Slide 1 (Hened): Good morning everyone. My name is Hened and alongside me are Ilyas, Alex, Zayd, and Ovais in team 31. Today we will be presenting our Capstone project which is Tempus II. Tempus II is a continuation of another project, called Tempus, from a team in this capstone class from a year prior. They were tasked by the company Cal & Associates based in the US to create a website which allows employees to upload their resumes, look for projects, essentially an onboarding website. Now with Tempus II, we were tasked to add another feature which we'll discuss in this presentation.

Slide 2 (Hened): The issues that Cal & Associates had were of the following:

- Inadequate communication between employees, supervisors, and client representatives which caused delays in timesheet approval.
- Furthermore, there was difficulty in managing user roles and permissions which impacted the accessibility of system functionalities.
- And finally, they were using an external timesheet web application that is not a part of the onboarding system.

Slide 3 (Alex): The solution to address these issues is to build and implement a timesheet management system that will be integrated within the existing onboarding system built for CAL & Associates. The solution is to:

- Implement an easy-to-use, digital timesheet module for efficient and accurate tracking of hours worked.
- Next, to facilitate communication between employees, supervisors, and client representatives through notifications and timesheet approval workflows.
- Also, to integrate a user-friendly interface for managing user roles and permissions.
- And lastly to generate cost and billing reports so that clients can keep track of their expenses.

Slide 4 (Ovais): As a result of us taking over from the previous team, we have decided to use the same tech stack that they had been using because we are just building off of what they have built. This tech tech stack mainly consists of NX, Angular, NestJS and Postgres. Me and Alex will discuss what these technologies are and how they are incorporated in the Tempus application

Slide 5 (Ovais) : The 1st technology that we are going to cover is called NX. NX is a suite of development tools for creating monorepos. A monorepo is a single repository which contains the code for multiple projects. In our case, NX was used to create the structure of the Tempus application. Furthermore, the various projects within the Tempus monorepo represent the various aspects of the Tempus system like the frontend, backend server, the database and other aspects

Slide 6 (Alex): The second technology that we are going to cover is Angular. For the front-end we are using the Angular web framework. Angular is a web application framework for designing dynamic, responsive, and scalable single-page applications using a structured, component-based architecture. Angular uses the Typescript language to provide a set of libraries and tools to help developers write efficient and maintainable code. There are many useful features in Angular that help the development process such as data binding, routing and forms handling. We are using Angular to create the various views and user interfaces for the application. We are also using it to provide functionality such as data access and communication with the back-end.

Slide 7 (Ovais): Next is NestJS which is a Node.js framework for creating scalable serverside web applications with TypeScript. It uses controllers to handle incoming HTTP requests and services to create the business logic of the application. NestJS was mainly used in the application in order to create a REST API that handles the requests from the angular frontend so that the server can get and send data from the Postgres database.

Slide 8 (Ovais): The last technology I will talk about is Postgres. Postgres is a database management system for creating databases which will store user and application data. We used Postgres along with TypeORM to simplify database interactions by using object oriented programming concepts to store information about the timesheet records like who submitted the timesheets and when.

Slide 9 (Zayd): Demo

Slide 10 (Zayd): Difficulty in integrating Angular with NestJS and Postgres, potentially causing inconsistencies and difficulties in data flow between frontend, backend, and database. Performance issues: Scalability and performance concerns may arise when handling large amounts of data or concurrent users, affecting the responsiveness and reliability of the system. Security vulnerabilities: Ensuring proper security measures are in place to protect sensitive data, including user authentication, authorization, and data encryption, while also addressing potential vulnerabilities in Angular, NestJS, and Postgres. Maintaining code quality: Ensuring consistent code quality and following best

Slide 11 (Hened): Some non-technical issues we've encountered include managing the SEG4910 course with the workload of other courses during our semester. We had problems dividing the work efficiently as well as communication effectively. Also, in the beginning of the semester, we were having a hard time with engaging stakeholders and finding the right project for us. And finally, listening to colleagues at times caused some issues but overall, we were able to overcome them and move forward.

Slide 12 (Ilyas):

Because of the issues that we faced during the course of the project, we came up with various solutions in order to overcome them.

When we faced technical issues, we first attempted to debug the code ourselves. If that didn't work we would then review similar code completed by the previous team since some of their code may have accomplished what we wanted to do. If after all this we were still running into issues, we did additional research on potential solutions, and then reached out to our team members for help. The code base we worked on was pretty complex so there was a lot of things to learn regarding navigating through all the code, and modifying it accordingly. But we managed to make progress by learning as we progressed through the coding.

We also tried to make sure we gave tasks to members who were best suited for them. For instance, we assigned the UI designs task to those of us who had previous experience with using Figma to design. We also emphasized collaboration between us whenever we got stuck and met regularly to ensure we were staying more or less on track with reaching our goals

Slide 13 (Ilyas):
In the future we have numerous plans in order to complete the application. Namely, deploying it to production mode, implementing the bill and cost reports to allow clients to keep track of their expenses, ensure that timesheets cannot be edited after they have been billed and audited, adjust our design according to the clients requests, create a notification system so that users

will be notified when there are updates related to timesheets that they need to know (like it needs their approval, or it has been approved), and ensure that the UI matches the company colors.

Slide 14 (Ilyas):

That concludes our presentation regarding our Capstone Project Tempus II. Are there any questions for us at this time?