**Summary:** This project develops a system based on an Android App, Raspberry Pi, and PaaS, to monitor a comatose patient. Crossing of the specified limits/thresholds on patient's vitals (blood pressure and blood sugar) is detected, and appropriate action taken. The actions are also available for the authorized doctor to initiate manually.

**Sensors and actuators:** see the relevant DB table below

**Constraints:**

                -if systolic BP and diastolic BP is greater than x give LED2

                -if systolic BP and diastolic BP is less than y give LED1

                -if blood sugar below z give LED4

                -if blood sugar below w give LED3

**Warnings/alarms:** see the relevant DB table(s) below

**DB:**

                -***DB table of sensors and actuators***

|  |  |  |  |
| --- | --- | --- | --- |
| devID | type | function | ctrl |
| SW1 | SENSOR | sense systolic BP > 140 mm Hg | RPI |
| SW2 | SENSOR | sense diastolic BP < 30 mm Hg | RPI |
| SW3 | SENSOR | sense blood sugar > 120 mg/dL | RPI |
| SW4 | SENSOR | sense blood sugar < 50 mg/dL | RPI |
| LED1 | ACTUATOR | IV infusion of pressor to increase BP | ANDROID |
| LED2 | ACTUATOR | IV infusion of Y to decrease BP | ANDROID |
| LED3 | ACTUATOR | IV infusion of W to decrease blood sugar | ANDROID |
| LED4 | ACTUATOR | IV infusion of Z to increase blood sugar | ANDROID |

-***DB table of alarms (may be avoided)***

|  |  |  |
| --- | --- | --- |
| alarmID | type | msgID |
| 1 | ALARM | 1 |
| 2 | ALARM | 2 |
| 3 | ALARM | 3 |
| 4 | ALARM | 4 |

-***DB table of active alarms***

|  |  |  |
| --- | --- | --- |
| alarmID | sinceTS | ACK |
| 3 | 2020-03-11 14:25:21 | 0 |
| 1 | 2020-03-11 13:17:23 | 1 |

**sinceTS:** Active warning started at

**ACK:** If an active warning was acknowledged it will not be given for the next N minutes

-***DB table of canned messages***

|  |  |  |
| --- | --- | --- |
| msgID | msgType | msgDesc |
| 1 | ALARM | Systolic BP HIGH |
| 2 | ALARM | Diastolic BP LOW |
| 3 | ALARM | BLOOD SUGAR HIGH |
| 4 | ALARM | BLOOD SUGAR LOW |
| 5 | CNST | Diastolic & Systolic BP simultaneously out of range |
| 6 | CNST | ? |
| 7 | SYSTEM | System up |
| 8 | SYSTEM | System shutting down |
| 9 | SYSTEM | User login - success |
| 10 | SYSTEM | User login - failure |
| 11 | SYSTEM | User logout |
| 12 | ADVISE | Reboot system |
| 13 | OPER | IV infusion of pressor |
| 14 | OPER | IV infusion of X |
| 15 | OPER | IV infusion of W |
| . . . | . . . |  |

               -***DB table of transactional logs***

|  |  |  |  |
| --- | --- | --- | --- |
| logID | TS | msgID | data |
| 1 | 2020-03-11 12:23:16 | 7 |  |
| 2 | 2020-03-12 9:14:55 | 9 | ben |
| 3 | 2020-03-12 10:8:37 | 11 | ben |
|  |  |  |  |
|  |  |  |  |

Website:

-password protected

-show charts of transactional logs (e.g. BP and blood sugar w.r.t time)

-update the timestamp when website is updated (auto-refresh using AJAX)

-show status of sensors and actuators

-show alarms (if any) and their status

Android:

              -doc chooses whether to actuate LED1/LED2/LED3/LED4/AUTO

-acknowledge an outstanding alarm

-show the number of times alarms have been activated since monitoring started (e.g. 15)

-show timestamp when status is updated

              -display status of all the sensors & actuators

**Schedule:**

|  |  |  |  |
| --- | --- | --- | --- |
| Task # | Description | Quiz #\* | comments |
| 1 V | Design & create DB tables in PaaS | 6 |  |
| 2 V | Create DB tables in PaaS | 6 |  |
| 3 V | Populate DB tables with "example" data to help in developing other parts of the project, e.g. website, charts, and notification of alarms. | 6 |  |
| 4 V | Implement Android App screen (labels & SwitchCompats) to show status of the sensors & actuators | 6 | Assignment #6 shows how this is done |
| 5 V | Implement sensors (SW1, SW2, . . .) and actuators (LED1, LED2, . . .) on breadboard and test their functionality with RPi ***locally*** | none | Not new... previously done... simply need to update the circuit on the breadboard |
| 6 V | Interface & test the functionality of sensors and actuators between DB tables & RPi | none | Not new... previously done... simply need to update *sync\_rpi\_data.php* and Python program on RPi |
| 7 V | ***Simulate*** the functionality of ***SwitchCompats*** between DB tables & Android using RPi (create a *sync\_android\_data.php*, similar to *sync\_rpi\_data.php*, on 000webhost.com, but trigger it from RPi to get all the bugs out. Later, *sync\_android\_data.php* will be triggered from Android. ***It is important to do this step since later if there is an issue, you just need to look at the Android Kivy side.*** | Q7 | Doing this step will save you a lot of time in debugging the interaction between the Android App and RPi |
| 8 (To Add, even is done) | Program Android App to interact with DB tables in PaaS | Q8 |  |
| 9 (To Add, even is done) | Test interaction between Android App and RPi through DB tables in PaaS | Q8 |  |
| 10 V | Design website | Q7 | should start working on it now... update your current website... |
| 11 V | Interface website components with the DB tables | Q7 | should start working on it now... update your current website... |
| 12 | Collect real data and test the overall functionality of the project | Q9 | debugging & fine-tuning |
| 13 | ? | ? | add tasks as you deem necessary |
| 14 | ? | ? |  |
|  |  |  |  |
| ? | PowerPoint Presentation | Q10 | 30 minute presentation... 15 minute per member... at your convenient time-slot (day/time... does not have to be during the class time) |

\***Note**: *You don't have to wait on different tasks until it is time to submit a quiz. For example, you can start now updating your website and PowerPoint presentation. Your PowerPoint presentation should be substantially completed as soon as Q9 is submitted because your project's last technical task is Q9. There is more time allotted to Q7 and Q8 because most of the technical details are taken care of in these quizzes. Q9 is basically for collecting real data and debugging/tuning.*

*Approximate deadlines for Quizzes:*

*Q7  Friday June 26*

*Q8  Monday June 29*

*Q9  Wednesday July 1*

*Q10  Thursday July 2 (flexible... you choose a convenient day/time to do your 30-minute presentation... does not have to be during the class time... but must be before July 7 which is the exam day... let me know by July 1)*