FORMAT: 1A HOST: http://localhost

digidiet

digidiet API provides RESTful access to digidiet User and Recipe resources.

Group User

User related resources of the **digidiet API**

Index and Store [/user]

Display a list of users or store a new user.

List all Users [GET]

- Request`digidiet/user`
- Response 200 (application/json) `[{"id": 1,`
  ```
  "username": "admin",
  "password": "password",
  "name": "Admin Istrator",
  "about_me": "I administrate",
  "avatar": "admin.png",
  "location": "adminville, AD" },
  {"id": 2,
  "username": "rainbowkitty",
  "password": "raaiiinbowww",
  "name": "Rainbow Kitty",
  "about_me": "raaiiinbowww",
  "avatar": "rainbowkitty.jpg",
  "location": "raaiiinbowww" }
  ]
  ```

Store a User [POST]

- Request (application/json) `{ "username": "admin",`
  ```
  "password": "password",
  "name": "Admin Istrator",
  "about_me": "I administrate",
  "avatar": "admin.png",
  "location": "adminville, AD" }
  ```
- Response 201

Create [/user/create]

Show the form for creating a new user

Create a User [GET]

- Request `digidiet/user/create`
- Response 200 `makes view: digidiet/register`

Show, Update, and Destroy [/user/{id}]

View a user profile by ID, update the specified user in storage, and delete a user + Parameters + id (required, number, `1`) ... Numeric `id` of the User to display. Has example value.

Show a User [GET]

- Request `/digidiet/user/1`
- Response 200 `makes view: digidiet/profile with user: 1`

Update a User [PUT]

- Request (application/JSON) `{ "username": "moderator",`
```
"password": "password",
"first_name": "Notan",
"last_name": "Admin",
"about_me": "just a moderator now",
"avatar": "thefall.jpg",
"location": "moderation, CA" }
```
- Response 200

Destroy a User [DELETE]

- Request `digidiet/user/1`
- Response 200

Edit [/user/{id}/edit]

Show the form for editing the specified user + Parameters + id (required, number, `1`) ... Numeric `id` of the User to display. Has example value.

Edit a User [GET]

- Request `/digidiet/user/1/edit`
- Response 200 `makes view: digidiet/edit with user: 1`

Group Recipe

Recipe related resources of the **digidiet API**

Index and Store [/recipe]

Display a list of recipes or store a new recipe.

List all Recipes [GET]

- Request `digidiet/recipe`
- Response 200 (application/json) `[{"id": 1,`

  ```
  "title": "fish sticks",
  "description": "fish sticks",
  "ingredients": "fish sticks",
  "instructions": "cook thoroughly",
  "author_id": 1},
  {"id": 2,
  "title": "chicken nuggets",
  "description": "chicken nuggets",
  "ingredients": "chicken nuggets",
  "instructions": "cook thoroughly",
  ```

```
   "author_id": 1}
   ]
```

Store a Recipe [POST]

- Request (application/json) `{ "title": "tater tots",`
   `"description": "bumpy potatoes",`
   `"ingredients": "bumpy potatoes",`
   `"instructions": "cook thoroughly"}`
- Response 201

Create [/recipe/create]

Show the form for creating a new recipe (must be authenticated)

Create a Recipe [GET]

- Request `digidiet/recipe/create`
- Response 200 `makes view: digidiet/recipes/add`

Show, Update, and Destroy [/recipe/{id}]

View a recipe by ID, update the specified recipe in storage, and delete a recipe + Parameters + id (required, number, `1`) ... Numeric `id` of the Recipe to display. Has example value.

Show a Recipe [GET]

- Request `/digidiet/recipe/1`
- Response 200 `makes view: digidiet/recipes/infor with recipe: 1`

Update a Recipe [PUT]

- Request (application/JSON) `{ "title": "cheese fish sticks",`
  `"description": "cheese fish sticks",`
  `"ingredients": "cheese, fish sticks",`

`"instructions": "cook thoroughly"}`

- Response 200

Destroy a Recipe [DELETE]

- Request `digidiet/recipe/1`
- Response 200

Edit [/recipe/{id}/edit]

Show the form for editing the specified recipe + Parameters + id (required, number, `1`) ... Numeric `id` of the Recipe to display. Has example value.

Edit a Recipe [GET]

- Request `/digidiet/recipe/1/edit`
- Response 200 `makes view: digidiet/edit with recipe: 1`