Flashcard System

Group 2:
Yicun Liu
Hua Chuang
Ryan Harmon

10/3/2019

2. Summary

One of the most effective ways of remembering information is through the combination of flashcards and a spaced repetition reviewing system. While it is possible to create and review flashcards by hand, they are time-consuming to make, prone to get lost, easy to mix up, and very difficult to systematically review in an efficient manner. In this proposal, we lay out a plan for a simple Java app which can be used to create, review and manage generic flash cards.

3. Why this work needs to be done

Flashcards are a very effective way to memorize vocabulary words, dates, formulae, or any variety of other facts. In its most basic physical form, a flashcard is simply a small piece of paper with a question written on one side and an answer to be memorized on the other. A user looks at the question, guesses the answer, and then flips the card to see if they are right.

This system, while simple for a small number of cards, is unwieldy when the number grows large. For this, a computer application can very effectively organize and manage a set of cards, allowing a user to spend more time learning and less time on hassle.

While a user may choose to simply review all of their flashcards periodically or even arbitrarily, it is much more efficient to use a spaced repetition system. Spaced repetition, proposed back in 1932 by British psychologist C. A. Mace[1], suggests that new flashcards should be shown more frequently to a learner, while older cards should gradually be show less often, until just the point that the learner is about to forget them. This method is much more efficient as less reviews are needed than a monotonic system and has been shown to even increase the rate of learning[2].

Computer applications make spaced repetition algorithms much easier to manage and totally transparent to a user, who may simply open a flashcard application, review today's cards and move on.

As flashcards are used in many applications, from students cramming for exams to people who want to learn another language, we believe it is extremely beneficial to implement a simple computer program which will handle the backend fuss of flashcard management and scheduling, while providing a flexible interface for flashcard creation and review.

1. Mace, C. A. (1932). Psychology of Study. p. 39
2. Smolen, Paul; Zhang, Yili; Byrne, John H. (January 25, 2016). "The right time to learn: mechanisms and optimization of spaced learning". Nature Reviews Neuroscience. 17 (2): 77–88. arXiv:1606.08370. Bibcode:2016arXiv160608370S. doi:10.1038/nrn.2015.18. PMC 5126970. PMID 26806627.

4. Objectives

This project aims to create a one-stop solution for a user to create flashcards, edit them and review them. The following specifications have been generated to accomplish this, broken down categorically:

Main Interface

- The flashcard system, when opened, shall present the user with the main interface that presents the following:
    - Option to create a new flashcard
    - Option to start a review session of today's scheduled cards
    - Option to view a list of cards in the database

Flashcard Editing/Creation

- The system shall present the user with a graphical interface to create a new or edit an existing flashcard
- The edit/create interface shall allow the user to enter text into "front" and "back" fields, which will correspond to the front and backsides of a flashcard
- The system shall present the user with an option to save or cancel edits made to a flashcard
- The system shall store the generated flashcards in a local database

Flashcard Database

- The user shall have the ability to view their cards in a database
- The user shall be able to sort cards by their front field, creation date, or last reviewed date
- The user shall have the ability to search for a flashcard by keywords in its front or
- The user shall have the ability to open the edit interface for a selected card from the database view
- The user shall have the ability to delete a selected flashcard

Review Sessions

- The system shall present the user with a daily set of cards to review per a spaced repetition algorithm
- Upon opening a review session, the system shall present the user with the front side of a card
- When presented with the front side of a card, the user may press a "reveal" button, at which point the system will show the backside of the card

- Upon revealing the backside of a card, the system shall present the user with buttons to mark their answer to the card correct or false
- The system shall use the user response to the card to schedule a next review of the card, per the spaced repetition specifications
- The user shall have the option to quit a review session at any time and return to it later

Design limitations

In this current proposal, flashcards will be limited to only “front” and “back” fields. Additionally, input in these fields will be limited to Unicode text. There will be no pictures, audio or other types of media. Flashcards will have limited customizability in their appearance, using only default settings for font, color, etc. These are features that may be added at a later date or if time allows in this project.

5. Approach

This project requires a flexible language with simple GUI capabilities and easy interfaces to database libraries. For this reason, we choose to use Java. This additionally provides easy cross-platform support as many team members develop on devices using different operating systems.

We believe it is prudent to use a database management system to store and retrieve flashcards. This will allow the team to focus more on user interface and visible features, rather than low-level database management. We have chosen MySQL as the database management system due to team familiarity with it and its easy integration with Java.

