

Team Redacted: Rachel Xiao, Michael Borczuk, Yuqing Wu, David Chong
SoftDev
Project 00 -- Move Slowly and Fix Things
2021-10-28

Program Components:

- HTML
 - Templates for Login Page, Create Account Page, Home Page, Choose Story to Add, Add to Story Page, Create New Story, View Stories Page
 - Buttons/links to move from page to page
- Flask
 - Takes user inputs from the templates and interacts with the database
 - Gets username and password and verifies it against the database
 - Stores session so that the user can stay logged in
 - Renders HTML templates and fills them in using data from the SQL database
- SQL
 - Creates a db file that stores data from the website
 - Has datatables for usernames, passwords, and the stories the user has contributed to
 - Stores story title, story content, and users contributed to the story in large strings separated by special characters (\n). First user could be the creator of the story **
 - Accesses the database and interacts with Flask

Database Organization:

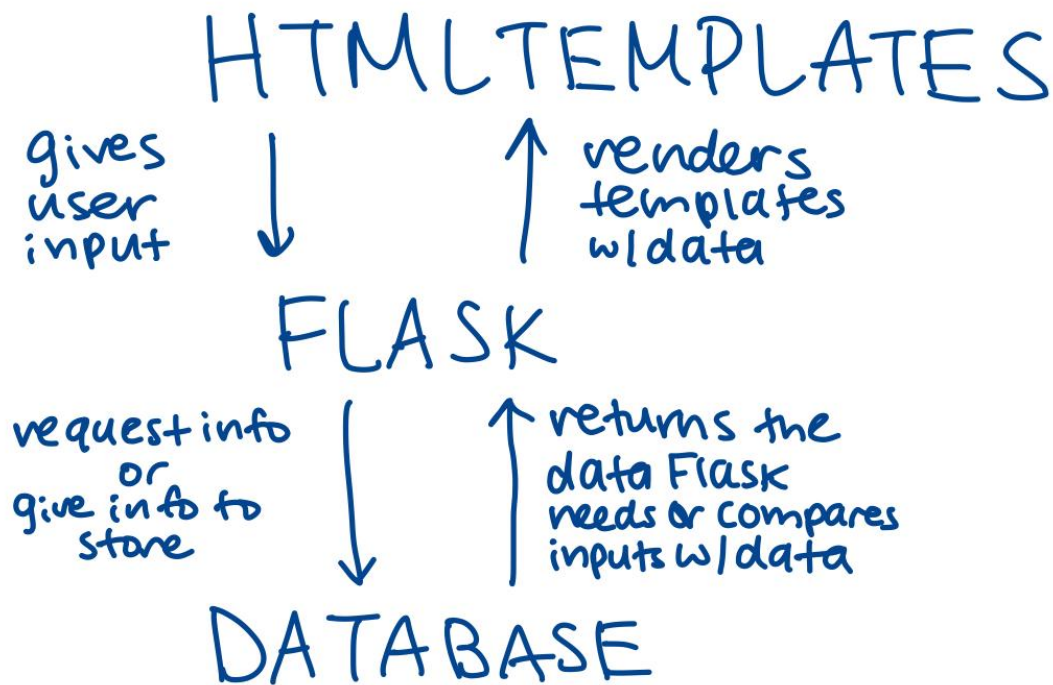
user_info:

username (TEXT) - UNIQUE	password (TEXT)	stories_contributed (TEXT)
--------------------------	-----------------	----------------------------

