FIU Knight Foundation School of Computing and Information Sciences

Knight Foundation School of Computing and Information Sciences

Summer 2024 Senior Design Project

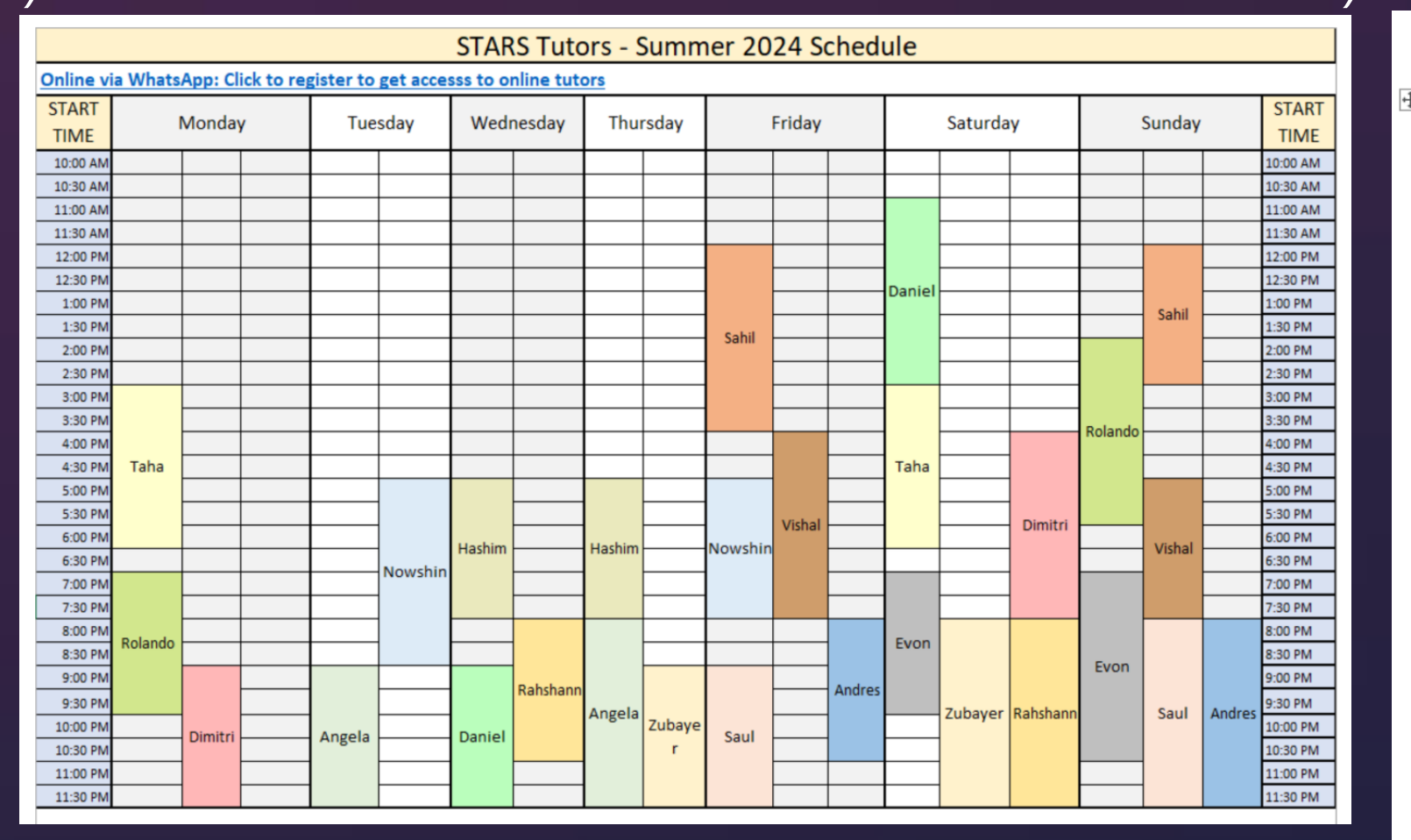
Enhancing the STARS Admin App for Automation

Student: Kristy Hamlin
Mentor: *Dr. Patricia Wells*, *Associate Teaching Professor at KFSCIS*
Instructor/Faculty: *Dr. Masoud Sadjadi*, Florida International University

