Báo cáo tuần 5:

-Tên nhóm: SE\_CNTT5

-Thành viên nhóm

1. Nguyễn Văn Sĩ\*\*\*\*\*
2. Nguyễn Công Hùng\*\*\*\*
3. Tạ Bá Sáng\*\*
4. Đào Bá Đạt\*\*
5. Đỗ Thế Toàn\*\*

-Bài toán: Quản lý chi tiêu trực tuyến của sinh viên trường ĐH Phenikaa.”

I. Yêu cầu chức năng:

1. Đăng nhập và đăng ký:
2. Cho phép sinh viên đăng ký tài khoản để sử dụng dịch vụ quản lý chi tiêu trực tuyến.
3. Sinh viên phải nhập đúng thông tin cần thiết và đảm bảo tính bảo mật cho tài khoản của mình.
4. Cho phép sinh viên đăng nhập vào hệ thống để sử dụng các tính năng quản lý chi tiêu.
5. Quản lý thu nhập:
6. Cho phép sinh viên nhập các khoản thu nhập của mình như lương, học bổng, tiền thưởng, thu nhập từ việc làm thêm,...
7. Cung cấp các công cụ thống kê để sinh viên có thể xem tổng thu nhập trong một khoảng thời gian nhất định.
8. Quản lý chi tiêu:
9. Cho phép sinh viên nhập các khoản chi tiêu của mình như tiền ăn, tiền điện, tiền nước, tiền sách, tiền học phí, tiền đi lại,...
10. Cung cấp các công cụ thống kê để sinh viên có thể xem tổng chi tiêu trong một khoảng thời gian nhất định.
11. Tính toán và cung cấp thông tin về số tiền còn lại trong tài khoản của sinh viên.
12. Tính năng nhắc nhở:

Cung cấp thông báo nhắc nhở cho sinh viên về các khoản chi tiêu sắp tới hoặc vượt quá ngân sách đã đặt ra.

1. Tính năng lên kế hoạch:
2. Cung cấp các công cụ để sinh viên có thể lên kế hoạch chi tiêu cho các khoản thu nhập của mình.
3. Các kế hoạch này có thể được lưu trữ để sinh viên có thể xem lại và điều chỉnh.
4. Cập nhật thông tin tài khoản:
5. Cho phép sinh viên cập nhật thông tin tài khoản của mình như thông tin cá nhân, đổi mật khẩu,..

