Astor

Motivation

What is the problem?

Logistics companies have to send and receive a large number of packages every day, how to efficiently deliver, how to reasonably dispatch personnel, how to solve some professional problems are often encountered.They hope to have a system that can provide feasible algorithms for their daily needs accurately and quickly, and support reasonable answers based on uploaded data on the online platform.

What is your vision for solving the problem?

- Algorithm store = Algorithm + store
- For store(Template: Taobao):
  - Front and back end and database
- For Algorithm:
  - Online platform
    - run on docker
      - Like ECS, more expenditure, more space and hashrate.
      - Security and independent environment.
      - More efficient use of system resources
      - Faster startup time than virtual machine
      - Consistent operating environment
      - Continuous payment and deployment
      - Easier migration, maintenance and expansion
  - Database
    - MySQL

What are your “silver bullets”?

- Book: <No Silver Bullet—Essence and Accidents of Software Engineering>
- This paper emphasizes that the real silver bullet does not exist because of the complexity nature of software. No silver bullet means that no single technology or method can increase software engineering productivity tenfold in a decade.
- But, we have some powerful method to make our project stronger:
  - node.js
    - Advantage
      - High concurrency

        - Suitable for I/O intensive applications
    - vue.js
        - lightweight, high-performance and componentized MVVM library with very easy-to-use APIs
        - Support for two-way data binding
        - provides instructions, filters in the template
    - Django
        - Powerful database function
        - With powerful background functions
        - Elegant website
        - With template system
        - Caching system
- So we did it by separating the front end from the back end.
    - Front end: responsible for the View and Controller layers.
    - Backend: only responsible for Model layer, business processing/data, etc
    - Advantages:
        - Developers are separated, and there are specialists
        - True front and rear decoupling can be achieved
        - Easier to locate bugs
        - Reduce concurrency/load pressure on backend servers
        - Even if the backend service is temporarily out of time or down, the front page will still be accessed, but the data will not come out
        - Asynchronous loading
        - Increased code maintainability & readability
        - Nginx supports hot deployment of pages without restarting the server
        - Improve development efficiency, because you can develop both front and back in parallel, instead of relying heavily on it

