Journal of Computing and Electronic Information Management
ISSN: 2413-1660 | Vol. 16, No. 3, 2025

The Design and Implementation of a Self-discipline Supervision Platform Based on WeChat Mini Program

Junyao Wu, Yujie Zhang, Zhiying Xu*

Wuhan Business University, School of Information Engineering, Hubei, Wuhan, China
* **Corresponding author Email:** 20160122@wbu.edu.cn

Abstract: With the improvement of social production, living standards, and the growth of people's income, China had fully entered a well-off society by 2020. However, this has been accompanied by the popularity of phenomena such as 'giving up' and 'lying flat,' leading to a lack of motivation to strive for many individuals who are content with the status quo. This project aims to respond to the national call by promoting self-discipline through a monetary incentive mechanism, helping users set and achieve personal goals, and conducting extensive questionnaire surveys to collect feedback and suggestions for improvement on the use of existing self-discipline mini-programs from different groups in order to develop a mini-program with broader applicability. Ultimately, users can earn coins by completing tasks, improve their living standards, and simultaneously help the impoverished gain income, thereby achieving dual benefits for society and the economy.

Keywords: Supervision Platform; WeChat Mini Program; Self-discipline.

1. Overview

In today's fast-paced society, college students face numerous temptations and challenges, with the lack of self-discipline becoming a stumbling block for many on the road to success. To address this issue, it is particularly important to develop a WeChat Mini Program that helps users improve their self-discipline through assigned tasks and mutual supervision among users. It is inevitable that one will encounter fatigue and confusion during implementation, so combining intrinsic motivation with external supervision can be beneficial [1].

Firstly, the core function of this mini program is task management. Users can set daily or weekly goals, such as reading, exercising, studying, etc., and record their progress within the mini program. By setting specific and quantifiable goals, users can have a clearer understanding of their direction of action, thereby improving their execution power.

Secondly, the mini program introduces a mutual supervision mechanism among users. Users can check each other's task completion status, encourage and remind each other to maintain discipline. The addition of this social element not only increases interactivity but also leverages group pressure to motivate users.

Additionally, the mini program can offer some motivational measures, such as rewards for completing tasks, to increase user enthusiasm. This positive feedback can give users a sense of accomplishment after completing tasks, further enhancing self-discipline. Utilizing the mini program for study achievement check-ins serves the purpose of regularly supervising learning outcomes [2].

However, the success of the mini program lies not only in the design of its functions but also in the user experience. The interface should be simple and easy to use, and the operation process should be as streamlined as possible to lower the barrier to usage. At the same time, while ensuring the security of user data, protecting user privacy is also of utmost importance, so that users can use the mini program with peace of mind.

In summary, this WeChat Mini Program, by combining task management, social supervision, and incentive mechanisms, provides an effective self-discipline tool for college students. It not only helps users overcome issues with poor self-control but also fosters habits of perseverance in achieving long-term goals, laying a solid foundation for future success [3]. With the promotion and improvement of the mini program, it is expected to become an important assistant for college students in their self-improvement journey.

2. Development Process

2.1. Development Environment

The development of WeChat Mini Programs is carried out using the WeChat Developer Tools. WeChat Developer Tools are an integrated development tool officially provided by WeChat, which supports developers in creating WeChat Mini Programs on Windows, macOS, and Linux operating systems [4] [5].

WeChat Developer Tools offer features such as code editing, debugging, simulator, project management, code uploading, version control, performance analysis, real-device debugging, plugin market, API documentation, and examples. They provide developers with API examples and code for WeChat Mini Programs, helping them to quickly learn and use the APIs provided by WeChat. With the real-device debugging feature, developers can connect to actual devices for debugging, more accurately simulating user scenarios. These features makes the development of WeChat Mini Programs more convenient and efficient.

WeChat Developer Tools is a vital part of the WeChat ecosystem, offering a complete solution from development to release, greatly enhancing the efficiency and quality of development. Developers can manage the entire lifecycle of a mini program within this tool.

2.2. Development Implementation

2.2.1. App.json File

In the WeChat Developer Tools, the app.json file serves as the global configuration file for WeChat Mini Programs. It defines the global settings of the mini program and the paths

to all pages, among other information. The configuration items within this file can be adjusted according to requirements. Mini program developers can modify these configuration items to meet the functional and design needs of the mini program. This file is crucial for the operation of the mini program.

