

Transfer Pathways

sdmay22-20

Curt Lengemann - Lead on Middleware/Database Components

Ben Greif - Lead on Testing/Frontend Developer

Luke Turczynski - Lead on API Management/Database Design Development

Cole Weber - Co-lead on Frontend

Cameron Brecount - Co-lead on User Interface

Scott Thurston - Co-lead on Frontend


Riess Radtke - Co-lead on User Interface

Client: Susie Demoss

Advisor: Dr. Ashraf Gaffar

Project Vision

The Transfer Pathways Tool is an improvement of the current TRANSIT system that prospective transfer students use to see which of their courses will transfer to Iowa State

IOWA STATE UNIVERSITY Admissions Search 

Future Students Academics Financing Visit Apply Student Life

TRANSIT

- Step 1. Login/Create Account
- Step 2. Enter Coursework**
- Step 3. Enter Degrees
- Step 4. Transfer Evaluation
- Step 5. Request/View Audits
- Log out
- Help

Add Coursework

Choose the courses you have already taken or will have completed at your community college or university.

Save your coursework by clicking the "Add Course" button. When finished, click "Next" to enter any degrees you have or will receive, or "Transfer Evaluation" to have your coursework evaluated.

Warning: You are currently logged in as a guest. While the system is available for your use, all data will be lost after 30 minutes of inactivity or once you have left TRANSIT system. To save your coursework please [create an account](#).

Course Information

Country/State: |

Institution:

Term/Year: |

Course Department:

Course Number:

Credit Earned:

Grade Earned:

Coursework

| <input type="checkbox"/> | Edit Course | Institution | Year/Term | Course | Credit/Score | Grade |
|--------------------------|-------------------------------------|---|-------------|---------|--------------|-------|
| <input type="checkbox"/> | <input type="button" value="Edit"/> | DES MOINES AREA COMM COLL-WEST DES MOINES | 2021 - Fall | PHY 213 | 6.00 | A- |

IOWA STATE UNIVERSITY Office of Admissions, Enrollment Services Center, Ames, IA 50011 - 2011 | Phone: (800) 262-3810
Copyright © 1995-2012. [Iowa State University](#) | [Consumer Information Disclosures](#) | [College Portrait](#)

Project Vision (Cont.)

