"How to Run" - Team 2

Garrett Davidson, Noah Maxey, Nate Ohlson, Riley Shaw

To run the server execute in the terminal ./HilingualServer

Execute the server on a MacOSX 10 or Linux machine, and make sure to install MySQL before using.

This will start the server and listen for requests locally on the port 8180.

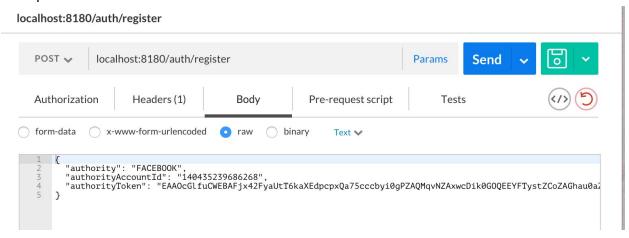
We recommend using Postman to send requests to the server. https://www.getpostman.com/ First, to create an account, you must use either Facebook or Google to get a test auth id and token. To do this with facebook visit the API documentation here -

https://developers.facebook.com/docs/graph-api/reference/v2.8/test-user To do this with google visit the API documentation here - https://developers.google.com/oauthplayground/ For google select the OAuth2 API v2 on that webpage and follow the instructions to get a test token. When getting a response from the OAuth2 test site, use the "id_token" parameter as the "authorityToken" when registering with HiLingual, the "authorityAccountId" can be any unique id for google, so keep track of whatever id you pass.

Register:

http://localhost:8180/auth/register

Request type	authority	authorityAccountId	authorityToke n	Body
POST	Either "FACEBOOK" or "GOOGLE"	ID from facebook that correlates the user. Google ID does not need to correlate to an ID, just has to be unique in the HiLingual DB.	Token given by Facebook or Google.	JSON containing authority, authorityAcco untld and authorityToke n.



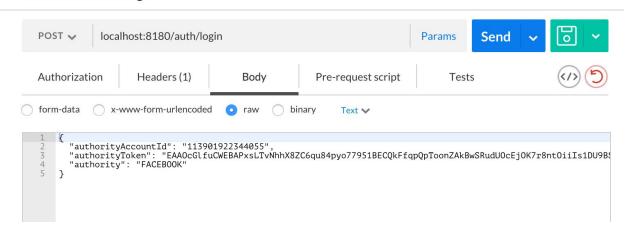
Login:

http://localhost:8180/auth/login

Request type	authority	authorityAccou ntId	authorityToken	Body
POST	Either "FACEBOOK" or "GOOGLE"	ID from facebook that correlates the user. Google ID does not need to correlate to an ID, just has to be unique in the HiLingual DB.	Token given by Facebook or Google.	JSON containing authority, authorityAccoun tld and authorityToken.

Example:

localhost:8180/auth/login



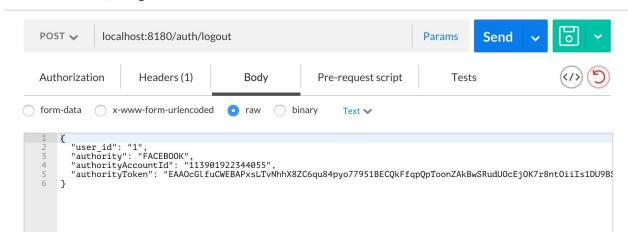
Logout:

http://localhost:8180/auth/logout

Request type	authority	authorityAc countld	authorityTok en	Body	Userld
POST	Either "FACEBOOK " or "GOOGLE"	ID from facebook that correlates the user. Google ID does not need to correlate to an ID, just has to be unique in the HiLingual DB.	Token given by Facebook or Google.	JSON containing authority, authorityAcco untId,user_id and authorityToke n.	HiLingual User id

Example:

localhost:8180/auth/logout



Messages:

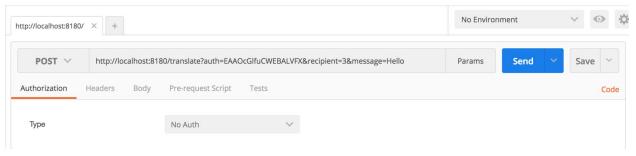
Response: Should not say invalid message.

Sending a Message:

http://localhost:8180/chat?auth={Auth-Token}&recipient={User-id}&message={Message}

Request type	Auth-Token	User-id	Message
POST	An already logged in users auth token.	The user you want to send too.(they are incrementally increased based on when you added them so 1,2,3) Is a number.	The message that you want to send.

Example:



Receiving Messages:

http://localhost:8180/chat?auth={Auth-Token}&recipient={User-id}

Request type	Auth-Token	User-id
GET	An already logged in users auth token.	The user you want to receive from.(they are incrementally increased based on when you added them so 1,2,3) Is a number.