II. Yêu cầu phi chức năng

1. Thiết kế giao diện đơn giản và dễ sử dụng:
2. Hệ thống cần có giao diện đơn giản, dễ sử dụng và thân thiện với người dùng.
3. Giao diện phải được thiết kế để đảm bảo tính tương thích với các thiết bị di động, vì đây là một tính năng quan trọng để sinh viên có thể sử dụng dịch vụ bất cứ khi nào và ở bất kỳ đâu.
4. Độ tin cậy và ổn định:
5. Hệ thống cần đảm bảo độ tin cậy và ổn định cao để tránh các lỗi hệ thống, giảm thiểu thời gian bảo trì và cải thiện trải nghiệm người dùng.
6. Hệ thống cần được thiết kế để đảm bảo tính sẵn sàng hoạt động, có khả năng phục hồi dữ liệu nhanh chóng trong trường hợp xảy ra sự cố.
7. Khả năng mở rộng:
8. Hệ thống cần được thiết kế để có khả năng mở rộng, đáp ứng được nhu cầu sử dụng của người dùng trong tương lai.
9. Hệ thống cần hỗ trợ tính năng tích hợp với các dịch vụ khác để cung cấp cho sinh viên nhiều lựa chọn hơn trong việc quản lý chi tiêu của mình.
10. Độ tương thích:
11. Hệ thống cần đảm bảo tính tương thích với các hệ thống khác để có thể tích hợp dữ liệu từ các nguồn khác nhau.
12. Hệ thống cần hỗ trợ đa nền tảng để có thể sử dụng trên các hệ điều hành và trình duyệt khác nhau.
13. Hỗ trợ khách hàng:
14. Hệ thống cần hỗ trợ khách hàng nhanh chóng và hiệu quả trong trường hợp sinh viên gặp sự cố trong quá trình sử dụng.
15. Hệ thống cần cung cấp các kênh hỗ trợ như hỗ trợ trực tuyến, email, điện thoại,... để giải quyết các vấn đề của sinh viên.
16. Độ bảo mật cao:
17. Hệ thống cần đảm bảo độ bảo mật cao để bảo vệ thông tin cá nhân và tài khoản của sinh viên.
18. Hệ thống cần sử dụng các công nghệ bảo mật như mã hóa dữ liệu, chứng thực và phân quyền để đảm bảo tính bảo mật của hệ thống.
19. Hệ thống cần có các biện pháp bảo vệ chống lại các cuộc tấn công mạng, sử dụng các giải pháp bảo mật mạnh mẽ như tường lửa (firewall), phát hiện xâm nhập (intrusion detection), chống virus và spam.
20. Quản lý và bảo vệ dữ liệu:
21. Hệ thống cần có cơ chế quản lý và bảo vệ dữ liệu để đảm bảo tính toàn vẹn, khả năng phục hồi và bảo mật của dữ liệu.
22. Hệ thống cần có cơ chế sao lưu dữ liệu thường xuyên để đảm bảo tính an toàn của dữ liệu trong trường hợp xảy ra sự cố.
23. Hệ thống cần có chính sách bảo mật rõ ràng và được thực hiện nghiêm ngặt để đảm bảo tính bảo mật của dữ liệu.
24. Đảm bảo tuân thủ pháp luật và quy định:
25. Hệ thống cần tuân thủ các quy định của pháp luật về bảo mật thông tin và quyền riêng tư.
26. Hệ thống cần có chính sách bảo mật và chính sách bảo vệ thông tin riêng tư rõ ràng để đảm bảo tính tuân thủ và giữ gìn uy tín của hệ thống.

III.Yêu cầu người dùng:

1. Giao diện thân thiện và dễ sử dụng:
2. Hệ thống cần có giao diện thân thiện, trực quan, dễ sử dụng và dễ tìm kiếm thông tin để giúp người dùng có trải nghiệm tốt nhất khi sử dụng hệ thống.
3. Hệ thống cần cung cấp hướng dẫn sử dụng rõ ràng và dễ hiểu để giúp người dùng tiếp cận và sử dụng hệ thống một cách dễ dàng.
4. Tính năng đa dạng và phù hợp với nhu cầu người dùng:
5. Hệ thống cần cung cấp các tính năng đa dạng và phù hợp với nhu cầu của người dùng, bao gồm quản lý chi tiêu, theo dõi tài khoản, phân tích và báo cáo chi tiêu.
6. Hệ thống cần cho phép người dùng tùy chỉnh và thêm các loại chi tiêu và tài khoản riêng của mình để quản lý một cách linh hoạt và thuận tiện.
7. Tính khả dụng và ổn định cao:
8. Hệ thống cần đảm bảo tính khả dụng và ổn định cao để người dùng có thể sử dụng hệ thống mọi lúc mọi nơi một cách thuận tiện và tin cậy.
9. Hệ thống cần được thiết kế để xử lý lượng dữ liệu lớn và đáp ứng tốt với tình trạng tải cao trong các thời điểm sử dụng nhiều nhất.
10. Hỗ trợ và chăm sóc khách hàng:
11. Hệ thống cần cung cấp kênh hỗ trợ và chăm sóc khách hàng để giải đáp các thắc mắc và hỗ trợ người dùng khi gặp sự cố hoặc có yêu cầu hỗ trợ.
12. Hệ thống cần có cơ chế phản hồi nhanh chóng và giải quyết vấn đề hiệu quả để giữ gìn sự hài lòng của khách hàng.

