EXPLORATION

WALT: definition and decomposition of complex problems in terms of functional and non-functional requirements

WILF
- Defined how each component works by identifying or describing their qualities.
- Identified where and why components are connected into the GPIO.
- Described how the circuit works and the relationships between components.

Difficulty: easy
Estimated Effort: 70 Mins
Value: 10%
Components List:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processes and production skills identifying and defining</td>
<td>Effective definition and decomposition of complex problems in terms of functional and non-functional requirements</td>
<td>Definition and decomposition of complex problems in terms of functional and non-functional requirements</td>
</tr>
<tr>
<td>- Defined how all components work by identifying or describing their qualities.</td>
<td>- Defined how most components work by identifying or describing their qualities.</td>
<td>- Defined how some components work by identifying or describing their qualities.</td>
</tr>
<tr>
<td>- Identified why and where all of the components are connected into the GPIO.</td>
<td>- Identified why and where most of the components are connected into the GPIO.</td>
<td>- Identified why and where some of the components are connected into the GPIO.</td>
</tr>
<tr>
<td>- Described how the whole circuit works, with both text and images.</td>
<td>- Described how the circuit works, with both text and images.</td>
<td>- Described how the circuit works.</td>
</tr>
</tbody>
</table>

Components List:

- Arduino Board
- USB Cable
- LED
- Touch Sensor

Touch Sensor
This digital capacitive touch sensor provides a one-touch style switch on your Arduino project. It uses the most popular capacitive sensing technology which is the same as your mobile phone.
This little sensor can “feel” people and metal touch and feedback a high/low voltage level. Even isolated by cloth or paper, it still can feel the touch. The sensitivity will decrease as the isolation getting thick.

NOTE: The Digital cable plugs into the Touch Sensor and is colour coded:

- Black wire = anode - or GRD
- Red wire = cathode + or 5V supply
- Green wire = input to board via a digital pin

The Light Emitting Diode

The LED is a very common, infinitely useful component that converts electrical current into light. LEDs come in various shapes, sizes, and colors. Figure 3-6 shows a common LED.

Connecting LEDs in a circuit takes some care, because they are polarized; this means that current can enter and leave the LED in one direction only. The current enters via the anode (positive) side and leaves via the cathode (negative) side, as shown in Figure 3-7. Any attempt to make too much current flow through an LED in the opposite direction will break the component. Thankfully, LEDs are designed so that you can tell which end is which. The leg on the anode side is longer, and the rim at the base of the LED is flat on the cathode side, as shown in Figure 3-8.

5 SIMPLE WAYS TO DETERMINE LED POLARITY

When adding LEDs to a project, you need to consider the operating voltage and current. For example, common red LEDs require around 1.7 V and 5 to 20 mA of current. This presents a slight problem for us, because the Arduino outputs a set 5 V and a much higher current. Luckily, we can use a current-limiting resistor to reduce the current flow into an LED.

NOTE: You can use the LED without a resistor if you code an analogWrite() below 100. eg analogWrite(6,100). A value of 255 would give you the full 5V and burn out the LED

Want More?
http://www.circuitbasics.com/arduino-basics-controlling-led/

Pins used: GND, GND, 5V, 6

The Touch Sensor is powered via the GND (black wire, anode -) and 5V (red wire, cathode +) pins.
The board receives a digital input signal from the touch sensor, when it is touched, via pin 3.
The LED is powered via pin 6

The Circuit:

The powered touch sensor is triggered by touch and sends a digital input signal to the board via pin 3. If touched, a signal is passed to the LED via pin 6

INSTRUCTIONS

1. Connect as shown in the Fritzing (circuit) diagram above. Take a photo for your portfolio, or set up a virtual circuit in tinkercad and take a screenshot.
2. Answer these questions:
Portfolio Task Questions for EXPLORE

1. Define how each of the components work? Identify each component and describe how they work with text and images.
2. Identify what and where components are plugged into the GPIO. Why there?
3. Describe How the circuit works? Tell the story. Include a picture.