Feature Description

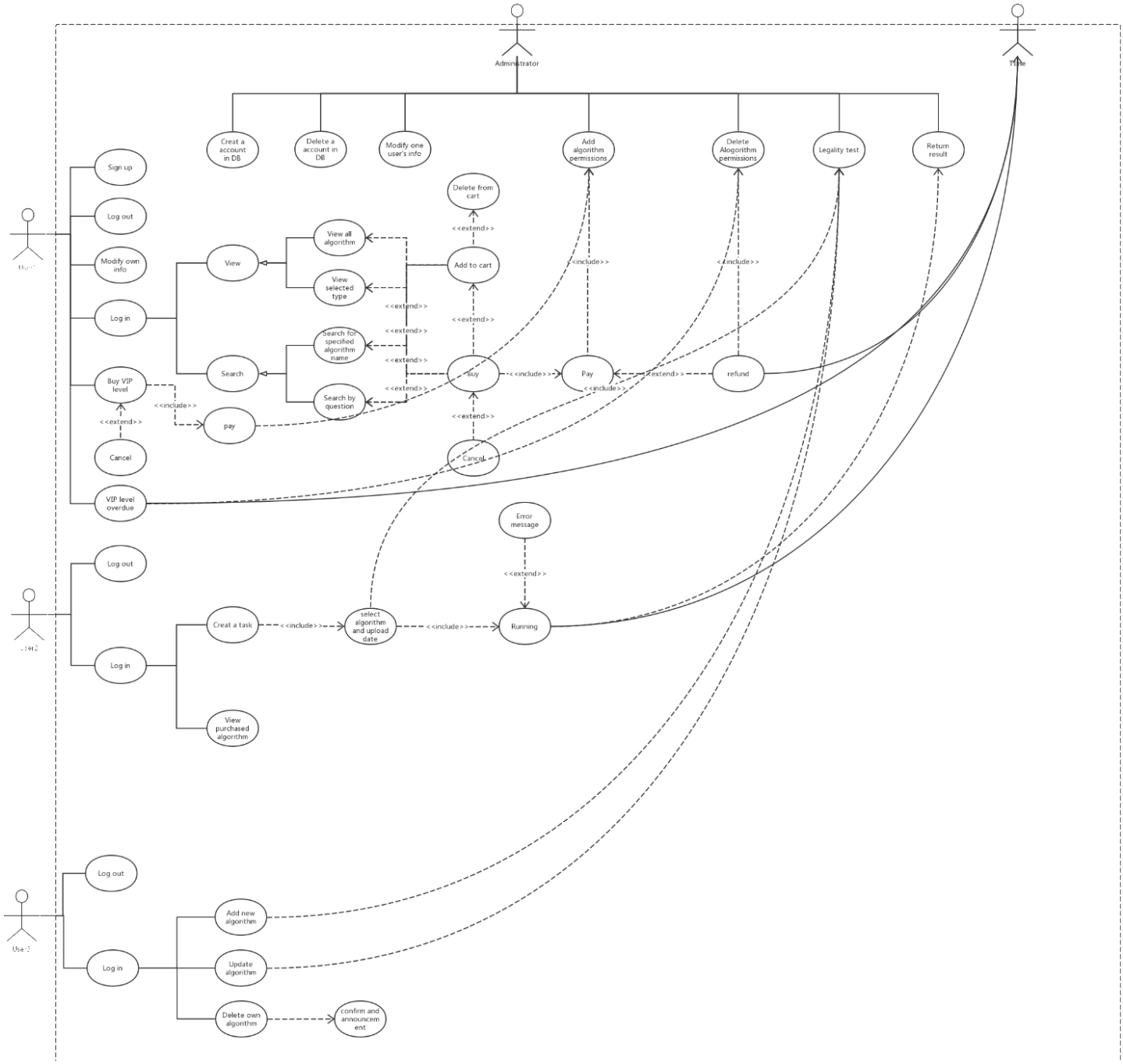
Users storys

- After a user logs in, he can view all the online algorithms in a list, or he can select one type of algorithm to view.There are two ways for users to search : (1)search by algorithm name (2) search by the problem, such as the shortest path problem. The user can add the algorithm to the shopping cart and delete items from the cart. The user makes a purchase by clicking the buy button. If he changes his mind after cliking the buy button, he can click cancel button. The user needs to pay to confirm the purchase. The user can return the algorithm and get refund within 7 days after payment.
- The user chooses one of the algorithms he has purchased to solve his problem. He will create a task, selecte an algorithm and upload the data. The administrator will test the legality of the uploaded data. If the data is invalid, the user will receive an error message. If it is legal, the administrator will allow the user to run the algorithm. In the running process, there may be errors

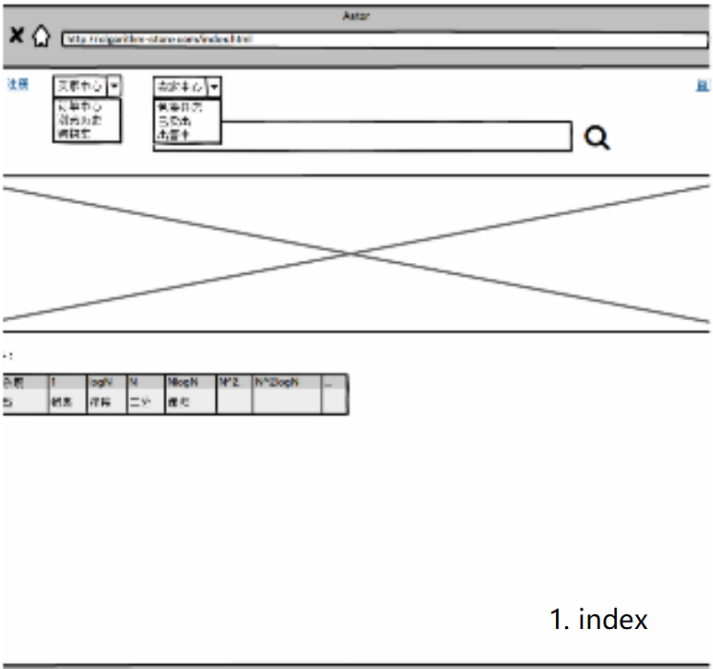
causing interruption, then the user will receive an error message. Else, after the process is done, the user will recieve the result successfully.

