Requirements Analysis Document

CITS3200 Project GROUP P

Revision History:

Version R1.0 26/08/2016

Preface:

This document outlines the scope of work to be done in extending the *Modelling Why We Wheeze* project.

Target Audience:

Group P members, Professor Peter Henry (client), Professor Michael Wise (unit coordinator).

Members:

21490093 - Tom Ankers

20503664 - Michael Baxter

21469038 - Dylan Johnson

21480279 - Gary Liao

21522889 - Harrison Marley

21506136 - Damon van der Linde

Meeting Schedule:

Meeting time	Signoff (initial)
Wed 10/08/2016 3pm, Science Library	
Wed 17/08/2016 3pm, Science Library	
Wed 24/08/2016 3pm, Science Library	
Wed 31/08/2016 3pm, Science Library	
Wed 07/09/2016 3pm, Science Library	
Wed 14/09/2016 3pm, Science Library	
Wed 21/09/2016 3pm, Science Library	
Wed 28/09/2016 3pm, Science Library	
Wed 05/10/2016 3pm, Science Library	
Wed 12/10/2016 3pm, Science Library	
Wed 19/10/2016 3pm, Science Library	
Wed 26/10/2016 3pm, Science Library	

Milestones

26/08/2016 Scope Documentation Draft (Deliverable A)

09/09/2016 Scope Document Final, Planning Documents (Deliverable B)

21/09/2016 Implementation Complete, Verification Documents (Deliverable C)

31/10/2016 Seminar Presentation (Deliverable D)

Client Sign Off

1.0 General Goals

The end product is to be a web based interactive application intended as an educational aid in UWA's Biomedical Science major. The application will provide a cross-sectional visualisation of an asthma patient's airways that responds in real time to changes in parameters modelling the patient's condition and treatment. Educational benefit comes from students being able to observe the physiological causes of airway obstruction due to the individual effects of each contributing factor.

2.0 Current System

The current system consists of a single web page. This web page provides a diagram of an airway, showing its 3-layered construction, accompanied by a graph expressing the current degree of airway obstruction. A user can manually adjust the size of each airway layer with a set of sliders to simulate asthma conditions. A second set of sliders allows the user to virtually administer medication doses and observe the effects on the airway.

3.0 Proposed System

3.1 Overview

Extensions to the system will consist of two additional web pages. Each page introduces new input and visualisation elements. Additional mathematical models govern the behaviour of these elements.

The first page explores the AHR (Airway Hyper-Responsiveness) condition endured by asthma sufferers, whereby airways drastically constrict in response to a minor stimulus. A clinical dose of methacholine can be simulated and causes a contraction of the ASM (Airway Smooth Muscle) on the airway cross-section diagram. The effects will also be shown through dynamic graphs of ASM shortening (%) and airway resistance against methacholine dose.

The second page enables the user to change the parameters of the original model to represent different airway generations. The effects of methacholine are visualised using another airway cross-section diagram and a dynamic graph of airway resistance against dose level.

3.2 Functional Requirements

The tables below list functional requirements with assigned client values out of $100, expected hours to complete, value estimate ratio (client value ÷ expected hours) and priority. Table 3.2.1 lists the requirements of the AHR page. Table 3.2.2 lists the requirements of the parameterised model page.

Req. No.	Title	Description	Client Value	Est. hours	Value Est. Ratio	Priority
1.1	R vs. methacholine graph	A graph of airway resistance vs. methacholine dose is displayed on page.	$6.50	6.5	$1.00	11
1.2	Airway cross section diagram	A diagram of airway cross-section is displayed on page.	$8.50	4.5	$1.89	3
1.3	ASM shortening vs. methacholine graph	A graph of ASM shortening % vs. methacholine dose is displayed on page.	$6.50	6.5	$1.00	12
1.4a	Asthma severity slider function (a)	User can change asthma severity via a discrete horizontal slider, which will highlight a different curve in req. 1.1	$6.50	6.5	$1.00	13
1.4b	Asthma severity slider function (b)	Asthma severity slider will also change the airway wall inflammation and mucus secretion levels in req. 1.2	$6.50	6	$1.08	9
1.5a	Methacholine dose slider function (a)	User can change the methacholine dose via a continuous discrete slider, which will change a position indicator on req. 1.1	$4.50	6.5	$0.69	14
1.5b	Methacholine dose slider function (b)	Methacholine dose slider will also change the ASM contraction amount in req. 1.2	$6.50	6	$1.08	10
1.5c	Methacholine dose slider function (c)	Methacholine dose slider will also change a position indicator on req. 1.3	$4.50	9	$0.50	15

