

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**



**BÁO CÁO THỰC TẬP**

**Lập trình ứng dụng Android**

**Công ty thực tập : Horical.Co**

**Người phụ trách : Leader Nguyễn Trung Vinh**

**Thực tập sinh : Nguyễn Minh Hiếu**

**Tp.Hồ Chí Minh, tháng 5 năm 2018**

**LỜI MỞ ĐẦU**

Sự bùng nổ của các thiết bị di động, đặc biêt là các thiết bị “thông minh” trong khoảng 5 năm trở lại đây đã tạo ra những xu hướng công nghệ, những phương thức kinh doanh mới trên các nền tảng di động. Hiện nay, trên thế giới, ứng dụng công nghệ Mobility càng được phát triển ở nhiều nghành nghề khác nhau như hàng không, ngân hàng, bán lẻ, y tế … từ những việc đơn giản như bán hàng, theo dõi thông tin doanh nghiệp, đến những thứ phức tạp như truyền dẫn tín hiệu máy bay, xác định đường bay cho phi công cần độ chính xác rất cao hiện đều đã được mobile hóa…

Ngay tại Việt Nam, công nghệ Mobility đang là nhu cầu thiết yếu của các doanh nghiệp với tiềm năng rất lớn khi tốc độ phát triển và hội nhập của Việt Nam đang gia tăng mạnh mẽ cùng với việc nhu cầu mua sắm điện thoại thông minh tăng rất nhanh. Trong giai đoạn 2015 đến 2018, xu hướng sử dụng smartphone và lượng người dùng Internet 3G được dự báo sẽ tiếp tục tăng mạnh. Trong khi các dịch vụ kết nối OTT (Over-the-top) và truyền thông xã hội đóng góp tới hơn 80% phương thức giao tiếp online, video online và nội dung số mobile. Điều này góp phần đẩy mạnh xu hướng truyền thông số đa phương tiện, đa màn hình (PC, smartphone, tablet và Smart TV) bùng nổ trong thời gian tới với độ phủ hơn 50% dân số Việt Nam.

Về mảng Mobility thì đang có 2 ông lớn chiếm lĩnh trong nhiều năm trở lại đây đó là Android của Google và IOS của Apple. Phần vì đã có kiến thức nền tảng Java ngay trên ghế nhà trường, phần vì Android không yêu cầu cao về thiết bị như IOS, do đó, em quyết định chọn lập trình android làm định hướng cho việc học tập và công việc tương lai của mình.

Sau ba năm học tập trên trường, do mong muốn có thêm kinh nghiệm thực tế, cũng như muốn được tham gia làm android trong một môi trường chuyên nghiệp, em có dự định là sẽ thực tập trong kì này. Vì vậy, em quyết định chọn Horical - một môi trường startUp trẻ năng động nhưng không kém phần chuyên nghiệp - là nơi sẽ giúp em thực hiện được dự định này.

**LỜI CẢM ƠN**

Trân trọng gửi lời cảm ơn Công ty Horical và hơn hết là anh giám đốc Nguyễn Văn Toàn đã tạo điều kiện cho em có cơ hội được thực tập tại công ty.

Chỉ trong một thời gian ngắn trong 3 tháng nhưng nhóm trainer đã hướng dẫn nhiệt tình giúp em cùng các bạn thực tập sinh khác có thêm nhiều kiến thức, kinh nghiệm bổ ích mà tụi em chưa có được trước khi tới công ty. Đã giúp em hòa mình vào được với các bậc đàn anh đi trước, cùng nhau chia sẻ các nhiệm vụ mà công ty giao và hơn hết là lần đầu trải nghiệm làm một lập trình viên Android thứ thiệt, làm dự án thật hẳn hoi.

Đặc biệt cảm ơn anh *Nguyễn Trung Vinh*, đã training Android, hướng dẫn, giúp đỡ cho chúng em tận tình cả những khó khăn trong công việc, đến những khó khăn việc làm quen với môi trường mới, đã chỉ dẫn chúng em về cách làm báo cáo, lên kế hoạch, những kỹ năng không thể thiếu; cảm ơn các anh *Võ Thành Nhơn* anh *Phan Vinh Quang* đã training cho chúng em những kiến thức quan trọng về core android, cách sử dụng Framework của công ty trong suốt thời gian thực tập tại công ty.

Cũng xin cảm ơn thầy cô trong khoa Công nghệ phần mềm đã nhiệt tình hỗ trợ, tạo điều kiện em làm bài báo cáo này.

