**Software Engineering**

**Graduation Project-I Report**



Covid-19 Detection Using Chest X-ray

**Consultant: Assist. Prof. Dr. ELNAZ PASHAEI**

**Prepared by**

**Emre GÖL**

**Kaan Berke UĞURLAR**

**December 202****0**

**Table of Contents**

|  |  |  |  |
| --- | --- | --- | --- |
|  | How to Use This Document ............................................................................................. | | 2 |
|  | List of Figures ................................................................................................................... | | 9 |
|  | List of Tables .................................................................................................................. | | 10 |
| I | Project Description ......................................................................................................... | | 12 |
| 1 | Project Overview ............................................................................................................ | | 12 |
| 2 | The Purpose of the Project ............................................................................................. | | 12 |
|  | 2a | The User Business or Background of the Project Effort......................................... | 12 |
|  | 2b | Goals of the Project ............................................................................................. | 12 |
|  | 2c | Measurement........................................................................................................... | 13 |
| 3 | The Scope of the Work ................................................................................................... | | 13 |
|  | 3a | The Current Situation .............................................................................................. | 13 |
|  | 3b | The Context of the Work ..................................................................................... | 14 |
|  | 3c | Work Partitioning.................................................................................................... | 16 |
|  | 3d | Competing Products ............................................................................................ | 18 |
| 4 | Product Scenarios ........................................................................................................... | | 18 |
|  | 4a | Product Scenario List ............................................................................................. | 18 |
|  | 4b | Individual Product Scenarios .............................................................................. | 18 |
| 5 | Stakeholders ................................................................................................................... | | 19 |
|  | 5a | The Client................................................................................................................ | 19 |
|  | 5b | The Customer ...................................................................................................... | 19 |
|  | 5c | Hands-On Users of the Product .............................................................................. | 20 |
|  | 5d | Priorities Assigned to Users ................................................................................ | 21 |
|  | 5e | User Participation.................................................................................................... | 21 |
|  | 5f | Maintenance Users and Service Technicians .......................................................... | 22 |
|  | 5g | Other Stakeholders .............................................................................................. | 22 |
| 6 | Mandated Constraints ..................................................................................................... | | 23 |
|  | 6a | Solution Constraints ................................................................................................ | 23 |
|  | 6b | Implementation Environment of the Current System ......................................... | 24 |
|  | 6c | Partner or Collaborative Applications .................................................................... | 25 |
|  | 6d | Off-the-Shelf Software ........................................................................................ | 26 |
|  | 6e | Anticipated Workplace Environment ..................................................................... | 27 |
|  | 6f | Schedule Constraints ............................................................................................... | 27 |
|  | 6g | Budget Constraints .............................................................................................. | 28 |
| 7 | Naming Conventions and Definitions ............................................................................ | | 29 |
|  | 7a | Definitions of Key Terms ....................................................................................... | 29 |
|  | 7b | UML and Other Notation Used in This Document ............................................. | 30 |
|  |  | 4 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 8 | Relevant Facts and Assumptions .................................................................................... | | 31 |
|  | 8a | Facts ........................................................................................................................ | 31 |
|  | 8b | Assumptions ........................................................................................................ | 31 |
| II | Requirements .................................................................................................................. | | 33 |
| 9 | Product Use Cases .......................................................................................................... | | 33 |
|  | 9a | Use Case Diagrams ................................................................................................. | 33 |
|  | 9b | Product Use Case List ......................................................................................... | 34 |
|  | 9c Individual Product Use Cases ................................................................................. | | 35 |
| 10 | Functional Requirements ................................................................................................ | | 35 |
| 11 | Data Requirements ......................................................................................................... | | 36 |
| 12 | Performance Requirements ............................................................................................ | | 38 |
|  | 12a | Speed and Latency Requirements ....................................................................... | 38 |
|  | 12b | Precision or Accuracy Requirements .................................................................. | 39 |
|  | 12c | Capacity Requirements ....................................................................................... | 39 |
| 13 | Dependability Requirements .......................................................................................... | | 40 |
|  | 13a | Reliability Requirements ..................................................................................... | 40 |
|  | 13b | Availability Requirements ................................................................................... | 40 |
|  | 13c | Robustness or Fault-Tolerance Requirements ..................................................... | 41 |
|  | 13d | Safety-Critical Requirements .............................................................................. | 42 |
| 14 | Maintainability and Supportability Requirements .......................................................... | | 43 |
|  | 14a | Maintenance Requirements ................................................................................. | 43 |
|  | 14b | Supportability Requirements ............................................................................... | 43 |
|  | 14c | Adaptability Requirements .................................................................................. | 44 |
|  | 14d | Scalability or Extensibility Requirements ........................................................... | 44 |
|  | 14e | Longevity Requirements ..................................................................................... | 45 |
| 15 | Security Requirements .................................................................................................... | | 45 |
|  | 15a | Access Requirements .......................................................................................... | 45 |
|  | 15b | Integrity Requirements ........................................................................................ | 46 |
|  | 15c | Privacy Requirements ......................................................................................... | 47 |
|  | 15d | Audit Requirements ............................................................................................. | 48 |
|  | 15e | Immunity Requirements ...................................................................................... | 48 |
| 16 | Usability and Humanity Requirements .......................................................................... | | 48 |
|  | 16a | Ease of Use Requirements ................................................................................... | 48 |
|  | 16b | Personalization and Internationalization Requirements ...................................... | 50 |
|  | 16c | Learning Requirements ....................................................................................... | 51 |
|  |  | 5 |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | 16d | Understandability and Politeness Requirements ................................................. | 52 |
|  | 16e | Accessibility Requirements ................................................................................. | 52 |
|  | 16f | User Documentation Requirements .................................................................... | 53 |
|  | 16g | Training Requirements ........................................................................................ | 54 |
| 17 | Look and Feel Requirements .......................................................................................... | | 54 |
|  | 17a | Appearance Requirements ................................................................................... | 54 |
|  | 17b | Style Requirements ............................................................................................. | 55 |
| 18 | Operational and Environmental Requirements .............................................................. | | 56 |
|  | 18a | Expected Physical Environment .......................................................................... | 56 |
|  | 18b | Requirements for Interfacing with Adjacent Systems ......................................... | 56 |
|  | 18c | Productization Requirements .............................................................................. | 57 |
|  | 18d | Release Requirements ......................................................................................... | 58 |
| 19 | Cultural and Political Requirements ............................................................................... | | 58 |
|  | 19a | Cultural Requirements......................................................................................... | 58 |
|  | 19b | Political Requirements ........................................................................................ | 59 |
| 20 | Legal Requirements ........................................................................................................ | | 60 |
|  | 20a | Compliance Requirements .................................................................................. | 60 |
|  | 20b | Standards Requirements ...................................................................................... | 61 |
| III | Design............................................................................................................................. | | 61 |
| 21 | System Design ................................................................................................................ | | 61 |
|  | 21a | Design goals ........................................................................................................ | 61 |
| 22 | Current Software Architecture ....................................................................................... | | 63 |
| 23 | Proposed Software Architecture ..................................................................................... | | 63 |
|  | 23a | Overview ............................................................................................................. | 63 |
|  | 23b | Class Diagrams .................................................................................................... | 63 |
|  | 23c | Dynamic Model ................................................................................................... | 63 |
|  | 23d | Subsystem Decomposition .................................................................................. | 63 |
|  | 23e | Hardware / software mapping ............................................................................. | 64 |
|  | 23f | Data Dictionary ................................................................................................... | 64 |
|  | 23g | Persistent Data management ............................................................................... | 64 |
|  | 23h | Access control and security ................................................................................. | 64 |
|  | 23i | Global software control ....................................................................................... | 64 |
|  | 23j | Boundary conditions ........................................................................................... | 65 |