IV. Yêu cầu hệ thống:

1. Đảm bảo bảo mật thông tin:
2. Hệ thống cần đảm bảo an toàn thông tin của người dùng, bao gồm các thông tin cá nhân, tài khoản và giao dịch.
3. Hệ thống cần có các biện pháp bảo mật như mã hóa dữ liệu và chứng thực người dùng để ngăn chặn các cuộc tấn công từ bên ngoài.
4. Độ tin cậy cao:
5. Hệ thống cần đảm bảo tính độ tin cậy cao để người dùng có thể sử dụng hệ thống một cách tin tưởng và không lo lắng về sự cố hệ thống hay mất dữ liệu.
6. Hệ thống cần được kiểm tra, giám sát và bảo trì định kỳ để đảm bảo tính độ tin cậy và khả năng phục hồi sau sự cố.
7. Hỗ trợ đa nền tảng:

Hệ thống cần hỗ trợ đa nền tảng, bao gồm các thiết bị di động và máy tính để bảng để người dùng có thể sử dụng hệ thống một cách linh hoạt và thuận tiện.

1. Tính mở rộng và tương thích:

Hệ thống cần được thiết kế để có tính mở rộng và tương thích với các hệ thống khác để có thể tích hợp với các hệ thống thanh toán trực tuyến hoặc các ứng dụng quản lý tài chính khác.

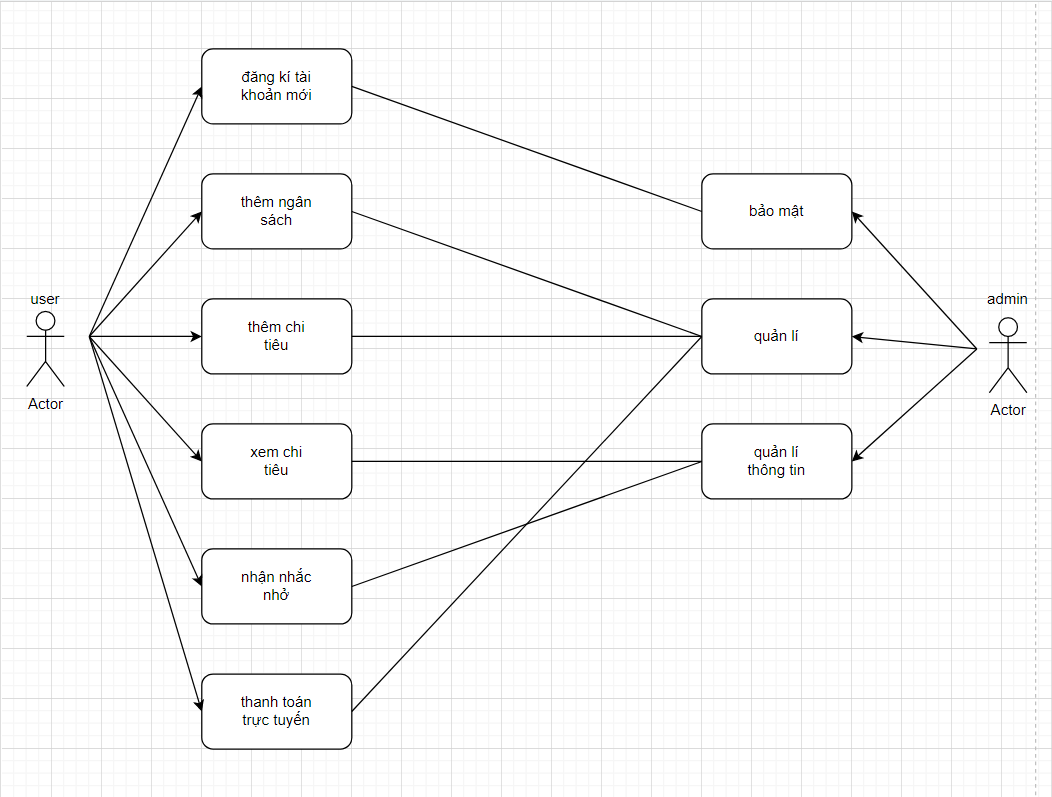
1. Tính dễ bảo trì và nâng cấp:

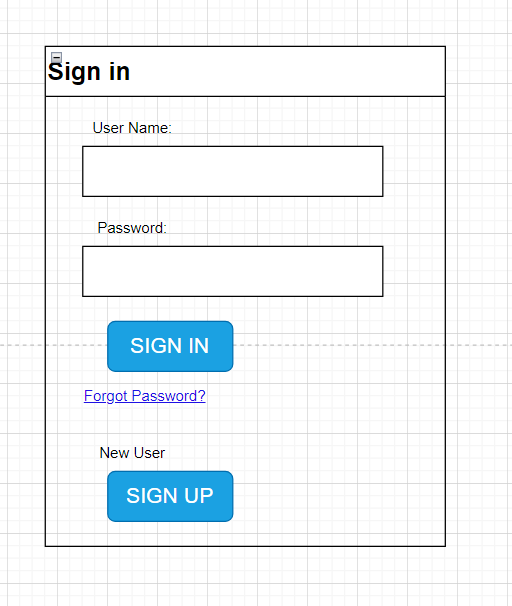
Hệ thống cần được thiết kế để dễ dàng bảo trì và nâng cấp, bao gồm việc tối ưu hóa cấu trúc dữ liệu và phần mềm để giảm thiểu các lỗi hệ thống và tăng tính ổn định của hệ thống.

V. Kịch bản

1. Đăng ký tài khoản:
2. Người dùng có thể truy cập vào trang đăng ký tài khoản trên website hoặc ứng dụng di động của hệ thống.
3. Người dùng cung cấp thông tin cá nhân, địa chỉ email và số điện thoại để đăng ký tài khoản.
4. Hệ thống sẽ gửi một email xác nhận cho người dùng và yêu cầu xác thực tài khoản.
5. Thêm ngân sách và chi tiêu:
6. Người dùng đăng nhập vào hệ thống và truy cập vào mục "Ngân sách" trên trang cá nhân của mình.
7. Người dùng nhập thông tin về ngân sách của mình, bao gồm tên ngân sách, số tiền cần tiết kiệm và thời gian để đạt được mục tiêu.
8. Người dùng cũng có thể thêm chi tiêu vào danh sách của mình bằng cách nhập thông tin về món chi tiêu, số tiền và thời gian.
9. Theo dõi chi tiêu:
10. Hệ thống sẽ tổng hợp thông tin về chi tiêu của người dùng dựa trên các giao dịch được thực hiện bằng thẻ tín dụng hoặc tài khoản ngân hàng đã liên kết với hệ thống.
11. Người dùng có thể theo dõi các giao dịch này trên trang cá nhân của mình và phân tích chi tiêu của mình bằng các biểu đồ và báo cáo.
12. Nhắc nhở và cảnh báo:
13. Hệ thống sẽ gửi thông báo nhắc nhở và cảnh báo cho người dùng khi họ tiêu quá mức ngân sách hoặc gần đạt đến giới hạn thời gian đã đặt ra.
14. Hệ thống cũng có thể cung cấp các gợi ý về cách tiết kiệm hoặc giảm chi tiêu cho người dùng.
15. Tích hợp thanh toán trực tuyến:
16. Hệ thống có thể tích hợp với các cổng thanh toán trực tuyến để người dùng có thể thực hiện các giao dịch thanh toán một cách thuận tiện và an toàn.
17. Người dùng có thể liên kết tài khoản ngân hàng hoặc thẻ tín dụng với hệ thống để thực hiện các giao dịch này.

