|  |  |
| --- | --- |
|  |  |
| NGUYỄN QUỐC DUY | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
| **Logo%20HaUI%20ban%20chuan** |
| ĐA, KLTN ĐẠI HỌC/ CAO ĐẲNG CÔNG NGHỆ THÔNG TIN |
|  |
| **XÂY DỰNG ỨNG DỤNG ĐIỀU KHIỂN TRANG TRẠI ĐIỆN THOẠI ANDROID** |
|  |
|  |
| **CBHD: TS. Đặng Trọng Hợp** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên: Nguyễn Quốc Duy** |
| **Mã số sinh viên: 2019603906** |
|  |
|  |
|  |
| Hà Nội – 2023 |
|  |

|  |
| --- |
|  |
| **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
| **Logo%20HaUI%20ban%20chuan** |
| ĐA, KLTN ĐẠI HỌC/ CAO ĐẲNG CÔNG NGHỆ THÔNG TIN |
|  |
| **XÂY DỰNG ỨNG DỤNG ĐIỀU KHIỂN TRANG TRẠI ĐIỆN THOẠI ANDROID** |
|  |
|  |
| **CBHD: TS Đặng Trọng Hợp** |
| **Sinh viên: Nguyễn Quốc Duy** |
| **Mã số sinh viên: 2019603906** |
|  |
|  |
|  |
| Hà Nội – 2023 |
|  |

MỤC LỤC

[Danh mục các ký hiệu, các chữ viết tắt 1](#_Toc134017854)

[Danh mục các bảng 2](#_Toc134017855)

[Danh mục hình vẽ và đồ thị 3](#_Toc134017856)

[**LỜI NÓI ĐẦU** 4](#_Toc134017857)

[**CHƯƠNG 1. TỔNG QUAN** 5](#_Toc134017858)

[**1.1.** **Hệ điều hành Android và công cụ Android Studio** 5](#_Toc134017859)

[**1.2.** **Công nghệ Windows Presentation Foundation** 6](#_Toc134017860)

[**1.3.** **Mô hình MVVM** 8](#_Toc134017861)

[**1.4.** **ADB - Android Debug Bridge** 9](#_Toc134017862)

[**1.5.** **Các hoạt động trên nền tảng Google** 11](#_Toc134017863)

[**1.5.1.** **Tạo tài khoản** 11](#_Toc134017864)

[**1.5.2.** **Xem video trên Youtube** 11](#_Toc134017865)

[**1.5.3.** **Theo dõi kênh Youtube** 11](#_Toc134017866)

[**1.6.** **Các hoạt động trên nền tảng Facebook** 12](#_Toc134017867)

[**1.6.1.** **Tạo tài khoản Facebook** 12](#_Toc134017868)

[**1.6.2.** **Theo dõi trang cá nhân** 12](#_Toc134017869)

[**1.7.** **Visual Studio 2022** 13](#_Toc134017870)

[**CHƯƠNG 2. PHÂN TÍCH, THIẾT KẾ HỆ THỐNG** 14](#_Toc134017871)

[**2.1.** **Biểu đồ use case** 14](#_Toc134017872)

[**2.1.1.** **Biều đồ use case tổng quan** 14](#_Toc134017873)

[**2.1.2.** **Biểu đồ phân rã use case** 15](#_Toc134017874)

[**2.1.2.1.** **Use case Tạo tài khoản Google** 15](#_Toc134017875)

[**2.1.2.2.** **Use case Tạo tài khoản Facebook** 15](#_Toc134017876)

[**2.1.2.3.** **Use case Tăng lượt xem** 16](#_Toc134017877)

[**2.1.2.4.** **Use case Tăng lượt đăng ký** 16](#_Toc134017878)

[**2.1.2.5.** **Use case Tăng lượt theo dõi** 17](#_Toc134017879)

[**2.2.** **Mô tả chi tiết các use case** 17](#_Toc134017880)

[**2.2.1.** **Use case Thông tin hệ thống** 17](#_Toc134017881)

[**2.2.2.** **Use case Quản lý thiết bị** 18](#_Toc134017882)

[**2.2.3.** **Use case Quản lý tài khoản Facebook** 18](#_Toc134017883)

[**2.2.4.** **Use case Quản lý tài khoản Google** 19](#_Toc134017884)

[**2.2.5.** **Use case Tạo tài khoản Google** 20](#_Toc134017885)

[**2.2.6.** **Use case Tăng lượt xem** 20](#_Toc134017886)

[**2.2.7.** **Use case Tăng lượt đăng ký** 21](#_Toc134017887)

[**2.2.8.** **Use case Tạo tài khoản Facebook** 22](#_Toc134017888)

[**2.2.9.** **Use case Tăng lượt theo dõi** 23](#_Toc134017889)

[**2.3.** **Biểu đồ trình tự** 24](#_Toc134017890)

[**2.3.1.** **Use case Quản lý tài khoản** 24](#_Toc134017891)

[**2.3.2.** **Use case Quản lý thiêt bị** 25](#_Toc134017892)

[**2.3.3.** **Use case Tạo tài khoản** 26](#_Toc134017893)

[**2.3.4.** **Use case Tăng lượt xem** 27](#_Toc134017894)

[**2.3.5.** **Use case Tăng lượt theo dõi** 28](#_Toc134017895)

[**2.4.** **Thiết kế giao diện** 29](#_Toc134017896)

[**2.4.1.** **Giao diện quản lý thiết bị** 29](#_Toc134017897)

[**2.4.2.** **Giao diện quản lý tài khoản** 30](#_Toc134017898)

[**2.4.3.** **Giao diện Tạo tài khoản** 31](#_Toc134017899)

[**2.4.4.** **Giao diện Tăng lượt xem** 32](#_Toc134017900)

[**2.4.5.** **Giao diện Tăng lượt đăng ký** 33](#_Toc134017901)

[**2.5.** **Thiết kế cơ sở dữ liệu** 34](#_Toc134017902)

[**CHƯƠNG 3. XÂY DỰNG VÀ PHÁT TRIỂN ỨNG DỤNG** 35](#_Toc134017903)

[**3.1.** **Giao diện thực hiện** 35](#_Toc134017904)

[**3.2.** **Các mã nguồn chính được thực hiện** 38](#_Toc134017905)

[**3.2.1.** **Thực hiện công việc trên các thiết bị** 38](#_Toc134017906)

[**3.2.2.** **Quản lý công việc cho các thiết bị** 39](#_Toc134017907)

[**3.2.3.** **Thêm công việc vào luồng** 40](#_Toc134017908)

[**3.2.4.** **Tìm kiếm theo thuộc tính trong file xml** 41](#_Toc134017909)

[**3.2.5.** **Lớp thực thi các lệnh ADB** 42](#_Toc134017910)

[**3.2.6.** **Lớp Base Script – Logic của một kịch bản** 44](#_Toc134017911)

[**3.2.7.** **Kịch bản Tạo tài khoản** 47](#_Toc134017912)

[**CHƯƠNG 4. KẾT LUẬN** 49](#_Toc134017913)

[**4.1.** **Kết quả đạt được** 49](#_Toc134017914)

[**4.2.** **Những khó khăn khi thực hiện** 50](#_Toc134017915)

[**4.3.** **Hướng phát triển trong tương lai** 51](#_Toc134017916)

[TÀI LIỆU THAM KHẢO 52](#_Toc134017917)

# 

# Danh mục các ký hiệu, các chữ viết tắt

WPF - Windows Presentation Foundation

MVVM – Model View ViewModel

CSDL – Cơ sở dữ liệu

.NET - Network

# Danh mục các bảng

# Danh mục hình vẽ và đồ thị

Hinh 2.1. Biều đồ Use case tổng quan

Hình 2.2. Biều đồ use case Tạo tài khoản Google

Hình 2.3. Biều đồ use case Tạo tài khoản Facebook

Hình 2.4. Biều đồ use case Tăng lượt xem

Hình 2.5. Biều đồ use case Tăng lượt đăng ký

Hình 2.6. Biều đồ use case Tăng lượt theo dõi

Hình 2.7. Biểu đồ trình tự cho use case Quản lý tài khoản

Hình 2.8. Biểu đồ trình tự cho use case Quản lý thiết bị

Hình 2.9. Biểu đồ trình tự cho use case Tạo tài khoản

Hình 2.10. Biểu đồ trình tự cho use case Tăng lượt xem

Hình 2.11. Biểu đồ trình tự cho use case Tăng lượt theo dõi

Hình 2.12. Giao diện Quản lý thiết bị

Hình 2.13. Giao diện Quản lý tài khoản

Hình 2.14. Giao diện Tạo tài khoản

Hình 2.15. Giao diện Tăng lượt xem

Hình 2.16. Giao diện Tăng lượt đăng ký

Hình 2.17. Thiết kế cơ sở dữ liệu lưu thông tin tài khoản

Hình 3.1. Giao diện quản lý thiết bị

Hình 3.2. Giao diện Quản lý tài khoản

Hình 3.3. Giao diện Tạo tài khoản

Hình 3.4. Giao diện tăng lượt xem

Hình 3.5. Giao diện Tăng lượt đăng ký

Hình 3.6. Giao diện chọn thiết bị

# **LỜI NÓI ĐẦU**

Trong thời đại công nghệ số hiện nay, mạng xã hội đã thay thế những nền tảng truyền thông cũ và trở nên phổ biến trên thế giới. Với lượng người dùng đông đảo của mạng xã hội thì đây là một cơ hội để có thể quảng bá sản phẩm với hàng tỷ người dùng. Tuy nhiên việc xây dựng một nơi như vậy trên những nền tảng này với những người mới bắt đầu còn gặp rất nhiều khó khăn.