|  |  |  |
| --- | --- | --- |
| 24 | Subsystem services | ......................................................................................................... 65 |
| 25 | User Interface ................................................................................................................. | 65 |
|  |  | 6 |

|  |  |  |  |
| --- | --- | --- | --- |
| 26 | Object Design ................................................................................................................. | | 65 |
|  | 26a | Object Design trade-offs ..................................................................................... | 65 |
|  | 26b | Interface Documentation guidelines .................................................................... | 65 |
|  | 26c | Packages .............................................................................................................. | 66 |
|  | 26d | Class Interfaces ................................................................................................... | 66 |
| IV | Test Plans ........................................................................................................................ | | 66 |
| 27 | Features to be tested / not to be tested ............................................................................ | | 66 |
| 28 | Pass/Fail Criteria ............................................................................................................ | | 66 |
| 29 | Approach ........................................................................................................................ | | 66 |
| 30 | Suspension and resumption ............................................................................................ | | 67 |
| 31 | Testing materials ( hardware / software requirements ) ................................................. | | 67 |
| 32 | Test cases ........................................................................................................................ | | 67 |
| 33 | Testing schedule ............................................................................................................. | | 67 |
| V | Project Issues .................................................................................................................. | | 67 |
| 34 | Open Issues ..................................................................................................................... | | 67 |
| 35 | Off-the-Shelf Solutions .................................................................................................. | | 68 |
|  | 35a | Ready-Made Products ......................................................................................... | 68 |
|  | 35b | Reusable Components ......................................................................................... | 69 |
|  | 35c | Products That Can Be Copied ............................................................................. | 69 |
| 36 | New Problems ................................................................................................................ | | 69 |
|  | 36a | Effects on the Current Environment .................................................................... | 69 |
|  | 36b | Effects on the Installed Systems.......................................................................... | 70 |
|  | 36c | Potential User Problems ...................................................................................... | 70 |
|  | 36d | Limitations in the Anticipated Implementation Environment That May Inhibit | |
|  | the New Product ............................................................................................................. | | 70 |
|  | 36e | Follow-Up Problems ........................................................................................... | 71 |
| 37 | Tasks | ............................................................................................................................... | 71 |
|  | 37a | Project Planning .................................................................................................. | 71 |
|  | 37b | Planning of the Development Phases .................................................................. | 72 |
| 38 | Migration to the New Product ........................................................................................ | | 73 |
|  | 38a | Requirements for Migration to the New Product ................................................ | 73 |
|  | 38b | Data That Has to Be Modified or Translated for the New System ..................... | 73 |
|  |  | 7 |  |

39 Risks 74

40 Costs 75

41 Waiting Room 76

42 Ideas for Solutions 76

43 Project Retrospective 77

VI Glossary 77

VII References / Bibliography 78

VIII Index 78

8

**Project Description**

**1 Project Overview**

A brief description of the product to be produced, before getting into details.

The COVID-19 pandemic, also known as the coronavirus pandemic, is an ongoing pandemic that is fatal. As of 9 November 2020, more than 50.4 million cases have been confirmed with more than 1.25 million deaths attributed to COVID-19, and more than 32.8 million recovered. As it is known, early diagnosis is vital in any of the diseases. In such cases as COVID-19, early diagnosis can save millions of lives. If a patient is diagnosed, then he can be quarantined and prevented from spreading by the authority. This is the place where our project steps in. The project can diagnose COVID-19 by only using x-ray images of a potential patient which helps to protect millions.

**RELATED WORK WILL BE HERE!**

1. **The Purpose of the Project**

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

The project has been planned over 3 (three) months and contributed by professionals in this area (i.e., doctors and professors). Subsequently, it was decided on the neural network architecture — which was the best among various of them. In those days, there were some nonsensical apps, that want you to blow through your microphone to detect whether you are infected or not, on the market. People who have created those apps were mocking with users and it was obvious that something must be done to prevent people from believing that absurdity. There was a clear need for a system that gives a genuine result whether you are infected or not. In the beginning, users will be able to upload X-ray images of their chests and get the results as soon as possible. There is going to be a variety of improvements in the oncoming versions not only about speed but also on the accuracy and images part.

**2b Goals of the Project**

The goal of this project is to use Artificial Intelligence, especially a deep learning-based approach, to detect COVID-19 infection from chest X-ray images. In addition, it is to verify and test the convolutional neural network for classification. The diagnosis of COVID-19 is typically associated with both symptoms of pneumonia (Pneumonia is an infection that inflames the air sacs in one or both lungs) and Chest X-ray tests. Chest X-ray is the first imaging technique that plays an important role in the diagnosis of COVID-19 disease. Another goal is to accurately identify at least 93% of COVID-19.