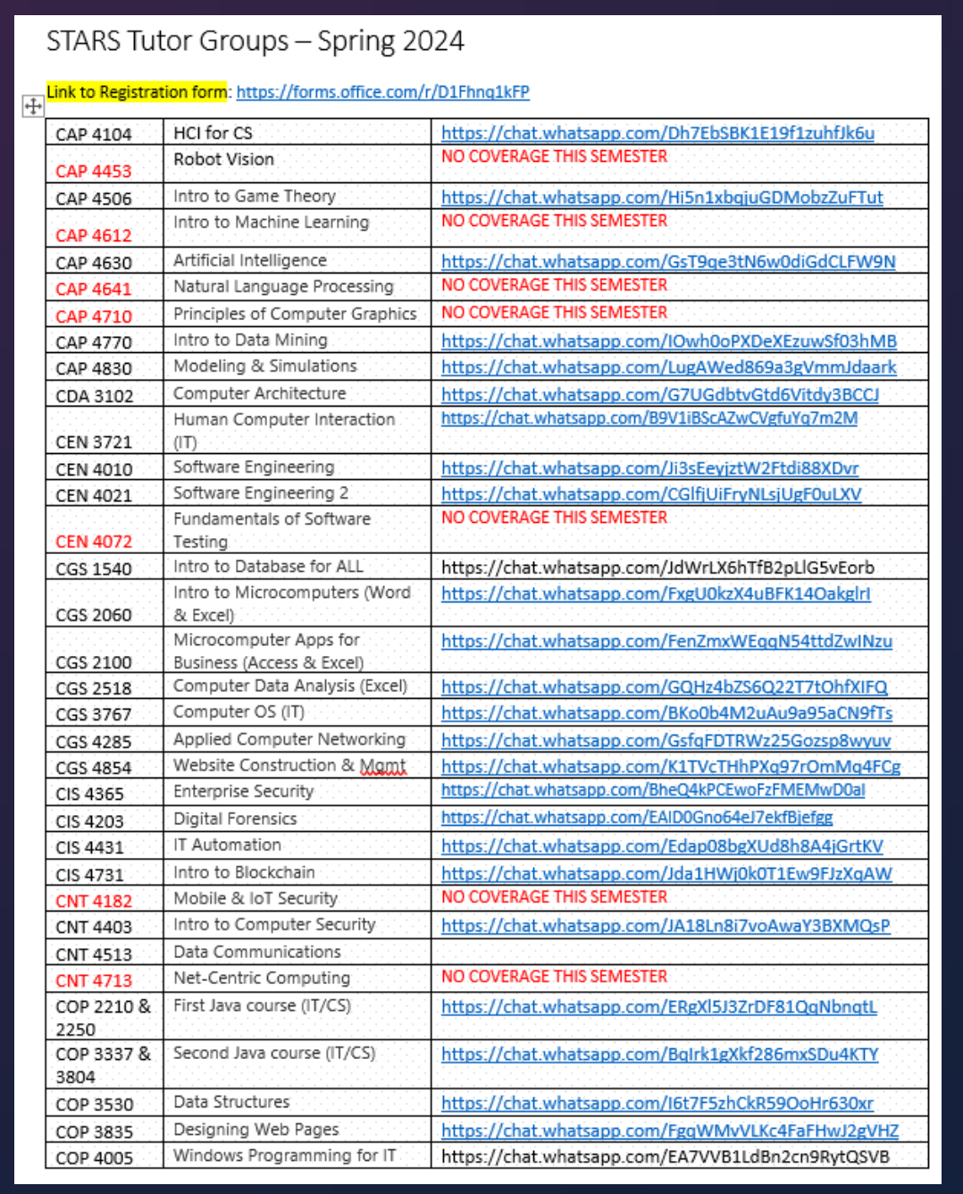
PROBLEM

Every semester, Dr. Patricia Wells coordinates a tutoring service known as STARS (Students in Technology, Academia, Research and Service). Administering this program involves dedicating hundreds of hours to manually approving and denying FIU students' requests for tutoring, identifying and recruiting students willing to volunteer as tutors, and creating WhatsApp chat groups for each tutoring course.

a)



b)



