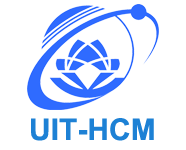
**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**



**DƯƠNG THÀNH ĐẠT – 1752004**

**BÙI TUẤN MINH - 17520107**

**ĐỒ ÁN 2**

**ỨNG DỤNG ĐỌC TIN TỨC**

**Build app for reading news from some source**

**KỸ SƯ NGÀNH CÔNG NGHỆ PHẦN MỀM**

**GIẢNG VIÊN HƯỚNG DẪN**

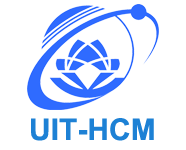
**GIẢNG VIÊN: THÁI THỊ HÀN UYỂN**

**TP. HỒ CHÍ MINH, 2021**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**



**DƯƠNG THÀNH ĐẠT – 17520040**

**BÙI TUẤN MINH - 17520107**

**ĐỒ ÁN 2**

**ỨNG DỤNG ĐỌC TIN TỨC**

**Build app for reading news from some source**

**KỸ SƯ NGÀNH CÔNG NGHỆ PHẦN MỀM**

**GIẢNG VIÊN HƯỚNG DẪN**

**GIẢNG VIÊN: THÁI THỊ HÀN UYỂN**

**TP. HỒ CHÍ MINH, 2021**

**LỜI CẢM ƠN**

Sau quá trình học tập và rèn luyện tại khoa Công nghệ phần mềm trường Đại học Công nghệ Thông tin – ĐHQG TP.HCM, em đã được trang bị các kiến thức cơ bản cùng các kỹ năng thực tế để có thể hoàn thành Đồ án 2 của mình.

Để hoàn thành khóa luận này, với lòng biết ơn sâu sắc em xin gửi lời cảm ơn chân thành đến:

Ban Giám hiệu trường Đại học Công nghệ Thông tin – ĐHQG TP.HCM vì đã tạo điều kiện thuận lợi để sinh viên tìm kiếm, nghiên cứu thông tin với hệ thống thư viện hiện đại, đa dạng các loại sách và tài liệu

Gần gũi hơn là những lời tốt đẹp nhất xin gửi đến đến cô Thái Thị Hàn Uyển đã tận tình giúp đỡ, định hướng cách tư duy và hướng làm việc khoa học. Đó là những góp ý hết sức quý báu không chỉ trong quá trình thực hiện luận văn mà còn là hành trang tiếp bước cho em trong quá trình học tập và làm việc sau này.

Sau cùng, xin chúc quý Thầy Cô trong khoa Công nghệ Phần mềm nói riêng cũng như các giáo viên tại trường Công nghệ thông tin nói chung thật dồi dào sức khỏe, niềm tin để tiếp tục thực hiện sứ mệnh cao đẹp của mình.

Thành phố Hồ Chí Minh, 10 tháng 06 năm 2021

Sinh viên 1

**Dương Thành Đạt**

Sinh viên 2

**Bùi Tuấn Minh**

MỤC LỤC

[Chương 1. TỔNG QUAN ĐỀ TÀI 1](#_Toc74722012)

[1.1. Lý do chọn đề tài 1](#_Toc74722013)

[1.2. Đối tượng nghiên cứu 1](#_Toc74722014)

[1.3. Khảo sát hiện trạng 1](#_Toc74722015)

[1.3.1. BÁO MỚI 1](#_Toc74722016)

[1.3.2. Tin tức 24h 3](#_Toc74722017)

[1.4. Mục tiêu của đề tài 4](#_Toc74722018)

[1.5. Phương pháp thực hiện 4](#_Toc74722019)

[Chương 2. KIẾN THỨC NỀN TẢNG 5](#_Toc74722020)

[2.1. FLASK 5](#_Toc74722021)

[2.1.1. Tại sao chọn Flask 5](#_Toc74722022)

[2.2. Flutter 6](#_Toc74722023)

[2.2.1. Tại sao chọn Flutter 7](#_Toc74722024)

[2.3. Firebase Authentication 9](#_Toc74722025)

[2.4. Cloud FireStore 10](#_Toc74722026)

[2.5. Beutifulsoup 11](#_Toc74722027)

[2.5.1. Web Crawling 11](#_Toc74722028)

[Chương 3. XÂY DỰNG HỆ THỐNG 12](#_Toc74722029)

[3.1. Xác định yêu cầu hệ thống 12](#_Toc74722030)

[3.2. Phân tích yêu cầu hệ thống 12](#_Toc74722031)

[3.3. Phân tích thiết kế hệ thống 14](#_Toc74722032)

[3.3.1. Sơ đồ use case 14](#_Toc74722033)

[3.4. Thiết kế kiến trúc phần mềm 21](#_Toc74722034)

[3.4.1. Kiến trúc phía backend 21](#_Toc74722035)

[3.4.2. Kiến trúc phía ứng dụng Mobile 22](#_Toc74722036)

[3.5. Thiết kế giao diện 24](#_Toc74722037)

[Chương 4. KẾT LUẬN, HƯỚNG PHÁT TRIỂN 32](#_Toc74722038)

[4.1. Ưu điểm 32](#_Toc74722039)

[4.2. Nhược điểm 32](#_Toc74722040)

[4.3. Hướng phát triển 32](#_Toc74722041)

DANH MỤC HÌNH VẼ

[Hình 1‑1: Ảnh minh hoạ ứng dụng Báo mới 2](#_Toc74759407)

[Hình 1‑2: Ảnh minh hoạ ứng dụng Tin tức 24h 3](#_Toc74759408)

[Hình 2‑1: Ảnh minh hoạ Flask 5](#_Toc74759409)

[Hình 2‑2: Ảnh minh hoạ Flutter. 7](#_Toc74759410)

[Hình 2‑3: Ảnh minh hoạ Firebase Authentication 9](#_Toc74759411)

[Hình 2‑4: Ảnh minh hoạ Cloud Firestore 10](#_Toc74759412)

[Hình 3‑1: Ảnh sơ đồ Use-case mức tổng quát 14](#_Toc74759413)