Nguyễn Minh Hiếu

Tp.Hồ Chí Minh, tháng 5 năm 2018

**NHẬN XÉT CỦA KHOA**

**MỤC LỤC**

[I. GIỚI THIỆU CÔNG TY THỰC TẬP 1](#_Toc514514679)

[1. Giới thiệu về công ty Horical 1](#_Toc514514680)

[2. Sản phẩm của Horical 2](#_Toc514514681)

[II. NỘI DUNG THỰC TẬP 2](#_Toc514514682)

[1. Tìm hiểu cơ chế vận hành công ty và các chuẩn mực liên quan 2](#_Toc514514683)

[2. Nghiên cứu kĩ thuật và tham gia dự án Gito 3](#_Toc514514684)

[2.1 Cài đặt, làm quen các công cụ làm việc 3](#_Toc514514685)

[2.2 Nghiên cứu tài liệu và những kĩ thuật cần chú ý trước khi đọc qua Framework của Horical 3](#_Toc514514686)

[2.3 Đọc Base Framework Gito của công ty 5](#_Toc514514687)

[2.4 Xây dựng chức năng Profit cho Gito 5](#_Toc514514688)

[2.5 Đọc hiểu Base Framework mới của công ty 6](#_Toc514514689)

[2.6 Xây dựng ứng dụng Gito Testcase cho công ty 6](#_Toc514514690)

[2.7 Nghiên cứu công nghệ nâng cao về Android 7](#_Toc514514691)

[3. Lịch làm việc 10](#_Toc514514692)

[III. CHI TIẾT VỀ PROJECT 11](#_Toc514514693)

[1. Giới thiệu về Gito 11](#_Toc514514694)

[2. Giới thiệu chức năng Profit 12](#_Toc514514695)

[3. Thực hiện 16](#_Toc514514696)

[4. Kế hoạch chi tiết 16](#_Toc514514697)

[IV. TỔNG KẾT 18](#_Toc514514698)

[V. TÀI LIỆU THAM KHẢO 19](#_Toc514514699)

[1. Tổng quan 19](#_Toc514514700)

[2. Link chi tiết 19](#_Toc514514701)

# **GIỚI THIỆU CÔNG TY THỰC TẬP**

## Giới thiệu về công ty Horical



Horical là công ty trách nhiệm hữu hạn một thành viên do anh Nguyễn Văn Toàn giám đốc công ty thành lập vào ngày 5/6/2015. Tọa lạc ngay P207,Tòa nhà A, Số 1 Đường số 1, Khu Công Nghệ Phần Mềm Đại, Phường Linh Trung, Quận Thủ Đức, TP Hồ Chí Minh. Là một startup trẻ năng động chuyên nghiên cứu, tích hợp và phát triển các giải pháp và công nghệ để tạo ra các sản phẩm phần cứng thông minh.

Đội ngũ công ty gồm hơn 40 nhân sự, trong đó có 24 nhân viên chính thức, là những sernior có kinh nghiệm, số còn lại là các bạn sinh viên mới ra trường và thực tập sinh có khao khát ham học hỏi cao chủ yếu là sinh viên Đại học Công Nghệ Thông Tin và sinh viên học viện Bưu Chính Viễn Thông.



Trải qua gần 3 năm tuổi đời, công ty đã có những thăm trầm nhất định. Tuy nhiên vị thế và tầm vóc vẫn đang phát triển theo từng ngày.

## Sản phẩm của Horical

Horical hoạt động trên 2 hình thức đó là: nhận outsourcing từ phía khách hàng. Chủ yếu tập trung các nguồn hàng từ châu âu. Hiện tại công ty đang nhận làm labo cho 2 sản phẩm đó là hệ thống sửa chữa xe oto tại gara : MTRX (mobiletechRx.com) và phần mềm hỗ trợ cuộc thi chạy tình nguyện: CM (charitymile.com).

Hình thức thứ 2 và cũng là hướng phát triển chiến lược đó là tạo ra những sản phẩm Product của riêng công ty. Và đứa con đầu tiên đó là phần mềm Gito (GitoSolutions.com) một phần mềm hỗ trợ các công ty quản lí toàn bộ hoạt động của công ty từ nhân viên tới lợi nhuận và nhiều hơn thế nữa.

# **NỘI DUNG THỰC TẬP**

Đợt thực tập “Lập trình với ứng dụng Android” nhằm mục đích giúp sinh viên được đào tạo toàn diện về mảng Android qua các dự án thực của Horical. Đồng thời rèn luyện cho thực tập sinh những kĩ năng mềm như làm việc nhóm, thuyết trình, giao tiếp. Tại Horical, sinh viên có cơ hội trao dồi kinh nghiệm với những người có kinh nghiệm lâu năm trong nghề và làm việc trong một môi trường phát triển Starup chuyên nghiệp.

## Tìm hiểu cơ chế vận hành công ty và các chuẩn mực liên quan

***Thời gian:*** 3 ngày

***Nội dung***: Được nghe giới thiệu về công ty, cách tổ chức bộ máy của công ty.

* Được nghe anh Toàn giám đốc công ty giới thiệu , quá trình thành lập và phát triển Horical. Sau đó là nghe giới thiệu về bộ máy công ty.
* Buổi chiều là được giới thiệu những chuẩn mực làm việc như điểm danh, chế độ ăn trưa, ngủ nghỉ tại công ty.

