PoS Baby

Testing Strategy

Tests

Milestone I

Components

LCD Screen - Test Case 1
This is used to output the instructions for the user as well as the points system, which includes the current score and the total score.
Inputs: String that is programmed
Expected output: String specified above
Type of test: Performance test
Requirements: N/A

Speaker - Test Case 2
This is used to output whatever noise the baby will make, such as crying or bad jokes. In this case, we just want to know that it works so we will use pwm so that the speaker emits noise.
Input: Frequency (programmed in)
Expected output: Frequency specified above
Type of test: Performance test
Requirements: N/A

Amplifier - Test Case 3
This is used to increase or decrease the volume of the speaker.
Input: Knob on the side of amp
Expected output: Adjustment of the speaker's volume
Type of test: Integration test
Requirements: Speaker works

Touch sensor - Test Case 4
The touch sensor is used to detect the user's finger for poking the baby's eyes.
Input: User's finger
Expected output: Speaker outputs noise and LCD screen outputs text
Type of test: Integration test
Requirements: Speaker works and LCD works

Motion sensor - Test Case 5
The motion sensor is used to detect movement. In this case, it will detect if the user is detaching the baby's arm.
Input: User's hand
Expected output: Speaker outputs noise or LCD screen outputs text
Type of test: Integration test
Requirements: Speaker works or LCD works

Motion sensor - Mocking Test Case

We won't be placing the components into the baby yet, so we will mock the baby's arm by placing the motion sensor inside a tube-like object, like a toilet paper roll. To mimic the user detaching the arm, we move the tube-like object away from the sensor.
Input: A tube-like object that the sensor can hide inside of (instead of the baby)
Expected output: Speaker outputs noise and LCD screen outputs text

Arduino Uno - Test Case 6

LED light blinks faster with lowering periods.
Input: (1000-200) ms
Expected output: LED light blinks more quickly with lowering periods
Type of test: Performance test
Requirements: LED lights and 300 Ohm resistor

Integration

Test cases 4 and 5 are essentially part of this subsection. This is because when we want to test if the touch sensor or motion sensor works, we need another component to let us know. In both cases, it is the speakers and LCD screen. Thus, if we were to test the touch sensor by touching it, we would know if it is working correctly if the speakers emit noise and the LCD screen changes to output something like "The touch sensors work!". Thus, test cases 4 and 5 will show that the entire system (which consists of the LCD screen, speaker, amp, touch sensor, and motion sensor in this milestone) works when we put them all together. But, we should do the following test case below to ensure that all components work together.

Integration Test Case

In this test case, the user should touch the touch sensor and then adjust the volume using the knob on the amplifier. The speaker's volume should change accordingly and the LCD screen should display: "Touch sensor is working!". Then, they should wave their hand close to the motion sensor and repeat with the knob on the amp. The speaker's volume should change accordingly and the LCD screen should display: "Motion sensor is working!".
Input: User's hand, touch sensor, motion sensor, amp knob
Expected output: Speaker emits noise and LCD screen changes output accordingly
Requirements: All components must work

Milestone II

Components

Alcohol Sensor - Test Case 1

The alcohol sensor is used to detect the presence of alcohol.
Inputs: Fluid of various types
Expected output: Correct reading of alcohol detected
Mocks Needed: n/a
Type of test: Stress Test
Requirements: The alcohol sensor must be able to correctly detect whether or not alcohol is detected within a given fluid.

Gyroscope Sensor - Test Case 2

The Gyroscope Sensor detects rotational movement
Inputs: Rotational movement
Expected output: Correct reading of movement.
Mocks Needed: n/a
Type of test: Performance Test
Requirements: The gyroscope sensor must detect rotational movement and correct data that matches the rotational movement it has detected.

Speaker (custom noise) - Test Case 3

The speaker outputs crying sounds, profanity, etc.
Mocks Needed: n/a
Type of Test: n/a
Type of test: Performance Test
Requirements: The speaker must be able to output custom, pre recorded sounds such as crying, screaming, and profanity.

Integration

The alcohol sensors will be placed within the dolls mouth and will run in a concurrent SynchSM in which it will become active whenever its function is called. The gyroscope will be placed within the doll's head and will also run within a SynchSM in conjunction with the other components, only becoming active when its function is called. The speaker must be able to interact with the other components within the system, and so must be integrated into the SynchSM that the rest of the system will be running on. Upon certain sensors activating, such as the gyroscope sensor, the Speaker must emit the correct noise, such as screaming. The speaker is what will be used to make sure that the rest of the components are functioning correctly and according to spec.