The implementation code for the app.json file of this mini program is as shown in Figure 1.

```
"pages": [
  "pages/home/home",
  "pages/chenjin/chenjin",
  "pages/wode/wode",
  "pages/index/index",
  "pages/test/test",
  "pages/logs/logs",
  "pages/touru/touru",
  "pages/renwu/renwu"
],
"window": {
  "navigationBarTextStyle": "black",
  "navigationBarBackgroundColor": "#FFFFFF",
  "backgroundColor": "#efefef"
},
"tabBar": {
  "list": [
    {
      "pagePath": "pages/home/home",
      "text": " Task Hall",
      "iconPath": "/images/tabs/home.png.png",
      "selectedIconPath": "/images/tabs/home-active.png.png"
    },
    {
      "pagePath": "pages/chenjin/chenjin",
      "text": "Immersion Model",
      "iconPath": "/images/tabs/chenjin.png.png",
      "selectedIconPath": "/images/tabs/chenjin-active.png.png"
    },
    {
      "pagePath": "pages/wode/wode",
      "text": "Personal Center",
      "iconPath": "/images/tabs/user.png.png",
      "selectedIconPath": "/images/tabs/user-active.png.png"
    }
  ]
},
```

Figure 1. App.json part code 1

In the above code, the “Pages” field specifies the page paths included in the mini program, with each path corresponding to a page entry file, including pages such as the Task Hall, Publish Check-in, Immersive Mode, My Page, and Data Statistics.

The “Windows” field is used to set the status, navigation bar, title, and window background color of the mini program. It includes settings such as the navigation bar text style (navigationBarTextStyle), navigation bar background color (navigationBarBackgroundColor), and background color (backgroundColor).

The “tabBar” field is used to set the tabs at the bottom of the mini program, allowing for quick switching between different module functions. In the list of options, it includes information such as the page path (PagePath), display text (text), default icon (iconPath), and selected icon (selectedIconPath).

2.2.2. Chenjin.js file

The chenjin.js file is used to implement the immersive task feature. Users can select the task they want to focus on by clicking the icon, set the desired immersive time by moving the time bar, and then click “Start Focus” to enter the countdown page. During the immersive countdown, users can choose to pause or give up at any time.

The partial code for the chenjin.js file of this mini program is as shown in Figure 2.

```
drawBg: function() {
  const lineWidth = 6 / this.data.rate;//px
  const query = this.createSelectorQuery()
  query.select('#progress_bg')
    .fields({ node:true, size: true})
    .exec((res) => {
      const canvas = res[0].node
      const ctx = canvas.getContext('2d')
      const dpr = wx.getSystemInfoSync().pixelRatio
      canvas.width = res[0].width * dpr
      canvas.height = res[0].height * dpr
      ctx.scale(dpr, dpr)
      ctx.lineCap='round'
      ctx.lineWidth="lineWidth"
      ctx.beginPath()
      ctx.arc(400/this.data.rate/2,400/this.data.rate/2,400/this.data.rate/2-2*lineWidth,0,2*Math.PI,false)
      ctx.strokeStyle ="#000000"
      ctx.stroke()
    })
},
```

Figure 2. Chenjin.js part code 1

By importing the utils file, you can use the auxiliary functions and tools within it. The corresponding code for the utils file is as shown in Figure 3.

```
const formatTime = date => {
  const year = date.getFullYear()
  const month = date.getMonth() + 1
  const day = date.getDate()
  const hour = date.getHours()
  const minute = date.getMinutes()
  const second = date.getSeconds()

  return [year, month, day].map(formatNumber).join('-') + ' ' + [hour, minute, second].map(formatNumber).join(':')
}
```

Figure 3. Utils part code

In the utils file, a function named formaTime is defined, which accepts a data parameter that should be a JavaScript Date object. The purpose of the function is to format this date object into a string with the format YYYY-MM-DDHH:MM:SS.

Clockshow: Timer Page Display — When clockshow is initially set to false, it means the timer page is not displayed. When the user clicks ‘start’ to begin focusing, the clockshow value changes to true, thus displaying the timer page where the user can see the countdown.

DrawActive: This function draws an active circular progress bar, using setInterval to periodically update the state of the progress bar. Within it, a timer is defined, which is set to execute the passed parameter at regular intervals, and uses an if...else statement to determine whether to continue drawing the progress bar.

2.2.3. Home.js

In home.js, the implementation allows the user to see a successful coin investment prompt after clicking to invest coins, and adds the corresponding coins to the balance of the user who received the investment.

The corresponding partial code for home.js is as shown in Figure 4.

```
tanc:function(){
  wx.showToast({
    title: 'Successful',
    duration:1000,
    icon:"success",
    mask:true
  })
```

Figure4. Home.js part code

If the user who has received the investment wants to check in for a task, they must do so on the “My Tasks” page. After completing the check-in, the “Check-in Success” page is

displayed, and the “Check-in” option changes to “Checked-in Today”.

