9.09am 9/12/2016 – Created Logbook Journal

9.32am 9/12/2016 – Created portfolio and added an identification of the problem statement.

4.51pm 5/01/2017 – Added ideas for solutions to the problem.

5.13pm 6/01/2017 – Added requirements for the program.

9.27pm 12/01/2017 – Added Gantt Chart for the project.

10.00pm 13/01/2017 – Created Storyboard

9.48pm 15/01/2017 – Added storyboard and context diagram to the portfolio.

9.35pm 19/01/2017 – Added Feasibility Study to the portfolio.

9.36pm 27/01/2017 – Added Social and Ethical Issues to the portfolio.

11.15am 3/02/2017 – Submitted Stage 1 of project. Received feedback from Rohan Barar to use Excel for the graph instead of drawing dots.

8.41pm 7/02/2017 – Created Data Flow Diagram

9.03pm 14/02/2017 – Created System Flowchart and Specifications

10.06pm 24/02/2017 – Added Test Data to program using the projectile motion problems from the book “Surfing Space” and created IPO Chart

9.31pm 25/02/2017 – Added Feedback for Projectile Motion Analyser and Data Structures

(Note: There is a gap here due to HSC Half-Yearly Exams)

9.03pm 10/04/2017 – Started implementation of the program, tested both creating a graph by drawing lines and by using Excel.

9.35pm 11/04/2017 – Found that drawing curved line graphs in Excel is possible (credits to Gordon Zhong and Rohan Barar). Made basic UI and included a PictureBox to display the Excel graph.

9.14pm 12/04/2017 – Used questions from “Surfing Space” book to find all variables that can be obtained and outputted from projectile motion problems, and listed them in the program. Constructed first algorithm which uses the initial velocity and angle as inputs.

8.40pm 13/04/2017 – Constructed second algorithm which uses initial velocity and time as inputs, by using the 2nd problem from the Surfing Space book. This algorithm had issues as Visual Basic 6 does not support inverse sin functions, and so the inverse tan function had to be used to work around this problem.

8.32pm 14/04/2017 – Constructed third and fourth algorithm which uses range and time as inputs and range and maximum height as inputs respectively. These algorithms could solve the rest of the problems given in the Surfing Space book except one.

9.47pm 15/04/2017 – Constructed fifth and last algorithm. This algorithm was quite hard to create and required rearranging variables carefully. The famous hard “Angry Bird” question from the HSC Physics 2012 paper uses this algorithm, and so many logic errors were given before the algorithm started working.

9.58pm 16/04/2017 – Created excelGraph function that would display the graph. The graph, although does show the curve, also shows another straight blue line that is unrelated to the projectile motion problem. The x axis also displays no other values than the numbers 1-11. These bugs have not been solved. Also created outputFunc function that outputs all projectile motion variables. An empty variable called “holder” had to be used in order to execute these functions due to limitations in Visual Basic 6.

8.23pm 17/04/2017 – Created user documentation and added it to portfolio. The program will be updated, so this is temporary documentation.

8.51pm 18/04/2017 – Started testing the program. Using HSC questions to aggressively test the program.

12.23pm 27/04/2017 – Feedback for projectile motion program received. Includes hiding the blue line in the graph, and ensuring error is not given then acceleration is made equal to zero.

12.21pm 30/04/2017 – Overhauled UI so entering variables into the program is easier. Found bug where if Algorithm5 is run with heightEnd greater than 0, the program crashes as a negative number cannot be square rooted. Found another bug where if Algorithm2 is run with an initial height greater than 0, the program crashes.

12.48pm 1/05/2017 – Found that bug is due to the variables given are impossible to occur in real life projectile motion, and the program was crashing due to it being unable to square root a negative number. Error messages were added to the program to ensure the program does not crash when this situation occurs. Both bugs were fixed.

6.35pm 1/05/2017 – Tested Algorithm5 again with HSC Question 27 2012 Physics, program returned correct answer. The algorithm currently is clear of bugs.

8.01pm 2/05/2017 – Added reset button to program, and fixed graph bug where 2 lines instead of 1 appear. This bug occurred due to a bug in Excel which makes it create 2 data plots instead of one if a line graph is created, but can be worked around by creating a scatter graph and changing the type to a line graph later.

12.23pm 5/05/2017 – Tested program on school computers. Excel currently does not work on school computers, possibly due to administrator privileges required.

8.11pm 7/05/2017 – Added maintenance section to portfolio.

9.01pm 13/05/2017 – Updated portfolio and improved some sections

1.03pm 15/05/2017 – Received feedback that textboxes should be disabled so no data can be entered. Found bug that if no data is entered, the program continues to attempt to calculate then crashes. Bug has been fixed

9.53am 16/05/2017 – Updated Gantt Chart and made it more comprehensive.

12.10pm 21/05/2017 – Distributed program to other people to ensure that program works on other computers. The program required MSSTDFT.dll to be placed in C:\Windows\syswow64 and added to the registry for it to work on computers that do not have Visual Basic 6 installed. As long as Microsoft Excel is installed, the program works.

1.21pm 21/05/2017 – Edited the program so the variables outputted are NOT user editable so users do not get confused that to reset the program they can change the variables outputted and click “Calculate”. This was changed due to feedback from other users. Also modified user documentation to the updated UI.

2.47pm 21/05/2017 – Received feedback from user where when ENTER is pressed, OK should be clicked instead of manually having to press OK. Also modified program so no negative numbers could be inputted.

8.05pm 22/05/2017 – Updated Data Dictionary. Added testing information and feedback. Added MIT License.