STARS Tutor Groups – Spring 2024

Link to Registration form: https://forms.office.com/r/D1Fhnq1kFP

CAP 4104	HCI for CS	https://chat.whatsapp.com/Dh7EbSBK1E19f1zuhfJk6u
CAP 4453	Robot Vision	NO COVERAGE THIS SEMESTER
CAP 4506	Intro to Game Theory	https://chat.whatsapp.com/Hi5n1xbqjuGDMobzZuFTut
CAP 4612	Intro to Machine Learning	NO COVERAGE THIS SEMESTER
CAP 4630	Artificial Intelligence	https://chat.whatsapp.com/GsT9qe3tN6w0diGdCLFW9N
CAP 4641	Natural Language Processing	NO COVERAGE THIS SEMESTER
CAP 4710	Principles of Computer Graphics	NO COVERAGE THIS SEMESTER
CAP 4770	Intro to Data Mining	https://chat.whatsapp.com/IOwh0oPXDeXEzuwSf03hMB
CAP 4830	Modeling & Simulations	https://chat.whatsapp.com/LugAWed869a3gVmmJdaark
CDA 3102	Computer Architecture	https://chat.whatsapp.com/G7UGdbtvGtd6Vitdy3BCCJ
CEN 3721	Human Computer Interaction (IT)	https://chat.whatsapp.com/B9V1iBScAZwCVgfuYq7m2M
CEN 4010	Software Engineering	https://chat.whatsapp.com/Ji3sEeyjztW2Ftdi88XDvr
CEN 4021	Software Engineering 2	https://chat.whatsapp.com/CGlfjUiFryNLsjUgF0uLXV
CEN 4072	Fundamentals of Software Testing	NO COVERAGE THIS SEMESTER
CGS 1540	Intro to Database for ALL	https://chat.whatsapp.com/JdWrLX6hTfB2pLlG5vEorb
CGS 2060	Intro to Microcomputers (Word & Excel)	https://chat.whatsapp.com/FxgU0kzX4uBFK14OakglrI
CGS 2100	Microcomputer Apps for Business (Access & Excel)	https://chat.whatsapp.com/FenZmxWEqqN54ttdZwINzu
CGS 2518	Computer Data Analysis (Excel)	https://chat.whatsapp.com/GQHz4bZS6Q22T7tOhfXIFQ
CGS 3767	Computer OS (IT)	https://chat.whatsapp.com/BKo0b4M2uAu9a95aCN9fTs
CGS 4285	Applied Computer Networking	https://chat.whatsapp.com/GsfqFDTRWz25Gozsp8wyuv
CGS 4854	Website Construction & Mgmt	https://chat.whatsapp.com/K1TVcTHhPXq97rOmMq4FCg
CIS 4365	Enterprise Security	https://chat.whatsapp.com/BheQ4kPCEwoFzFMEMwD0al
CIS 4203	Digital Forensics	https://chat.whatsapp.com/EAlD0Gno64eJ7ekfBjefgg
CIS 4431	IT Automation	https://chat.whatsapp.com/Edap08bgXUd8h8A4jGrtKV
CIS 4731	Intro to Blockchain	https://chat.whatsapp.com/Jda1HWj0k0T1Ew9FJzXqAW
CNT 4182	Mobile & IoT Security	NO COVERAGE THIS SEMESTER
CNT 4403	Intro to Computer Security	https://chat.whatsapp.com/JA18Ln8i7voAwaY3BXMQsP
CNT 4513	Data Communications	
CNT 4713	Net-Centric Computing	NO COVERAGE THIS SEMESTER
COP 2210 & 2250	First Java course (IT/CS)	https://chat.whatsapp.com/ERgXl5J3ZrDF81QqNbnqtL
COP 3337 & 3804	Second Java course (IT/CS)	https://chat.whatsapp.com/Bqlrk1gXkf286mxSDu4KTY
COP 3530	Data Structures	https://chat.whatsapp.com/I6t7F5zhCkR59OoHr630xr
COP 3835	Designing Web Pages	https://chat.whatsapp.com/FgqWMvVLKc4FaFHwJ2gVHZ
COP 4005	Windows Programming for IT	https://chat.whatsapp.com/EA7VVB1LdBn2cn9RytQSVB

Figure 1. a) A Screenshot of the Summer 2024 STARS Tutoring availability Excel Sheet [1]. b) A Screenshot of the courses and links to Whatsapp tutoring groups from Spring 2024 [1]. These screenshots give the reader a sense of the amount of manual labor involved in administrating the STARS tutoring service each semester.

This management consumes a significant amount of time and represents a critical obstacle to the scalability of the tutoring service. Figure 1 shows real screenshots from Dr. Wells' work this semester.

In Spring 2024, an initial Capstone II group began implementing a strategy to automate as much of this administration as possible using Microsoft Power Applications. This semester, my team picked up work on the application.