Table 3.2.1 List of functional requirements for AHR model page

Req. No.	Title	Description	Client Value	Est. hours	Value Est. Ratio	Priority
2.1	Display equation	An equation representing the mathematical model used to calculate airway resistance R is displayed on page.	$5.00	1	$5.00	1
2.2	Description text	Text describing the model is displayed on page.	$2.50	1	$2.50	2
2.3	Airway cross section diagram	A copy of the airway cross-section diagram (req. 1.2) is displayed on page.	$7.50	4.5	$1.67	6
2.4	R vs. methacholine graph	A graph of airway resistance vs. methacholine dose is displayed on page.	$10.00	5.5	$1.82	5
2.5a	Parameter buttons function (a)	User can change model parameters via parameter button (up/down). The values of these parameters are dynamically displayed next to the buttons.	$7.50	4	$1.88	4
2.5b	Parameter buttons function (b)	Parameter button will also change req. 2.3 in accordance with the model.	$7.50	6	$1.25	7
2.5c	Parameter buttons function (c)	Parameter button will also change the shape of req. 2.4 in accordance with the model.	$10.00	8	$1.25	8

Table 3.2.2 List of functional requirements for parameterised model page

3.3 Non-functional Requirements

3.3.1 User Interface and Human Factors

The intended primary users of the application are undergraduate biomedical science students. Educational benefit is gained through interacting with dynamic visualisations. It is important that the user interface is intuitive and that it is easy to identify what each element represents and the unique effect it has on the model.

It is reasonable to expect that student users will have medical knowledge provided by their course. Therefore the interface requires clear labelling of all visualisations, without the need for in depth explanations.

3.3.2 Documentation

Documentation is important for the future maintainers of the system, and will be provided through adequate in-code comments and a system handbook.

3.3.3 Hardware Consideration

The system will be accessed in class on standard university workstations, but may also operate on home PCs, laptops, tablets and mobile devices. Therefore constraints on memory usage must be considered and the software must be written in a reasonably efficient manner.

3.3.4 Performance Characteristics

The application must be responsive to reduce user dissatisfaction. Given that the application will not store data locally or send it back to servers, the primary issue will be ensuring that pages load efficiently.

3.3.5 Error Handling and Extreme Conditions

The application will be designed to limit user input to a narrow range of interactions, thus reducing errors caused by invalid inputs.

3.3.6 System Interfacing

The web pages should render on a variety of browsers on several platforms, otherwise the system must inform the user of incompatibility issues.

3.3.7 Quality Issues

N/A

3.3.8 System Modifications

The user interface may face future alteration or extension. New visualisations or input controls may be added, and the visual style of the interface may be changed. Flexibility is provided by keeping each interface element logically distinct and modular, and by separating their graphical representations from the code that calculates output.

3.3.9 Physical Environment

N/A

3.3.10 Security Issues

The system will not communicate user data back to the server, limiting security risks to the user and the server.

3.3.11 Resource Issues

The application will reside entirely on the client's website. Maintenance and backup will be responsibilities of the website administrator.

3.4 Constraints

3.4.1 Programming Language

The system will be implemented using the HTML mark-up language, CSS style language and JavaScript scripting language. Server side scripting will not be used so the web pages can be operated locally.

The use of JavaScript has several limitations: there is limited multithreading support, it is a prototype-based object oriented language and it is an interpreted language where anyone can view, copy and modify the source code.

3.4.2 Rendering

Different browsers use different JavaScript engines, which can cause rendering issues and inconsistencies. This will constrain the development to cross-browser standards that will work on the four major browsers (Chrome, Firefox, Safari and Edge/IE).

3.4.3 Libraries

jQuery, a free and open source JavaScript library, is used in the current system and will be used to extend the system. jQuery requires browser support, which restricts the use of legacy browsers. Table 3.4.3.1 lists browsers supporting the current jQuery release.