**2c Measurement**

To test the goal of the project, we need to calculate the accuracy of COVID-19 test. So, what is accuracy? The accuracy of a machine learning classification algorithm is one way to measure how often the algorithm classifies a data point correctly. Accuracy is the number of correctly predicted data points out of all the data points. How we calculate accuracy is:

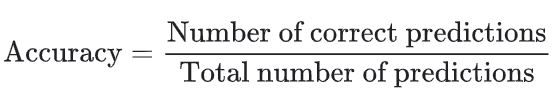


Figure 2.1: Accuracy Calculation Method

The accuracy can be defined as the percentage of correctly classified instances Where TP, FN, FP and TN represent the number of true positives, false negatives, false positives and true negatives, respectively. For good classifiers, TPR and TNR both should be nearer to 100%.



Figure 2.2: Accuracy Calculation by Confusion Matrix

In accuracy, the actual value is the value that is obtained by observation or by measuring the available data. It is also called the observed value. The predicted value is the value of the variable predicted based on the regression analysis.

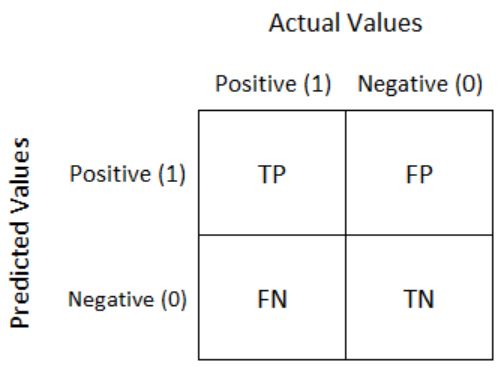


Figure 2.3: Confusion Matrix

Now, we know the accuracy. In our project we are going to identify at least %93 of the COVID-19 correctly. In there, we calculate at least %93 using the accuracy of the dataset.

There are some other metrics to evaluate measurement. Such as, we already learn what true positive, true negative, false negative, false positive is. Let’s learn what recall and precision is. Precision measures the number of positive class predictions that belong to the positive class. Recall measures positive class predictions created by all positive samples in the dataset.

Text, letter

Description automatically generated

Figure 2.4: Recall and Precision Calculation by Confusion Matrix

Accuracy interested in classified observations both positive and negative sides. What if we only check the positive side. For that, we may want to use F1-score. F1-score is a measure of a test’s accuracy. Unlike accuracy, F1-score only measures precision and recall on the positive class.

A picture containing text

Description automatically generated

Figure 2.5: F1-Score Calculation Method

There are some metrics generally used for medical areas. These are sensitivity and specificity and they are quite popular metrics. Sensitivity is the ability of a test to accurately identify patients with a disease. The ability of a test to accurately identify people who do not have the disease.

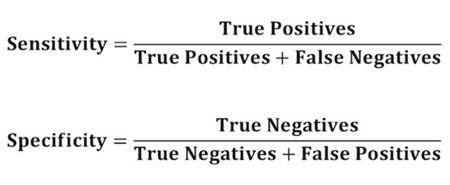


Figure 2.6: Sensitivity and Specificity Calculation by Confusion Matrix

If we want to know the relationship between sensitivity and specificity what will we use? For this, we can use the ROC curve. Then, what is the ROC curve? ROC curve is a graph that shows performance of a classification.

Diagram

Description automatically generated

Figure 2.7: Sensitivity Over Specificity Plot

In a ROC curve the true positive rate which is sensitivity is plotted in function of the false positive rate which is specificity for different cut-off points of a parameter. Each point on the ROC curve represents a sensitivity and specificity pair corresponding to a particular decision threshold.

We also may want a quick summary of the ROC curve. Let’s use AUC for that. AUC means Area Under the Curve which measures the two-dimensional area under the ROC curve.

Chart

Description automatically generated

Figure 2.8: ROC and AOC Curves

**3 The Scope of the Work**

Scope of Work is the field in a contract where the work to be done is explained. The Scope of Work should include all milestones, reports, deliverables and final products expected to be achieved by the performing party. The Scope of Work should also include a timetable for all deliverables. The Scope of Business also defines the business environment in which the product will be used.

**3a The Current Situation**

The current situation of COVID-19 is affecting almost all countries around the world.

New cases and deaths of people due to COVID-19 are increasing. Some countries and territories deal with the second wave of COVID-19. Even some of them are dealing with the third wave. Many countries try to find the vaccine and some of them succeed.

The current situation is dealing requirements before the implementation part. We have been dealing with the planning, analyzing, and designing parts of the project very carefully and successfully.

**3b The Context of the Work**

Work context refers to concepts that can be used to describe the specific context of different works in the same profession. The working context can describe, for example, a workplace, types of companies, environmental conditions, products, technologies or business activities.

This product will be used by doctors, nurses and health care workers. So, this product will be used in hospitals. Hospital is an institution that is equipped for diagnosing and curing the disease both medical and surgical. Hospitals are the main place for detection and cure of COVID-19. Types of hospitals are an important part of health care. Some of the hospitals don’t even have X-ray devices for detection of disease. For those hospital’s options of detection of COVID-19 are PCR and rapid diagnostic tests. But if hospitals have x-ray devices, we recommend them to use it for diagnosing of COVID-19 because it is fast and effective.

Our product is highly recommended for detection of the disease. It is so simple to understand the system for use. Any person who has the X-ray film upload the film, wait for the process and the result will be printed on the screen. This workplace that runs the film is a computer or an android device. The requirement of the devices is internet connection.

**3c Work Partitioning**

The event list includes the following elements:

●      Uploading chest x-ray as an image

●      Probability of having the disease

●      Users (doctor, nurse or health worker) upload a chest x-ray as an image and wait for the process. After that, the program will print the result.

Using our program is very effective. You can see the result very fast. It is faster than other diagnosing ways like PCR and rapid diagnostic tests. People who take PCR or other tests will get a result in 48 hours. Imagine how much the disease will change if we take the results in one hour. People who take other tests will be still working or interacting with other people so it will increase the disease transmission. If they could get chest X-ray results, they could go directly in quarantine and that would decrease so much the number of cases and deaths.

