1. Company Routes
   1. List all companies
      1. path: /company
      2. Method: GET
      3. methodName: companyList
   2. Create new company
      1. path: /company
      2. Method: POST
      3. methodName: createCompany
   3. Get a company by floor
      1. path: /company/floor/{floor}
         1. Throws an exception if the value passed in for floor is not an int 1-8.
      2. Method: GET
      3. methodName: findByFloor
   4. Get a company by company name
      1. path: /company/name/{name}
      2. Method: GET
      3. methodName: findByCompanyName
2. Employee Routes
   1. Get an employee by user ID
      1. path: /employee/user/{id}
      2. Method: GET
      3. methodName: findByUserId
   2. Create new employee
      1. path: /employee
      2. Method: POST
      3. methodName: createEmployee
   3. Delete an employee
      1. path: /employee/{id}
      2. Method: DELETE
      3. methodName: deleteEmployee
   4. Update an existing employee
      1. path: /employee/{id}
      2. Method: PUT
      3. methodName: updateEmployee
   5. Get a list of employees by company name
      1. path: /employee/company/{companyName}
      2. Method: GET
      3. methodName: findByCompanyName
3. Building Log Routes
   1. Get list of building log entries by user ID
      1. path: /log/user/{userId}
      2. Method: GET
      3. methodName: findLogByUserId
   2. Get list of building log entries by companyName
      1. path: /log/company/{companyName}
      2. Method: GET
      3. methodName: findLogByCompanyName
   3. Get master log of all building entries
      1. path: /log
      2. Method: GET
      3. methodName: getMasterBuildingLog
   4. Create new entry in building log
      1. path: /log
         1. Throws exceptions for values passed in for logTime is outside of workingHours for building.
      2. Method: POST
      3. methodName: newLogEntry