***Software Engineering Project Report***

**

***A Sample Document for   
Generating Consistent Professional Reports***

***Prepared by***

***John T. Bell***

***for use in* CS 440**

**at the**

**University of Illinois Chicago**

**September 2018**

### ***REMOVE OR REPLACE ALL TEXT IN RED ITALICS BEFORE SUBMITTING REPORT***

### ***How to Use This Document***

*This document is intended as a sample template that can be copied and edited to suit a particular software engineering project. It was assembled from a combination of documents [1], [2], and [3].*

***Styles***

*This document was written in Microsoft Word, and makes heavy use of styles. The styles dialog is initially located on the menu bar under the “Home” tab in MS Word. It is recommended that the styles dialog be pulled off into a separate window when working on formatting of the document. If each paragraph is assigned a style, then modifying that particular style will affect all paragraphs in the document having the same style.*

*The table of contents uses the document headings and sub-headings to automatically generate table of contents information.*

***Tracking Changes and Multiple Authors***

*The “Review” tab in MS Word contains several tools that are of particular use when editing large documents, particularly when multiple authors are involved:*

*The “Tracking” section allows you to track the ( proposed ) changes to a document, and to step through each proposed change to either accept or reject the proposed changes.*

*The “Compare” section allows you to merge changes proposed by different authors, ( which will be marked in separate colors for identification ), and then to use the change tracking tools described above to accept or deny each change.*

*The recommended procedure is to start with each author having a copy of a base document, ( possibly this template. ) Then each author changes the section(s) they are responsible for, and submits their changed version to one person who acts as the overall document editor. This author merges the changes, selectively accepts or rejects each change, and then distributes a new base document to all authors for the next round of changes. It is also possible to merge the changes and then distribute the document, so that all authors can review the proposed changes. ( The latter approach may be appropriate for documents such as bylaws, in which the changes must be approved by a committee or a vote before they can be accepted. )*

***Dealing With Material that is Unwanted ( Right Now )***

*Much of this document includes material that is not needed for every project, and/or which may not have been written yet, and so should be removed before printing or distributing the document. There are several ways to do this, however each has their drawbacks:*

1. *Delete the material completely. The drawback is that now it is completely gone, and the only way to get it back is to copy it from some other document, if that is even available.*
2. *Change the font to "Hidden". The drawback is that this does not affect the numbering of sections, either in the text or in the table of contents. However the original style information is retained, so when the text is unhidden, individual paragraphs do not need to be restyled.*
3. *Change the style to "Hidden". This does cause the document to renumber properly, ( because the paragraphs are no longer numbered paragraphs ), but all the original style information is gone, so if it is unhidden later, then all the styles of all the unhidden paragraphs will have to be restored manually.*
4. *Move the text, say to the end of the document, and then do not print or distribute the material at the end of the document. If this is all that is done, then this material will continue to appear in the table of contents. However if the font is also changed to "hidden", then the material will not appear in the TOC, but all the original style information will be retained. The drawback to moving the text is that now you need to keep track of where the information was moved from, in case you ever want to move it back.*

*Hidden text can be made visible by selecting the backwards "P" paragraph mark on the "Home" tab.*

***Table of Contents***

*The table of contents lists the sections of the document and the page upon which each section starts. The table of contents may or may not include subsections, etc.*

*Microsoft Word ( and many similar programs ) has the ability to generate a table of contents automatically from section headings, and to update it when the document changes. The table of contents included in this template is automatically generated by MS Word, and can be updated by selecting the table and clicking on the menu that appears above the table at that time.*

***Note: Remove all instructional materials before finalizing and submitting this document, including this entire page and the one before it.***

**Table of Contents**

[*REMOVE OR REPLACE ALL TEXT IN RED ITALICS BEFORE SUBMITTING REPORT* 2](#_heading=h.gjdgxs)

[*How to Use This Document* 2](#_heading=h.30j0zll)

[List of Figures 9](#_heading=h.3znysh7)

[List of Tables 10](#_heading=h.2et92p0)

[I](#_heading=h.1t3h5sf) Project Description 12

[1](#_heading=h.4d34og8) Project Overview 12

[2](#_heading=h.2s8eyo1) The Purpose of the Project 12

[2a](#_heading=h.17dp8vu) The User Business or Background of the Project Effort 12