Đặc biệt, việc tăng lượt xem, lượt đăng ký trên kênh Youtube hay lượt theo dõi trên trang Facebook sẽ giúp người dùng quảng bá và phát triển nhanh chóng trên thị trường, giúp cho các nội dung trở nên phổ biến hơn và thu hút được đông đảo khán giả.

Ngoài ra, việc tăng lượt xem, đăng ký kênh Youtube bằng điện thoại di động còn giúp cho các nội dung video được quảng bá và phát triển một cách bền vững, đồng thời cũng giúp cho các kênh Youtube có thể tạo ra thu nhập ổn định từ hoạt động kinh doanh trên Youtube.

Xuất phát từ mặt lý luận và thực tiễn đặ ra cho vấn đề trên, em đã chọn đề tài “**Xây dựng ứng dụng điều khiển trang trại điện thoại Android**” để giải quyết vấn đề trên. Để hoàn thành được ứng dụng này em xin cảm ơn thầy TS. Đặng Trọng Hợp và các anh chị bên công ty Samsung R&D Center đã giúp đỡ em để em có thể phát triển ứng dụng này.

Nội dung của báo cáo gồm 4 phần

**Phần Một:** Tổng quan về cơ sở lý thuyết và các nền tảng được sử dụng

**Phần Hai:** Phân tích, thiết kế hệ thống

**Phần Ba:** Xây dựng và phát triển ứng dụng

**Phần Bốn:** Kết quả đạt được

1. **TỔNG QUAN**
   1. **Hệ điều hành Android và công cụ Android Studio**

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Ban đầu, Android được phát triển bởi Android Inc. với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005.

Đối với những nhà phát triển, Android là một nền tảng hấp dẫn để phát triển ứng dụng di động do tính linh hoạt và tính mở của nó. Mã nguồn mở của Android cho phép nhà phát triển có thể tùy chỉnh và phát triển các ứng dụng trên nền tảng này một cách linh hoạt và dễ dàng.

Android cung cấp một bộ công cụ phát triển mạnh mẽ và đa dạng để giúp nhà phát triển tạo ra các ứng dụng chất lượng cao. Android Studio là một trình biên dịch tích hợp mạnh mẽ, cho phép nhà phát triển thiết kế, phát triển và kiểm thử ứng dụng trên nền tảng Android.

Ngoài ra, Android cũng hỗ trợ các ngôn ngữ lập trình phổ biến như Java, Kotlin và C++, giúp nhà phát triển có thể phát triển ứng dụng trên nền tảng này một cách linh hoạt và tiện lợi.

Một số tính năng đáng chú ý của Android Studio bao gồm tích hợp sẵn với Android Emulator, cho phép nhà phát triển kiểm thử ứng dụng trên một môi trường giả lập một cách dễ dàng và nhanh chóng.

Hơn nữa, cộng đồng phát triển Android cũng đang ngày càng phát triển với hàng triệu nhà phát triển đang hoạt động trên toàn cầu. Cộng đồng này cung cấp cho nhà phát triển nhiều tài nguyên, công cụ và hỗ trợ giúp họ phát triển các ứng dụng chất lượng cao trên nền tảng Android.

* 1. **Công nghệ Windows Presentation Foundation**

WPF - Windows Presentation Foundation, là một công nghệ của Microsoft được sử dụng để xây dựng ứng dụng máy tính trên nền tảng Windows. WPF được giới thiệu lần đầu tiên vào năm 2006, là sự kế thừa của công nghệ Windows Forms, và cho phép các nhà phát triển tạo ra các ứng dụng máy tính hiện đại và tương tác với người dùng một cách đẹp mắt và dễ dàng hơn.

WPF được xây dựng nhằm mục đích:

* Cung cấp được một nền tảng thống nhất để xây dựng giao diện của người dùng.
* Giúp người lập trình và người thiết kế giao diện làm việc cùng nhau một cách dễ dàng.
* Cung cấp một công nghệ chung để xây dựng giao diện người dùng trên cả Windows và trình duyệt Web.

Các đặc điểm và tình năng của WPF:

* XAML: WPF sử dụng ngôn ngữ XAML (Extensible Application Markup Language) để mô tả giao diện người dùng. XAML là một ngôn ngữ khai báo, tương tự như HTML, cho phép các nhà phát triển mô tả giao diện người dùng của ứng dụng một cách dễ dàng hơn.
* Đồ họa vector: WPF hỗ trợ đồ họa vector, cho phép các nhà phát triển tạo ra các đối tượng đồ họa (hình ảnh, biểu đồ, đường cong) dễ dàng và có tính tương thích cao trên nhiều thiết bị.
* Animation: WPF hỗ trợ animation, cho phép các nhà phát triển tạo ra các hiệu ứng động và chuyển động trên giao diện người dùng của ứng dụng.
* Styles: WPF cung cấp tính năng styles, cho phép các nhà phát triển tạo ra các kiểu giao diện người dùng có thể được áp dụng cho nhiều đối tượng khác nhau trong ứng dụng.
* Data Binding: WPF hỗ trợ data binding, cho phép các nhà phát triển liên kết dữ liệu từ các đối tượng khác nhau trong ứng dụng, giúp tạo ra các ứng dụng có tính tương tác cao và dễ bảo trì.
* Trình điều khiển: WPF cung cấp một loạt các trình điều khiển (control) như Button, Textbox, ComboBox, ListView, và TreeView, cho phép các nhà phát triển tạo ra các giao diện người dùng phức tạp và tương tác.
* MVVM: WPF hỗ trợ mô hình MVVM (Model-View-ViewModel), giúp tách rõ logic ứng dụng, giao diện người dùng và dữ liệu.
* Đa nền tảng: WPF cho phép phát triển ứng dụng trên nhiều nền tảng khác nhau, bao gồm Windows, Web, và các thiết bị di động.
* Đa ngôn ngữ: WPF hỗ trợ đa ngôn ngữ, cho phép các nhà phát triển tạo ra các ứng dụng có khả năng hiển thị giao diện người dùng trên nhiều ngôn ngữ khác nhau.
* Các công cụ hỗ trợ: WPF tích hợp với các công cụ phát triển của Microsoft như Visual Studio.
* Cấu trúc phân tầng: WPF hỗ trợ cấu trúc phân tầng, cho phép các nhà phát triển tách rõ logic ứng dụng, giao diện người dùng và dữ liệu.
* Điều khiển màu sắc: WPF cho phép các nhà phát triển điều khiển màu sắc của giao diện người dùng một cách dễ dàng, giúp tạo ra các ứng dụng có thiết kế đẹp và chuyên nghiệp.
* Hỗ trợ đa phương tiện: WPF hỗ trợ đa phương tiện, cho phép các nhà phát triển tạo ra các ứng dụng có khả năng hiển thị video, âm thanh và các đối tượng đa phương tiện khác.
* Tích hợp trình xử lý dữ liệu: WPF tích hợp trình xử lý dữ liệu (data processing engine), cho phép các nhà phát triển tạo ra các ứng dụng có khả năng xử lý dữ liệu phức tạp và hiển thị kết quả một cách dễ dàng.
* Hỗ trợ đồ họa 3D: WPF hỗ trợ đồ họa 3D, cho phép các nhà phát triển tạo ra các ứng dụng có khả năng hiển thị các đối tượng 3D và các hiệu ứng đặc biệt khác.

Một trong những đặc điểm nổi bật của WPF là sự phân tách giữa giao diện người dùng và logic ứng dụng. Các nhà phát triển có thể tách riêng giao diện người dùng và mã ứng dụng, giúp cho việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn. Ngoài ra, WPF còn hỗ trợ việc phát triển ứng dụng theo mô hình MVVM (Model-View-ViewModel), giúp tách rõ logic ứng dụng, giao diện người dùng và dữ liệu.

Bên cạnh đó, WPF còn kết hợp tốt với các công nghệ khác của Microsoft như C#, .NET Framework và Visual Studio, giúp cho việc phát triển ứng dụng trên nền tảng này trở nên thuận tiện và dễ dàng hơn.

* 1. **Mô hình MVVM**

Mô hình MVVM (Model-View-ViewModel) là một mô hình kiến trúc phần mềm được sử dụng trong phát triển ứng dụng desktop và mobile, đặc biệt là trong các ứng dụng sử dụng nền tảng WPF và Xamarin. Mô hình này giúp tách rõ logic ứng dụng, giao diện người dùng và dữ liệu, giúp tăng tính mở rộng và dễ bảo trì của ứng dụng.

Các thành phần trong mô hình MVVM bao gồm:

* Model: Là đại diện cho dữ liệu của ứng dụng, được định nghĩa trong lớp Model.
* View: Là phần giao diện người dùng, hiển thị dữ liệu và tương tác với người dùng, được định nghĩa trong lớp View.
* ViewModel: Là lớp trung gian giữa Model và View, thực hiện xử lý logic ứng dụng, lấy dữ liệu từ Model và cung cấp cho View để hiển thị. ViewModel cũng nhận lệnh từ View để thực hiện các thao tác trên dữ liệu, sau đó cập nhật lại Model.

