Andrew Campbell

Term Milestone Part 1

Project 5: Happy Heart Program

-----------------------------------------------------------------------------------------------------------

**Agile: Planning the Project:**

This is a heart function monitoring system that will raise an alarm if there is something wrong. It will be monitoring Pulse Rate, Blood pressure, and Blood oxygen level. This is a simulation so the data will be imported or typed in.

Requirements:

Inputs - There will need to be a way to read in the data from a file or be typed in manually

* Filename can be given on command line and there isn’t one then assume the input will be manually entered with the keyboard.
* The file input will need to simulate collecting data every 10 seconds or the user will need to enter every 10 seconds.
  + Each line input will need a pulse reading, an oxygen level (which might not be there), and a blood pressure (which is often not there.)
* Pulse and oxygen level will be read every time and get a blood pressure reading at irregular intervals (somewhere around every few minutes(not specified))

Outputs – Every time data is read it needs to be printed out with a time stamp, alert level, then a description of the problem causing the alert.

* Start time at 0 and do it in 10 second intervals.
* Alert levels are as followed: None, Low, Medium, Highest
* Description are as followed:
  + None: “Everything Normal”
  + Low: either “Blood Pressure Above Normal” or “Blood Pressure Below Normal”
  + Medium: “Blood Pressure is too High” or “Blood Pressure is too Low”
  + Highest: “Blood Pressure is Dangerously High” or “Blood Pressure is Dangerously Low”
* Times are mm:ss, and wrap around every hour.

Incorrect Data – Bad data or impossible values are considered equipment malfunction and should be a minor alarm.

* Wrong inputs should tell user and mark it as invalid and keep program running.

**Monitoring Alert Levels:** 

Project Pulse levels:

* 0 – 19 = "Pulse Extremely Low:" Alert Level = Highest
* 20 – 39 = "Pulse Below Normal:" Alert Level = Medium
* 40 – 110 = “None” Alert Level = None
* 111 – 130 = "Pulse Slightly Above Normal:" Alert Level = Low
* 131 – 170 = "Pulse Above Normal:" Alert Level = Medium
* 171 – 210 = "Pulse Extremely High:" Alert Level = Highest

Project Blood Oxygen levels:

* 0 – 49 = "Oxygen Extremely Low:" Alert Level = Highest
* 50 – 79 = "Oxygen Really Low:" Alert Level = Medium
* 80 – 84 = "Oxygen Low:" Alert Level = Low
* 85 – 99.9 = “None” Alert Level = None

Project Blood Pressure Levels (systolic(top) and diastolic(bottom)):

* Systolic > 200 = "Pressure Dangerously High:" Alert Level = Medium
* Systolic 151 – 199 = "Pressure Somewhat High:" Alert Level = Low
* Systolic 70 – 150 = "None" Alert Level = None
* Systolic 69 – 50 = "Pressure Somewhat Low:" Alert Level = Medium
* Systolic < 50 = "Pressure Dangerously Low:" Alert Level = Highest
* diastolic > 200 = "Pressure Dangerously High:" Alert Level = Medium
* diastolic 151 – 199 = "Pressure Somewhat High:" Alert Level = Low
* diastolic 70 – 150 = "None" Alert Level = None
* diastolic 69 – 50 = "Pressure Somewhat Low:" Alert Level = Medium
* diastolic < 50 = "Pressure Dangerously Low:" Alert Level = Highest

Ranges for Each category:

* Pulse Rate- Range: 0 – 260
* Blood Pressure- systolic/diastolic range: systolic – 0-260, diastolic- 0-150.
* Blood oxygen level- Range: 0 – 99.9

**Agile: Creating the Product Roadmap**

- 09/24 – 10/09 coding

- 10/10 – 10/16 Test plan

- 10/16 – 10/23 Test: Inputs, Frequency, Alerts, Time

- 10/23 – 10/30 Test: Data, Bad Data, Monitoring, Oxygen

**Agile: Planning the Release**

The project is due October 30th so this is the release date.

**Agile: Planning the Sprint(s)**

Sprint 1: 09/24 – 10/09, Develop code to the requirement specifications.

Sprint 2: 10/10 – 10/16, Code, Create test plan to the requirement specifications.

