**Intro (Cover Slide):**

* I’d like to introduce our team**:** Matt Tennis, Connor Williams, Ryan Mahoney and Chris Gonsalves.
* Together we created ESS the Employee Scheduling System.
* The Employee Scheduling System (ESS) is a system used to provide simple and efficient means for an employee to request time off from appointed supervisors.
* ESS allows appointed supervisors to manage scheduling request once they're are submitted to ESS by user.
* The ESS interface provides a streamline process for users requesting a schedule change and supervisor’s approval or denial of schedule changes.
* The system includes secure login, logout functionality in addition to the primary scheduling applications.

**Requirements (Slide 2):**

**Login** – All users, Employees and Supervisors, must supply valid login

credentials (EmployeeID and password) to be authorized to access and use the

system. Upon doing so, the user will have created a session with ESS, where a

user can modify database contents through normal usage. Valid login will

direct the user to his or her appropriate activity based on the user’s class.

**Logout** – All users must have clear and immediate access to a Logoff button in

order to gracefully and securely close the connection with ESS. Resources

allocated to a user session must be terminated in an orderly fashion as to

eliminate potential software bugs. Every form or interface must have a clearly

marked Logoff button.

**TimeOffRequest** – Employees must be able to supply a time off request in

the Time Off Request form. Employee will select dates via the calendar GUI.

Radio buttons enable the Employee to indicate the reason (and weight) of his

or her request. The user can then submit or logout from that form. Then the

system sends time off request to the database.

**RequestResponse** – Supervisors must be able to view the time off requests

that have been submitted in a scroll box queue. The queue will have

highlighted regions that correspond to the reason (or weight) supplied by the

user’s time off request. The Supervisor can then approve, deny, or logout

from this window. Approvals and denials modify database contents and

update the queue, while logout will terminate the session gracefully.

**Interactions(Slide 3):**

**Object Design and Interaction**

**Class Diagram**

This class diagram represents a portion of the structure of the ESS. It shows the attributes and methods.

**Class interaction**

These diagrams shows association, multiplicity and navigability.

- Here is a GENERAL association where each child class has is-a relationship with the parent

class. UserControl and TORControl inherit attributes and operations from Control.

- Here is multiplicity, the USER has a 1 to MANY relationship with TOR(Time Off Request). For example, one USER can submit many TOR’S.

- This interaction also shows that ESS supports navigation from TOR to USER.

**Implementations(Slide 4):**

Explain how state chart mapped to code.

**Conclusion(Slide 5):**

* In closing, the team worked **synchronously** and **asynchronously** in collaborating on the **ESS** project. One of our strengths in working on this project was using time efficiently. Managing and committing to deadlines. Uniformity and consistency was an area that we tried to improve on deliverable to deliverable.
* This presentation outlined the program's **functional** and **non-functional requirements**, secure login/ logout, User Time Off Request, Supervisor's Request Response
* **Object designs** and **interactions**, **class diagrams** and **class associations**
* And lastly, **implementation**, transforming our **Object Design** and **Interaction** into source code

**Questions / Concerns**