[2b](#_heading=h.3rdcrjn) Goals of the Project 12

[2c](#_heading=h.26in1rg) Measurement 13

[3](#_heading=h.lnxbz9) The Scope of the Work 14

[3a](#_heading=h.35nkun2) The Current Situation 14

[3b](#_heading=h.1ksv4uv) The Context of the Work 14

[3c](#_heading=h.44sinio) Work Partitioning 17

[3d](#_heading=h.2jxsxqh) Competing Products 19

[4](#_heading=h.z337ya) The Scope of the Product 19

[4a](#_heading=h.3j2qqm3) Scenario Diagram(s) 20

[4b](#_heading=h.1y810tw) Product Scenario List 20

[4c](#_heading=h.4i7ojhp) Individual Product Scenarios 20

[5](#_heading=h.2xcytpi) Stakeholders 21

[5a](#_heading=h.1ci93xb) The Client 21

[5b](#_heading=h.3whwml4) The Customer 21

[5c](#_heading=h.2bn6wsx) Hands-On Users of the Product 22

[5d](#_heading=h.qsh70q) Maintenance Users and Service Technicians 23

[5e](#_heading=h.3as4poj) Other Stakeholders 23

[5f](#_heading=h.1pxezwc) User Participation 24

[5g](#_heading=h.49x2ik5) Priorities Assigned to Users 25

[6](#_heading=h.2p2csry) Mandated Constraints 26

[6a](#_heading=h.147n2zr) Solution Constraints 26

[6b](#_heading=h.3o7alnk) Implementation Environment of the Current System 27

[6c](#_heading=h.23ckvvd) Partner or Collaborative Applications 28

[6d](#_heading=h.32hioqz) Off-the-Shelf Software 29

[6e](#_heading=h.1hmsyys) Anticipated Workplace Environment 30

[6f](#_heading=h.41mghml) Schedule Constraints 31

[6g](#_heading=h.2grqrue) Budget Constraints 32

[7](#_heading=h.vx1227) Naming Conventions and Definitions 32

[7a](#_heading=h.3fwokq0) Definitions of Key Terms 32

[7b](#_heading=h.1v1yuxt) UML and Other Notation Used in This Document 33

[7c](#_heading=h.4f1mdlm) Data Dictionary for Any Included Models 34

[8](#_heading=h.2u6wntf) Relevant Facts and Assumptions 35

[8a](#_heading=h.19c6y18) Facts 35

[8b](#_heading=h.3tbugp1) Assumptions 35

[II](#_heading=h.28h4qwu) Requirements 37

[9](#_heading=h.nmf14n) Product Use Cases 37

[9a](#_heading=h.37m2jsg) Use Case Diagrams 37

[9b](#_heading=h.1mrcu09) Product Use Case List 38

[9c](#_heading=h.46r0co2) Individual Product Use Cases 39

[10](#_heading=h.2lwamvv) Functional Requirements 39

[11](#_heading=h.111kx3o) Data Requirements 40

[12](#_heading=h.3l18frh) Performance Requirements 42

[12a](#_heading=h.206ipza) Speed and Latency Requirements 42

[12b](#_heading=h.4k668n3) Precision or Accuracy Requirements 43

[12c](#_heading=h.2zbgiuw) Capacity Requirements 43

[13](#_heading=h.1egqt2p) Dependability Requirements 44

[13a](#_heading=h.3ygebqi) Reliability Requirements 44

[13b](#_heading=h.2dlolyb) Availability Requirements 44

[13c](#_heading=h.sqyw64) Robustness or Fault-Tolerance Requirements 45

[13d](#_heading=h.3cqmetx) Safety-Critical Requirements 46

[14](#_heading=h.1rvwp1q) Maintainability and Supportability Requirements 47

[14a](#_heading=h.4bvk7pj) Maintenance Requirements 47

[14b](#_heading=h.2r0uhxc) Supportability Requirements 47

[14c](#_heading=h.1664s55) Adaptability Requirements 48

[14d](#_heading=h.3q5sasy) Scalability or Extensibility Requirements 48

[14e](#_heading=h.25b2l0r) Longevity Requirements 49

[15](#_heading=h.kgcv8k) Security Requirements 49

[15a](#_heading=h.34g0dwd) Access Requirements 49

[15b](#_heading=h.1jlao46) Integrity Requirements 50

[15c](#_heading=h.43ky6rz) Privacy Requirements 51

[15d](#_heading=h.2iq8gzs) Audit Requirements 52

[15e](#_heading=h.xvir7l) Immunity Requirements 52

[16](#_heading=h.3hv69ve) Usability and Humanity Requirements 52

[16a](#_heading=h.1x0gk37) Ease of Use Requirements 52

[16b](#_heading=h.4h042r0) Personalization and Internationalization Requirements 54

[16c](#_heading=h.2w5ecyt) Learning Requirements 55

[16d](#_heading=h.1baon6m) Understandability and Politeness Requirements 56

[16e](#_heading=h.3vac5uf) Accessibility Requirements 56

[16f](#_heading=h.2afmg28) User Documentation Requirements 57

[16g](#_heading=h.pkwqa1) Training Requirements 58

[17](#_heading=h.39kk8xu) Look and Feel Requirements 58

[17a](#_heading=h.1opuj5n) Appearance Requirements 58

[17b](#_heading=h.48pi1tg) Style Requirements 59

[18](#_heading=h.2nusc19) Operational and Environmental Requirements 60

[18a](#_heading=h.1302m92) Expected Physical Environment 60

[18b](#_heading=h.3mzq4wv) Requirements for Interfacing with Adjacent Systems 60

[18c](#_heading=h.2250f4o) Productization Requirements 61

[18d](#_heading=h.haapch) Release Requirements 62

[19](#_heading=h.319y80a) Cultural and Political Requirements 62

[19a](#_heading=h.1gf8i83) Cultural Requirements 62

[19b](#_heading=h.40ew0vw) Political Requirements 63

[20](#_heading=h.2fk6b3p) Legal Requirements 64

[20a](#_heading=h.upglbi) Compliance Requirements 64

[20b](#_heading=h.3ep43zb) Standards Requirements 65

[III](#_heading=h.1tuee74) Design 65

[21](#_heading=h.4du1wux) System Design 65

[21a](#_heading=h.2szc72q) Design goals 65

[22](#_heading=h.184mhaj) Current Software Architecture 66

[23](#_heading=h.3s49zyc) Proposed Software Architecture 66

[23a](#_heading=h.279ka65) Overview 66

[23b](#_heading=h.meukdy) Class Diagrams 66

[23c](#_heading=h.36ei31r) Dynamic Model 66

[23d](#_heading=h.1ljsd9k) Subsystem Decomposition 66

[23e](#_heading=h.45jfvxd) Hardware / software mapping 67

[23f](#_heading=h.2koq656) Data Dictionary 67

[23g](#_heading=h.zu0gcz) Persistent Data management 67

[23h](#_heading=h.3jtnz0s) Access control and security 67

[23i](#_heading=h.1yyy98l) Global software control 67

[23j](#_heading=h.4iylrwe) Boundary conditions 67

[24](#_heading=h.2y3w247) Subsystem services 67

[25](#_heading=h.1d96cc0) User Interface 67

[26](#_heading=h.3x8tuzt) Object Design 68

[26a](#_heading=h.2ce457m) Object Design trade-offs 68

[26b](#_heading=h.rjefff) Interface Documentation guidelines 68

[26c](#_heading=h.3bj1y38) Packages 68

[26d](#_heading=h.1qoc8b1) Class Interfaces 68

[IV](#_heading=h.4anzqyu) Test Plans 68

[27](#_heading=h.2pta16n) Features to be tested / not to be tested 68

[28](#_heading=h.14ykbeg) Pass/Fail Criteria 68

[29](#_heading=h.3oy7u29) Approach 68

[30](#_heading=h.243i4a2) Suspension and resumption 68

[31](#_heading=h.j8sehv) Testing materials ( hardware / software requirements ) 69

[32](#_heading=h.338fx5o) Test cases 69

[33](#_heading=h.1idq7dh) Testing schedule 69

[V](#_heading=h.42ddq1a) Project Issues 69

[34](#_heading=h.2hio093) Open Issues 69

[35](#_heading=h.wnyagw) Off-the-Shelf Solutions 70

[35a](#_heading=h.3gnlt4p) Ready-Made Products 70

[35b](#_heading=h.1vsw3ci) Reusable Components 70

[35c](#_heading=h.4fsjm0b) Products That Can Be Copied 71

[36](#_heading=h.2uxtw84) New Problems 71

[36a](#_heading=h.1a346fx) Effects on the Current Environment 71

[36b](#_heading=h.3u2rp3q) Effects on the Installed Systems 72

[36c](#_heading=h.2981zbj) Potential User Problems 72

[36d](#_heading=h.odc9jc) Limitations in the Anticipated Implementation Environment That May Inhibit the New Product 73