Sprint 3: 10/16 – 10/23, Code, Test: Inputs, Frequency, Alerts, Time

Sprint 4: 10/23 – 10/30, Code, Test: Data, Bad Data, Monitoring, Oxygen

**Agile: 2week Stand-up Meetings**

09/25: Sprint 1- Start coding

10/09: Starting coding/sprint 1 start sprint2 and when done do sprint3. Missing requirements to work on. Start creating test plan. Run sprint 3 tests after test plan created.

10/23: Work on sprint 4.

**Agile: Sprint review and retrospective meetings**

09/25: Sprint1 - Coding.

10/09: Some unfinished requirements: Randomly getting pressure inputs, always getting oxygen level. Start test plan and while testing address missing requirements if possible. Do sprint 2 and then 3.

10/23: Sprint 4.

**Testing Plan:**

**Scope:**

Inputs: Test if it allows the data to be read in from a input file or data to be entered in manually through the keyboard.

Frequency: Test to see if it is getting a pulse and oxygen level every 10 seconds and pressure at random.

Alerts: Test alerts to see if the proper alert is showing depending on the level and the test that’s being run.

Time: Test to see if time wraps around every hour.

Data: Test to see if the program displays correctly when oxygen and/or pressure is not entered/given.

Bad Data: Test to see if program recognizes bad data and reports it as invalid.

Monitoring: Test that it is showing the highest alarm level when there are multiple alarms in the order of precedence.

Oxygen: Test if displays previous level if no input is given or alert if 3 consecutive no inputs.

**Strategy:**

***Inputs***: Decision Coverage Testing

Testing: readInData() and manualInput() functions

* Input: yes, Output: file input.
* Input: no, Output: manually input.
* Input: Anything else, Output: re-enter option.

***Frequency***: Equivalence Testing

Testing: manualInput() funtion

* Input: 2 lines of data, Output: 10 seconds.

***Alerts***: Boundary Testing

Testing: alertLevel() function

* Pulse:

Testing: pulseRate() function

* + Testing all level boundaries (Boundary testing, tests are at the end)
* Oxygen:

Testing: bloodOxygenLevel() function

* + Testing all level boundaries Testing all level boundaries (Boundary testing, tests are at the end)
* Pressure:

Testing: bloodPressure() function

* + Testing all Systolic level boundaries Testing all level boundaries (Boundary testing, tests are at the end)
  + Testing all Diastolic level boundaries Testing all level boundaries (Boundary testing, tests are at the end)

***Time***: Equivalence Testing

Testing: readInData() function

* Input: 8 lines of data, Output: 1 min 10 sec.

***Data***: Decision Coverage Testing

Testing: I was unable to implement a way to not have all 3 inputs entered. Therefor I am unable to test these.

* Input: Pulse, Output: Check alert for pulse and oxygen (based on previous oxygen reading).
* Input: Pulse and Oxygen, Output: Check alert for pulse and oxygen (based on previous oxygen reading).
* Input: Pulse Oxygen Pressure, Output: Check alert for pulse, oxygen (based on previous oxygen reading) and pressure.
* Input: No inputs, Output: Display: invalid

***Bad Data***: Equivalence Testing

* Pulse:

Testing: pulseRate() function

* + Input: 211 =Output: Invalid Input
  + Input:-1 =Output: Invalid Input
* Oxygen:

Testing: bloodOxygenLevel() function

* + Input : 101 =Output: Invalid Input
  + Input: -1 =Output: Invalid Input
* Pressure:

Testing: bloodPressure() function

* + Systolic:
    - Input: 231=Output: Invalid Input
    - Input: -1 =Output: Invalid Input
  + Diastolic
    - Input 151 =Output: Invalid Input
    - Input: -1 =Output: Invalid Input

***Monitoring***: Equivalence Testing

Testing: alertLevel() function

* Input: Highest levels for all 3, Output: Displays Pulse alarm
* Input: Highest levels for Oxygen and Pressure, Output: Displays Oxygen alarm
* Input: Highest levels for Pulse and Pressure, Output: Displays Pulse alarm

Similar test for all the levels.

**Resources:**

* Requirements Document – For project description and requirements.
* Microsoft Word – For writing the test plan.
* Visual Studio Code – For developing the code.
* Google – For researching and help with coding.
* Slides – For referencing how to create test plan.