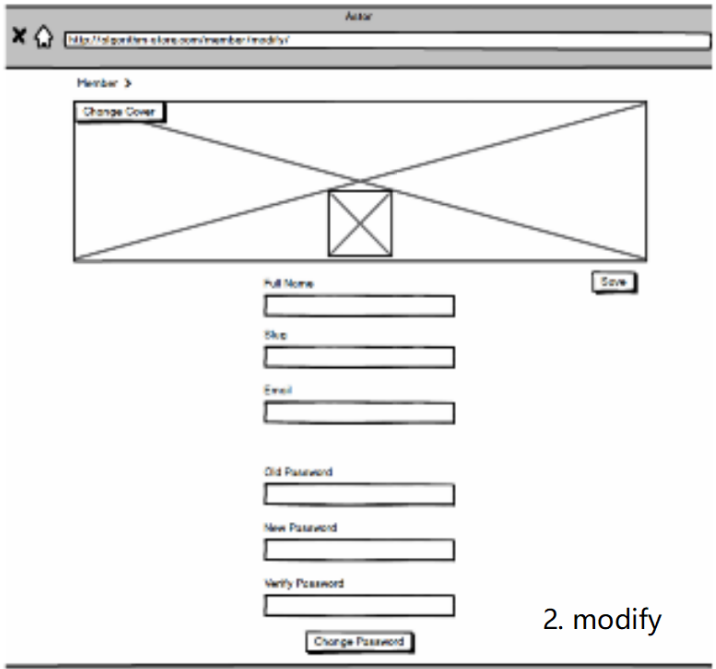
- The user can add his own algorithms to the store. When someone else buys his algorithm, he will reap the benefits. The administrator needs to test the validity of the algorithm added by the user. Only the algorithm that passes the test will be added. Users can update their uploaded algorithms, which must also be tested by admins. The user can also delete the algorithm he has uploaded but he needs to confirm deletion and make an announcement.

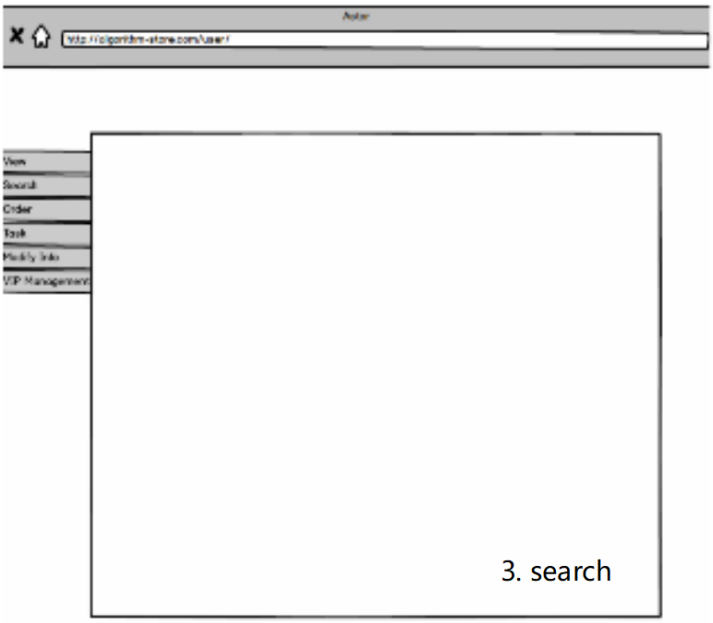
UML user casees



Mockcups







Requirements

Functional requirements

For user:

- Register
- Login and logout
- Check and modify own information
- Search for algorithm by algorithm name or given question
- View the algorithm, add algorithm into cart and buy algorithm
- Create new task, upload the data to run the algorithm and get the result message
- All the data store in the database

