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

Table of Contents

List of Figures

List of Tables

I Project Description

1 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.

2 The Purpose of the Project

2a 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.

2b 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.

2c 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

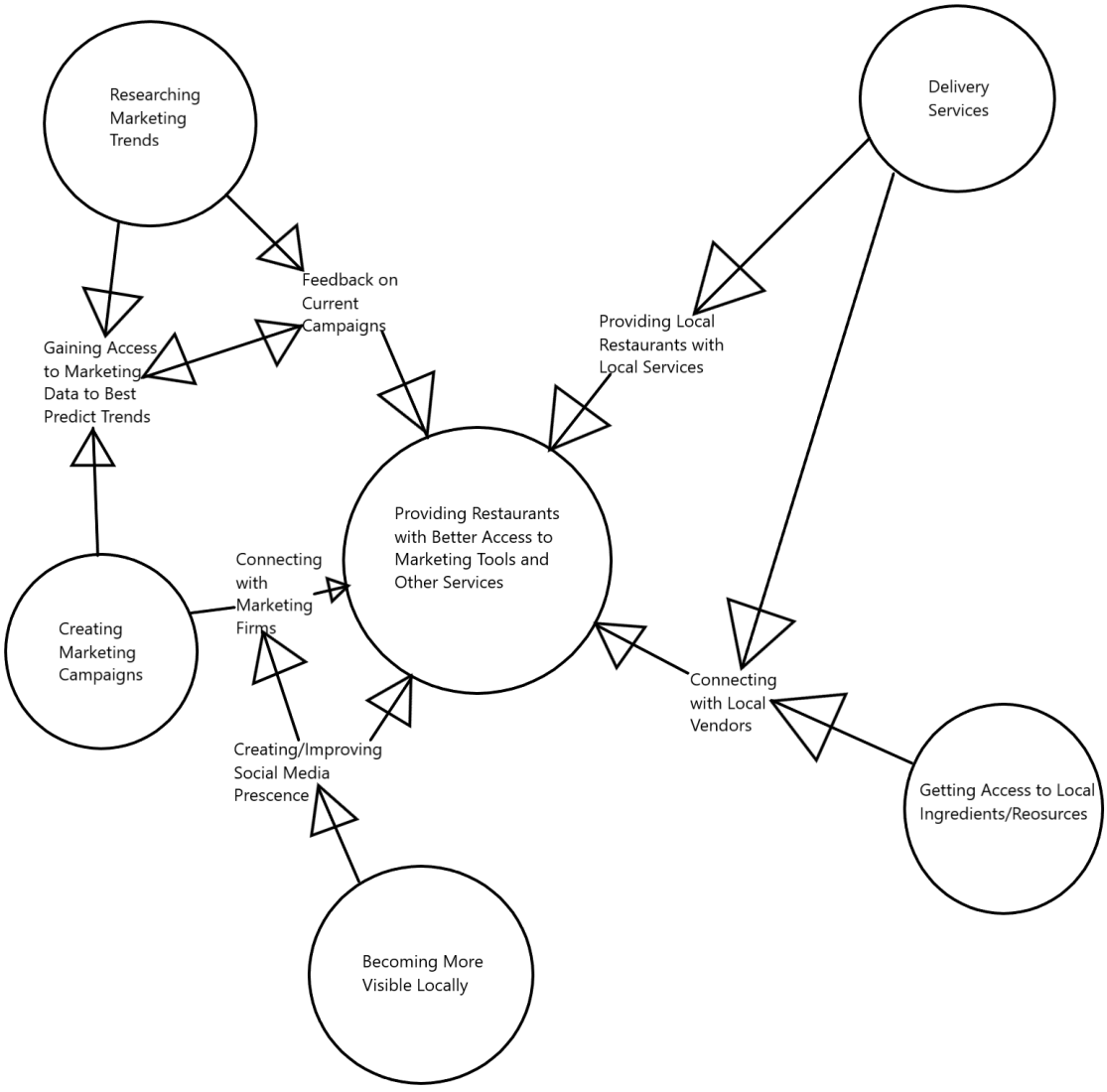
3 The Scope of the Work.

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

3a 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.

