

Team Name: **LexiconArtists**

**Members:** Yuki Chen, Jessica Titensky, Billy Wong, Henry Zhang

**Project Manager:** Billy Wong

### Role Delegation:

Jessica: Posts handling, some front end

- Will handle Flask app. process posts, comments, edits; assist Yuki when necessary

Henry: Account management

- Will handle account settings and database component of posts, comments, edits

Yuki: Front end, Javascript

- Will design the html pages with Bootstrap, and work with Virtual Keyboard API

### Project Purpose:

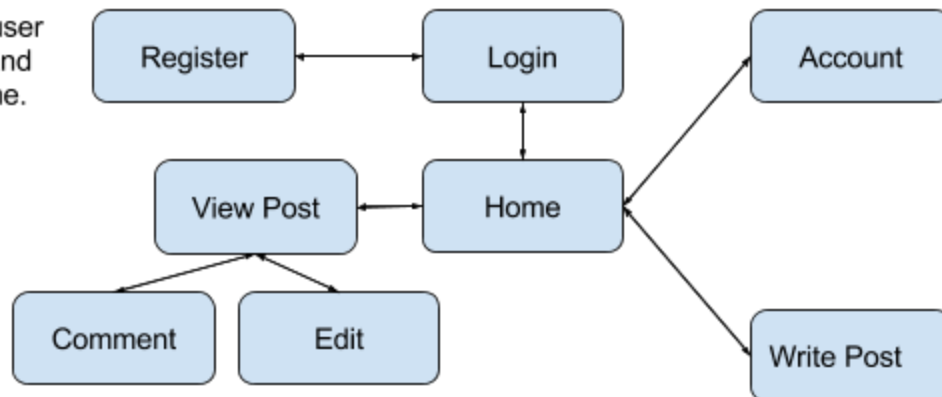
Create a forum based on peer support and communication between those looking to practice new languages.

### Site Map:

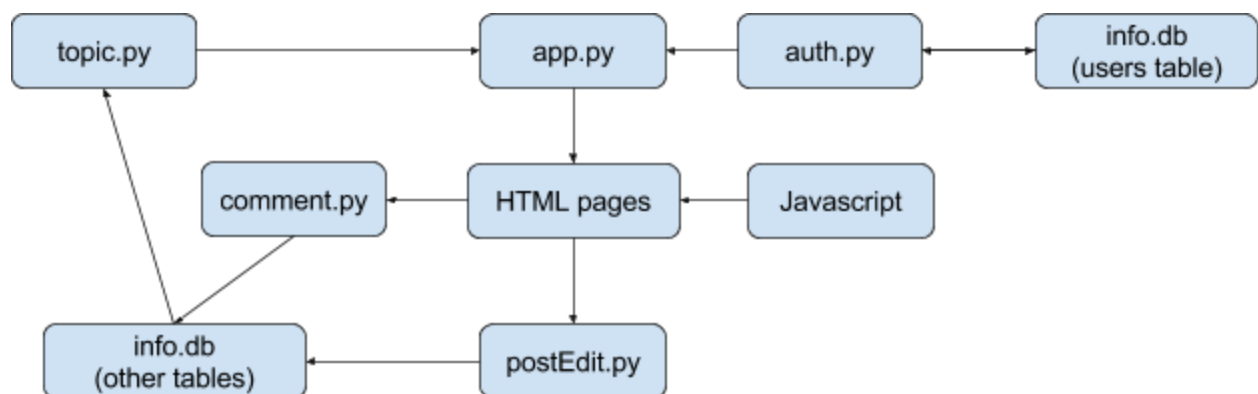
- login.html
  - Username & Password
  - Register button to go to Register.html
- register.html
  - Username & Password & Confirm Password
  - A dropdown list for user to choose his/her native language
  - Checkboxes for languages that the user is interested in.  
The preferred language(s) will determine what posts will initially be shown.
- home.html
  - Posts will be shown on right, user can comment/edit
  - On the left, user can change the language(s) of the posts shown instead of the initial preferred language
- account.html
  - User can change language settings
  - User can see topics they contributed to
  - User can view comments on topics they created
- writePost.html
  - A drop down menu for user to choose which keyboard he/she wants to write this post in. Can either use keyboard directly or the virtual keyboard
- viewPost.html
  - Versions allows user to change between original and edited versions
  - Comment: any language
    - Sent directly to user
  - Edit: limited to language of post
    - The content of the original post will be given in the edit textbox, then user can make necessary edits

The wireframes illustrate the user interface for the forum application. The first wireframe shows the Login and Register options. The second wireframe shows the Register form with fields for Username, Password, Confirm Password, Native Language, Preferred Language(s), and a Register button. The third wireframe shows the Home page with a sidebar for language selection and a main area for posts (Post A, Post B) with Comment and Edit buttons. The fourth wireframe shows the Home page with a Preferred Language(s) section and a Change button. The fifth wireframe shows the Account page with a Dropdown menu, a TextBox, a Submit button, and a Virtual Keyboard button. The sixth wireframe shows the Account page with an Original Post section, Versions, and Dropdown menus for C/E and language, along with a Submit button and a Virtual Keyboard button.

Once logged in, user can go to home and account at anytime.



Component Map:



app.py

- Flask app with routes

auth.py

- Create account/Login
- Manage settings
- Sends settings to view on the account page

topic.py

- Sends information to display on the topic pages
- Sends information to show on the home page

postEdit.py

- Adds posts to the database
- Adds edits to the database
- Adds marks to the edits

comments.py

- Add comments to the database

info.db

- Refer to database schema

Javascript

- Uses Virtual Keyboard API
- Changes keyboard based on selection

## Database Schema:

Users Table

Username - TEXT	Password (hash) - TEXT	Pref. Lang - TEXT	Native Lang - TEXT	... other data
-----------------	------------------------	-------------------	--------------------	----------------

Posts Table

Username - TEXT	PostID - INTEGR	Title - TEXT	Content - VARCHAR	Language - TEXT
-----------------	-----------------	--------------	-------------------	-----------------

Revisions

Username - TEXT	PostID - INTEGER	RevisionID - INTEGER	Text - VARCHAR
-----------------	------------------	----------------------	----------------

Comments

Username - TEXT	PostID - INTEGER	CommentID - INTEGER	Text - VARCHAR
-----------------	------------------	---------------------	----------------

## Additional Features to Work On:

- Allow users to see changes in edited versions via strikethroughs
  - Further: Interface in editing for users to directly highlight, strikethrough, bold, etc.
- Find a free translation API to provide users with computer generated translations
- Use Google Maps API to create statistics about location and languages
- Possibly provide a way to allow users to make corrections to Google Translate