frontend modules

listing.client.
                                 • handles view state

    * routes: relates url paths to views' states   for each path, there is a view

    * controller: controls scope variables in response to factory method calls.
    factory: creates 'Listing' and defines methods which pass back Http calls
            to urls mapped out/supported by server backend.

app.js
- creates 'listings' module
- adds in dependencies
- configures app to use html 5 and redirect to /listings for unmapped route requests.
- starts app in state 'listings.list'

all html: views associated with each feature.
        index.html dynamically nests the other html views inside
                 itself for state changes.

Backend

                            • describes url routes to server

app.js : calls express.js to configure server, then starts server on config. port

express.js: connects to mlab, inits the app module, starts logging, sets
        up middleware use, serves html files in public & client,
        set up router to path on /api/listings/...., sets up
        server wrapper for google maps, and redirects to index.html
        for all other unspecified paths.

coord.server.controller: sends address from request body to google
                api to get Lat/Long coordinates.
                Passes them into request. results. Acts as
                middleware for the map. function.

listings.server.controller: handles mongoose queries to listings
                collection in mlab. methods to manipulate
                database.

listings.server.model: defines Listing schema to mongoose. defines
                a 'pre' method for the schema.

listings.server.routes: router is set up to associate a url and
                http request type with a behavior defined
                in listings.server.controller