3b The Context of the Work



3c 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
2. Website gives	Service List (out)	For each item selected in

quotes for requested services		above form, display page with companies providing services and price quotes
3. 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
4. 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.
5. 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
6. 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

3d Competing Products

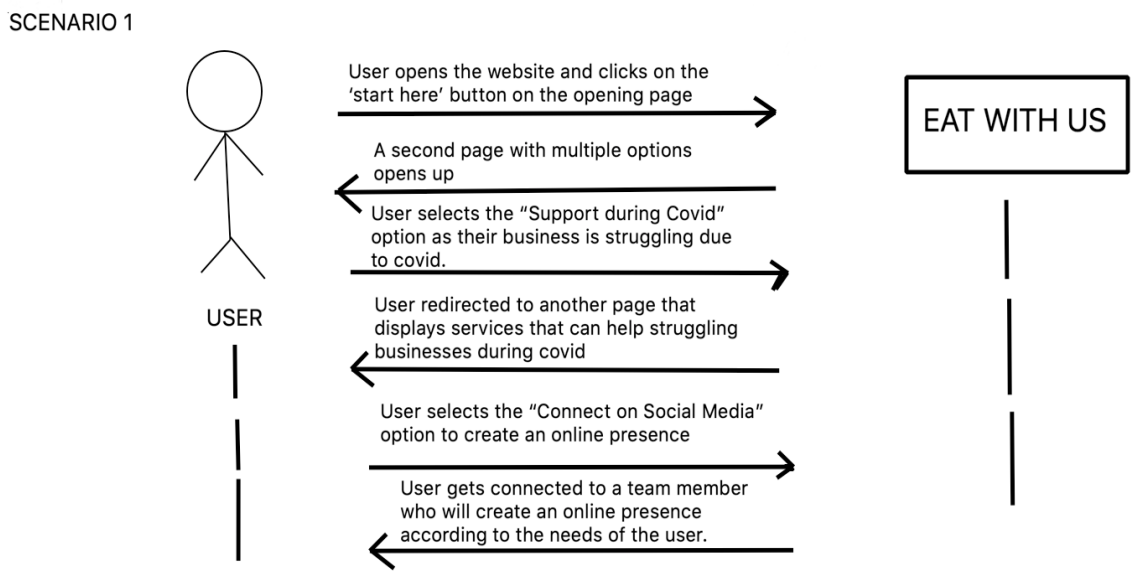
There are many delivery services available like: ubereats, postmates, grubhub which help restaurants set up delivery services and give them an option to be found online when a customer is ordering through those apps, but these platforms charge a 30-40% service/delivery fee per order from these restaurants, 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.

4 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).

4a Scenario Diagram(s)



4b Product Scenario List

Scenarios	External actors involved
1. User opens the website and clicks on	Internet Connection and Connected

the ‘start here’ button on the opening page	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

4c 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.

5 Stakeholders

5a 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.

5b 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.

5c 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.

5d 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.

5e 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.

5f 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.

5g 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

6 Mandated Constraints

6a 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.

6b 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.

6c 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.

6d 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.

6e 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.

6f 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".

6g 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.

7 Naming Conventions and Definitions

7a 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.

7b UML and Other Notation Used in This Document

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

7c 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

8 Relevant Facts and Assumptions

8a 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/)

8b 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.

