Routing & Instruction manual

By team: Monash Victims

Authored by: Bryan Brady & Sin Yi Tan

Table of contents:

A) Routing Table:

I) Angular route table:

1. General pages

#	Name	Pathname	Description
1	Home Page	/	Responds with the html file: /ema/ema.component.html
2	404 Page not found	/[invalid path]	Redirects and sends the user to the html file: /page-not-found/page-not-found.component.html
2	400 Invalid data	/invaliddataerror	Sets all requests of "/bryan/" to be managed by event.js controller

2. Events & G2 Pages

#	Name	Pathname	Description
1	Add event page	/addevent	Responds with the html file: /add-event/add-event.component.html
2	Delete event page	/deleteevent	Responds with the html file: /delete-event/delete-event.component.html
3	List events page	/listevents	Responds with the html file: /list-event/list-event.component.html
4	Update event page	/updateevent	Responds with the html file: /update-event/update-event.component.html
5	View event page	/viewevent/:id	Responds with the html file: /view-event/view-event.component.html Where :id is the events' id "id" (not "_id")
6	View Statistics G2 Page	/statsg2	Responds with the html /stats-g2/stats-g2.component.html
7	Language Translation Websocket Page	/languagebot	Responds with the html file: /language-bot/language-bot.component.html

3. Category & G1 Pages

#	Name	Pathname	Description
1	Add category page	/addcategory	Responds with the html file: /add-category/add-category.component.html
2	Delete category page	/deletecategory	Responds with the html file: /delete-category/delete-category.component.html
3	List category page	/liscategory	Responds with the html file: /list-category/list-category.component.html
4	Update category page	/updatecategory	Responds with the html file: /update-category/update-category.component.html
5	View category page	/viewcategory/:categoryId	Responds with the html file: /view-category/view-category.component.html Where :categoryId is the category ID
6	View Statistics Page	/stats	Responds with the html /stats/stats.component.html
7	Text to Speech Websocket Page	/speechbot	Responds with the html file: /speech-bot/speech-bot.component.html

II) Backend routing table

1. backend/routes/event.js

#	Name	Path name	HTTP method	Description
1	Events Add API	/bryan/api/v1/event/add	POST	Performs add via API, acts as backend as well for Angular, redirects to /listevents Sample input: * { * "name": "test", * "description": "Desc ", * "startDate": "2023-08-01T22:22Z",

				* "durationInMinutes": 100, * "capacity": 100, * "categories": "CCA-3001,CCA-7895" * }
2	Events get ALL API	/bryan/api/v1/event/view-all	GET	Retrieves all Events VIA API, returns in a json format [ex: postman] Integrated Angular to work.
3	Events Delete API	/api/v1/event/delete [Postman] /bryan/api/v1/event/delete?eventId=[eventId] [angular]	DELETE	Performs Delete via API [ex: Postman], Angular event deletion via the delete event page, uses events' "id" to delete an event, redirects to /listevents Sample input: * { * "eventId": "[event id here]" * }
4	Events Update API	/bryan/api/v1/event/update	PUT	Performs Update via API [ex: Postman], Integrated Angular to work, displays error 400 page if invalid data, displays /listevents if valid data Sample input: * { * "eventId":"EKU-1429", * "name":"new event", * "capacity":300 * }
5	Get an Event API	/bryan/get-event?eventId=[eventId]	GET	Retrieves 1 event object based on the provided ID, used to display an event [/viewevent/:id]

2. /backend/routes/category.js

1	Create category API	/api/v1/category/32880545/add	POST	Responsible for inserting a new category into the Category class Mongoose Schema. Responds with a JSON object containing the category ID of the newly added category.
2	List categories	/api/v1/category/32880545/list	GET	Responsible for listing all categories. Responds with the list of all categories

	API			and the details for their events in a JSON format.
3	Delete category API	/api/v1/category/32880545/delete?categoryId=<categoryId>	DELETE	Responsible for deleting a category by its ID from the query and all the events that are listed in the ‘eventsList’ field. Responds with a JSON object containing the number of deleted documents.
4	Update category API	/api/v1/category/32880545/update	PUT	Responsible for updating category name and description by ID. Responds with a JSON object with a message to confirm the update or if the ID is not found
5	View category API	/api/v1/category/32880545/view?categoryId=<categoryId>	GET	Responsible for getting a category by its category ID. Responds with the details of the category in a JSON format.

3. /backend/controllers/operation.js

1	Create operation & more	/update-records	GET	Refactored of previous assignment [“/”] in order to create an operation if it does not exist & updates current number of event + category
2	Retrieve 1 operation	/getOperation	GET	Used to retrieve the first operation object so that it can display in G1 & G2 stats page

II) Instructions to run:

1. Ensure you open the application in the “root” folder (emaAngular)
2. Build the application via “ng build”
3. Once complete, run server.js

III) Bugs & issues

- “Translate” feature will not work in VM since the socket is configured to localhost:8080