Tính năng của mô hình MVVM:

* Tách rõ logic ứng dụng, giao diện người dùng và dữ liệu: MVVM giúp tách rõ các thành phần của ứng dụng, giúp dễ dàng quản lý và bảo trì ứng dụng.
* Tính mở rộng: MVVM cho phép phát triển ứng dụng một cách dễ dàng và linh hoạt.
* Tính tái sử dụng: MVVM giúp tạo ra các thành phần có thể tái sử dụng được trong nhiều ứng dụng khác nhau.
* Tính độc lập giữa UI và logic ứng dụng: MVVM giúp tách rõ UI và logic ứng dụng, giúp việc thiết kế giao diện người dùng trở nên dễ dàng hơn.
* Dễ dàng kiểm thử: MVVM giúp việc kiểm thử ứng dụng trở nên dễ dàng hơn, vì các thành phần của ứng dụng được tách rõ ràng.
* Tính linh hoạt: MVVM có thể thay đổi giao diện người dùng một cách dễ dàng mà không ảnh hưởng đến logic ứng dụng

Mô hình MVVM giúp tách rõ logic ứng dụng, giao diện người dùng và dữ liệu, giúp tăng tính mở rộng và dễ bảo trì của ứng dụng. Nó cũng giúp giảm sự phụ thuộc giữa các thành phần của ứng dụng, giúp phát triển ứng dụng dễ dàng hơn. Ngoài ra, MVVM còn giúp tách rõ các trách nhiệm của từng thành phần, giúp quản lý dự án hiệu quả hơn.

* 1. **ADB - Android Debug Bridge**

ADB (Android Debug Bridge) là một công cụ dòng lệnh trong Android SDK (Software Development Kit), cho phép các nhà phát triển kết nối và quản lý các thiết bị Android từ máy tính.

ADB được bao gồm trong gói Android SDK Platform-Tools, bạn có thể tài và cài đặt thông qua SDK Manager của Adndroid Studio bằng cách đánh dầu kiểm vào mục Android SDK Platform-Tools.

Các tính năng của ADB bao gồm:

Kết nối với thiết bị Android qua USB hoặc WiFi.

* Điều khiển thiết bị Android từ máy tính bằng lệnh dòng lệnh ADB, cho phép truy cập và quản lý các tệp tin, ứng dụng, dữ liệu và các thiết lập trên thiết bị.
* Thực hiện các hoạt động liên quan đến debugging, như lấy thông tin về lỗi, ghi log hoặc chạy các ứng dụng để xem các thông báo lỗi.
* Root thiết bị Android để có quyền truy cập đầy đủ vào hệ thống.
* Cài đặt ứng dụng và các file APK từ máy tính vào thiết bị Android.
* Lấy thông tin về thông số kỹ thuật của thiết bị, như số phiên bản Android, thông tin phần cứng, thông tin mạng,…

ADB có thể được sử dụng để tạo các công cụ tự động hoá, giúp cho việc kiểm thử và triển khai ứng dụng Android trở nên dễ dàng hơn.

Các cách sử dụng ADB để tạo công cụ tự động:

* Tạo script ADB: Sử dụng các lệnh ADB trong một tập tin script để tự động hóa các tác vụ như cài đặt ứng dụng, khởi động ứng dụng, tạo các tệp tin log, ....
* Sử dụng ADB với các ngôn ngữ lập trình: Sử dụng ADB với các ngôn ngữ lập trình để tạo các ứng dụng tự động.
* Sử dụng ADB với các công cụ tự động hóa bên thứ ba: Có nhiều công cụ tự động hóa bên thứ ba được phát triển sử dụng ADB như Appium, Robotium, … Những công cụ này cho phép kiểm thử ứng dụng Android trên nhiều thiết bị cùng một lúc và tạo ra các kết quả kiểm thử tự động.

Các lệnh ADB cơ bản:

* adb devices: Kiểm tra thiết bị được kết nối với máy tính
* adb install <path to apk>: Cài đặt một file APK lên thiết bị
* adb uninstall <package name>: Gỡ bỏ một ứng dụng trên thiết bị
* adb shell: Truy cập vào giao diện dòng lệnh trên thiết bị
* adb push <local> <remote>: Đẩy tệp từ máy tính vào thiết bị
* adb pull <remote> <local>: Kéo tệp từ thiết bị về máy tính
* adb logcat: Hiển thị các thông báo log của thiết bị
* adb reboot: Khởi động lại thiết bị
* adb forward <local> <remote>: Chuyển tiếp cổng giữa máy tính và thiết bị
* adb shell dumpsys: Hiển thị thông tin hệ thống của thiết bị

Một số lệnh được sử dụng trong quá trình phát triển ứng dụng:

* adb shell am start -a android.settings.SETTINGS: mở chức năng cài đặt trên điện thoại
* Adb shell uiautomator dump /sdcard/view.xml: xuất màn hình điện thoại ra file xml lưu trên bộ nhớ của điện thoại
* Adb pull /sdcard/view.xml D:\AndroidPhoneFarm: lấy file view.xml từ điện thoại ra máy tính
* adb shell input tap <x y>: Click vào vị trí có tọa độ x, y
* adb shell input <text>: Chèn dữ liệu vào vùng được chọn
* adb shell input keyevent <event\_code>: bắt sự kiện của các phím trên máy tính
* adb shell am start -a android.intent.action.VIEW <url>: mở một url trên thiết bị
  1. **Các hoạt động trên nền tảng Google**

Trên các thiết bị chạy hệ điều hành Android, việc sử dụng các dịch vụ của Google là một điều rất phổ biến. Đặc biệt, với một tập đoàn lớn như Google, việc tự động hóa các chức năng trở nên tương đối khó khăn.

* + 1. **Tạo tài khoản**

Các bước tạo tài khoản Google trên điện thoại:

* Mở ứng dụng Cài đặt trên điện thoại của bạn.
* Chọn Tài khoản.
* Chọn Thêm tài khoản.
* Chọn Google.
* Nhập địa chỉ email của bạn và chọn Tiếp theo.
* Nhập mật khẩu của bạn và chọn Tiếp theo.
* Chọn Đồng ý để đồng ý với các điều khoản và điều kiện của Google.
* Nếu được yêu cầu, bạn có thể thêm thông tin cá nhân của mình, như tên và số điện thoại.
* Sau khi hoàn tất, tài khoản Google của bạn sẽ được tạo.

Việc tạo tài khoản tự động có thể gặp nhiều vấn đề do các thuật toán đặc biệt của Google nhằm ngăn chặn việc tạo tự động.

* + 1. **Xem video trên Youtube**

Các bước xem video trên Youtube:

* Mở ứng dụng YouTube trên điện thoại của bạn.
* Tìm kiếm video
* Bấm vào video để xem.

Các video thường chèn thêm quảng cáo và thuật toán hiển thị lượt xem của Google chưa real-time dẫn đến việc khó kiểm soát mình xem có thành công hay chưa

* + 1. **Theo dõi kênh Youtube**

Các bước theo dõi kênh Youtube:

* Mở ứng dụng YouTube trên điện thoại của bạn và tìm kiếm kênh mà bạn muốn theo dõi.
* Khi đã tìm thấy kênh, nhấn vào tên của kênh để truy cập vào trang kênh.
* Nhấn vào nút "Đăng ký" trên trang kênh để theo dõi kênh đó.

Vì một tài khoản Google chỉ theo dõi kênh một lần khiến việc tăng lượt theo dõi yêu cầu một lượng lớn tài khoản Google.

* 1. **Các hoạt động trên nền tảng Facebook**
     1. **Tạo tài khoản Facebook**

Các bước tạo tài khoản Facebook trên điện thoại:

* Tải ứng dụng Facebook trên điện thoại của bạn từ cửa hàng ứng dụng.
* Mở ứng dụng Facebook và chọn "Tạo tài khoản mới".
* Nhập tên đầy đủ của bạn, số điện thoại hoặc địa chỉ email, mật khẩu, ngày sinh và giới tính của bạn. Sau đó, nhấn "Tiếp tục".
* Facebook sẽ gửi một mã xác nhận đến số điện thoại hoặc địa chỉ email của bạn. Nhập mã xác nhận đó vào ứng dụng Facebook để hoàn tất đăng ký.

Tạo tài khoản Facebook yêu cầu nhập mã xác thực gây khó khăn cho việc tạo.

* + 1. **Theo dõi trang cá nhân**

Các bước theo dõi trang cá nhân:

* Tìm kiếm tên người dùng hoặc tên trang cá nhân trong thanh tìm kiếm.
* Nhấp vào tên người dùng hoặc trang cá nhân trong kết quả tìm kiếm để truy cập vào trang cá nhân đó.
* Nhấn nút "Theo dõi" để bắt đầu theo dõi trang cá nhân của người dùng đó.

Vì một tài khoản Facebook chỉ theo dõi trang cá nhân một lần khiến việc tăng lượt theo dõi yêu cầu một lượng lớn tài khoản Facebook.

* 1. **Visual Studio 2022**

Visual Studio 2022 Community là một IDE (Integrated Development Environment) phát triển ứng dụng miễn phí của Microsoft. Được thiết kế để hỗ trợ các nhà phát triển xây dựng ứng dụng cho các nền tảng khác nhau như Windows, Android, iOS và web.

Visual Studio 2022 Community có nhiều tính năng hữu ích như:

Cung cấp các công cụ cho phát triển ứng dụng desktop, web và di động.

* Hỗ trợ nhiều ngôn ngữ lập trình, bao gồm C#, C++, Visual Basic, Python, JavaScript và nhiều ngôn ngữ khác.
* Cung cấp nhiều tính năng hỗ trợ debugging, testing và deployment cho các ứng dụng.
* Có khả năng tích hợp với các công cụ khác như Git, Docker, và Azure.
* Có khả năng tùy chỉnh giao diện sử dụng và cài đặt theo nhu cầu của người dùng.