Browser	**Version**
Chrome	51.x, 52.x
Edge	37.x, 38.x
Firefox	47.x, 48.x
Internet Explorer	9+
Safari	8.x, 9.x
Opera	38.x, 39.x

Table 3.4.3.1 List of browser versions that support jQuery 1.12.4.

3.5 System Model

3.5.1 Scenarios

As a student user who can visually interact with the model:

- I am able to separately adjust muscle contraction, airway wall inflammation, and mucus secretion, through a slider or number entry
- I can visualise airway response to asthma medication at different doses using a slider or number entry
- I can see how the airways respond to varying methacholine doses, which I control
- I can select or randomise an assortment of airway generations
- I control manually the different airway generations
- I am able to select which information to show or hide
- I am visually informed by way of graph and airway visualisation, how the variables interact

As an administrative user:

- I can easily find the object I am looking for in the source code
- I am able to read documentation to understand how the application pieces together
- If I want to add a new element, I can refer to a style guide to ensure continuity
- I am able to alter a function/object without impairing functionality
- I can navigate through the file structure without getting lost
- I can traverse the source files logically because classes and function calls are well ordered

3.5.2 Use Case Models

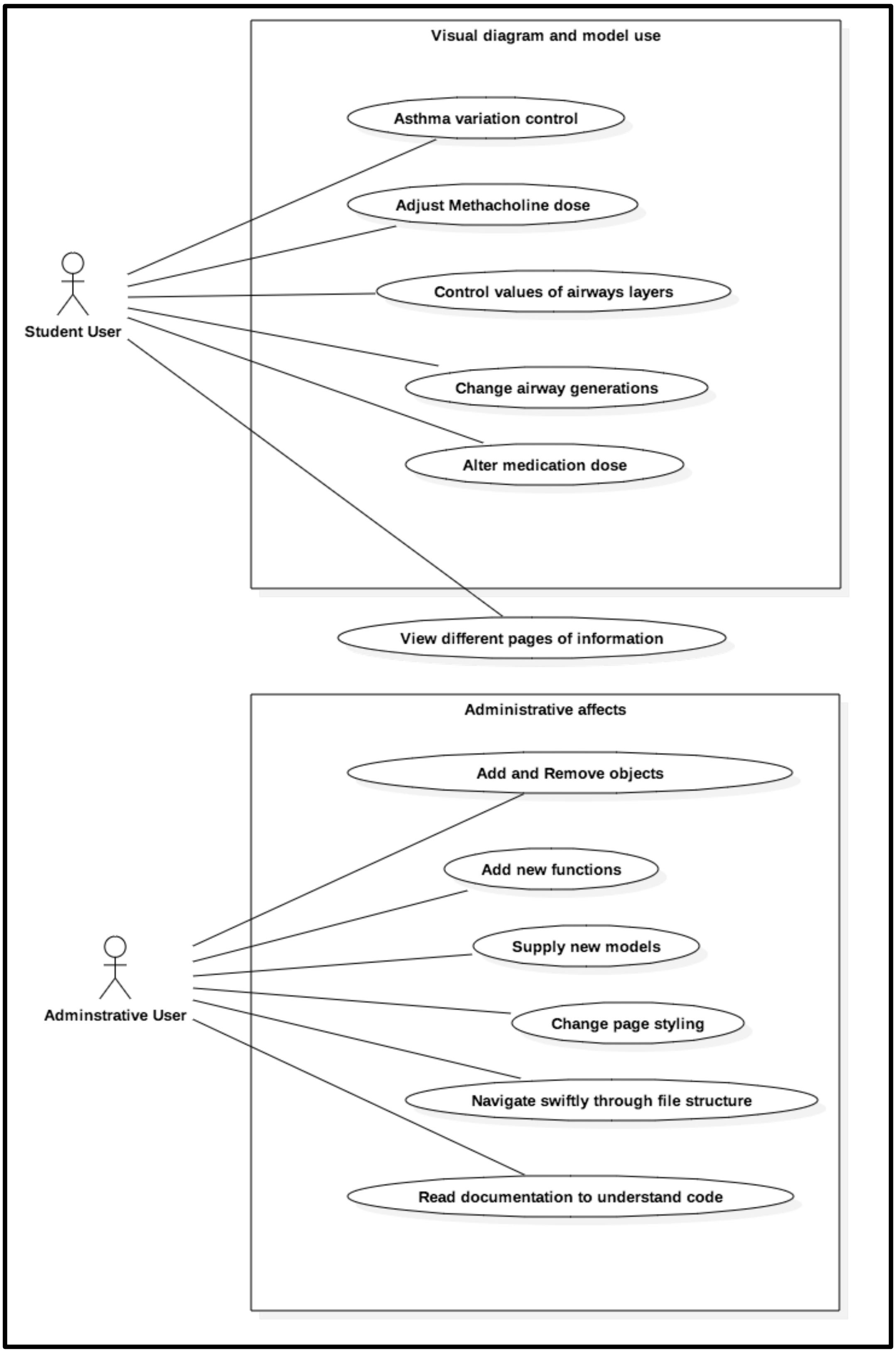


Figure 3.5.2.1 Use case diagram

3.5.2.1 Actors

- Students / General users
- Administrative users

3.5.2.2 Use Cases

Student/General User:

- Use the model to visualise AHR for varying levels of asthma (controlled with slider)
- Visualise different projections when adjusting methacholine dose