CURRENT SYSTEM

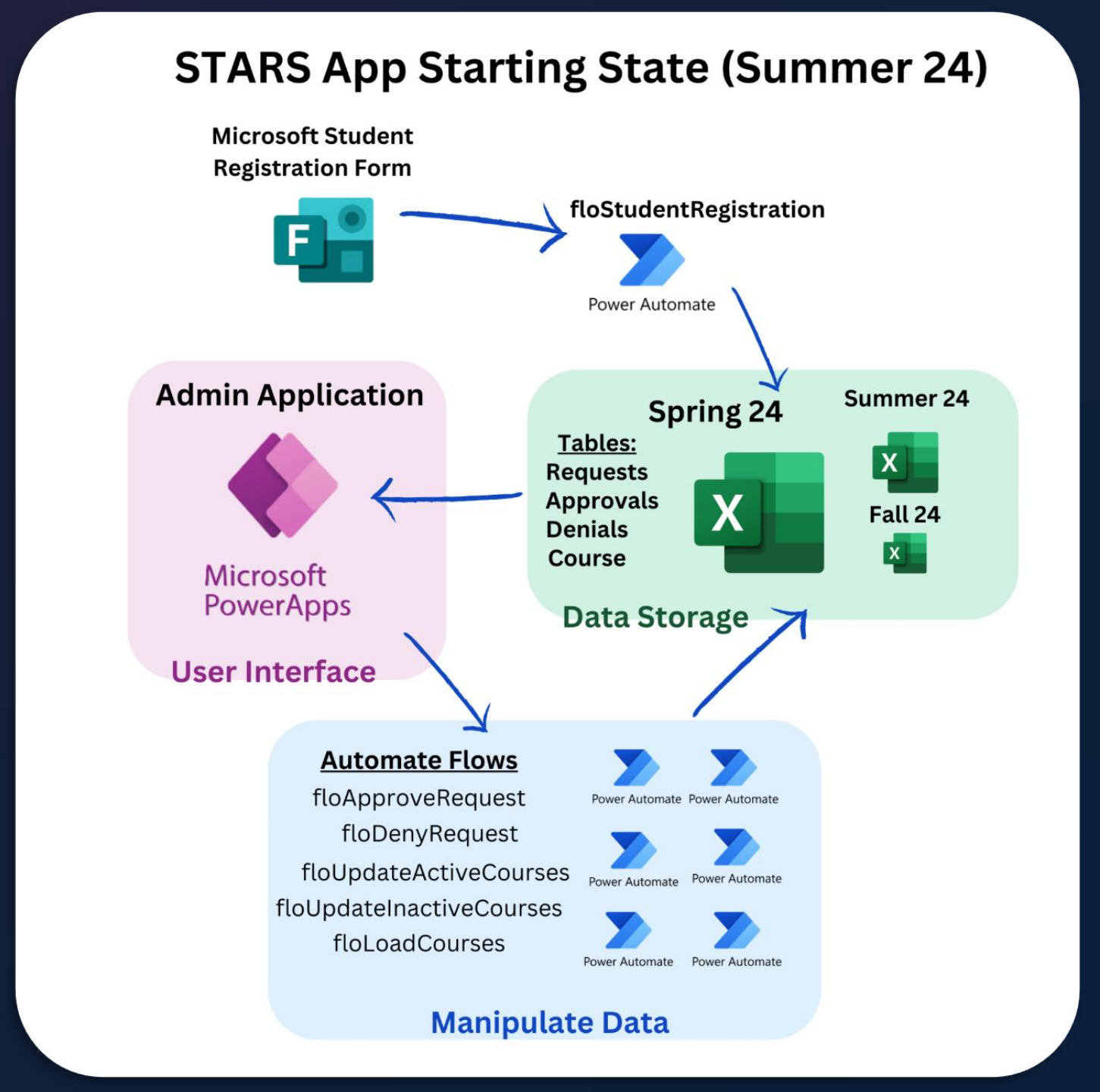


Figure 2. The components of the STARS Application left by the Spring '24 Capstone team. Great work!

Microsoft Power Applications is a development platform designed to allow users to rapidly develop custom apps for a variety of business needs. Microsoft Canvas is the tool used to design the user interface and pages of an app, while Power Automate is used to create workflows (called "flows") that can automate a large assortment of tasks.

The initial system created by the previous Capstone team, and inherited by my team, is represented in Figure 2. All components of the system are hosted in Microsoft 365.

The submission of a new Student Registration Form triggers the first Power Automate Flow, which adds the student's information into the Requests table of the appropriate semester.

The STARS Admin App is the main interface by which the user interacts with system. From the App, the Admin can manipulate data stored in a series of Excel files. Each excel file represents a different semester, which introduced difficulties into the system (more on this in following sections). On the "backend", the app manipulates the data using the flows created by the development team. Such actions include approving and denying student tutoring requests.

REQUIREMENTS

The system implemented by the Spring 24 team, although not deployed, was capable of automating several of the functions that were manually performed by Dr. Wells. These functions include organizing and gathering student requests into an Excel, keeping track of active/inactive courses, and approving/denying student requests from a convenient interface (along with sending the student an email).

However, the system was not perfect, and there were still tasks that needed automation. Because the scope of this poster is restricted to my contributions, I will describe the areas I worked on:

Firstly, the system attempted to use burdensome switches within every flow to change between Excel files representing each semester. This is partially because data sources for Power Apps cannot (as of the writing of this poster) be dynamically chosen – they must already exist and be specified. Likely because of how troublesome this was, the ability to choose a semester was not uniformly implemented throughout the App or Flows. This system required fixing.

Secondly, Dr. Wells requested that I implement the ability to approve student requests either via email or using the app, an option which was not trivial and not previously attempted. She also requested some minor improvements, including the ability to sort the Approval and Denial tables. Finally, she requested that one student (me) research the process for deploying a Microsoft Power Application.