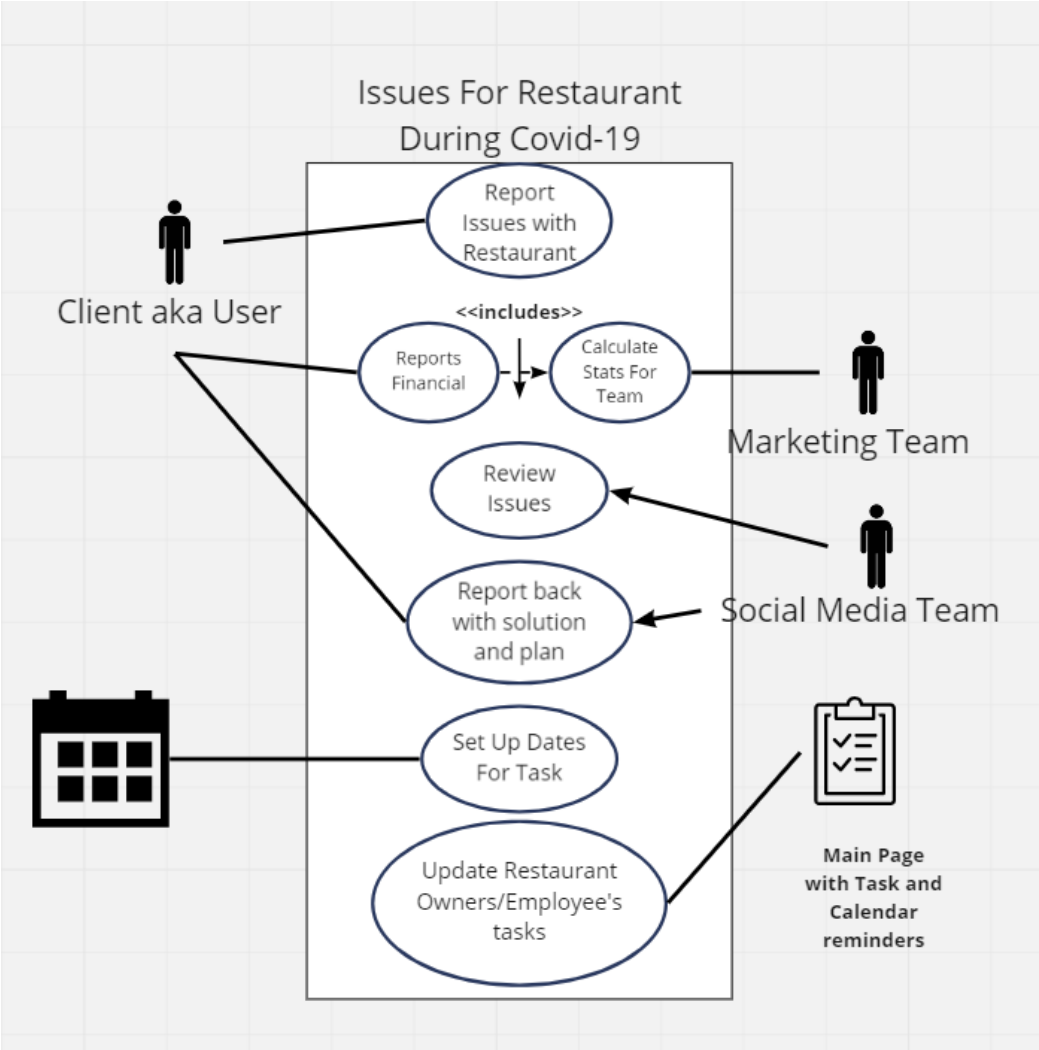
The development team will assume that the user will have access to the internet and all necessary software/hardware required of them, and that the user is one of the targeted users for this product.

II Requirements

1 Product Use Cases

1a Use Case Diagrams

Use CaseDiagrams:



1b Product Use Case List

Use Cases	Actors
ReportPaymentFailure	User, Customer service rep.
CompareMonthlyExpenses	User
AddNewService	User
GetHelp	User, Customer service rep.
Begin	User, Customer service rep.
CreateSocialMediaPresence	User, Customer service rep.

1c Individual Product Use Cases

Use case name	ReportPaymentFailure
Participating Actors	Initiated by user Communicates with customer service rep
Flow of events	1. The **user** is unable to make monthly payment and chooses the ReportPaymentFailure option. 2. Is connected to **customer service rep**., who provides a Problem Report Form. 3. The **user** completes and submits the form by filling in all the required fields and giving a summary of the problem. 4. The **customer service rep** receives the form and informs the customer that he will be contacted by a team member within 24 hours. This form is added to the CustomerComplaints database.
Entry Condition	The user is logged into their account on the website.
Exit condition	The user has received acknowledgement and selected response from a customer service rep. OR The user has been informed as to why the payment could not be processed.
Quality Requirements	User is connected to a customer service rep within a minute. The User is contacted within 24 hours with a solution to their problem.