[ use https://www.tinkercad.com/ ]

DEVELOP

WALT: design and evaluation of user experiences and algorithms

WILF

Design
- interpret existing code:
  > Analyse to identify where input, output, processing and storage occurs in the code.
  > translated code into pseudocode
- Plan to modify parts of the code to make it work differently or more effectively

Evaluation
- Recommended ways to use the technology differently.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processes and production skills</td>
<td>Processes and production skills</td>
<td>Processes and production skills</td>
</tr>
<tr>
<td>Generating and designing - producing and implementing</td>
<td>Generating and designing - producing and implementing</td>
<td>Generating and designing - producing and implementing</td>
</tr>
<tr>
<td>design and evaluation of user experiences and algorithms</td>
<td>design and evaluation of user experiences and algorithms</td>
<td>design and evaluation of user experiences and algorithms</td>
</tr>
<tr>
<td>- identified where input, output, processing and storage occurs in the code.</td>
<td>- identified where input, output, processing and storage occurs in the code.</td>
<td>- identified where input, output, processing and storage occurs in the code.</td>
</tr>
<tr>
<td>- translated all code into pseudocode</td>
<td>- translated most code into pseudocode</td>
<td>- translated some code into pseudocode</td>
</tr>
<tr>
<td>- Plan to modify several parts of the code to make it work differently</td>
<td>- Plan to modify some parts of the code to make it work differently</td>
<td>- Plan to modify some parts of the code to make it work differently</td>
</tr>
<tr>
<td>- Recommended ways to use the technology differently.</td>
<td>- Recommended ways to use the technology differently.</td>
<td>- Recommended ways to use the technology differently.</td>
</tr>
</tbody>
</table>

Algorithms: Sequence, selection

Code structure, values and functions:

```plaintext
int [Data Types]

To store a value in code, we use variables. There are many kinds of values or types of data that we may want to store.
Integers are your primary data type for number storage

```

```plaintext
pinMode() [Digital I/O]

Configures the specified pin to behave either as an input or an output

Syntax
pinMode(pin, mode)

Parameters
pin: the number of the pin whose mode you wish to set
mode: INPUT, OUTPUT, or INPUT_PULLUP. (see the digital pins page for a more complete description of the functionality.)

```

```plaintext
digitalRead() [Digital I/O]

Description
Reads the value from a specified digital pin, either HIGH or LOW.

Syntax
digitalRead(pin)
```
Parameters
pin: the number of the digital pin you want to read

Returns
HIGH or LOW


analogWrite() [Analog I/O]

Description
Writes an analog value (PWM wave) to a pin. Can be used to light a LED at varying brightnesses or drive a motor at various speeds. After a call to analogWrite(), the pin will generate a steady square wave of the specified duty cycle until the next call to analogWrite() (or a call to digitalRead() or digitalWrite() on the same pin. The frequency of the PWM signal on most pins is approximately 490 Hz. On the Uno and similar boards, pins 5 and 6 have a frequency of approximately 980 Hz. On most Arduino boards (those with the ATmega168 or ATmega328P), this function works on pins 3, 5, 6, 9, 10, and 11.

You do not need to call pinMode() to set the pin as an output before calling analogWrite(). The analogWrite function has nothing to do with the analog pins or the analogRead function.

Syntax
analogWrite(pin, value)

Parameters
pin: the pin to write to. Allowed data types: int.
value: the duty cycle: between 0 (always off) and 255 (always on). Allowed data types: int


delay() [Time]

Description
Pauses the program for the amount of time (in milliseconds) specified as parameter. (There are 1000 milliseconds in a second.)

Syntax
delay(ms)

Parameters
ms: the number of milliseconds to pause (unsigned long)


If...else [Control Structure]

Description
The if statement checks for a condition and executes the proceeding statement or set of statements if the condition is 'true'.