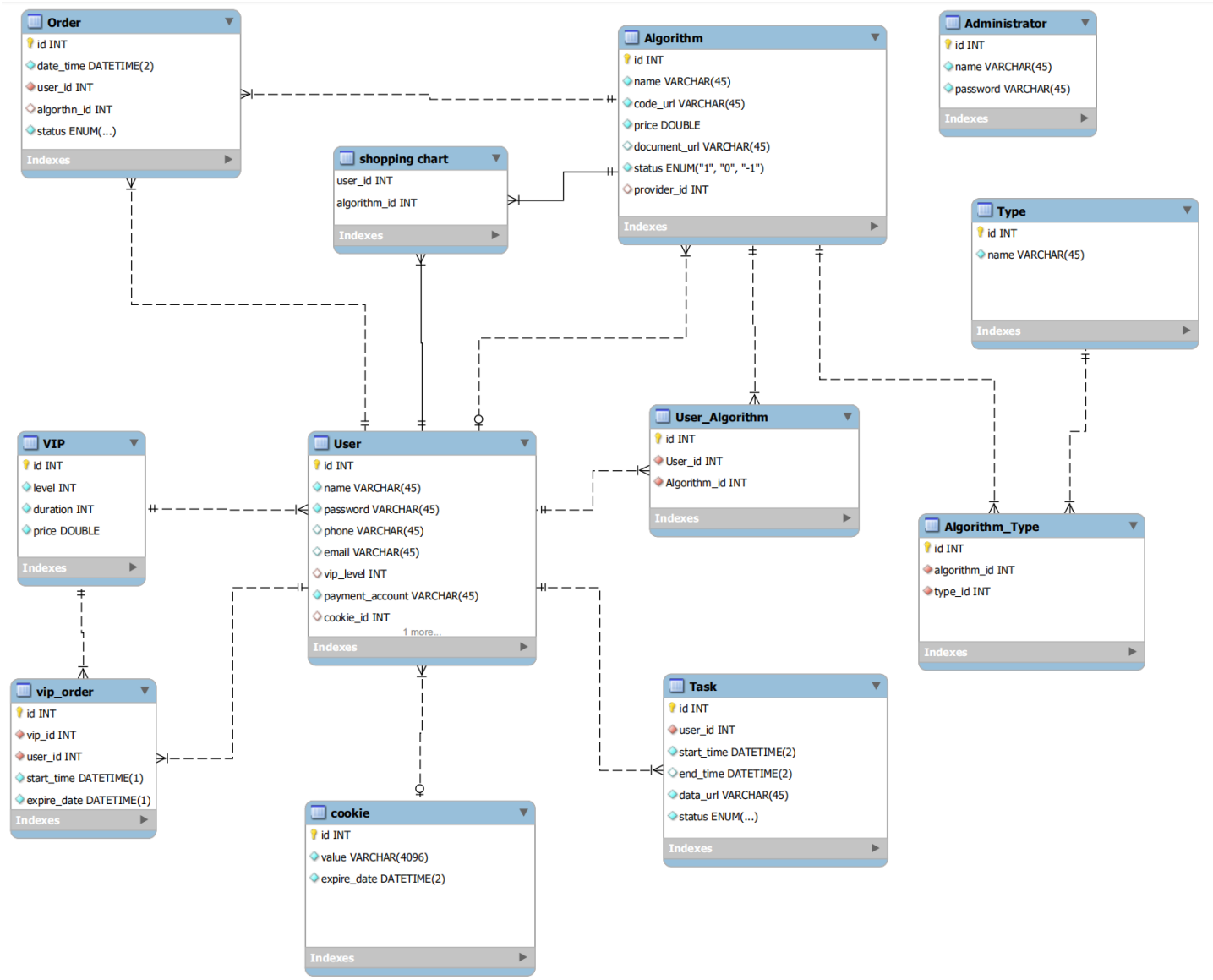
For Administator
(1)Create/Delete an account(2)Modify user info(3)Add/Delete algorithm permission(4)Legality test(5)Return result

non-Functional requirements

- Faster response time
- Stable front and rear communication
- More accurate and more comprehensive search
- Use docker for online platform
- Upgrade the configuration of the running environment for deploy
- For secuity
    - Check if HTML code is filtered
    - Check that the variable did escape before doing the database operation

Design Document

Architecture(ER diagram)



Timeline

Preparatory Phase and Little Test: **09.01 - 10.09**
The front end provides the interface definition. The back end presents the interface document and passes the review: **10.09 - 10.20**
Front and back end synchronous develop I: **10.20 - 11.03**
Front and back end synchronous develop II: **11.03 - 11.17**
Front and back end coordination: **11.17 - 12.01**
Software Test: **12.01 - 12.15**
Finishing touches: **12.15 - 12.22**

Divide the work

胡玉斌(11712121): Back-end
武羿(11712738): Database
张佳晨(11713020): Front-end
陈宇航(11712412): Connection front and end

APIs, services

- Django, Docker, Element, Express, Git, HTML/CSS, HTML Forms, JWT Authentication, Mysql, Node.js, Python, Vue.js

Feasibility

- Vue.js + Django + python's split front and back development model has matured and is widely used in project development
- Python development cycle is shorter than PHP, Java, also has a rich library
- vue.js has lightweight, high-performance and componentized MVVM library with very easy-to-use APIs
- There are many different Web frameworks in Python.Django is the most representative of the heavyweights.Many successful websites and apps are based on Django.Django is an open source Web application framework written in Python.