**3d Competing Products**

To compare the product with other alternatives that already exist, it must be understood well. First, it needs to be comprehended that what do the other products on the market do exactly. Secondly, what can be added extra to the other products to create a brand-new one. However, it was discussed with all the developers that the product must be faster and better. On the other hand, if the product cannot be used by anyone on the face of the planet, then it is nothing but a waste. A light bulb went on in every individual's head in the team simultaneously. These bulbs led the project to a broader area.

The product has now two different kinds of areas to serve. Anyone, who is mature enough to know how to use a smartphone, can use the product both by installing its mobile application and uploading her/his chest x-ray image easily or by clicking its website and again following the same uploading procedure. Either way is elementary level and sufficient for a complex project like this. Any other project, which has the same goal with this project, does not have any place to serve neither their technology or results, yet this project has bested all over its opponents based on accuracy.

**4 Product Scenarios**

Scenarios are somewhat informal stories describing how the end users would use the product once it is completed. They take the form of narratives and may involve specific individuals and examples.

**4a Product Scenario List**

The product scenario list is quite simply a list of the product scenarios that will appear in the next section. It is a good idea to either number or name each scenario for later reference, and it can also be a good idea to organize the list so that related scenarios appear together. (Depending on the naming / numbering scheme, they can be grouped into sections and subsections, etc. )

**4b Individual Product Scenarios**

Product scenarios are written in a natural narrative fashion, easily understood by clients and other non-technical stakeholders. Each one tells a story of how the end users are expected to eventually use the finished product. For example:

In both cases image(s) should not be any blurred or glared or as little as possible to prevent to detect wrongly.

**Upload Images:** As soon as Mary got her chest’s X-ray image(s), she now can either take a photo of the image(s) by her phone or scan them with help of a scanner. Thereafter, she uploads the image(s) into our system to make it predict through the product’s android app or website. As she sees the uploading process completed successfully, then she needs to wait 2 hours to get the results. Even if there is a slight evidence of Covid-19, then the product notifies the patient, who is Mary in this case, as she is infected, and she needs to see a doctor or call the hospital to have them checked on her. On the other hand, if the product says that she is not infected, she will be seeing another notification that says she is healthy but still she needs to protect herself.

1. **Stakeholders**

**5a The Client**

Clients will be private hospitals or clinics, mostly. According to the accuracy rate we offered, some of the head doctors of the relevant hospitals were thrilled. We are not looking for many clients thanks to the alpha version of the product which will be released by us, anytime soon. Even though clients provide money and space to build a product, we want to release it for benefit of humanity. Also, Clients, who supported us in the first place, will have some special access through the product, undoubtedly.

**5b The Customer**

The customers are anyone who has internet access. They will open either the internet page of the product or the android app and upload their chest x-ray image (s). If uploading would be finished successfully, the system will show a result to the customer in 2 hours.

In spite of the product will be released as free, a system that involves anything we need working properly will also have some particular needs and to meet them, we need to earn as much as money the product needs. We will not show ads to prevent irritating the customers, but we determined to ask for money, which is an insignificant amount, to show results as soon as possible.

**5d Priorities Assigned to Users**

* Key users: They are critical to the continued success of the product. Give greater importance to requirements generated by this category of user. In this case the key users will be the doctors who can help to continue the success of the product by interpreting the results, suggesting new techniques on improving the speed of detection or at least leaving a comment about UI/UX designs in both app and website.
* Secondary users: They will use the product, but their opinion of it has no effect on its long-term success. Where there is a conflict between secondary users’ requirements and those of key users, the key users take precedence. Secondary users would be any adult who could use the product appropriately. They may not add things to the product, but the developers can deduce things by interpreting the behaviors of the secondary users.
* Unimportant users: This category of user is given the lowest priority. It includes infrequent, unauthorized, and unskilled users, as well as people who misuse the product. If the subject is medical, then no user is unimportant but, in this case, there is an exception. The exception is children. They might not know how to use the product and upload some irrelevant images. Which makes children have the lowest priority.

**5e User Participation**

The most critical part of the product’s improvement is user participation. Once the alpha version of the product is released, so many people will be willing to use the product, but it will need some improvements, eventually. Even though ordinary users would not be able to contribute, their logs are going to be inspected and there will be enhancements on the product if needed.

Furthermore, the clients’ participation means a lot more than regular users. Firstly, they will have the products way before the regular user has it. Secondly, they will provide some high-quality feedback due to avoid spending money for nothing. Last but not least, they have a medical background, and this is such a marvelous thing to build a medical product because the clients can be the director for the functionality of the product.

**5f Maintenance Users and Service Technicians**

Maintenance users are a special type of hands-on users who have requirements that are specific to maintaining and changing the product. In the project, maintenance users will be the clients who are doctors, especially, and employees in the hospital. They have every right to criticize and maintain the product.

**6 Mandated Constraints**

This section describes constraints on the eventual design of the product. They are the same as other requirements except that constraints are mandated, usually at the beginning of the project. Constraints have a description, rationale, and fit criterion, and generally are written in the same format as functional and nonfunctional requirements.

**6a Solution Constraints**

This specifies constraints on the way that the problem must be solved. Describe the mandated technology or solution.

Description:

In this paper, a convolutional neural network (CNN) is used to classify the COVID-19-infected patients as infected (+ve) or not (−ve).

Rationale:

COVID-19 patients must be detected as soon as possible. A model which is created by plain CNN is not satisfying. Accuracy must be higher whilst prediction does not take too much time.

Fit criterion:

The initial parameters of CNN are tuned using multi-objective differential evolution (MODE). Extensive experiments are performed by considering the proposed and the competitive machine learning techniques on the chest CT images. Extensive analysis shows that the proposed model can classify the chest CT images at a good accuracy rate.

Description:

The product shall accept X-ray chest images.

Rationale:

Users want to upload input images easily

Fit criterion:

Both the mobile app and websites should be understood by users at first sight. Apart from the sophisticated artificial intelligence model, that is used to predict, UI/IX design will be minimal and even a 7-year-old would use it although the audience is not her/him.