Visual Studio 2022 hỗ trợ cài đặt các thư viện mã nguồn mở dưới dạng gói NuGet. Trong đó, NuGet Package Manager là một công cụ quản lý gói cho phép người dùng tìm kiếm, cài đặt và cập nhật các gói NuGet trong dự án của họ.

Đặc biệt, Visual Studio 2022 còn có tính năng Server Explorer. Đây là một tính năng quan trọng trong Visual Studio cho phép bạn quản lý các kết nối và đối tượng trên các server khác nhau. Nó cung cấp giao diện đồ họa dễ sử dụng để quản lý các thành phần cơ sở dữ liệu, máy chủ và dịch vụ web.

Trong Visual Studio, ta có thể tạo file cơ sở dữ liệu SQL Server bằng định dạng .mdf bằng cách sử dụng SQL Server Object Explorer. Nó chứa các đối tượng cơ sở dữ liệu như bảng, quan hệ, khóa ngoại và chứa các dữ liệu thực tế của cơ sở dữ liệu.

Khi kết hợp tính năng này với Entity Framework sẽ giúp ta dễ dàng quản lý dữ liệu, lập trình viên có thể tạo ra các ứng dụng có khả năng thích ứng với thay đổi của cơ sở dữ liệu một cách dễ dàng hơn và nhanh chóng hơn.

1. **PHÂN TÍCH, THIẾT KẾ HỆ THỐNG**
   1. **Biểu đồ use case**
      1. **Biều đồ use case tổng quan**

****

Hinh 2.1. Biều đồ Use case tổng quan

* + 1. **Biểu đồ phân rã use case** 
       1. **Use case Tạo tài khoản Google**



Hình 2.2. Biều đồ use case Tạo tài khoản Google

* + - 1. **Use case Tạo tài khoản Facebook**

****

Hình 2.3. Biều đồ use case Tạo tài khoản Facebook

* + - 1. **Use case Tăng lượt xem**

Hình 2.4. Biều đồ use case Tăng lượt xem

* + - 1. **Use case Tăng lượt đăng ký**

Hình 2.5. Biều đồ use case Tăng lượt đăng ký

* + - 1. **Use case Tăng lượt theo dõi**

Hình 2.6. Biều đồ use case Tăng lượt theo dõi

* 1. **Mô tả chi tiết các use case**
     1. **Use case Thông tin hệ thống**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng xem thông tin hệ thống

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Thông tin” trên thanh Menu. Hệ thống sẽ hiển thị ra giao diện quản lý hệ thống.
  2. Người dùng có thể xem các thông tin giới thiệu của hệ thống
* Luồng rẽ nhánh

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Quản lý thiết bị**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng xem thông tin các thiết bị đã kết nối với hệ thống

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Quản lý thiết bị” trên thanh Menu. Hệ thống sẽ hiển thị ra giao diện quản lý thiết bị gồm danh sách các thiết bị và một nút làm mới.
  2. Người dùng click vào nút làm mới thì hệ thống sẽ quét và hiển thị lại số lượng thiết bị đã kết nối
* Luồng rẽ nhánh

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Quản lý tài khoản Facebook**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng quản lý các tài khoản Facebook

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Quản lý tài khoản” trên thanh Menu. Hệ thống sẽ hiển thị ra một menu con.
  2. Người dùng click vào nút “Facebook”. Hệ thống hiển thị giao diện Quản lý tài khoản Facebook gồm các chức năng làm mới, xóa tài khoản lỗi.
  3. Khi người dùng click vào nút làm mới, hệ thống sẽ lấy thông tin từ cơ sở dữ liệu và hiển thị lại danh sách tài khoản.
  4. Khi người dùng click vào nút xóa tài khoản lỗi, hệ thống sẽ truy xuất các tài khoản lỗi trong cơ sở dữ liệu và xóa tài khoản đó.
* Luồng rẽ nhánh

Tại bất kỳ thời điểm nào trong quá trình thực hiện usecase nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiện thị thông báo lỗi và usecase kết thúc.

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Quản lý tài khoản Google**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng quản lý các tài khoản Google

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Quản lý tài khoản” trên thanh Menu. Hệ thống sẽ hiển thị ra một menu con.
  2. Người dùng click vào nút “Google”. Hệ thống hiển thị giao diện Quản lý tài khoản Google gồm các chức năng làm mới, xóa tài khoản lỗi.
  3. Khi người dùng click vào nút làm mới, hệ thống sẽ lấy thông tin từ cơ sở dữ liệu và hiển thị lại danh sách tài khoản.
  4. Khi người dùng click vào nút xóa tài khoản lỗi, hệ thống sẽ truy xuất các tài khoản lỗi trong cơ sở dữ liệu và xóa tài khoản đó.
* Luồng rẽ nhánh

Tại bất kỳ thời điểm nào trong quá trình thực hiện usecase nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiện thị thông báo lỗi và usecase kết thúc.

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Tạo tài khoản Google**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng tự động tạo tài khoản Google

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Google” trên thanh Menu. Hệ thống sẽ hiển thị ra một menu con.
  2. Người dùng click vào nút “Tạo tài khoản”. Hệ thống hiển thị giao diện Tạo tài khoản Google gồm các chức năng tự động tạo và dừng tạo
  3. Khi người dùng click vào nút thực thi, hệ thống sẽ hiển thị ra một màn hình chọn thiết bị. Người dùng chọn thiết bị cần sử dụng và click vào nút thực thi. Hệ thống sẽ tự động tạo tài khoản trên thiết bị đã chọn.
  4. Khi người dùng click vào nút dừng, hệ thống sẽ dừng tạo tài khoản đối với các thiết bị đang chạy tạo tài khoản.
* Luồng rẽ nhánh

Tại bước 3 trong luồng cơ bản, nếu người dùng không chọn thiết bị hay ấn vào nút hủy bỏ thì hệ thống sẽ hiển thị lại giao diện tạo tài khoản.

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Tăng lượt xem**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng tự động tăng lượt xem video Youtube

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Youtube” trên thanh Menu. Hệ thống sẽ hiển thị ra một menu con.
  2. Người dùng click vào nút “Tăng lượt xem”. Hệ thống hiển thị giao diện Tăng lượt xem gồm các trường Đường dẫn, Số lượt cần tăng và thời gian xem và các chức năng tự động tăng, lấy thông tin video và dừng tăng.
  3. Khi người dùng nhập đúng thông tin và click vào nút thực thi, hệ thống sẽ hiển thị ra một màn hình chọn thiết bị. Người dùng chọn thiết bị cần sử dụng và click vào nút thực thi. Hệ thống sẽ tự động xem video theo số lần đã nhập.
  4. Khi người dùng click vào nút dừng, hệ thống sẽ dừng xem video đối với các thiết bị đang chạy tăng lượt xem.
  5. Khi người dùng nhập đúng đường dẫn và click vào nút lấy thông tin, hệ thống sẽ lấy thông tin video gồm tiêu đề, thời lượng và số lượt xem của video đó và hiện thị trên màn hình.
* Luồng rẽ nhánh
  1. Tại bước 3 trong luồng cơ bản, nếu người dùng không chọn thiết bị hay ấn vào nút hủy bỏ thì hệ thống sẽ hiển thị lại giao diện tạo tài khoản.
  2. Tại bước 3,5 trong luồng cơ bản, nếu người dùng nhập sai thông tin thì sẽ hiển thị thông báo lỗi.

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Tăng lượt đăng ký**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng tự động tăng lượt đăng ký kênh Youtube

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Youtube” trên thanh Menu. Hệ thống sẽ hiển thị ra một menu con.
  2. Người dùng click vào nút “Tăng lượt đăng ký”. Hệ thống hiển thị giao diện Tăng lượt đăng ký gồm các trường Đường dẫn, Số lượt cần tăng và các chức năng tự động tăng, lấy thông tin kênh và dừng tăng.
  3. Khi người dùng nhập đúng thông tin và click vào nút thực thi, hệ thống sẽ hiển thị ra một màn hình chọn thiết bị. Người dùng chọn thiết bị cần sử dụng và click vào nút thực thi. Hệ thống sẽ tự động đăng ký kênh theo số lần đã nhập.
  4. Khi người dùng click vào nút dừng, hệ thống sẽ dừng đăng ký đối với các thiết bị đang chạy tăng lượt đăng ký.
  5. Khi người dùng nhập đúng đường dẫn và click vào nút lấy thông tin, hệ thống sẽ lấy thông tin video gồm tên kênh và số video của video đó và hiện thị trên màn hình.
* Luồng rẽ nhánh
  1. Tại bước 3 trong luồng cơ bản, nếu người dùng không chọn thiết bị hay ấn vào nút hủy bỏ thì hệ thống sẽ hiển thị lại giao diện tạo tài khoản.
  2. Tại bước 3,5 trong luồng cơ bản, nếu người dùng nhập sai thông tin thì sẽ hiển thị thông báo lỗi.

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Tạo tài khoản Facebook**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng tự động tạo tài khoản Facebook

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Facebook” trên thanh Menu. Hệ thống sẽ hiển thị ra một menu con.
  2. Người dùng click vào nút “Tạo tài khoản”. Hệ thống hiển thị giao diện Tạo tài khoản Facebook gồm các chức năng tự động tạo và dừng tạo
  3. Khi người dùng click vào nút thực thi, hệ thống sẽ hiển thị ra một màn hình chọn thiết bị. Người dùng chọn thiết bị cần sử dụng và click vào nút thực thi. Hệ thống sẽ tự động tạo tài khoản trên thiết bị đã chọn.
  4. Khi người dùng click vào nút dừng, hệ thống sẽ dừng tạo tài khoản đối với các thiết bị đang chạy tạo tài khoản.
* Luồng rẽ nhánh

Tại bước 3 trong luồng cơ bản, nếu người dùng không chọn thiết bị hay ấn vào nút hủy bỏ thì hệ thống sẽ hiển thị lại giao diện tạo tài khoản.

