Electronic Program of Study

sddec22-15

Advisor

Maruf Ahamed

Team Members

Saljooq Altaf, Nathan Marquardt,
Carson Campbell, Noah Nickel, William Hunt

Client

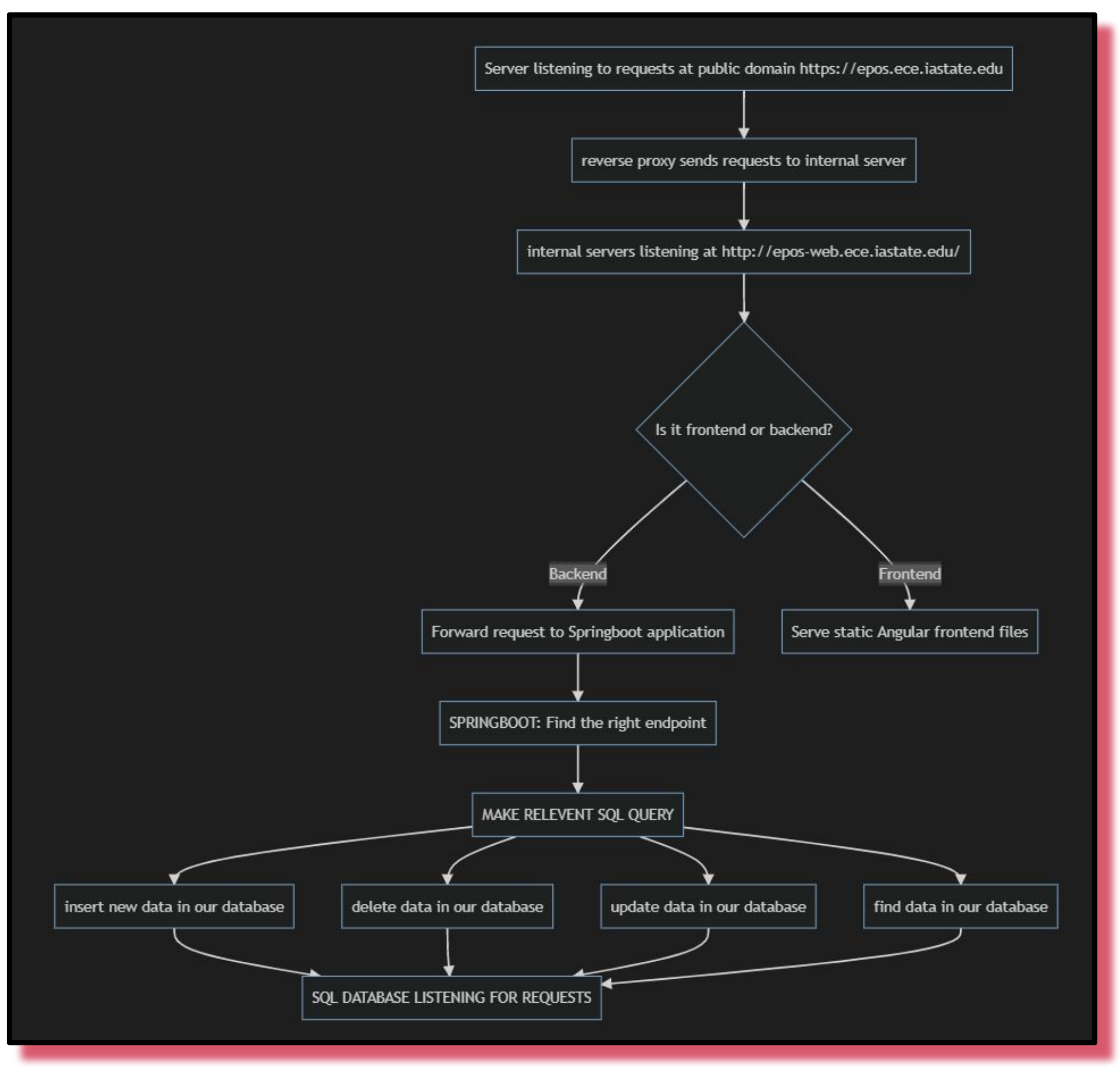
Tina Prouty

Intro/Motivation

- Students at ISU are required to make a Program of Study to help them design their 4 year plan
  - The current method is a spreadsheet that must be manually updated and checked by the student
- Our solution to this problem is to create an online interface that has all the necessary information for students to create a Program of Study to share with their advisors.