[Hình 3‑2: Mô hình kiến trúc Web Crawling + API 21](#_Toc74759414)

[Hình 3‑3: Mô hình kiến trúc phía ứng dụng Mobile 22](#_Toc74759415)

[Hình 3‑4: Màn hình đăng nhập chung 25](#_Toc74759416)

[Hình 3‑5: Màn hình  đăng nhập bằng tài khoản hệ thống 26](#_Toc74759417)

[Hình 3‑6: Màn hình đăng ký 27](#_Toc74759418)

[Hình 3‑7: Màn hình trang tin tức 28](#_Toc74759419)

[Hình 3‑8: Màn hình nội dung tin tức 29](#_Toc74759420)

[Hình 3‑9: Màn hình tài khoản 30](#_Toc74759421)

[Hình 3‑10: Màn hình kho lưu trữ 31](#_Toc74759422)

DANH MỤC BẢNG

[Bảng 3‑1: Danh sách các actors 14](#_Toc74759396)

[Bảng 3‑2: Danh sách các use-cases 15](#_Toc74759397)

[Bảng 3‑3: Đặc tả chức năng đăng ký 15](#_Toc74759398)

[Bảng 3‑4: Đặc tả chức năng đăng nhập 16](#_Toc74759399)

[Bảng 3‑5: Đặc tả chức năng Xem danh sách tin tức 17](#_Toc74759400)

[Bảng 3‑6: Đặc tả chức năng Đọc tin tức 17](#_Toc74759401)

[Bảng 3‑7: Đặc tả chức năng Lọc nguồn tin tức 18](#_Toc74759402)

[Bảng 3‑8: Đặc tả chức năng Xem tin tức theo chủ đề 18](#_Toc74759403)

[Bảng 3‑9: Đặc tả chức năng Lưu trữ tin tức 19](#_Toc74759404)

[Bảng 3‑10: Đặc tả chức năng Xóa tin tức lưu trữ 20](#_Toc74759405)

[Bảng 3‑11: Bảng danh sách màn hình 23](#_Toc74759406)

**TÓM TẮT ĐỒ ÁN 2**

1. Tổng quan đề tài: Lý do chọn đề tài, xác định đối tượng nghiên cứu, khảo sát hiện trạng, định hướng thực hiện và kết quả mong đợi.
2. Tìm hiểu các công nghệ mới phù hợp để xây dựng ứng dụng.
3. Xây dựng hệ thống: Xác định và phân tích yêu cầu, thiết kế cơ sở dữ liệu, kiến trúc back-end, fron-end và các màn hình.
4. Triển khai hệ thống lên máy chủ Linux và nhận kết quả phản hồi.
5. Viết báo cáo, đưa ra kết luận và hướng phát triển cho ứng dụng.

# TỔNG QUAN ĐỀ TÀI

## Lý do chọn đề tài

Trong một thế giới số đang ngày càng bùng nổ, việc tiếp cận thông tin một cách nhanh chóng trở thành nhu cầu tất yếu của mọi người. Trong đó, không thể không kể đến vai trò của thông tin báo chí, nơi cung cấp nhanh nhất những thông tin chính xác về cuộc sống hằng ngày.

Không khó để nhận ra hiện nay chúng ta có khá nhiều trang báo chính thống, hợp pháp của Việt Nam như VNEXPRESS, TUỔI TRẺ. Điều đó làm cho người dùng phải truy cập từng trang web để đọc tin tức từ những nguồn báo khác nhau, và có khả năng bỏ sót các nguồn báo khác. Vì vậy nhu cầu cần có một ứng dụng để tổng hợp các nguồn báo để phục vụ người dùng

## Đối tượng nghiên cứu

Đối tượng trong phạm vi đề tài hướng đến:

Những người Việt Nam có nhu cầu cập nhật tin tức báo chí

## Khảo sát hiện trạng

Thực tế hiện nay, Việt Nam đã có nhiều ứng dụng đọc tin tức quen thuộc có thể kể đến như app Báo mới, app Tin tức 24h. Ta cùng đi qua một số thông tin chung về các ứng dụng đọc báo hiện nay của Việt Nam

### BÁO MỚI

**Mô tả chung**

Báo mới thu thập tin tức từu 200 nguồn khác nhau của từ các trang báo điện tử của Việt Nam. Báo mới tự động phân tích, cập nhật và phân chia dữ liệu theo sở thích của người dùng



Hình 1‑1: Ảnh minh hoạ ứng dụng Báo mới

**Ưu điểm**

Giao diện sinh động

Nhiều tính năng tùy chỉnh cho người dùng

Tự động gộp các bài báo giống nhau về nguồn gốc

Nhiều nguồn báo

**Nhược điểm**

Có những tính năng thừa không cần thiết như đổi font, đổi theme

Nhiều lúc performance bị thấp

Hay bị mất kết nối mạng khi dùng app, không thể load dữ liệu

### Tin tức 24h

**Mô tả chung**

App thu thập tin tức từ 60 nguồn báo điện tử khác nhau của Việt Nam



Hình 1‑2: Ảnh minh hoạ ứng dụng Tin tức 24h

**Ưu điểm**

Giao diện mượt mà

Phù hợp với tất cả điện thoại và máy tính bảng

Dễ dàng chia sẻ tin tức lên mạng xã hội

**Nhược điểm**

Hay bị out khi đang sử dụng

Hay bị quảng cáo

Nhiều tin bị trùng lặp

## Mục tiêu của đề tài

Sau khi sử dụng và đưa ra những nhận xét về ưu nhược điểm của một vài ứng dụng đọc tin tức quen thuộc của Việt Nam, chúng em muốn triển khai một ứng dụng có thể đáp ứng các nhu cầu sau:

\_ Có thể lấy tin tức từ nhiều nguồn báo

\_ Giao diện tối giản nhất có thể để người dùng không gặp khó khan khi sử dụng

\_ Bỏ các tính năng thừa

\_ Cải thiện performance cho ứng dụng

## Phương pháp thực hiện

Tìm hiểu và đánh giá các giải pháp hiện có

Đề xuất, cải thiện các tính năng cần thiết

Tham khảo ý kiến giảng viên hướng dẫn để có định hướng tốt nhất

Phân tích và thiết kế hệ thống

Xây dựng ứng dụng.