[36e](#_heading=h.38czs75) Follow-Up Problems 73

[37](#_heading=h.1nia2ey) Migration to the New Product 74

[37a](#_heading=h.47hxl2r) Requirements for Migration to the New Product 74

[37b](#_heading=h.2mn7vak) Data That Has to Be Modified or Translated for the New System 75

[38](#_heading=h.11si5id) Risks 75

[39](#_heading=h.3ls5o66) Costs 76

[40](#_heading=h.20xfydz) Waiting Room 78

[41](#_heading=h.4kx3h1s) Ideas for Solutions 78

[42](#_heading=h.302dr9l) Project Retrospective 79

[VI](#_heading=h.3z7bk57) Glossary 80

[VII](#_heading=h.2eclud0) References / Bibliography 80

[VIII](#_heading=h.thw4kt) Index 80

### **List of Figures**

[Figure 1 - Context of the Work](#_heading=h.3dy6vkm)

Figure 2 - Scenario Diagram [11](#_heading=h.3dy6vkm)

**List of Tables**

[Table 1- Work Partitioning for the Product](#_heading=h.tyjcwt)

Table 2 - Product Scenario List [11](#_heading=h.tyjcwt)

# Project Description

## Project Overview

The website is for local restaurant owners to be able to better communicate with the public with better access to digital marketing support. This will allow for restaurants, who have been suffering the most during the current pandemic, to be able to communicate with marketing companies, to better serve their needs.

## The Purpose of the Project

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

Local restaurants: access to digital marketing and relevant metrics, allow for better, more localized marketing during the current pandemic related recession, provide quotes and general information for the type of marketing being asked for. In other words, help restaurants continue to get customers during an economic downturn.

### Goals of the Project

Provide businesses a platform to be able to promote and keep their business afloat due to the pandemic.

Provide those who do not know how to promote their business online a platform and guide them how to promote it and keep engagement with their customers and to get new ones within the process.

Allow businesses to avoid huge delivery fees that they get charged while working with apps like uberEats, grubhub etc.

### Measurement

Collect data on what services are being purchased and at what quantities/frequencies by the restaurant owners. Ideal goal would be for each restaurant to have at least one marketing strategy/company selected, with a return on investment. Can determine the return via surveying the restaurants and marketing firms for satisfaction and previously mentioned data collection

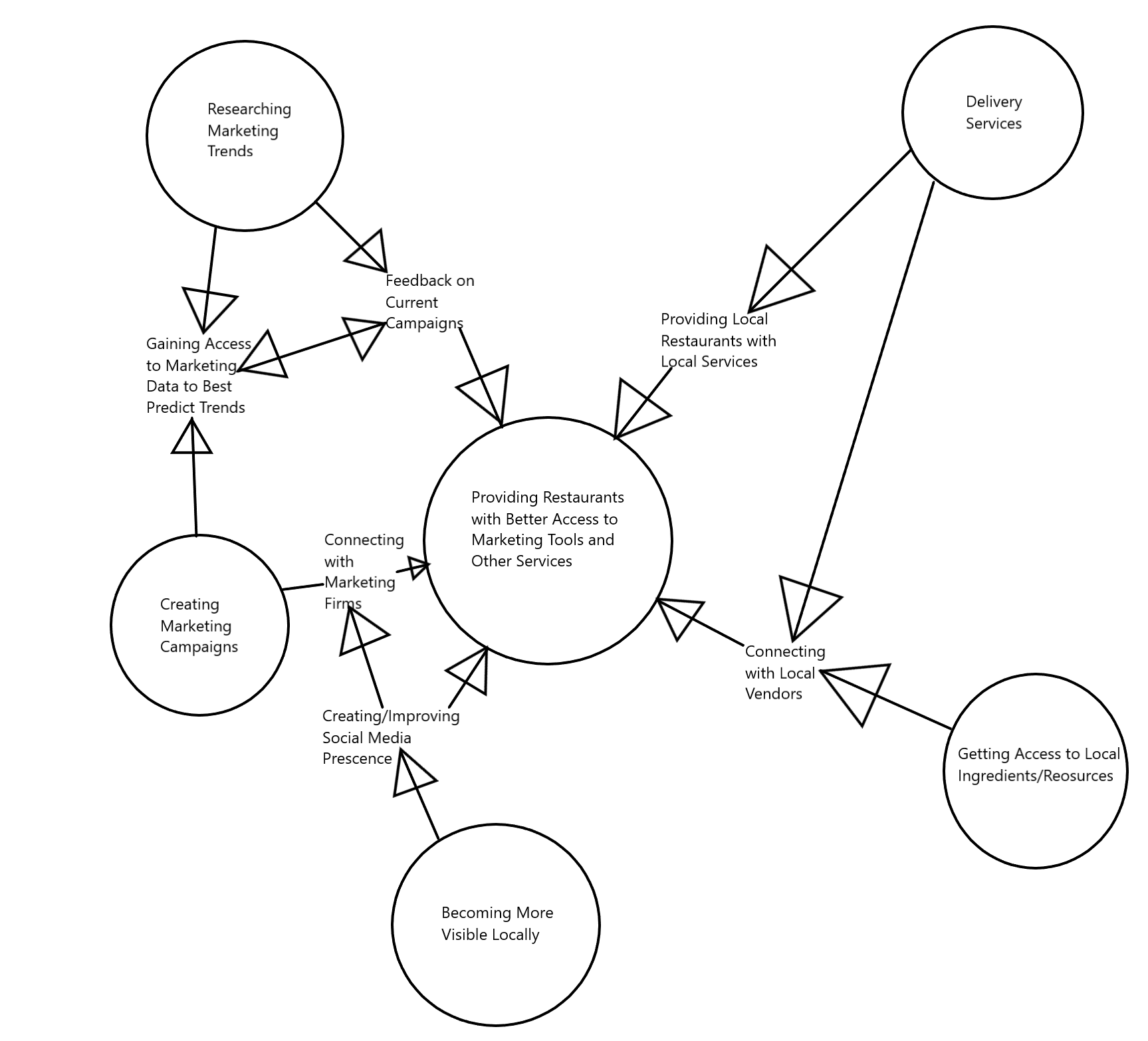
## The Scope of the Work*.*

Our product will address local restaurant’s needs for better digital marketing solutions.

