* Posts
  + api/posts/test
    - GET
    - Test route only
    - Public
  + api/posts/
    - POST
    - Used to create a new post
      * Props passed in:
        + Text – REQUIRED
        + Name
        + User ID – Logged on user
    - Private – uses passport
  + api/post/
    - GET
    - Will return all posts
    - Public
  + api/posts/:id
    - DELETE
    - Used to delete a post by id
      * Passed in from req.params.id
      * Can only be deleted by User who made post
    - Private – uses passport
  + api/posts/like/:id
    - POST
    - Used to like a post
      * Post is passed in from req.params.id
      * Currently logged on user (req.user.id) will be added to likes array of post
      * Can only be added to array once
    - Private – uses passport
  + api/posts/unlike/:id
    - POST
    - Used to unlike a post
      * Post is passed in from req.params.id
      * Currently logged in user (req.user.id) will be removed from likes array of post
    - Private – uses passport
  + api/posts/comment/:id
    - POST
    - Use to add a comment on a post by id
      * Post is passed in from req.params.id
      * Props passed in for comment:
        + Text – REQUIRED
        + Name
        + User ID – Logged on user
      * Comment will be added to comments array of post
    - Private – uses passport
  + api/posts/comment/:id/:comment\_id
    - DELETE
    - Used to delete a comment from a post
      * Post is passed in from req.params.id and comment is passed in from req.params.comment\_id
      * If comment is in comment array of post, it will be removed
    - Private – uses passport
* Profile
  + api/profile/test
    - GET
    - Test route only
    - Public
  + api/profile/
    - GET
    - Returns current user’s profile
      * Profile is passed via req.user.id
    - Private – uses passport
  + api/profile/all
    - GET
    - Will return all profiles
    - Public
  + api/profile/handle/:handle
    - GET
    - Will return profile using handle id
      * Profile handle id passed in from req.params.handle
    - Public
  + api/profile/user/:userid
    - GET
    - Will return profile using user id
      * Profile user id passed in from req.params.user\_id
    - Public
  + api/profile/
    - POST
    - Used to create new or edit existing profile
      * Fields passed from front end to API route:
        + User – passed in from req.user.id
        + Handle
        + Location
        + Phone
        + Specializations
        + Website
        + Price Range
        + Certifications
        + Social media object

Youtube

Twitter

Facebook

Instagram

Linkedin

* + - * Search Profile collection
        + If profile already exists under user id, update
        + If profile does not exist, create
    - Private – uses passport
  + api/profile
    - DELETE
    - Used to delete user and profile
    - Private – uses passport
* User
  + api/users/test
    - GET
    - Test users route
    - Public
  + api/users/register
    - POST
    - Used to create a new user
      * Fields passed in (ALL REQUIRED):
        + First name – must be between 2 and 30 characters
        + Last name – must be between 2 and 30 characters
        + User name
        + User Type
        + Email
        + Password – must be between 6 and 30 characters
      * Checks if email already exists in user collection
      * Uses bcrypt to hash password, which is stored in database
    - Public
  + api/users/login
    - POST
    - Used to login a user and return jwt token
      * Fields passed in from frontend (ALL REQUIRED):
        + Email
        + Password
      * Use bcrypt to compare passed in password with database
      * Passes back jwt (jsonWebToken) used for authentication
    - Public
  + api/users/current
    - GET
    - Return current user
    - Private – uses passport