Intended Users/Uses

- Students
  - Create a required Program of Study
  - Save a Program of Study
  - Load and edit a previously created Program of Study
- Academic Advisors
  - Review Student's Program of Study



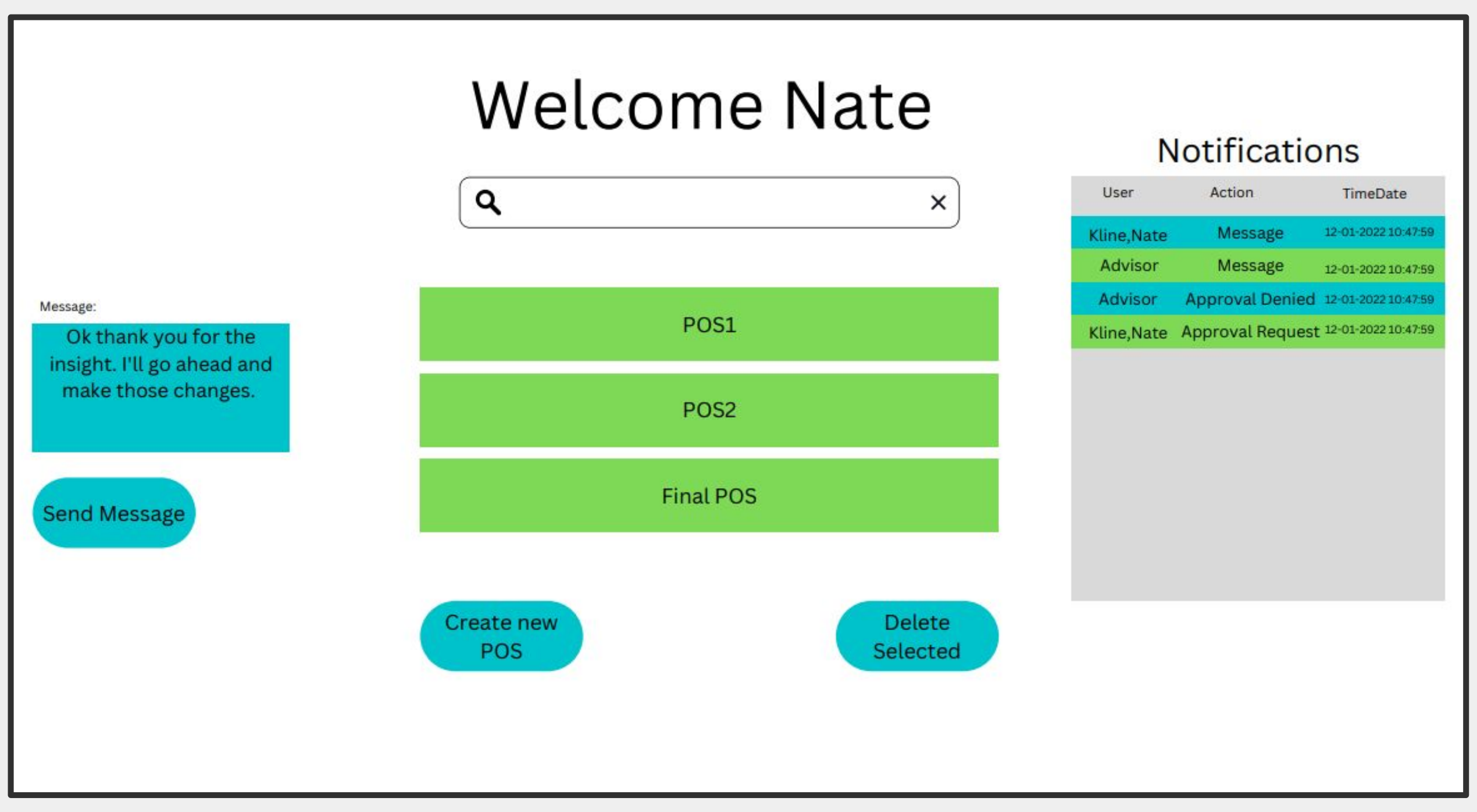
Server listening to requests at public domain https://epos.ece.iastate.edu
reverse proxy sends requests to internal server
internal servers listening at http://epos-web.ece.iastate.edu/
Is it frontend or backend?
Backend
Frontend
Forward request to Springboot application
Serve static Angular frontend files
SPRINGBOOT: Find the right endpoint
MAKE RELEVENT SQL QUERY
insert new data in our database
delete data in our database
update data in our database
find data in our database
SQL DATABASE LISTENING FOR REQUESTS