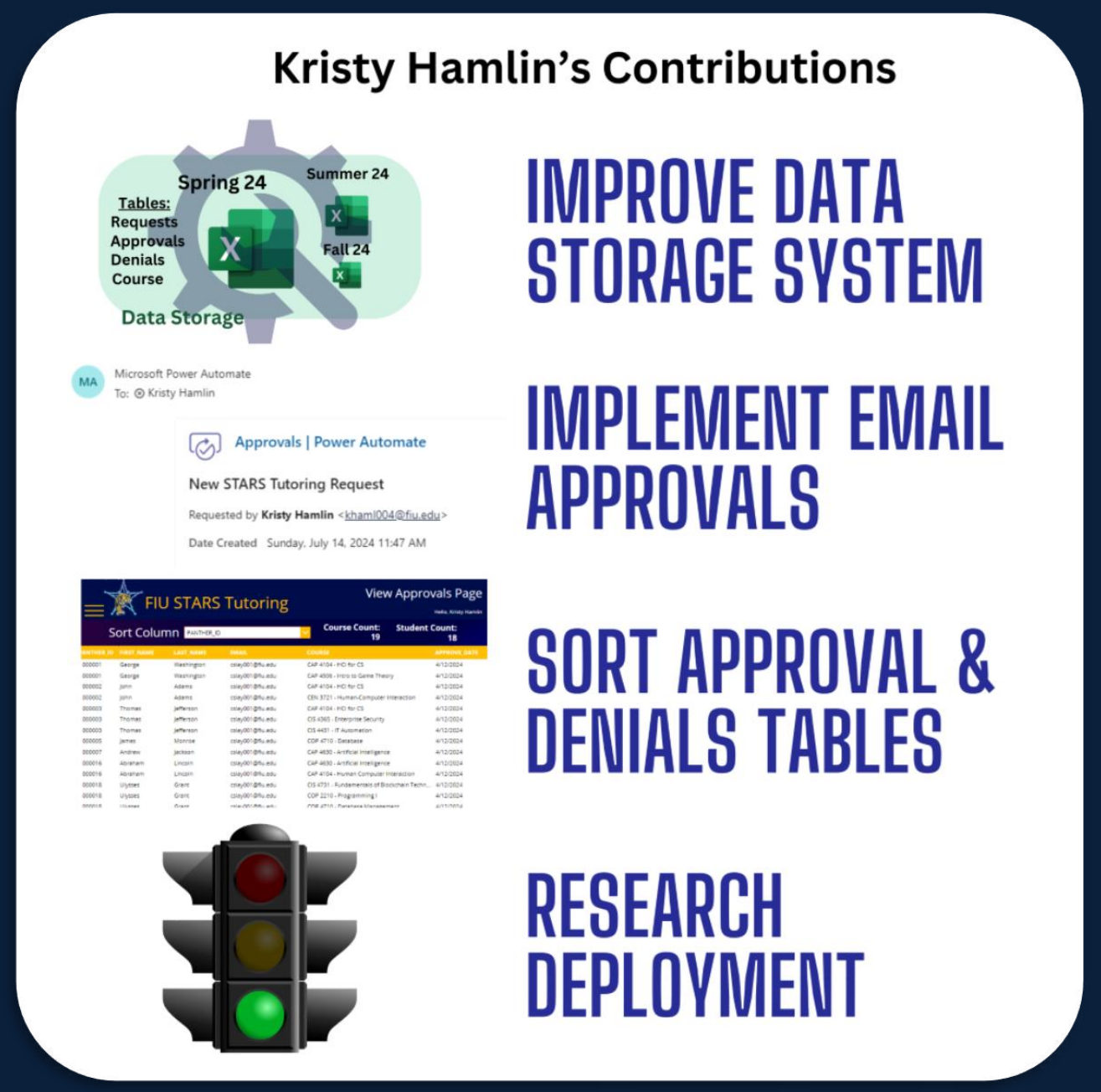

Kristy Hamlin's Contributions
Spring 24
Summer 24
Tables:
Requests
Approvals
Denials
Course
Fall 24
Data Storage
IMPROVE DATA STORAGE SYSTEM
Microsoft Power Automate
To: Kristy Hamlin
Approvals | Power Automate
New STARS Tutoring Request
Requested by Kristy Hamlin <khaml004@fiu.edu>
Date Created Sunday, July 14, 2024 11:47 AM
IMPLEMENT EMAIL APPROVALS
FIU STARS Tutoring
View Approvals Page
Sort Column
Course Count: 19
Student Count: 18
SORT APPROVAL & DENIALS TABLES
RESEARCH DEPLOYMENT

Figure 3. Kristy Hamlin's main areas of contribution to the STARS Application.

