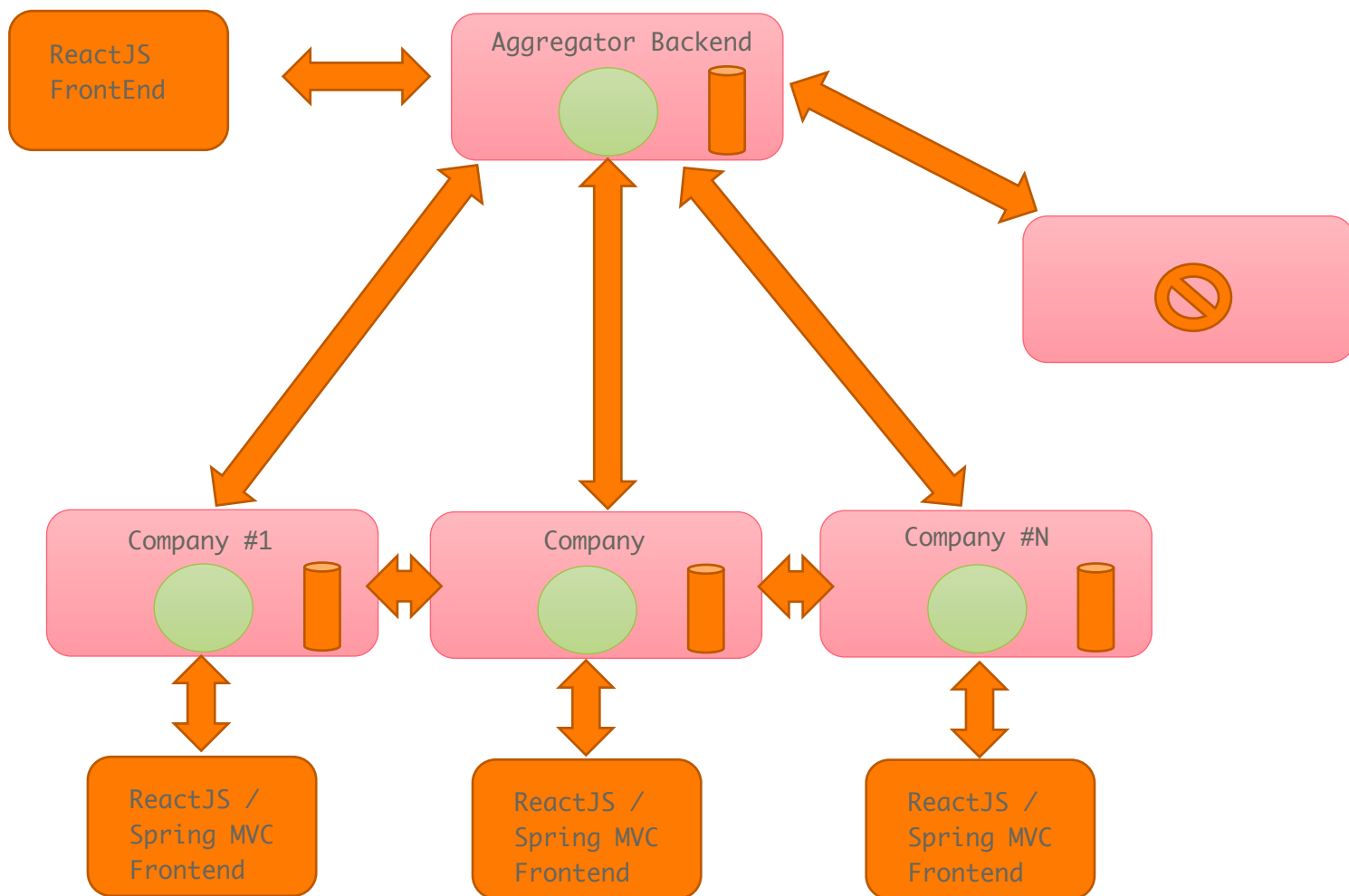


# INFO 6250 – Web Tools Tech.

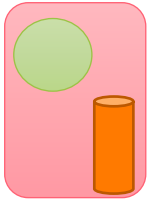
## Aggregator System (Updated March 29, 2023)

AIM: Create a system, which and aggregate various entities data. Perform business logic and display the results the user.

Suggested Architecture:



## Terminology:



- Spring BOOT + Spring REST + Spring JPA + Database



- Entity is not available or Connection is not available



- Front End Technology communicating with Spring Boot Application via REST API



- Front End Technology – React JS / Spring MVC

## Use Cases

- 1) Company #N
  - a. Admin Role
    - a) View Managers
    - b) Add Managers
  - b. Manager Role
    - a) CRUD Operations for data
- 2) Aggregator
  - a. Guest User / Registered User
    - a) Display Aggregated information to a user, which is retrieved from various companies and/or 3<sup>rd</sup> party API (weather )
  - b. Admin
    - a) View Reports
    - b) View Usage (Transaction done via system)
- 3) Architecture
  - a. Secure while accessing REST API
  - b. Aggregator will communicate to all companies available
  - c. Companies can talk to each other via a secure mechanism.
  - d. Backed End: Controller <-> Service <-> Repository <-> Entity
  - e. Front End: ReactJS / Spring MVC

### Note:

- The above use case are guidelines for expected functionality. More use cases done, will secure a higher grade.
- B+, a fully functional project.
- B-, project does not work.
- A/A-, additional logical features
  - Ability to demonstrate a flexible algorithm
  - Display valuable features about your system
  - Ease of use of system – System should have a logical flow.