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* + 1. **Use case Tăng lượt theo dõi**

**Tác nhân:** Người dùng

**Mô tả:** Use case này cho phép người dùng tự động tăng lượt xem video Youtube

**Luồng sự kiện**

* Luồng cơ bản:
  1. Use case này bắt đầu khi người dùng click vào “Youtube” trên thanh Menu. Hệ thống sẽ hiển thị ra một menu con.
  2. Người dùng click vào nút “Tăng lượt đăng ký”. Hệ thống hiển thị giao diện Tăng lượt đăng ký gồm các trường Đường dẫn, Số lượt cần tăng và các chức năng tự động tăng, lấy thông tin kênh và dừng tăng.
  3. Khi người dùng nhập đúng thông tin và click vào nút thực thi, hệ thống sẽ hiển thị ra một màn hình chọn thiết bị. Người dùng chọn thiết bị cần sử dụng và click vào nút thực thi. Hệ thống sẽ tự động đăng ký kênh theo số lần đã nhập.
  4. Khi người dùng click vào nút dừng, hệ thống sẽ dừng đăng ký đối với các thiết bị đang chạy tăng lượt đăng ký.
  5. Khi người dùng nhập đúng đường dẫn và click vào nút lấy thông tin, hệ thống sẽ lấy thông tin video gồm tên kênh và số video của video đó và hiện thị trên màn hình.
* Luồng rẽ nhánh
  1. Tại bước 3 trong luồng cơ bản, nếu người dùng không chọn thiết bị hay ấn vào nút hủy bỏ thì hệ thống sẽ hiển thị lại giao diện tạo tài khoản.
  2. Tại bước 3,5 trong luồng cơ bản, nếu người dùng nhập sai thông tin thì sẽ hiển thị thông báo lỗi.