***Kết quả:*** Hiểu thêm về cách tổ chức của một công ty là như thế nào. Có thêm các kĩ năng làm việc như gửi mail, cách hỏi và đặt vấn đề, …giúp cho em tự tin hơn trong những thời gian làm viêc sau này.

## Nghiên cứu kĩ thuật và tham gia dự án Gito

### Cài đặt, làm quen các công cụ làm việc

***Thời gian:*** 3 ngày

***Nội dung***: Được hướng dẫn cài đặt và sử dụng các công cụ mà công ty dùng để làm việc nhóm, chia sẻ Source Code nội bộ, Đọc hiểu Design và lấy các thông số liên quan.

* Kênh thông tin làm việc nhóm : Skype.
* Chia sẻ Source Code : Github, Git.
* Đọc thiết kế: Figma
* Xem thông tin cập nhật thiết kế: Google Drive
* Phần mềm code: Android Studio là IDE phổ biến cho lập trình viên Android với nhiều tính năng hỗ trợ từ Google
* Phần mềm xem API: Postman
* Đọc thêm các tài liệu liên quan đến quy chuẩn code, quy tắc đặt tên biến của công ty

***Kết quả:*** Hiểu thêm về những phần mềm liên quan tới công việc của công ty. Biết thêm những quy tắc lập trình, cách chia sẻ code của mình với team, cách đọc một bản design hay xem 1 API và ứng dụng nó như thế nào.

### Nghiên cứu tài liệu và những kĩ thuật cần chú ý trước khi đọc qua Framework của Horical

***Thời gian:*** 1 tuần

***Nội dung***: Được anh Nguyễn Trung Vinh chia sẻ những link tài liệu liên quan về những kĩ thuật quan trọng khi làm dự án tại công ty, sau đó nghiên cứu và đưa ra thắc mắc nếu có, những tài liệu được nhắc tới bao gồm:

* MVP Framework Android :

<https://antonioleiva.com/mvp-android/>

MVP (Model – View - Presenter) là một mô hình dẫn xuất từ mô hình MVC (Model – View - Controler). Là mô hình kiến trúc hướng giao diện người dùng, được thiết kế để tạo điều kiện thuận lợi cho việc kiểm thử đơn vị Unit Test, có sự tách biết giữa Model và View trong MVC. Tất cả logic dữ liệu đều được đưa vào Presenter. Trong MVP, Model xử lí dữ liệu, View hiển thị dữ liệu và tiếp nhận tương tác từ người dùng, còn Presenter là cầu nối giữa Model và View (Tiếp nhận tương tác từ View và cập nhật lại Model).

* Dagger 2: <http://www.vogella.com/tutorials/Dagger/article.html>

Dagger 2 API (một số annotation đặc biệt)

- @Scope: annotation định nghĩa vòng đời tồn tại của Object graph, nó thật sự hữu ích trong việc quản lý vòng đời của các Service sẽ được cung cấp cho Client. Chúng ta có thể define các loại Scope khác nhau như một annotation mới và sử dụng trong từng trường hợp cụ thể

- [@Module](https://viblo.asia/u/Module): cho các lớp mà có phương thức cung cấp sự phụ thuộc.

- [@Providers](https://viblo.asia/u/Providers): cho các phương thức bên trong các lớp [@Module](https://viblo.asia/u/Module).

- @Singleton: được tạo ra 1 lần và đươc sử dụng trong toàn ứng dụng

- [@Inject](https://viblo.asia/u/Inject): để yêu cầu một sự phụ thuộc ( một hàm khởi tạo, một trường, hoặc một phương thức).

- [@Component](https://viblo.asia/u/Component): là cầu nối interface giữa các modules và injection.

- @Qualifier: được cung cấp bởi javax inject package, và được dùng để xác định rõ dependency. Ví dụ, một class có thể yêu cầu cả application context và activity context, nhưng cả 2 object này đều trả về kiểu Context, do đó Dagger2 sẽ cấu hình việc biến nào sẽ provide cho cái nào. @Qualifier được dùng nhiều để phân biệt giữa các object cùng loại khác nhau (như activity context và application context)

- @Name: tương tự @Qualifier. Ngoài ra cung cấp String để đặt tên cho các class object cùng loại.

* Retrofit :

<http://square.github.io/retrofit/>

[Retrofit](https://square.github.io/retrofit/) là một HTTP client type-safe cho Android và Java.

Retrofit giúp dễ dàng kết nối đến một dịch vụ REST trên web bằng cách chyển đổi API thành Java Interface.

Tất cả các yêu cầu GET, POST, PUT, PATCH, và DELETE đều có thể được thực thi.

Tại sao nên sử dụng Retrofit?

