

---

# **EE/CprE/SE 492**

## **Iowa State AI Schedule Companion**

### **Weekly Report #1**

*End of 491 - Monday, Sept. 5, 2024*

**Group number:** ssdec24-08

**Client:** Koby Fowler

**Faculty Advisor:** Abraham Aldaco

**Team Members:**

*Koby Fowler* - Overall Application Leader

*Chandrashekar Tirunagiri* - Frontend Design Leader

*Raghuram Guddati* - Backend Design Leader

*Jacob Paustian* - Artificial Intelligence Research Leader

*Christian Deam* - Meeting Manager

*Anna Huggins* - Team Manager/Communicator

---

## **Weekly Summary**

On August 30th, our team met to give an overview of the current objectives for our team. Emphasis is put on: adapting our application to work with Workday API, adding file upload capabilities for the database, allowing the AI to access file uploads for training, and modernizing the frontend designs. Following this meeting, our designated team members broke down to work on their unique efforts. This week, we have also taken the time to contact our faculty advisor for a meeting.

## **Past Week Accomplishments**

### ***Koby Fowler***

- Completed researching the Workday REST API to explore how it can be used to automatically pull data such as student schedules and class locations. This research aimed to streamline data integration and improve the accuracy and efficiency of accessing academic information.

### ***Chandrashekar Tirunagiri***

- I was working on the file upload feature with Raghuram. We were discussing on the database schema should look based on the front-end design and backend design we came up with the below solution, which we will be implementing.
- Database structure summary - We'll have five main tables:
  - Student Table: Stores student information like ID, name, email, total credits, and GPA.
  - Requirement Table: Data about degree requirements, such as total credits and GPA requirements.
  - Requirement Group Table: Stores information about requirement groups, including group name and minimum credits.
  - Course Table: Contains details of individual courses, like course code, title, credits, and grade.
  - Student Course Table: Establishes a relationship between students and the courses they've taken or are currently enrolled in.
- To extract data from the uploaded degree audit PDFs, we'll:
  - Parse the PDF using libraries like pdf-parse or pdfjs-dist .

## ***Raghuram Guddati***

- Worked with Chan on File Uploading Feature: Developed and integrated a file uploading feature for the DegreeAudit table, allowing users to upload and associate text files with their degree audits. This feature ensures that uploaded files are correctly linked to audit records and stored securely in the system.
- Modified and Expanded Database Schema: Added and adjusted several tables in pgAdmin, including:
  - DegreeAudit: Created to store degree audit records with columns for DegreeAuditID, UserID, and Audits, enabling users to track their degree progress.
  - ClassInfo: Introduced to store detailed information about classes, such as course names, schedules, and credits, improving course management.
  - StudentClassInfo: Added to maintain student-specific details about their enrolled classes, facilitating personalized tracking and management of academic progress.
- Worked on the creation of tables that Chan mentioned as well.

## ***Jacob Paustian***

- Researched how to take in PDFs as a training source for the AI. This is necessary for two reasons:
  - Users will want to be able to upload their own documents for the AI to be able to process.
  - Certain files containing class information may be in PDF form since web-scraping HTML is inefficient with ChatGPT tokens.
- This was largely successful, however, it will take additional research on how to update the AI's data on the fly if a user uploads a new document or conversation (if we choose to support that.)
- I reworked the AI backend to use WebSockets instead of POST requests. This enabled me to send to the frontend a data stream, which is necessary to return text as it is generated by the AI model. Once again, this accomplishes two things:
  - The user no longer needs to wait for the entire answer to be generated for it to be returned.

- It is visually appealing and psychologically indicative to the user that they are speaking to AI.
- I also spent some of my own money on the correct ChatGPT perks. We, unfortunately, were not given the correct tiers of usage by Iowa State first semester, which is entirely our fault. Instead of waiting to resolve the issue, I just gave up five dollars. This allows us infinitely higher rate usages and token usage, which is paramount for development and the final product.

### ***Christian Deam***

- N/A. I was working on reinstalling and figuring out GitHub/GitLab and Visual Studio to properly see the code
  - I reset my PC over the summer

### ***Anna Huggins***

- Made updates to the report templates for 492 and clarity
- Created the Project Review document and began filling out

## **Individual Contribution Tracker**

<b>Name</b>	<b>Individual Contributions</b>	<b>Hours this Week</b>	<b>Cumulative Hours this Semester (492)</b>	<b>Total Cumulative Hours (491/492)</b>
Raghuram Guddati	- Created additional backend features for file upload - Expanded database schema	5.0	5.0	39.5
Christian Deam	N/A	0	0	24.0
Jake Paustian	- PDF research for AI training - Updated backend for use of WebSockets - Corrected ChatGPT perks	5.0	5.0	42.5
Anna Huggins	- Updated the report template - Setup Project Status Report document - Reviewed team meeting messages	2.0	2.0	41.0
Chandrashekar Tirunagiri	- Worked with Raghuram to discuss, plan, and design the new database structure	5.0	5.0	35.0
Koby Fowler	- Research Workday REST API	5.0	5.0	60.0

## **Pending Issues**

N/A.

## **Summary of Weekly Meeting(s)**

***Team Meeting - August 30th, 2024***

- Call to Order

- A team meeting to review individual objectives and redivide team efforts for frontend design, backend development, and AI user-interface design
- Roll Call
  - Present
    - Koby Fowler
    - Chandrashekar Tirunagiri
    - Raghuram Guddati
    - Jacob Paustian
  - Absent
    - Anna Huggins
    - Christian Deam
- Outcomes
  - Team Assignments
    - Frontend - Anna Huggins and Christian Deam
      - Continue modernizing existing user interfaces to maximize visual appeal
      - Improve the current page navigation system
      - Populate more information on web pages
    - Document Upload Functionalities and Manual User Information- Chandrashekar Tirunagiri and Raghuram Guddati
      - Utilize APIs to allow for file uploads from users, i.e.) users can upload their existing degree audits into our application, and our AI can read the class information
      - Allow users to submit custom course information via a form
    - Workday API Management - Koby Fowler and Jacob Paustian
      - Increase app functionality and useability by integrating functionality with Iowa State's new class registration system, Workday
      - Obtain class information, i.e.) seat information, course sections, times, professors, etc.

## **Plans for the Upcoming Week**

### ***Koby Fowler***

- Look into implementing the Workday REST API to integrate features such as pulling student schedules and class locations.
- Research and determine the process for obtaining the necessary permissions to access the Workday API.
- Begin initial development and setup for API integration, including any required authentication and authorization.

### ***Chandrashekar Tirunagiri***

- Implementing the login and sign-up page, especially the backend logic.
- Implementing the file upload feature with Raghuram.

### ***Raghuram Guddati***

- Will work on the logic to verify the password matching (login logic).
- Will work on the tables on pgAdmin to store the information of degree Audits and create relations between tables.

### ***Jacob Paustian***

- Work with Koby on what Workday API would look like.
- Continue research on the fly updates to AI data updates.
- Potentially look into Agents, and how we can get AI to call methods from our custom API.

### ***Christian Deam***

- Figure out the code and how to launch it
- Work on frontend
- Incorporate three.js into frontend

### ***Anna Huggins***

- Meet with Christian Deam
  - Discuss frontend design plans
  - Divide effort for frontend updates
  - Help with any issues launching/running code
- Begin frontend update/redesign efforts
  - Objective: complete one page update/week to share with the team