**Các yêu cầu đặc biệt:** Không

**Tiền điều kiện:** Không

**Hậu điều kiện:** Không

**Điểm mở rộng:** Không

* 1. **Biểu đồ trình tự**
     1. **Use case Quản lý tài khoản**

Hình 2.7. Biểu đồ trình tự cho use case Quản lý tài khoản

* + 1. **Use case Quản lý thiêt bị**

Hình 2.8. Biểu đồ trình tự cho use case Quản lý thiết bị

* + 1. **Use case Tạo tài khoản**

Hình 2.9. Biểu đồ trình tự cho use case Tạo tài khoản

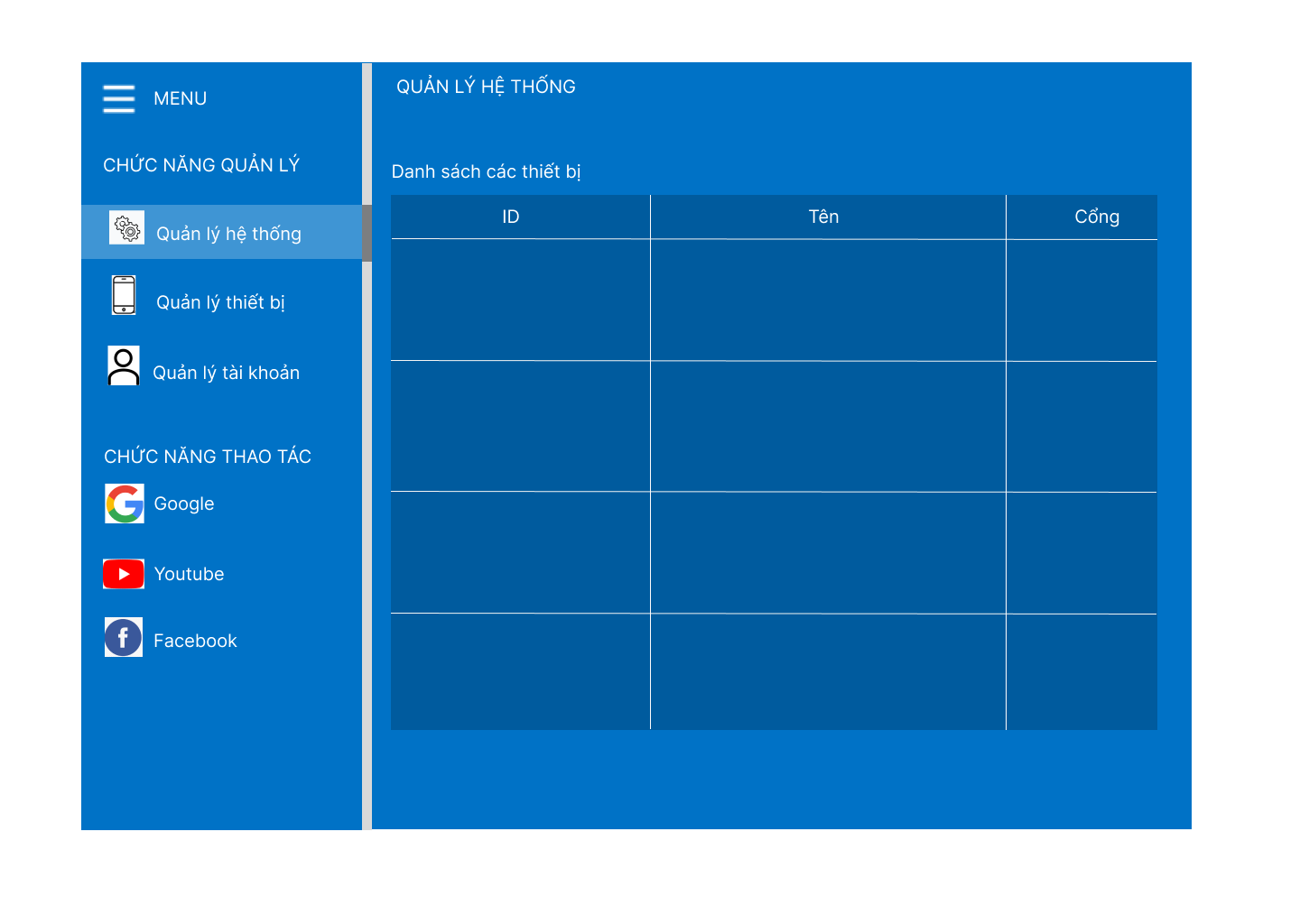
* + 1. **Use case Tăng lượt xem**

Hình 2.10. Biểu đồ trình tự cho use case Tăng lượt xem

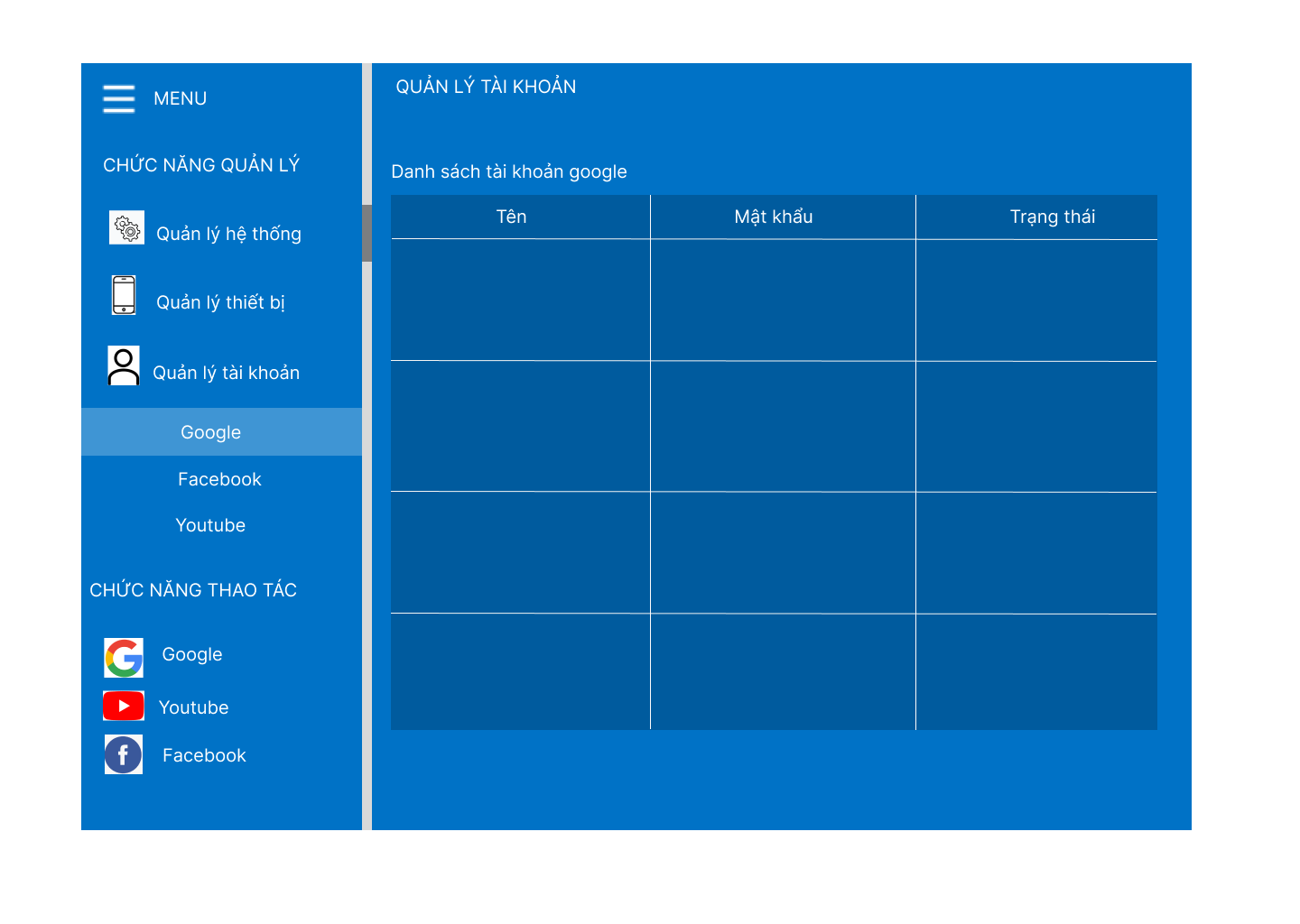
* + 1. **Use case Tăng lượt theo dõi**

Hình 2.11. Biểu đồ trình tự cho use case Tăng lượt theo dõi

* 1. **Thiết kế giao diện**
     1. **Giao diện quản lý thiết bị**

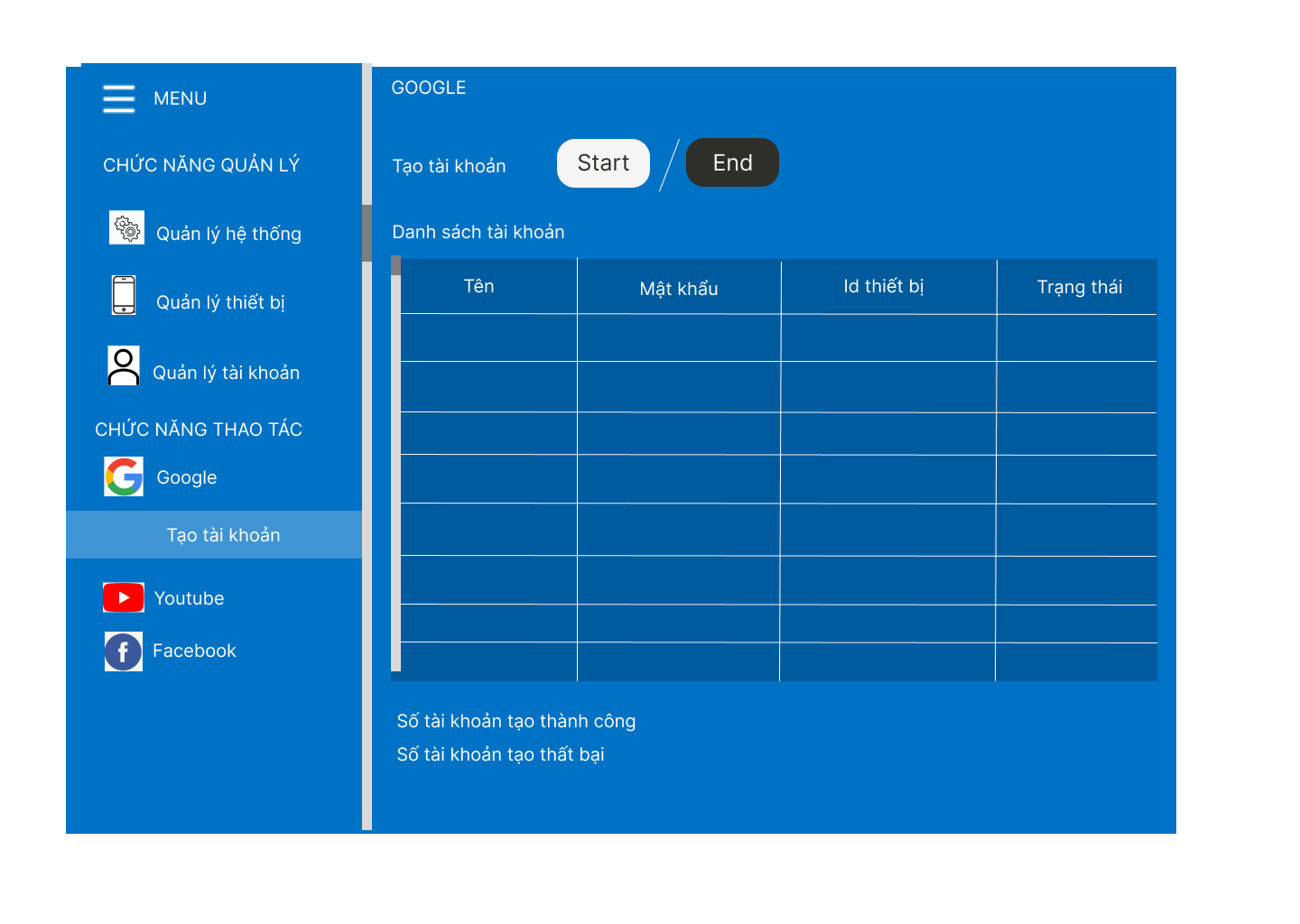
****Hình 2.12. Giao diện Quản lý thiết bị

