BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC YERSIN ĐÀ LẠT**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO THỰC TẬP**

**MAPBOX FLUTTER**

**GVHD : Nguyễn Thị Thu Thảo**

**SVTH : Vũ Chí Hùng Mã số SV : 12001070**

**Lớp : Công Nghệ Phần Mềm K17**

**Khóa học : 2020 - 2023**

Đà Lạt, tháng 05- 2023

Mục Lục

[LỜI CẢM ƠN 4](#_Toc135855662)

[CHƯƠNG 1. GIỚI THIỆU VỀ ĐƠN VỊ THỰC TẬP 9](#_Toc135855663)

[1.1 Thông tin về đơn vị thực tập 9](#_Toc135855664)

[1.2 Các trung tâm 9](#_Toc135855665)

[1.3 Lĩnh vực hoạt động 11](#_Toc135855666)

[CHƯƠNG 2. GIỚI THIỆU VỀ FLUTTER 11](#_Toc135855667)

[I. Flutter là gì? 11](#_Toc135855668)

[II. Tổng quan về Flutter 12](#_Toc135855669)

[1. Dart là gì? 12](#_Toc135855670)

[2. Layout & Naviagtion là gì? 12](#_Toc135855671)

[3. Widget, Lifecycle & Tabbar 12](#_Toc135855672)

[4. State Management, Stream & Bloc là gì? 14](#_Toc135855673)

[5. Scoped access & Provider là gì? 14](#_Toc135855674)

[6. Mapbox Flutter là gì? 15](#_Toc135855675)

[III. Giới thiệu về App Mapbox Flutter 15](#_Toc135855676)

[1. App Mapbox Flutter là gì? 15](#_Toc135855677)

[2. Giao diện App MapBox Flutter? 16](#_Toc135855678)

[CHƯƠNG 3. NỘI DUNG THỰC TẬP 22](#_Toc135855679)

[1. Kế hoạch thực tập 22](#_Toc135855680)

[2. Mục tiêu thực tập. 22](#_Toc135855681)

[3. Nội dung thực tập 22](#_Toc135855682)

[4. Tiến độ thực tập 22](#_Toc135855683)

[CHƯƠNG 4: GIẢI PHÁP THỰC HIỆN 23](#_Toc135855684)

[CHƯƠNG 5: KẾT QUẢ THỰC TẬP 23](#_Toc135855685)

[1. Mức độ hoàn thiện 23](#_Toc135855686)

[2. Kỹ năng làm việc thực tế 25](#_Toc135855687)

[3. Kinh nghiệm tích lỹ 25](#_Toc135855688)

[4. Kết quả đạt được 25](#_Toc135855689)

[5. Nhược điểm và thiếu sót 26](#_Toc135855690)

[6. Hướng phát triển của dự án 26](#_Toc135855691)

[DANH MỤC TỪ VIẾT TẮT 26](#_Toc135855692)

# LỜI CẢM ƠN

Lời đầu tiên, em xin gửi đến quý thầy giáo, cô giáo trường Đại HọcYerSin Đà Lạt, quý lãnh đạo công ty TMA Solutions Bình Định lời cảm ơn chân thành và sâu sắc nhất.

Đặc biệt, em xin gửi lời cảm ơn đến Cô Nguyền Thị Thu Thảo – giảng viên hướng dẫn thực tập, anh Trần Duy Tân – mentor trực tiếp hướng dẫn em trong kỳ thực tập này lời cảm ơn chân thành và sâu sắc nhất.

Em xin chân thành cảm ơn các anh, các chị nhân viên bộ phận DC2 - TIP đã tạo điều kiện thuận lợi cho em tìm hiểu, trải nghiệm thực tế những công việc thực tiễn ở một công ty phát triển phần mềm.

Sau một thời gian học tập và trau dồi kiến thức trên ghế nhà trường, em đã có cơ hội thực tập tại công ty TMA Solutions Bình Định. Đây là cơ hội để em áp dụng những kiến thức đã học vào thực tế và tự mình tích lũy thêm nhiều kinh nghiệm quý báu. Sau những nỗ lực của bản thân cùng với sự giúp đỡ của mọi người, em đã hoàn thành xong chương trình thực tập của mình và nhận lại một hành trang kiến thức vô cùng quý báu để em tiếp bước sau này.

Với thái độ nghiêm túc, dù đã hoàn thành chuyên đề với những nỗ lực cao nhất nhưng sẽ không tránh khỏi những thiếu sót do kiến thức còn hạn chế. Em rất mong nhận được những góp ý của mọi người để chuyên đề được hoàn thiện hơn.

Em trân trọng cảm ơn.

*Quy Nhơn,* ngày 24 tháng 04 năm 2023

Vũ Chí Hùng

**NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP**

**Xác nhận của Công ty** *Quy Nhơn, ngày … tháng … năm 2023*

**TP. Hành chính và Tổng hợp Người hướng dẫn**

**Lâm Thị Thanh Thảo**

| **Tuần** | **Ngày** | **Nội dung** | **Nhận xét của CBHD tại đơn vị TT** *(Ký tên)* | **Nhận xét của GVHD tại trường** *(Ký tên)* |
| --- | --- | --- | --- | --- |
| 1 | 9/3/2023-12/3/2023 | - Chào mừng.  - Set up máy tính và nhận mentor.  - Tìm hiểu và cài đặt môi trường flutter. |  |  |
| 2 | 13/3/2023-19/3/2023 | - Learn basic dart syntax about the type of variable, function, list, OOP, special keywords…  - Types of class constructors.  - Factory constructor. |  |  |
| 3 | 20/3/2023  26/3/2023 | -Flutter Basic  +Widget Lifecycle  +Layout  +Gestures & InkWell  + TextFiled and TextFormFiled  +….. |  |  |
| 4 | 27/3/2023  2/4/2023 | -Navigate to a new screen and back  -Navigate with named routes  - Send data to a new screen & Return data from a screen. |  |  |
| 5 | 3/4/2023  9/4/2023 | - Food App Exercise |  |  |
| 6 | 10/4/2023  16/4/2023 | - Tabbar  - Drawer  -Animation |  |  |
| 7 | 17/4/2023  23/4/2023 | -States Management  - Bloc design pattern |  |  |
| 8 | 24/4/2023  30/4/2023 | -Counter App Exercise  -Flutter Permission  - Future and Completer |  |  |
| 9 | 5/5/2023  6/5/2023 | -Flutter Firebase  -Flutter Rest API  -Stream |  |  |
| 10 |  | -Display Map  -Zoom in & Zoom out  - Add symbol & Move camera |  |  |
| 11 |  | -Read/Write file JSON  -Add symbol with JSON data  - Filtering symbol |  |  |
| 12 |  | -Login/Sign up  -Setting Screen  -Custom Menu and UI |  |  |

|  |  |
| --- | --- |
| **Cán bộ hướng dẫn tại đơn vị TT**  *(Ký và ghi rõ họ tên)* | *Đà Lạt, ngày … tháng … năm ……….*  **Giảng viên hướng dẫn tại trường**  *(Ký và ghi rõ họ tên)* |

# 

# CHƯƠNG 1. GIỚI THIỆU VỀ ĐƠN VỊ THỰC TẬP

## Thông tin về đơn vị thực tập

TMA Solutions được thành lập năm 1997, là công ty phần mềm hàng đầu Việt Nam. Với sự phát triển bền vững về nhân lực, hiện nay, TMA đã xây dựng được đội ngũ 4000 kỹ sư trẻ, nhiệt huyết luôn nhận được sự tin cậy, đánh giá cao từ phía khách hàng.

Hiện Công ty có 6 văn phòng tại TP HCM và Công viên sáng tạo TMA tại Bình Định.

