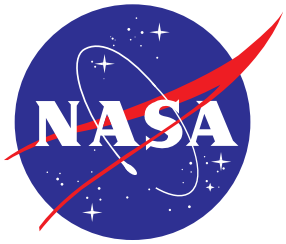


CS CAPSTONE REQUIREMENTS DOCUMENT

NOVEMBER 1, 2017

ISS BAROMETER APP

PREPARED FOR

NASA

DON PETTIT _____ *Signature* _____ *Date*

PREPARED BY

GROUP20

CADE RAICHART _____ *Signature* _____ *Date*

DANIEL KATO _____ *Signature* _____ *Date*

NATHAN SHEPHERD _____ *Signature* _____ *Date*

Abstract

This Requirements Document is meant as a repeatable and detailed description of using the iPad Air 2 Barometer to create an app that can be used to measure atmospheric pressure in emergency situations aboard the International Space Station. Within this document is the description, constraints and dependencies that will be used for this app. The Specific requirements that must be completed are laid out within the user, hardware and software interfaces sections. The System features intails the more in depth features of the pressure readings, recording button, graph and the settings pages.

CONTENTS

1 INTRODUCTION

1.1 Purpose

This document will detail the required final product of this project, the ISS Barometer App . The sections will go over the scope of the project, as well as the specific details of it's completion. This document will also serve as the metrics in which we will be graded at the end of Spring term. This Requirements Document is meant as a repeatable and detailed description of the problem that we aim to solve, as stated in our Problem Statement. That goal is to provide an auxiliary barometer to the MANOVACUMETERS aboard the ISS with an iPad app. The intended audience for this document is the client, NASA and Don Pettit, as well as anyone seeking to repeat this process, using the base setup. This document will also act as a success metric once this school year is finished, allowing clear decisive evaluation. Additionally it will provide a detailed agreement between the group and our client, to ensure that we are all on the same page. This document can be changed throughout the project, but scope changes must be submitted and verified by both teachers and clients.

1.2 Scope

The ISS Barometer App will measure the current pressure of its surroundings using the built in pressure transducer. The application will display the pressure in mmHg upon starting the app and whenever the 'Record Pressure' button is pressed the current pressure along with a timestamp will be recorded and displayed. It will also display the current pressure change in $\frac{\Delta t}{\Delta p}$ and include a graph of $\frac{\Delta p}{\Delta t}$ will be displayed with the capability to pinch to zoom and scale the graph. This graph will have a readout of $\Delta \frac{\Delta p}{\Delta t}$ displayed next to it. A settings page will be provided that allows the user to choose the plot scale of the graph, orientation of the app, and number of significant digits. The graph's data will also be exportable as a csv file. The application will not read remote pressure data. The ISS Barometer App will be used as a tool for astronauts to compute the amount of time they have before they must evacuate the International Space Station. This application has the benefit of allowing the user to track the change of pressure data more efficiently than taking readings from a manual barometer.

1.3 Definitions, Acronyms, and Abbreviations

- MANOVACUMETER: The current mechanical air pressure gauge aboard the ISS
- NASA: National Aeronautics and Space Administration
- ISS: International Space Station
- iOS: The internal operating system of Apple products, specifically iPads
- iPad: A touch screen tablet device with barometer capabilities
- mmHg: Millimeters of Mercury, the standard pressure unit
- t-Res: The chart that NASA uses to calculate residual time left on station

1.4 References

1.5 Overview

This document will follow the IEEE 830-1995 guidelines, in both headings and content. The following section, Overall Description, will detail the description of our application, The third section, Specific Requirement, will outline the requirements in user interface, hardware interface, software interface, and communications interface. Our Gantt chart

can be found in section 2.6, Schedule Dependencies, and lays out the rough schedule in terms of dependent features. It provides a birds-eye view of the schedule and dependencies of this project. It is subject to change, as more information and insight comes to light. The final section will describe, in detail, each feature, additionally it will describe second version features.

