HydraCare

왕가혜

NAME	
왕가혜	1.INTRODUCTION(Motivation,client needs ,Problem Statement) 2.REQUIREMENTS(Functional Requirements,User design requirements,Database requirements,non-functional requirements) 3.DEVELOP ENVIRONMENT 4.REQUIREMENTS DETAILS((Functional Requirements,User design requirements,Database requirements,non-functional requirements) 5.STRUCTURE AND IMPLEMENT(Overall Architecture,Directory Organization,Modules) 6.USE CASE DISPLAY 7.REFERENCE 8.REPORT（1,2,3）

***Abstract*— This app is designed to help busy working and studying people develop a healthy habit of drinking water regularly. Users can track their daily water intake, set regular water reminders, view clear water intake statistics and get healthy drink suggestions**

I. INTRODUCTION

Motivation

With the acceleration of life pace, many white-collar workers often forget to drink water or delay drinking time due to their busy work. Long-term dehydration can lead to physical dehydration and decreased normal body function. Therefore, it is of great significance to develop a water reminder APP. This APP can track users' daily water intake, set water reminders, and pop up reminders when reaching the set time to help users develop the habit of drinking water periodically. At the same time, the APP can also recommend some healthy drinks, such as various fruit teas, homemade drinks, etc., to enrich users' choices.

client needs

Through market research and user interviews, we learned that users' main needs include: being able to set daily water intake goals and reminder times independently; water data needs to be displayed clearly and intuitively for users to view; the APP needs to have the function of reminding users to drink water on time; it can recommend some healthy drink options. Users expect the APP to be simple to operate, with good reminder effects, and can help them develop good drinking habits.

Problem Statement

The problem at hand is the potential for adverse health effects resulting from insufficient water intake due to a hectic lifestyle. Dehydration, accompanied by a decline in normal bodily functions, is a serious concern. Existing schedules and work commitments make it challenging for individuals to prioritize hydration. The need for an effective and user-friendly solution becomes evident, prompting the development of HydraCare. This app seeks to mitigate the consequences of inadequate water consumption by offering a comprehensive set of features designed to remind, track, and encourage users to drink water regularly.

II. REQUIREMENTS

Functional Requirements

The APP contains modules such as user registration, water settings, water reminders, water statistics, and healthy drink recommendations.

A.Water Settings

In this module, users can set their daily water intake goal, such as 2000 ml. Users can also set timed water

reminders based on their schedule, such as setting reminders to drink water at 10 am, 3 pm and 7 pm. Reminder methods include mobile notification reminders, alarm reminders, etc.

B.Water Reminders

When the set reminder time is reached, the APP will pop up a reminder to remind users to drink water. Notification forms may include text reminders, sound effects reminders, etc. If the user is temporarily unable to drink water, they can click to postpone the reminder.

C.Water Record

After each drink, users can record the amount of water drunk in the APP. An automatic recording mode can also be set to automatically record the amount of water drunk by connecting with smart water cups and other devices. The recorded drinking data will be updated in real time to the user's water statistics.

D.Water Statistics

This module will statistics the user's total daily water intake and compare it with the set goal to form an intuitive chart display for the user to clearly see their water intake. Users can choose different time dimensions (day/week/month) to view statistics.

E.Healthy Drink Recommendations

Recommend some healthy drinks based on user preferences, such as lemon water, honey grapefruit tea, green plum beer, etc., and provide recipes to enrich users' drink choices. Users can also evaluate the recommended drinks.

User design requirements

A. Login page

Provides a user-friendly login interface, including username/email and password.

B. Registration page

Provides a simple and intuitive registration process, including required fields such as username, password, email, and more. Send a confirmation email or text message to verify the new user's identity.

C. Drinking water settings

Create a clear interface that allows users to set daily water drinking goals. Includes reminder preferences, allowing users to select reminder times and notification preferences.