### The Current Situation

During the current pandemic many restaurants are going out of business because they are unable to get enough customers for dine-in, and they do not have delivery available. A lot of ‘old-school’ restaurants do not have an online presence which means that potential customers do not know about the existence of such restaurants. With this website they will be able to combat such problems. More or less, the idea is to make it easier to get access to marketing strategies and services, without having to resort to a middleman.

### The Context of the Work



### Work Partitioning

|  |  |  |
| --- | --- | --- |
| **Event Name** | **Input & Output** | **Summary** |
| 1. Gathering client needs for marketing | Form (In) | Create a form that will allow user to select needed services and search for services not listed on the form in the beginning |
| 1. Website gives quotes for requested services | Service List (out) | For each item selected in above form, display page with companies providing services and price quotes |
| 1. Website displays breakdowns on costs and expenses | Monthly Stats Page (out) | Display the costs for all marketing services purchased monthly. Pie chart to graph where the money is being spent. Create a button to pay monthly bill |
| 1. Restaurant owner/manager requests to pay bill | Billing and Accounts Page (in/out) | Ask for the user to log in again. When authenticated, show the amount owed and owed to whom. Feature the payment option after the owed amount is shown. Also same screen for actually buying services from firms/contractors. |
| 1. Work Order is Placed | Service List (out) | When a service is selected by the user and the service selected for purchase, page loads in a in-page confirmation and delivers user to specifications screen |
| 1. Work Order Specifics Recorded | Work Order Page (in) | Form is generated for the work order, displaying the firm/contractor service is being purchased from, type of service, and cost per time period. User can request specific changes from the standard service through a text box that sends request to firm/contractor after order submission and payment |

### Competing Products

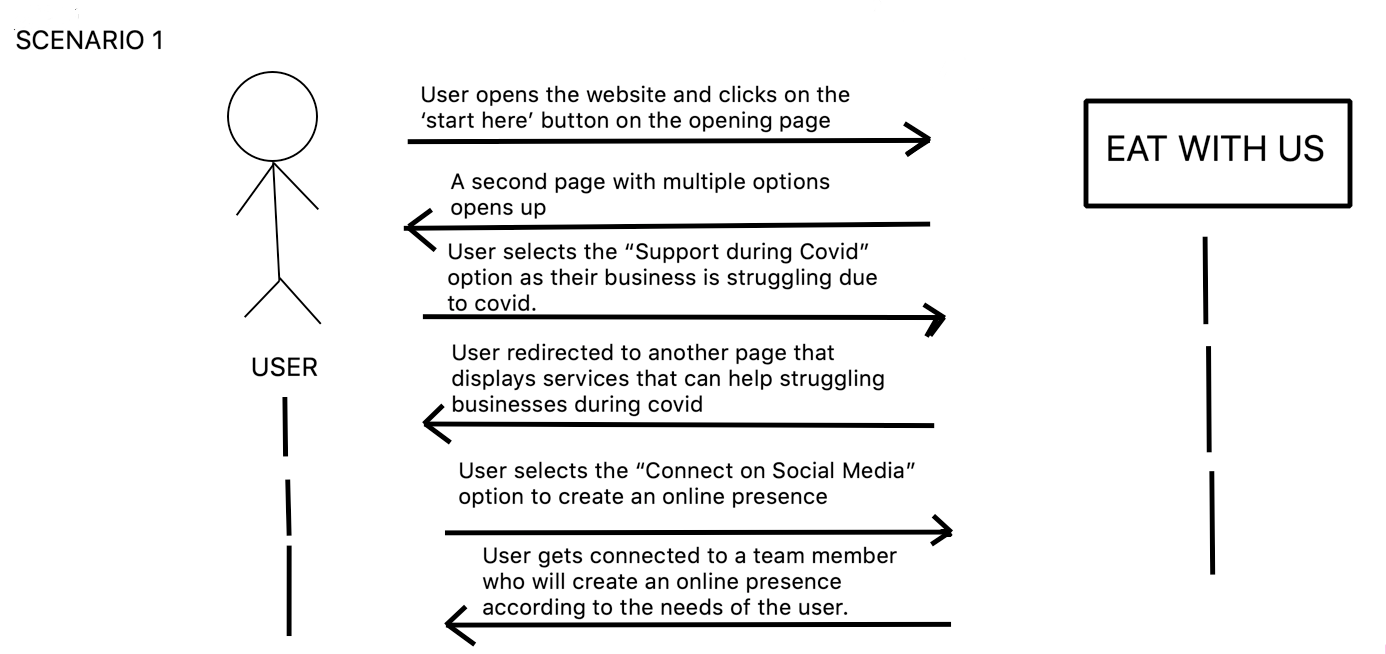
There are many delivery services available like: ubereats, postmates, grubhub which help restaurants set up delivery services and give them an option to be found online when a customer is ordering through those apps, but these platforms charge a 30-40% service/delivery fee per order from these restaurants, which often leads to them not making enough money to stay in business. And while there are websites that offer marketing solutions, they are not combined with other restaurant related services, driving the total cost up for the restaurant owner.

With this website, such businesses will now have a chance to have full control over their online presence and will be able to avoid paying huge amounts of money to third party websites/apps.

## The Scope of the Product

The product is designed to handle the majority of the work related to getting a marketing campaign underway, from finding the correct strategy to employ to setting up the meeting between the restaurant and the marketing firm that can best serve their needs, to handling the financial side of the marketing agreements (payments, subscriptions).

### Scenario Diagram(s)



### Product Scenario List

|  |  |
| --- | --- |
| Scenarios | External actors involved |
| 1. User opens the website and clicks on the ‘start here’ button on the opening page | Internet Connection and Connected Device |
| 2. A second page with multiple options opens up | Internet Connection and Connected Device |
| 3. User selects the “Support during Covid” option as their business is struggling during covid | Services provided by local vendors |
| 4. User redirected to another part that displays services that can help struggling businesses during covid | Companies that provide those services |
| 5. Users select the “Connect on Social Media” option to create an online presence. | Information has been filled out |
| 6. Users gets connected to a team member who will create an online presence according to the needs of the user. | Availability of the team and team’s connection |

### Individual Product Scenarios

A restaurant owner finds out about our service and is interested in avoiding huge fees associated with third party apps and delivery services such as ubereats. His business is already struggling due to the pandemic and on top of that his business is charged 30-40% in delivery fees. This has led to the business struggling to keep its doors open. He opens our website and reads the description of what the website offers. He then clicks on the “Start Here” button, which redirects him to a second page where he can see all the services that are offered. After this, he clicks on the “Current Issues” button and is prompted to fill out a form about all the issues his business has been facing due to the pandemic. After this he is connected to a team member who helps set up his profile. After this, he adds menu items and pictures to finish the profile set up. Now, he can also put in work orders and get in contact with local vendors for a small monthly fee. He can also see his monthly expenses breakdown with detailed graphs and charts.