Syntax
if (condition) {
  //statement(s)
}

Parameters
condition: a boolean expression i.e., can be true or false

The statements being evaluated inside the parentheses require the use of one or more operators shown below.
Comparison Operators:
- x == y (x is equal to y)
- x != y (x is not equal to y)
- x < y (x is less than y)
- x > y (x is greater than y)
- x <= y (x is less than or equal to y)
- x >= y (x is greater than or equal to y)


See Item 6 for analogWrite()

Use of if statements

```cpp
if(digitalRead(touchSensor)==HIGH){       //Read Touch sensor signal and if 5V is passing through pin 3 then flash the LED for a second.
  analogWrite(led, 100);
  delay(1000);
}
```

digitalRead() works just the same as digitalWrite(), only it is an input from the pin this time.

### INSTRUCTIONS

1. Save the sketch (code) below and then open in the Arduino IDE (File>Open..)

2. Copy the code into a place where you can edit it to add comments.

3. Answer the following:

#### Portfolio Task Questions for DEVELOP

1. Where are the Input, output, storage in the code?

<table>
<thead>
<tr>
<th>Feature</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. How does the code work - pseudocode

   a. Copy the existing code

   b. Place comments (//) next to each line of code in pseudocode

3. How will you modify the circuit and/or the code to make it do something different or more effectively?

4. How could you use this technology in another way to improve the way it works or apply it to a different situation?

#### How to write pseudocode

<table>
<thead>
<tr>
<th>Operation</th>
<th>Pseudocode</th>
<th>Programming equivalent</th>
</tr>
</thead>
</table>
| Accepting inputs | Prompt user for surname
Input surname | var person = prompt("Please enter your name"); |
| Producing outputs | Print the message Hello World | console.log("Hello World"); |
| Assigning values to variables | Set the total to 0 | var total = 0; |
| Performing arithmetic | Set total to items times price | var total = items * price; |
| Selection and Perform alternative actions | IF mark is greater than 49 THEN
print Pass
ELSE
print Fail | if (mark > 49) {
console.log("Pass");
} else {
console.log("Fail");
} |
| Iteration or Repeating operations | WHILE total greater than 0
subtract new purchase from total
ENDWHILE | while (total > 0) {
  total = total - new_purchase;
} |
| Iteration or Repeating operations | REPEAT
<Block>
UNTIL condition | var i = 0;
do{
  text += "The number is "+ i;
  i++;
} while (i < 5); |
| Iteration or Repeating operations | FOR Index = START_VALUE to FINISH_VALUE
<Block>
ENDFOR | var i = 0;
for (i = 0; i < cars.length; i++) {
  text += "<br>
  text += cars[i] + "\n";
} |
WALT: design and implementation of modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities

WILF
- Code implemented

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>The student work has the following characteristics:</td>
<td>The student work has the following characteristics:</td>
<td>The student work has the following characteristics:</td>
</tr>
<tr>
<td>comprehensive design and proficient implementation of modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities</td>
<td>effective design and effective implementation of modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities</td>
<td>design and implementation of modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities</td>
</tr>
<tr>
<td>- All code implemented</td>
<td>- Most code implemented</td>
<td>- Some code implemented</td>
</tr>
</tbody>
</table>

INSTRUCTIONS

1. Connect as shown in the Fritzing (circuit) diagram above. Take a photo for your portfolio, or set up a virtual circuit in tinkercad and take a screenshot.
2. Test your board with the ‘Blink’ sketch
3. Save the sketch (code) below and then open in the Arduino IDE (File>Open...)
4. Double-check that you have the right board (Tools>Board) and COM port (Tools>Port) selected.
5. Upload your sketch (Sketch>Upload or ...
6. You should hear a ‘ticking’ sound coming from the piezo buzzer and the light gradually fading.
7. Copy and paste the code to your portfolio. Read and translate the code into pseudocode by commenting (add ‘//’ at the end of each line) each line of code. Eg:

```
pinMode(11, OUTPUT);                     // set pin 11 as an output pin
```