2 Overall Description

2.1 Product Perspectives

Today in the Apple store there are apps that use the internal barometer and displays earth's atmospheric pressure, one such app is Barometer and Altimeter Pro.Apps on the market are designed for earth's atmospheric pressure and to help with detection of weath. The product we are creating will be used for detection atmospheric pressure in space specifically from hull breaches. To accomplish the requirements the system interfaces will have three pages the display, graph and settings. The Display will have a big button, with a digital display of atmospheric pressure. The first page will display the initial pressure, current pressure and the pressure change as sec/mmHG. Graphs display will have a graph that spans the whole screen, that the user will have the ability to zoom in and out. The graph axis will be pressure vs the function of time and our slope will be dp/dt. The third page, settings page, will control the characteristics and look of the other two pages. Settings page will contain the option of changing the plot scale, scroll functionality, orientation, decimal point, as well as other units of measure for pressure. All displays and pages need to be able to be read fast and easily in low light. We will need to get the software to use the internal barometer that the ipad Air-2 uses, this will capture our atmospheric pressure for us. There is no outside interface that the app needs to use to complete the product. There is a chance that the running graph could impeded on the amount of memory the iPad has, the longer it's on the more data it stores. This being said the data isn't large and shouldn't be a problem.

2.2 Product Functions

- Data Screen with initial pressure, current pressure, start/stop button, time stamp, pressure change (measured in sec / mmHg)
- Graph Screen with plot pressure as a function of time, pinch to zoom functionality, scaling options (to be set in settings)
- Additional settings menu that includes plot scale, scroll functionality, orientation, decimal point, as well as other other units of measure for pressure
- Graph will be able to be exported as a csv, excel readable file
- All items will be optimized for efficient reading and clear interface

2.3 User Characteristics

The intended audience for the ISS Barometer App is astronauts and cosmonauts aboard the International Space Station. The personnel aboard the ISS are expected to have a working of knowledge of the relationship between $\frac{\Delta p}{\Delta t}$, $\frac{\Delta p}{\Delta t}$, and $\Delta\frac{\Delta p}{\Delta t}$. Because the intended user is aboard the ISS in space, it is expected that the app be used in micro-gravity.