As a team, we plan to rapidly prototype small versions of the system to test the viability of the previous choices and readjust as early as possible. The basic layout as we see it is:

1. Design database format for flashcards and write wrapper functions allowing the system easy access
2. Create a fixed set of flashcards for testing
3. Implement the spaced repetition algorithm
4. Create a minimal system to test the viability of the full system
5. Create a graphical interface to display cards
6. Create the main interface to select reviews/edits/database lookup
7. Create user interface to view cards in the database
8. Create user ability to add/edit cards graphically
9. Cleanup graphical interface

6. Project management

For ease of integration, the project will be managed in Github through the project management feature. Tasks for each member will be created in Github and tracked as “Todo”, “In Progress”, or “Done”.

Additionally, Github will provide the team with SCM. Members will work off of “feature” branches and merge changes into a master branch. It is yet to be determined what the appropriate continuous integration tool for the project will be, but potential candidates are Jenkins and Travis. This decision will be made before the first iteration.

As a team, we will scrum after class (Monday and Wednesdays at 9:45) and on weekends as needed. Here we will discuss issues, bugs, and potential design changes we see in the future.

Outlined below is a project schedule containing the tasks in each iteration of the project, as well as their assignee.

Project Schedule			
Iteration	Dates	Task	Responsible
Project Proposal	9/29/2019	Proposal	Team
	10/5/2019	Install tools and setup personal computers	Team
	10/8/2019	Build Card Database	Howard
	10/8/2019	Build flash Card scheduling system	Ryan
	10/10/2019	Build a deck of test cards - Dependent on card database	Yicun
	10/13/2019	Build command-line interface to the system - Dependent on all previous tasks	Team
Iteration1	10/13/2019		Team
	10/17/2019	Implement the main user interface	Howard
	10/17/2019	Implement database viewing interface	Yicun
	10/17/2019	Implement card reviewing interface	Ryan
	10/20/2019	Integrate various interface views into the main program	Yicun
	10/20/2019	Implement sorting system for database viewing interface	Howard

		- Dependent on implementation of database viewing interface	
	10/25/2019	Add the ability to suspend or delete cards from the scheduler	Ryan
Iteration2	10/29/2019		
	10/30/2019	Create card creating/editing interface	Howard
	10/31/2019	Create search bar for database viewer	Yicun
	11/5/2019	Integrate card editor into database system	Ryan
	11/15/2019	Finalize interface and fix remaining bugs in system	Team
Iteration3	11/17/2019		
	12/1/2019~ 12/4/2019	Final Presentation	Team

7. Deliverables for each Iteration

Iteration 1

In iteration 1, the team plans to have a basic command line functionality of the intended system. This command-line tool will interface a database system of flash cards and present them to a user for review. The user may provide “pass” or “fail” responses to the cards which will allow the scheduler to handle them appropriately. Cards will not yet be editable from the tool, nor will they be searchable.

Iteration 2

By iteration 2, the team will have a graphical interface to the tool capable of displaying cards for review. The user will be able to flip cards to see the answer on the back, as well as provide responses indicating their successful/unsuccessful answer to the card. Additionally, the user will have the ability to open a graphical interface to the database and show cards sorted by various properties. The user may delete cards from the database or suspend them from being reviewed.

Iteration 3

Iteration 3 will provide the user with fully-fledged editing capabilities of both new cards and existing cards. The database will be searchable by keywords in the card fields. All user interfaces will be polished and provide a clean and easy to use mechanism for interacting with the system.