8. Modify the code to make it work differently or more effectively.
   Eg.
   Change the delays
   On/off switch:
   ```
   void loop(){
    //Read button state (pressed or not pressed?)
    buttonState = digitalRead(buttonPin);
    
    //If button pressed...
    if (buttonState == LOW) {
        //...ones, turn led on!
        if ( flag == 0){
            digitalWrite(ledPin, HIGH);
            flag=1; //change flag variable
        }
        //...twice, turn led off!
        else if ( flag == 1){
            digitalWrite(ledPin, LOW);
            flag=0; //change flag variable again
        }
    }
    
    // OR Fade
    
    if(led_pin == 9){ // the PWM pin the LED is attached to
        brightness += 1; // how bright the LED is
        fadeAmount = 5; // how many points to fade the LED by
        
        // the setup routine runs once when you press reset:
        void setup() {
            // declare pin 9 to be an output:
            pinMode(led, OUTPUT);
        }
        
        // the loop routine runs over and over again forever:
    }
```
void loop() {
// set the brightness of pin 9:
analogWrite(led, brightness);

// change the brightness for next time through the loop:
brightness = brightness + fadeAmount;

// reverse the direction of the fading at the ends of the fade:
if (brightness <= 0 || brightness >= 255) {
  fadeAmount = -fadeAmount;
}

// wait for 30 milliseconds to see the dimming effect
delay(30);
}

From <https://www.arduino.cc/en/tutorial/fade>
From <https://www.electroschematics.com/8964/turn-on-led-button-arduino/>

9. Re-upload your code
10. Take a video for your portfolio.
11. Answer the questions below:

Portfolio Task Questions for GENERATE
1. Get the hardware and software working
2. Get your modifications working and highlight where you have altered the code. document
3. Record a very short video of your working solution.

Evaluate and Refine
WALT: evaluation of information systems and their solutions in terms of risk, sustainability and potential for innovation and enterprise

WILF - Makes judgments about ideas, works, solutions or methods in relation to risk, sustainability and potential for innovation and enterprise

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processes and production skills</td>
<td>The student work has the following characteristics:</td>
<td>The student work has the following characteristics:</td>
</tr>
<tr>
<td>Evaluating</td>
<td>- Discerning evaluation of information systems and their solutions in terms of risk, sustainability and potential for innovation and enterprise</td>
<td>- Informed evaluation of information systems and their solutions in terms of risk, sustainability and potential for innovation and enterprise</td>
</tr>
<tr>
<td></td>
<td>- All sections of evaluation completed and thorough.</td>
<td>- Most sections of evaluation completed and thorough.</td>
</tr>
</tbody>
</table>

Portfolio Task Questions for EVALUATE AND REFINE
1. Update your kanban and burndown chart
2. Write a short evaluation

Evaluation
Critically evaluate your completed digital solution by using the organiser below. Make sure that you use full sentences and copy and paste text into paragraphs rather than leaving it in table form.
<table>
<thead>
<tr>
<th>Digital solution evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise needs and opportunities</td>
</tr>
<tr>
<td>What needs or opportunities does the solution address?</td>
</tr>
<tr>
<td><strong>Risks</strong></td>
</tr>
<tr>
<td>What risks does the digital solution pose to the user’s personal security?</td>
</tr>
<tr>
<td>Could the digital solution have any adverse effects on the stakeholders?</td>
</tr>
<tr>
<td><strong>Sustainability</strong></td>
</tr>
<tr>
<td>How could the digital solution impact the environment?</td>
</tr>
<tr>
<td>What economic factors might influence the digital solution?</td>
</tr>
<tr>
<td>Is your solution easy to use and learn? Why/Why not?</td>
</tr>
<tr>
<td>Are there any social factors which could affect the solution?</td>
</tr>
<tr>
<td><strong>Innovative</strong></td>
</tr>
<tr>
<td>How is the digital solution innovative?</td>
</tr>
<tr>
<td><strong>Recommendations</strong></td>
</tr>
<tr>
<td>Recommend at least one improvement that you would like to see made to the digital solution.</td>
</tr>
</tbody>
</table>

More?