**Timeline:**

- 09/24 – 10/09 Coding

- 10/10 – 10/16 Coding, Test plan

- 10/16 – 10/23 Coding, Test Plan: Inputs, Frequency, Alerts, Time

- 10/23 – 10/30 Coding, Test Plan: Data, Bad Data, Monitoring, Oxygen

**Risks:**

* Run out of time
  + Focus on main functionality parts first to have a working project to turn in.
  + Focus on main parts of test plan to have a decent test plan to turn in.
* Stuck on implementing some requirements
  + Research using Google to try and implement.
  + Make sure program works without them and come back to implement if I have time.
* Too much testing
  + Design test plan to minimize duplicate testing and stick to it.
* Too little testing
  + Design test plan to cover all requirements so nothing is missed.
* Focused on less important parts compared to important parts
  + Identify major parts of development and work on them first.
* Misunderstanding requirements
  + Read through requirements thoroughly to understand
  + If developed a part wrong adapt it to make it pass the requirement
  + If developed a part wrong and cant adapt it, delete it and start over.

**Boundary Testing Cases Alerts:**

Testing Pulse:

* Input: -1 =Output: Invalid
* Input: 0 = Output: Highest alert
* Input: 1 = Output: Highest alert
* Input: 19 = Output: Highest alert
* Input: 20 = Output: Medium alert
* Input: 21 = Output: Medium alert
* Input: 39 = Output: Medium alert
* Input: 40 = Output: None
* Input: 41 = Output: None
* Input: 109 = Output: None
* Input: 110 = Output: Low alert
* Input: 111 = Output: Low alert
* Input: 129 = Output: Low alert
* Input: 130 = Output: Low alert
* Input: 131 = Output: Medium alert
* Input: 169 = Output: Medium alert
* Input: 170 = Output: Medium alert
* Input: 171 = Output: Highest alert
* Input: 209 = Output: Highest alert
* Input: 210 = Output: Highest alert
* Input: 211 = Output: Invalid

Testing Oxygen:

* Input: -1 = Output: Invalid
* Input: 0 = Output: Invalid
* Input: 1 = Output: Highest (based off moving average of last 6 readings)
* Input: 49 = Output: Highest (based off moving average of last 6 readings)
* Input: 50 = Output: Highest (based off moving average of last 6 readings)
* Input: 51 = Output: Medium (based off moving average of last 6 readings)
* Input: 79 = Output: Medium (based off moving average of last 6 readings)
* Input: 80 = Output: Low (based off moving average of last 6 readings)
* Input: 81 = Output: Low (based off moving average of last 6 readings)
* Input: 84 = Output: Low (based off moving average of last 6 readings)
* Input: 85 = Output: No alert
* Input: 86 = Output: No alert
* Input: 99 = Output: No alert
* Input: 100 = Output: Invalid
* Input: 101 = Output: Invalid

Testing Pressure:

* Systolic
  + Input: 231 = Output: Invalid
  + Input: 230 = Output: Medium alarm
  + Input: 229 = Output: Medium alarm
  + Input: 201 = Output: Medium alarm
  + Input: 200 = Output: Low alarm
  + Input: 199 = Output: Low alarm
  + Input: 151 = Output: Low alarm
  + Input: 150 = Output: None
  + Input: 149 = Output: None
  + Input: 71 = Output: None
  + Input: 70 = Output: None
  + Input: 69 = Output: Medium alarm
  + Input: 51 = Output: Medium alarm
  + Input: 50 = Output: Medium alarm
  + Input: 49 = Output: Highest alarm
  + Input: 1 = Output: Highest alarm
  + Input: 0 = Output: Highest alarm
  + Input: -1 = Output: Invalid
* Diastolic
  + Input: 151 = Output: Invalid
  + Input: 150 = Output: Medium alarm
  + Input: 149 = Output: Medium alarm
  + Input: 121 = Output: Medium alarm
  + Input: 120 = Output: Low alarm
  + Input: 119 = Output: Low alarm
  + Input: 91 = Output: Low alarm
  + Input: 90 = Output: None
  + Input: 89 = Output: None
  + Input: 41 = Output: None
  + Input: 40 = Output: None
  + Input: 39 = Output: Medium alarm
  + Input: 34 = Output: Medium alarm
  + Input: 33 = Output: Medium alarm
  + Input: 32 = Output: Highest alarm
  + Input: 1 = Output: Highest alarm
  + Input: 0 = Output: Highest alarm
  + Input: -1 = Output: Invalid