Kiểm thử, triển khai và đánh giá kết quả.

**Kết quả mong đợi**

Hoàn thành ứng dụng với đầy đủ các yêu cầu chức năng đề ra

Giao diện thân thiện và mang lại trải nghiệm tốt nhất cho người dùng

Ứng dụng có độ hoàn thiện cao, có khả năng đưa vào thực tiễn

# KIẾN THỨC NỀN TẢNG

## FLASK

Flask là một framework web của Python, còn được coi là một microframework bởi nó không yêu cầu các công cụ hoặc thư viện cụ thể. Được phát triển bởi Armin Ronacher, người dẫn đầu một nhóm những người đam mê Python quốc tế có tên là Poocco.

Flask cung cấp cho bạn tất cả các công cụ, thư viện và công nghệ cho phép bạn xây dựng một ứng dụng web. Ứng dụng web này có thể là blog, wiki hay một ứng dụng lịch dựa trên web hoặc một web thương mại

**Flask**cũng **dễ mở rộng**để xây dựng các ứng dụng web phức tạp



Hình 2‑1: Ảnh minh hoạ Flask

### Tại sao chọn Flask

Flask có những ưu điểm sau:

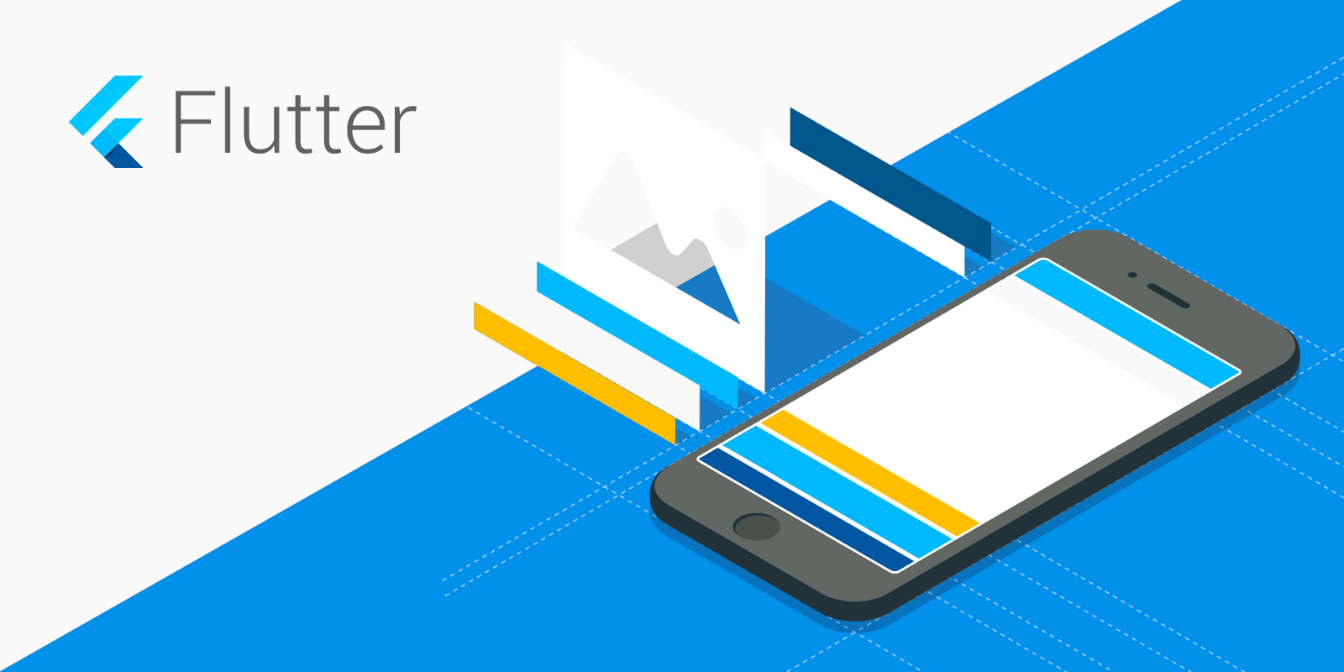
* Tốc độ
* Hỗ trợ cho NoQuery
* Độ phức tạp tối thiểu
* Chủ nghĩa tối giản tuyệt đối
* Không có ORM, dễ dàng kết nối với tiện ích mở rộng
* Trình gỡ lỗi được nhúng trong trình duyệt
* Mã ngắn và đơn giản trong số các bộ xương Python khác

Điểm nổi bật khi sử dụng Flask để lập trình web là sẽ rất ít bị phụ thuộc bên thứ ba, do đó đề phòng được các lỗi bảo mật.

Ta có thể kiểm soát mọi thứ khi sử dụng Flask. Và quan trọng, Flask giúp ta hiểu các cơ chế bên trong các Framework khác. Đây là tiền đề tốt để bạn có thể nắm giữ nhiều công nghệ hơn.

## Flutter

Flutter là mobile UI framework của Google để tạo ra các giao diện chất lượng cao trên iOS và Android trong khoảng thời gian ngắn. Flutter hoạt động với những code sẵn có được sử dụng bởi các lập trình viên, các tổ chức.



Hình 2‑2: Ảnh minh hoạ Flutter.

**Flutter** được phát triển nhằm giải quyết bài toán thường gặp trong mobile là Fast Development và Native Performance. Nếu như React Native chỉ đảm bảo Fast Development và code native thuần chỉ đảm bảo Native Performance thì Flutter làm được cả 2 điều trên.

