the result as a monetary amount or time to build.*

*There is no best method to use when estimating. Keep in mind, however, that your estimates should be based on some tangible, countable artifact. If you are using this template, then, as a result of doing the work of requirements specification, you are producing many measurable deliverables. For example:*

*● Number of input and output flows on the work context*

*● Number of business events*

*● Number of product use cases*

*● Number of functional requirements*

*● Number of nonfunctional requirements*

*● Number of requirements constraints*

*● Number of function points*

*The more detailed the work you do on your requirements, the more accurate your deliverables will be. Your cost estimate is the amount of resources you estimate each type of deliverable will take to produce within your environment. You can create some very early cost estimates based on the work context. At that stage, your knowledge of the work will be general, and you should reflect this vagueness by making the cost estimate a range rather than a single figure.*

*As you increase your knowledge of the requirements, we suggest you try using function point counting—not because it is an inherently superior method, but because it is so widely accepted. So much is known about function point counting that it is possible to make easy comparisons with other products and other installations’ productivity.*

*It is important that your client be told at this stage what the product is likely to cost. You usually express this amount as the total cost to complete the product, but you may also find it advantageous to point out the cost of the requirements effort, or the costs of individual requirements.*

*Whatever you do, do not leave the costs in the lap of hysterical optimism. Make sure that this section includes meaningful numbers based on tangible deliverables.*

Your text goes here . . .

## 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.*

*Requirements that will not be part of the next release. These requirements might be included in future releases of the product.*

*Content*

*Any type of requirement.*

*Motivation*

*To allow requirements to be gathered, even though they cannot be part of the current development. To ensure that good ideas are not lost.*

*Considerations*

*The requirements-gathering process often throws up requirements that are beyond the sophistication of, or time allowed for, the current release of the product. This section holds these requirements in waiting. The intention is to avoid stifling the creativity of your users and clients, by using a repository to retain future requirements. You are also managing expectations by making it clear that you take these requirements seriously, although they will not be part of the agreed-upon product.*

*Many people use the waiting room as a way of planning future versions of the product. Each requirement in the waiting room is tagged with its intended version number. As a requirement progresses closer to implementation, then you can spend more time on it and add details such as the cost and benefit attached to that requirement.*

*You might also prioritize the contents of your waiting room. “Low-hanging fruit”—requirements that provide a high benefit at a low cost of implementation—are the highest-ranking candidates for the next release. You would also give a high waiting room rank to requirements for which there is a pent-up demand.*

Your text goes here . . .

## 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.*

*When you gather requirements, you focus on finding out what the real requirements are and try to avoid coming up with solutions. However, when creative people start to think about a problem, they always generate ideas about potential solutions. This section of the template is a place to put those ideas so that you do not forget them and so that you can separate them from the real business requirements.*

*Content*

*Any idea for a solution that you think is worth keeping for future consideration. This can take the form of rough notes, sketches, pointers to other documents, pointers to people, pointers to existing products, and so on. The aim is to capture, with the least amount of effort, an idea that you can return to later.*

*Motivation*

*To make sure that good ideas are not lost. To help you separate requirements from solutions.*

*Considerations*

*While you are gathering requirements, you will inevitably have solution ideas; this section offers a way to capture them. Bear in mind that this section will not necessarily be included in every document that you publish.*

Your text goes here . . .

## 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.*

*Content*

*At the end of every project you should reflect upon what methods were used that worked out well and should be repeated in the future, and also what methods did not work out well and should be avoided. Any recommendations, suggestions, or ideas for how to do things better in the future should also be documented*

*Motivation*

*To learn from experience, and to continually strive for process improvement.*

*Considerations*

*When things don't go well, it is important to distinguish whether the methods themselves were poor, or simply poorly implemented in this particular case, or whether they just weren't right for this particular project / group of engineers.*

Your text goes here . . .

# 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.*

*The glossary defines terms that may not be familiar to all readers. This is especially important if the document is expected to reach a wide and varied audience, such as school children. The glossary may be placed at either the beginning or the end of the document.*

***Flotsam:*** *Any part of a ship or its cargo found floating on the water, whether it was deliberately or accidentally lost by its original owners.*

***Jetsam:*** *Any part of a ship or its cargo that is deliberately cast off ( jettisoned ) by its original owners, generally in order to lighten the ship, whether it floats or sinks.*

Your text goes here . . .

# 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. |

# 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