Trải qua 26 năm hoạt động, TMA tự hào nhận được rất nhiều bằng khen, giải thưởng và liên tục được báo chí, truyền thông trong và ngoài nước đưa tin.

Là doanh nghiệp CNTT có 26 năm kinh nghiệm, làm việc với rất nhiều khách hàng hàng đầu thế giới, TMA tự hào đạt các chuẩn quốc tế cao cấp như CMMI-L5, TL 9000, ISO 9001:2000, ISO 27001:2013.

Để có được những thành công trên, TMA chủ trương tuyển dụng những lập trình viên tài năng, có khả năng giao tiếp bằng tiếng Anh và có kỹ năng tốt.

Ứng viên sau khi gia nhập TMA sẽ được hướng dẫn cụ thể về công ty, các quy định và vị trí công việc. Theo đó, ngay trong quá trình thử việc, nhân viên sẽ được tham gia các khóa training về kỹ năng làm việc, giao tiếp để bổ sung đầy đủ các yêu cầu cần thiết khi làm việc tại TMA.

Là trung tâm phần mềm đầu tiên tại Thung lũng Sáng tạo Quy Nhơn, Công viên Sáng tạo TMA mang sứ mệnh trở thành trung tâm phát triển phần mềm và công nghệ cao hàng đầu tại miền Trung, góp phần quan trọng đưa Thung lũng sáng tạo Quy Nhơn trở thành một điểm đến của công nghệ 4.0 tại Việt Nam. Công viên Sáng tạo TMA bao gồm Trung tâm Phát triển Phần Mềm, Xưởng Phần mềm, Trung tâm R&D, Trung tâm Khoa học Dữ liệu, Học viện Công Nghệ…với tổng diện tích sử dụng hơn 15,000m2.

## Các trung tâm

* Trung tâm Phát triển Phần mềm (Software Development Center)
  + Thừa hưởng 25 năm kinh nghiệm, công nghệ và quy trình phát triển phần mềm của TMA.
  + Cung cấp các giải pháp và dịch vụ phần mềm cho khách hàng trong và ngoài nước (thị trường hiện có của TMA tại 30 nước). Trong năm đầu tiên đã hoàn thành 8 dự án cho các khách hàng từ Mỹ, Canada, Úc, Nhật Bản, Hàn Quốc và Việt Nam
* Xưởng Phần mềm (Software Factory)
  + Chắt lọc và phát huy kiến thức dày dặn hơn 25 năm làm phần mềm của TMA, xưởng phần mềm sẽ là nơi tập trung sản xuất các sản phẩm ứng dụng công nghệ 4.0 không chỉ cho thị trường Việt Nam mà còn là quốc tế.
  + Xưởng phần mềm chào đón tất cả các cơ hội hợp tác, liên doanh với các đối tác trong và ngoài nước có mong muốn ứng dụng CNTT và công nghệ mới vào sản xuất kinh doanh cũng như đời sống.
* Trung tâm Khoa học Dữ liệu (Data Science Lab)
  + Phát huy thế mạnh về toán của Đại học Quy Nhơn
  + Đào tạo chuyên sâu về Khoa học Dữ liệu và Trí tuệ Nhân tạo
  + Phát triển các giải pháp đột phá dựa trên Khoa học Dữ liệu và Trí tuệ Nhân tạo
  + Sau 4 năm, Data Science Lab đã đào tạo chuyên sâu cho đội ngũ 15 data scientists và phát triển được nhiều giải pháp mới về AI & Data Science
* Học viện Công nghệ (TMA Academy)
* Đào tạo nguồn nhân lực về công nghệ cao, kết hợp với Đại học Quy Nhơn và các trường ĐH trong khu vực để trở thành trung tâm cung cấp nhân lực CNTT chất lượng cao tại miền Trung
* Đào tạo chuyên sâu các công nghệ mới: AI, data science, IoT
* Hợp tác với các đại học trong và ngoài nước, tiếp nhận hàng ngàn sinh viên kiến tập, thực tập mỗi năm các ngành CNTT, Điện tử - Viễn thông, Toán.
* Trung tâm Nghiên cứu và Chuyển giao Công nghệ (R&D Center)
  + Nghiên cứu phát triển công nghệ 4.0 cho thị trường Việt Nam và thế giới
  + Hợp tác R&D với các trường đại học tại miền Trung nhằm mục đích chuyển giao và ứng dụng CNTT và các công nghệ mới vào công nghiệp, nông nghiệp, sản xuất cũng như đời sống tại các tỉnh miền Trung
  + Thu hút các nhà khoa học trong và ngoài nước để trở thành Trung tâm nghiên cứu và triển khai CNTT hàng đầu tại miền Trung

## Lĩnh vực hoạt động

* TMA Solutions hoạt động trong các lĩnh vực:
* Telecom;
* Finance & Insurance;
* E-Commerce;
* Retail & Distribution;
* ISV (Independent Software Vendor);
* Media & Publishing;
* Healthcare;
* Education;
* Hospitality & Travel;
* Logistics & Transportation.

# CHƯƠNG 2. GIỚI THIỆU VỀ FLUTTER

## I. Flutter là gì?

Flutter là một framework phát triển ứng dụng di động mã nguồn mở được phát triển bởi Google. Nó được sử dụng để xây dựng các ứng dụng di động cho iOS, Android, Web và các nền tảng khác từ một nguồn duy nhất. Flutter sử dụng ngôn ngữ lập trình Dart, được phát triển bởi Google, và nó có thể được sử dụng để tạo ra các ứng dụng di động với các tính năng đa dạng như đồ họa, điều hướng, quản lý trạng thái, v.v.

Flutter có các tính năng sau:

* Widget độc lập khỏi nền tảng: Flutter sử dụng các widget riêng để xây dựng giao diện người dùng, điều này cho phép các widget này độc lập với nền tảng và giúp dễ dàng tái sử dụng code.
* Hot Reload: Tính năng Hot Reload cho phép các nhà phát triển xem các thay đổi trực tiếp trong ứng dụng mà không cần phải khởi động lại ứng dụng.
* Performance cao: Flutter được xây dựng trên nền tảng đồ họa chéo nền tảng Skia, giúp cải thiện hiệu suất và tốc độ của ứng dụng.
* Hỗ trợ phát triển tương tác: Flutter cung cấp các thư viện hỗ trợ phát triển tương tác như animating, scrolling, routing, và gesture detection.
* Hỗ trợ đa nền tảng: Flutter cho phép phát triển ứng dụng cho nhiều nền tảng khác nhau như iOS, Android, Web, Windows, Mac, Linux, v.v.

Flutter là một lựa chọn hữu ích cho những nhà phát triển muốn xây dựng các ứng dụng di động chất lượng cao với tính năng đa dạng và hiệu suất tốt.

## II. Tổng quan về Flutter

### Dart là gì?

Dart là một ngôn ngữ lập trình đa nền tảng được phát triển bởi Google vào năm 2011. Nó được thiết kế để phát triển các ứng dụng web, mobile, desktop và IoT. Dart có cú pháp giống với các ngôn ngữ lập trình khác như Java hoặc C++, tuy nhiên, nó cũng hỗ trợ các tính năng đặc biệt như mixin, async/await, generics, optional typing, và isolate để hỗ trợ lập trình đa luồng.