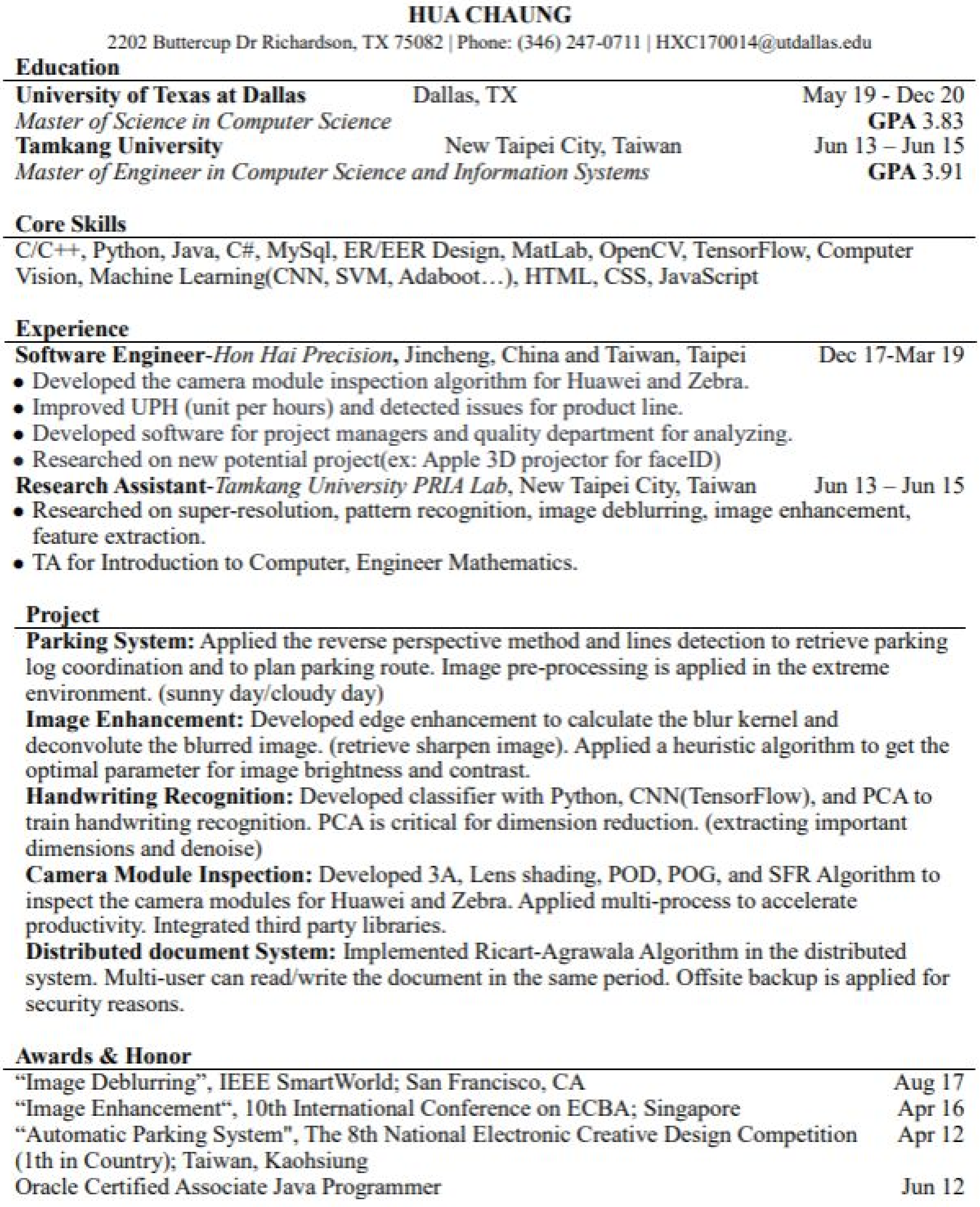
8. Team qualification

Team 2 is a highly qualified group of go-getters who are more than capable of completing this project by the deadline. With years of industry experience, multiple degrees spanning various technical disciplines and dashing good looks, Team 2 will have this project knocked out in no time. What follows is the resumes of the brilliant young men who have chosen to undertake this daunting effort.

Howard:
Hua Chuang worked at Foxconn as a camera module software engineer in China and Taiwan. His responsibility was developing the camera's module inspection algorithm. This job also covers a lot of application development and teamwork with different departments and clients. With those experiences, He is competent for this project.

Yicun:
Yicun Liu has machine learning in Cloudwalk, China. His work was developing software to detect electrical substation equipment defects and dangerous actions in electrical substations.Yicun's strong self-learning skills and problem-solving ability helped him in improving this project.

Ryan:
Ryan has several years of industry experience, both in the world of defense contracting and commercial robotics. Having worked in Agile environments, Ryan is ready to lead the team through fast paced development towards full-featured software products. In addition, Ryan is experienced with software testing and continuous integration development tools, which he will use to root out bugs and create a robust and error-free final product.



