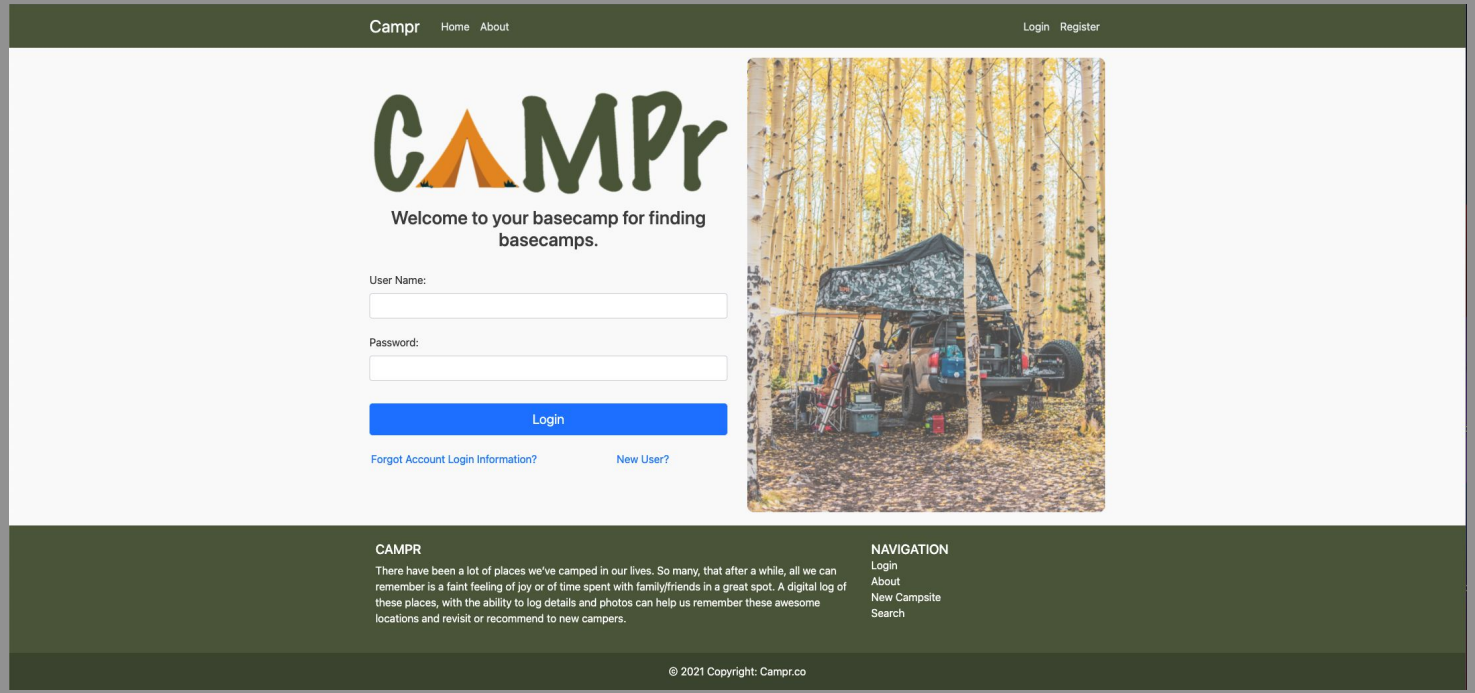
CAMPr

Joseph Kelliher
Kelsey Kosnik
Matthew Martin
Jacob Unger

Tool Used to Create CAMPr

Tool Name	Purpose	Evaluation
Discord	Communication	Excellent
Trello	Project Tracker/Task management	Good
GitHub	Version Control/Code Storage	Excellent
Database	SQLite3	Good (enough for our usage)
Python3	Front End Architecture	Excellent
Python3/HTML/CSS/SQLite3	Testing/page layout/page format/database	Excellent
IntelliJ/VSCode	Code Editor	Good
Sublime	Code Editor	Good