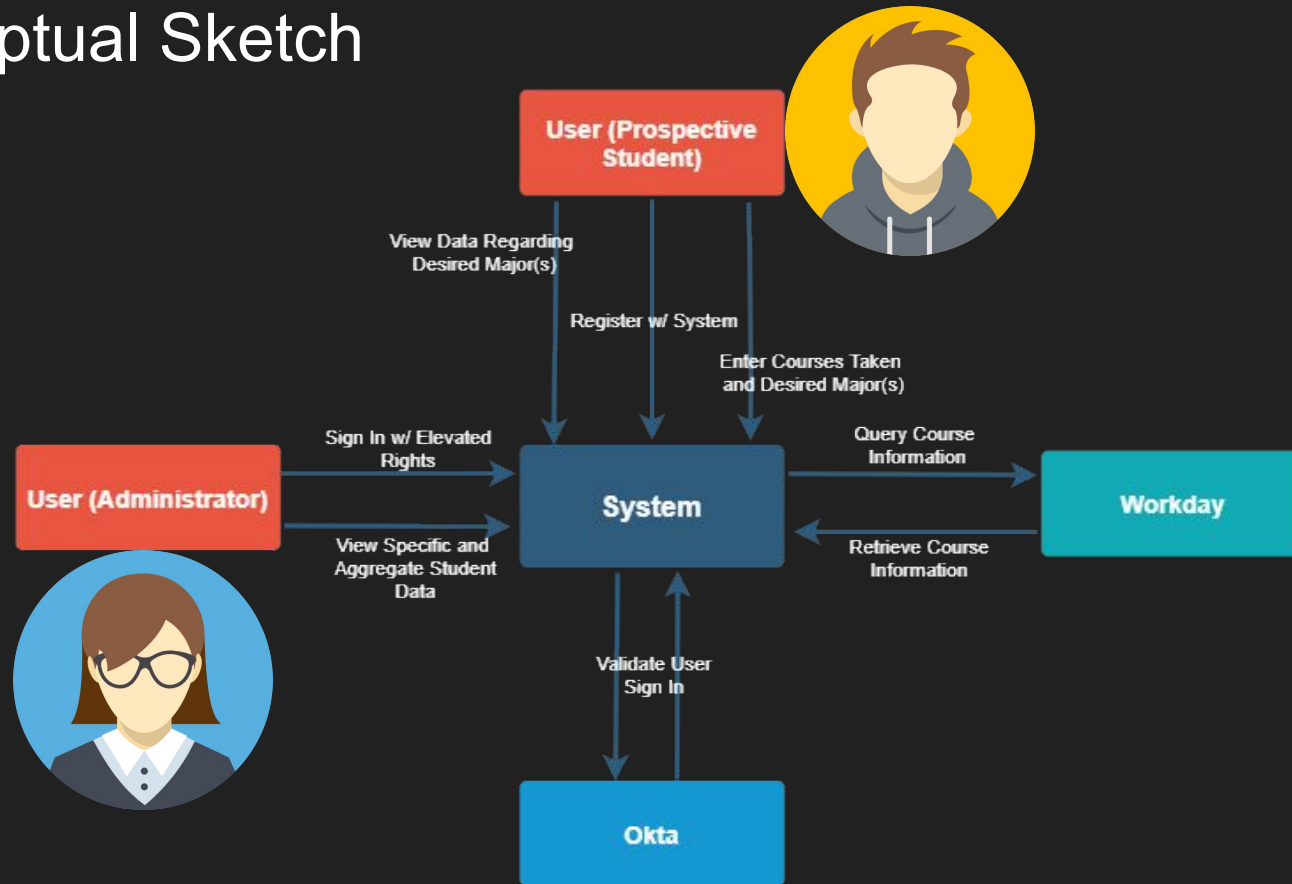


Project Users:

- Transfer Student: Potential Student w/ Credits From Another College
 - The user should be able to choose to create an account with the system to save their inputs.
 - The user should be able to sign back in and continue their work.
 - The user shall be able to enter the school they transferred from and input classes they have taken.
 - The user shall be able to enter their intended major at Iowa State and view a four-year plan of that major with the classes that have successfully transferred crossed off in both a flowchart and tabular format.
- Admin: Admissions/Advisors
 - The admin should be able to sign into an account that has elevated rights.
 - The admin should be able to view specific students' saved data.
 - The admin should be able to view aggregate data regarding students.



Conceptual Sketch



Requirements

- Constraints
 - The project shall use the Iowa State website template.
 - The project shall be able to interact with a workday backend, or a mocked version.
- Nonfunctional Requirements
 - The user interface shall be easy to navigate.
 - The user shall be able to complete a session in under ten minutes.

