SE 3XA3: Software Requirements Specification
CamRuler

Team 10,
Meet Patel, patelm16
Prince Kowser, kowserm
Kshitij Mehta, mehtak1

December 7, 2017

Contents

List of Tables

List of Figures

1 Revision History

Table 1: **Revision History**

Date	Version	Notes
Oct 6 2017	1.0	Rev0
Oct 26 2017	1.0	Rev0.5
Dec 06 2017	1.0	Rev1

This document describes the requirements for a mobile application named CamRuler. The template for the Software Requirements Specification (SRS) is a subset of the Volere template. There were no further modifications to the template.

2 Project Drivers

2.1 The Purpose of the Project

The purpose of CamRuler is to simplify the process of measuring an object by allowing the user to use their phones camera to measure any object. Having this app on mobile devices will enable users to conveniently measure objects without the struggle of nding a ruler and doing tedious measurements by hand. Our motivation for this project is to increase the efficiency of measuring an object by providing an accurate and easy-to-use platform. This project can be worthwhile for many clients as it can be used in various industries including construction, architecture, product modelling or even something as trivial as a student lab. Using this application can increase productivity and provide a good business opportunity for our clients.

The goal of this application is to make it easier for anyone to measure objects. We will be using Java and Android Studio to build this application. We hope to boost productivity and efficiency for our customers, thus making our goal a service goal.

2.2 The Stakeholders

2.2.1 The Client

The client of this product is the group of developers themselves. The developers are interested in re-deploying the app with improvements that make the app more efficient and user friendly .

2.2.2 The Customers

The customers of this product are specifically students and the general public for daily use. Students often need precise measurements in labs, or in school for specific courses like construction class. This app can eliminate the use of long ruler sticks and uncertainty with measurements for students. The general public also need daily measurements and often times they dont have measuring tools at the tip of their hands. The may need to measure a table, their height, or even something large. This app can come very handy for them as it allows them to measure anything they can fit in a picture.

2.2.3   Other Stakeholders

- Carpenters

- Plumbers

- Technicians who need quick and precise measurements

2.3   Mandated Constraints

2.3.1   Solution Constraints

1. Description: The product shall operate on mobile phones with an Android operating system.

   Rationale: The product will be targeting Android users

   Fit Criterion: The product shall be approved by the testing group of the operating system

2. Description: The product shall use the phones camera to take a picture and measure the object

   Rationale: The client will not use any other devices other than their phone.

   Fit Criterion: Any applications (eg. camera) required to run the product will already be existing in the phone and there shall be no need to install anything else.

3. Description: The product shall allow the user to select the object they would like to measure.

   Rationale: The client should be able to pick the object they would like to measure from the picture that they take.

   Fit Criterion: Any selection of objects will be done by allowing the user to draw a line on the screen where the line will be going across the objects length on the screen.

4. Description: The product shall use a reference object to measure the desired object.

   Rationale: The product cannot measure an objects size unless it has another objects measurements it can use as a reference.

   Fit Criterion: Reference objects will be selected using the same process where the client will be prompted to input the known or estimated dimensions of that object. The product will use those dimensions to create a ratio, and output the dimensions of the desired object.

2.3.2   Implementation Environment of the Current System

The product will require an Android environment along with the necessary storage space required to install the product on the clients phone.

2.3.3   Partner or Collaborative Application

The product will heavily rely on the phones camera. A camera is not part of the product however the product will use the camera application installed on the phone to achieve its functionality.

2.3.4   Off The Shelf

The product will use Java, and Android Studio in order to implement the requirements of the product. All the OTS software listed above is free and can be installed from the Internet.

2.3.5   Anticipated Workplace Environment

The product should be used in an environment with bright lighting so that users can see the objects taken in a picture. The reference object and desired object should be on the same surface in order to get the best measurements. The product can be used at any workplace,whether they be indoor or outdoor.

2.3.6   Schedule Constraints

The product must be completed by December 2017. There is no constraint on windows of opportunity as this product can be used anywhere at anytime.

2.3.7 Budget Constraints

A budget is not applicable to this project. Any required resources are free and can be downloaded from the Internet.

2.3.8 Enterprise Constraints

The product will be free to install and accessible to any user with an Android mobile device. There are no enterprises who will be making an investment in this project other than the developers themselves.