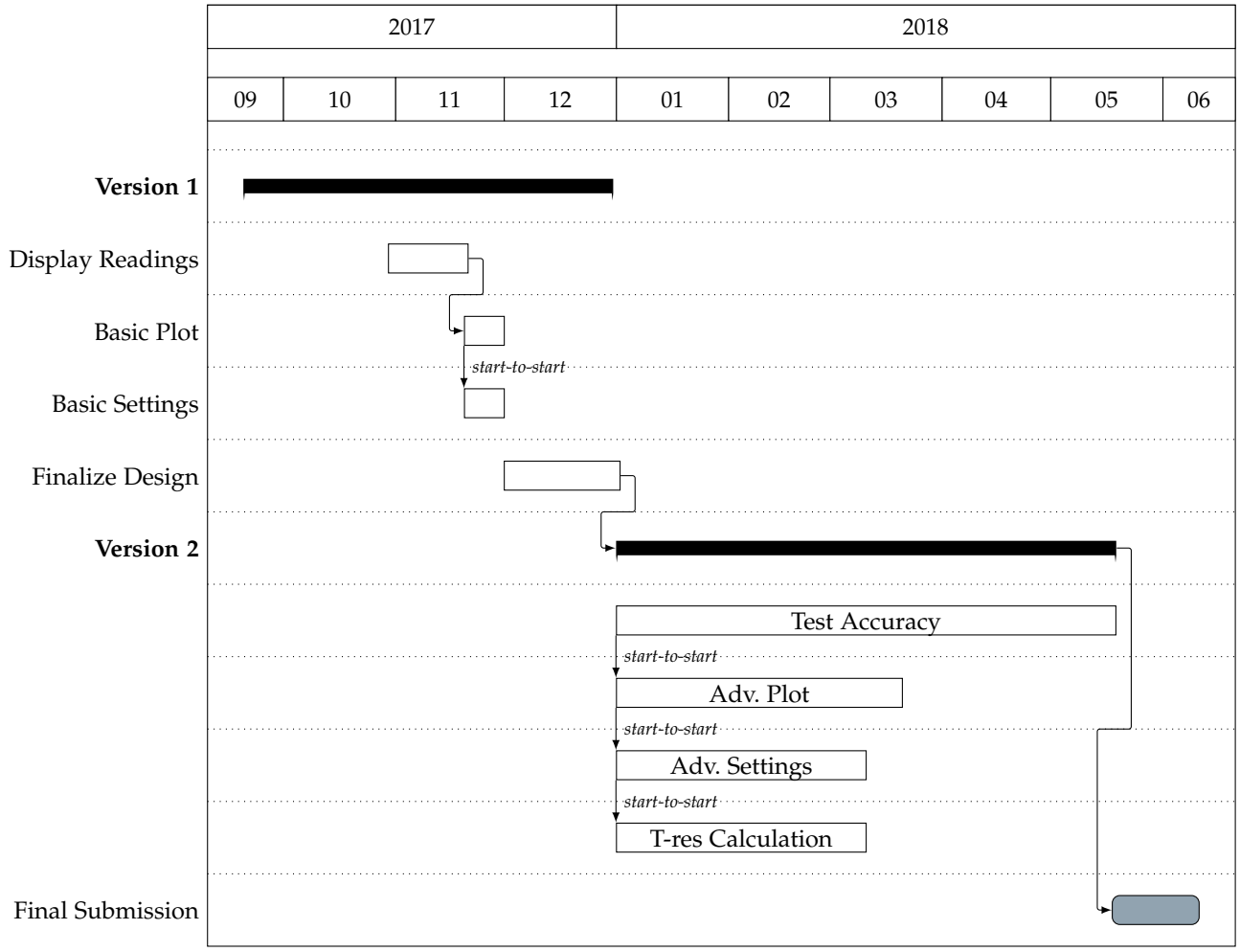
2.4 Constraints

The constraints of this project involve mostly the constraints in place by working on iPads. The language that we will use is Swift, which must be developed on Apple's IDE, XCode. This further limits our development environment options, in that XCode is only made for Mac OS machines, and not for Windows computers. The fact that this application will need to be deployed on the ISS also adds some constraints. One constraint is the download and size limits of iPad Apps; which must have executables under 500 MB, and a decompressed size of 4 GB. The download speed of the ISS, unknown at this time, will also limit the size of the app. The fact that the ISS is unreachable, and our audience are the few crew members aboard, problems and updates might take a while to be addressed and applied. Although not likely a problem, time also is a constraint, with our client requesting a product before the end of the calendar year, 2017. Based on what we know now, this will all be achievable.

2.5 Assumptions and Dependencies

We are assuming that the iPad Air 2 has a barometer inside of it as the manufacture has described. We will also assume that the ipad has enough memory to hold onto the running data and display it onto our timeline graph. The iPads OS on the ISS use version 10, if this is updated then there may become compatibility problems we will need to face. Our product mostly uses mm/Hg as our measurement for pressure, this is an assumption that our client may change and we will have to change. If any of our assumptions are incorrect or not available then we will have to change accordingly.

2.6 Schedule Dependencies



3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The user interface will include a button to record the pressure at that given moment, a graph and the ability to pinch-to-zoom, and a settings page. The record button will simply record the pressure at the time the button was pressed and display the result with a timestamp. The graph will be interactive and allow the user to zoom in on certain areas and scroll to view other portions of the graph. The settings page will include toggles to allow the user to change the number of significant digits, orientation, plot scale, scroll functionality, and other units of pressure. There will be no privileged users, so anybody who uses the app will be able to use all features.

3.1.2 Hardware Interfaces

The ISS Barometer App will not directly interface with any hardware, but will instead make use of Apple's iOS APIs in the Swift language that wrap all interaction with iOS hardware.

3.1.3 Software Interfaces

The ISS Barometer App will use the Apple's iOS APIs in the Swift 4 programming language to read pressure data from the pressure transducer, to display the data to the screen, and to create the graph of $\frac{\Delta p}{\Delta t}$. Each use of an iOS API will be carried out by a function call to the appropriate Swift 4 library. Information on these APIs can be found at https://developer.apple.com/documentation.

3.1.4 Communications Interfaces

The ISS Barameter App will not need to communicate outside of the iPad, it will need to use the internal barometer and exported as a csv to an excel file.

4 SPECIFIC SYSTEM FEATURES

4.1 Initial Pressure Reading

4.1.1 Purpose of Function

This number will be on the initial data screen. It will show the initial pressure reading, in the units defined in the settings page, defaulted to mm Hg. When the application is not in the recording mode it will show the current pressure, and once the record button is pressed it will show the pressure at the time of recording. This provides the users with a clear, simple view of the current pressure, and once recording starts, it will provide a base line pressure that will be used to gauge the trends.