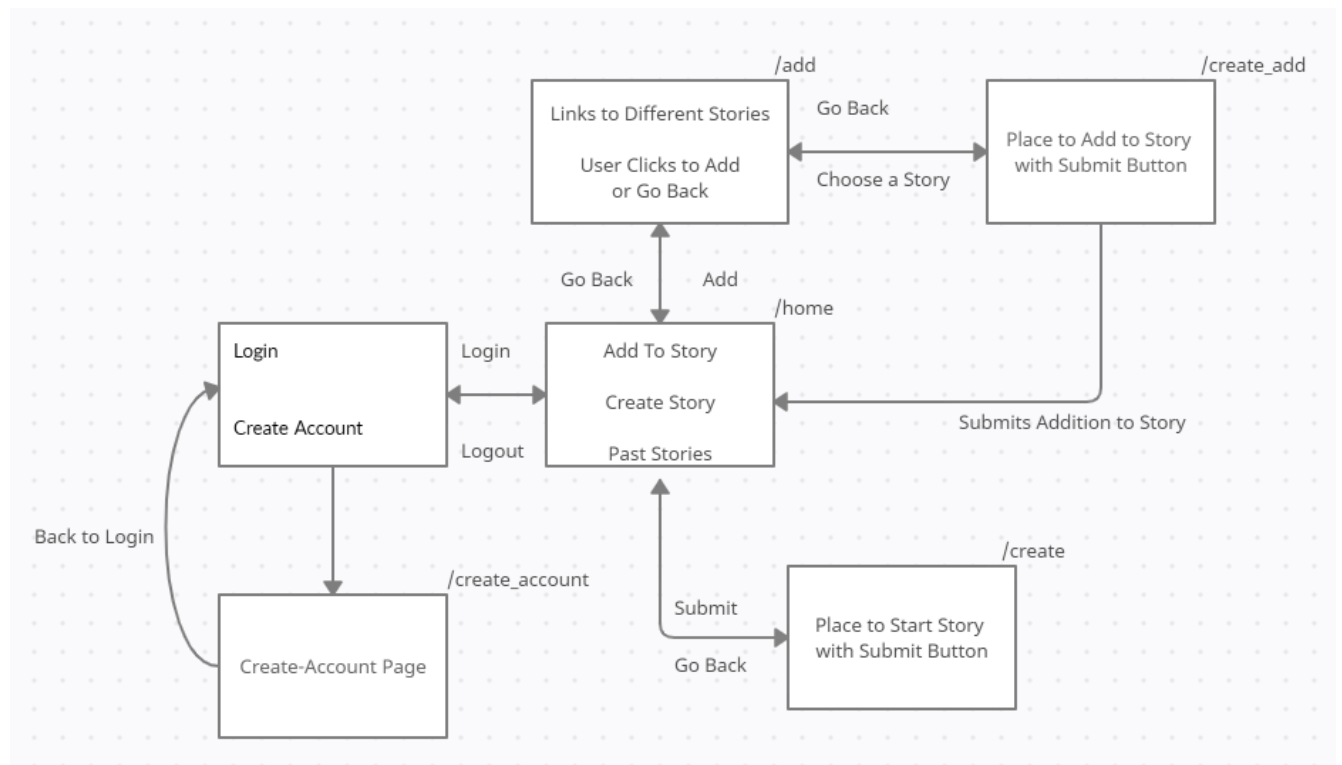
story:

title (TEXT) - UNIQUE	contributor (TEXT)	entry (TEXT)
-----------------------	--------------------	--------------

Component Map:



Site map:



Breakdown of each page:

- Login Page @app.route("/")
 - Template for Login Page
 - Database to store user information and compare with login inputs
 - Render template with Flask and store cookies
 - Redirect to ("/create_account") if user clicks the create account button
 - Redirect to ("/home") when the user successfully logs in
- Create an account ("/create_account")
 - Template form asks for username, password, and confirm password
 - If both password entries are the same, then store the password in the database using Flask and SQL
 - If the username is unique, then store it in the database
 - Error messages if username already exists or if the passwords don't match
 - After creating an account, the user is redirected to the "/login" page
- Home Page ("/home") → landing page after login
 - Contains links to add to or create a story
 - Redirect to:
 - Create Story Page ("/create")
 - Uses an HTML template to take inputs for the "title" and "content"
 - Check if "title" is unique
 - Check if "title" or "content" is empty
 - Add to Story Page ("/add")
 - When user clicks on certain stories would redirect to "/add/<story>"
 - Display the latest entry in the database
 - Form to input a limited amount of text in the story
 - Store the story title into the datatable for stories the user has contributed to
 - View full stories that the user has contributed to
 - Preview Featured Story (advanced feature) **

Tasks:

- HTML: Create simple and elegant templates for each page. Allows templates to display data with Flask. Login page, home page, create page, add page, and create account page → David
- Flask: Bring information from the database to HTML and vice versa. Maintain session login and redirect users accordingly. Check user inputs according to constraints → Rachel and Michael
- SQL: Store login information when creating a new account. Store stories information and create functions to access the correct information from database (like most recent addition to a story) → Michael and Yuqing