Letter of Recommendation Generator

————

Setup Instructions

Jerry Tejada | Rohan Shastri | Chris Pack | Hrishi Vora | Sparsh Singh

Summary

This is an onboarding document for the Letter of Recommendation Generator application. Everything you need to know about setting up and running the app (both locally and on the USC server) is highlighted in the following sections. Contact me (jmtejada@usc.edu) if you have questions.

Deploying the app

1. In order to deploy the app, you need to be on USC's secure wireless server
2. You can use a VPN to access this server (Cisco AnyConnect Secure Mobility Client is the only one I know works for sure, it's available on Mac and Windows)
   a. Once the app is download connect to sslvpn2@usc.edu
   b. It will then ask for USC login credentials
3. Once connect via VPN, ssh into the server by typing,
   a. ssh uscadmin@128.125.100.147
   b. Pwd: USC*summer!
4. *If the app is already running, see the "TMUX" section below.*
5. Once connected to USC secure, navigate to the "app" folder
   a. cd csci-401-capstone/app/
6. To deploy, enter
   a. sudo npm run dev

TMUX

***Note: We use tmux so that you can close the terminal and still leave the app running.

7. The app will already be running in a "tmux" session. Once connected to USC secure, attach to that session by typing,
   a. tmux a -t lor2
   b. *This command is telling tmux to attach "a" into the session called lor2 "-t lor2"*
8. To terminate the app that's running, enter:
   a. Ctrl + c
9. To detach from a session, press "ctrl + b", then let go, and press "d".
   a. Ctrl + b → d
10. While developing, I prefer exiting the tmux window because it's easier to read the terminal. However, once you want to leave the app running and close the terminal, reattach to the tmux session "lor2" (or make a new one if you want), and redeploy the app on that session, then detach as explained in step 8.
11. If for whatever reason you need to restart mongoDB, there is a separate tmux window called "lor" running mongo. To attach, type:
    a. tmux a -t lor

Making Changes

1. Once you make a change to the code, commit and push that change to the repo
2. Terminate the app if it's already running, then git pull to load the new code changes, then deploy the app again using the same command from above (step 6)

Github Branches, plus Running the App Locally vs. on Server

1. The most up-to-date version of the code is on the branch called "server".
2. There may be no need to develop the local version of the app, however, we've left the setup instructions just in case.
3. "login2" is the most up-to-date local version of the app. However, since this version is pretty far behind the server version, it'd probably be best to create a new branch off of "server" and then change all of the URLs to "localhost:3000" with a "search entire folder" function in your IDE.

App Flow

1. After creating an account and logging in, you are taken to the recommender dashboard page
2. Before you can send a recommendation letter survey, you first have to create a template
   - When creating the template, you have to include at least one tag (ex: <!FNAME>) in the text editor, this tag will be populated with the first name that the student enters when they complete the form)
   - the user can also upload an input.docx document with a custom header and footer, but this doc must have a {description} tag somewhere so that the app knows where to place the text from the actual body of the letter
3. Once a template is created, you email a survey form to the student asking for the recommendation
4. The student then receives the form via email and completes the form
5. The status icon on the user's recommender dashboard will be a green checkmark once the form is complete
6. The user then presses the 'Preview Letter' button, where they will see their template text auto-populated with the student's survey information. On this page the user can then download the letter, which will produce a .docx output file.
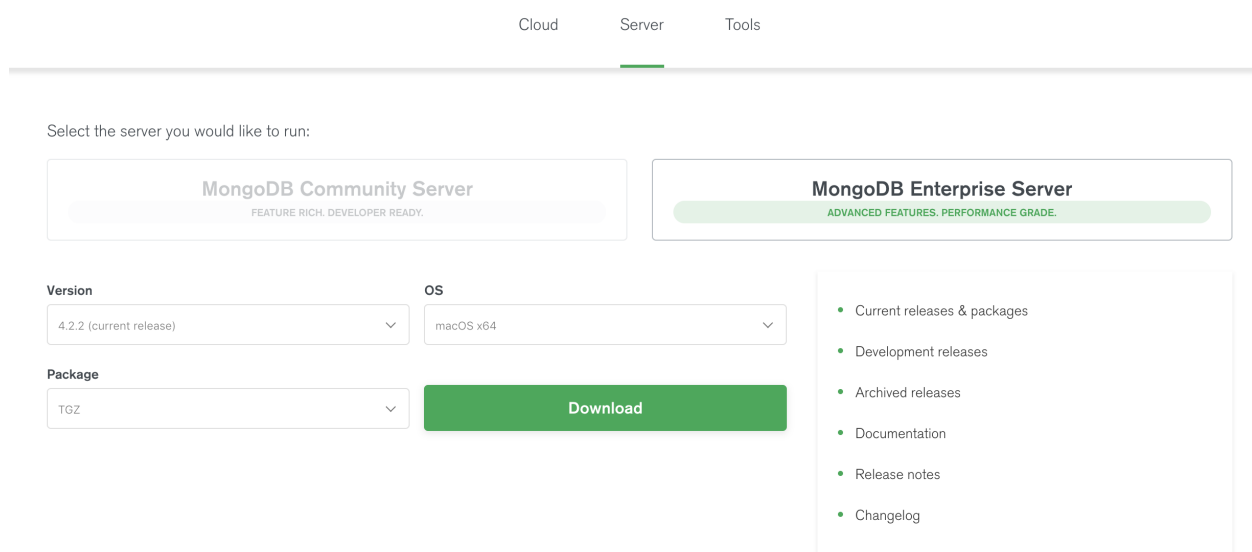
New Features and Enhancements