Project


Campr
Home
About
Login
Register
CAMPr
Welcome to your basecamp for finding basecamps.
User Name:
Password:
Login
Forgot Account Login Information?
New User?
CAMPR
There have been a lot of places we've camped in our lives. So many, that after a while, all we can remember is a faint feeling of joy or of time spent with family/friends in a great spot. A digital log of these places, with the ability to log details and photos can help us remember these awesome locations and revisit or recommend to new campers.
NAVIGATION
Login
About
New Campsite
Search
© 2021 Copyright: Campr.co

Front End Architecture Overview

- Flask!
  - This is what we used to lay forth the routing and paths of the webpage, made that portion very easy, integrated nicely with other tools such as SQLite3 (database) and WTforms (form validation).
- HTML/CSS/Bootstrap
  - Pretty boilerplate stuff! Allowed us to all get the same colors and containers and sizing even though the pages were developed by different group members. Bootstrap gave us a lot of really nice built in features such as the buttons that we used, saving us some time

Database Overview

- SQLite3
  - The primary form of database that we used!
  - Gave us the flexibility of having a local database that we grabbed with each pull if it was updated, immediate feedback when testing the code, and persistent saved data, even we we closed the project and worked on it elsewhere
- SQLALchemy
  - An SQL tool kit that we used as an interface for our database, made integration easier
  - This is how we accessed the database

Testing Overview

- We used python for testing, namely the unittest library. We focused on testing the campsite database and the user credential database, since that is the data that we aimed to store
- Using a tester database with verified entries, the tests verified that expected matched actual with checking campsites are matched to correct states, campsites are correctly added, and expected city state campsite pairs
- The login credential tests followed verification on input, making sure that a username is provided and detected, that its a string, password is correct length, email is provided and in correct format. This is a secondary level of verification to the form verification that's done on time of account set-up with FormWTF Python Package

CAMPr Demonstration here

Challenges

- Team coordination (initially) – took a couple of weeks to find a common time and communication method that worked for the team
  - Resolved by committing to a standard time each week that team members could plan to and establishing communication methods for more real time communication
- Fuzzy understanding of goal (initially) – no one had a great understanding of what “done” was supposed to look like initially
  - Developed a high level product requirements document and UI sketches to ensure a common understanding and goal
- Estimation – started out with a definition of done that would have been very difficult to meet during the semester
  - Defined MVP items and challenged/addressed scope creep in weekly standups
- Linking – had some hiccups along the way getting all the pieces to play together
  - Resolved through teamwork, spikes to investigate, and working sessions

Citations

25.3. Unittest — Unit Testing Framework — Python 2.7.18 Documentation. Accessed November 29, 2021. https://docs.python.org/2/library/unittest.html#module-unittest.

Bayer, Michael. *SQLAlchemy - The Database Toolkit for Python*. Accessed November 29, 2021. https://www.sqlalchemy.org/.

contributors, Mark Otto, Jacob Thornton, and Bootstrap. *Bootstrap*. Accessed November 29, 2021. https://getbootstrap.com/.

“Flask-WTF — Flask-WTF Documentation (1.0.x).” Accessed November 29, 2021. https://flask-wtf.readthedocs.io/en/1.0.x/.

Python 3.10.0 Documentation.” Accessed November 29, 2021. https://docs.python.org/3/reference/.

Sqlite3 — DB-API 2.0 Interface for SQLite Databases — Python 3.10.0 Documentation. Accessed November 29, 2021. https://docs.python.org/3/library/sqlite3.html#module-sqlite3.

Welcome to Flask — Flask Documentation (2.0.x). Accessed November 29, 2021. https://flask.palletsprojects.com/en/2.0.x/.

Werkzeug — Werkzeug Documentation (2.0.x). Accessed November 29, 2021. https://werkzeug.palletsprojects.com/en/2.0.x/.

WTForms — WTForms Documentation (3.0.x). Accessed November 29, 2021. https://wtforms.readthedocs.io/en/3.0.x/.“The Python Language Reference —