Project Name: LangLeap

Team Number: Group 016-04

Team Name: Team Polyglots

Team Members:

Amandaliss Dropik - Pandaliss
Beckett Hyde - BeckettHydeCU
Maxwell Pettit - MaxPettit
Heather Monteson - heatherMonteson
Joseph Rizzo - jrizzo790
Matthew Recksiedler - mare8229

Project Summary :

LangLeap is an intelligent language-learning flashcard tool that can tailor sets of words to the user's personal goals and interests. It utilizes user specified input which allows for more efficient and effective learning through the avoidance of unnecessary content unrelated to the users ultimate goals. Through LangLeap users are able to build customized flash card decks by uploading specific words, text from articles, business proposals or anything related to the subject matter they need or want to learn and LangLeap will generate a list of immediately useful flashcards. Once the user creates a deck it is added to their collection of pre-existing decks which they can transition between and study with an animation that gives the experience of studying with a normal deck of flash cards. Once a user is confident with their knowledge of a flash card deck they can further test themselves in LangLeaps' quiz feature. The quiz feature is similar to flash cards but allows the user to also practice their spelling. In quizzes, the user types in the translations of words from a deck of their choosing which is then tested against the actual translation on the card. Upon completion of a quiz, LangLeap logs the user's progress and posts their stats to their profile page so that the user can see their improvements over time.

Demo Video

https://drive.google.com/file/d/1lLo46Y3gqTwBvnK2QRnH8AICSi9NTYP2/view?usp=sharing

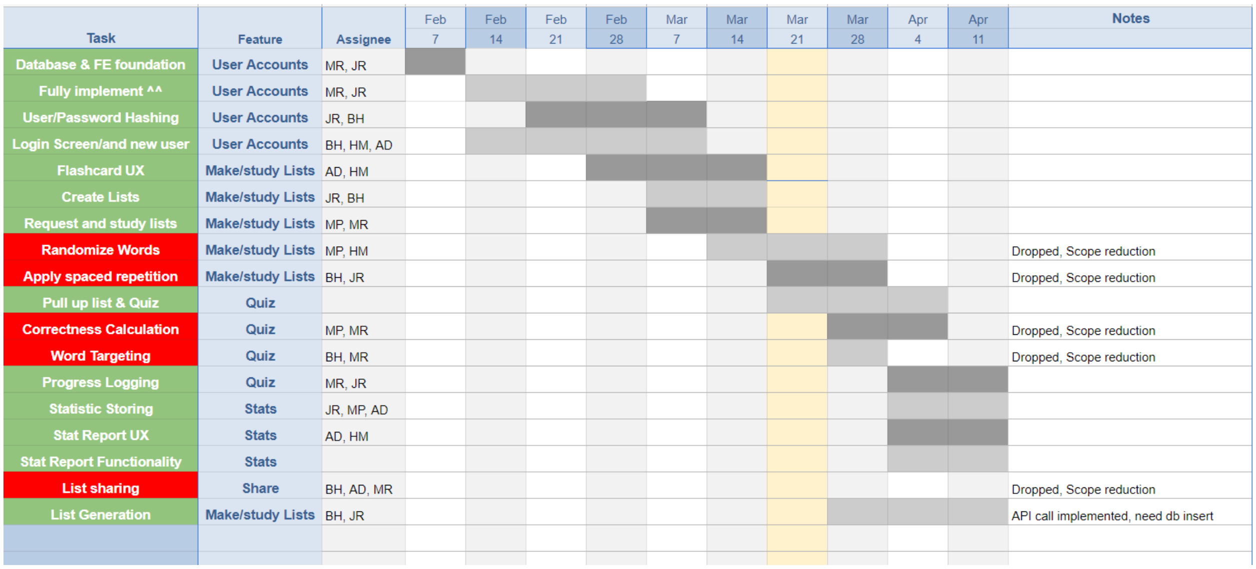
Github Repo:

https://github.com/cub-csci-3308-spring-2022/csci-3308-spring22-016-04

Deployment (heroku/local host):