The corresponding code in renwu.js is as shown in Figure5.

```
tanc1:function(){
  this.setData({
    message:"Punch Clock Today"
  })
  wx.showToast({
    title: 'Successful',
    duration:1000,
    icon:'success',
    mask:true
  })
}
```

Figure5. Renwu.js part code

3. Core Function Implementation

3.1. Main Functions of the Mini Program

The “Mind Rudder” WeChat Mini Program includes a Task Hall, Immersive Mode, and Personal Center. The functional architecture of the mini program is shown in Figure 6.

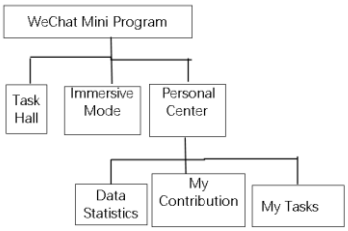

Figure 6. Mini Program Functional Architecture

3.2. Unique Features of the Mini Program

The mini program, targeting the issue of college students lacking sufficient intrinsic motivation to set and execute plans, has designed a supervision interaction module. This module aims to enhance motivation through the supervision of others, thereby promoting plan execution and increasing the sense of accomplishment in completing tasks.

Users can publish supervision projects to the plaza, setting the amount of gold coins, check-in tasks, project duration, etc. The project starts checking in on the day it is successfully published, and other users can selectively join as supervisors based on the task details. The tasks of those being supervised are monitored by others within the user-defined task cycle and require mandatory check-ins. If the supervised project is successfully completed, the supervisors will reward all the gold coins spent on joining the project to the person being supervised; if the supervised project fails to check in, the supervisors will split the gold coins invested by the supervised person when creating the project based on the duration of their involvement.

In this process, users are driven to complete tasks due to the pressure of being supervised and the desire to avoid losing gold coins, thereby effectively improving their execution.

3.3. Supervision Project Interaction

Users can simultaneously act as both supervisors and the supervised.

3.3.1. Supervisor’s Perspective

Supervisors can browse supervision projects in the plaza, spend gold coins to join the projects they wish to supervise, and after joining a project, they can view the check-in status of the person being supervised, as shown in Figure7.

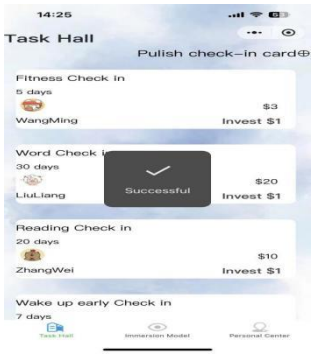

Figure 7. Task Hall Interface

If the supervisor’s participation in the supervision project successfully ends with a check-in, then the coins spent by the supervisor when joining the project will be fully rewarded to the supervisee. If the supervisor’s participation in the supervision project results in a check-in failure, then the supervisor will divide the coins invested by the supervisee when creating the project according to the duration of their involvement.

3.3.2. From the Perspective of the Supervisee

The supervisee can post supervision projects on the square, setting the amount of coins, check-in tasks, project duration, etc., as they see fit. Once the project is successfully published, check-ins begin on the same day, as shown in Figure8.

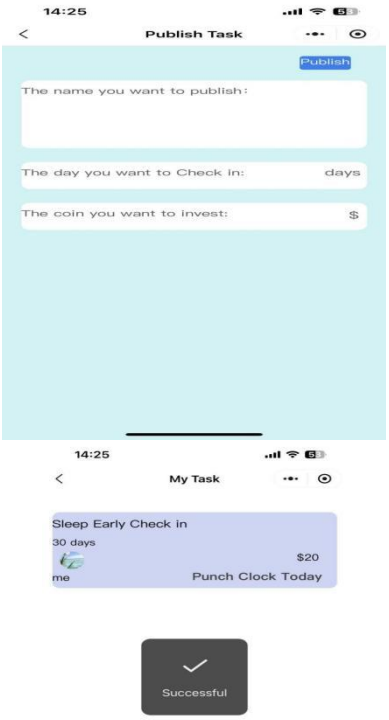

Figure 8. Task Posting and Check-in Interface

If the supervised party successfully completes the check-in for the supervision project, then all the coins spent by the supervisor when joining the project will be rewarded to the supervised party; if the supervised party fails to check in for the supervision project, then the supervisor will divide the coins invested by the supervised party when creating the project according to the duration of their involvement.

3.4. Immersive Mode

The immersive mode is a tool for calculating study duration, and an immersive environment is a prerequisite for adolescent self-discipline. [4] Users can set their study tasks independently on this page. Once the task is selected and the duration is set, the user will enter immersive mode. In immersive mode, the page is very tidy, with only a countdown clock displayed. In this state, the phone will automatically enter do-not-disturb mode, and a prompt will be given when the countdown ends.