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

This describes the technological and physical environment in which the product is to be installed

By the time it was decided to build the product, all the technological and physical environments, which the product is to be installed, were determined. For building the product, all the newest and the stabilized of the state-of-art technologies are used. Thanks to the best tools in the market, developing the product, running it on the server and the other users to use it were eased.

To begin with, the deep learning model was created and trained on Tensorflow’s Keras API by using public datasets, and thanks to features of the API, it was feasible to both saving and loading the model that was trained. Moreover, training was half of the project because if you cannot serve what you have done, then it can be said that you have done nothing, basically. Thanks to TensorflowLite, the model’s size decreased but the accuracy of the model was not affected. Then, the lite model was transferred to the Android project to use it on the Android app. Whilst the Android app’s outputs have been working quite well and sufficient, then the idea of using the Lite model on the website is become sensible due to the normal model’s heaviness. Finally, the Tensorflow model has been switched to the Lite model in order to reduce heaviness and increase the prediction time in the backend of the website. In the backend, Python programming language was used due to working with Tensorflow would be easier with the native language the model has been trained and as Python’s Django Rest API framework was used as a backend framework because of it is easy to read, write and relatively fast.

**6c Partner or Collaborative Applications**

AFAIK WE DO NOT USE ANY 3RD PARTY APPLICATION, YET.

Content

This describes applications that are not part of the product but with which the product will collaborate. They can be external applications, commercial packages, or preexisting in-house applications.

Motivation

To provide information about design constraints caused by using partner applications. By describing or modeling these partner applications, you discover and highlight potential problems of integration.

Examples

This section can be completed by including written descriptions, models, or references to other specifications. The descriptions must include a full specification of all interfaces that have an effect on the product.

Considerations

Examine the work context model to determine whether any of the adjacent systems should be treated as partner applications. It might also be necessary to examine some of the details of the work to discover relevant partner applications.

**6d Off­the­Shelf Software**

Initially, it needs to be clarified that what is “off-the-shelf” software. If the software needs to be specially configured to match the customer’s needs, then the software is not “off-the-shelf”.

The product, which has been describing in the report, is off-the-shelf because a client won’t have to add any configuration to it. The client or the customer is going to upload her/his chest X-ray image, then the product will assign a number to it to avoid confusion. This is all the user will do.

The product does not get any off-the-shelf application to run, although it is off-the-shelf. The product was created by configuring all the hyperparameters and adding more to the trained models. Needless to say, the website and the app created from scratch.

**6e Anticipated Workplace Environment**

There is not anticipated workplace environment for using the product, but a proper chest x-ray image must be uploaded into the system not to obtain an error or, worse, a misleading result, such as false negative. In that case, the result could be catastrophic. There will be precaution appeared on the screen once the user open product’s user interface.

The chest x-ray is one of the foremost common imaging tests performed in clinical practice, generally for cough, shortness of breath, chest pain, chest wall trauma, and assessment for the occult disease. normal x-rays are performed with the patient standing facing an X-ray film or digital cassette, 6 feet far from an x-ray tube. The tube fires x-rays thru the patient from the lower back to front, i.e., posterior to anterior (PA). This reduces the magnification of the center and different anterior mediastinal structures that are placed near to the film within the PA position. different factors to contemplate for a decent quality chest x-ray are centering (the trachea ought to be equal between the clavicular heads), penetration (the spine should be simply clear through the internal organ density), and breath effort (at the full inspiratory effort, the anterior finish of the correct sixth rib should purpose mid-way on the right hemidiaphragm).

**6f Schedule Constraints**

Schedule constraints are any known deadlines or windows of opportunity. It is crucial to identify critical times and dates that have an effect on product requirements.

The vast majority of the medical sector has been looking for a product like this for over 12 months. The product’s marketing opportunity is enormous, and investors will trust the project blindfolded. Not to exaggerate, the project gives power to its developers to select the investors, but the product must be ready —or at least an alpha version is released— for letting developers have this kind of authority.

The schedule has been varied several times due to technology and procedure issues. Nonetheless, the final schedule was decided eventually. The project must be ready at the end of January, uttermost. The beginning of February is not even a matter of discussion. Besides the client and the many customers are waiting for the product and the financial impact will be huge if the product is late even a day, the whole human race needs a product like this. That is why the product will be ready on the 15th of January and even if the project has been developing by testing, it will take another 15 days to make sure that the product is almost perfect to serve the human race.

**6g Budget Constraints**

The financial budget for the project is so low that a student can create this project in her/his room, but the real budget for the project is time. Every part of the project wants more time than any others in the market.

The time budget is very narrow, and this steers the planners to either hire more developers/mentors or buy/rent much more powerful servers to train models by using brute force instead of training wisely. These days, servers are server rents are lower than mentors', then the team has decided to rent a powerful server to train the model by using the cross-validation method which was surprisingly satisfactory.

To answer the following question “Is it realistic to build a product within this budget?”, it totally is. In the first place, the project was handled by two students who have an average budget. Later on, a few hospitals wanted to invest in the product after the alpha version has been released. If the investments come true, then the product will rival the companies, globally.

1. **Naming Conventions and Definitions**

**7a Definitions of Key Terms**

All Terms, including acronyms and abbreviations, used in the project is defined at some point. The most important ones are listed here.

Names are very important. They invoke meanings that, if carefully defined, can save hours of explanations. Attention to names at this stage of the project helps to highlight misunderstandings. The glossary produced during requirements is used and extended throughout the project.

SARS: Severe acute respiratory syndrome is a viral respiratory disease caused by a SARS-associated coronavirus.

ROC: Receiver operating characteristic curve is a graph showing the performance of a classification model at all classification thresholds.

AUC: Area under the ROC curve measures the entire two-dimensional area underneath the entire ROC curve form (0, 0) to (1, 1).

API: An application programming interface is a computing interface that defines interactions between multiple software intermediaries.

LOC: Source lines of code (SLOC), also known as lines of code (LOC), is a software metric used to measure the size of a computer program by counting the number of lines in the text of the program's source code.