Một số đặc điểm của Dart:

* Được thiết kế để tối ưu hóa hiệu năng, Dart sử dụng JIT (Just-in-Time) hoặc AOT (Ahead-of-Time) để biên dịch mã nguồn thành mã máy.
* Dart hỗ trợ việc lập trình hướng đối tượng và lập trình hàm, cùng với các tính năng đặc biệt như mixins, interfaces, abstract classes, extension methods, v.v.
* Dart cũng hỗ trợ tính năng async/await để lập trình đa luồng và xử lý các tác vụ bất đồng bộ.
* Dart đi kèm với một bộ thư viện chuẩn phong phú để hỗ trợ việc phát triển ứng dụng, bao gồm các thư viện cho việc xử lý chuỗi, đọc/ghi tệp tin, phát triển web, v.v.
* Dart cũng được sử dụng như là ngôn ngữ chính cho việc phát triển ứng dụng di động trên nền tảng Flutter, một framework phát triển ứng dụng di động đang trở nên ngày càng phổ biến.

Với sự phát triển và ứng dụng rộng rãi trên các nền tảng, Dart đang trở thành một ngôn ngữ lập trình quan trọng trong cộng đồng lập trình viên.

### Layout & Naviagtion là gì?

Layout, widget và components là ba khái niệm cơ bản trong phát triển ứng dụng di động, đặc biệt là trên nền tảng Android.

Layout là cách bố trí các thành phần giao diện trong một màn hình. Nó xác định cách thức sắp xếp các thành phần như text, hình ảnh, các button, input fields, và các controls khác để tạo nên một giao diện đẹp và dễ sử dụng.

Widget là các thành phần giao diện như button, textview, imageview, edittext, checkbox, và radiobutton. Chúng cung cấp các tính năng và chức năng cho người dùng để tương tác với ứng dụng, như hiển thị thông tin, nhập liệu, chọn lựa và thao tác.

Components là các thành phần của ứng dụng được sử dụng để thực hiện các chức năng cụ thể. Ví dụ, một component có thể là một đối tượng để xử lý dữ liệu, một đối tượng để kết nối với cơ sở dữ liệu, hoặc một đối tượng để quản lý các tác vụ đồng bộ và bất đồng bộ.

Trong phát triển ứng dụng di động trên nền tảng Android, các layout, widget và components đóng vai trò quan trọng trong việc tạo ra một ứng dụng có giao diện đẹp, tương tác người dùng tốt, và chức năng hoạt động ổn định. Chúng được sử dụng trong việc thiết kế và triển khai các màn hình, đối tượng và chức năng của ứng dụng.

### Widget, Lifecycle & Tabbar

Trong Flutter, các giao diện (UI) được tạo ra bằng cách sử dụng các widget. Widget trong Flutter là một khái niệm rất quan trọng và được sử dụng để tạo ra giao diện cho ứng dụng. Widget có thể là các thành phần giao diện đơn giản như Text, Button, hay là các thành phần giao diện phức tạp hơn như ListView, GridView, v.v.

Trong Flutter, có hai loại widget chính: Stateful và Stateless.

**Stateless Widget**:

* Stateless widget là một widget không thể thay đổi trạng thái (immutable).
* Khi một stateless widget được khởi tạo, nó sẽ trả về một giao diện dựa trên các thuộc tính và các tham số được truyền vào khi khởi tạo.
* Stateless widget thường được sử dụng cho những thành phần giao diện như Text, Icon, Image, Button, v.v.

**Stateful Widget**:

* Stateful widget là một widget có thể thay đổi trạng thái (mutable).
* Khi một stateful widget được khởi tạo, nó sẽ trả về một giao diện dựa trên các thuộc tính và các tham số được truyền vào khi khởi tạo.
* Tuy nhiên, nếu trạng thái của stateful widget thay đổi, nó sẽ cần phải tạo lại giao diện của nó để phản ánh các thay đổi này.
* Stateful widget thường được sử dụng cho những thành phần giao diện phức tạp hơn như ListView, GridView, Form, v.v.

Một stateful widget bao gồm hai lớp chính: StatefulWidget và State. Lớp StatefulWidget là nơi để định nghĩa các thuộc tính và các phương thức, trong khi lớp State là nơi để định nghĩa trạng thái và các phương thức để cập nhật trạng thái. Khi trạng thái của một stateful widget thay đổi, Flutter sẽ gọi phương thức setState() để cập nhật giao diện của widget.

**Lifecycle** trong Flutter là quá trình các widget hoạt động trong ứng dụng của bạn, bao gồm các giai đoạn:

* createState(): Được gọi mỗi khi một widget được tạo ra.
* initState(): Được gọi khi widget được thêm vào cây widget.
* build(): Được gọi khi widget cần được hiển thị.
* didUpdateWidget(): Được gọi khi các thuộc tính của widget được cập nhật.
* dispose(): Được gọi khi widget bị xóa khỏi cây widget.

**TabBar** là một widget trong Flutter cho phép bạn tạo ra một thanh tab, cho phép người dùng chuyển đổi giữa các tab khác nhau. TabBar bao gồm một danh sách các tab và một vùng hiển thị nội dung tương ứng với tab được chọn. Các tab trong TabBar có thể được định nghĩa bằng cách sử dụng TabBar.tabs, và nội dung của mỗi tab có thể được hiển thị bằng cách sử dụng TabBarView. TabBar thường được sử dụng trong ứng dụng có nhiều nội dung khác nhau, như ví dụ như ứng dụng tin tức, cửa hàng trực tuyến, v.v.

### State Management, Stream & Bloc là gì?

State management, stream, và bloc là các kỹ thuật được sử dụng trong Flutter để quản lý trạng thái của ứng dụng và cải thiện hiệu suất.

**State management**: Đây là quá trình quản lý trạng thái của ứng dụng. Khi ứng dụng thay đổi trạng thái, các widget trong ứng dụng cần được cập nhật để hiển thị đúng thông tin. Flutter có các kỹ thuật quản lý trạng thái như StatefulWidget, InheritedWidget, Provider, v.v. để giúp quản lý trạng thái ứng dụng hiệu quả hơn.

**Stream**: Stream là một khái niệm trong Dart và Flutter để xử lý dữ liệu trong các luồng. Các luồng đại diện cho một dòng dữ liệu liên tục có thể được xử lý và truyền qua các phương thức khác nhau. Trong Flutter, Stream thường được sử dụng để tải dữ liệu từ internet hoặc đọc dữ liệu từ cơ sở dữ liệu.

**Bloc:** Bloc (Business Logic Component) là một kỹ thuật quản lý trạng thái ứng dụng, cũng như xử lý sự kiện và đưa ra quyết định. Bloc được sử dụng để tách biệt phần giao diện người dùng và code business logic của ứng dụng. Bloc giúp cho ứng dụng có cấu trúc rõ ràng, dễ bảo trì và kiểm thử.

Các kỹ thuật này được sử dụng để quản lý trạng thái của ứng dụng và cải thiện hiệu suất của ứng dụng Flutter. Tùy thuộc vào ứng dụng cụ thể, các kỹ thuật này có thể được sử dụng độc lập hoặc kết hợp với nhau để đạt được kết quả tốt nhất.

### Scoped access & Provider là gì?

Scoped Access và Provider là hai kỹ thuật quản lý trạng thái trong Flutter được sử dụng để truyền dữ liệu giữa các widget và quản lý trạng thái của ứng dụng.

1. **Scoped Access:** Scoped Access là một thư viện được sử dụng để quản lý trạng thái trong ứng dụng Flutter. Nó cho phép bạn tạo các widget có phạm vi hạn chế để truyền dữ liệu giữa các widget con. Các widget con được phép truy cập vào các dữ liệu được chia sẻ trong phạm vi được chỉ định, nhưng không thể truy cập vào các dữ liệu bên ngoài phạm vi đó.
2. **Provider**: Provider là một thư viện được sử dụng để quản lý trạng thái của ứng dụng trong Flutter. Nó giúp cho việc truyền dữ liệu giữa các widget trở nên dễ dàng hơn và cho phép bạn quản lý trạng thái của ứng dụng hiệu quả hơn. Provider cho phép bạn tạo các widget được cung cấp với dữ liệu được chia sẻ trong toàn bộ ứng dụng và cập nhật dữ liệu đó khi cần thiết.