### Tại sao chọn Flutter

* Nếu bạn đang tìm kiếm các phương pháp thay thế để phát triển ứng dụng Android, bạn nên cân nhắc thử Flutter của Google, một framework dựa trên ngôn ngữ lập trình Dart.
* Các ứng dụng được xây dựng với Flutter hầu như không thể phân biệt với những ứng dụng được xây dựng bằng cách sử dụng Android SDK, cả về giao diện và hiệu suất. Hơn nữa, với những tinh chỉnh nhỏ, chúng có thể chạy trên thiết bị iOS.
* Chạy ở 60 fps, giao diện người dùng được tạo ra với Flutter thực thi tốt hơn nhiều so với những ứng dụng được tạo ra với các framework phát triển đa nền tảng khác chẳng hạn như React Native và Ionic. Một số lí do khiến bạn có thể hứng thú với Flutter:
  1. Flutter sử dụng Dart, một ngôn ngữ nhanh, hướng đối tượng với nhiều tính năng hữu ích như mixin, generic, isolate, và static type.
  2. Flutter có các thành phần UI của riêng nó, cùng với một cơ chế để kết xuất chúng trên nền tảng Android và iOS. Hầu hết các thành phần giao diện người dùng, đều sẵn dùng, phù hợp với các nguyên tắc của Material Design.
  3. Các ứng dụng Flutter có thể được phát triển bằng cách sử dụng IntelliJ IDEA, một IDE rất giống với Android Studio

Điểm nổi bật:

* Fast Development: Tíng năng Hot Reload hoạt động trong milliseconds để hiện thị giao diện tới bạn. Sử dụng tập hợp các widget có thể customizable để xây dựng giao diện trong vài phút. Ngoài ra Hot Reload còn giúp bạn thêm các tính năng, fix bug tiết kiệm thời gian hơn mà không cần phải thông qua máy ảo, máy android hoặc iOS.
* Expressive and Flexible UI: Có rất nhiều các thành phần để xây dựng giao diện của Flutter vô cùng đẹp mắt theo phong cách Material Design và Cupertino, hỗ trợ nhiều các APIs chuyển động, smooth scrolling..
* Native Performance: Các widget của fluter kết hợp các sự khác biệt của các nền tảng ví dụ như scrolling, navigation, icons, font để cung cấp một hiệu năng tốt nhất tới iOS và Android.

## Firebase Authentication

Firebase Authentication là chức năng dùng để xác thực người dùng bằng Password, số điện thoại hoặc tài khoản Google, Facebook hay Twitter, v.v. Đồng thời cũng xác thực nặc danh cho các ứng dụng. Hoạt động xác thực có thể giúp thông tin cá nhân của người sử dụng được an toàn và đảm bảo không bị đánh cắp tài khoản



Hình 2‑3: Ảnh minh hoạ Firebase Authentication

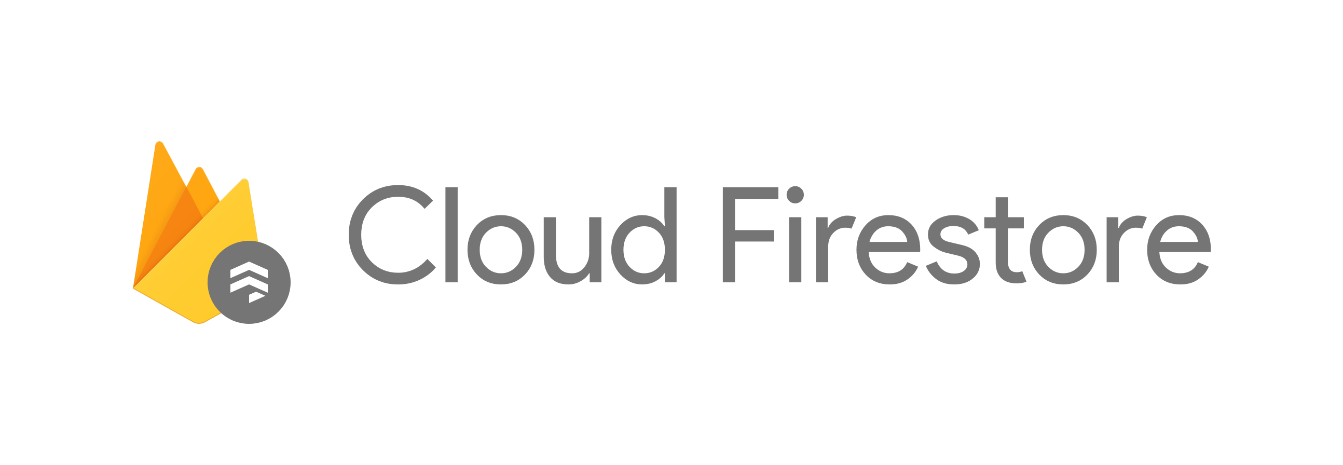
Việc xác thực người dùng là một chức năng quan trọng trong phát triển ứng dụng. Tuy nhiên, việc đối ứng với nhiều phương pháp xác thực khác nhau sẽ tốn nhiều thời gian và công sức.

Firebase Authentication giúp thực hiện việc chia sẻ ID giữa các ứng dụng, giúp người dùng dễ dàng tiếp cận sản phẩm hơn. Vì thế, nó là một chức năng rất quý.

## Cloud FireStore

Cloud Firestore là một Database linh hoạt và dễ mở rộng cho mobile, web và server được phát triển từ Firebase and Google Cloud Platform. Cũng giống như Firebase realtime database Cloud Firestore giúp cho việc đồng bộ dữ liệu giữa các ứng dụng phía client một các nhanh chóng (Realtime) và hộ trợ lưu offline data trong ứng dụng của bạn.

Cloud Firestore là một cloud-hosted, NoSQL database mà các ứng dụng phía client có thể trực tiếp truy cập thông qua native SDKs. Nó lưu dữ liệu theo mô hình dữ liệu NoSQL. Dữ liệu được lưu trữ trong các file tài liệu chứa các trường được ánh xạ vào các giá trị. Các file tài liệu này được lưu trữ trong các tập hợp chúng có thể sử dụng nó để tổ chức dữ liệu và truy vấn dữ liệu. Cloud Firestore hỗ trợ rất nhiều kiểu dữ liệu từ đơn giản như String, Integer hay những kiểu dữ liệu phức tạp như các nested object.