Use case name	CompareMonthlyExpenses
Participating Actors	Initiated by user
Flow of Events	1. The user logs into their account and wants to compare their monthly expenses for past one year. 2. The user clicks on the monthly expenses tab and then clicks on the compare past expenses tab. 3. The user is redirected to a page where they can select the months they want to compare. 4. The user then selects the months and the type of data chart they want their info in. 5. The requested information is displayed.
Entry Conditons	The user is logged into their account.
Exit Condition	The user receives the info they requested. OR If the information cannot be displayed they are connected to a customer service rep.
Quality Requirements	The requested information is displayed within a minute.

Use case name	AddNewService
Participation Actors	Initiated by user
Flow of Events	1. The user logs into their account and wants to add a new service to his existing services. 2. They press the services button and are redirected to the services page displaying all their existing services. 3. The user then presses the add new service button and a menu of services offered is displayed along with the price associated with that service. 4. The user selects the required service and presses add. The service is then added to their existing services.
Entry Conditions	The user is logged into their account
Exit condition	The user is able to successfully add any desired service to their existing services.
Quality requirements	The user is able to add any desired service within 30 seconds.

2 Functional Requirements

- **Begin**
  Description: When the user clicks on the begin button, they should be prompted to fill out a form and after that the user should be connected to a customer service sales rep.
  Rationale: The user should be able to submit their requirements and become familiar with how the website works.
  Fit Criterion: Check if the user can use the website effortlessly and check if the information added by the user is correct.
  Acceptance Criteria: The user should begin using the website efficiently and the website should have general information about what the user's needs are.

- **GetHelp**
  Description: User is able to ask for help if stuck on a feature on the website.
  Rationale: It is important for users to be able to ask for help in order for them to actually benefit from using the website.
  Fit Criterion: After getting help the user is able to overcome the problem.
  Acceptance Criteria: User feedback should be positive for customer service and overall usability for the website.

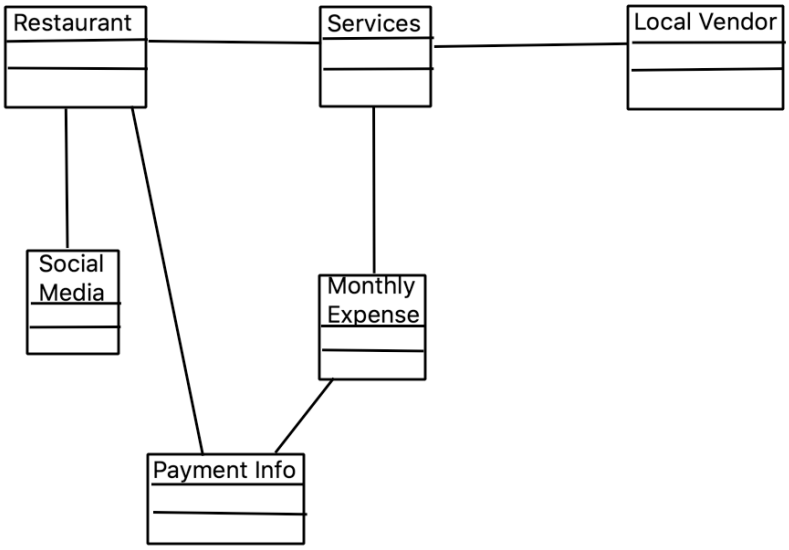
- **CreateSocialMediaPresence**
  Description: User is able to create a social media profile for their restaurants and control their online presence through the website.
  Rationale: It is important for restaurants to have a good online presence because these days people always google the place they are interested in going to.
  Fit Criterion: The user is able to create a good online profile with the help of the social media team.
  Acceptance Criteria: The profile created is fun and interesting.

- **AddNewService**
  Description: The user should be able to add new services for their restaurant.
  Rationale: It is important for the user to be able to add or remove services according to their changing needs.
  Fit Criterion: New service added without any problems.
  Acceptance Criteria: The user should be able to successfully add a new service and should be able to provide feedback if they encounter any problems.

- **CompareMonthlyExpenses**
  Description: The user checks and compares monthly expenses.
  Rationale: The user should be able to check and compare monthly expenses in order to keep their spending in check and manage the restaurant efficiently.
  Fit Criterion: The user views correct and easy to read monthly reports.
  Acceptance Criteria: When the users selects the monthly expense option, they should be able to correct reports without any problems.