- Accurately adjust the visual model for different airway generations
- Visualise ASM shortening on a graph that updates dynamically with sliders/user input
- Visualise airway resistance on a graph that updates dynamically with sliders/user input
- Control airway layers by click-dragging and/or number entry
- Customise the widget display:
    - Hide/reveal input fields/graphs

Admin:

- Add new functions/objects easily by following a clear style used throughout the rest of the project
- Read clearly written documentation to easily identify what pieces of the program fit together and how
- Altering existing functions and objects without hurting functionality (i.e. minimise negative referential integrity)
- Navigate through a well ordered file structure
- Traverse through the source files following a logical path of well ordered classes and function calls

Use case ID	Use Case Name	Actor	Complexity	Priority
1	Asthma variation control	General user	med	high
2	Methacholine dose	General user	med	high
3	Airway generation control	General user	low	high
4	ASM shortening graph	General user	low	high
5	Airway resistance graph	General user	low	high
6	Number entry control of airway layers	General user	med	med
7	Widget display	General user	med	low
8	Select from a range of pre-sets	General User	low	low
9	Neat file structure	Admin	low	med

10	Code documentation	Admin	low	med
11	Style guide	Admin	med	med

Table 3.5.2.2.1 Use case index

3.5.2 Object Models

3.5.3.1 Data Dictionary

See Appendix for draft data dictionary. (Not Deliverable A requirement)

3.5.3.2 Class Diagrams

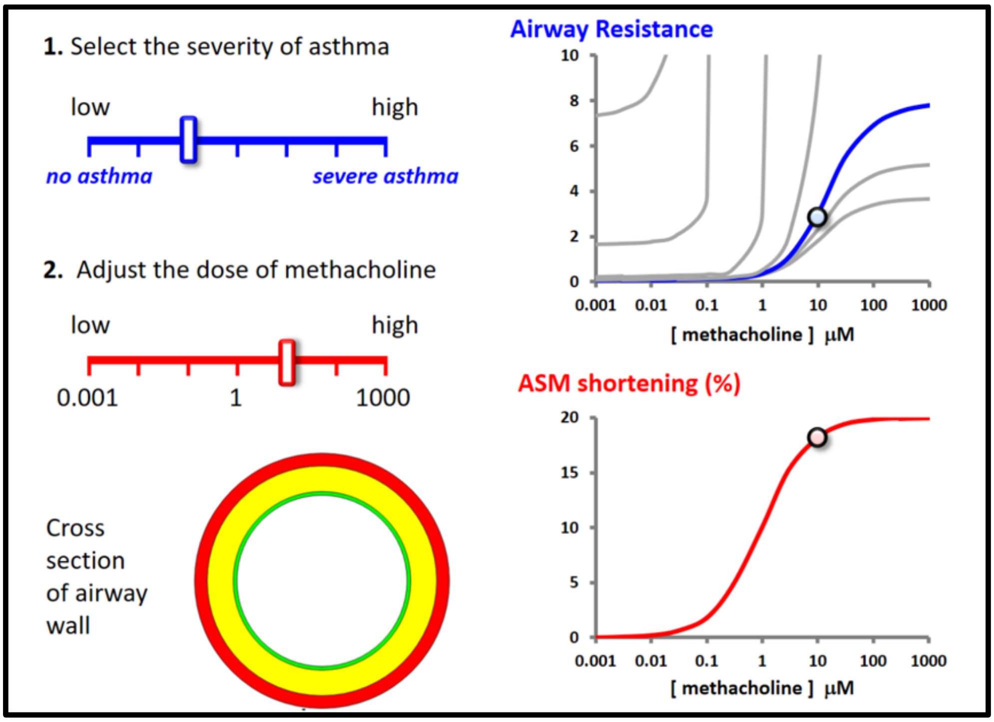
See Appendix for draft class diagrams. (Not Deliverable A requirement)