Hình 2‑4: Ảnh minh hoạ Cloud Firestore

**Một số điểm đặc biệt:**

* Sự ổn định và hiệu năng: dữ liệu của bạn được đặt ở nhiều nơi đảm bảo tính mở rộng và độ tin cậy cao
* Khả năng mở rộng: Việc mở rộng này là hoàn toàn tự động, vậy nên bạn sẽ không cần phải quan tâm đến việc dữ liệu của mình lưu trữ theo nhiều phiên bản
* Bảo mật: bảo mật đơn giản và mạnh mẽ cho SDK di động, web và server: SDK trên thiết bị di động và web sử dụng các quy tắc bảo mật của Cloud Firestore.

## Beutifulsoup

Beautifulsoup là một thư viện Python hỗ trợ trong việc craw data từ web

### Web Crawling

**Web crawling** là quá trình tự động trích xuất các thông tin từ các trang web và lưu trữ nó dưới một định dạng phù hợp. Chương trình mà thực hiện công việc này gọi là **web crawler**.

Thông thường, khi muốn lấy một số thông tin từ các trang web, chúng ta sẽ dùng các API mà các trang đó cung cấp. Đây là cách đơn giản, tuy nhiên không phải trang web nào cũng cung cấp sẵn API cho chúng ta sử dụng. Do đó chúng ta cần một kĩ thuật để lấy các thông tin từ các trang web đó mà không thông qua API.

Việc crawling tùy thuộc vào layout của từng trang web và layout đó có thể thay đổi theo thời gian. Do đó khi layout thay đổi, chúng ta cũng có thể sẽ phải thay đổi code.

# XÂY DỰNG HỆ THỐNG

## Xác định yêu cầu hệ thống

Qua khảo sát tìm hiểu, nhận thấy ứng dụng cần đạt được các yêu cầu sau:

Thông tin chính xác: Các bài tin tức được thu thập từ các nguồn chính thống, có tính chính xác cao.

Tin tức:

* Cung cấp nội dung về tin tức.
* Cung cấp các hình ảnh minh họa (nếu có).
* Các bài tin tức được cập nhật mới theo thời gian được cài đặt, đảm bảo không bị out-of-date.

Lưu lại bài tin tức: tạo thành kho lưu trữ khi người dùng có nhu cầu lưu lại để đọc sau hoặc thu thập các tin tức cần thiết.

Lọc bài theo chủ đề: Kết quả trả về là các bài tin tức được sắp xếp theo các chủ đề khác nhau.

Lọc bài theo nguồn cung cấp: Kết quả trả về là các bài tin tức của một nguồn tin

Tài khoản người dùng:

* Đăng nhập
* Kho lưu trữ tin tức

## Phân tích yêu cầu hệ thống

**Đăng ký:** Người dùng sẽ được tạo tài khoản mới.

**Đăng nhập:** Người dùng được chọn 1 trong số các phương pháp đăng nhập bằng tài khoản Facebook, Google và tài khoản đã đăng ký để đăng nhập hoặc dùng tài khoản ẩn danh. Khi đăng nhập vào thành công, khách hàng có thể sử dụng các chức năng ứng dụng cung cấp.

**Xem thông tin user profile**: Xem được ảnh đại diện, tên hiển thị trong ứng dụng và kho lưu trữ tin tức cá nhân.

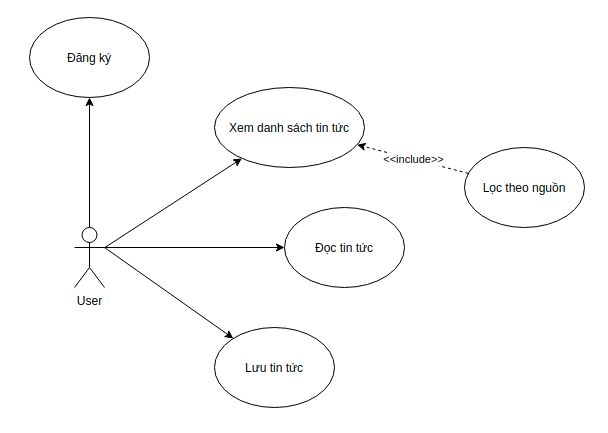
**Tin tức:** Tổng hợp các bài viết từ các trang tin tức uy tín Tuổi trẻ, VNExpress, Zing News, … bao gồm hình ảnh, nội dung, tác giả, thời gian

**Tin tức theo chủ đề:** Danh sách các chủ đề tin tức phổ biến như Thế giới, Kinh doanh, Giải trí, Pháp luật,.... Các bài tin tức sẽ được phân chia theo chủ đề

**Kho lưu trữ của tôi**: Người dùng có thể lưu lại những bài tin tức vào một kho lưu trữ cá nhân tùy vào mục đích sử dụng

## Phân tích thiết kế hệ thống

### Sơ đồ use case



Hình 3‑1: Ảnh sơ đồ Use-case mức tổng quát

Bảng ‑: Danh sách các actors

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên actor** | **Ý nghĩa** |
| 1 | User | Người dùng đã có tài khoản đăng nhập để được sử dụng các tính năng ứng dụng cung cấp |

Bảng ‑: Danh sách các use-cases

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên use case** | **Ý nghĩa** |
| 1 | Use case Đăng ký | Người dùng đăng ký tài khoản trên hệ thống |
| 2 | Use case Đăng nhập | Người dùng đăng nhập để yêu cầu xác thực và sử dụng các tính năng trong ứng dụng |
| 3 | Use case Xem danh sách tin tức | Người dùng được xem danh sách các bài tin tức |
| 4 | Use case Đọc tin tức | Người dùng đọc nội dung tin tức |
| 5 | Use case Lọc nguồn tin tức | Người dùng lọc ra nguồn tin tức muốn xem |
| 6 | Use case Xem tin tức theo chủ đề | Người dùng được xem các tin tức sắp xếp theo các chủ đề khác nhau |
| 7 | Use case Lưu trữ tin tức | Người dùng có thể lưu lại các tin tức mà họ quan tâm |
| 8 | Use case Xóa tin lưu trữ | Người dùng có thể xóa đi các tin tức đã lưu khi không cần thiết nữa |