## Stakeholders

### The Client

Currently, there is no client attached to the product, and as such, the development team (Group 13) is acting as the client until one can be found.

### The Customer

There will be two major types of customer that will be paying a subscription when using the product: restaurant owners/managers and marketing firms/contractors. The restaurant owner/manager will be subscribing to the product as a way to help grow their business during a period of economic recession and help to bring in customers. The marketing firm would not be paying a subscription per se, but more of a percentage of the money used by the restaurants to purchase their services. For the restaurants, the subscription will be a small amount of money, no more than $100/month, while the marketing firms would be paying at maximum 40% of the sale per purchase of their services.

### Hands-On Users of the Product

Generally, the main users of the website will be restaurant owners and managers, who will be responsible for running the restaurant and promoting it to attract business. In terms of restaurant experience, the user should be a journeyman or master at their jobs, so more technical language related to the restaurant industry would be applicable and appropriate to use. There will be another type of user on this website as well, the marketing firms showcasing their services on the website. Naturally, they are going to be masters at their craft, and should be capable of using the product with minimal effort on their part.

The website is meant to make the job easier for the user and not want to tear their hair out from an overly complicated program. In other words, the program should be simple enough that a novice with very little experience with technology can use it easily and effectively with minimal effort.

The website itself should be neutral towards concepts like age or gender identity, and should not make any distinction from person to person unless requested. There should be an option to change the language used for it, however, as it is not only English speakers that run restaurants/marketing firms and would be using the product.

Just because someone may have a disability does not mean that they are incapable of running a restaurant, so it is imperative that accessibility is key for the product. When dealing with physical disabilities, there needs to be the option for tools like text-to-speech and vocal dictation present in the website, as well as an option to change the size of icons, text, and other items in the UI that may be difficult to see for someone with poor eyesight.

When dealing with intellectual disabilities, simplicity is key. Any colors on the screen should NOT be blazingly bright, as in neon colors, and should be calmer, more neutral tones. Sounds made by the website should not be as loud as to be disorienting for someone with auditory sensitivity. The language on the site should be as simple as possible without the loss of information, as in complex enough for the message to get through completely, but not so complicated that people would get lost reading it.

### Maintenance Users and Service Technicians

Installation, maintenance, and updating will be done by the main users, restaurant owners/managers and marketing firms. The main idea is that installation should be as simple as logging onto the website, and that updates can be done automatically when new updates are pushed to production. If there are needs outside of basic operations (updating and such), there will be assistance available either through emailing the developers for help or trying to do self help through a general help wiki of sorts.

### Other Stakeholders

As it currently stands, the project currently does not have stakeholders, but will hopefully be able to attract the attention of a major firm involved in either the technology or marketing firms. As a matter of thought however, there are a few other types of stakeholders that would be interested in the product not yet listed:

Business Experts: As stated earlier, the product is designed to help restaurants get better connections to marketing firms. A product like this would be very beneficial to Business Experts and Business Assessment firms, as it means that they have to do less research and decision making on marketing and can focus more on other, more pertinent areas. In the event that there may be a business expert and a marketing expert/firm are both stakeholders and wishing to have access to the same information, this can be remedied easily

Testers: Building any major program requires quite a bit of testing, and this one will be no different to say the least. Considering that the website will be handling payment for marketing services, it will need extensive testing to ensure that there will be no issues with payment processing, either on our end or through a service like PayPal. The Product and Quality Assurance Testers will be absolutely instrumental in the development of the product, and will work hand in hand with the Security Experts in the wings.

Security Experts: Security is going to be the top priority for the website outside of its main purpose. The restaurants and the marketing firms will have accounts created on the site, and many will be paying for services through it as well. With this in mind, we would want to have security as airtight as possible, to minimize any attempts to break into a customer’s account or gain access to financial information. There is also the highly unlikely possibility of a malicious person gaining access to a restaurant’s account and ordering a marketing campaign that goes directly against the business. By having a Security Expert as a stakeholder for the product, the product will be prodded and poked constantly through development, to ensure that the worst that can happen does not and that any security concerns are minimized.

### User Participation

The users will be massively important to the development of the product, as they are going to be the main factor for shaping the website properly. The user will be providing input on usability requirements, as in what services need to be included besides the basic functionality, ratings systems, as in feedback on the quality of the provided services, and in terms of interface design, as in how they wish for the website to look and how they can customize their profile and company page(s).

In simple terms, there will be allocation needed for user input on the development as well as feedback on design, bugs, etc.

### Priorities Assigned to Users

Key User(s): The restaurant owners/managers are our biggest concern for feedback, as they are the reason the product is being developed in the first place. Any issues generated by them are to be of the highest priority, and any suggestions should be at least slightly considered. Suggestions made by them should only be rejected if they would impact the whole of the product and/or its other users.

Secondary Users:The marketing firms advertising their services on the platform will be second behind the key users for priority, but only just slightly, as they are still what the key users will be spending money on. They should be treated respectfully when they have issues or suggestions, but if those suggestions were to cause problems for the key users, they are to be ignored and/or politely rejected if possible.

Unimportant Users: The unimportant users are pretty much the ones browsing the site without having spent much time on it or without engaging with the service providers on the site. In other words, if they are just browsing without spending, anything they say, unless it is informing the team of a product-breaking issue, should be ignored. If needed, we can deliver these types of users advertisements and disable them if they create an account on the site for a trial period, and then disable permanently if they spend purchase services through the product. Any users that misuse the product should be dealt with directly. For the first offence, a warning, and the next offence will result in the deactivation of their account and a product wide ban if the offence is severe enough

## Mandated Constraints

### Solution Constraints

Description: The product will be capable of operating on multiple different web browser clients (Google Chrome, Microsoft Edge, Mozilla Firefox, Opera, Apple Safari).

Rationale: The clients will almost definitely not be delegating themselves to just one web browser, and will be accessing the product from multiple sources.