- https://langleap.herokuapp.com/
- **Can also run locally using docker, running on port 3000**

Project Tracker: https://docs.google.com/spreadsheets/d/1DOoQuMnbfUAef5bVedy85C4EgUgjobP3xUHeiFSibBw/edit?usp=sharing



Task	Feature	Assignee	Feb 7	Feb 14	Feb 21	Feb 28	Mar 7	Mar 14	Mar 21	Mar 28	Apr 4	Apr 11	Notes
Database & FE foundation	User Accounts	MR, JR	■										
Fully implement ^^	User Accounts	MR, JR		▨	▨	▨							
User/Password Hashing	User Accounts	JR, BH			■	■	■						
Login Screen/and new user	User Accounts	BH, HM, AD		▨	▨	▨	▨						
Flashcard UX	Make/study Lists	AD, HM				■	■	■					
Create Lists	Make/study Lists	JR, BH					▨	▨					
Request and study lists	Make/study Lists	MP, MR					■	■					
Randomize Words	Make/study Lists	MP, HM						▨	▨	▨			Dropped, Scope reduction
Apply spaced repetition	Make/study Lists	BH, JR							■	■			Dropped, Scope reduction
Pull up list & Quiz	Quiz								▨	▨	▨		
Correctness Calculation	Quiz	MP, MR								■	■		Dropped, Scope reduction
Word Targeting	Quiz	BH, MR								▨			Dropped, Scope reduction
Progress Logging	Quiz	MR, JR									■	■	
Statistic Storing	Stats	JR, MP, AD									▨	▨	
Stat Report UX	Stats	AD, HM									■	■	
Stat Report Functionality	Stats										▨	▨	
List sharing	Share	BH, AD, MR											Dropped, Scope reduction
List Generation	Make/study Lists	BH, JR								▨	▨	▨	API call implemented, need db insert

Individual Contributions Throughout the Project:

Amandaliss Dropik:

- General
  - Worked on general design of webpage
  - Created profile page
  - Helped with milestones
- Frontend:
  - Wrote CSS and JS for profile page and some of decks page
  - Helped with general styling of webpage
  - Contributed to creation of navbar
  - Created login page
- Backend
  - Helped with querying for statistics on profile page
  - General fix bugs on server file
  - Helped with login page

Beckett Hyde:

- General
  - Developed initial idea
  - Resolved some issues with the dictionary api and list generation
  - Set up initial docker structure to interact w/ front and back end
  - Got us live on heroku
- Front End:
  - Converted front end files from initial html to .ejs
  - Created partials files that propagated navbars across all pages
  - Reformatted original css files to work across partials
- Backend:
  - Built initial getter node.js file and methods
  - Built initial docker structure
  - Built first login & sign-up system
  - Helped finish login & sign-up and implement hashing (joe did a lot here)
  - Helped tie PostgreSQL into the node.js renderer (built some queries and restructured some stuff)

Maxwell Pettit:

- General:
  - Created flashcard and quiz features/animations. Wrote tests and took meeting notes.
- Front End:
  - Wrote CSS and JS for flashcards and quiz cards to move left, right, and flip. Also the code to pull the words from a javascript object to get the previous, current, and next words.
  - Started EJS to dynamically populate decks page (Heather finished this).
- Backend:
  - Wrote the query to get decks from the database belonging to the user.
  - Worked on node js files so when a deck is clicked for study or quiz it loads that specific list of words from the database.
  - Created sql file to spawn default decks.

Heather Monteson:

- General:
  - Worked with the team on all milestones and created logo/background.
- Front End:
  - Main focus was the decks and make decks pages: layout, stying, html, css, js.
  - Utilized a get request (written by Max and Joe) to populate decks dynamically.