SYSTEM DESIGN

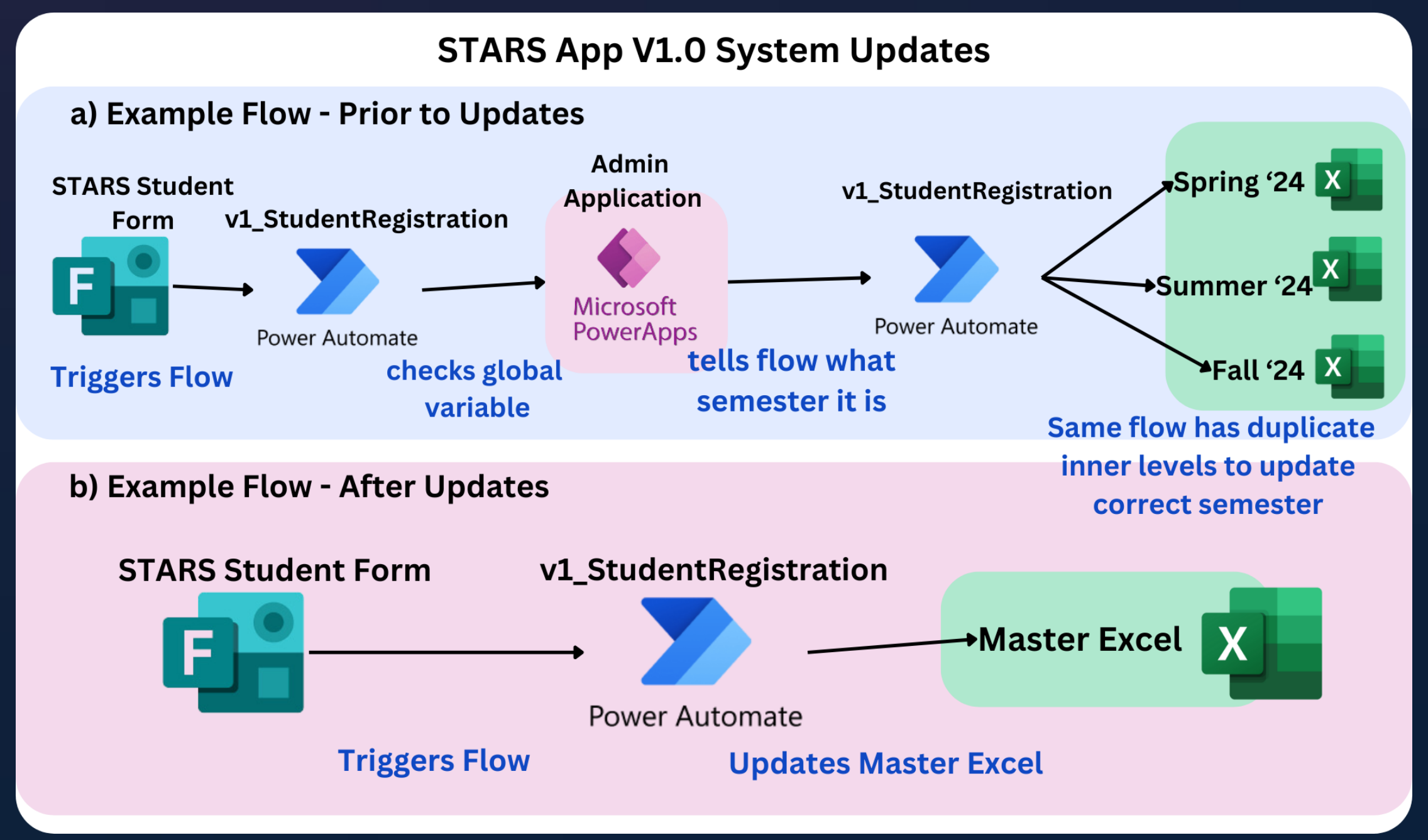


Figure 4. a) An example flow which uses switch statements to switch between alternate Excels. It must check with the App to determine what semester it is, and then the Excel connections must be hard-coded into the flow. When the pre-created choices run out, the developers will need to add more and change all the connections. b) An example of the same flow following the logic I propose, where a single Master Excel file is copied and cleared at the end of every semester.

One of my biggest contributions this semester was proposing an updated system design to eliminate the switch statements alluded to in the "Requirements" section. The previous team had tried to implement a system by which each semester's data was stored in a different Excel file. The problem with this is that Power Apps currently does not easily implement switching between data sources for flows and Canvas apps. Any data sources must already exist and the connections to the files must be in the solution. Therefore, the previous Capstone team needed to pre-create Excel files for future semesters. Even though this was possible, it resulted in complicated switch statements within all of the flows, and it was not uniformly implemented throughout the application.

To fix this, I proposed a new system design that I dubbed "v1.0" of the app. I wrote a report ("STARS App V1.0 Proposal.docx") outlining an implementation strategy which will be included in the project documents. In my updated design (see Figure 4), the Admin App and flows are all programmed to work with one Excel file which I dubbed the "Master Excel", and the "End Semester" process every semester makes a copy of this file and clears the Master Excel to be ready for a new semester. This simple change makes a world of difference in terms of complexity for future development teams, and also in terms of maintaining the app and flows.

Although the system design was my idea, the whole Capstone II teams has been working on the user stories I proposed in this report.

IMPLEMENTATION

Within this new system design, I was responsible for implementing the updated Student Registration system (Figure 5). This system required me to create a new flow with the switch statements removed and the connections updated. In addition, I was also responsible for adding the feature Dr. Wells had requested to be able to approve Student Requests via email.

Implementing this feature required me to create a child flow – called v1_AdminEmailApprovals – to send the Power Automate Approvals email. This is because Automate Approvals cause a flow to halt execution until the approval is received. Secondly, I implemented a table to keep track of up to 3 admin email addresses along with the GUI in the Admin App for updating them. This is needed to that the v1_AdminEmailApprovals flow knows where to send the approvals to.

Finally, implementing this system required testing to ensure that a student request couldn't be approved or denied twice via alternate methods. These complications will be discussed in greater detail in the Project Documentation.

Figure 5 shows the completed STARS App Student Registration subsystem, which I improved and updated this semester.