Fit Criterion: The product will be tested to be compatible with the listed browsers.

Description: The product will operate as lightly as possible, and try to minimize the amount of data and memory needed to operate it.

Rationale: Clients will not all have the most up to date and powerful PCs within their workplace and will have varying broadband speeds and possible data caps.

Fit Criterion: Each web page in the product will be no more than 1MB in size and operate using React methodology for loading elements for the user. Videos will not be hosted by the website directly, and the team will not use any video components in the product anywhere.

Description: The product will stay up to date with all security concerns during time of operation and consistently update security to minimize any hacking attempts.

Rationale: Customers will be purchasing work orders through the site, as well as paying their monthly bill. Firms/contractors will be receiving payments for work orders and also paying their bill through the site.

Fit Criterion: Development team will employ the services of security experts to continually improve on the security of the product, and will stay up to date with any major news regarding possible security/data breaches attempted.

### Implementation Environment of the Current System

Hardware needed for this system is any Graphical Operating System that has a web browser installed, capable of downloading files if needed, and a network connection. Development communications will happen in person, online, or through a phone. For customer and firm/contractor support, there will be a self-help page and a direct line to a public relations agent connected to the team if there are any problems that are not covered by the self help page.

### Partner or Collaborative Applications

The product will be compatible with reading PDFs and processing them as needed, as well as giving information about the customer’s costs and expenses in the form of a Microsoft Excel spreadsheet and/or CSV file.

### Off-the-Shelf Software

The product must be packaged with a server and database to hold not only the website itself but also to hold the customer’s and firm’s/contractor’s information and accounts, and must also come with a payment processing service as well, to handle the transactions that will happen through the site.

### Anticipated Workplace Environment*.*

Ideally, the work environment would be happening in a back office, away from the noise and distractions inherent in a kitchen environment. However, this is a very ideal environment for the product to exist in, and is in no way guaranteed in every case. In a non-ideal environment, the system being put in place may be in a more hectic area of the customer’s workplace, such as the front desk of the establishment or, worst case, the actual kitchen environment. As such, if any sound is to be made by the product, it must be volume controllable, either silent in more quiet environments or capable of being heard in a loud environment. Visual alerts will not be implemented with the product, as the possible environments that the product will be operating in can become health and safety hazards if visual distractions are introduced (kitchen environments).

As the product is meant for business applications, allowing the download of forms from the website itself is needed as well, as many businesses will wish to have access to paper copies of these documents for their own records.

Returning back to the physical environments the product will be operating in, touch screen capability will be implemented as well, as having to use a keyboard and mouse in a kitchen is anathema to cleanliness, which is king in the kitchen. An easy to clean screen with minimized areas for dirt and germs to gather is key, and will hopefully be how the product is to be implemented in a kitchen environment.

### Schedule Constraints

As it has been stated by the World Health Organization, as the effects of climate change really begin to ramp up in terms of environmental and economic impact, the likelihood of another new disease spreading globally and creating a new pandemic will increase as well. The current pandemic has already claimed the livelihoods of numerous restaurants across the United States, and that number will only continue to increase until some form of herd immunity is achieved. Another pandemic will most likely follow the same pattern as the current pandemic is. Therefore, it is imperative that the product release version 1.0 before the beginning of a new pandemic, whenever that may be. Pandemics are not something that can be predicted easily or planned for, like an election or a holiday, so it would be best to get a working, complete version of the product out the door as soon as possible to mitigate the chance that the team is “caught with its pants down”.

### Budget Constraints

The max that should be spent on development should revolve around whatever the developers, in this case college students, can afford. Considering that college students are stereotypically lacking in pretty much any type of disposable income, this should be incredibly close to the floor, as in no more than $2500, which is a stretch in of itself.

## Naming Conventions and Definitions

### Definitions of Key Terms

Customer: The representative of the restaurant, either the owner of the establishment or its manager, buying the work order from the firm/contractor.

Work Order: Order from a customer to a firm/contractor specifying what service they are requesting, what special requests the customer wishes to have fulfilled, the duration that the service will take place in, and the specific type of service requested (Digital Marketing Campaign, Newspaper/Local Radio Ads, Local Delivery Service Contracts, etc.)

Local: Referring to businesses, resources, and services within the operating area of the customer.

Firm/Contractor: referring to the marketing firms, delivery service companies, and other relevant service companies that offer their services through the site.

Product: in reference to the project itself, which will operate as a website. Can be used interchangeably with “website” when referring to the project.

Service List: The directory of services provided by the firms/contractors operating on the site itself. The customer will be shown this after they input a search request to find the work order that best fits their needs.

Client: Refers to both the customers and the firms/contractors operating the product.

### UML and Other Notation Used in This Document

There is no relevant diagrams or notation used in this document so far.

### Data Dictionary for Any Included Models

Work\_Order = Invoice\_Number + Service\_Type + Specific\_Requests\_Sent\_By + Service\_Timeline + Service\_Duration + Cost\_Of\_Service + Service\_Provider + Is\_Repeating\_Request

Is\_Repeating\_Request: Boolean value that tells if this is a monthly service or a one time only service being requested in the work order

User\_Acct = User\_Name + User\_Address + User\_Workorder\_History + User\_Financial\_Info

## Relevant Facts and Assumptions

### Facts

The average traditional restaurant has a 3% to 5% profit margin

(https://upserve.com/restaurant-insider/profit-margins/)

10,000, or 17%, of restaurants in the United States have permanently closed their doors because of CoVID-19

(<https://www.cnn.com/2020/12/08/business/restaurant-closures-coronavirus/index.html>)

17% of restaurants close within the first year of operation (not the same as those closed because of the CoVID-19 pandemic)

(<https://www.forbes.com/sites/modeledbehavior/2017/01/29/no-most-restaurants-dont-fail-in-the-first-year/?sh=537e978c4fcc>)

Customers will pay up to 40% of the cost of the food being delivered to them compared to what the restaurant is charging.

(<https://techcrunch.com/2020/03/16/the-hidden-cost-of-food-delivery/>)

### Assumptions

There will be no competing product released in the time it will take to develop and release the first working version of the product to market.

There will still be support for the product’s third party components, as in payment processing and server support.

The development team will assume that there will still continue to be enough restaurants open to warrant the continued development of the product.

The development team will assume that the user will have access to the internet and all necessary software/hardware required of them, and that the user is one of the targeted users for this product.