**Đặc tả một số Use cases chính**

Bảng ‑: Đặc tả chức năng đăng ký

|  |  |
| --- | --- |
| **Tên chức năng** | Đăng ký |
| **Tóm tắt** | Chức năng đăng ký tài khoản người dùng |
| **Dòng sự kiện chính** | Hệ thống hiển thị form thông tin đăng ký  Người dùng nhập thông tin bắt buộc và nhấn “Create an account”  Hệ thống kiểm tra thông tin đã nhập và tạo tài khoản mới |
| **Dòng sự kiện khác** | Thông tin nhập không hợp lệ: Hệ thống hiển thị thông báo các trường bỏ trống hoặc trùng username |

Bảng ‑: Đặc tả chức năng đăng nhập

|  |  |
| --- | --- |
| **Tên chức năng** | Đăng nhập |
| **Tóm tắt** | Chức năng đăng nhập vào hệ thống |
| **Dòng sự kiện chính** | Đăng nhập bằng tài khoản Google:   * Người dùng chọn “Sign in with Google”   Đăng nhập bằng tài khoản Facebook:   * Người dùng chọn “Sign in with Facebook"   Đăng nhập bằng tài khoản đăng ký trên ứng dụng   * Hệ thống hiển thị form đăng nhập * Người dùng nhập username và password và nhấn “Sign in”   Hệ thống kiểm tra thông tin đăng nhập  Hệ thống hiển thị trang chủ |
| **Dòng sự kiện khác** | Hệ thống trả ra thông báo khi thông tin đăng nhập không hợp lệ |
| **Trạng thái hệ thống trước khi thực hiện use case** | Người dùng chưa đăng nhập không được sử dụng các tính năng |
| **Trạng thái hệ thống sau khi thực hiện use case** | Người dùng đã đăng nhập được dùng sử dụng các tính năng |

Bảng ‑: Đặc tả chức năng Xem danh sách tin tức

|  |  |
| --- | --- |
| **Tên chức năng** | Xem danh sách tin tức |
| **Tóm tắt** | Chức năng Xem danh sách tin tức |
| **Dòng sự kiện chính** | Người dùng chọn tab “News" trong ứng dụng để xem danh sách các tin tức thu thập từ các nguồn khác nhau |
| **Dòng sự kiện khác** | Không có |
| **Trạng thái hệ thống trước khi thực hiện use case** | Không có |
| **Trạng thái hệ thống sau khi thực hiện use case** | Hệ thống hiển thị danh sách tin tức |

Bảng ‑: Đặc tả chức năng Đọc tin tức

|  |  |
| --- | --- |
| **Tên chức năng** | Đọc tin tức |
| **Tóm tắt** | Chức năng đọc nội dung các bài tin tức |
| **Dòng sự kiện chính** | Người dùng tìm kiếm bài tin tức và chọn vào tin tức muốn xem |
| **Dòng sự kiện khác** | Không có |
| **Trạng thái hệ thống trước khi thực hiện use case** | Không có |
| **Trạng thái hệ thống sau khi thực hiện use case** | Hệ thống hiển thị nội dung tin tức, bao gồm hình ảnh, mô tả, nội dung, tác giả, nguồn |

Bảng ‑: Đặc tả chức năng Lọc nguồn tin tức

|  |  |
| --- | --- |
| **Tên chức năng** | Lọc nguồn tin tức |
| **Tóm tắt** | Chức năng lọc tin tức theo nguồn tin cung cấp |
| **Dòng sự kiện chính** | Người dùng chọn nguồn có trong dropdown button “Source" |
| **Dòng sự kiện khác** | Không có |
| **Trạng thái hệ thống trước khi thực hiện use case** | Không có |
| **Trạng thái hệ thống sau khi thực hiện use case** | Hệ thống trả về danh sách tin tức theo nguồn tin đã lựa chọn |

Bảng ‑: Đặc tả chức năng Xem tin tức theo chủ đề

|  |  |
| --- | --- |
| **Tên chức năng** | Xem tin tức theo chủ đề |
| **Tóm tắt** | Chức năng cho phép người dùng xem tin tức sắp xếp theo các chủ đề |
| **Dòng sự kiện chính** | Người dùng chọn tab chủ đề tin tức muốn xem |
| **Dòng sự kiện khác** | Không có |
| **Trạng thái hệ thống trước khi thực hiện use case** | Không có |
| **Trạng thái hệ thống sau khi thực hiện use case** | Hệ thống hiển thị danh sách tin tức theo chủ đề đã chọn |

Bảng ‑: Đặc tả chức năng Lưu trữ tin tức

|  |  |
| --- | --- |
| **Tên chức năng** | Lưu trữ tin tức |
| **Tóm tắt** | Chức năng cho người dùng có thể lưu trữ tin tức quan tâm |
| **Dòng sự kiện chính** | Người dùng chọn tin tức quan tâm và nhấn vào nút “Book mark”(icon) |
| **Dòng sự kiện khác** | Không có |
| **Trạng thái hệ thống trước khi thực hiện use case** | Hệ thống kiểm tra tin tức đã được lưu hay chưa |
| **Trạng thái hệ thống sau khi thực hiện use case** | Hệ thống sẽ lưu lại tin tức trong kho lưu trữ của người dùng  Người dùng có thể vào tab “Account” chọn “Repository” để xem kho lưu trữ |