Cả Scoped Access và Provider đều được sử dụng để quản lý trạng thái trong ứng dụng Flutter, nhưng có sự khác biệt trong cách thức triển khai. Scoped Access giới hạn phạm vi truy cập dữ liệu, trong khi Provider cung cấp dữ liệu cho toàn bộ ứng dụng. Cách sử dụng tùy thuộc vào loại ứng dụng cụ thể và yêu cầu của nó.

### Mapbox Flutter là gì?

Mapbox Flutter là một thư viện mã nguồn mở được xây dựng trên framework Flutter, giúp chúng ta tích hợp các bản đồ và dịch vụ địa lý từ Mapbox vào ứng dụng Flutter của họ. Mapbox là một nền tảng dịch vụ địa lý mạnh mẽ, cung cấp các công cụ và tài nguyên cho việc hiển thị bản đồ tương tác, định vị, tìm kiếm địa điểm và nhiều tính năng địa lý khác.

Với thư viện Mapbox Flutter, nhà phát triển có thể tạo và tùy chỉnh các thành phần bản đồ như các marker, polyline, polygon và các layer địa lý khác. Thư viện cung cấp các widget sẵn có cho việc hiển thị bản đồ và tương tác với người dùng, cho phép điều chỉnh quy mô, quay, kéo và phóng to/ thu nhỏ bản đồ. Ngoài ra, Mapbox Flutter cũng hỗ trợ các tính năng như tìm kiếm địa điểm, định vị GPS và định tuyến.

Mapbox Flutter kết hợp sự mạnh mẽ của Flutter framework với các tính năng địa lý của Mapbox, cho phép chúng ta xây dựng các ứng dụng di động đa nền tảng với khả năng hiển thị bản đồ tương tác và tính năng địa lý phong phú. Thư viện này đã thu hút sự quan tâm và sử dụng rộng rãi trong cộng đồng Flutter để xây dựng các ứng dụng có liên quan đến địa điểm và bản đồ.

## III. Giới thiệu về App Mapbox Flutter

### App Mapbox Flutter là gì?

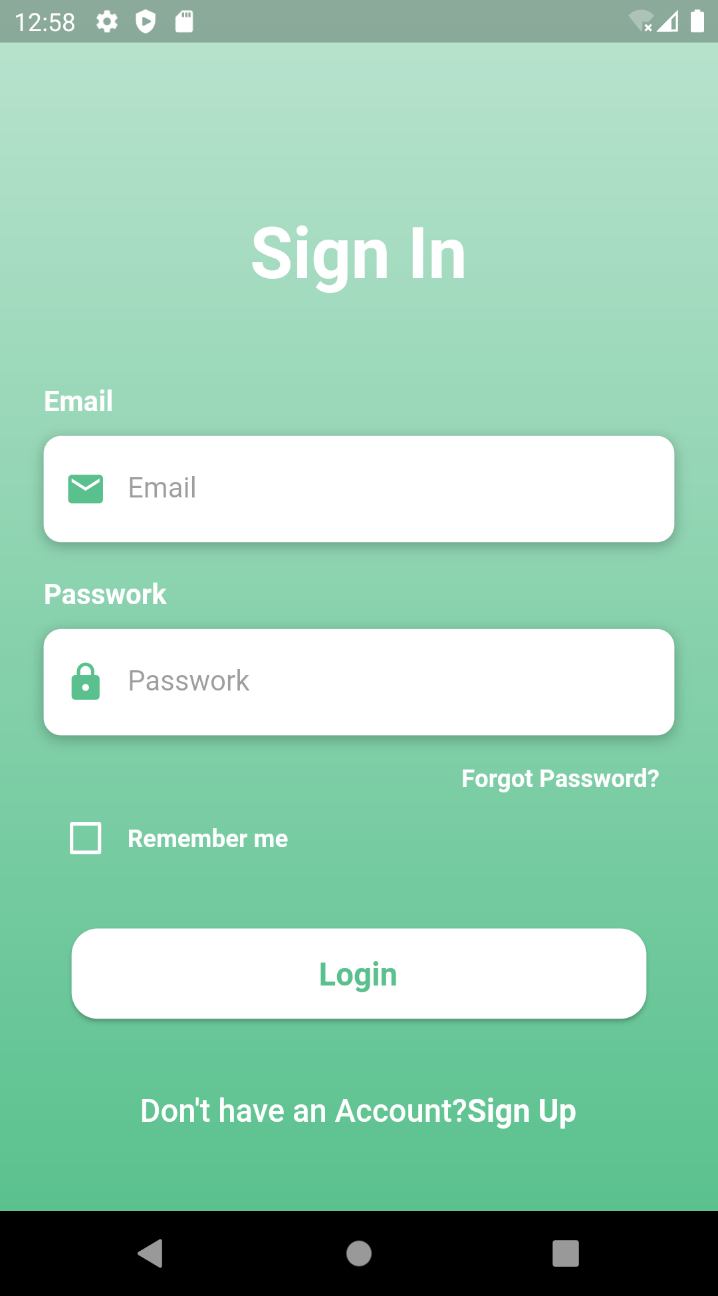
App Mapbox Flutter là một áp giúp cho người dùng có thể sử dụng để filter các tuyến đường đang sửa chữa, các tuyến đường đang được sửa chữa hoặc các tuyến đường mới được xây dựng.