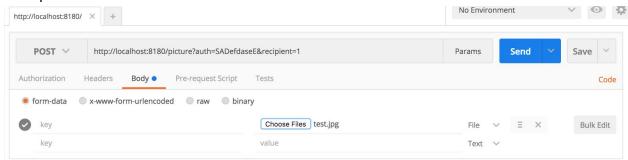


Sending a Picture Message:

http://localhost:8180/picture?auth={Auth-Token}&recipient={User-id}

Request type	Auth-Token	User-id	Picture
POST	An already logged in users auth token.	The user you want to send too.(they are incrementally increased based on when you added them so 1,2,3) Is a number.	Should be a valid Image file.

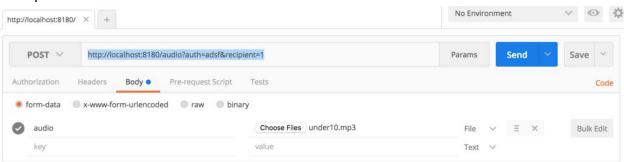
Example:



Sending an audio message:

http://localhost:8180/audio?auth={Auth-Token}&recipient={User-id}

Request type	Auth-Token	User-id	Audio
POST	An already logged in users auth token.	The user you want to send too.(they are incrementally increased based on when you added them so 1,2,3) Is a number.	Should be a valid m4a audio data.



Flashcards:

Sending flashcards:

http://localhost:8180/flashcard?auth={Auth-Token}&setid{Set-id}

Request type	Auth-Token	Set-id	Flashcards
POST	An already logged in users auth token.	What you want to set of flashcards to be called. Has to be Unique.	A valid JSon array with front and back fields.

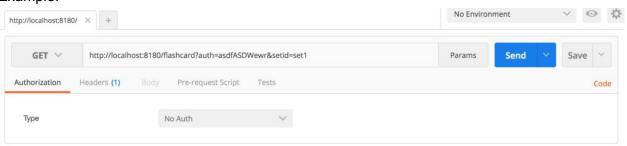
Example:



Get flashcards:

http://localhost:8180/flashcard?auth={Auth-Token}&setid{Set-id}

Request type	Auth-Token	Set-id
Get	An already logged in users auth token.	What you want to set of flashcards to be called.

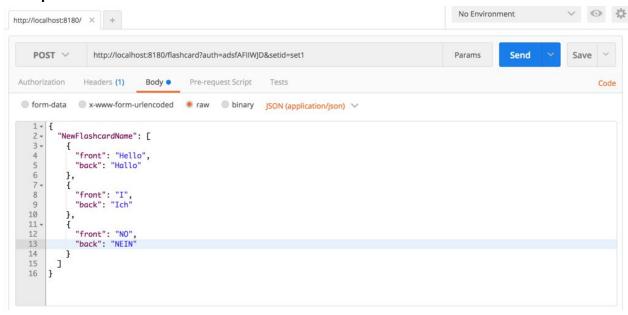


Editing flashcards:

http://localhost:8180/flashcard?auth={Auth-Token}&setid{Set-id}

Request type	Auth-Token	Set-id	Flashcards
POST	An already logged in users auth token.	The set-id should match a previously submit set.	A valid JSon array with front and back fields.

Example:

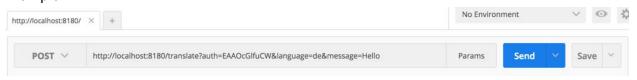


Translations:

Send a translation:

http://localhost:8180/translate?auth={Auth-Token}&language={Language}&message={message}

Request type	Auth-Token	Language	message
POST	An already logged in users auth token.	A language code for a language. http://www.science.c o.il/Language/Locale-codes.asp	The message you want to be translated



User:

Update:

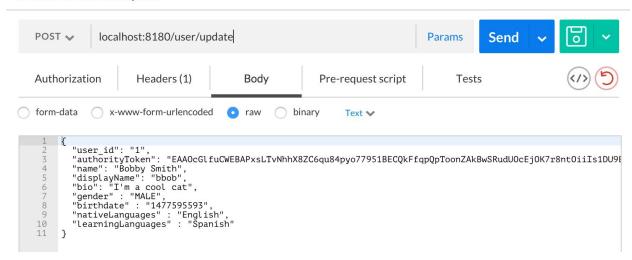
http://localhost:8180/user/update

Request type	Auth-Token	User Id	name	displayName
POST	An already logged in users auth token.	User's user id	Proper name of the user	The name that other users see

bio	gender	birthdate	Native languages	Learning languages
The bio of the person	Either all caps "MALE" or "FEMALE"	An integer of unix timestamp	Comma separated list of languages	Comma spearated list of languages

Example:

localhost:8180/user/update



Matching

http://localhost:8180/user/match

Request type	authorityToken	Body
POST	Token given by Facebook or Google.	JSON containing authorityToken.