D.Daily chart statistics

Displays chart statistics of users' daily water consumption, showing users' drinking habits in an intuitive form. Provide time range selection..

E. personal information

Provides an interface for users to view and edit personal information. Allow users to change their password or update other personal information.

Database requirements

A. User information table

Stores basic user information, including user ID, user name, password, email, etc.

B. Water Setting Table

Store information related to the user's daily water intake goal and reminder preferences, including user ID (associated with the user information table), daily water intake goal, reminder time, notification preferences, etc.

C. Drinking water record form

Record each instance of the user drinking water, including user ID (associated with the user information table), water drinking volume, timestamp, etc.

non-functional requirements

Website introduction

HydraCare is focused on simplifying the process of drinking healthy water for you. HydraCare is a smart app dedicated to recording, analyzing and improving your water drinking habits. With user-friendly login and registration processes, personalized drinking water settings, real-time recording functions, and intuitive chart statistics, HydraCare provides a comprehensive range of healthy drinking water solutions.

III. DEVELOP ENVIRONMENT

A. Platform

- Windows

The software development platform I would choose is Windows, because Windows systems are widely used, with mature development and debugging tools, allowing for high development efficiency.

- Python

The programming language I would choose is Python, because Python is a simple, easy to learn yet powerful language, with abundant mature open source libraries available, suitable for rapid development. Also, Python

supports multi-platform, making cross-platform deployment easy in the future.

• Cost estimation
Software purchase: Visual Studio Code (free)
Server rental: lightweight application server, about $20 per month
Office computer: $1000

• Development environment
Operating system: Windows 10
Programming language: Python C.7
Development tools: Visual Studio Code
Database: MySQL 8.0
Version control: Git/GitHub
Cloud services can be used to provide computing and storage resources on a pay-as-you-go basis, reducing initial costs.

IV. REQUIREMENTS DETAILS

Functional Requirements Details

A.Water Settings
• Display page:
Input field: daily water goal amount
Time picker: reminder time 1
Time picker: reminder time 2
Time picker: reminder time 3
Radio buttons: reminder type (notification, alarm)
Save button
• Save input
goalAmount = get input field value
reminderTimesArray = get time picker values
reminderType = get radio button value
saveToDB(goalAmount, reminderTimesArray, reminderType)
• Trigger reminder
read reminder details from DB
if reminder type is notification:
  showNotificationPopup()
if reminder type is alarm:
  playAlarmSound()[1]

B. Water Reminder
• When reminder triggers:
showReminderPopup()
  text show "Drink water"
  button: Drink Now
  button: Snooze
• If Drink Now clicked:
  closePopup()
• If Snooze clicked:
  rescheduleReminder(later time)

C. Water Record
• Display page:
Input field: amount drank
• Save amount:
amount = get input field value
saveToDB(amount)
update daily total

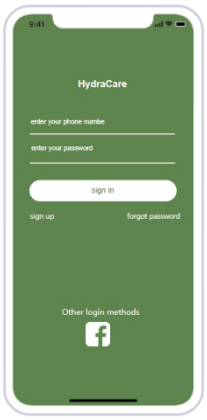
D . Water Statistics
read daily total from DB
calculate week total, month total
generate chart data points
• Display page:
show chart with day/week/month data
time range selector

E . Healthy Drink Recommendations
healthyDrinksList array
userDrinkRatings dictionary
• Display page:
show healthy drinks list
filter by user ratings
user rating input field
• Save rating:
get user rating
save to ratings dictionary

User Design Requirements Details

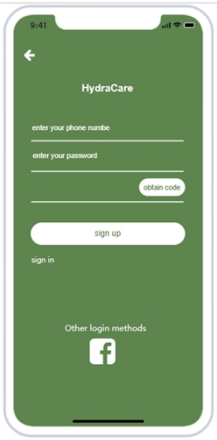
When designing the interface of the HydraCare app, I used Axure software and focused on user-friendliness and intuitiveness, ensuring that each function had clear and simple buttons to use.