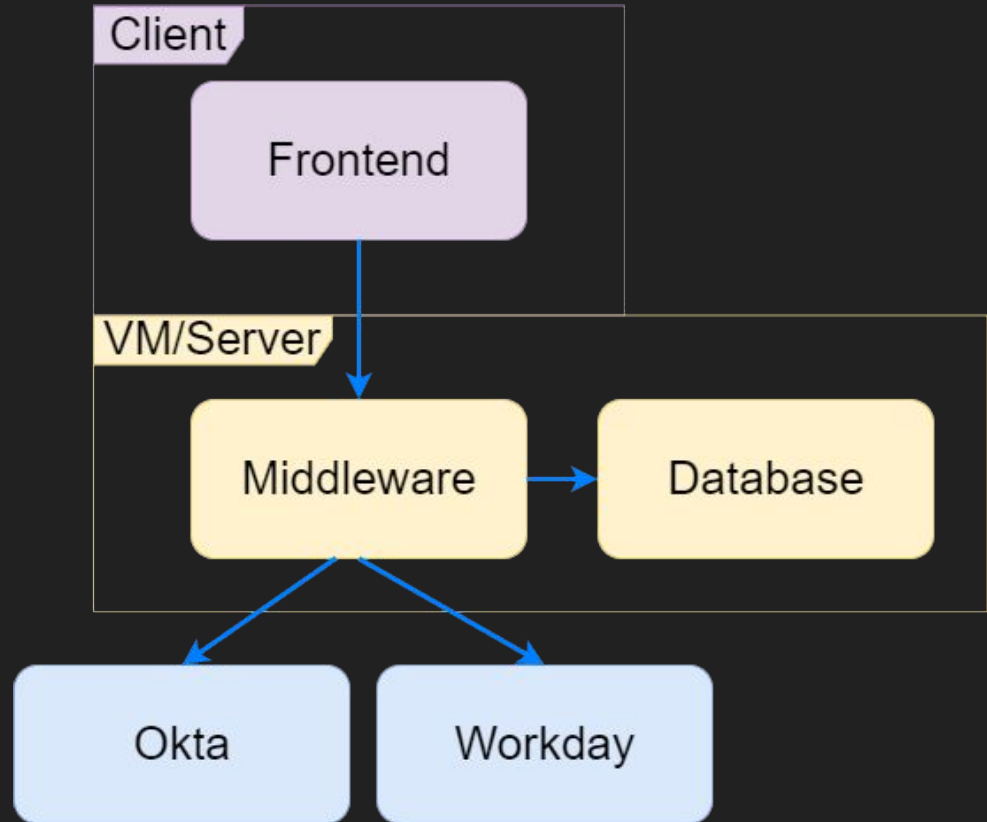
Selected Functional Requirements

- The project shall accept as input transfer courses and grades received from other universities.
- The project shall output a four-year plan of the intended major of the prospective student in table format based on the existing-four year plan for that major.
- The project shall allow administrator users to view data regarding individual prospective student users.
- The project shall allow for guest prospective student users to use the application without creating or signing into an account.

Conceptual Design Diagram

Three main parts

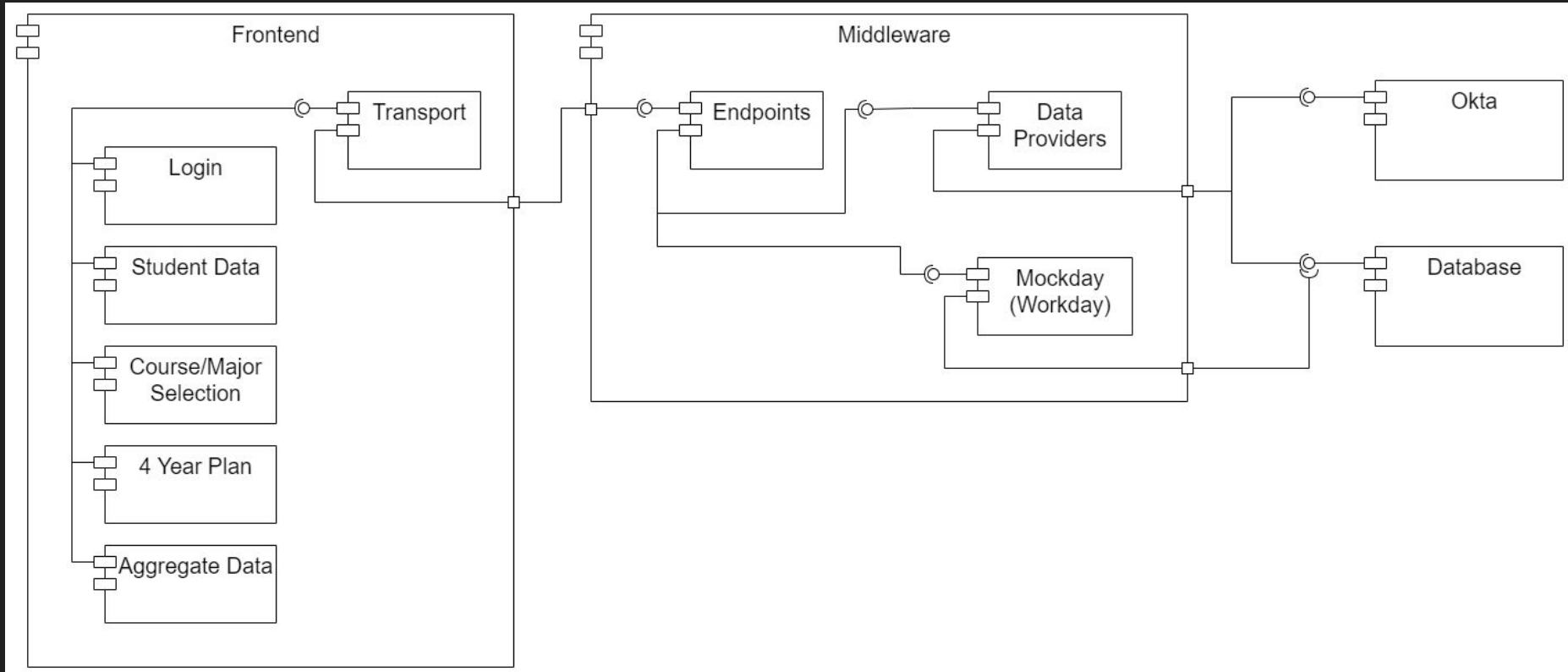
1. Frontend
2. Middleware
3. Backend Services



