Sudoku W/ Ease
Lab 2 Report

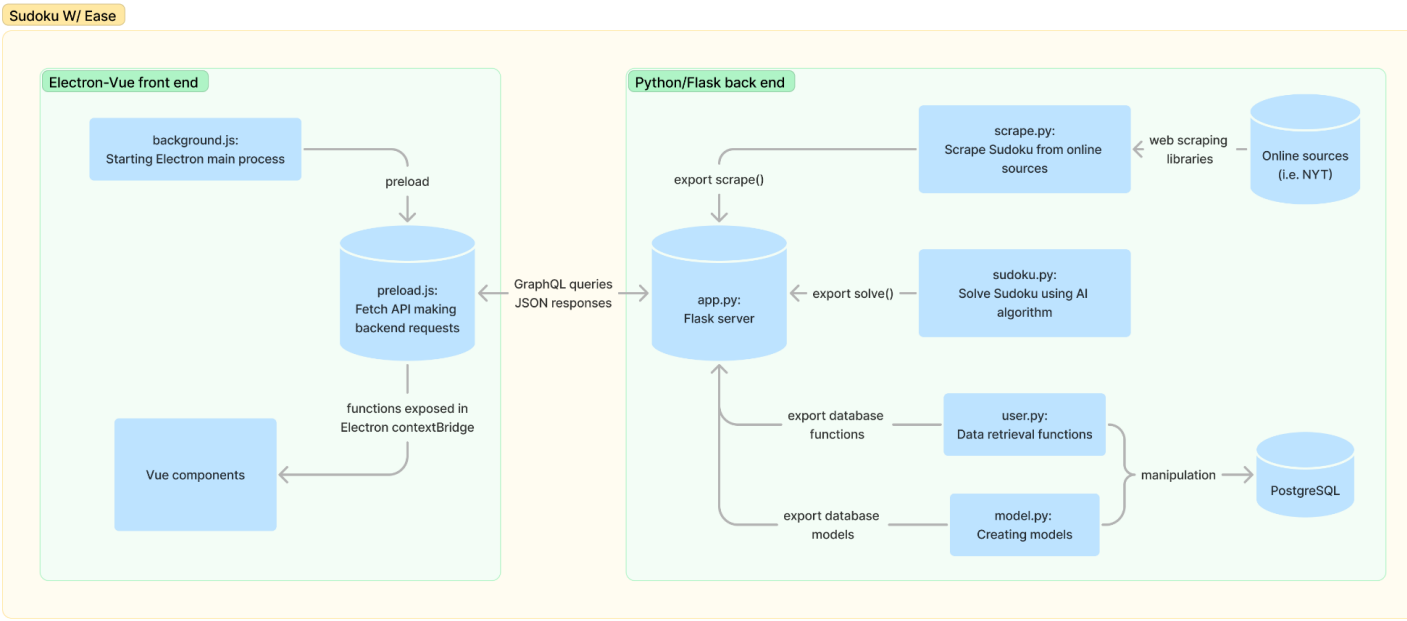
Yichen Guo yg2258@nyu.edu

06/25/23

Experiment 1: We will be upgrading the original Flask backend server to Flask server with GraphQL endpoints and Web Socket subscription system. Also, Flask server will be connected to PostgreSQL for data storage and retrieval. Please draw the new system design graph here.

Answer:

System diagram



Sudoku W/ Ease
Lab 2 Report

Yichen Guo yg2258@nyu.edu

06/25/23

GraphQL Requests

#	GraphQL Request	Backend Functions
1	SolveSudoku	solve(sudoku, size)
2	ScrapeSudoku	scrape(index, difficulty = “”)
3	RegisterUser	register(email, password, name)
4	LoginUser	login(email, password)
5	RemoveUser	removeUser(userId)
6	AddFavorite	fav(sudoku, userId)
7	ShowFavorite	getFav(userId)
8	RemoveFavorite	removeFav(sudokuId, userId)

Experiment 2: We will be implementing a PostgreSQL database and an user system. Please research and define the database structure by drawing the database schema and define the primary key for each database.