2.4 Naming Conventions and Terminology

Name	Meaning
Compiler	A software utility that converts code written in any programming language into machine language that can be understood by the computer
GUI	Graphical User Interface. An interface that humans can use to interact with computers
OS	Operating System
Git	Git Lab
CamRuler	The name of the application
Product	The mobile application that is being developed
Project	The overall development of the product
Client	The people that this product is being developed for
Customer	The people that will be using this product
API	Application Programming Interface. A set of functions and the purpose of those functions that a client can call in their software
App	Application

Table 2: Dictionary

2.5 Relevant Facts and Assumptions

A relevant fact about this project is that there are approximately 800 lines of code for the original implementation of CamRuler. An assumption that will be made is that the device on which this application will be installed will also have a camera application. It will also be assumed that the user has the required storage space and memory in order to install the application on their device. Another assumption that will be made is that the developers will have the product completed and ready to be used by the deadline.

3 Functional Requirements

3.1 The Scope of the Work and the Product

3.1.1 The Context of the Work
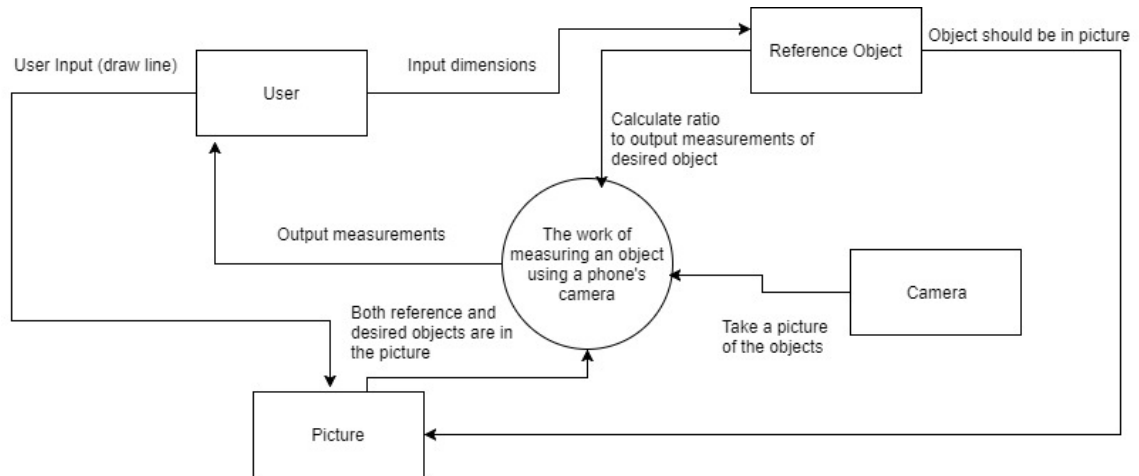
Figure 1: Work Context Diagram

3.1.2 Work Partitioning

Event Name	Input/Output	Summary
User Takes Picture	Picture(IN)	The user takes a picture of the object and the application prepares the pic to be analyzed for length measurement
User selects object and reference object	Lines on the objects(IN)	The user draws on the line on the objects for the application to do calculation
User inputs the reference objects length	Reference Object length (IN)	The user inputs the reference object's length in the picture in cm, meters, or inches in order to calculate actual object's length
Application does calculation	Length(OUT)	The application uses all the information given by the user to output the objects length