Bảng ‑: Đặc tả chức năng Xóa tin tức lưu trữ

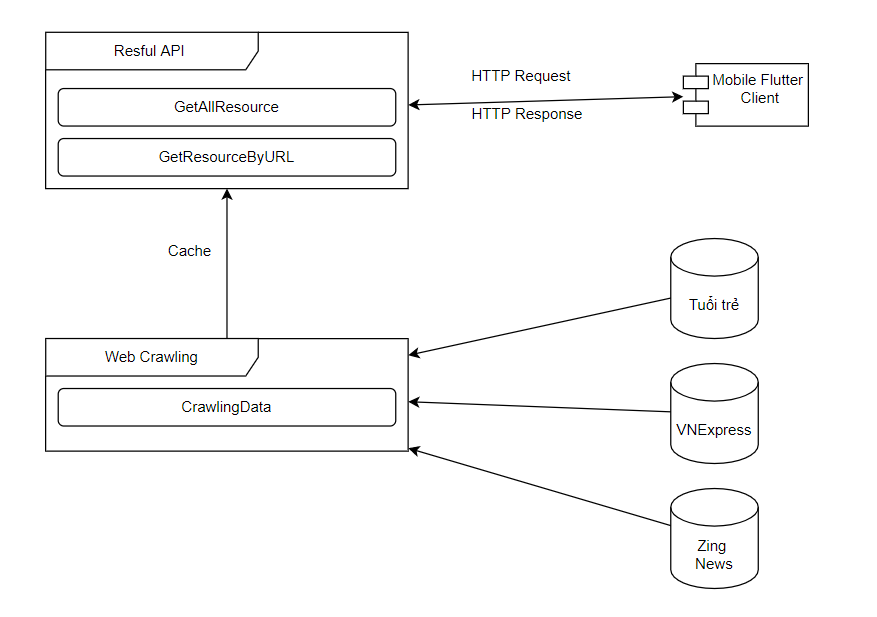
|  |  |
| --- | --- |
| **Tên chức năng** | Xóa tin tức lưu trữ |
| **Tóm tắt** | Chức năng cho phép người dùng có thể xóa đi các tin tức đã lưu khi không cần thiết nữa |
| **Dòng sự kiện chính** | Người dùng chọn tab “Account” và tiếp tục vào “Repository”  Người dùng kéo tin tức qua trái chọn “Delete” |
| **Dòng sự kiện khác** | Không có |
| **Trạng thái hệ thống trước khi thực hiện use case** | Không có |
| **Trạng thái hệ thống sau khi thực hiện use case** | Hệ thống xóa tin tức người dùng chọn và cập nhật lại danh sách kho lưu trữ |

## Thiết kế kiến trúc phần mềm

Hệ thống bao gồm 2 phần là ứng dụng Mobile và Web Crawling + API. Trong đó ứng dụng Mobile được xây dựng trên công nghệ Flutter và Web Crawling + API được xây dựng bằng công nghệ Flask với sự hỗ trợ của thư viện Beautifulsoup.

### Kiến trúc phía Web Crawling + API

Phía backend được thiết kế theo mô hình Layers. Trong đó có 3 layers chính là Controler, Services và Repositories; bên cạnh có một module Helper. Chi tiết các thành phần như sau:

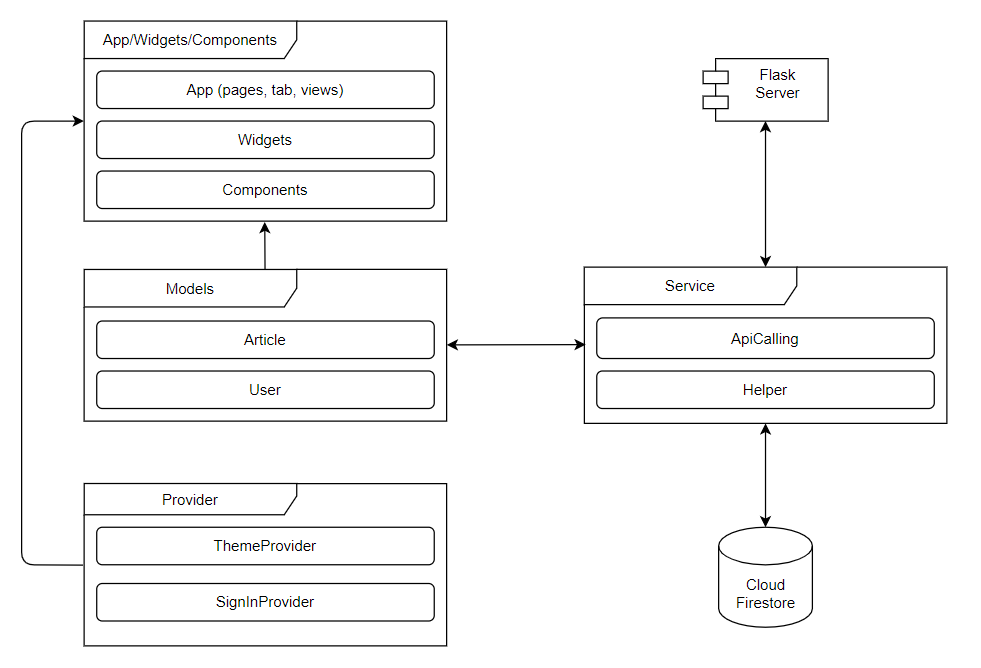


Hình 3‑2: Mô hình kiến trúc Web Crawling + API

* **Resful API**: Được dùng để cấu hình và xử lý các request, response giữa ứng dụng và Data. Resful API là nơi định nghĩa và xử lý các web api. Trong khi đó, Resources sẽ là nơi định nghĩa các loại data nằm trong request và response, cùng với một số phương thức chuyển đổi giữa resources và models.
* **Web Crawling**: Được dùng để cấu hình và xử lý các hoạt động crawl data từ các nguồn thông tin, người ra còn có nhiệm vụ lưu lại dữ liệu đã crawl vào bộ nhớ đệm của server. Sau đó chuyển các data này cho các API xử lý.

### Kiến trúc phía ứng dụng Mobile

Phía ứng dụng Mobile được xây dựng dựa trên quy chuẩn của Flutter. Trong đó có 4 thành phần gồm có: Layout, Models, Provider, Services và chi tiết như sau:



Hình 3‑3: Mô hình kiến trúc phía ứng dụng Mobile

* **App/Widgets/Components**: thành phần này gồm nhiều thư mục trong source code. Nhưng chung quy thì vẫn là thực hiện nhiệm vụ định nghĩa các thành phần giao diện cho ứng dụng Mobile
* **Models**: Thư mục chứa mọi vấn đề liên quan đến định nghĩa loại dữ liệu sử dụng trong ứng dụng
* **Provider**: Thư viện Provider đã quá quen thuộc cũng như là cần thiết cho việc quản lý state trong Flutter. Và đây là nơi triển khai source code cho Provider
* **Services**: Đây là module dùng để chứa những hàm hỗ trợ xử lý cho phía ứng dụng. Nó có thể là việc gọi API, liên kết với CSDL, xử lý lỗi,...