# Requirements

## Product Use Cases

*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

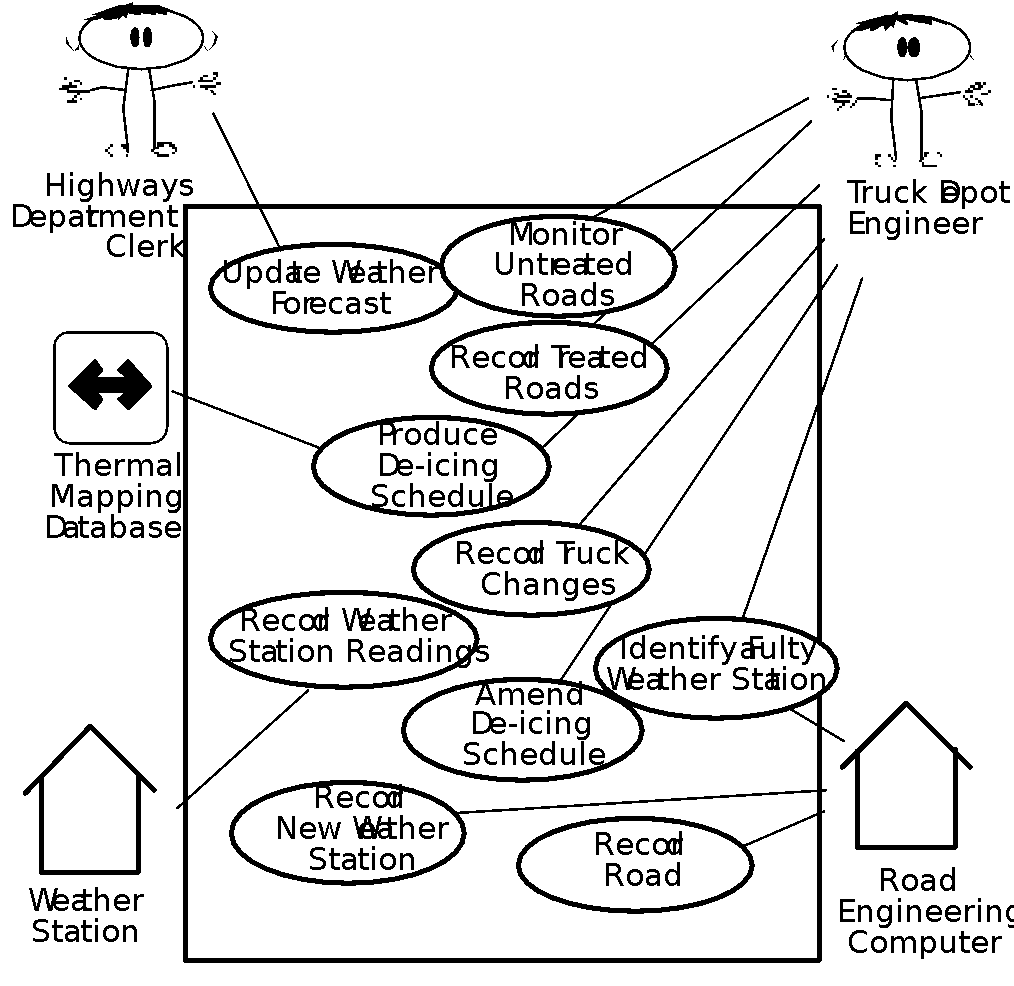
*Use Case diagrams serve two purposes: As a form of graphical table of contents listing the individual use-cases, and also to define the boundary of what is included as part of the proposed system and what is not included.*

*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.*

*Example*

**

*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.*

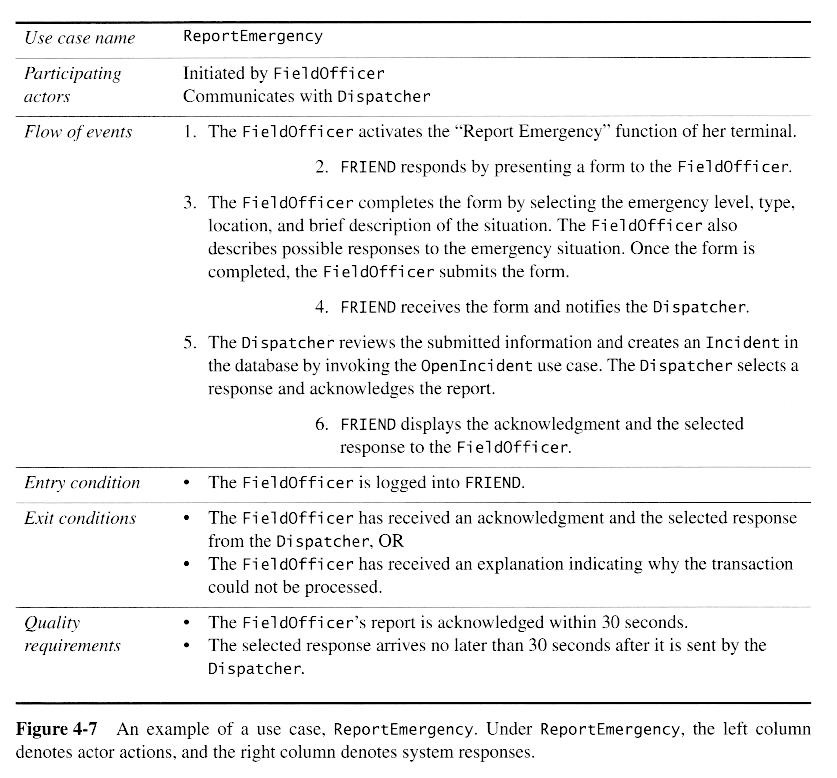
### Product Use Case List

*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.*

*For example, here is Figure 4.7 from "Object Oriented Software Engineering" by Bruegge and DuToit. . ( See also the sample Use-Case form provided on the CS 440 web site. )*

**

## Functional Requirements

*Content*

*A specification for each functional requirement. As with all types of requirements, use the requirements shell. A full explanation is included in this template’s introductory material.*