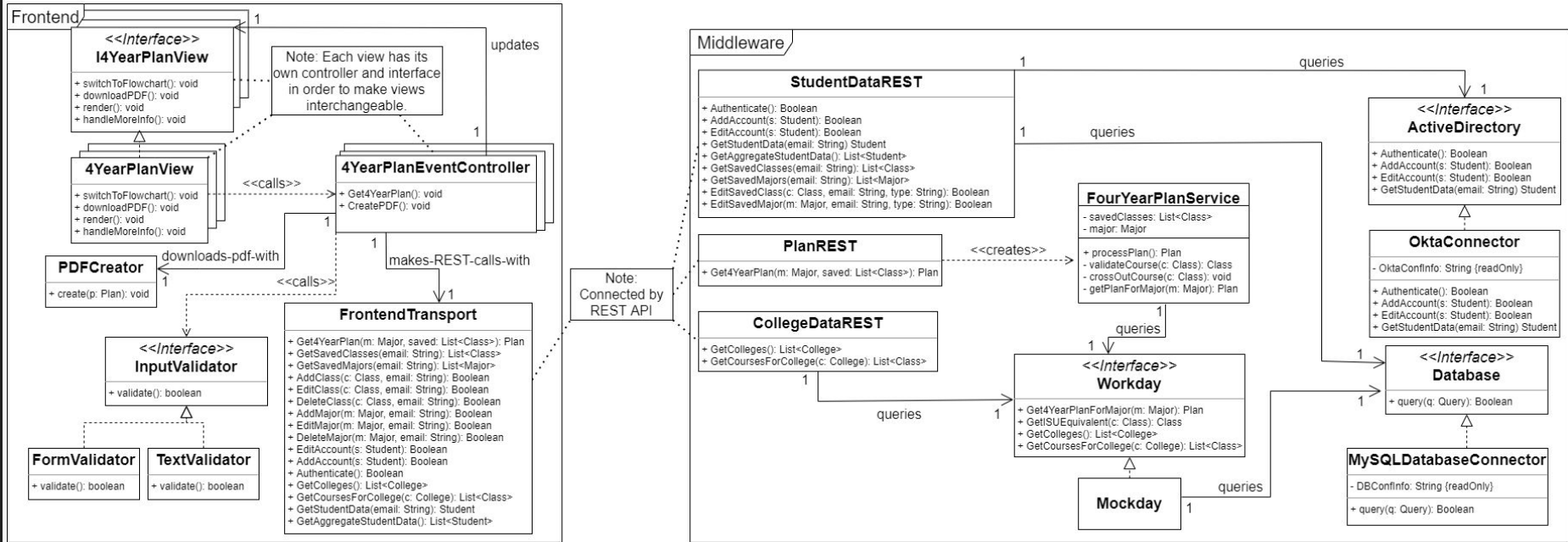
System Design - Frameworks and Technologies

