May 20, 2021
Jack Woods
Rami Hansen
Sachin Betrabet

Executive Summary

The purpose of this project is to design and build a high resolution biometric monitor for use in evaluating pilot stress levels while flying. Current biometric devices are typically “fitness band” products that do not integrate with existing avionics solutions. Scratch building an LED-based heart rate monitor (and accompanying software systems) will enable advanced computerized avionics systems to monitor pilot stress levels in real time, helping pilots more effectively manage in-cockpit automation (such as autopilots, autothrottles, pressurization systems, and control augmentation systems). Aviation authorities, pilots, FAR Part 121 common air carriers, aviation military organizations, and passengers will benefit from a safer, more integrated, cockpit environment.

Team Approach

Through one week-long sprints, our software and hardware teams delivered product designs and project documentation during fall quarter, implemented and proved individual system blocks during winter quarter, and implemented the entire system during spring quarter. Throughout the entire process, agile practices and adherence to project specifications enabled our team to deliver portions of a minimum viable product through a continuous improvement cycle. In the end, an affordable and reliable system that monitors and classifies a pilot’s state in real time was created. The system utilizes a hardware device to collect and transmit biometric data to a software system over a Bluetooth connection in order to make predictions about stressed and unstressed states of the wearer. In addition to making pilot state classifications, the software system provides real time data visualization through a browser accessible web interface as well as the ability to export raw recorded flight data.

Project Timeline

Deadline	Description	Involved
11/26/20	Final CS block diagram	All
11/20/20	Recorded ECG data available	ECE sub team
12/10/20	Initial prototype PQRST classifier functional	Rami
12/10/20	Initial prototype data interpreter	Sachin
12/10/20	Initial prototype data visualization	Jack
2/1/21	PQRST classifier final	Rami
2/1/21	Data interpreter final	Sachin

2/1/21	Data visualization final	Jack
2/8/21	Begin software integration testing	CS sub team
2/8/21	Begin hardware integration testing	ECE sub team
3/29/21	Spring term start - Pilot classification system works	All
Spring 21	Engineering Expo - Pilot classification system finalized	All

Lessons Learned

As with any long term project, choices the team made in the early stages of the project were later shown to be poor ones as new information arose. Several of the team's biggest headaches could have been reduced or outright avoided had we made different decisions early on. While frustrating to deal with, these issues became important lessons that we can take with us when we tackle our next challenges.

Reliable test data is essential for testing each block in the system. Our group relied on templating code and making assumptions about what data would appear in recorded biometric samples. Because of this, sweeping changes had to be made throughout the code base when a final sample format was decided. This could have been avoided by deciding on a data type and obtaining test data earlier in the project. Additionally, the biometric data collection device that the group chose for collecting sample data was less accurate than desired. Had the team defined a concrete sample type early and compared data with a medical grade biometric device, the end product could have been improved greatly and the development time could have been reduced.

Our team developed software blocks mostly in isolation. This ultimately resulted in unplanned hours spent on rewriting certain components so they would be compatible. Integration testing also revealed shortcomings that were not apparent through exploratory unit testing. The nature of the class places an emphasis on individual work during the early and mid stages of the class before the group must work together to integrate each individual piece into the final system. Making a greater effort to develop blocks in tandem with other blocks could have reduced the time spent fixing errors that arose during final system integration.

In addition to problems with integration, the group had some issues concerning unfamiliarity with software technologies. C++ was chosen as a primary language for the implementation of the system. While this was a logical choice as far as system performance was concerned, it did limit the contributions of team members who were less familiar with the language and its practices. Had the group spent more time weighing the different technology options available, a better choice could have been made.