VI. Sơ đồ use-case



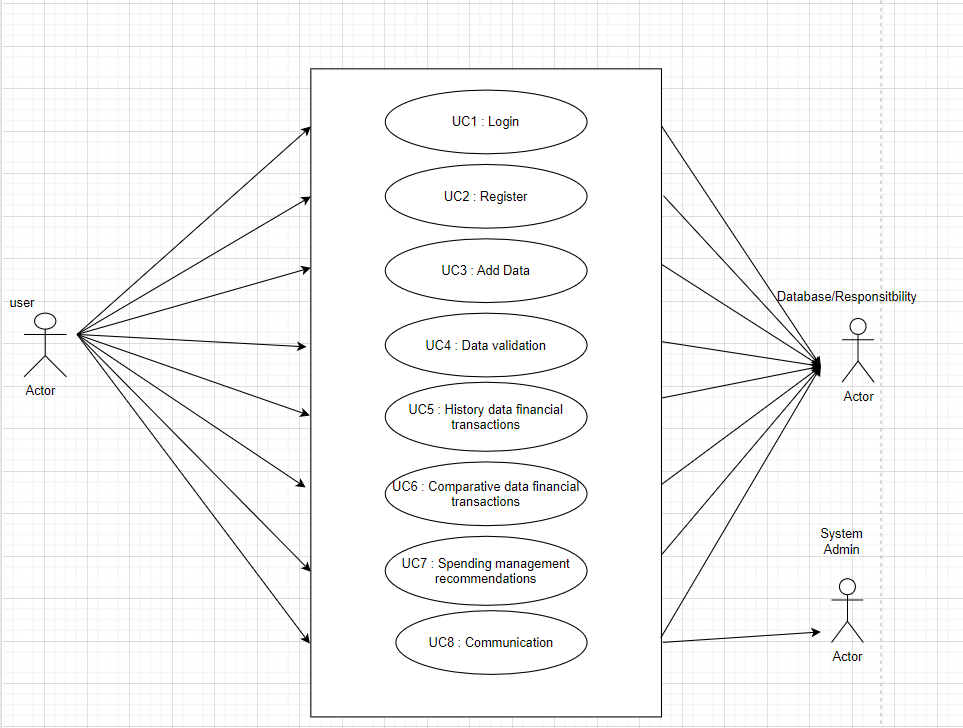


III. Các đặc tả yêu cầu chức năng

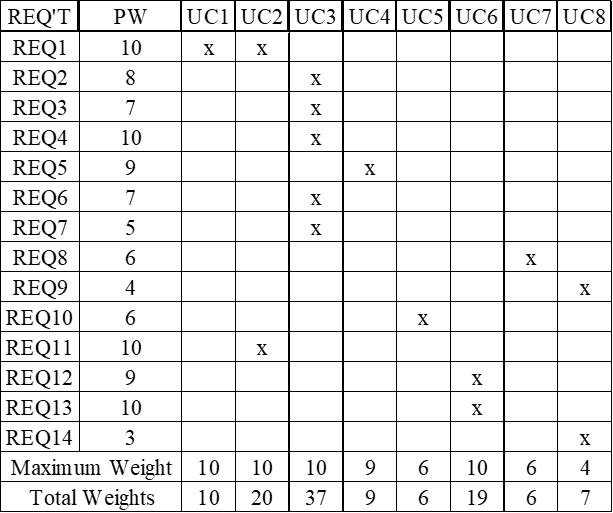
1. Bảng tác nhân và mục đích

|  |  |  |
| --- | --- | --- |
| Actor | Goals | Use-case name |
| Visitor/User | To login to the web/app portal to enter spending data, view stats, get spending recommendations, view comparison reports with previous transactions, etc. | Login (UC-1) |
| Visitor/User | To register in the system/application via web portal | Register (UC-2) |
| Visitor/User | To add data in the system/application via web portal | Add Data (UC-3) |
| Visitor/User | To have its data validated before adding data in the system | Data validation (UC-4) |
| Visitor/User | To view historical spending data for analysis | History data financial transactions(UC-5) |
| Visitor/User | To compare spending data with other registered and active data user | Comparative data financial transactions(UC-6) |
| Visitor/User | To get data-driven spending recommendations access into the system | Spending management recommendations(UC-7) |
| Visitor/User | To resolve issues that users might face while user performs any operations or has any other issues | Communication(UC-8) |