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

This document generally follows the Version 2.0 OMG UML standard, as described by Fowler in [4]. Any exceptions are noted where used.

1. **Relevant Facts and Assumptions**

**8a Facts**

* Implementation will be written on platforms such as pycharm for neural network and modelling, vscode for web development, android studio for android development.
* Implementation will be written in several programming languages such as python, java, javascript.
* When a user clicks the upload button he/she  can choose x-ray as an image and that image must be successfully uploaded to the system.
* When a user clicks the application. The application must open without any error.
* The system must print whether the person has the virus or not.
* The existing application is 2000 LOC.

**8b Assumptions**

* Identify at least %95 of the covid-19 correctly
* This project will help the situation of pandemic.
* The application will get around five hundred requests a day.

Examples

Assumptions about new laws or political decisions.

Assumptions about what your developers expect to be ready in time for them to use— for example, other parts of your products, the completion of other projects, software tools, or software components.

Assumptions about the technological environment in which the product will operate.

These assumptions should highlight areas of expected compatibility. The software components that will be available to the developers. Other products being developed at the same time as this one. The availability and capability of bought-in components. Dependencies on computer systems or people external to this project the requirements that will specifically *not* be carried out by the product. Considerations

We often make unconscious assumptions. It is necessary to talk to the members of the project team to discover any unconscious assumptions that they have made. Ask stakeholders (both technical and business-related) questions such as these:

* What software tools are you expecting to be available?
* Will there be any new software products?
* Are you expecting to use a current product in a new way?
* Are there any business changes you are assuming we will be able to deal with?

It is important to state these assumptions up front. You might also consider the probability of whether the assumption is correct and, where relevant, a list of alternatives if something that is assumed does not happen.

The assumptions are intended to be transient. That is, they should all be cleared by the time the specification is released—the assumption should have become either a requirement or a constraint. For example, if the assumption related to the capability of a product that is intended to be a partner product to yours, then the capability should have been proven satisfactory, and it becomes a constraint to use it. Conversely, if the bought-in product is not suitable, then it becomes a requirement for the project team to construct the needed capability.

32

1. **Requirements**

**9 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.

**9a 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).

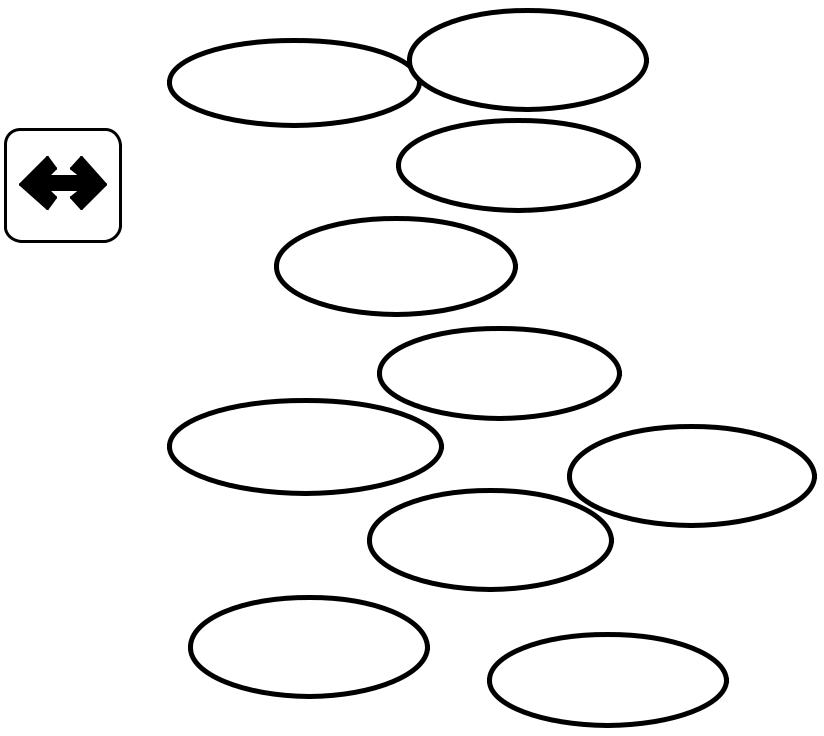
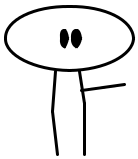
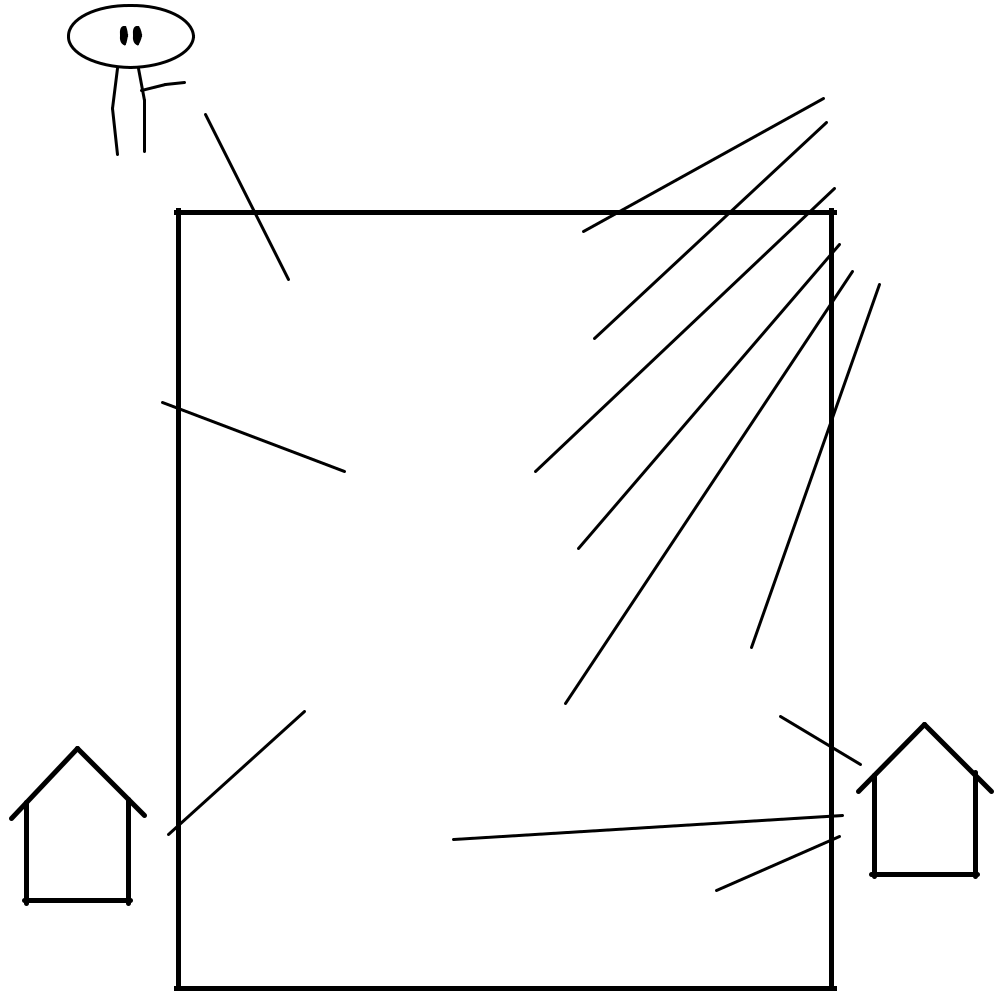
The product use cases are the ellipses inside the boundary. The lines denote usage.