3.5.4 Dynamic Models

A dynamic model represents the behaviour of an object over time.
The components of the dynamic model are: States, state transition, events, actions and activities.
In the case of this project, a state can be seen as the graphical representation of the airway. As the input independent variable changes (an event), a state transition occurs. The jQuery or JavaScript will modify the CSS elements as an action. Lastly, the graph will show the final output of the airway modelling which can be considered as an activity.

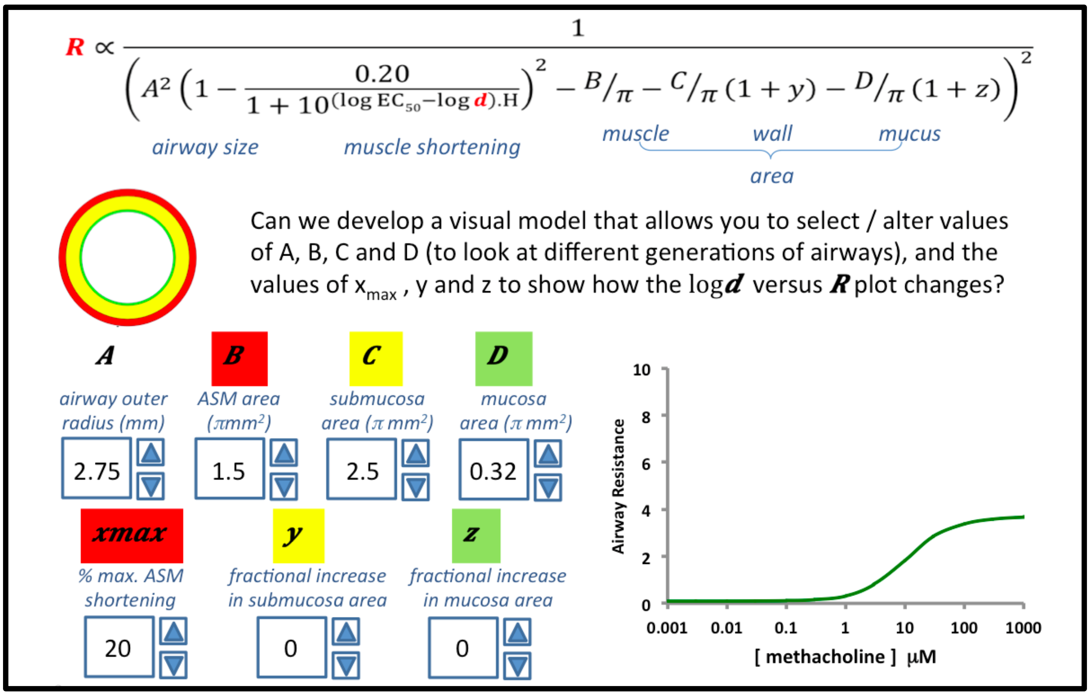
3.5.5 User Interface - Navigational Paths and Screen Mock-ups

Screen 1:



Courtesy of Associate Professor Peter Henry

Screen 2



Courtesy of Associate Professor Peter Henry

4.0 Glossary

AHR	Airway Hyper-Responsiveness
ASM	Airway Smooth Muscle, outermost layer of an airway
CSS	Cascading Style Sheet, used to define visual presentation of web pages.
Deliverable A	Scope Documentation Draft
Deliverable B	Scope Document Final, Planning Documents
Deliverable C	Implementation Complete, Verification Documentation
Deliverable D	Seminar Presentation
GUI	Graphical User Interface
HTML	Hyper Text Mark-up Language, used to write web pages.
JavaScript	A scripting language used to give dynamic behaviour to web pages.
jQuery	A free and open source JavaScript library.