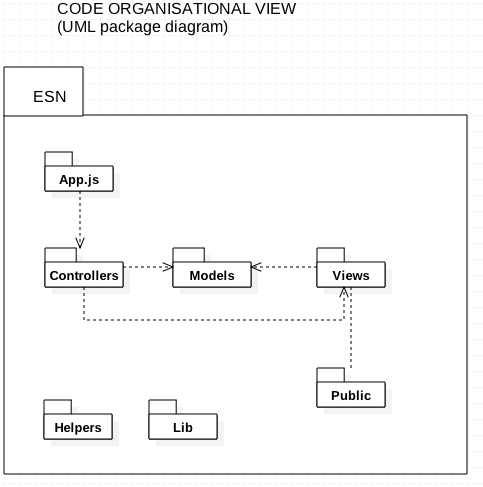
# Emergency Social Network (ESN) Team SV-5

The goal is to provide civilians with a social network that they can use during emergency situations. The system is different from other existing social networks because it is specifically designed to effectively support small communities of civilians seriously affected in case of natural disasters like earthquake, tsunami, tornado, wildfire, etc.

## Technical Constraints

* Solution will be deployed on Cloud platform
* Clients connect to the app server via their mobile phone browsers.
* No native app, only web stack (HTML5, CSS, JS) on mobile browser will be supported)
* System has a RESTful API - should function with and without UI
* System supports real-time dynamic updates

## High-Level Functional Requirements

* User should be able to join community
* User should be able to share status
* User should be able to chat publicly
* User should be able to chat privately with another user
* User should be able to search information
* Coordinator should be able to post announcement
* Administrator should be able to edit user profile details
* User should be able to edit his profile.

## Top 3 Non-Functional Requirements

Usability > Testability > Maintainability

## Architectural Decisions with Rationale

* Client-Server as main architectural style
* Server-side JS (node.js) for small footprint and performance
* Lightweight MVC on the server side using **express** framework
* RESTful API provides core functionality and reduces coupling between UI and back-end
* Web-sockets allow event-based fast dynamic updates
* Database MongoDB

## Design Decisions with Rationale

* Encapsulate data and behavior in models for easy testing and better modularization
* **Adapter** design pattern to substitute a test database for the production database during testing

## Responsibilities of Main Components

* **models:** encapsulate data and behavior for entities of the system
* **controllers**: accepts input and converts it to commands for the model or view
* **views**: output representation of information. Generates new output to the user based on changes in the model