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.

33

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Example |  |  |  |  |  |
| Highways |  |  |  | Truck Depot |  |
| Department |  | Monitor | | Engineer |  |
| Clerk |  |  |  |
| Update Weather | Untreated | |  |  |
|  |  |  |
|  | Forecast | Roads | |  |  |
|  |  | Record Treated | |  |  |
|  |  | Roads |  |  |  |
| Thermal | Produce | |  |  |  |
| De-icing | |  |  |  |
| Mapping | Schedule | |  |  |  |
| Database |  | Record Truck | |  |  |
|  |  |  |  |
|  | Record Weather | Changes |  |  |  |
|  |  | Identify Faulty | |  |
|  | Sta tion Readings |  |  |
|  |  | Amend | Weather Sta | tion |  |
|  |  |  |  |  |
|  |  | De-icing |  |  |  |
|  |  | Schedule |  |  |  |
|  | Record |  |  |  |  |
|  | New Weather |  | Record |  |  |
|  | Sta tion |  |  |  |
|  |  | Road | Road |  |
| Weather |  |  |  |
|  |  |  | Engineering |  |
| Sta tion |  |  |  |  |
|  |  |  | Computer |  |
|  |  |  |  |  |



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.

**9b 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.

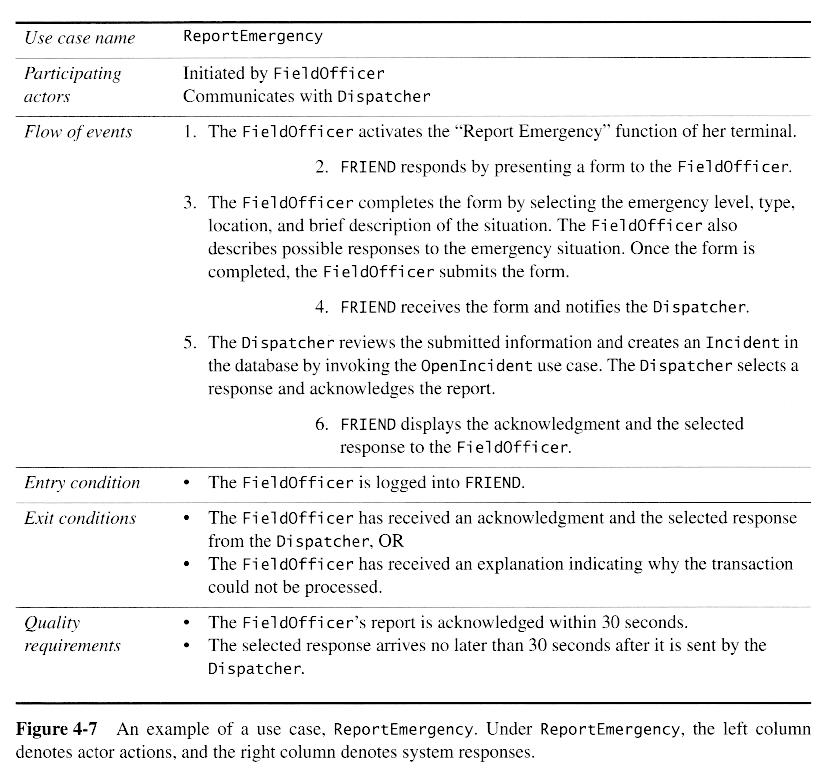
34

**9c 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:



**10 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.

35

Motivation

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

Examples

Requirement #: 75 Requirement Type: 9 Event/use case #: 7, 9

Description: The product s hall re cord a ll t he roads t hat h ave been t reated

Rationale: To be able to schedule unt reated roads and highlight

pot ent ial danger

Originator: Arnold Snow - Chief Engineer

Fit Criterion: The recorded treat ed and unt reated roads shall agree wit h

t he drivers’road treat ment logs.

Customer S atisfaction: 3 Customer Dissatisfaction: 5

Priority: Conflicts:

|  |  |  |  |
| --- | --- | --- | --- |
| Suppor ting Materials: | | CopyrightVolere©AtlanticSystemsGuild |  |
| History: | Creat ed Februar y 29,2006 |  |
|  |  |

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.

**11 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

36

object model, or a domain model. Alternatively, these requirements might be described by defining the terms in the dictionary described in section 5.

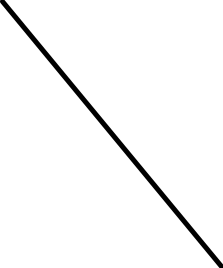
Motivation

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.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| District | |  |  | Road | 1 | S | Road | | S S | Truck | |  |
|  |  | S | S |  | Section | |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  | S |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  | S | |  |  |  |  |  | 1 | S |  |  |
|  |  |  |  |  |  |  |  |  |
|  | S | |  |  |  |  |  | S |  | 1 1 |  |  |
| Forecast | |  |  | Temperatur | e |  | Wea ther | |  | Depot | |  |
|  |  |  |  | Reading | S | 1 | Sta tion | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |



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?