* + 1. **Giao diện quản lý tài khoản**



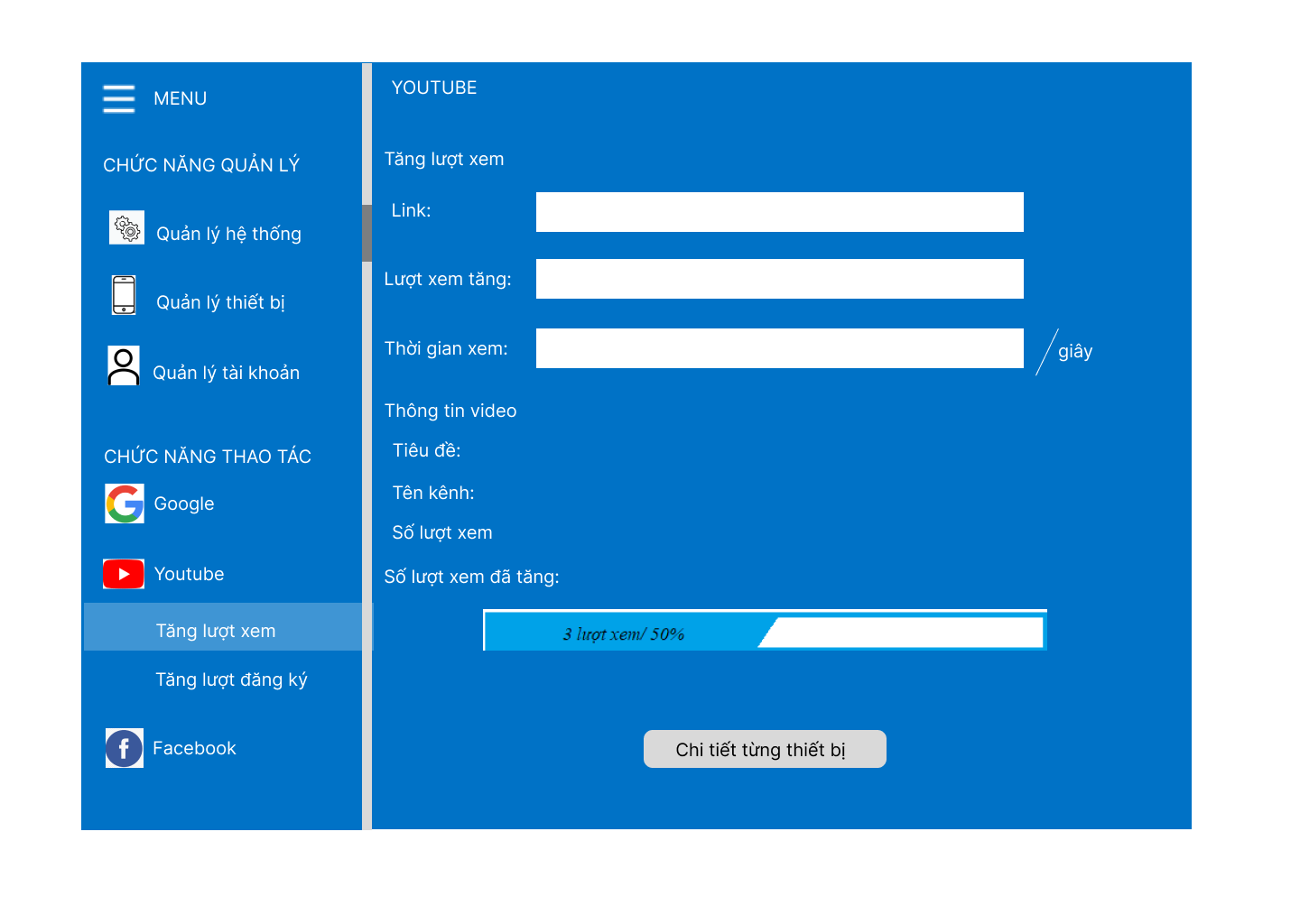
Hình 2.13. Giao diện Quản lý tài khoản

* + 1. **Giao diện Tạo tài khoản**



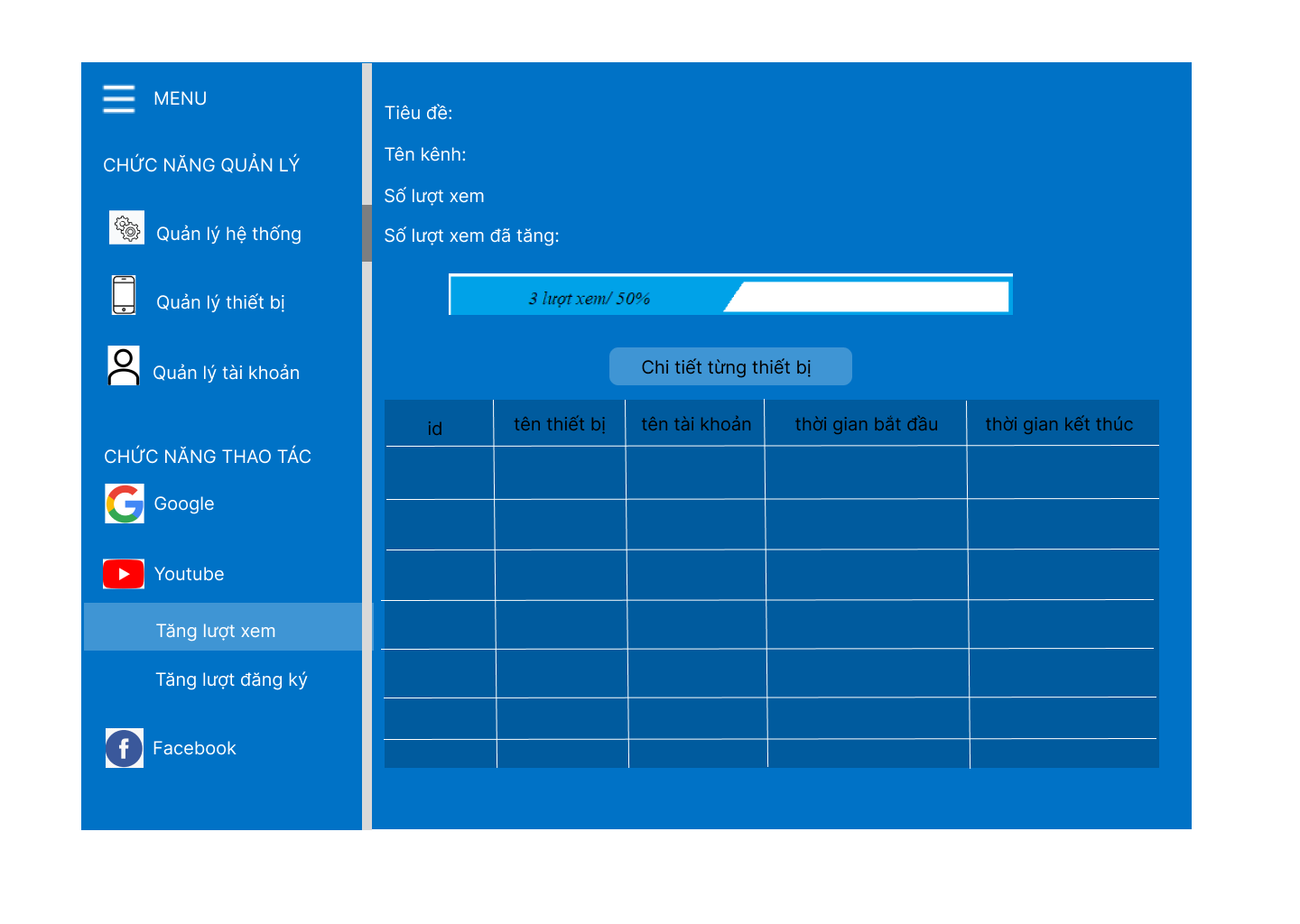
Hình 2.14. Giao diện Tạo tài khoản

* + 1. **Giao diện Tăng lượt xem**



Hình 2.15. Giao diện Tăng lượt xem

* + 1. **Giao diện Tăng lượt đăng ký**



Hình 2.16. Giao diện Tăng lượt đăng ký

* 1. **Thiết kế cơ sở dữ liệu**

Graphical user interface, text, application

Description automatically generated

Hình 2.17. Thiết kế cơ sở dữ liệu lưu thông tin tài khoản

1. **XÂY DỰNG VÀ PHÁT TRIỂN ỨNG DỤNG**
   1. **Giao diện thực hiện**

**A screenshot of a computer

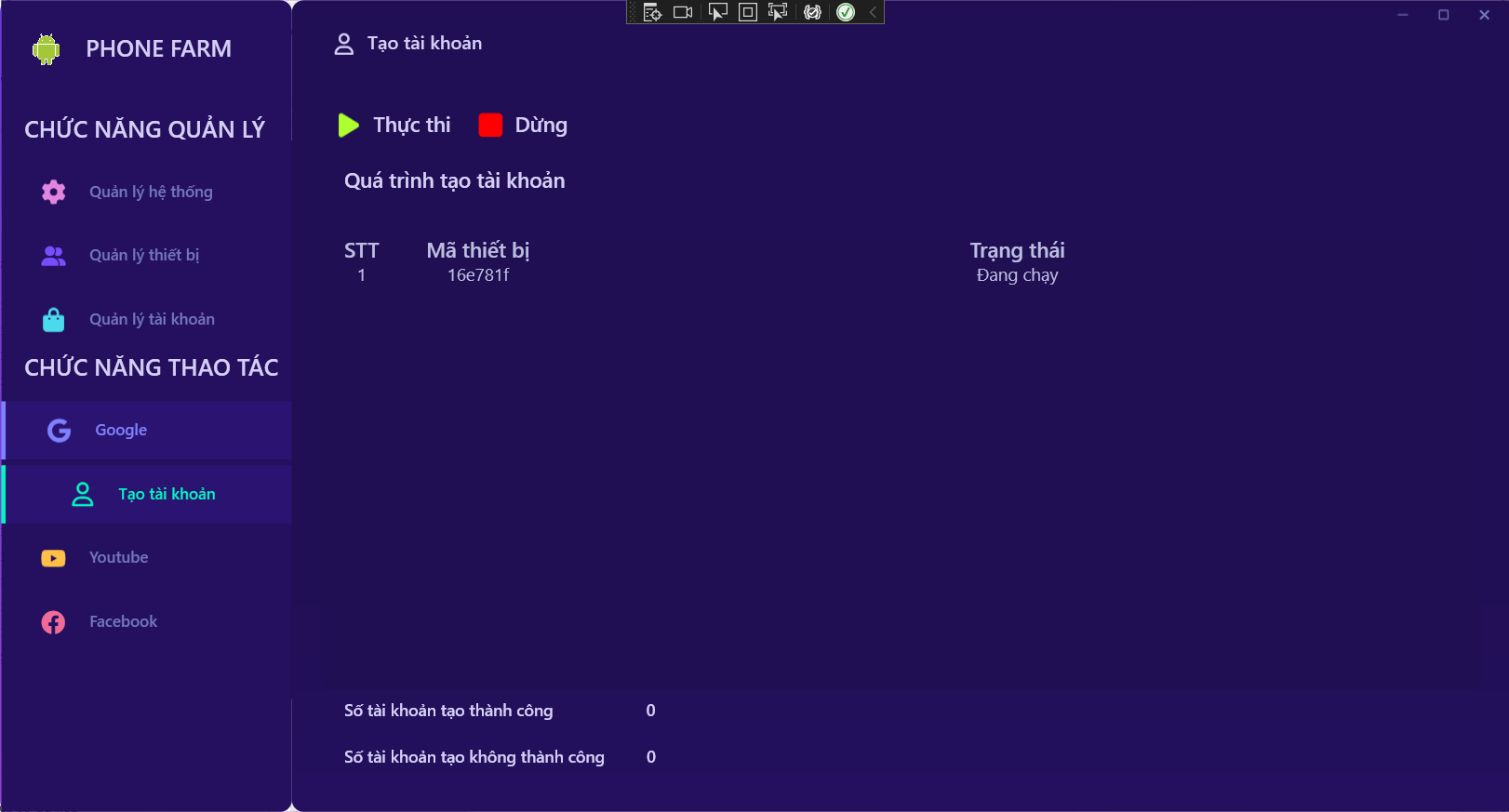
Description automatically generated**

Hình 3.1. Giao diện quản lý thiết bị

A screenshot of a computer

Description automatically generated

Hình 3.2. Giao diện Quản lý tài khoản



Hình 3.3. Giao diện Tạo tài khoản

A screenshot of a computer

Description automatically generated

Hình 3.4. Giao diện tăng lượt xem

A screenshot of a computer

Description automatically generated

Hình 3.5. Giao diện Tăng lượt đăng ký

A screenshot of a game

Description automatically generated with medium confidence

Hình 3.6. Giao diện chọn thiết bị

* 1. **Các mã nguồn chính được thực hiện**
     1. **Thực hiện công việc trên các thiết bị**

protected virtual void ThucHienCongViecTrenThietBi(string idThietBi)

{

themDongChoBangMutex.WaitOne();

var status = new DeviceStatus { Stt = devices.Count + 1, DeviceId = idThietBi, Status = DeviceStatus.TrangThai.DANG\_CHAY };

mainDispatcher.Invoke(() =>

{

devices.Add(status);

});

themDongChoBangMutex.ReleaseMutex();

var job = createScriptToRun(idThietBi);

try

{

if (job.RunScript())

{

status.Status = DeviceStatus.TrangThai.THANH\_CONG;

TangThanhCong();

}

else

{

status.Status = DeviceStatus.TrangThai.THAT\_BAI;

TangThatBai();

}

}

catch (Exception ex)

{

status.Status = DeviceStatus.TrangThai.THAT\_BAI;

TangThatBai();

}

}

* + 1. **Quản lý công việc cho các thiết bị**

protected virtual void QuanLyCongViecChoCacThietBi(List<string> thietbi, long soLanLap)