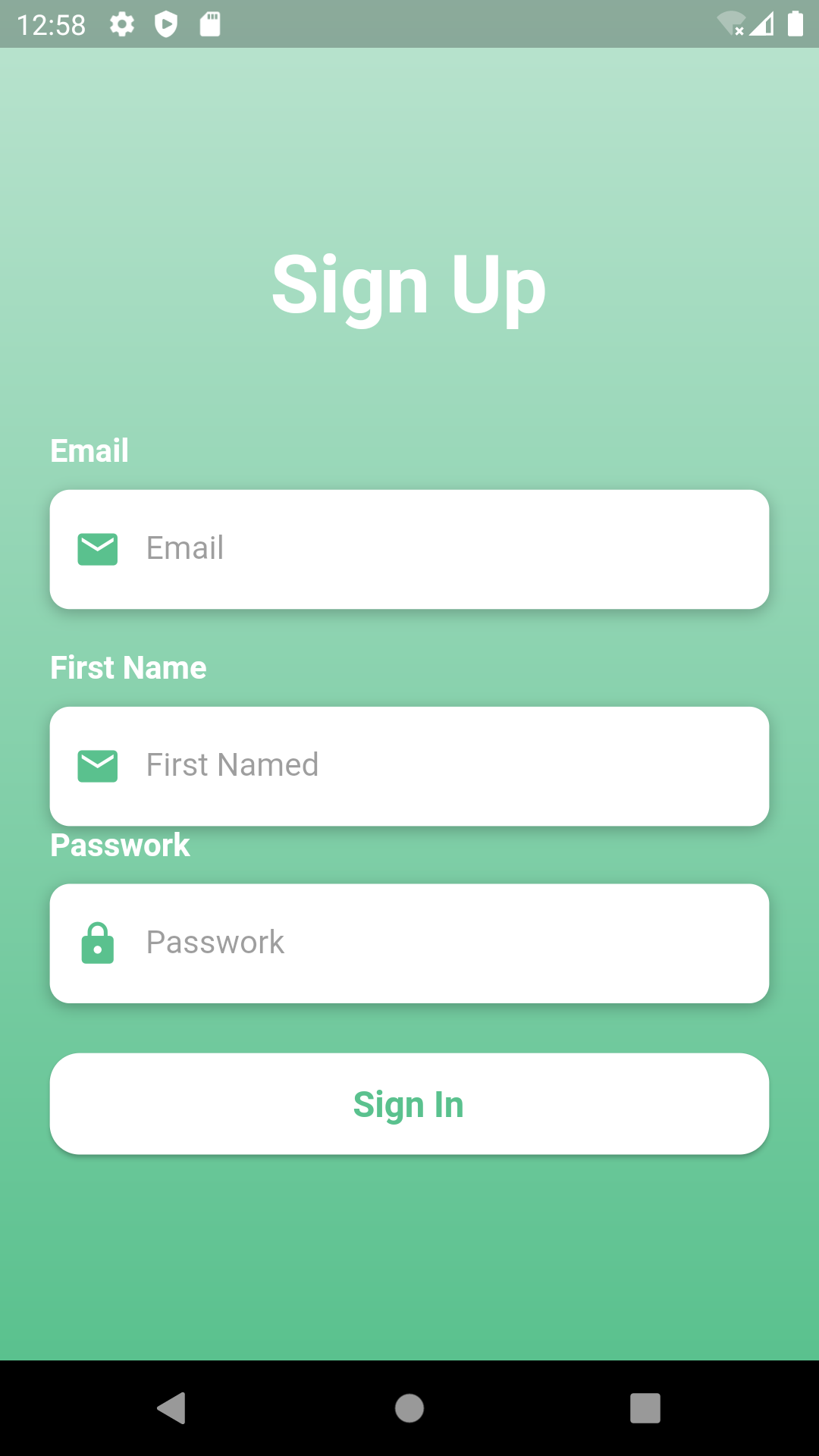
### Chức năng App Mapbox

* Kiểm tra đăng nhập.
* Tạo tài khoản mới.
* Lấy được thông tin đăng nhập và cập nhật vị trí người dùng.
* Phóng to và thu nhỏ bản đồ.
* Setting cập nhật số lượng confident và numberofpoint để người dùng tìm kiếm. Giúp cho người dùng có thể quản lý số lượng các tuyến đường do mình mong muốn.
* Tìm kiếm tuyến đường theo từng ngày.
* Filtering tìm kiếm các tuyến đường có chứa các từ khóa như manhole, peeling, pothole
* Filtering theo các từ khóa new, required, close
* New tìm kiếm các đoạn đường mới được xây dựng và cập nhật.
* Required tìm kiếm ra các đoạn đường đang cần được sửa chữa hoặc cần sửa chữa.
* Close các tuyến đường đang sửa hoặc đang mở rộng.

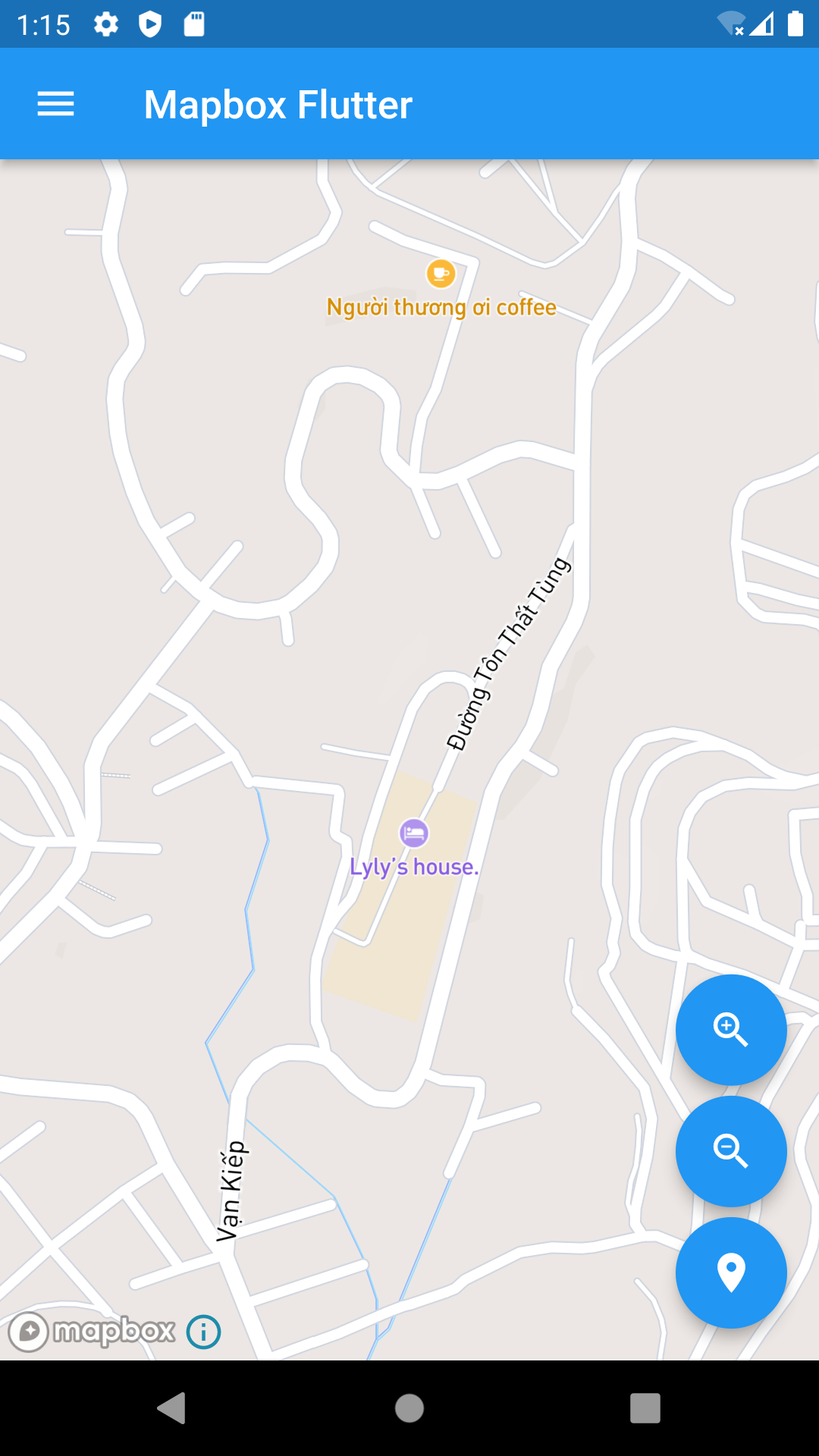
### Giao diện App MapBox Flutter?