*Motivation*

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

*Examples*

**

*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.*

## Data Requirements

*Content*

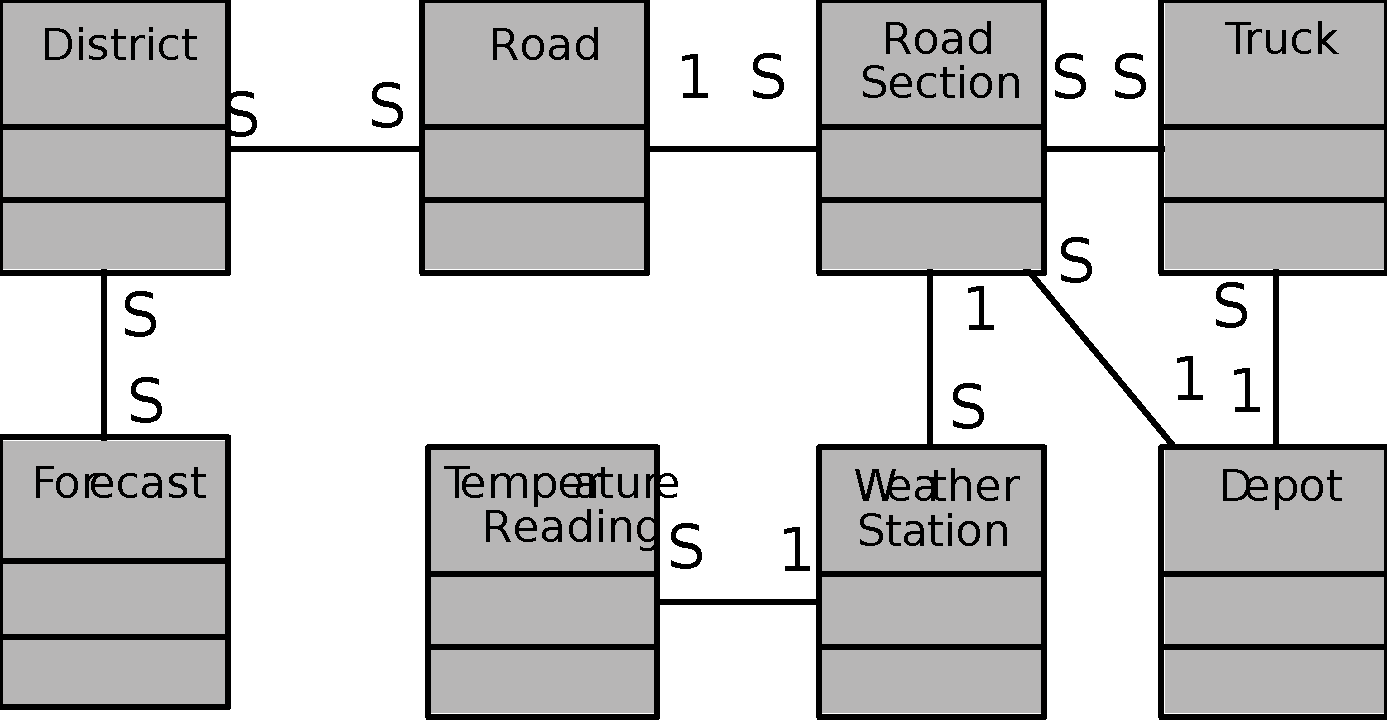
*A specification of the essential subject matter, business objects, entities, and classes that are germane to the product. It might take the form of a first-cut class model, an object model, or a domain model. Alternatively, these requirements might be described by defining the terms in the dictionary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

## Performance Requirements

### Speed and Latency Requirements

*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.*

### Precision or Accuracy Requirements

*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.*

### Capacity Requirements

*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.*

## Dependability Requirements

### Reliability Requirements

*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.*

### Availability Requirements

*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 )3and 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.*

### Robustness or Fault-Tolerance Requirements

*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.*

### Safety-Critical Requirements

*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.*

## Maintainability and Supportability Requirements

### Maintenance Requirements

*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.*

### Supportability Requirements

*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.*

### Adaptability Requirements

*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.*

### Scalability or Extensibility Requirements

*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.*

### Longevity Requirements

*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.*

## Security Requirements

### Access Requirements

*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.*

### Integrity Requirements

*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.*

### Privacy Requirements

*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.*

### Audit Requirements

*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.*

### Immunity Requirements

*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.*

## Usability and Humanity Requirements

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

### Ease of Use Requirements

*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.*

### Personalization and Internationalization Requirements

*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.*

### Learning Requirements

*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.*

### Understandability and Politeness Requirements

*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.*

### Accessibility Requirements

*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.*

### User Documentation Requirements

*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?*

### Training Requirements

*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?*

## Look and Feel Requirements

### Appearance Requirements

*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.*

### Style Requirements

*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.*

## Operational and Environmental Requirements

### Expected Physical Environment

*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.*

### Requirements for Interfacing with Adjacent Systems

*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*

### Productization Requirements

*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.*

### Release Requirements

*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?*

## Cultural and Political Requirements

### Cultural Requirements

*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?*

### Political Requirements

*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.*

## Legal Requirements

### Compliance Requirements

*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?*

### Standards Requirements

*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?*

# Design

## System Design

### Design goals

*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 Software Architecture

*SV:*

Your text goes here . . .

## Proposed Software Architecture

### Overview

*SV:*

Your text goes here . . .

### Class Diagrams

*SV:*

Your text goes here . . .

### Dynamic Model

*SV:*

Your text goes here . . .

*Content*

*Include sequence diagrams of each 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. )*

*Depending on the particular design, this section may also include finite state diagrams.*

### Subsystem Decomposition

*SV:*

Your text goes here . . .

### Hardware / software mapping

*SV:*

Your text goes here . . .

### Data Dictionary

*SV:*

Your text goes here . . .

### Persistent Data management

*SV:*

Your text goes here . . .

### Access control and security

*SV:*

Your text goes here . . .

### Global software control

*SV:*

Your text goes here . . .

### Boundary conditions

*SV:*

Your text goes here . . .

## Subsystem services

*SV:*

Your text goes here . . .

## User Interface

*SV:*

Your text goes here . . .

## Object Design

### Object Design trade-offs

*SV:*

Your text goes here . . .

### Interface Documentation guidelines

*SV:*

Your text goes here . . .

### Packages

*SV:*

Your text goes here . . .

### Class Interfaces

*SV:*

Your text goes here . . .

# Test Plans

## Features to be tested / not to be tested

*SV:*

Your text goes here . . .

## Pass/Fail Criteria

*SV:*

Your text goes here . . .

## Approach

*SV:*

Your text goes here . . .

## Suspension and resumption

*SV:*

Your text goes here . . .

## Testing materials ( hardware / software requirements )

*SV:*

Your text goes here . . .

## Test cases

*SV:*

Your text goes here . . .

## Testing schedule

*SV:*

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