{

Thread[] jobThreads = new Thread[thietbi.Count];

try

{

int nextIndex = 0;

soLanLap = soLanLap == 0 ? LaySoLanLapCongViec() + \_thanhCong : soLanLap + \_thanhCong;

while (\_thanhCong < soLanLap)

{

if (nextIndex != -1)

{

jobThreads[nextIndex] = new Thread(new ThreadStart(() =>

{

ThucHienCongViecTrenThietBi(thietbi[nextIndex]);

}));

jobThreads[nextIndex].Start();

}

Thread.Sleep(2000);

nextIndex = -1;

for (int i = 0; i < jobThreads.Length; i++)

{

if (jobThreads[i] == null || !jobThreads[i].IsAlive)

{

nextIndex = i;

break;

}

}

}

foreach (var tb in this.ThietBiDuocDung)

{

thietBi.danhSachThietBi[tb].trangThai = false;

}

ThietBiDuocDung = new List<string>();

}

catch (Exception)

{

foreach (var thread in jobThreads)

{

thread?.Abort();

}

}

}

* + 1. **Thêm công việc vào luồng**

protected virtual void ThemCongViec(List<string> thietbi)

{

foreach (var tb in thietbi)

{

this.thietBi.danhSachThietBi[tb].trangThai = true;

this.ThietBiDuocDung.Add(tb);

}

var thread = new Thread(new ThreadStart(() =>

{

QuanLyCongViecChoCacThietBi(thietbi,0);

}));

thread.Start();

this.threads.Add(thread);

}

* + 1. **Tìm kiếm theo thuộc tính trong file xml**

public static List<XmlNode> findNode(string str, Matcher matcher)

{

var result = new List<XmlNode>();

try

{

var doc = new XmlDocument();

doc.LoadXml(str);

var root = doc.DocumentElement.FirstChild;

var stack = new Stack<XmlNode>();

stack.Push(root);

while (stack.Count != 0)

{

var e = stack.Pop();

if (matcher.Invoke(e))

{

result.Add(e);

}

foreach (XmlNode node in e.ChildNodes)

{

stack.Push((XmlNode)node);

}

}

}

catch(Exception ex)

{

result.Clear();

}

return result;

}

* + 1. **Lớp thực thi các lệnh ADB**

public class ADBUtils

{

public readonly static string path = "adb.exe";

private readonly string deviceId;

private readonly string trangThai;

public ADBUtils(string deviceId)

{

this.deviceId = deviceId;

}

public static string RunAdbCommand(string cmd, bool needOutput = false)

{

ProcessStartInfo startInfo = new ProcessStartInfo();

startInfo.CreateNoWindow = true;

startInfo.RedirectStandardOutput = needOutput;

startInfo.FileName = ADBUtils.path;

startInfo.Arguments = cmd;

startInfo.UseShellExecute = false;

var process = new Process();

process.StartInfo = startInfo;

process.Start();

string output = "";

if (needOutput)

{

output = process.StandardOutput.ReadToEnd();

}

process.WaitForExit();

return output;

}

public string runAdbCommand(string cmd, bool needOutput = false)

{

return RunAdbCommand(String.Format("-s {0} {1}", this.deviceId, cmd), needOutput);

}

public string getCurrentView()

{

runAdbCommand("shell uiautomator dump /sdcard/view.xml");

var output = runAdbCommand("shell cat /sdcard/view.xml", true);

return output;

}

public static List<Tuple<string, string>> getListDevices()

{

var result = new List<Tuple<string, string>>();

string[] lines = RunAdbCommand("devices", true).Split('\n');

lines = lines.Skip(1).ToArray();

foreach (var line in lines)

{

var l = line.Trim();

if (l != "")

{

var words = l.Split('\t');

result.Add(new Tuple<string, string>(words[0], words[1]));

}

}

return result;

}

}

* + 1. **Lớp Base Script – Logic của một kịch bản**

public class BaseScript

{

private int tryCounter;

public BaseScript(int maxTry = 1)

{

this.tryCounter = maxTry;

}

public virtual bool RunScript()

{

init.Invoke();

while (!isError.Invoke() && !canAction.Invoke())

{

wait.Invoke();

if (--tryCounter == 0)

{

break;

}

}

if (canAction.Invoke())

{

action.Invoke();

if (isCompleted.Invoke())

{

onCompleted.Invoke();

return chooseAndRunNextScript();

}

else return false;

}

return false;

}

public Init init = () => { };

public Wait wait = () => { };

public Action action = () => { };

public CanAction canAction = () => true;

public IsCompleted isCompleted = () => true;

public IsError isError = () => false;

public OnCompleted onCompleted = () => { };

private List<Candidate> \_candidates = new List<Candidate>();

private List<BaseScript> \_scripts = new List<BaseScript>();

public BaseScript AddNext(BaseScript nextScript, Candidate choose)

{

this.\_candidates.Add(choose);

this.\_scripts.Add(nextScript);

return this;

}

public BaseScript AddNext(BaseScript nextScript)

{

this.\_candidates.Add(() => true);

this.\_scripts.Add(nextScript);

return this;

}

private bool chooseAndRunNextScript()

{

int index = 0;

foreach (Candidate candidate in this.\_candidates)

{

if (candidate.Invoke())

{

break;

}

++index;

}

if (index != this.\_candidates.Count)

{

return this.\_scripts[index].RunScript();

}

else return this.\_candidates.Count == 0;

}

}

}

* + 1. **Kịch bản Tạo tài khoản**

public override bool RunScript()

{

script.AddNext(

stopAcivity.AddNext(

startSetting.AddNext(

clickAccountSettings.AddNext(

clickAddAccount.AddNext(

clickAddGooleAccount.AddNext(

waitAndChooseCreateAccount.AddNext(

waitAndChooseCreateAccountForMyself.AddNext(

importNameAndChooseNext.AddNext(

importInforAndChooseNext.AddNext(

importUserNameAndChooseNext.AddNext(

importPasswordAndChooseNext.AddNext(

clickSkip.AddNext(

scrollClickSkip.AddNext(

clickNext.AddNext(

clickAgree )

)

)

)

)

)

)

)

)

)

)

)

)

)

);

return script.RunScript();}

1. **KẾT LUẬN**
   1. **Kết quả đạt được**

* Hiểu thêm về một số quy trình trên các nền tảng mạng xã hội
* Sử dụng được mô hình MVVM trong quá trình sản xuất ứng dụng bằng WPF hay Android
* Sử dụng tốt công cụ Visual Studio 2022
* Áp dụng các thuật toán tìm kiếm nâng cao vào thực tiễn (Thuật toán tìm kiếm chiều sâu để tìm nội dung trong các thuộc tính của file XML)
* Hiểu thêm về cách hoạt động của luồng (Thread) trong C#
* Xây dựng được ứng dụng WPF với các chức năng thuộc nền tảng Google và Youtube.
* Hiểu thêm về cách chạy lệnh Command Prompt sử dụng ngôn ngữ C#
* Sử dụng ADB để tạo thao tác tự động trên điện thoại
* Biết thêm về một số Framework về các API public của Google và Facebook
* Nâng cao khả năng tư duy, tự học, tự giải quyết vấn đề.

* 1. **Những khó khăn khi thực hiện**
* Do chưa được tiếp xúc với mô hình MVVM nên việc thực hiện code còn ban đầu còn nhiều khó khăn
* Do điều kiện không cho phép nên số thiết bị test còn ít (2 thiết bị) nên chưa chắc chắn về khả năng của ứng dụng
* Các nền tảng của các thiết bị khác nhau (MIUI, OneUI, ColorOS,…) trên các thiết bị khác nhau gây khó khăn trong việc đồng bộ hoạt động.
* Hệ thống bảo mật của Google, Facebook,… gây khó khăn trong các hoạt động tự động trên các nền tảng này.
  + Khi tạo tài khoản, sẽ có rất nhiều kịch bản khác nhau có thể xảy ra trong từng luồng thực hiện tự động. Việc phân tích tất cả các màn hình trở nên khó khăn.
  + Google, Facebook,… yêu cầu xác thực bằng số điện thoại dẫn đến việc yêu cầu một lượng lớn số điện thoại và hoạt động chúng – một việc rất khó khăn với trình độ hiện tại.
  + Các thuật toán tính lượt xem, lượt theo dõi khó lường (không cập nhật theo thời gian thực, lượng lượt xem có thể bị tăng giảm bất định,…)
  1. **Hướng phát triển trong tương lai**
* Xây dựng modul tạo các số điện thoại ảo và lấy thông tin từ số điện thoại đó
* Hướng đến phát triển ứng dụng phù hợp với đa nền tảng, đa thiết bị
* Nâng cao năng suất của ứng dụng
* Thực hiện test với số lượng lớn thiết bị khác nhau
* Xuất được màn hình các thiết bị và điều khiển chúng trên máy tính

# TÀI LIỆU THAM KHẢO

1. Android Developers. (n.d.). Android Developers. Truy cập từ <https://developer.android.com/docs/>
2. Wikipedia. Wikipedia. Truy cập từ <https://vi.wikipedia.org/>
3. Microsoft Learn. Learn. Truy cập từ <https://learn.microsoft.com/>
4. Pham Van Thinh. Tìm hiểu về mô hình MVVM. Viblo. Truy cập từ <https://viblo.asia/p/tim-hieu-ve-mo-hinh-mvvm-maGK7vW95j2/>
5. Stack Overflow. Stack Overflow. Truy cập từ <https://stackoverflow.com/>