Table 3: Work Partitioning

3.1.3 Individual Product Use Cases
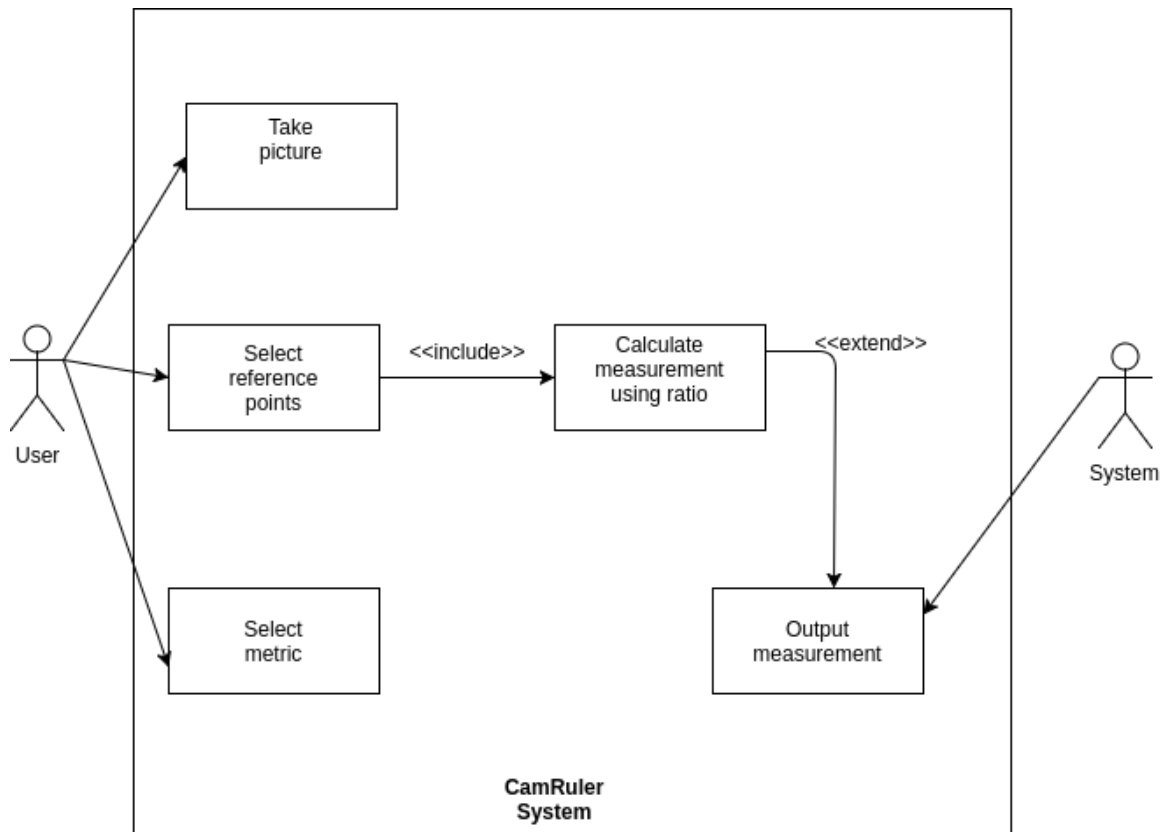
Figure 2: Use Cases Diagram

3.2 Functional Requirements

1. FR1
   Description: The executable android application will open an graphical user interface on android phone.

   Test Case: Check to see if the GUI opens upon executing the application

2. FR2
   Description: The android application displays instructions for the use

of the application in and waits for user input

Test Case: Check if application does what the instructions say

3. FR3
Description: Upon the reception of user input the application opens the phones camera application for the user to take a picture of the reference object(an object with its length known) and the objects to be measured, all together in the same picture. ~~Otherwise if the camera application fails, it allows the user to pick a picture from the phone's gallery~~

Test Case: Check to see if the camera opens up to take a picture

4. FR4
Description: After the picture is taken, it is set to the specified image area in the application. The application then allows the User to draw two dots or four dots, each one at the ends of the reference object on the picture as well as the object to be measured.

Test Case: Check to see if one can draw properly on the picture and if a line is created
Check to see if the picture becomes the background of the application

5. FR5
Description: After the 2nd dot of each object is drawn, a line is shown from one point to the other. The user can no longer draw after the line is shown.

Test Case: Line from 1st dot to 2nd dot after the 2nd dot is drawn

Check to see what happens if drawing is still possible after the line is shown

6. FR6
Description:The user has an option to retake picture and re-select their points on the picture

Test Case: Take two pics from the same state and redraw points after drawn once from the same state

7. FR7
Description: After the lines are drawn, the application opens a dialog box for the user to input the measurements of the reference object

Test Case: Check to see if the dialog box opens
Check to see if the dialog box allows the user to input measurements in different units (cm, m, inches)

8. FR8
Description: After the the two lines are shown, the application calculates the length of the object and displays the length in cm, meters, and inches.