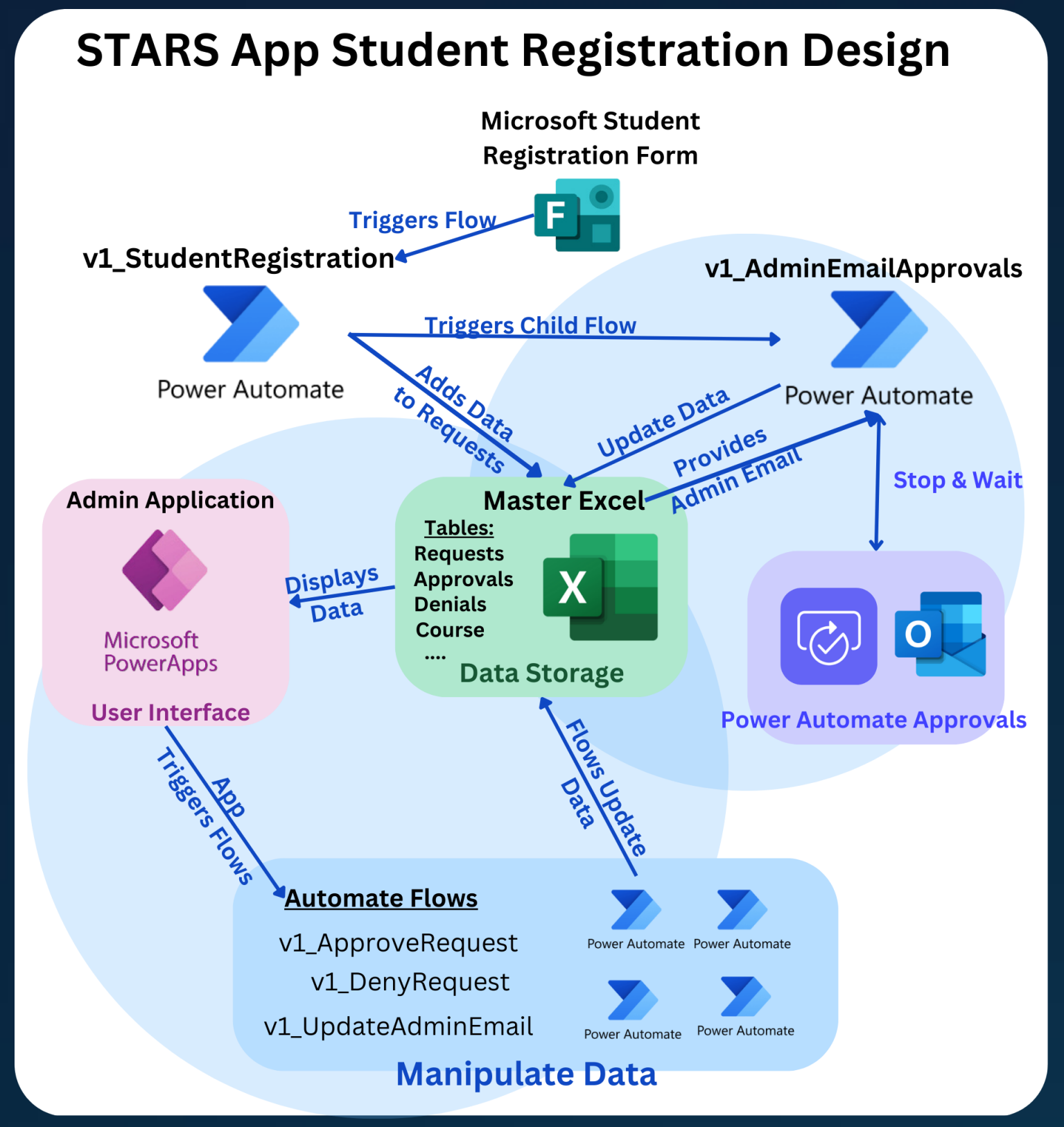


Figure 5) The student registration system updated by me. This system allows the admin to approve student registrations either via the Admin App, or via email using Power Automate Approvals. Setting up the system to allow both options and be able to use the Admin's email while the admin was not logged into the app was not trivial.

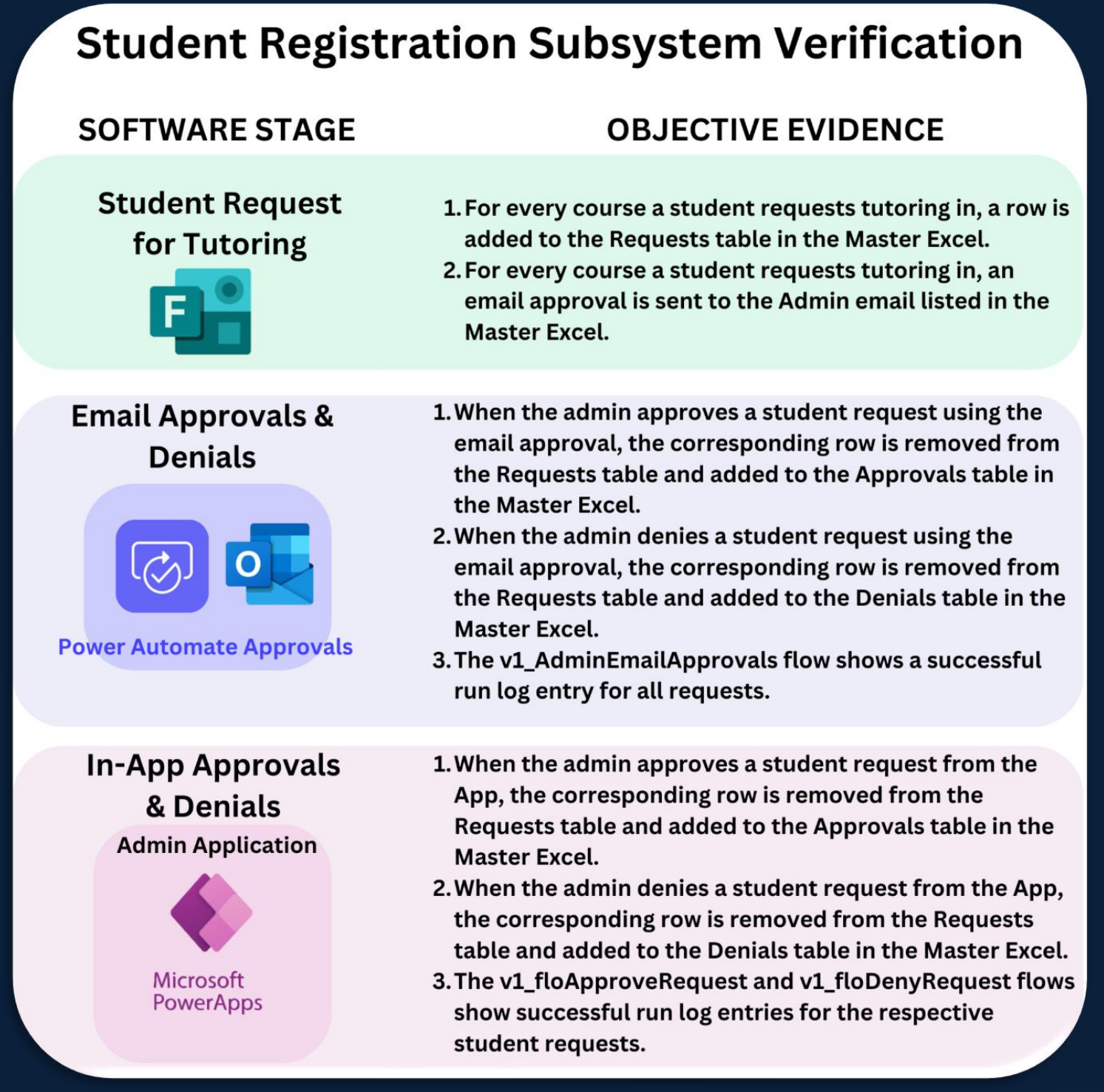
My smaller achievements this semester included implementing the ability to sort the Approval and Denials tables within the app based on any column. This was my first user story as I was initially the most junior member of the team in terms of CS experience, although I compensated for this by studying the app thoroughly. I also researched the deployment of Microsoft Power Apps throughout the semester.