4.1.2 Stimulus/Response Sequence

Before and after a recording session, this text will display the current pressure polled from the barometer at regular intervals, depending on the settings. Once the button is pressed, this number will pause, and display the pressure at the time the recording started, with a time stamp.

4.1.3 Associated Functional Requirements

- Ability to poll on-iPad barometer according to set rate
- Ability to save and time stamp the pressure, in specified unit
- Ability to start, stop, and discard/save recordings

4.2 Record Button

4.2.1 Purpose of Function

4.2.2 Stimulus/Response Sequence

4.2.3 Associated Functional Requirements

- Filler

4.3 Current(recording) Pressure Reading

4.3.1 Purpose of Function

This number will be shown on the initial data screen, during recording. It will show the current pressure reading, in the defined units. This number will be polled from the barometer on the iPad, at a rate defined in the settings, which will be no greater than 2 seconds. It will be the same clear text as the other readings, and have a decimal amount define in the settings. This measurement will be used for the $\frac{\Delta t}{\Delta p}$ plot and readout.

4.3.2 Stimulus/Response Sequence

This won't be visible before recording, but once the record mode is activated, then this will start displaying the current pressure. Once the recording is stopped it will pause, and will be stored with with the plot.

4.3.3 Associated Functional Requirements

- Ability to poll on-iPad barometer readings according to set rate
- Ability to start, stop, and discard/save recordings

4.4 Time over Pressure Change Reading

4.4.1 Purpose of Function

This display of change in pressure or $\frac{\Delta t}{\Delta p}$ will be used to inform the user of how quickly the surrounding pressure is changing. This will be used to track changes in pressure in real time.

4.4.2 Stimulus/Response Sequence

The pressure will be read at a consistant time interval, and at each reading, $\frac{\Delta t}{\Delta p}$ will be calculated and converted into the specified unit of measurement. This number will then be displayed in a readable manner on the main page

4.4.3 Associated Functional Requirements

- Ability to poll on-iPad barometer readings according to set rate
- Ability to control the view of the app

4.5 Plot of Pressure Change

4.5.1 Purpose of Function

This graph will display the change in pressure over time, in such a way that the user can quickly see the whole timeline.

4.5.2 Stimulus/Response Sequence

The graph's y axis will be pressure, while the x axis will be the function of time. The slope of the graph will be dp/dt, it will be constantly updated as the app is run. The graph will be a running graph. The time constraint is always getting longer and the slope is constantly updated to give you the newest read.

4.5.3 Associated Functional Requirements

- Ability to store and calculate dp/dt
- Ability to zoom into and out of the graph.
- Ability to display the whole timeline or only the newest data

4.6 Configurable Settings Page

4.6.1 Purpose of Function

This will be its own page in the application that allows the user to configure the app in a way that they like. The page will include toggles for number of significant digits, orientation, plot scale, scroll functionality, other units of pressure, and sample rate.

4.6.2 Stimulus/Response Sequence

When the user selects a number from the significant digits selector, the all of the data displayed on the main page should have that number of digits to the right of the decimal place. When a user selects an orientation from the menu, the display should roll to that orientation in the settings view, and persist to the main page. When a user selects a plot scale, the graph on the main page's scale will adjust to match. When a user selects a specific graph scrolling functionality, the graph will change to reflect. When a user selected a different unit of measurement, the pressure displays on the main page will use that unit instead of the default mmHg. When a user selects a sample rate, the graph and pressure display will update at the specified number of times per second.

4.6.3 Associated Functional Requirements

- Ability to poll on-iPad barometer according to set rate.

4.7 V2 Consult T-Res

4.7.1 Purpose of Function

This function would allow users to look up the t-Res values in relation to the $\frac{\Delta t}{\Delta p}$. The t-Res look up would provide users with time remaining in a leak situation. This is something that is already done aboard the ISS, but with the addition of a digital table it would make the time spent much shorter. The display would provide the time remaining, with context of NASA's parameters for evacuation.

4.7.2 Stimulus/Response Sequence

This functionality would be available after a user had opted for it in the form of a button. It would then display time remaining, as a function of pressure loss.

4.7.3 Associated Functional Requirements

- Ability to poll on-iPad barometer readings
- Ability to use t-Res look up, and access to evacuation parameters
- Ability to show/hide information

4.8 V2 Save Function

4.8.1 Purpose of Function

4.8.2 Stimulus/Response Sequence

4.8.3 Associated Functional Requirements

- Filler