Integration Test Case
Upon a full rotation of the gyroscope sensor, or the motion sensor detects motion, the speaker should emit a pre recorded scream made by the baby. When the alcohol sensor is triggered, the speaker should emit a pre recorded burp sound. When the touch sensor is activated, the speaker should emit a pre recorded profanity (up to programmers discretion).

Final Demo

Requirements

Integration of all components and inserted into the doll. Every component should work as designed when used as part of the baby.

LCD Screen embedded into the forehead of the baby and should display the next command.

Mistakes are actions the user can take that are not demanded by the baby (command displayed on the LCD Screen). If the user makes a mistake then the baby should become frustrated, as simulated by the loudening of the speaker. On the other hand, if the user takes the correct course of action, the LCD should display the next command and the baby should become quieter, as if being soothed.

Speaker should output the sounds of a baby crying when the puzzle begins and amps (as part of the Amplifier functionality) up as the timer dwindles or the user makes mistakes and it should make awful jokes when the user makes mistakes as well. The speaker is to be embedded in the throat or chest cavity of the baby, in order to mimic a realistic/intuitive feeling baby.

Amplifier should also turn down the sound as the user soothes the baby with brute force. The amplifier will be placed near the speaker (an inch apart) in the throat or chest cavity as well.

Command: “Baby Tired”
Touch sensors go into the eyes of the baby which will have to be carved in order to place the sensors into them. The sensors will also have to be mounted with duct tape. Functionally, they should detect the user’s thumbs as the user presses their thumbs into the baby’s eyes, just like mother used to do.

Command: “Baby Sore”
Motion sensors will go in the arms and legs of the doll and will be mounted near the detach point. Their function is solely to detect when an arm or leg has been either removed or inserted.

Command: “Baby Anxious”
Breathalyzer will be embedded into the back of the head of the baby, jutting out. This way, the user can hold the baby close to them, facing away from them and can breathe into the sensor like a reverse bong. The sensor should detect whether the user has had an adult beverage.

Command: “Baby $&#!”
Gyroscope Sensor will be placed within the head/neck where it should detect the rotation of the baby's head. This component will be hot glued to the baby itself around the base of the neck to detect the motion.

Command: “Baby Thirsty”
Alcohol Sensor will be placed into the mouth of the baby and should notify the system when alcohol is poured into the mouth.

After X amount of correct moves, the puzzle will be solved (demonstrated by the baby’s silence).

Components

LCD Screen - Test Case 1
This is used to output the instructions for the user as well as the points system, which includes the current score and the total score.
Inputs: String that is programmed
Expected output: String specified above
Type of test: Performance test
Requirements: N/A

Speaker - Test Case 2
This is used to output whatever noise the baby will make, such as crying or bad jokes. In this case, we just want to know that it works so we will use pwm so that the speaker emits noise.
Input: Frequency (programmed in)
Expected output: Frequency specified above
Type of test: Performance test
Requirements: N/A

Amplifier - Test Case 3
This is used to increase or decrease the volume of the speaker.
Input: Knob on the side of amp
Expected output: Adjustment of the speaker's volume
Type of test: Integration test
Requirements: Speaker works

Touch sensor - Test Case 4
The touch sensor is used to detect the user's finger for poking the baby's eyes.
Input: User's finger
Expected output: Speaker outputs noise and LCD screen outputs text
Type of test: Integration test
Requirements: Speaker works and LCD works

Motion sensor - Test Case 5

The motion sensor is used to detect movement. In this case, it will detect if the user is detaching the baby's arm.
Input: User's hand
Expected output: Speaker outputs noise or LCD screen outputs text
Type of test: Integration test
Requirements: Speaker works or LCD works

Alcohol Sensor - Test Case 1

The alcohol sensor is used to detect the presence of alcohol.
Inputs: Fluid of various types
Expected output: Correct reading of alcohol detected
Mocks Needed: n/a
Type of test: Performance Test
Requirements: The alcohol sensor must be able to correctly detect whether or not alcohol is detected within a given fluid.

Gyroscope Sensor - Test Case 2

The Gyroscope Sensor detects rotational movement
Inputs: Rotational movement
Expected output: Correct reading of movement.
Mocks Needed: n/a
Type of test: Performance Test
Requirements: The gyroscope sensor must detect rotational movement and correct data that matches the rotational movement it has detected.