3 Data Requirements

- Database containing monthly expense history for each restaurant: The database should be able to store all information regarding past and present monthly expenses for each restaurant. The user should be able to access this information. Failure to do either of these tasks will result in test failure.
- Database containing all services being used by every restaurant: The database should be able to store all the information regarding past and present services being used by every restaurant. Users should be able to view their past and current services. Failure to do either of these tasks will result in test failure.
- Customer Complaints Database: This database should be able to store all customer complaints sorted by months. Users should not be able to view this. Failure to do either of these tasks will result in test failure.



4 Performance Requirements

4a Speed and Latency Requirements

The product will operate fast enough to not interfere with the user's day to day operations. Any and all responses from the product to the user will take no longer than 3 seconds, unless there is an issue on the user's end. The same limit shall be imposed on all tasks that the user requests from the product.

Any changes made to the product during operation will be loaded in when the user next refreshes their page, where it will be automatically delivered to them.

Any reports sent out by the product to the user will take no more than one day from the time of generation to be delivered electronically.

4b Precision or Accuracy Requirements

All monetary amounts will be rounded when appropriate and possible, with the value rounding up to the nearest value, and running to two decimal places. All times within the product will be synchronized to their respective time zones. Any measurements not otherwise mentioned will use the imperial system when inside the United States, and the metric system otherwise.

4c Capacity Requirements

During standard business hours in all timezones the product is operating in, the product will cater to up to 200 users simultaneously during this period. During non-business hours, the maximum load the product will be expected to handle without issue will be 130 users. In the event of there being more users than the maximum load for the time period, the development team will attach a backup server to handle the excess load on the product.

5 Dependability Requirements

5a Reliability Requirements

No data will be lost or damaged because of a failure in the product's systems, and the product's systems will fail no more than one half-hour period a week. Any failures that occur will not last longer than that one half-hour period total. In the need of system maintenance, the product will warn the user two hours in advance, then one hour in advance, cutting this time in half until maintenance begins, as to avoid any issues with the users being blindsided with an unexpected maintenance period. These maintenance periods will be on a monthly basis at maximum, bimonthly preferably.

5b Availability Requirements

The product will be available 24 hours a day, 365 days a year without any interruptions lasting longer than half a day. Ideally, the product will achieve at least 99% uptime while in operation. This will not include maintenance periods, as they will be expected downtimes and these downtimes will be broadcasted to the users before they happen.

5c Robustness or Fault-Tolerance Requirements

In the event that the user loses connection to the internet for any reason, the product will still display all information on the current webpage being displayed to the user. If the user tries to load a new page without an internet connection, the product will not be expected to load that at all.

If the user is to lose connection to the product when entering in any information, the product will save any fields or information that has been provided beforehand, but not

expected to save any information that was being loaded into the product. This will include strings being input by the user when using a chat function in the product.

5d Safety-Critical Requirements

This product will be accessed by the user through a web browser, so the only physical component(s) needed for operating the product will be third party and not supplied by us. This does not mean that there are no safety measures in place, however.

The product will not be party to or cause any issues with the health and safety of the user and their device, as such, there will be absolutely zero cryptocurrency miners attached to the product under any circumstances. If it was there, it would impact the product's performance and the lifespan of the user's device. The product will also not be capable of causing any issues with the user's health at all, such as causing an epileptic seizure or revealing the user's location to other users and individuals outside of the product without the user's consent.

6 Maintainability and Supportability Requirements

6a Maintenance Requirements

Data verification and validation will happen on a weekly basis, at a minimum once a month, to check and correct any errors that may have occurred with the data. All general maintenance, server maintenance, database maintenance, and otherwise, will be done on a regular basis, at minimum once a month. These systems are critical to the success and operation of the product, so their health is top priority.

The product will be easy to maintain, so that if a different team than the one that made the product is in charge of the product's operation, they can do so easily. The product's code will be heavily documented, with a general summary of each major file and its function being attached to the top as a comment. The users will not be expected to maintain the product, that duty will fall to the development team alone.