Test Case: test if the calculation is correct my measuring the actual object

9. FR9
Description: When placing the points on the picture, the user is able to drag the points.

Test Case: Check if you can drag each point when placing on picture

4   Non-functional Requirements

4.1   Look and Feel Requirements

4.1.1   Appearance Requirements

NF-AP
The product shall display the logo as the application icon. CamRulers background will also contain the logo until a picture is taken to fill the background. ~~It will have a slick green theme that all the buttons and message windows~~

~~will follow~~ Fit Criterion: Ask 10 people to rate the looks of the app and 75 percent of the people should rate it over 5

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

NF-EU
The product shall make it easy to select objects in the picture and provide a satisfactory experience for any user using this product. Fit Criterion: Ask 10 people to rate the looks of the app and 75 percent of the people should rate it over 5

4.2.2 Understandability and Politeness Requirements

NF-UPR
The product shall use metrics familiar with the user and hide the details of length calculation from the user Fit Criterion: Ask 10 people to rate the metrics familiarity of the app and 75 percent of the people should rate it over 5

4.2.3 Learning Requirements

NF-LR
The user must have some knowledge of operating a phone Fit Criterion: User should have some experience with a phone

4.2.4 Accessibility Requirements

NF-AR
The android application will be executed by different models of android phones Fit Criterion: The android application should run with no errors on 5 different models

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

NF-SLR
The application shall respond almost immediately upon user input and dur-

ing calculation of length. Fit Criterion: The application should respond with in 1 second upon user input and calculation

4.3.2 Safety-Critical Requirements

NF-SCR
The product shall not compromise phone or the picture the user takes Fit Criterion: The product shall not compromise data used from the phone

4.3.3 Precision or Accuracy Requirements

NF-PAR
Output of length will be accurate to 3 decimal places Fit Criterion: The output shall be accurate to 3 decimal places

4.3.4 Reliability and Availability Requirements

NF-RAR
The product shall be available on the users phone once the user downloads the application Fit Criterion: 100 percent of users that download the app shall have the product available on the user's phone

4.3.5 Robustness or Fault-Tolerance Requirement

NF-RFTR
The product shall work without access to internet Fit Criterion: The product shall work 100 percent of the time wihtout access to internet

4.3.6 Capacity Requirements

NF-CR
The product shall be able to measure multiple objects from the taken picture Fit Criterion: The product shall be able to measure 1 or 2 objects from the taken picture

4.3.7 Scalability or Extensibility Requirements

NF-SER
The product shall be capable of measuring multiple objects in one picture

after a few days of launch Fit Criterion:The product shall be able to measure 2 to 3 object after a fewdays of launch

4.3.8 Longevity Requirements

NF-LR
The product shall be relevant for the lifetime of android phones Fit Criterion: The product shall be relevant for 20years

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

NF-EPE
The product shall be usable in an environment where a phone is acceptable Fit Criterion: The product shall be usable indoors and outdoors in a non-wet environment

4.4.2 Requirements for Interfacing with Adjacent Systems

NF-RIA
The production shall work on recent versions of android Fit Criterion: The product shall work on android 4.0 and up

4.4.3 Productization Requirements

NF-PR
The product shall be distributed as an android application Fit Criterion: The product shall be able to distribute on a phone running an android operating system

4.4.4 Release Requirements

NF-RR
The product will undergo maintenance upon realization of any errors in calculation or user interface Fit Criterion: The product will undergo maintentance after realizing 1 or more errors

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirement

NF-MR
Minimal product maintenance Fit Criterion: The product shall require maintenance once a year

4.5.2 Supportability Requirements

NF-SR
Ensure that most android users can use the app Fit Criterion: Ask 10 people if they can use the app and 90 percent of the people should be able to use it

4.5.3 Adaptability Requirement

NF-AR
The app shall be universally accessible by users Fit Criterion: The app should be accessible in all counties with access to internet

4.6 Security Requirements

4.6.1 Access Requirement

NF-SAR
The entire general public shall be able to access the game assuming they have internet to download the app Fit Criterion: 100 percent of people with access to the internet shall be able to download the app

4.6.2 Integrity Requirement

NF-SIR
The app shall not allow the user to put invalid input Fit Criterion: The app should allow 0 invalid input

4.6.3 Privacy Requirement

NF-SPR
The product shall not save any picture the user takes Fit Criterion: The product shall save 0 picture the user takes

4.7 Cultural Requirements

NF-SCR

The product shall offer output in multiple types of metric unit Fit Criterion: The product shall offer output in 4 different metric types

4.8 Legal Requirements

4.8.1 Compliance Requirements

NF-LCR

The app shall not compromise any laws Fit Criterion: The app shall break 0 laws

4.8.2 Standards Requirements

NF-LSR

The app shall adhere to the GNU general public license Fit Criterion: The app shall adhere to the GNU general public license

4.9 Health and Safety Requirements

NF-LHSR
not applicable

5 Project Issues

5.1 Open Issues

We still need to investigate how to use Android Studio as we have little experience with it. Other open issues will be added after we have begun developing our implementation.