VERIFICATION

This software project is being developed using the iterative cycle, whereby every semester a new team of Capstone II students improves upon the same project. System verification takes place throughout every phase of process development, and crucially starting from the system design [2]. According to Snoderly and Faisandier, "Verfication is the confirmation, through the provision of objective evidence, that specified requirements have been fulfilled" [2].

With this in mind, I performed testing of the Student Registration subsystem by submitting test data to the system, and comparing the effects of the system versus what I anticipated. To see the different stages of execution within the Student Registration subsystem, and the corresponding output I used for verification, see Figure 6. I am pleased to say that this subsystem is fully tested and implemented as of the end of our last development sprint.

Figure 6. (Right) A chart of the software component undergoing verification along with the objective evidence I searched for to verify that the component functioned properly.



Student Registration Subsystem Verification

SOFTWARE STAGE	OBJECTIVE EVIDENCE
Student Request for Tutoring	1. For every course a student requests tutoring in, a row is added to the Requests table in the Master Excel. 2. For every course a student requests tutoring in, an email approval is sent to the Admin email listed in the Master Excel.
Email Approvals & Denials Power Automate Approvals	1. When the admin approves a student request using the email approval, the corresponding row is removed from the Requests table and added to the Approvals table in the Master Excel. 2. When the admin denies a student request using the email approval, the corresponding row is removed from the Requests table and added to the Denials table in the Master Excel. 3. The v1_AdminEmailApprovals flow shows a successful run log entry for all requests.
In-App Approvals & Denials Admin Application Microsoft PowerApps	1. When the admin approves a student request from the App, the corresponding row is removed from the Requests table and added to the Approvals table in the Master Excel. 2. When the admin denies a student request from the App, the corresponding row is removed from the Requests table and added to the Denials table in the Master Excel. 3. The v1_floApproveRequest and v1_floDenyRequest flows show successful run log entries for the respective student requests.

SUMMARY

This semester, I worked with a team of Capstone II students to make improvements to the STARS Application. This application is intended to automate the STARS tutoring service that is currently administered manually by Dr. Patricia McDermott-Wells.

The STARS App was created by a Capstone II group last semester using Microsoft Power Applications, a platform designed to allow users to quickly develop custom business applications hosted in Microsoft 365.

This semester, I was the team leader, and proposed a new system design to simplify the data storage and Power Automate flows in the App. I was responsible for implementing the updated Student Registration system with additional features, as well as some other small user stories.

REFERENCES

1. McDermott-Wells, P. (2024). *Scope of the Administrative Work for STARS Tutoring* [Word File]. Florida International University, CIS 4951.
2. Snoderly, J. and Faisandier, A. "System Verification." in SEBoK Editorial Board. 2024. *The Guide to the Systems Engineering Body of Knowledge (SEBoK)*, v. 2.10, N. Hutchison (Editor in Chief). Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed July 19, 2024. www.sebokwiki.org.

FIU Engineering & Computing

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Dr. Patricia McDermott-Wells. I thank Dr. Wells for her assistance and mentorship that I received throughout the senior design project.

FLORIDA INTERNATIONAL UNIVERSITY