A.Login page
On the login page, you'll find a bright and eye-catching "Log In" button designed to make it easy for users to find and perform the login action.



B.Registration page

On the registration page, a "Register" button is highlighted. After users fill in the necessary information, they can simply click this button to easily complete the registration.

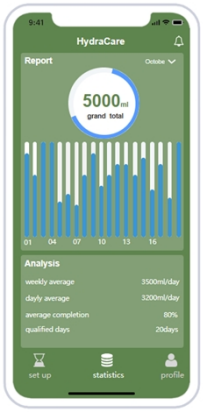


C.Drinking water settings

The water drinking settings page contains an "Add" button. Clicking this button will ensure that the user's water drinking goals and reminder preferences are effectively saved.

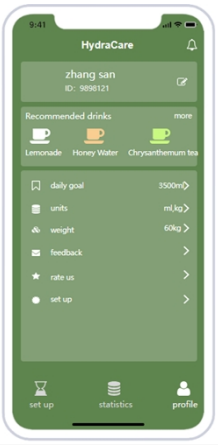
C.Daily chart statistics

On the Chart Statistics page, allow users to customize the time period for which statistics they want to view.



D.personal information

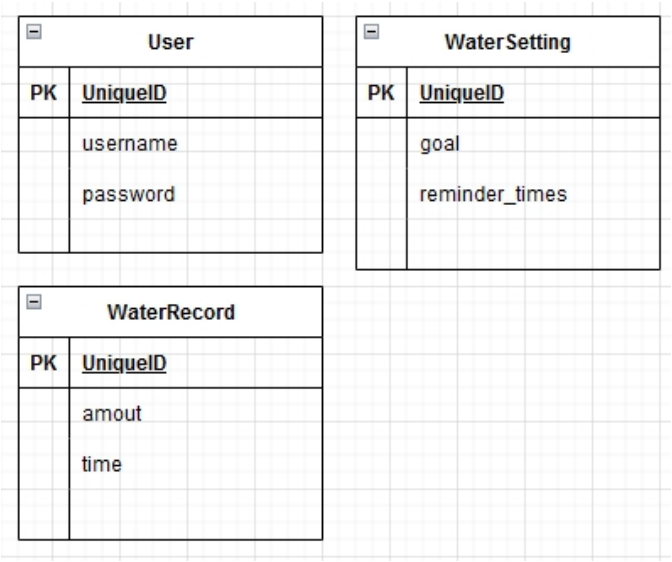
On the personal information page, users can easily modify their personal information.



Database Details

A.The main purpose of the APP's database is to store user information, water settings, water records, and other data. The technical solution is to use the MySQL relational database and store the needed data through the database table structure. The main table design includes:

- User Information Table: Stores user's basic information, account, etc.
- Water Setting Table: Stores the user's daily water intake goal, reminder times, etc.
- Water Record Table: Stores the user's each water drinking record, including the amount of water, time, etc.[2]

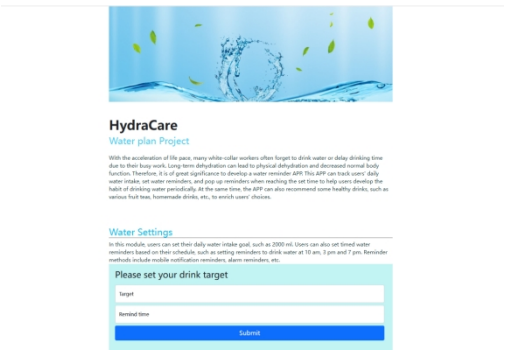


B. System architecture design ideas

Adopt C/S architecture, the client side is the web page, and the server side is the background service. The client submits and fetches data through HTTP requests, and the server handles requests and returns results.