#### **Login & Sign up**

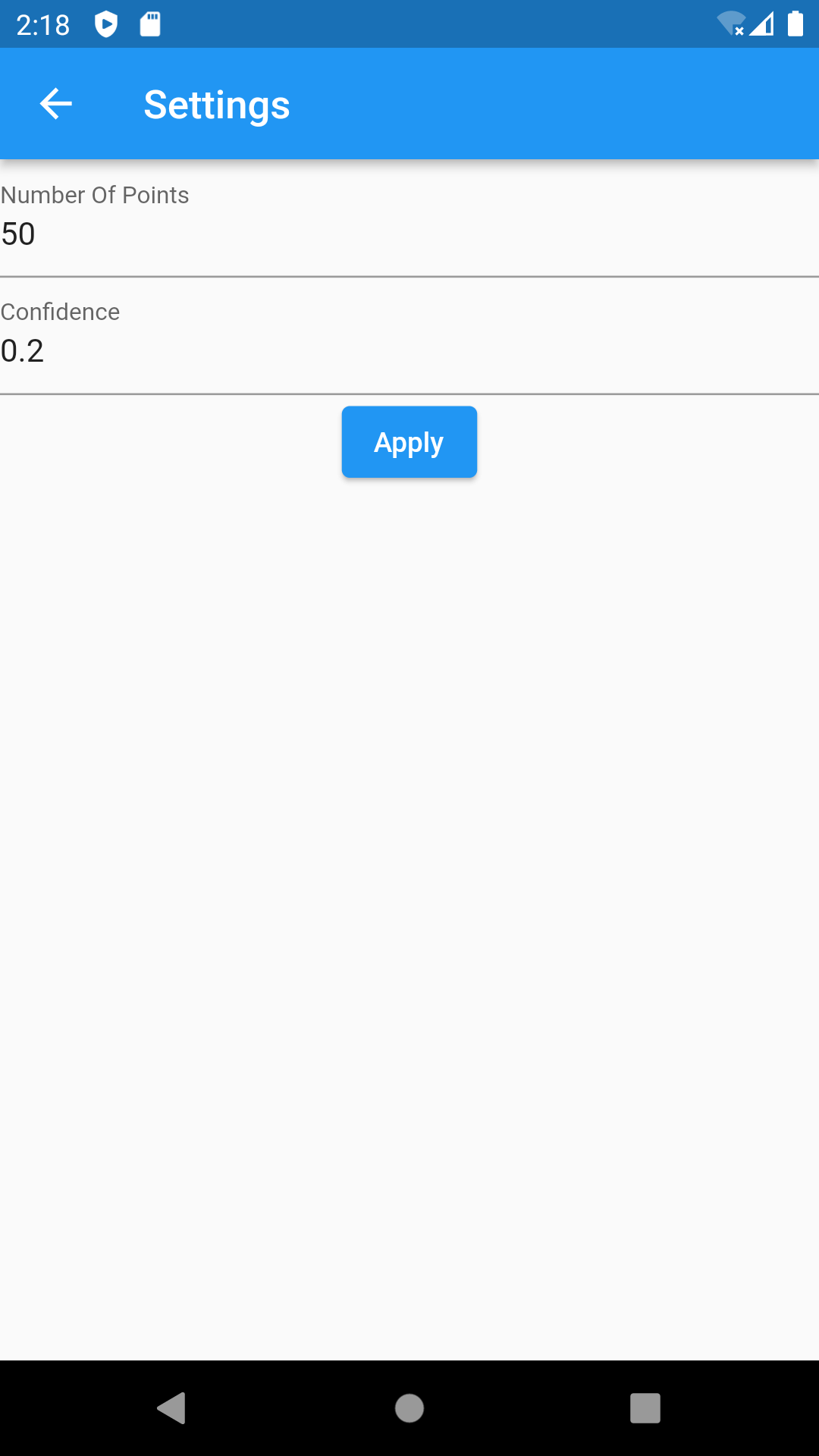




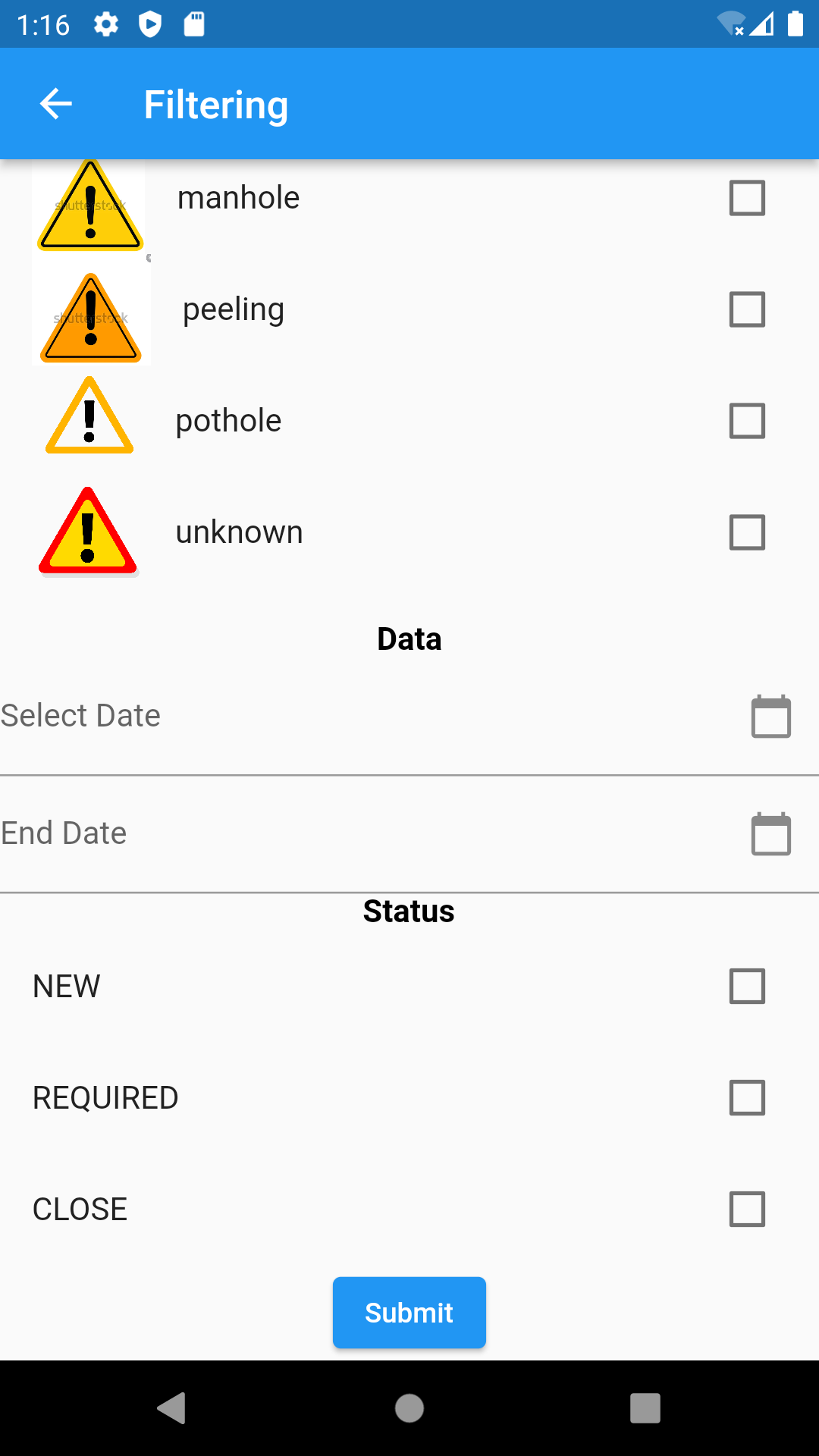
#### **Home Screen (giao diện chính)**



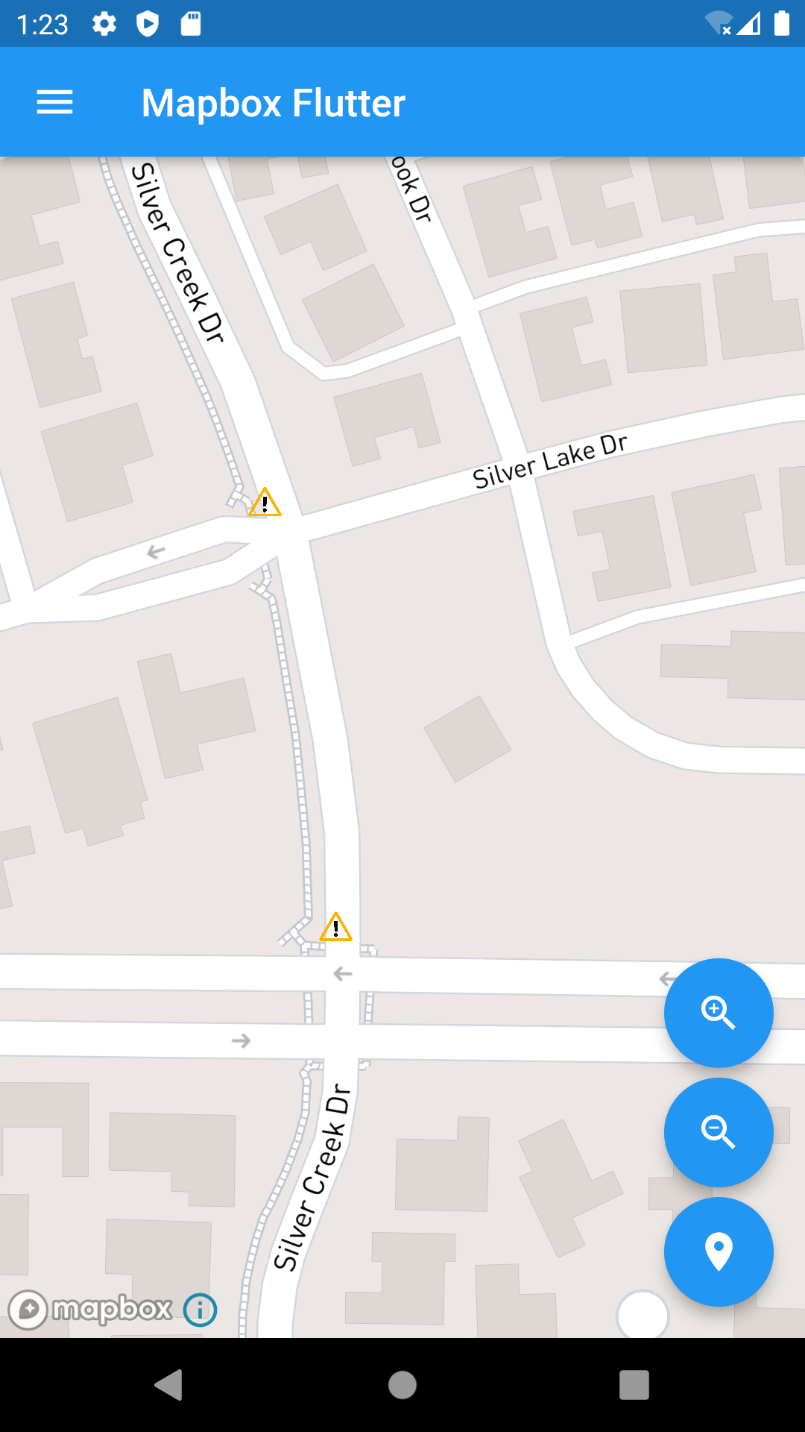
#### **Settings**



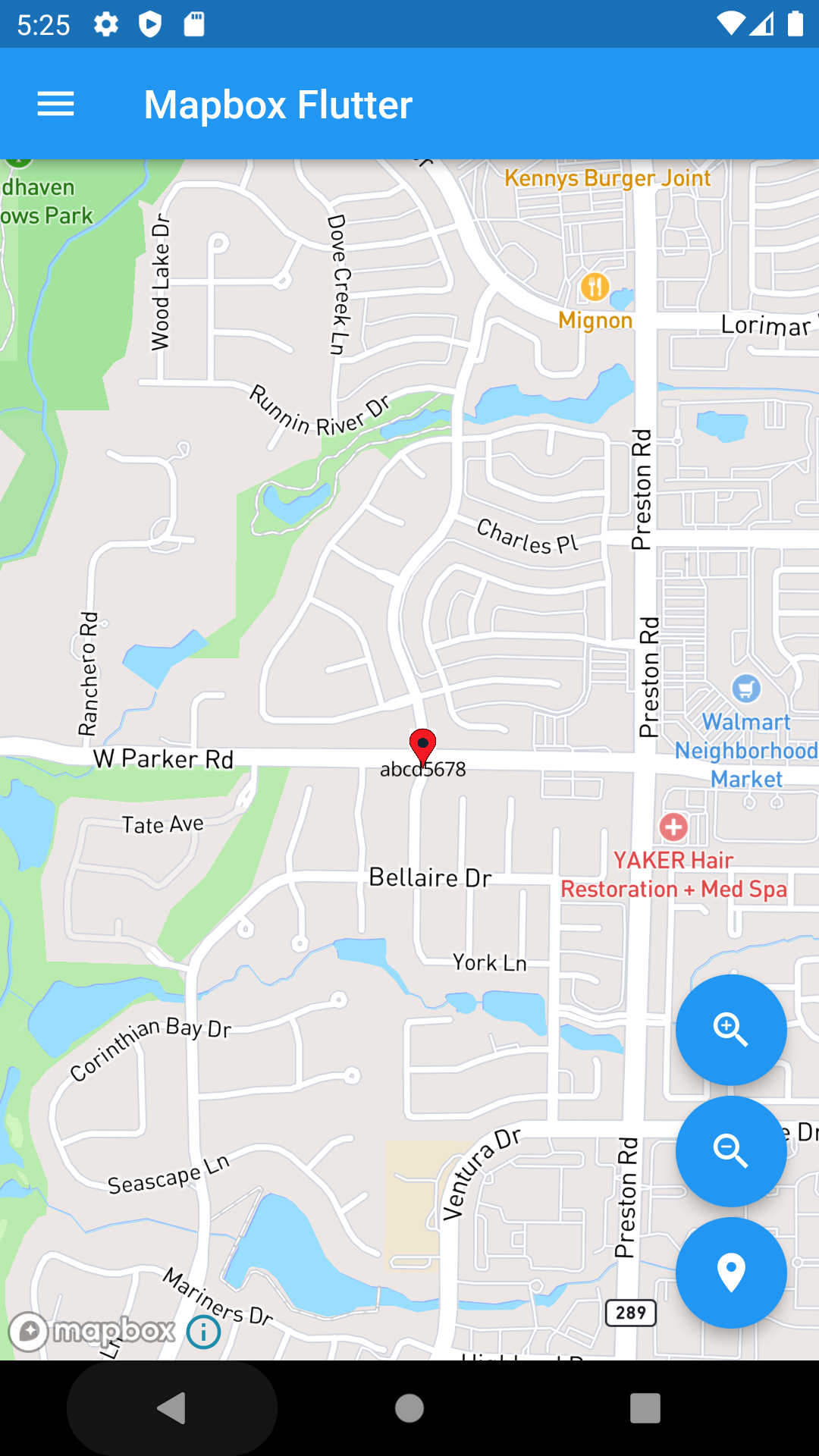
#### **Filtering symbol**



#### **Filtering symbol add symbol Mapbox Flutter**



#### **Location user**



# CHƯƠNG 3. NỘI DUNG THỰC TẬP

## Kế hoạch thực tập

Kế hoạch thực tập được chia làm 12 tuần. Bắt đầu từ ngày 09/03/2023 và kết thúc vào ngày 26/05/2023.

## Mục tiêu thực tập.

Học hỏi những kinh nghiệm thực tế trong công việc, áp dụng những kiến thức đã được học vào thực tế và có cơ hội tiếp xúc với các công nghệ, công cụ và quy trình làm việc thực tế. Phát triển kỹ năng và kỹ thuật mà còn phát triển thêm về các kỹ năng mềm như: giao tiếp, làm việc nhóm, quản lý thời gian, giải quyết vấn đề và thích ứng với môi trường làm việc.