Giảm bớt việc xử lý chức năng như: tạo các kết nối, lưu trữ cache, thử lại cache, phân tích luồng phản hồi, xử lý lỗi,…

* Glide Android :

<https://github.com/bumptech/glide>

Glide là một thư viện open source hỗ trợ load ảnh trên Android. Dùng Glide sẽ đơn giản hóa các công việc mà bạn cần làm khi sử dụng một bức ảnh trong Android đi rất nhiều. Chúng ta không cần quan tâm đến việc decoding, memory and disk caching mà thay vào đó chỉ cần sử dụng interface rất đơn giản từ Glide.

***Kết quả:*** Đọc hiểu tài liệu và tạo ra những ứng dụng nhỏ để demo về kĩ thuật mà mình nghiên cứu. Hỗ trợ cho việc đọc code chính thức trong thời gian tới.

### Đọc Base Framework Gito của công ty

***Thời gian:*** 3 ngày

***Nội dung***: Được đọc Base Framework của phần mềm Gito, product của công ty đang làm. Kĩ thuật base được xây dựng bởi anh Nguyễn Trung Vinh (Team Leader Android). Trong đó bao gồm sườn mô hình MVP và màn hình Login làm màn hình mẫu.Sau đó tự mình làm một màn hình nhỏ có tích hợp mô hình MVP (Ví dụ màn hình đăng kí tài khoản)

***Kết quả:*** Biết được thêm một luồng xử lí khoa học cho việc xây dựng một ứng dụng Android là như thế nào. Hiểu thêm về công việc của các thành viên trong team.

### Xây dựng chức năng Profit cho Gito

***Thời gian:*** ~1 tháng

***Nội dung***: Trong 1 tháng xây dựng, các công việc đã thực hiện được đó là :

* Tham gia vào source code Gito trên Github.
* Tham gia vào hệ thống tài liệu Gito trên Google Drive.
* Đọc qua thiết kế để hiểu thêm về chức năng được giao, chức năng Profit là một chức năng bổ sung của Gito nhằm thống kế con số lợi nhuận của công ty theo từng năm. Bên cạnh đó là hiển thị trực quan bằng biểu đồ
* Tiến hành xây dựng giao diện XML, có sử dụng một số CustomView của công ty.
* Tiến hành xử lí logic thông qua gọi API từ phía Server, sử dụng Retrofit 2.
* Bảo trì và Fix Bug.

***Kết quả:*** Tự mình xây dựng được 1 module nhỏ từ Base Framework của công ty. Qua đó nâng cao kĩ năng lập trình và bổ sung thêm cho mình những kinh nghiệm làm việc với dự án thực tế. Hiểu thêm về quy trình phát triển phần mềm.

### Đọc hiểu Base Framework mới của công ty

***Thời gian:*** 1 tuần

***Nội dung***: Framework mới này do anh Phan Vinh Quang nghiên cứu và xây dựng nhằm cải tiến việc bắt sự kiện cho view:

* Đọc hiểu tài liệu về Framework

Framework này khá mới lạ với những người chuyên code android theo hướng thuần. Chúng ta sẽ phải thường xuyên gói những view của mình thành những Component nhỏ rồi có thể mang chúng qua các View khác để xài lại nếu muốn. Sau đó sẽ đưa chúng lên tầng trên để hiển thị ra cho người dùng. Công việc nghe có vẻ rườm rà nhưng lại giúp lập trình viên dễ dàng quản lí việc bắn các sự kiện ra bên ngoài.

* Nắm luồng xử lí

Luồng xử lí sẽ đi 2 chiều vừa từ dưới lên và vửa từ trên cũng có thể bắn sự kiện về lại các Component tạo thành nó.

* Xây dựng demo nhỏ

***Kết quả:*** Hiểu thêm một Base Framework mới, hỗ trợ cho công việc sắp tới. Nâng cao kĩ năng thích ứng với công nghệ mới.

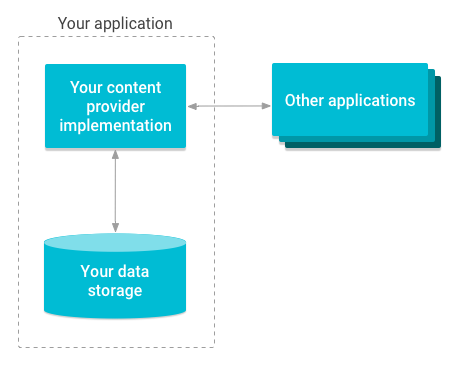
### Xây dựng ứng dụng Gito Testcase cho công ty

***Thời gian:*** ~1 tháng

***Nội dung***: Tiến hành tìm hiểu và xây dựng ứng dụng Gito Testcase, nội dụng công việc bao gồm:

* Đọc, hiểu design ứng dụng
* Nắm luồng xử lí
* Xây dựng giao diện XML cho ứng dụng
* Tự mình xây dựng những Customview (tham khảo từ App Gito) cho ứng dụng Gito Testcase
* Xử lí logic thông qua việc gọi API từ phía Server
* Bảo trì và Fix bug
* Liên kết từ app cha đó là Gito