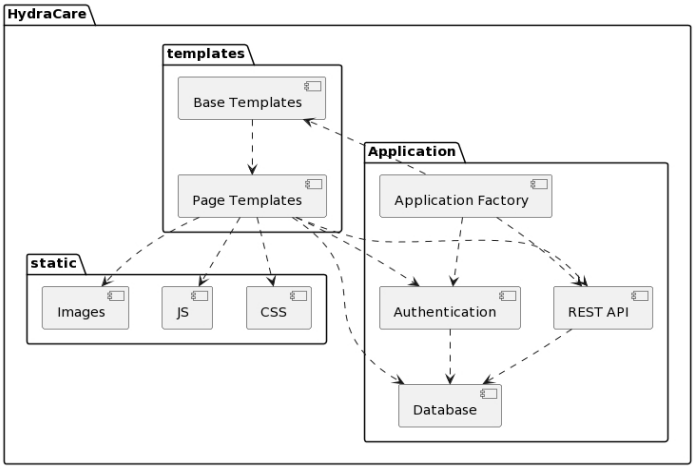
Non-functional Requirements

The HydraCare introductory page is designed to communicate the non-functional requirements and key features of the software in a clear and engaging manner. This introductory page is intended to interest users and give them a deeper understanding of HydraCare.



V. STRUCTURE AND IMPLEMENT

Overall Architecture



The main components

- templates: Provides page templates, including base templates and page templates
- static: Provides static resources, including CSS, JS and images
- application:
  - auth: Authentication module
  - db: Database operation module
  - rest: Provides REST API[3]
  - app: Application factory to create Flask app

The main relationships between components

- *Page templates depend on base templates*
- Page templates use Authentication, Database and REST API
- Authentication and REST API both depend on database module
- Application registers Authentication and REST API modules when created .It's based on Flask's MVC architecture, use template mechanism to implement pages. Authentication and REST API provide main functionality.

Directory Organization

Directory	File names	Description
functionde mo	__init__.py	The init file of the server modual
functionde mo	auth.py	Authority logic file to execute register and login logic of user
functionde mo	db.py	Database operation logic, store and load data from json file
functionde	rest.py	Hydracare business

mo		logic, to store user input information and record information.
functiondemo	requirements.txt	Server modual requirement file
functiondemo/static/css	Index.css Bootstrap.min.css	Css files of the pages
functiondemo/static/js	bootstrap.bundle.min.js echarts.min.js feather-4.28.0.min.js index.js jquery-3.5.1.js	Library javascript files
functiondemo/static/lib	axios.0.21.1.min.js bootstrap.bundle.min.js vue-2.6.12.js	Library javascript files
functiondemo/static/templates	404.html 500.html base.html base_login.html hello.html index.html login.html macros.html message.html record.html register.html settings.html statistics.html	Static page templates of hydra care project. Includes register pages, login pages, index pages and hydra care logic pages.

Modules

A. templates

- Purpose: Provide page templates for rendering UI
- Functions:
  - Define base templates with common UI elements
  - Implement specific page templates that extend base templates
- Code location: templates folder
- Components:
  - base.html: Base template
  - *.html: Page templates
- Source: Implemented for this application
- Usage:
  - Page templates inherit styles and structure from base templates
  - Views render page templates to generate UI

B. static

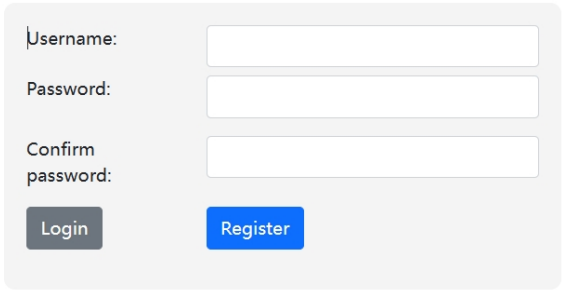
- Purpose: Provide static resources like CSS, JS, images
- Functions: Store resource files
- Code location: static folder
- Components:
  - CSS files
  - JS files
  - Image files
- Source: External libraries and custom code
- Usage:
  - Page templates link to CSS/JS to apply styling
  - Images displayed in pages

