**Description:** The client is contracting our company to design and implement an employee scheduling system. Employees must be able to request time off [and view?], and employers must be able to confirm or deny that request.

**Brief:**

1. Users/Actors: Employer, Employee
2. Functional Requirements:
3. Login and Logout functionality for both users
4. Employee Request and Employer Confirm/Deny with a graphical display
5. [We haven’t really addressed much of the actual scheduling. Unless we plan on setting up static data and just making request accommodations, I think we should consider replacing the request/confirm function with a scheduling algorithm.]
6. Non-functional Requirements [OPTIONAL]
7. Secure login and auth with secure hash algorithm
8. Employee & Employer GUI should implement Google Calendar API for better business integration.
9. [High performance and reliable sorting algorithm. Maybe include a weight system so more senior employees / employers? can get precedence for conflicting requests (see Stable Marriage algorithm).
10. Use Case Diagram

**Employer** and **Employee** are the two users/actors that interface with the system. Both **Employer** and **Employee** must interact with only the **Login** UI elements until the actor enters valid credentials. **Employees** may **Request** time off, to which an **Employer** may **Approve** or **Deny** the **Employee’s Request**.