- PHP - frontend and middleware
- HTML/CSS - UI
- Bootstrap ISU templates
- Laravel REST API framework - frontend and middleware communication
- MySQL - database
- PHPUnit - Testing
- Selenium -Testing
- Synk - Testing

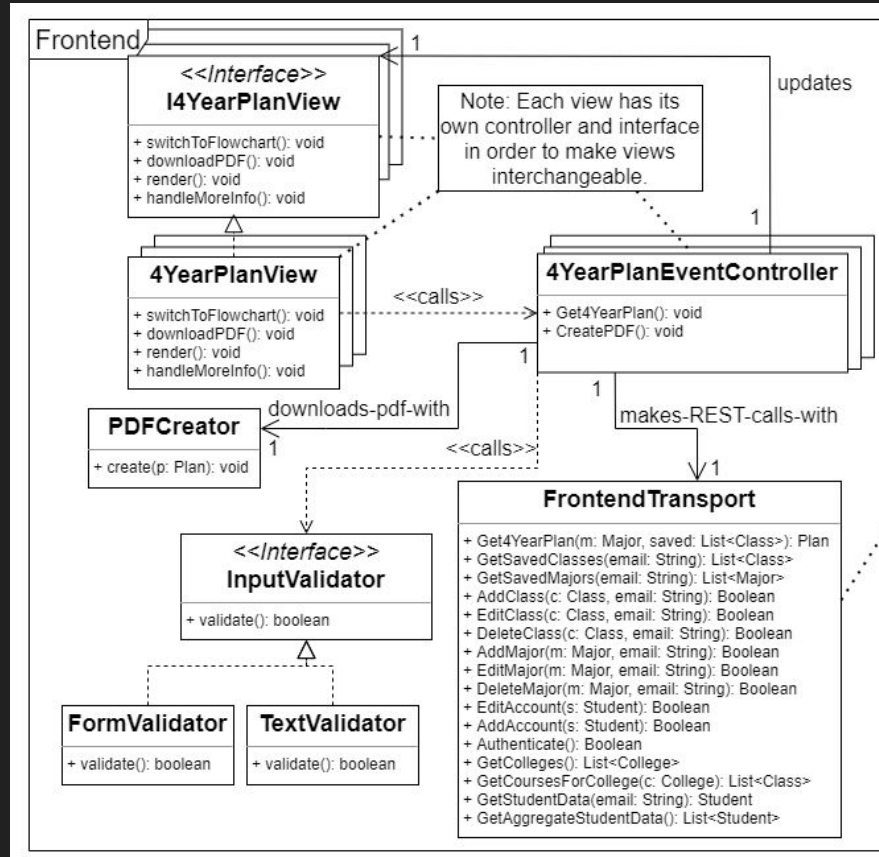
System Design - Component Diagram



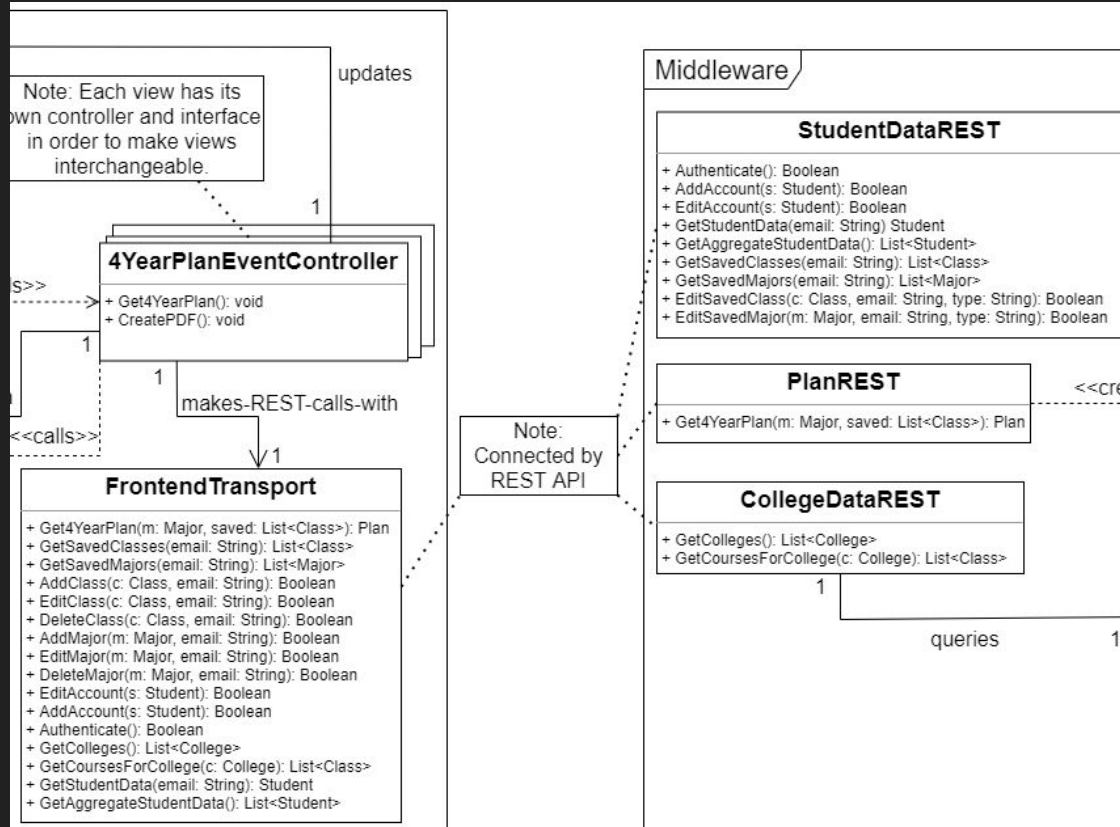
System Design - Class Diagram



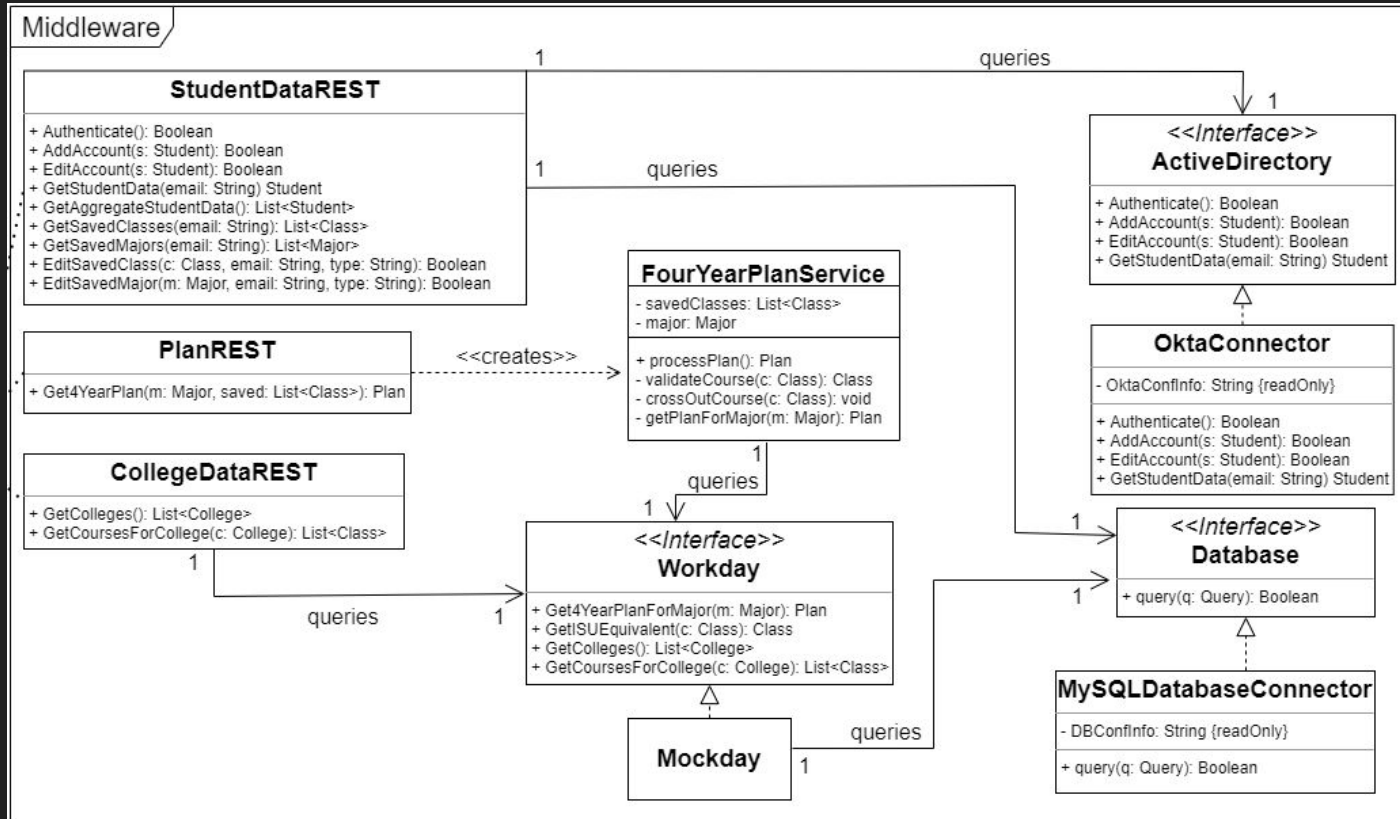
System Design - Class Diagram



System Design - Class Diagram



System Design - Class Diagram



Prototype

[Prototype Demo](#)

Design Complexity

1. Frontend: Page Flow Diagramming, UI/UX Components
 - a. Scientific Principle: Harmony, not discord