Design Requirements

- Delete a Course
- Save to a PDF
- Lists Prerequisites, Course Description and Number of Credits
- Search for Courses by Dept. and Num.
- Drag and Droppable Courses
- List of Student's POS
- Load POS
- Save POS
- Create New POS
- Okta Authentication
- Add Courses to POS
- Semesters are Numbered in Order
- No ISU Course Database
- No Support for ISU Okta Authentication
- No cost
- Scalability for multi-user real-time access

Design Approach

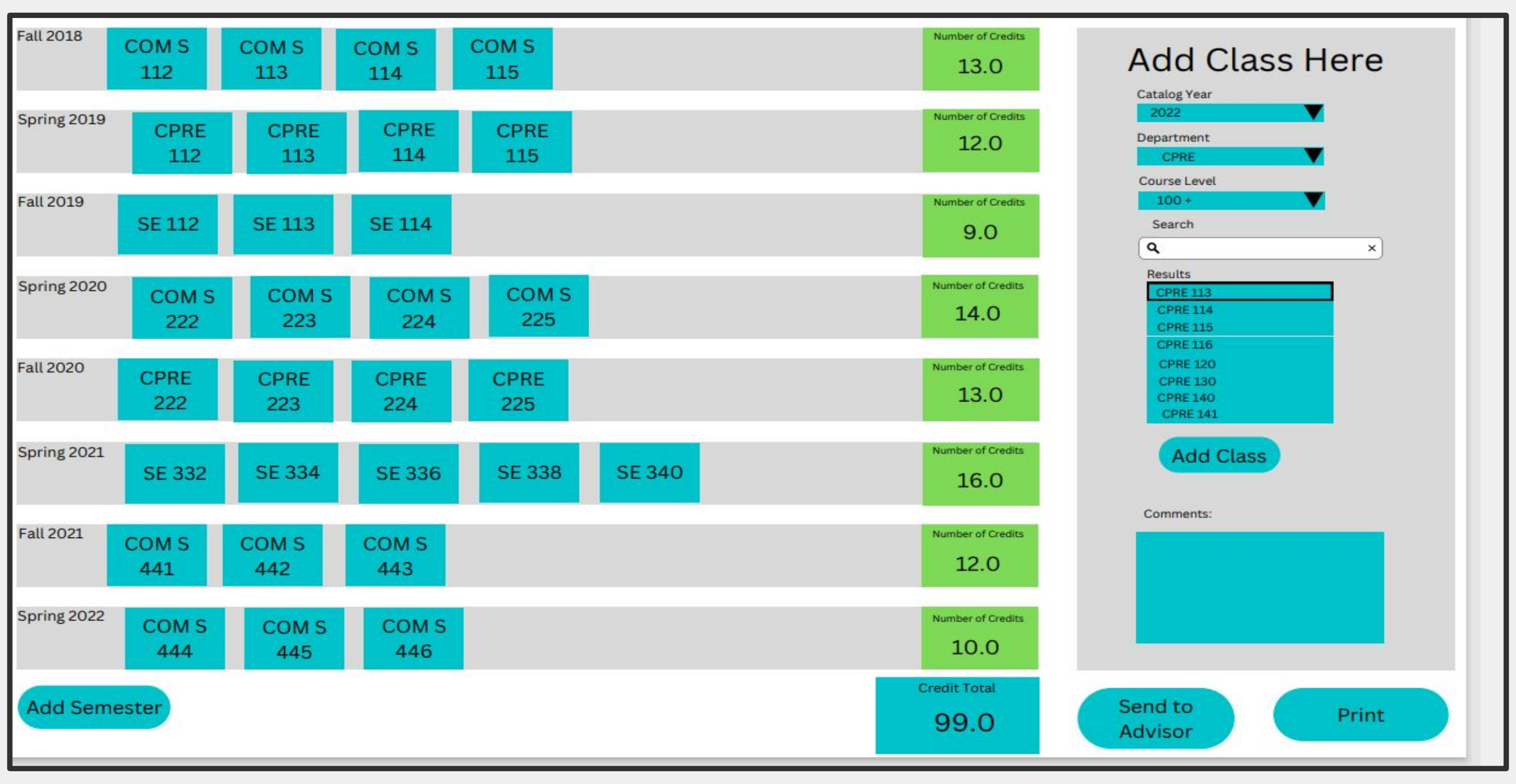
Original Sketches



Message:

Ok thank you for the insight. I'll go ahead and make those changes.

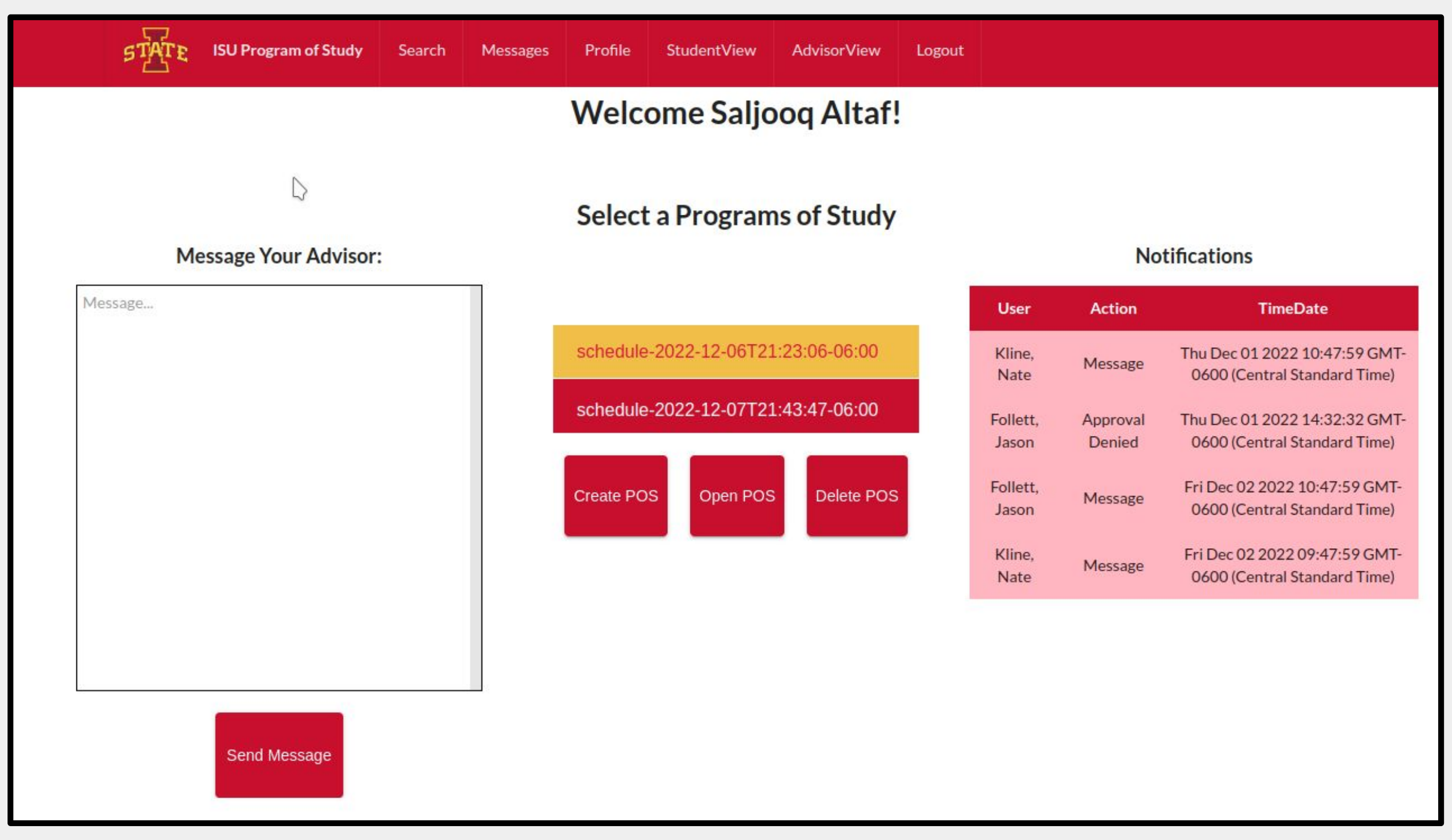
User	Action	TimeDate
Kline,Nate	Message	12-01-2022 10:47:59
Advisor	Message	12-01-2022 10:47:59
Advisor	Approval Denied	12-01-2022 10:47:59
Kline,Nate	Approval Request	12-01-2022 10:47:59