C. Application

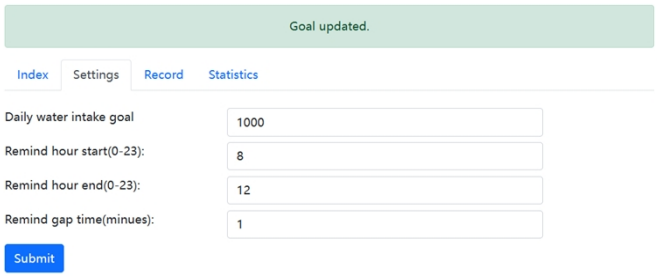
- Purpose: Create Flask application and implement app logic
- Functions:
  - App routing, request handling
  - Connect to data layer
  - Authentication and permissions
- Code location:
  - __init__.py - Application factory
  - auth.py - Authentication module
  - db.py - Database module
  - rest.py - REST API module
- Components:
  - Flask application object
  - App modules
- Source: Implemented for this application
- Usage:
  - Creates Flask app
  - Registers modules
  - Handles requests
  - Connects frontend to backend

VI. USE CASE DISPLAY

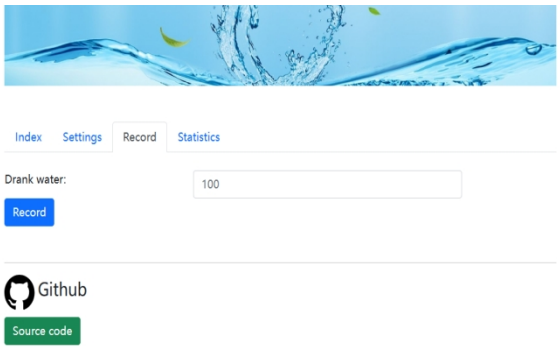
A. Register an account and set a password



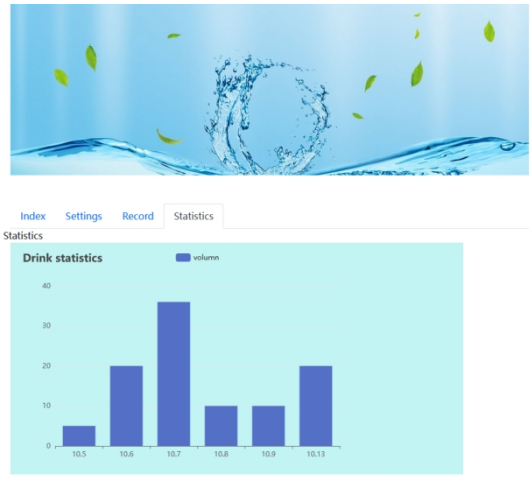
B. Set daily target water intake, start time, end time, and reminder interval



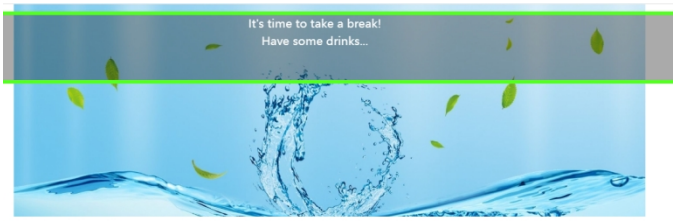
C. Record the amount of water you drink each time



D. Data statistics, users can visually see the data.



E. When the reminder time is up, a reminder to drink water will appear.



VII .REFERENCE

[1] Daily Drink Water Goal Project Using HTML CSS & Javascript, Daily Drink Water Goal Project Using HTML CSS & Javascript (codepen.io)

[2] Save data to local database using Room, https://developer.android.com/training/data-storage/room?hl=zh-cn

[3] REST APIs: How They Work and What You Need to Know, https://blog.hubspot.com/website/what-is-rest-api