2. Frontend & Middleware: Web Development Standards of PHP, Bootstrap, and HTML/CSS
 - a. Mathematical Principle: Using Appropriate Tools Strategically
3. Middleware: JSON Parser for Business Logic
 - a. Engineering Principle: Develop and Understand
4. Backend Service: Workday / Okta API
 - a. Engineering Principle: Understanding
 - b. Scientific Principle: Cooperation, not individualism

| | | | | | |
|--|-----------------------|--|----------------|---|--|
| | Email preset password | | Okta-SSO login | Login with Google | |
| | Other | | | Traditional | |
| | Do not implement | | | Login from scratch | |
| | Save as .txt | | Other | Traditional | |
| | File-based | | File-based | <i>A way for users to save data for later use</i> | |
| | Save as .xml | | | | |

Project Plan - Tasks

- Frontend
 - Create User Interface screen flow diagram
 - Design different components for each page
 - Articulate flow from one page to another
 - Create each individual UI component by picking high risk components first
 - Create UI for entering in transfer courses
 - Create UI for 4 year plan - highest risk
 - Create UI for login - lowest risk
 - Create UI for aggregate data screen
 - Create UI for viewing students that have accessed the site
 - Connect Frontend to Middleware
 - Verify UI design with Iowa State UI/UX domain experts