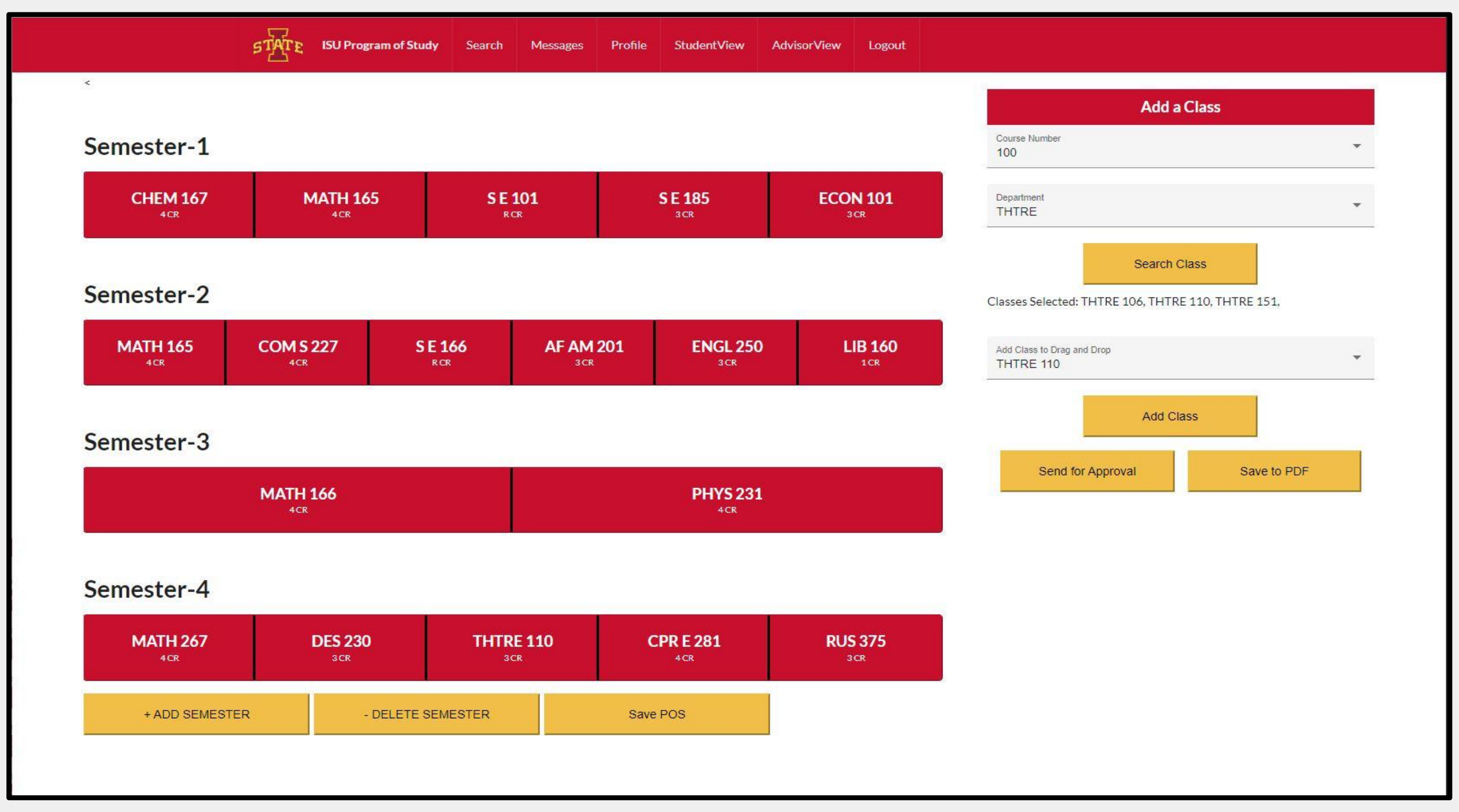
						Number of Credits
Fall 2018	COM S 112	COM S 113	COM S 114	COM S 115		13.0
Spring 2019	CPRE 112	CPRE 113	CPRE 114	CPRE 115		12.0
Fall 2019	SE 112	SE 113	SE 114			9.0
Spring 2020	COM S 222	COM S 223	COM S 224	COM S 225		14.0
Fall 2020	CPRE 222	CPRE 223	CPRE 224	CPRE 225		13.0
Spring 2021	SE 332	SE 334	SE 336	SE 338	SE 340	16.0
Fall 2021	COM S 441	COM S 442	COM S 443			12.0
Spring 2022	COM S 444	COM S 445	COM S 446			10.0