    - On quiz page html, css and js for quiz accent buttons.
    - Preliminary format/style for cards page and worked on nav bar.
- Backend:
    - Wrote generic word lists for the database.
    - Added logout functionality.
    - Adjusted decks page get request (Max wrote and Joe also worked on query).
    - Worked with the team on pull requests for the profile page stats.
    - Set up mocha and chai and wrote a test case to verify deck title before building.

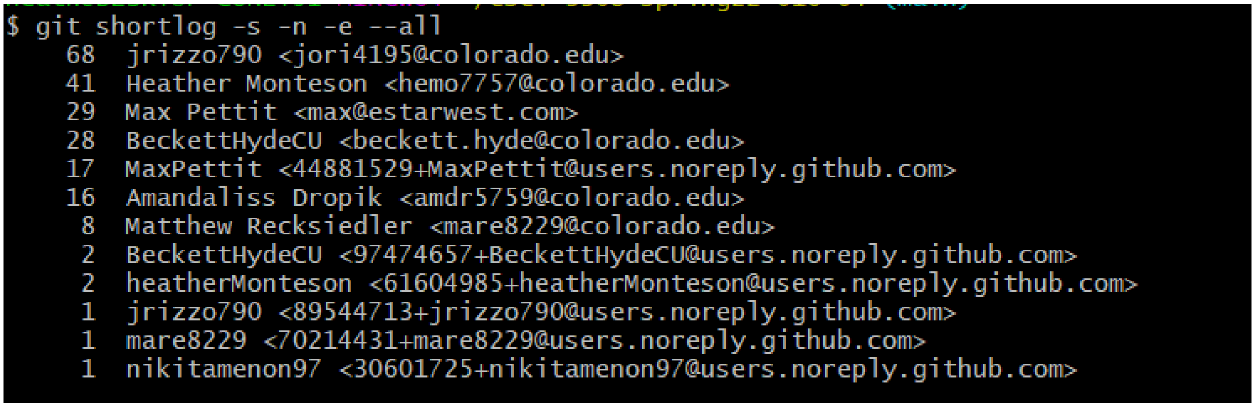
Joseph Rizzo:

- General:
    - Worked with the group to design architecture for our app.
- Frontend:
    - Added show/hide password functionality on Login and Signup pages.
    - Helped fix bugs regarding deck display on Decks page.
    - Helped display user data and stats on the Profile page.
- Backend:
    - Wrote rough database layout and SQL script to create database upon deployment.
    - Increased security with hashing and salting algorithm using bcrypt with NodeJS. Added hash comparison function that checks passwords upon login (only hashes and salts are stored in the database).
    - Implemented process for generating flashcard words from text. Helped with the Axios call / Merriam-Webster API, parsing that data to generate their translations.
    - Added statistics within the quiz mode, tracking correct, incorrect answers, and total attempts.

Matthew Recksiedler:

- General:
    - Helped to visualize how exactly LangLeap would initially work.
- Frontend:
    - Helped with general styling of some web pages
    - Set up the JSON data so that it can be easily displayed in the frontend
- Backend:
    - Helped develop the database layout
    - Set up the API to take user input and output translations and other useful data to the database.

Git (as of 4/18):



```
$ git shortlog -s -n -e --all
    68	jrizzo790 <jori4195@colorado.edu>
    41	Heather Monteson <hemo7757@colorado.edu>
    29	Max Pettit <max@estarwest.com>
    28	BeckettHydeCU <beckett.hyde@colorado.edu>
    17	MaxPettit <44881529+MaxPettit@users.noreply.github.com>
    16	Amandaliss Dropik <amdr5759@colorado.edu>
     8	Matthew Recksiedler <mare8229@colorado.edu>
     2	BeckettHydeCU <97474657+BeckettHydeCU@users.noreply.github.com>
     2	heatherMonteson <61604985+heatherMonteson@users.noreply.github.com>
     1	jrizzo790 <89544713+jrizzo790@users.noreply.github.com>
     1	mare8229 <70214431+mare8229@users.noreply.github.com>
     1	nikitamenon97 <30601725+nikitamenon97@users.noreply.github.com>
```