Chúng ta sẽ sử dụng một kĩ thuật bự trong Android đó là Content Provider. Content provider là một thành phần để quản lý truy cập dữ liệu, nó cung cấp các phương thức khác nhau để các ứng dụng có thể truy cập dữ liệu từ một ứng dụng khác bằng cách sử dụng ContentResolver. Content Provider có thể giúp cho một ứng dụng quản lý quyền truy cập đến dữ liệu được lưu bởi ứng dụng đó, hoặc các ứng dụng khác, và đó là một cách để ta có thể chia sẻ dữ liệu cho các ứng dụng khác nhau. Hình dưới đây biểu diễn cho việc cách content providers quản lý việc truy cập tới bộ nhớ:



***Kết quả:*** Nâng cao kĩ năng lập trình với một framework mới. Tăng khả năng làm việc nhóm, phân tích yêu cầu.

### Nghiên cứu công nghệ nâng cao về Android

***Thời gian:*** 3 ngày

***Nội dung***: Do khối lượng công việc cũng đã giảm nên anh team leader Nguyễn Trung Vinh phân công cho các thành viên trong team tự nghiên cứu một vấn đề được giao về Android rồi sau đó thuyết trình lại cho các thành viên khác trong team. Và vấn đề mà em được giao đó là:

* Service trong Android

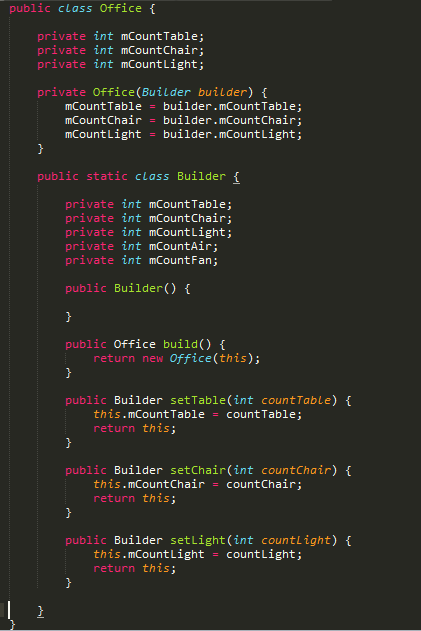
Service là 1 trong bốn component cơ bản của android(services, activities, content providers, broadcast receivers). Service chạy ẩn ở dưới để thực hiện các thao tác mà không cần tương tác với người dùng. Ví dụ, service có thể mở một bản nhạc trong khi người dùng đang sử dụng ứng dụng khác, hoặc là tải dữ liệu thông qua mạng mà không ảnh hưởng gì đến các hoạt động của người sử dụng. Một service sẽ có 2 loại:

* *Started*: Một service được bắt đầu giống như component khác như activities, bắt đầu service bằng cách gọi hàm  startService(). Thông thường, 1 "started service" thực hiện một hành động đơn lẻ và không trả về kết quả cho đối tượng gọi. Ví dụ, nó có thể thực hiện tải hoặc upload file thông qua mạng và khi thực hiện xong thì nó tự đông dừng lại.
* *Bound*: Một service được ràng buộc khi các component ràng buộc thông qua gọi hàm bindService(). Service ràng buộc thường là kiểu giao diện client-server, nó cho phép components tương tác với service, gửi yêu cầu, nhận kết quả trả về. Một service ràng buộc có thể chạy với nhiều components ràng buộc đến nó khi chaỵ lần đầu tiên, khi tất cả components của chúng không còn ràng buộc nữa thì service sẽ bị hệ thống hủy.

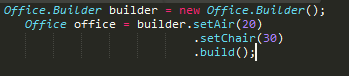


* Design Pattern – Builder
* *Định nghĩa*
* Builder Pattern là một pattern thuộc nhóm khởi tạo (creation patterns) cho phép tạo ra đối tượng với nhiều thể hiện khác nhau, giúp cho quá trình khởi tạo đối tượng đơn giản hơn.
* *Ưu điểm*
* Cung cấp thêm một cách khởi tạo đối tượng
* Hỗ trợ, loại bớt việc phải viết nhiều constructor
* *Hạn chế*
* Phải tạo builder cho từng class khác nhau.
* *Ví dụ*

Giả sử bạn cần xây dựng một văn phòng làm việc (Office) thì chúng ta cần khai báo:



Sau đó chúng ta có thể khởi tạo 1 đối tượng Office bằng cách:



***Kết quả:*** Có thêm những kiến thức về chủ đề được giao tìm hiểu. Tăng kĩ năng thuyết trình, tự tin trong giao tiếp.

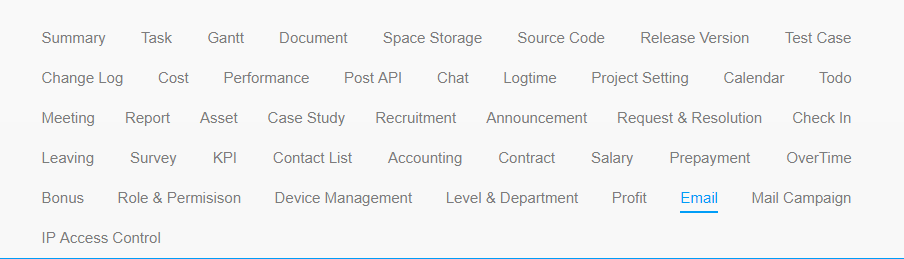
## Lịch làm việc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tuần** | **Công việc** | **Người hướng dẫn** | **Mức độ hoàn thành** | **Nhận xét của người hướng dẫn** |
| 1 | * Tìm hiểu về công ty, cách tổ chức của công ty. * Làm quen với các công cụ làm việc trong công ty. * Học cách trao đổi, làm việc nhóm. | - Anh *Nguyễn Văn Toàn*  - Chị *Chu Thị Tâm* |  |  |
| 2 | * Nghiên cứu tài liệu và những kĩ thuật cần chú ý trước khi đọc qua Framework của Horical | - Anh *Nguyễn Trung Vinh* |  |  |
| 3 | * Đọc Base Framework Gito của công ty | - Anh *Nguyễn Trung Vinh*  - Anh *Võ Thành Nhơn* |  |  |
| 4 -> 8 | * Xây dựng chức năng Profit cho Gito | - Anh *Nguyễn Trung Vinh*  - Anh *Nguyễn Văn Nam*  - Anh *Võ Thành Nhơn* |  |  |
| 5 | * Đọc hiểu Base Framework mới của công ty | - Anh *Phan Vinh Quang* |  |  |
| 6 -> 10 | * Xây dựng ứng dụng Gito Testcase cho công ty | - Anh *Nguyễn Trung Vinh*  - Anh *Phan Vinh Quang* |  |  |
| 11 | * Nghiên cứu công nghệ nâng cao về Android | - Anh *Nguyễn Trung Vinh* |  |  |