HUA CHAUNG

2202 Buttercup Dr Richardson, TX 75082 | Phone: (346) 247-0711 | HXC170014@utdallas.edu

Education

University of Texas at Dallas Dallas, TX May 19 - Dec 20
Master of Science in Computer Science **GPA** 3.83
Tamkang University New Taipei City, Taiwan Jun 13 – Jun 15
Master of Engineer in Computer Science and Information Systems **GPA** 3.91

Core Skills

C/C++, Python, Java, C#, MySql, ER/EER Design, MatLab, OpenCV, TensorFlow, Computer Vision, Machine Learning(CNN, SVM, Adaboot…), HTML, CSS, JavaScript

Experience

Software Engineer-*Hon Hai Precision*, Jincheng, China and Taiwan, Taipei Dec 17-Mar 19

- Developed the camera module inspection algorithm for Huawei and Zebra.
- Improved UPH (unit per hours) and detected issues for product line.
- Developed software for project managers and quality department for analyzing.
- Researched on new potential project(ex: Apple 3D projector for faceID)

Research Assistant-*Tamkang University PRIA Lab*, New Taipei City, Taiwan Jun 13 – Jun 15

- Researched on super-resolution, pattern recognition, image deblurring, image enhancement, feature extraction.
- TA for Introduction to Computer, Engineer Mathematics.

Project

Parking System: Applied the reverse perspective method and lines detection to retrieve parking log coordination and to plan parking route. Image pre-processing is applied in the extreme environment. (sunny day/cloudy day)
Image Enhancement: Developed edge enhancement to calculate the blur kernel and deconvolute the blurred image. (retrieve sharpen image). Applied a heuristic algorithm to get the optimal parameter for image brightness and contrast.
Handwriting Recognition: Developed classifier with Python, CNN(TensorFlow), and PCA to train handwriting recognition. PCA is critical for dimension reduction. (extracting important dimensions and denoise)
Camera Module Inspection: Developed 3A, Lens shading, POD, POG, and SFR Algorithm to inspect the camera modules for Huawei and Zebra. Applied multi-process to accelerate productivity. Integrated third party libraries.
Distributed document System: Implemented Ricart-Agrawala Algorithm in the distributed system. Multi-user can read/write the document in the same period. Offsite backup is applied for security reasons.

Awards & Honor

"Image Deblurring", IEEE SmartWorld; San Francisco, CA	Aug 17
"Image Enhancement", 10th International Conference on ECBA; Singapore	Apr 16
"Automatic Parking System", The 8th National Electronic Creative Design Competition (1th in Country); Taiwan, Kaohsiung	Apr 12
Oracle Certified Associate Java Programmer	Jun 12