6b Supportability Requirements

Most support will be accomplished with a self help page, to the extent that the product should be entirely self supporting if possible. This page will be designed the same way stylistically as the rest of the product, and will host information on how to resolve any possibly common problems that users may have when using the product. The page will be simple to navigate and easy to look and search through, where it will take no more than 3 minutes for a user to find what they were looking for on the self help page, assuming that their issue is one that is commonly reported.

However, there may be a time that the issue that a user has with the product cannot be resolved with a self help page. Therefore, the product will have a small help desk of no more than 5 employees. There will be no more than that as the self help page will be the main problem solving resource for the users, and that only the most uncommon or unsolved problems will be brought to the help desk's attention.

6c Adaptability Requirements

This product will be designed to operate not only in an office environment, but also in a front desk and restaurant kitchen environment as well. The product will be able to run on all major web browsers on desktop machines and mobile devices, which will allow it to run on any operating system that can run those browsers. In the future, the product will be able to be run in a mobile environment as a standalone application rather than using a web browser, as well as adding translations for the product for less common languages in the United States, such as Swedish or Chinese.

6d Scalability or Extensibility Requirements

The product will be able to handle up to 10,000 transactions per hour within a year of its launch, and will increase this capacity by 10,000 transactions minimum per year. As more users begin to use the product, more data will be collected from them. Therefore, the product will have a database and server storage capacity of at least forty terabytes on launch, and will multiply this capacity by two every time the current filled capacity reaches four fifths of the total capacity. In terms of bandwidth, the product will be capable of handling up to 15,000 users within the first year, bringing up this number whenever possible without any major updates to the product.

6e Longevity Requirements

The product shall be expected to operate within its operational budget without any major overhauls or upgrades for a period of three years, with updates during this time consisting almost exclusively of security updates and data redundancy operations.

7 Security Requirements

7a Access Requirements

As the product will be handling financial information for processes such as billing, the team will hire a security consultant to best prevent any security issues from happening, rather than waiting for an event to occur and handling it then and there. Physical access to the hardware components of the product, such as the main server or the databases, will be limited to those with the minimum security clearance as dictated by the security expert. At minimum, assuming that the team will be working in an office environment and not working remotely, all team members will have physical access to the main part of the office, but anywhere that would house critical components of the product or its data will be limited to the individuals listed above.

Data access for individual users will be limited to the user themselves and to a systems administrator or security expert if an issue arises with the user's data that they cannot fix themselves or are having trouble accessing. The user can also limit what data the team can access as well, assuming that the data is not critical to the team's support for the product or financial information. If the user loses access to their account, there will be two factor authentication processes and account recovery processes installed to allow the user to recover their account.

7b Integrity Requirements

All data created by the users and stored by the product's servers and databases will be checked and looked over with a program designed specifically for that purpose or by the hands and eyes of the development team when and if necessary. All data will be stored in a redundancy system, where there will be more than one copy of the collected data at any point in time, at minimum two copies of the data collection. Periodically, at minimum once every month, the data will be checked over for any corruptions or incorrectness and be fixed by hand if necessary or by restoring the data to a previous state from a redundant storage system.

The product will protect itself from intentional abuse by the user and from any attempts to intentionally destroy data without the express go ahead from the development team. Any attempts by a user to abuse the product or its data will result in a monetary fine if minor enough, and being completely removed and banned from the platform if severe enough.

7c Privacy Requirements

The product will collect the minimum amount of data necessary to protect the development team from issues related to financial records and evidence of communications, and none of it will be sold to third parties or used by the team to gather advertising revenue. All information will be stored within a system of heavily encrypted databases, where new data will be held in a database connected to the system network for a period of one month, after which they will be moved to a database not connected to the development network, and will only be connected to it for the transfer of data. This will help to mitigate the chance that large amounts of data are lost to a malicious attack. This system will be accessible to the customer should they request that they wish to correct any non-financial information that has been collected.

As for smaller bits of information, such as cookies, the product will include the bare minimum amount of data necessary to have these services be operational.

The users will be made aware of the information policy and be notified of any changes of this policy that may happen at any time. They will be notified when their stored/collected data is being scheduled for a transfer to the offline database and when this transfer occurs.

7d Audit Requirements

The product will record who visits the site itself and stores that as an encrypted log file. This file will be stored the same way as the data mentioned in the above section, inside a system of encrypted databases. All financial transactions and related information will be encrypted and stored in the same way, in case of an event like a warrant or federal inquiry.