# **CHI TIẾT VỀ PROJECT**

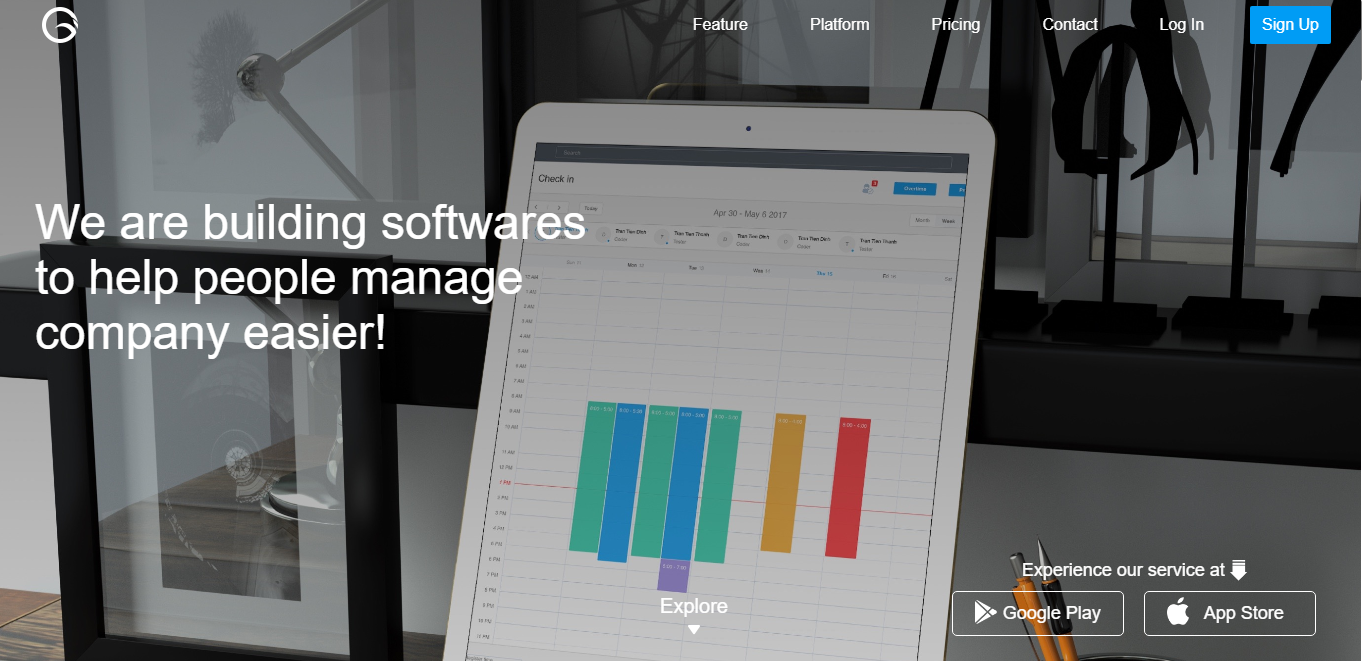
## 1. Giới thiệu về Gito

Gito là một Product bự của công ty Horical. Bắt đầu tiến hành xây dựng từ năm 2016. Ý tưởng xuất phát từ nhu cầu công nghệ thông tin hóa việc quản lý, điều hành công ty thay vì trên giấy tờ. Gito có rất rất nhiều chức năng lớn nhỏ hỗ trợ cho việc này.



*Danh sách các chức năng mà Gito mang lại*

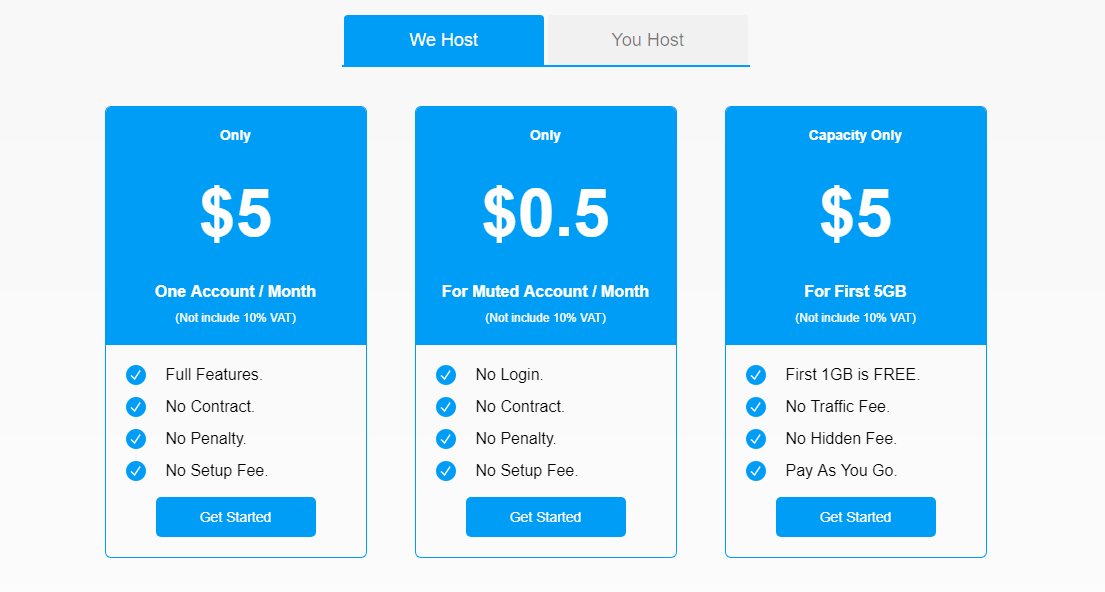
- Trang chủ: <https://gitosolutions.com/home/>



Bênh cạnh app Gito chính, Horical còn cho ra mắt các sản phẩm sắm vai phụ đó chính là : Gito Testcase, Gito Chat, Gito Mail

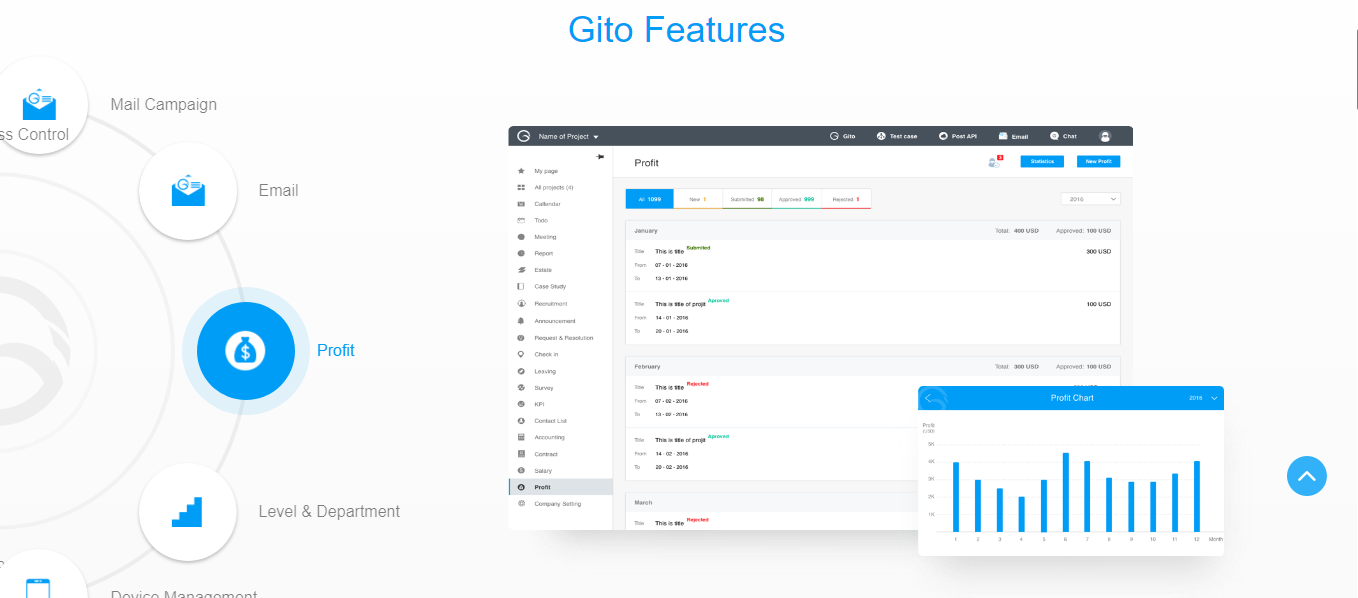
Mới đây, tháng 5/2018, Gito vừa cho ra mắt phiên bản 1.3 trên hệ thống cửa hàng Google Play và 1.1 trên hệ thống App Store.