Credit Total 99.0

Final Product



User	Action	TimeDate
Kline, Nate	Message	Thu Dec 01 2022 10:47:59 GMT-0600 (Central Standard Time)
Follett, Jason	Approval Denied	Thu Dec 01 2022 14:32:32 GMT-0600 (Central Standard Time)
Follett, Jason	Message	Fri Dec 02 2022 10:47:59 GMT-0600 (Central Standard Time)
Kline, Nate	Message	Fri Dec 02 2022 09:47:59 GMT-0600 (Central Standard Time)



Classes Selected: THTRE 106, THTRE 110, THTRE 151,

Semester-1

CHEM 167	MATH 165	S E 101	S E 185	ECON 101
4 CR	4 CR	R CR	3 CR	3 CR

Semester-2

MATH 165	COM S 227	S E 166	AF AM 201	ENGL 250	LIB 160
4 CR	4 CR	R CR	3 CR	3 CR	1 CR

Semester-3

MATH 166	PHYS 231
4 CR	4 CR

Semester-4

MATH 267	DES 230	THTRE 110	CPR E 281	RUS 375
4 CR	3 CR	3 CR	4 CR	3 CR

Resources

- Swagger
- Postman
- Angular
- Spring Boot
- Okta
- PostgreSQL
- Apache httpd
- Material UI

Standards

SE Code of Ethics:

- 2.01 Work competence
- 4.04 Financial responsibility
- 1.04 Communication Honesty
- 2.03 Property Ownership

Technical Details

We implemented a Python web crawler to scrape the Iowa State catalog website to get the information we needed for each course. Okta authentication is integrated into the login of our site. Angular was used to implement the Frontend and Java with Spring Boot was used to implement the Backend. We utilized GitLab for version control. Using a reverse proxy we were able to let users not have the need for a VPN. Has scaling capability if many users access at the same time.

Testing

- Backend:
  - Postman
    - Testing all the Endpoints
  - Swagger
    - Run a Request in a Bash Emulator
- Frontend:
  - Running Angular Locally
  - Running Spring Boot Locally with VPN
  - Testing Each Feature Manually