Yicun Liu

Email:yxl170028@utdallas.edu

Education

THE UNIVERSITY OF TEXAS AT DALLAS, Richardson, Texas
MSc in Computer Science

SHANGHAI JIAOTONG UNIVERSITY, Shanghai
B.S in Physics,

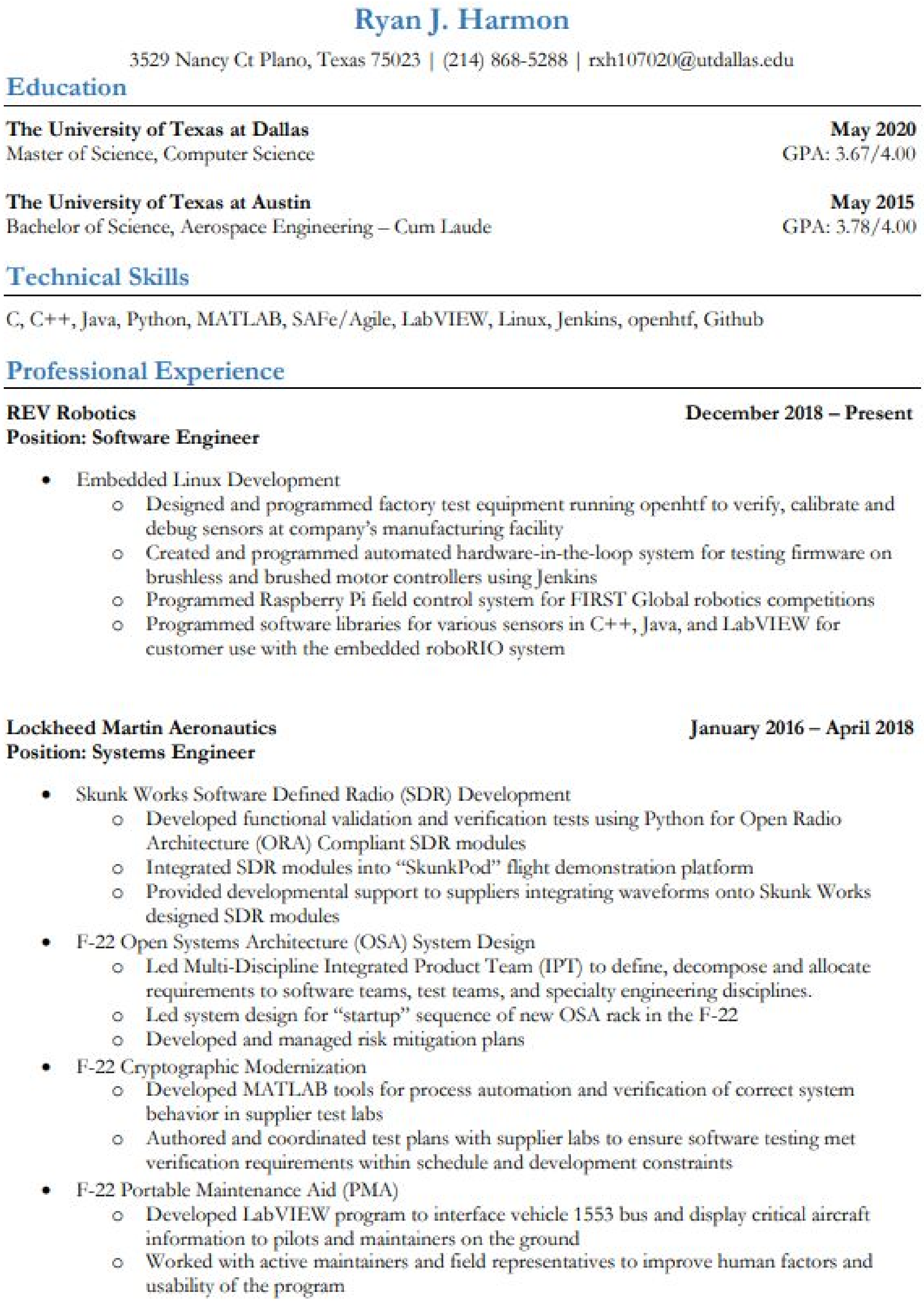
Computer Skills

Languages: Python, Java, C++, HTML, JavaScript, MIPS, Common Lisp, Prolog
Database: Oracle SQL
Operating System: UNIX, Linux Ubuntu, Windows

Projects And Experience

Machine Learning Intern Summer 2019

Object detection to detect electrical substation equipment defects
Action Recognition to detect dangerous actions in electrical substation



Ryan J. Harmon

3529 Nancy Ct Plano, Texas 75023 | (214) 868-5288 | rxh107020@utdallas.edu

Education

The University of Texas at Dallas — **May 2020**
Master of Science, Computer Science — GPA: 3.67/4.00

The University of Texas at Austin — **May 2015**
Bachelor of Science, Aerospace Engineering – Cum Laude — GPA: 3.78/4.00

Technical Skills

C, C++, Java, Python, MATLAB, SAFe/Agile, LabVIEW, Linux, Jenkins, openhtf, Github

Professional Experience

REV Robotics — **December 2018 – Present**
Position: Software Engineer

- Embedded Linux Development
  - Designed and programmed factory test equipment running openhtf to verify, calibrate and debug sensors at company's manufacturing facility
  - Created and programmed automated hardware-in-the-loop system for testing firmware on brushless and brushed motor controllers using Jenkins
  - Programmed Raspberry Pi field control system for FIRST Global robotics competitions
  - Programmed software libraries for various sensors in C++, Java, and LabVIEW for customer use with the embedded roboRIO system

Lockheed Martin Aeronautics — **January 2016 – April 2018**
Position: Systems Engineer

- Skunk Works Software Defined Radio (SDR) Development
  - Developed functional validation and verification tests using Python for Open Radio Architecture (ORA) Compliant SDR modules
  - Integrated SDR modules into "SkunkPod" flight demonstration platform
  - Provided developmental support to suppliers integrating waveforms onto Skunk Works designed SDR modules
- F-22 Open Systems Architecture (OSA) System Design
  - Led Multi-Discipline Integrated Product Team (IPT) to define, decompose and allocate requirements to software teams, test teams, and specialty engineering disciplines.
  - Led system design for "startup" sequence of new OSA rack in the F-22
  - Developed and managed risk mitigation plans
- F-22 Cryptographic Modernization
  - Developed MATLAB tools for process automation and verification of correct system behavior in supplier test labs
  - Authored and coordinated test plans with supplier labs to ensure software testing met verification requirements within schedule and development constraints
- F-22 Portable Maintenance Aid (PMA)
  - Developed LabVIEW program to interface vehicle 1553 bus and display critical aircraft information to pilots and maintainers on the ground
  - Worked with active maintainers and field representatives to improve human factors and usability of the program