7e Immunity Requirements

The product will be tested thoroughly by security experts and will be equipped with enough security measures to give the users as much protection and confidence as possible when using the product. The product will scrub all inputs as standard and will employ a CAPTCHA service to limit the possibility of bot activity on the product.

8 Usability and Humanity Requirements

8a Ease of Use Requirements

- The product should be tested to check if it can be easily used by people of different ages and experiences, it should also be checked for ease of use with users who do not understand english.
- The product/website can be used easily irrespective of age. The user does not need to remember much about using the product as the website is extremely straightforward and users are usually redirected from one page to another without any confusion. The chances of error committed by the user are very low as every link and heading is pretty self explanatory. The user will also have the option to chat with a team member in case they are confused about a feature or want to know if they are using the website correctly.
- Motivation: The product is easy enough for anyone to use irrespective of age or the language they speak. It is almost impossible to make mistakes as the product is straightforward and if a user clicks on a wrong link they can always go back to the previous page by pressing the back button. The website will also have an option to translate all the info into different languages, so it will be easy to use for someone who doesn't know english.

8b Personalization and Internationalization Requirements

- Initially the product will only be available in the US, but as its use extends to other countries, users will be able to select the currency according to the country it is being used in, and the user will also have an option to select a language of their choice. The website and the info on it will be available in most languages used worldwide.
- In the monthly expenses page, users will be able to decide the type of graph (pie chart, histogram etc) they want to view their info in. They will also be able to change colors on the monthly expenses page.

8c Learning Requirements

- The product should be as easy to use for a non-tech savvy person as it is for a tech savvy person.
- The user will be able to learn how to use the product properly after just one use.
- Any person, old or young, irrespective of the language they speak and the country they are from should be able to learn how to use the website easily after just one use.

8d Understandability and Politeness Requirements

The language used in the product should be complex enough to be understood by people in the industry, but simple enough for anyone else to understand. In other words, the language should include the user's industry jargon when necessary but be simple, short, and concise everywhere else. Any language used should be as neutral as possible and free of any controversial or divisive terminology, words, phrases, or other expressions.

In terms of the product's construction, the product will hide all of that information from the users, as they will not need to know or understand how it works to be able to use the product.

8e Accessibility Requirements

The product shall be compatible with standard text to speech software, with a special UI generated if needed. There will also be options for changing the font to one that is easier to read for people with dyslexia, as well as color change options for different types of color blindness as well. The product will be compliant with the Americans with Disabilities Act for all other accessibility options and disabilities not listed here.

8f User Documentation Requirements

The only documentation that will be supplied with the product will be the EULA, the product will be simple enough that there will be no need to include any major documentation with it.

8g Training Requirements

Training will be minimal, as the product will be designed to be as simple as possible. The users are expected to already be busy with their day to day operations, and will not be expected or required to be trained to use the product. At most, the development team will create a basic guide to use the software, though the design should be made as simple as possible to avoid any issues on how to use the product.

9 Look and Feel Requirements

9a Appearance Requirements

The product will use neutral tones for the color scheme, and will not use anything that would be incredibly bright/obtrusive to the user's experience. The product will comply with corporate branding standards, and should allow for the user's corporate logo to be displayed on the user's personalized homepage.

9b Style Requirements

The UI will be as minimalistic and clutter free as possible. The appearance should cause a positive reaction in most of its users. The product should look and feel as simplistic as possible, where most users will feel that the product is easy to use and satisfactory for their needs. However, the minimalist design will not impact any user functionality or usability at all.

10 Operational and Environmental Requirements

10a Expected Physical Environment

The product will be used in two environments: a kitchen environment and a back office environment. In the kitchen, the product should be loud enough to overcome the noises of a busy kitchen, such as the dishwasher and cooking sounds. In a back office environment, noise generated by the product should be no more than half the noise that should be expected in the environment. All physical requirements will be handled by the device the user operates to access the product, as the product as seen by the user is a website.