5.2 Off-the-Shelf Solutions

5.2.1 Ready-Made Products

- Camera Ruler Beta Version Android Application

- Application works, but is still being further developed before being released to general public on app store

5.2.2 Reusable Components

- Development not yet started, so reusable components not yet recognized (if available).

5.2.3 Products That Can Be Copied

- Can use interface and organization of Camera Ruler Beta Application

5.3 New Problems

5.3.1 Effects on the Current Environment

Adding in the feature to enable users to zoom into pictures for a more precise selection of the reference point may cause some errors which we will need to account for as the code right now does not account for zooming into the picture. If no additional features are added, no effects on the current environment are foreseen.

5.3.2 Effects on the Installed Systems

Recreating an Android application to work on Android devices should cause no errors. Also, this interface is planned to stand alone and hence, will not impact any other systems.

5.3.3 Potential User Problems

Prior to implementation, a potential user problem we think may occur is not being able to click the correct spot and hence, getting a slightly inaccurate measurement. Users will be warned that the measurement is only as accurate as they tap on the correct spot. However, as developers, we plan to try and create a zoom feature which will enable the user to be more accurate in point selection.

5.3.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

Being an already existing application that works in beta form, we know that our application, if done correctly and as planned, should work on Android devices. Only limitation is doing the application on Android Studio may be tough as there is a time constraint of approximately 2.5 months.

5.4 Tasks

Tasks for this project have been outlined by the course instructor of Software Engineering 3XA3. Deliverables include the Problem Statement, Proof of Concept plan, Requirements document revision 0, Test plan revision 0, Proof of concept demonstration, Design document revision 0, Revision 0 demonstration, Peer evaluation of other teams revision 0, Users guide revision 0, Test report revision 0, Final demonstration (Revision 1), and Final documentation (Revision 1). The final project is due on December 6th, 2017. Our project plan is created on a Gantt chart which is available on gitlab. Use of the waterfall method will be used to get our project successfully completed. However, the main thing is that we need to stay on top of time constraints as that is our only worry.

5.5 Migration to the New Product

5.5.1 Requirements for Migration to the New Product

We may need to find similar libraries to which Camera Ruler used to stay consistent with requirements which would work with Android Studio as well. However, we are still planning our implementation so this may change and we may not need similar libraries to the parent application.

5.5.2 Data That Has to Be Modified or Translated for the New System

No data needs to be translated for the new system, however, to be safe, we will verify the mathematical calculations provided by Camera Ruler to be safe. This verification is planned to take place right before the implementation of the application and will be done by ratio calculations.

5.6   Risks

Risks we found were related to the development of the product. It may be too complex for us as we are still learning Android Studio and making an Android Application. After having this completed, we also need to make sure it works properly where testing may also be complicated as there are multiple aspects to test such as if the picture taken is the background of the app, and if reference points are able to be placed and dragged to the correct spot. All in all, schedule pressure seems to be the only risk at the moment.

5.7   Costs

No money is going into the making of this application, but the cost we may have would be related to time as we may not get enough time to implement all that we have planned to implement at the moment.

5.8   User Documentation and Training

5.8.1   User Documentation Requirements

Instructions on how to use the application will follow in sequence on the application. With our planned interface being very simple, the user will be prompted to take a picture they want to measure along with a reference object in the same picture and then will be asked to pick points to measure in the metric they have selected. Due to this simplicity, we do not plan on creating a separate user manual/help guide, as the instructions will be on screen for the user to follow.

5.8.2   Training Requirements

As aforementioned, the simplicity of this application is such that no training will be required for the user.

5.9   Waiting Room

A new feature which could be added would be ~~dragging the reference points so that more accurate measurements could be made. To do this, more research will be done on dragging and dropping on a live interface. Another idea~~ to have more accurate measurements would be to zoom into a picture so the

exact point of measurement could be determined. As of now we only have measurements in straight lines, so a future implementation could be to allow users to measure objects in curved shapes as well.

5.10   Ideas for Solutions

For zooming into a picture, ~~dragging points~~, or measuring curved objects we do not have any ideas, so more research and planning will need to be done in order to actually implement these ideas.