1. Use-case diagram



IV. Ma trận truy xuất



V. Miêu tả đầy đủ cho các Use cases

|  |
| --- |
| **Use Case UC#2: New Registration**  **Related Requirements:** REQ-1,REQ-11  **Initiating Actor:** Users  **Actor’s Goal:** To allow user to have secure login.  To allow user to unregister and delete the data.  **Participating Actors:** Active users,System Admin,System  **Precondition:** System should be active and running.  User should have established connection with the system.  User should meet the requirement for registering.  **Postcondition:** User can enter his data in the system.  User can update or delete his data as per his requirement.  User will have option to share/hide his data from other users.  **Failed end condition:** User entering wrong credentials, re-enter valid credentials  Unable to correct to system,check system status |
| Flow of events for success scenario:   1. → **User** enters his credentials into the portal 2. .← The credentials are verified by the system. 3. .→ The **user** is allowed to access the portal.   Flow of events for extensions:  User enters invalid/out-of-bounds data.   1. ← **System** detects the error and sends an error message to the user prompting to enter the data again.   User enters a weak password while registering.  1. ← **System** should suggest a strong password to the user in order to keep the data more secure. |

|  |
| --- |
| **Use Case UC#3: User Data Addition**  **Related Requirements:** REQ-2, REQ-3, REQ-4, REQ-6, REQ-7  **Initiating Actor:**  Users  **Actor’s Goal:** To add and update data which is within appropriate range.  To append encrypted data into the blockchain with Verification from all users.  To inform the user if any new parameter is available in the System for monitoring.  **Participating Actors:**  User fitness device  System, other Active Users, System admin  **Precondition:**  User should be active on the system  User’s communication link with the system should be online  The blockchain must be initialised and active.  **Postcondition:**  New encrypted data blocks should be appended to the blockchain and updated across all the users present in the network.  Failed end condition: Data input unsuccessful, check data range  Unable to connect with system, check connection |
| Flow of events for success scenario:   1. → **User** enters the web application and chooses option to input data. 2. → **User** gives permission to the system to access to financial management software. 3. → **User** inputs data. 4. **System** uses the data input by the **user** to make a new block 5. include:: ValidationOfData (UC-4) 6. ← **System** adds the new validated block to the blockchain and gives user success message Flow of events for extensions:   3a. **User** inputs invalid/out-of-bounds data   1. ← **System** detects error and sends an error message to the **user** asking to re-input the data. 2. → **User** inputs correct data   5a. Failure to validate new block from more than 50% active **Users.**   1. ← **System** a) detects error, b) marks failed attempt, c) signals to **System Admin** 2. ← **System Admin** will try to detect issue (maybe problem with hashing function)   3. → **User** data gets validated. |

|  |
| --- |
| **Use Case UC#6: Comparison of Data Report**  **Related Requirements:**  REQ-12, REQ-13  **Initiating Actor:** Users  **Actor’s Goal:**  To provide users the option to choose to share their data for statistical comparison.  To generate the relevant histograms, pie charts, scatter plots so that the user can compare his data with other users  **Participating Actors:**  Active users, Other Users, System  **Precondition:**  User should be active on the system.  User must have the links available for selecting or deselecting the information sharing option and to view their comparison report.  System must have the active users who have selected the option of sharing their data for comparisons.  User must have actively participated in the block chain system based on which the user must have maintained the statistical data block chains for other users.  **Postcondition:**  User must be able to be or not be a part of the comparison subsystem of our system and can accordingly view the comparison report while having the comparison of its own data with the statistical information maintained based on other users’ data.  **Failed end condition:**  Not able to view the comparison data properly (maybe only his own data without any comparison or with partial comparison) ,User unable to connect to comparison page (Error message page displayed) |
| **Flow of events for success scenario:**   1. → User can select/deselect the option of being part of the comparison sub system.   2. ← System can provide the user with the services accordingly  3. → User click the link for accessing the comparison report page.  4. ← System provides the user with the comparison report.  5. → User can view the comparison report comparing his/her data with the statistical information derived out of other users’ data.  **Flow of events for extensions:**   1. When users select the option for doing some necessary selection to be the part of comparison of subsystem and some abnormality happens. 2. User can then raise the issue with the system administrator. 3. User click the link and error message is displayed or improper comparison report is visible.   1. User can raise the issue with the system administrator. The user can check its connection and troubleshoot system from its end. |