Answer:

Sudoku W/ Ease
Lab 2 Report

Yichen Guo yg2258@nyu.edu

06/25/23

PostgreSQL User Table

Type	Name	Restrictions
INT	userId (PRIMARY KEY)	
TEXT	email	Allowed characters: • latin letters (a-z). • numbers (0-9). • special characters: underscores (_), periods (.), and dashes (-). A period (.) is not permitted at the start or end. Consecutive periods or special characters (e.g., john...doe@company.com) are not allowed. Special characters should be followed by one or more letters or numbers. Must include an at sign (@).
TEXT	password	Minimum password length >= 6 Maximum password length <= 24 Require at least one lowercase letter Require at least one uppercase letter Require at least one number Require at least one of the following characters: !"#$%&'()*+,-./:;⇔?@[\]^_`{|}~
TEXT	name	Allowed characters: • latin letters (a-z). • numbers (0-9). • special characters: underscores (_), periods (.), and dashes (-). Usernames cannot contain an ampersand (&), equals sign (=), apostrophe ('), dash (-), plus sign (+), comma (,), brackets (<,>), or more than one period (.) in a row. Maximum username length <= 24
INT[]	fav	

Sudoku W/ Ease
Lab 2 Report

Yichen Guo yg2258@nyu.edu

06/25/23

PostgreSQL Sudoku Table

Type	Name	Restrictions
INT	sudokuId (PRIMARY KEY)	
INT[]	sudoku	Array length is one of 16, 81, 256, and 625.
INT	userId	

Experiment 3: We will be using Web Sockets to implement realtime updates from application's back end. Please respond to the following questions.

- What is Web Sockets? What is it used for?
  Answer:
  WebSockets is a communication protocol that provides full-duplex communication channels over a single TCP connection between a client and a server. Unlike traditional HTTP requests, which follow a request-response pattern, WebSockets enable **real-time, bidirectional communication** between a web browser or client application and a server.

1. **Real-Time Communication**: WebSockets allow for real-time communication between a client and a server, enabling instantaneous data transfer. It provides a persistent connection that allows both the server and the client to send messages to each other at any time without the need for frequent polling or refreshing the page.
2. **Efficient and Low Latency**: WebSockets have a lightweight protocol overhead, reducing the data transfer size compared to traditional HTTP requests. This efficiency results in lower latency and better performance, making it suitable for applications that require real-time updates, such as chat applications, collaborative tools, live streaming, and stock market tickers.
3. **Bi-directional Communication**: WebSockets support bi-directional communication, allowing both the client and the server to send messages to each other independently. This enables the server to push data or updates to the client in real-time without the need for the client to repeatedly request information.
4. **Event-Driven Model**: WebSockets follow an event-driven model, where both the client and the server can trigger events and respond to events asynchronously. This event-driven nature allows developers to build interactive and responsive web applications.

- Please compare the differences between WebSockets and SocketIO.
  Answer:

1. Socket.IO is better for implementing production-ready realtime features more quickly, though it's not as flexible.

Yichen Guo yg2258@nyu.edu

Socket.IO is a library built on WebSockets. It adds features including a heartbeat, auto-reconnect, multiplexing, rooms, and the ability to fallback to long polling in case the browser can't establish a WebSocket connection.

2. If none of these features are necessary for your project, you can build on WebSockets directly. However, you should carefully weigh this decision against the time and effort required to essentially recreate the same core functionality offered by Socket.IO across platforms.
3. Another consideration is server-side scalability. If building a Socket.IO and Node server side, some additional tools are available to help developers to scale horizontally.

Experiment 4: If we are creating a secured Python user system, what types of technology are necessary? Illustrate the reason, the technology you choose, and which part of the system you want to integrate them into.

Answer:

- PostgreSQL
- SQLAlchemy: SQLAlchemy is an Object-Relational Mapping (ORM) library that provides a high-level interface for interacting with databases. It supports PostgreSQL and provides features like connection pooling, query generation, and data mapping.
- Psycopg2-binary: psycopg2 is a PostgreSQL adapter for Python that allows you to connect to a PostgreSQL database from your Flask application. It provides a secure and efficient way to interact with the database.
- Hashing Algorithms (e.g., bcrypt): Hashing algorithms are crucial for securely storing and comparing user passwords. When a user registers or updates their password, it should be hashed before storage. When authenticating a user, their input password is hashed and compared to the stored hash.
- (Data migration tools)