In immersive mode, if the user taps pause, the countdown clock will stop, and buttons for “Continue” and “Give Up” will be displayed below the clock. Tapping “Continue” will resume the countdown in real-time; tapping “Give Up” will end the countdown prematurely, returning to the initial page, as shown in Figure9.

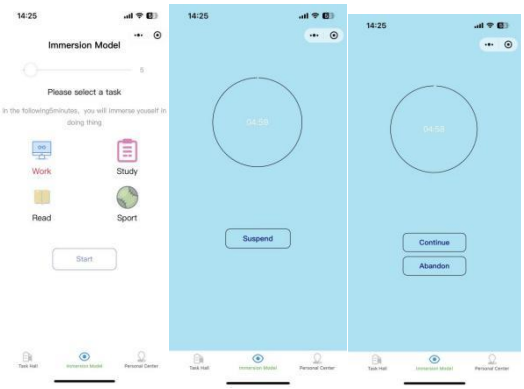

Figure 9. Immersive Mode Interface

3.4.1. User Data Statistics and Analysis

User data is primarily divided into three parts: account basic data, supervision project data, and immersive mode data.

3.4.2. Account Basic Data

Account basic data includes user avatar, user name, account coin balance, and the number of tasks invested in. The user avatar, user name, and user signature are customizable, with each user allowed to make changes once every 30 days. The account coin balance is generated by the system and will automatically increase or decrease after the completion and settlement of the supervision projects in which the user participates, as shown in Figure 10.

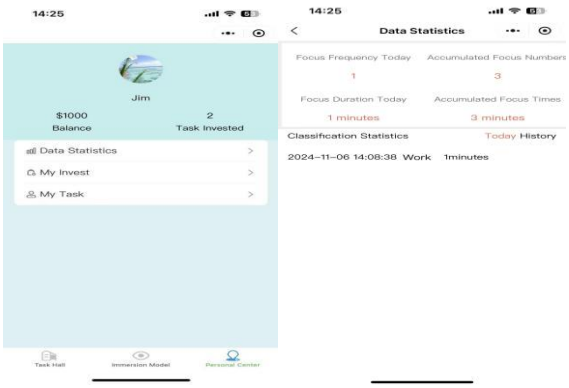

Figure 10. User Data Interface

3.4.3. 3.4.3 Supervision Project Data

In the supervision project statistics, users can view the supervision projects they have published (My Tasks) and the projects they have participated in (My Investments). Users can rely on this data to understand the completion status of the projects, as shown in Figure 11.

Figure 11. “My Investments” and “My Tasks” Interface

3.4.4. Immersive Mode Data

The system provides statistical data on “Today’s Focus Sessions,” “Total Focus Sessions,” “Today’s Focus Duration,” and “Total Focus Duration” based on the user’s usage of immersive mode, as shown in Figure 12.

Figure 12. Data Statistics Interface

4. Conclusion

The project provides a detailed introduction to the design and implementation process of a self-discipline supervision platform based on the WeChat Mini Program. The interactive supervision projects can motivate the supervised individuals to generate a sense of learning drive, thereby completing their learning tasks more comprehensively within the specified time. The immersive mode can help users eliminate distractions and generate learning data, allowing users to fully understand their own learning status. It is hoped that this self-discipline supervision platform can help users gain learning motivation, solve procrastination issues, and enable users to have a more thorough understanding of themselves.

Funding

This paper is published with the funding of the 2023 Wuhan Business University Student Innovation and Entrepreneurship Project (Project Number 202311654200).

References

[1] Ding Yi. The Counterattack of "Average-level Students" [J]. Zhejiang Examinations, 2024, (10): 59-60.

[2] Gu Liyan, Wu Xinyu, Sun Chaowei, et al. Exploring the Training Mode of Talents with Comprehensive Qualities in Traditional Chinese Medicine Colleges and Universities under the Background of the Internet [J]. Guangming Journal of Chinese Medicine, 2024, 39(18): 3776-3779.

[3] Wei Lei, Zhu Shuai. A Brief Discussion on How College Students Can Improve Their Time Management Ability [N]. Xinxiang Daily, 2024-08-16(008). DOI: 10.28909 / n.cnki.nxxbr.2024.001317.

[4] Chen Gang. Rational Thinking on the Immersive Trend of Media Education for Adolescents [J]. Shanghai Research on Education, 2018, (07): 37-41 + 36. DOI:10.16194/j.cnki.31-1059/g4.2018.07.009.

[5] Qianfeng Education. Development and Practice of WeChat Mini Programs [M]. Posts and Telecom Press:202209.367.