37

**12 Performance Requirements**

**12a 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.

38

**12b 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.

**12c 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.

39

Fit Criterion

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

**13 Dependability Requirements**

**13a 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 doso 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.

**13b 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.

40

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.

**13c Robustness or Fault­Tolerance 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.

41

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.

**13d Safety­Critical 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

42

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.

**14 Maintainability and Supportability Requirements**

**14a 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.

**14b 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.

43

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.

**14c 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.

**14d 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.

44

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.

**14e 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.

**15 Security Requirements**

**15a 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.

45

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.

**15b 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.

46

**15c 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.

47

**15d 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.

**15e 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.

**16 Usability and Humanity Requirements**

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

**16a 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.

48

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.

49

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.

**16b 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.

50

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

**16c 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.

51

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.

**16d 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.

**16e 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.

52

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.

**16f 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

53

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?

**16g 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?

**17 Look and Feel Requirements**

**17a 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.

54

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.

**17b 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.

55

**18 Operational and Environmental Requirements**

**18a 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.

**18b 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.

56

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

**18c 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.

57

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.

**18d 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?

**19 Cultural and Political Requirements**

**19a 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.

58

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?

**19b 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.

59

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.

**20 Legal Requirements**

**20a 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?

60

**20b 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?

1. **Design**

**21 System Design**

**21a 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.

61

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. )

62

Motivation

Considerations

Example

**22 Current Software Architecture**

Content

Motivation

Considerations

Example

**23 Proposed Software Architecture**

**23a Overview**

Content

Motivation

Considerations

Example

**23b Class Diagrams**

Content

Motivation

Considerations

Example

**23c Dynamic Model**

Content

Motivation

Considerations

Example

**23d Subsystem Decomposition**

Content

63

Motivation

Considerations

Example

**23e Hardware / software mapping**

Content

Motivation

Considerations

Example

**23f Data Dictionary**

Content

Motivation

Considerations

Example

**23g Persistent Data management**

Content

Motivation

Considerations

Example

**23h Access control and security**

Content

Motivation

Considerations

Example

**23iGlobal software control**

Content

Motivation

64

Considerations

Example

**23j Boundary conditions**

Content

Motivation

Considerations

Example

**24 Subsystem services**

Content

Motivation

Considerations

Example

**25 User Interface**

Content

Motivation

Considerations

Example

**26 Object Design**

**26a Object Design trade­offs**

Content

Motivation

Considerations

Example

**26b Interface Documentation guidelines**

Content

Motivation

65

Considerations

Example

**26c Packages**

Content

Motivation

Considerations

Example

**26d Class Interfaces**

Content

Motivation

Considerations

Example

**IV Test Plans**

**27 Features to be tested / not to be tested**

Content

Motivation

Considerations

Example

**28 Pass/Fail Criteria**

Content

Motivation

Considerations

Example

1. **Approach**

Content

66

Motivation

Considerations

Example

**30 Suspension and resumption**

Content

Motivation

Considerations

Example

**31 Testing materials ( hardware / software requirements )**

Content

Motivation

Considerations

Example

**32 Test cases**

Content

Motivation

Considerations

Example

**33 Testing schedule**

Content

Motivation

Considerations

Example

1. **Project Issues**

**34 Open Issues**

67

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?

1. **Off­the­Shelf Solutions**

**35a Ready­Made Products**

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.

68

**35b Reusable Components**

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.

**35c Products That Can Be Copied**

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.

**36 New Problems**

**36a Effects on the Current Environment**

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.

69

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.

**36b Effects on the Installed 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.

**36c Potential User Problems**

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.

**36d** **Limitations in the Anticipated Implementation Environment That May Inhibit the New Product**

Content

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

70

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.

**36e Follow­Up Problems**

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.

1. **Tasks**

**37a Project Planning**

Content

Details of the life cycle and approach that will be used to deliver the product. A high-level process diagram showing the tasks and the interfaces between them is a good way to communicate this information.

71

Motivation

To specify the approach that will be taken to deliver the product so that everyone has the same expectations.

Considerations

Depending on the maturity level of your process, the new product will be developed using your standard approach. However, some circumstances are unique to a particular product and will necessitate changes to your life cycle. While these considerations are not product requirements, they are needed if the product is to be successfully developed.

If possible, attach an estimate of the time and resources needed for each task based on the requirements that you have specified. Attach your estimates to the events, use cases, and/or functions that you specified in sections 8 and 9.

Do not forget issues related to data conversion, user training, and cutover. These needs are usually ignored when projects set implementation dates.

**37b Planning of the Development Phases**

Content

Specification of each phase of development and the components in the operating environment.

Motivation

To identify the phases necessary to implement the operating environment for the new system so that the implementation can be managed.

Fit Criterion

Name of the phase.

Required operational date.

Operating environment components included.

Functional requirements included.

Nonfunctional requirements included.

Considerations

Identify which hardware and other devices are necessary for each phase of the new system. This list may not be known at the time of the requirements process, as these devices may be decided at design time.

72

**38 Migration to the New Product**

**38a Requirements for Migration to the New Product**

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.

**38b Data That Has to Be Modified or Translated for the New System**

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.

73

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?

**39 Risks**

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.

74

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.

**40 Costs**

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.

75

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.

**41 Waiting Room**

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.

**42 Ideas for Solutions**

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

76

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.

**43 Project Retrospective**

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.

**VI Glossary**

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 deliberatelyor accidentally lost by its original owners.

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

77

**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.

7- Naming Part.

SARS:

<https://www.who.int/health-topics/severe-acute-respiratory-syndrome>

AUC/ROC:

<https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>

API:

<https://en.wikipedia.org/wiki/API>

LOC:

https://en.wikipedia.org/wiki/Source\_lines\_of\_code

**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. }

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Design ................................................. | 61, | 63 | Test...................................................... | 64, 65 |
| Requirements ................................ | 35, 51, | 58 |  |  |

78