FRESH PROJECT

* An Authentication which includes
  + Register
  + Log in
  + Reset password
  + Me
  + Log out
* Discussions page
  + Contains 10 discussions with their last post and user
  + Contains the count of the posts
  + Contains the sum of the count votes of each posts of the discussion
  + Order by last post date
  + Order by last created date
  + Order by activity ( number of posts and votes )
  + Scope by channel
  + Scope by closed and opened discussions
  + Scope by no posts yet discussions
* Discussion page
  + Contains last post and its user (removed)
  + Same as discussions page
  + Discussion Owner
* Posts page
  + Contains 10 oldest posts with their users and votes
  + Contains the count of votes
  + Contains if the authenticated user liked the post
  + Check if the current user can edit
  + Check if the current user can delete.
* Create discussion page
  + Authentication required.
  + Requires :
    - Title
    - Channel ID
    - Content
  + It stores title, slug and channel\_id in the discussions table.
  + It stores content and user\_id in the posts table. (ip, user\_agent or some other metadata)
* Create post page
  + Authentication required.
  + Check if the discussion is open.
  + Check if the parent\_id belongs to the discussion ☺
  + Requires
    - Parent\_id (if it’s a reply to a specific post ☺)
    - Content
    - Discussion\_id (or maybe we’ll work with discussions/{}/posts)
  + Same as above in the discussion store page.
* Update post page
  + Authentication required.
  + Check if the authenticated is the owner.
  + Check if the discussion is open.
  + Check if the post belongs to discussion
  + Requires
    - Content
  + Impossible to change the parent id.
* Delete post page
  + Authentication required.
  + Check if the authenticated is the owner.
  + Check if the discussion is open.
  + Check if the post belongs to discussion
* User page
  + Authentication required
  + Shows user information
    - Name
    - Username
    - Email (is the user allows that)
    - Avatar
    - Number of posts
    - Posts
* Like a post
  + Authentication required
  + Check if the post belongs to discussion
  + Check if the user already liked the post
* Delete like post
  + Authentication required
  + Check if the user is the owner of the vote
* Update(title, closed\_at) the discussion
  + Authentication required
  + Check if the user is the owner
  + Check if the discussion is open.
* Delete the discussion
  + Authentication required
  + Check if the user is the owner
* Channels
  + Latest channels
  + Most used channels (scopes)
  + Pagination

NOTES:

1. I have to close ***discussions/{discussion}*** route and add discussion data to ***discussions/{discussion}/posts***
2. I have to add user agent, IP address and deleted at columns to personal access tokens table.
3. Maybe we have to make a separate resource file for last post in discussion resource.