

app.js

creates angular module called 'listings', dependent on nothing

creates an angular module called 'directoryApp' dep. on 'listings'.
var app

listingFactory.js

Creates a service & new instance of service called 'Listings'
from 'listings' module, returns a \$q promise list of
functions which return Http calls (CRUD) to
'/api/listings.'

e.g. \$Http.get('http ... /listings');

listingController.js

Create scope functions that exposes listingFactory.js
methods.

\$scope vars:

- listings.

- detailedInfo (default null)

- addListing {

- \$scope.newListing

- {

- deleteListing.

- showDetails

Frontend.

backend

config *

vars

config file with settings

Mongoose

express.js *

starts the server by running express.js code.

.init() {

connects to db

inits app through express()

enables logging for http req&res

enables "body parsing middleware."

*google maps coordinates.

{ // serve static files

{ // use listings router for req. to api

{ // go to humpcase for req. not spec.

{ returns app created by express that listens on
the config. port

controllers

coordinates:

takes req/res where body of req contains address. this address is queried in google api for coordinates. they are saved as req.results
if the body is empty of address, then next();

listings: *

Mongoose queries for CRUD.

models

listing model & pre() method

Listings

Routes

Listings:

Middle were from Listings controller.

↳ finds by ID and returns obj Listings as req. listing.

On get: ^{by ID} .read() from Listings controller
put .update coordinates.
delete .delete

'/' on get Listings.list

on post: Listings.create()