Integration (System)

The Doll - Test Case 1

Eye gouge
Inputs: press on the eyes
Expected output: LCD Screen display to confirm touch has been detected
Mocks Needed: the completed baby
Type of test: Stress Test
Requirements: The baby’s integrity should not be compromised and a correct reading should be output.

Head Spin - Test Case 2

Gyroscope sensor detects the rotation of the PoSB’s head.
Inputs: Spin the head of the doll.
Expected output: LCD screen displays “Successful!”
Types of test: Integration
Requirements: Completion of test numbers 1.1 and 2.2.

Detach Limb - Test case 3

Motion sensor detects the detachment of the arm.

Inputs: User detaches the arm.
Expected output: Speaker volume should decrease while the LCD screen displays “Successful!”
Types of test: Integration
Requirements: Completion of test numbers 1.1 and 1.5 as well as the detachment of the limb of the PoSB.

Feed Alcohol - Test case 4
Liquid containing alcohol is given to the PoSB.
Inputs: Liquid containing alcohol
Expected Output: LCD screen displays “Successful!”
Types of test: Integration
Requirements: Completion of test numbers 1.1 and 2.1.

Complete Puzzle (Fail) - Test case 5
Solve the puzzle incorrectly.
Inputs: [Incorrect order of required action]
Expected output: Speaker should remain playing crying baby noises and LCD screen displays “Failed! Try again!”.
Types of test: Reliability
Requirements: Completion of previous tests.

Complete Puzzle (Success) - Test case 6
Solve the puzzle correctly.
Inputs: [Correct order of required action]
Expected output: Speaker should turn off and the LCD screen should display “Puzzle Solved!”
Types of test: Integration
Requirements: Completion of previous tests.

[Note: Custom changes in the anatomy of the doll are required for all test cases under integration (system).]
Integration (Doll and Components)
The back of the baby will be torn open in order for us to enter our components. The base, or breadboard in this case, will be entered into the Torso of the baby. Wires will then extend out to the Eyes of the baby (Touch Sensors) to the Forehead (LCD Screen)

and to the Back of Head (Alcohol sensor). We will also have a Gyroscope sensor in the head of the baby, a speaker and amplifier in the chest cavity, and motion sensors in the arms and legs.

The LCD should display the command and on a correct attempt, switch to a different command whilst also quieting the baby by 1 level. If an incorrect input is given, the baby will instead get louder by 1 level - at this point the LCD will not change. The LCD will keep the same command prompt until the correct task is completed.

The baby will also have to undergo stress testing. The game should be played several times using maximum force to ensure the baby holds together. It should be shaken, yanked, slapped, and gouged to insure structural integrity of the baby.

[Note: Each test, except 1.6, assumes the completion of test number 1.6.]

Automation

Components: For the components, we will use performance and stress tests to confirm their workability and gauge their accuracy. We will connect the Arduino Uno into the Arduino IDE and run the IDE while testing individual components. The strategy will be similar to black box testing where we have various inputs and expected outputs. We will use simple tester functions to gauge a reading of the component. This will be done for each individual component and the process for each should be similar, and since we will be testing them individually with separate tester functions, we can be rest assured that the result or the process of each test will not affect the outcome of the others. Since testing the components should be straight-forward we won't need to apply any testing framework for this part of the testing process.

Integration: Once we pass the performance tests, to ensure the workability of our components, we can proceed to do integration tests that combine the functionalities of multiple components. For this portion of the process we will use the regression testing framework to ensure that each change we make does not affect other functionalities that we have previously integrated in the system and if we detect any bugs we should be able to quickly identify the problem and fix it. To further ensure the workability of our system, it will be important to keep track of the data flow of our program to help us find bugs more quickly in our code. We will keep track of our tests through a spreadsheet for better organization. This spreadsheet will continuously be updated as we progress towards the completion of our project. The first test number in the spreadsheet indicates which part of our milestone the test is being conducted and the second part is used to identify the tests, this was done so we can quickly identify which portion of the process the test is being conducted in. The link for the spreadsheet is below.

Test Strategy Spreadsheet:
https://docs.google.com/spreadsheets/d/1jRNkEPkvqBENGNxnG9g6oYIPKOP_SmrFcCzFks8mpgI/edit#gid=0