Model

1 - User Dana
---usermane
---password
---email

3 - Way (tied to user) Remi
---_id
---profileID
---startLocation
---endLocation
---[wayerz]
---timestamp

2 - Profile Dana
---display name [required]
---email
---bio
---user_id (points at user)
---timestamp

4 - Message collection Darcy
---user
---user
---[messages]
---text
---timestamp

5 - Review   Anna
---rating
---comment
---timestamp
---tripID
---user ID
---reviewed user ID

6 - Location Remi
---number: { type: String },
---prefix: { type: String },
---street: { type: String },
---type: { type: String },
---city: { type: String },
---state: { type: String },
---zip: { type: String, required: true },
---timestamp: { type: Date, default: Date.now }

Routes

User | Profile | Way | Message | Review

- User
POST: 'api/user' --> sign up/in
PUT: 'api/user' --> update username or password
DELETE: 'api/user/' --> delete account

- Profile
POST: 'api/profile' --> create profile
PUT: 'api/profile/:id' --> update profile
GET: 'api/profile?search' --> search for profile

- Way
POST: /api/way
GET: /api/way?location=   UNAUTH
GET: /api/way AUTH <-- auth users way
GET: /api/way/:id
PUT: api/way/:id
DELETE: /api/way/:id

- Review   ALL AUTH
POST: /api/way/:wayid/wayerz/:wayerzid/review
  <-- post a review
GET: /api/review   <-- your own
PUT: /api/review/:id   <---edit your review
DELETE: /api/review/:id
GET: /api/way/:wayid/wayerz/:wayerzid/review
  <-- when considering a trip, list of reviews about them

- Message ALL AUTH
POST: /api/profile/:profileid/message
PUT: /api/message/:messageid
GET: /api/message
   <-- frontend: org inbox etc.... blanket delete
DELETE: /api/message/:messageid
GET: /api/message/:messageid

Stretch DV

- Mind map
- Wireframes
- Prototype

Stretch Feature

MVP Deliverables

- Full CRUD REST API
- Docs
- Passed Tests

Tests

- resources
- servers (features)