## Nội dung thực tập

* Tìm hiểu và học về Flutter và Dart
* Tham gia làm Project
* Tham gia các hoạt động của công ty.
* Tập huấn các kĩ năng mềm.
* Tham gia lớp học ngoại ngữ
* Thực hiện đánh giá năng lực ngoại ngữ.
* Báo cáo tiến độ dự án mỗi ngày.
* Báo cáo tiến độ giai đoạn của dự án.
* Báo cáo dự án kết thúc thực tập
* Kiểm tran đầu vào tiếng Anh
* Tham gia khóa học kỹ năng mềm: Work Etiquette & Professional Communication.
* Tham gia đào tạo quản lý dự án: Agile methodology
* Tham gia khóa đào tạo kỹ năng mềm: Presentation skills
* Tham gia khóa đào tạo nhận thức an toàn thông tin: Security Awareness
* Tham gia quản lý và xây dụng dự án Mapbox Flutter.

## Tiến độ thực tập

* Tiến độ thực tập của các tuần đúng với kế hoạch đề ra.
* Các buổi đào tạo kĩ năng mềm, kĩ năng quản lý dự án, báo cáo tiến độ dự án đều đúng với kế hoạch.

# CHƯƠNG 4: GIẢI PHÁP THỰC HIỆN

Trong quá trình thực tập và thực hiện dự án cuối cùng(Mapbox Flutter) để hoàn thành khóa thực tập, em đã thực hiện và làm việc dựa trên phương pháp Agile thông qua mô hình Scrum.

Agile là một phương pháp quản lý dự án và phát triển phần mềm linh hoạt và tương tác. Nó tập trung vào việc phát triển sản phẩm một cách linh hoạt và tận dụng các sự thay đổi trong quá trình phát triển.

Các nguyên tắc cốt lõi của Agile bao gồm:

1. Tập trung vào sự tương tác và hợp tác giữa các thành viên trong nhóm phát triển.
2. Ưu tiên đáp ứng nhanh chóng và linh hoạt đối với sự thay đổi yêu cầu.
3. Phân chia công việc thành các đợt nhỏ hơn và phát triển theo từng chu kỳ ngắn gọi là "sprints".
4. Liên tục đánh giá, đánh giá và cải tiến quá trình phát triển.
5. Đặt giá trị cao vào việc cung cấp phần mềm có giá trị cho khách hàng và tạo sự hài lòng cho khách hàng.

Agile khác biệt với các phương pháp phát triển phần mềm truyền thống bằng việc đặt trọng tâm vào tương tác liên tục giữa nhóm phát triển và khách hàng, ưu tiên linh hoạt và phản hồi nhanh chóng đối với thay đổi. Agile cung cấp một cách tiếp cận linh hoạt và có khả năng thích ứng với môi trường phát triển phần mềm thay đổi và không chắc chắn.

Theo như em đã giải thích ở trên thì Agile là một phương pháp thực hiện khá phổ biến để quán lý dự án. Ở dự án của em Agile chỉ phát triển về phần phương pháp tách nhỏ dự án để quán lý và phân chia công việc.

# CHƯƠNG 5: KẾT QUẢ THỰC TẬP

## Mức độ hoàn thiện

|  |  |  |
| --- | --- | --- |
| STT | Nội dung công việc | Mức độ |
| 1 | Tìm hiểu về ngôn ngữ lập trình Dart | *Hoàn thành* |
| 2 | Tìm hiểu về các Flutter Basic | Hoàn thành |
| 3 | Tìm hiểu về Navigation | Hoàn thành |
| 4 | Tìm hiểu về các widget có trong flutter | Hoàn thành |
| 5 | Tìm hiểu States Management | Hoàn thành |
| 6 | Tìm hiểu về Bloc | Hoàn thành |
| 7 | Tìm hiểu Flutter Advanced | Hoàn thành |
| 8 | Xây dựng dự án cuối cùng | Hoàn thành |
| 9 | Reoport từng ngày làm việc | Hoàn thành |
| 10 | Báo cáo quá trình làm dự án | Hoàn thành |
| 11 | Báo cáo dự án kết thúc thực tập | Hoàn thành |
| 12 | Kiểm tra đầu vào tiếng Anh | Hoàn thành |
| 13 | Tham gia khóa đào nhận thức an toàn bảo mật thông tin: Security Awareness | Hoàn thành |
| 14 | Tham gia khóa đào tạo quản lý dự án: Agile methodology | Hoàn thành |
| 15 | Tham gia khóa đào tạo kỹ năng mềm: Presentation skills | Hoàn thành |
| 16 | Tham gia khóa học kỹ năng mềm: Work Etiquette & Professional Communication | Hoàn thành |
| 17 | Báo cáo thực tập | Hoàn thành |

## Kỹ năng làm việc thực tế

Trong quá trình làm việc thực tế tại công ty em cũng đã được học hỏi rất nhiều kỹ năng để làm việc như:

* Kỹ năng làm việc nhóm
* Kỹ năng quản lý dự án
* Kỹ năng giao tiếp
* Kỹ năng xử lý công việc
* Kỹ năng tìm kiếm và đọc tài liệu tiếng Anh
* Kỹ năng quản lý dự án thông qua Agile để phân chia thời gian phù hợp.

## Kinh nghiệm tích lỹ

* Chủ động học hỏi và tìm kiếm công nghệ mới phát huy khả năng học hỏi.
* Kinh nghiệm đọc và tìm tài liệu tiếng Anh
* Kinh nghiệm quản lý dự án qua Agile
* Kinh nghiệm làm việc nhóm

## Kết quả đạt được

* Nâng cao kỹ năng mềm khi tham gia các các buổi đào tạo tại TMA: Biết cách viết một email chuyên nghiệp, cách trình bày slide và thuyết trình trước đám đông, rèn luyện kỹ năng giao tiếp và kỹ năng làm việc nhóm.
* Cải thiện khả năng sử dụng tiếng Anh trong đọc tài liệu
* Nắm rõ quy trình phát triển phần mềm và kỹ năng quản lý dự án.
* Bổ sung và nâng cao kiến thức cần thiết chưa học qua.
* Học được một số cách xử lý lỗi.
* Áp dụng được những kiến thức đã được học tập tại trường vào đề tài thực tế.
* Rèn luyện cũng như học tập, tiếp xúc với môi trường làm việc thực tế.
* Học tập được tính tỉ mỉ và cẩn trọng trong công việc.

## Nhược điểm và thiếu sót

Trong quá trình thực tập, đôi khi chúng ta sẽ mắc phải những sai sót và thiếu sót, và điều đó không phải là điều hiếm gặp. Việc gặp lỗi và vấn đề trong quá trình thực tập là điều tất yếu để em có thể rèn luyện kỹ năng và học hỏi các kinh nghiệm giá trị.

Điều quan trọng là chúng ta phải học hỏi từ những sai sót đó và sửa chứa cho phù hợp. Việc này không chỉ giúp chúng ta nâng cao chất lượng công việc, mà còn cải thiện khả năng làm việc và phát triển đúng hướng.

## Hướng phát triển của dự án

Dự án còn nhiều vấn đề và còn nhiều chức năng chưa hoàn thiện, nó có thể em chứ nắm vững kiến thức để giải quyết vấn đề hoặc có quá nhiều vấm đề phát sinh em chưa kịp xử lí. Em sẽ tiếp tục tìm hiểu và học hỏi để tiếp tục phát triển dự án hoàn thiện nhất.

# DANH MỤC TỪ VIẾT TẮT

TMA Solutions - Công ty TNHH Giải Pháp Phần Mềm Tường Minh

TMA Innovation Park - Công viên sáng tạo TMA

CNTT - Công nghệ thông tin