10b Requirements for Interfacing with Adjacent Systems

The product will work on the last five releases of the ten most popular web browsers. The data content will be a standard web page, with general features and languages attached (JavaScript, PHP, etc.) and the product will be able to handle frequent access from the web browsers, at minimum once every hour. The product will also be capable of operating and communicating with its server(s) and databases up to the previous two versions of their operating systems. Their contents will be the actual website and its features and the data collected from the users respectively.

10c Productization Requirements

The product should be easily accessible through any device with a search engine on it. Since the product is a website, all the user needs is a smartphone/tablet/pc, a search engine installed on the device and an account on the website.The user will just have to look up the name of the website on a search engine to be able to open it and use it. Once the user subscribes to the website, they will have to pay a monthly subscription fee in order to continue using it.

10d Release Requirements

New functionalities and features will be added to the website regularly.

For example: In the beginning the website will only be functional in the US, and with time it will be made available in more and more countries with the options of more and more languages and currencies. These changes will come with different releases of the website.

11 Cultural and Political Requirements

11a Cultural Requirements

The product will be culturally neutral if possible, and will attempt to conform to cultural standards as much as possible as the product begins to move outside of the United States and into the global market. As such, the product will keep a record of all public holidays in all areas that the product is available and operating in. Also, the product will not be offensive to any religious or ethnic group.

11b Political Requirements

The product will be politically neutral, and will not comment on any political events happening or act as a statement for or against any political statement, unless said statement is to deny someone their human rights, in which case it can be used only to be against said statement. The product is expected to be developed and hosted in the United States. That is not a guarantee, however, and it is expected that the tools and resources used when developing the product will come from local sources.

In terms of corporate politics, the idea is the same, that the product will be politically neutral as much as it can be. That does not mean that the product and/or the development team should not be responding to any potential competition.

12 Legal Requirements

12a Compliance Requirements

The product will handle and implement personal information in compliance with the Data Protection Act, and handle all accessibility requirements in compliance with the American with Disabilities Act. In regards to all laws on financial information and other unrelated laws, it will be handled as to be compliant with any and all applicable laws. The product will be compliant with all applicable laws within the nations that the product will be operating in as well.

12b Standards Requirements

The product will be compliant with all industry standards in regards to server implementation; database maintenance and construction; website implementation,

development, maintenance and design; and all standards for all other parts of the product not mentioned above.

III Design

1 System Design

1a 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 . . .

2 Current Software Architecture

SV:

Your text goes here . . .

3 Proposed Software Architecture

3a Overview

SV:

Your text goes here . . .

3b Class Diagrams

SV:

Your text goes here . . .

3c 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.

3d Subsystem Decomposition

SV:

Your text goes here . . .

3e Hardware / software mapping

SV:

Your text goes here . . .

3f Data Dictionary

SV:

Your text goes here . . .

3g Persistent Data management

SV:

Your text goes here . . .

3h Access control and security

SV:

Your text goes here . . .

3i Global software control

SV:

Your text goes here . . .

3j Boundary conditions

SV:

Your text goes here . . .

4 Subsystem services

SV:

Your text goes here . . .

5 User Interface

SV:

Your text goes here . . .

6 Object Design

6a Object Design trade-offs

SV:

Your text goes here . . .

6b Interface Documentation guidelines

SV:

Your text goes here . . .

6c Packages

SV:

Your text goes here . . .

6d Class Interfaces

SV:

Your text goes here . . .

IV Test Plans

1 Features to be tested / not to be tested

SV:

Your text goes here . . .

2 Pass/Fail Criteria

SV:

Your text goes here . . .

3 Approach

SV:

Your text goes here . . .

4 Suspension and resumption

SV:

Your text goes here . . .

5 Testing materials (hardware / software requirements)

SV:

Your text goes here . . .

6 Test cases

SV:

Your text goes here . . .

7 Testing schedule

SV:

Your text goes here . . .

V Project Issues

1 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 . . .

2 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 . . .

2a 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 . . .

2b 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 . . .

2c 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 . . .

3 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.

3a 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 . . .

3b 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 . . .

3c 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 . . .

3d 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 . . .

3e 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 . . .

4 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.

4a 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 . . .

4b 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 . . .

5 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 . . .

6 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 . . .

7 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 . . .

8 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 . . .

9 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 . . .

VI 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 . . .

VII 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.

VIII 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. }