1. SSL Certificate

2. Archive functionality
   a. Fix the ability to restore an archived recommendation letter or an archived template.
   b. The buttons are there (in the archive tables), but they don't do anything
   c. The functions are already written in "archive.js" in the "routes" folder, but they need to be fixed.

3. Button on the "Preview Letter" page that sends an email to the student with the letter attached.
   a. Right under the "Download Letter" button make a "Send Letter" button

4. Student user accounts
   a. Right now only "Recommenders" (ex: Professors) have accounts and not "Recommendees" (ex: Students)
   b. Prof. Miller has talked about wanting a student version account where they can store their letters and view them at any time.

5. Adding USC Shibboleth for login authorization

6. See if there's a way to send letters directly to the organizations they are applying to.

7. Front-end Change Suggestions
   a. The error messages associated with login and register could be improved
   b. Buttons all around the app should be uniform
   c. Template Editor page could be improved

How to Set Up Your Machine to Run Locally

1. First, you must download and set up the MongoDB Enterprise server.

   a. Go to this link: https://www.mongodb.com/download-center/enterprise
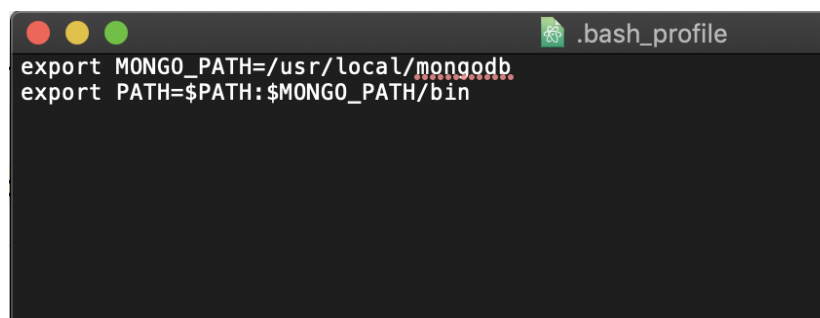
   b. Choose the latest version, appropriate OS, and TGZ package, and click "Download". *Make note of where this file is downloaded.*

   c. Once downloaded, open up your terminal

   d. Ensure you're in the home directory. Type:

      i. `cd` and click enter

      ii. Then, navigate to the folder where you downloaded the file

         ● Example: If you downloaded it to your Desktop, in terminal type: `cd Desktop`

   e. Now, unzip the TGZ folder. Type:

      i. `tar xzf mongodb-macos-x86_64-enterprise-4.2.2`

      ii. Replace "mongodb-macos-x86_64-enterprise-4.2.2" with the name of the folder you downloaded

   f. Copy the contents of the "mongodb-macos-x86_64-enterprise-4.2.2" into a new folder we're creating called "mongodb". Type:

      i. `sudo mv mongodb-macos-x86_64-enterprise-4.2.2 /usr/local/mongodb`

     ii.   Again, replace "mongodb-macos-x86_64-enterprise-4.2.2" with the name of the folder you downloaded

     iii.   You may be prompted to enter your system's password after executing this command

g. Navigate to the new "mongodb" folder. Type:

     i.   cd /usr/local/mongodb/

h. Create a new folder and enter it. Type the following commands:

     i.   sudo mkdir -p /data/db

     ii.   cd /data/db

i. Change the permission settings of the folder. Type the following commands:

     i.   whoami

         ● *This command will show you your username*

     ii.   sudo chown REPLACE_USERNAME /data/db

         ● Example: sudo chown jerrytejada /data/db

j. Go back to the home directory and display all hidden files. Type the following commands:

     i.   cd

     ii.   ls -al

k. Look through the hidden files for a file called ".bash_profile"

     i.   If you see this file, open it by typing: open .bash_profile

     ii.   If you do not see this file,

         ● create it by typing: touch .bash_profile

         ● Then open it by typing: open .bash_profile

l. A text editor will open the ".bash_profile" file. Write these lines into this file:

     i.   export MONGO_PATH=/usr/local/mongodb

     ii.   export PATH=$PATH:$MONGO_PATH/bin

> iii. **Save the file** and exit the text editor.

