SHARD

Structural Health and Rupture Detection

Team Members:

Julian Herrera, jherrera2020@my.fit.edu

Matthew Manley, mmanley2018@my.fit.edu

Matthew Meesit, mmeesit2019@my.fit.edu

John Bruce, jbruce2020@my.fit.edu

Paul Awad, pawad2021@my.fit.edu

CSE Faculty Advisor:

Dr. Silaghi, msilaghi@fit.edu

Client:

Aerospace and Mechanical Engineer Senior Design GSA: Felix

Client Meeting Dates:

Every Friday at 3pm

Goal and Motivation:

The overall goal of the project is to provide the client with a system that protects against small debris in low earth orbit. Our goal is to provide the system with an interface between the human and the system. More specifically, to design and develop an application that utilizes specific data obtained from sensors attached to the outside structure (tiles), to display said data and tiles in a user friendly way, and to program a microcontroller to activate a mechanism that autonomously repairs the system when the data is not acceptable.

Approach:

Visualize the physical model of the structure. In the application, the user will be able to create various model templates of a structure that is representative of what it should and does look like in the real world. Having the user create the model instead of choosing from a selection allows the user to have modularity when it comes to designing and visualizing the system. The user will be able to see a model of the system that they are a part of, making life easier when a rupture occurs because the user will know where it happened.

Interact with real time data that is fed into the application from outside sensors. Interaction of the data is key when it comes to the system because the user must know what is happening around the structure in order to be made aware of the situation outside. And when a breach occurs, the user will want to know where that happens. The real time data is visualized through the model on the application in a way that is intuitive.

Choose what to do when sensor data reveals a breach in the structure. When a breach occurs, the user will want to take necessary actions in order to save his or her life. From the application, the user can take the necessary steps to ensure safety.

Novel features/functionalities:

None

Technical Challenges

Collecting data from the sensors and turning it into understandable information
Knowing which tile in the physical model corresponds to a tile in the virtual model. This will be a challenge because all that the sensors will be providing us is with the pressure applied to it (we assume).
Learning more about microcontrollers (or other electronics that interface with sensors) and how they interact with computers. We have never interfaced with sensors before.

Milestone 1 (Oct 3): itemized tasks:

1. Compare and select technical tools for which microcontroller we will use to interface with the sensors and the application. The different microcontrollers are: PIC, Arduino, and Raspberry PI. Compare and select technical tools for which programming language to use for the GUI.
2. Provide small ("hello world") demo(s) to evaluate the tools for the microcontroller and the programming language.
3. Resolve technical challenges:
    a. “collecting data from sensors and turning it into understandable information” by finishing the above “hello world” demo(s) and hooking up sensors to the microcontroller.
    b. doing research and making practical programs to test.
4. Compare and select collaboration tools for software development, documents/presentations, communication, task calendar
5. Create Requirement Document
6. Create Design Document
7. Create Test Plan

Milestone 2 (Oct 31): itemized tasks:

1. Implement, test, and demo the user interface, which should be able to create a template/blueprint of a model that is to be used for “connecting” the tiles (by using the microcontroller and sensor data).
2. Implement, test, and demo the microcontroller, which should be able to be connected to the application.

Milestone 3 (Nov 28): itemized tasks:

1. Implement, test, and demo the application and the microcontroller with the sensors. It should be able to be connected to the application and the application should show the sensor data for each tile in the template.
2. Implement, test, and demo that the application should at the very least tell the user if a rupture happens. Perhaps if the rest of the team has a system that can repair the rupture, we can tell that system to go ahead and repair it when the rupture happens.

Task matrix for Milestone 1

Task	Julian	Matthew
Compare and select Technical Tools	Programming language for GUI	Arduino vs Raspberry PI vs PIC
"hello world" demos	Programming language for GUI	Arduino vs Raspberry PI vs PIC
Resolve Technical Challenges	Test programs	Microcontroller research
Compare and select Collaboration Tools	Version Control System	Communication
Requirement Document	50%	50%
Design Document	25%	75%
Test Plan	75%	25%

Approval from Faculty Advisor

a. "I have discussed with the team and approve this project plan. I will evaluate the progress and assign a grade for each of the three milestones."
b. Signature: _____ Date: _____