Project Plan (Cont.) - Tasks

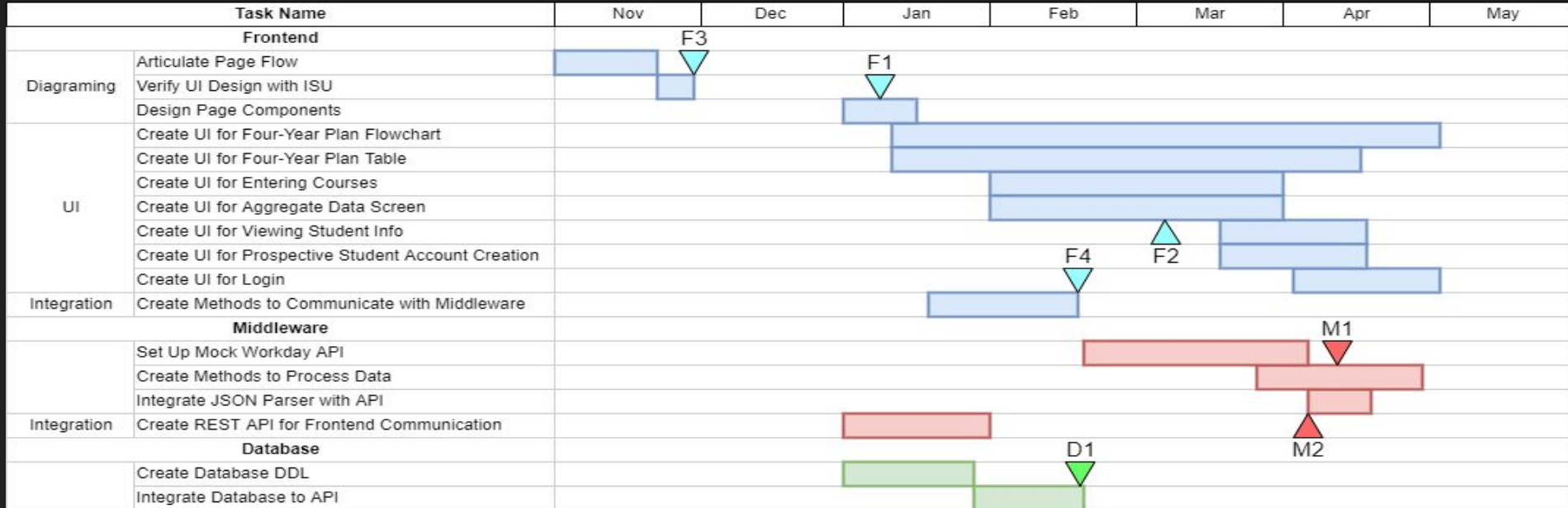
- Middleware
 - Integrate JSON parser for Workday API
 - Create methods to communicate with Workday API
 - Create methods to follow business logic rules to process data
 - Create REST API
 - Set up Mocked Workday API
- Database
 - Set up database
 - Integrate database with API

Project Plan (Cont.) - Selected Risks

- Response Time Constraint Failures
 - Affected Tasks:
 - Connect frontend to middleware
 - Create REST API for middleware
 - Mitigation:
 - REST API
- Integration Issues
 - Affected Tasks:
 - Create each individual UI component by picking high risk components first
 - Integrate Database and Mocked Workday API
 - Mitigation:
 - Research
 - Modular UI components

Project Plan - Milestones and Schedule

- Frontend
 - F1: UI Components
 - F2: Efficient Navigation
 - F3: Screen Flow Interaction
 - F4: Middleware Integration
- Middleware
 - M1: Retrieval Efficiency
 - M2: Mockday API
- Database
 - D1: Database with Workday



Test Plan

- Unit Tests
 - Frontend
 - Selenium Framework for PHP - UI components
 - PHPUnit - Controller, Transport, other individual functions
 - Middleware
 - PHPUnit - Mock Workday, Data Provider, endpoints
- Integration and Interface Tests
 - Postman - Mock Interface API Calls
 - MySQL Workbench - Mock Server
 - PHPUnit - Frontend to Middleware, Middleware to Database

Test Plan (Cont.)

- System Tests
 - Selenium Framework for PHP
 - Synx - Automated security testing
 - Focusing on critical requirements such as entering courses, displaying four-year plan, switching accounts based on user type
- Regression Tests
 - GitLab CI/CD Pipeline - Run all Unit, Integration, and System tests
- Focusing on Testing Pyramid philosophy

Conclusions

- Ready to start developing page components
- Next semester
 - Start UI development
 - Start system development

Individual contributions

- Ben - Requirements, testing plan, prototyped four-year plan page
- Cole - Design complexity, project plan, profile prototype
- Scott - Prototype/html template, UI design, requirements, project plan
- Cameron - Effort estimations, architecture design, flowchart UI design
- Curt - Architecture design, UI prototype, project risks, conceptual design diagram, requirements
- Riess - Requirements, security testing plan, user needs
- Luke - Testing, project plan - schedule