m. Reload the ".bash_profile" file. Type:

> i. source .bash_profile

n. **Congrats! MongoDB is now ready to go.**

o. **To test that it is working:**

> i. Close the terminal window.
>
> ii. Open two new terminal windows.
>
> iii. From the home directory,
>
> > ● Type: mongod in one window and execute the command
> >
> > ● Type: mongo into the other window and execute the command

> iv. If your terminal windows look something like this, you're good to go!

2. Clone the Github repo to your computer

a. I recommend downloading the Github Desktop application. The link can be found here: https://desktop.github.com/

b. Open Github Desktop, then go to:

> i. File → Clone Repository (a pop-up window will appear)
>
> ii. Go the URL tab → enter the repo URL (below) → click "Clone"
>
> > ● https://github.com/jeffream/csci-401-capstone.git

- *Make note of where this repo is saved*

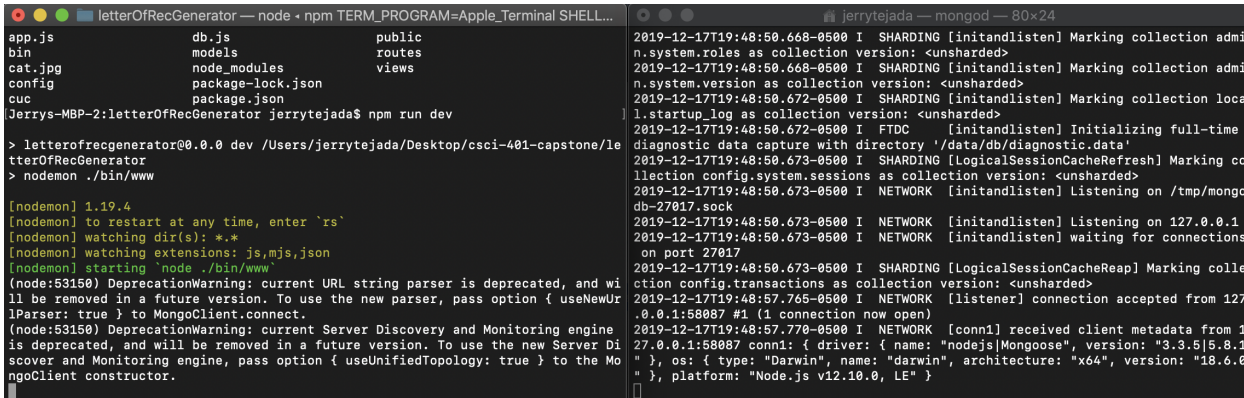3. Download the NodeJs node modules

    a. Open a terminal window and navigate to the repo you just cloned.
        i. Example: in my case, the repo is cloned onto my Desktop, so I'll type:
            - cd Desktop/csci-401-capstone/
    b. Enter the "letterOfRecGenerator" folder. Type:
        i. cd letterOfRecGenerator
    c. Install the node modules. Type the following commands:
        i. npm install
        ii. npm update

4. **Congrats!** Your system is set to run the Letter of Rec Generator app locally!

_____

How to Run Locally

1. Open Github Desktop, enter the 'login2' branch.
    a. Click the 'Fetch Origin' button on the top row of the window and pull any new code if necessary.
    b. *'login2' is our most up to date branch for local execution*
2. Open two terminal windows.
3. In the first window, start MongoDB by running the command: mongod
4. In the second window, navigate to the "letterOfRecGenerator" folder in the repo.

    a. Example (if repo saved in Desktop), type:
        i. cd Desktop/csci-401-capstone/letterOfRecGenerator

b. Run the developer start script. Type: npm run dev

c. Your two terminal windows should look something like this. If so, you're good to go!

5. Open up your browser. Type into the url field: "localhost:3000" and click enter.

6. You should now land on our welcome page and can use the app accordingly.

_____

How to Reset MongoDB

● When trying to execute the "mongod" command, you will sometimes get an error saying that mongo is already running. If this is the case simply enter the command: npx kill-port 27017

● This will terminate the currently running instance. Now you should be able to successfully run: mongod