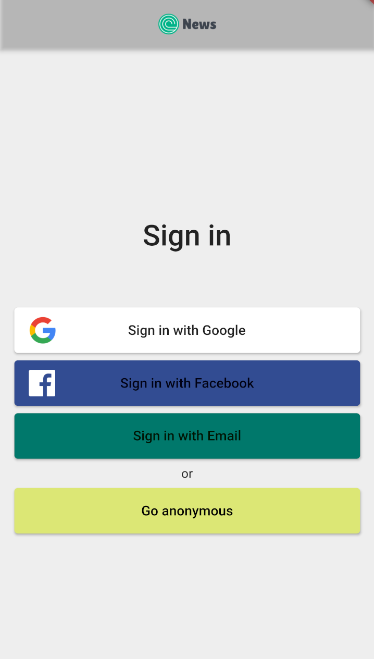
## Thiết kế giao diện

Ứng dụng News chia làm 2 nhóm người dùng, bao gồm: người dùng đã đăng nhập bằng tài khoản đăng ký (khách hàng) và guest (người dùng sử dụng tài khoản ẩn danh).

Bảng ‑: Bảng danh sách màn hình

|  |  |  |
| --- | --- | --- |
| STT | Tên màn hình | Mô tả chức năng |
| 1 | Đăng ký | Đăng ký tài khoản mới |
| 2 | Đăng nhập | Đăng nhập vào hệ thống |
| 3 | Trang tin tức | Danh sách tin tức mới nhất, và các chủ đề |
| 4 | Nội dung tin tức | Hiển thị nội dung chi tiết của tin tức |
| 5 | Tài khoản | Xem thông tin cá nhân của tài khoản |
| 6 | Kho lưu trữ | Danh sách các tin tức được người dùng lưu lại |
| 7 | Cài đặt | (Chưa phát triển) |

**Một số màn hình ứng dụng nổi bật.**



Hình 3‑4: Màn hình đăng nhập chung

Ảnh có chứa văn bản

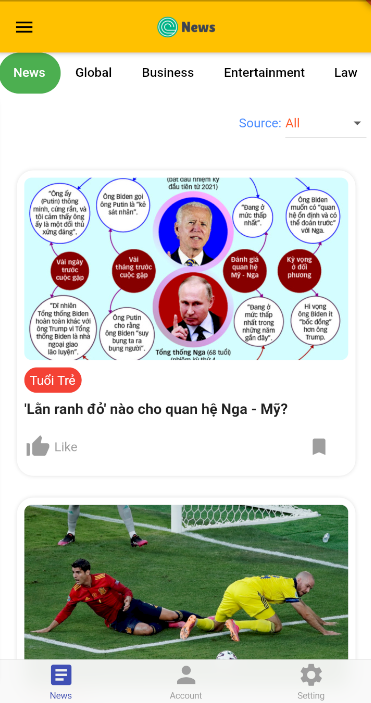
Mô tả được tạo tự động

Hình 3‑5: Màn hình  đăng nhập bằng tài khoản hệ thống

Ảnh có chứa văn bản

Mô tả được tạo tự động

Hình 3‑6: Màn hình đăng ký

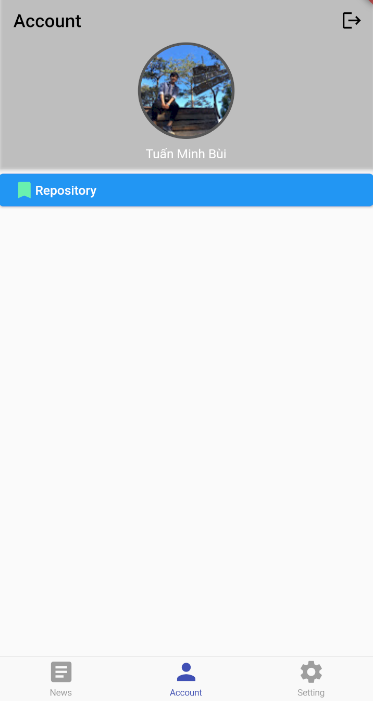


Hình 3‑7: Màn hình trang tin tức

Ảnh có chứa văn bản

Mô tả được tạo tự động

Hình 3‑8: Màn hình nội dung tin tức



Hình 3‑9: Màn hình tài khoản



Hình 3‑10: Màn hình kho lưu trữ

# KẾT LUẬN, HƯỚNG PHÁT TRIỂN

## Ưu điểm

Hoàn thành yêu cầu đặt ra ban đầu, có đầy đủ tính năng cần thiết của một ứng dụng đọc tin tức trên thiết bị di động.

Giao diện đơn giản, bố cục hợp lý, người dùng dễ dàng theo tác.

Sinh viên thực hiện đã nắm được các kiến thức và tổ chức trong việc lập trình ứng dụng mobile.

## Nhược điểm

* Còn hạn chế một số tính năng:
* Tối ưu code chưa tốt nên tốc độ làm mới dữ liệu khi ứng dụng cập nhật tin tức còn chậm
* Chưa phát triển tính năng xác thực qua bằng email, số điện thoại cho tài khoản
* Còn hạn chế số lượng nguồn tin tức do chưa tối ưu hóa khả năng cập nhật tin tức
* Chưa có thông báo khi ứng dụng cập nhật tin tức mới
* Chưa có chức năng xóa tài khoản

## Hướng phát triển

Xây dựng các chức năng còn thiếu, hoàn thiện các chức năng đã có

Tối ưu hóa code để tăng tốc độ tương tác

Hỗ trợ thêm đa ngôn ngữ cho các quốc gia

Triển khai trên thiết bị thực tế

TÀI LIỆU THAM KHẢO

[1] <https://flutter.dev/docs>

[2] <https://flask-restful.readthedocs.io/en/latest/>

[3] <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

[4] <https://firebase.google.com/docs>