Để có thể sử dụng được Gito cho công ty, chúng ta cần phải thuê host và server, mức giá như sau:



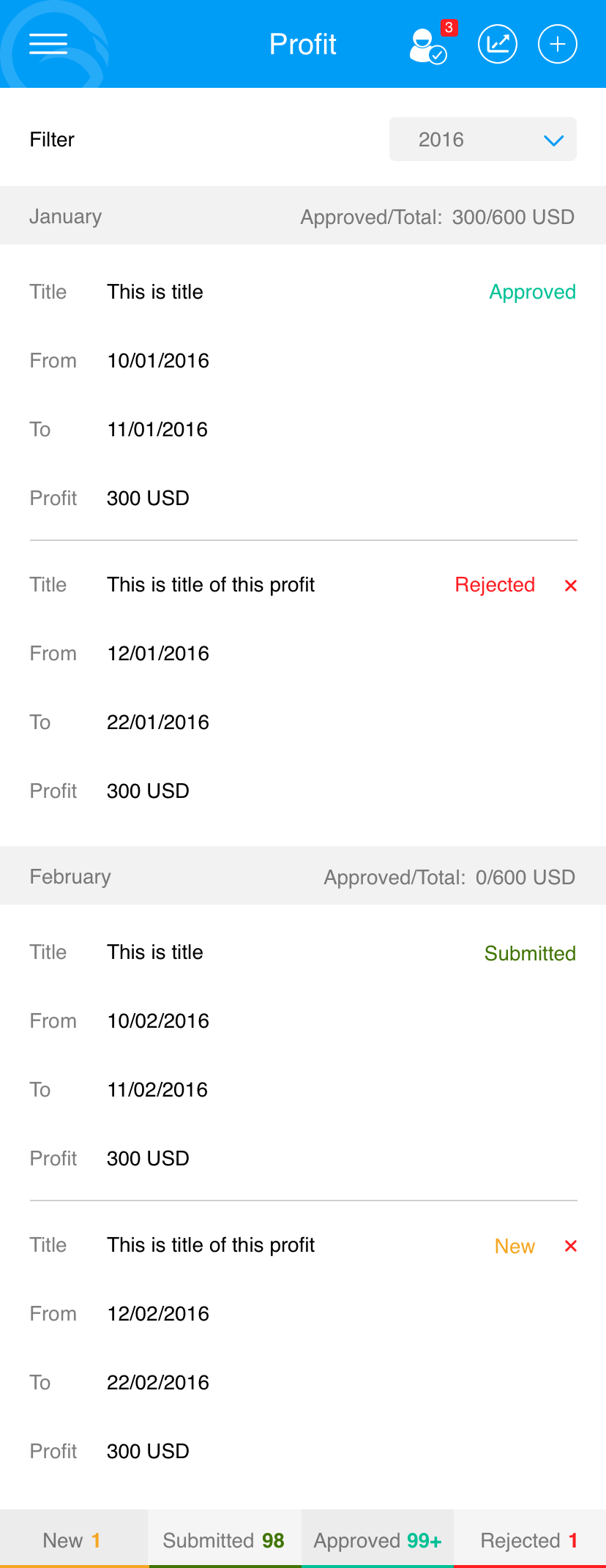
## 2. Giới thiệu chức năng Profit

Profit giúp người dùng thống kê tổng số lợi nhuận theo tháng hoặc năm. Biểu diễn trực quan bằng danh sách hoặc biểu đồ.

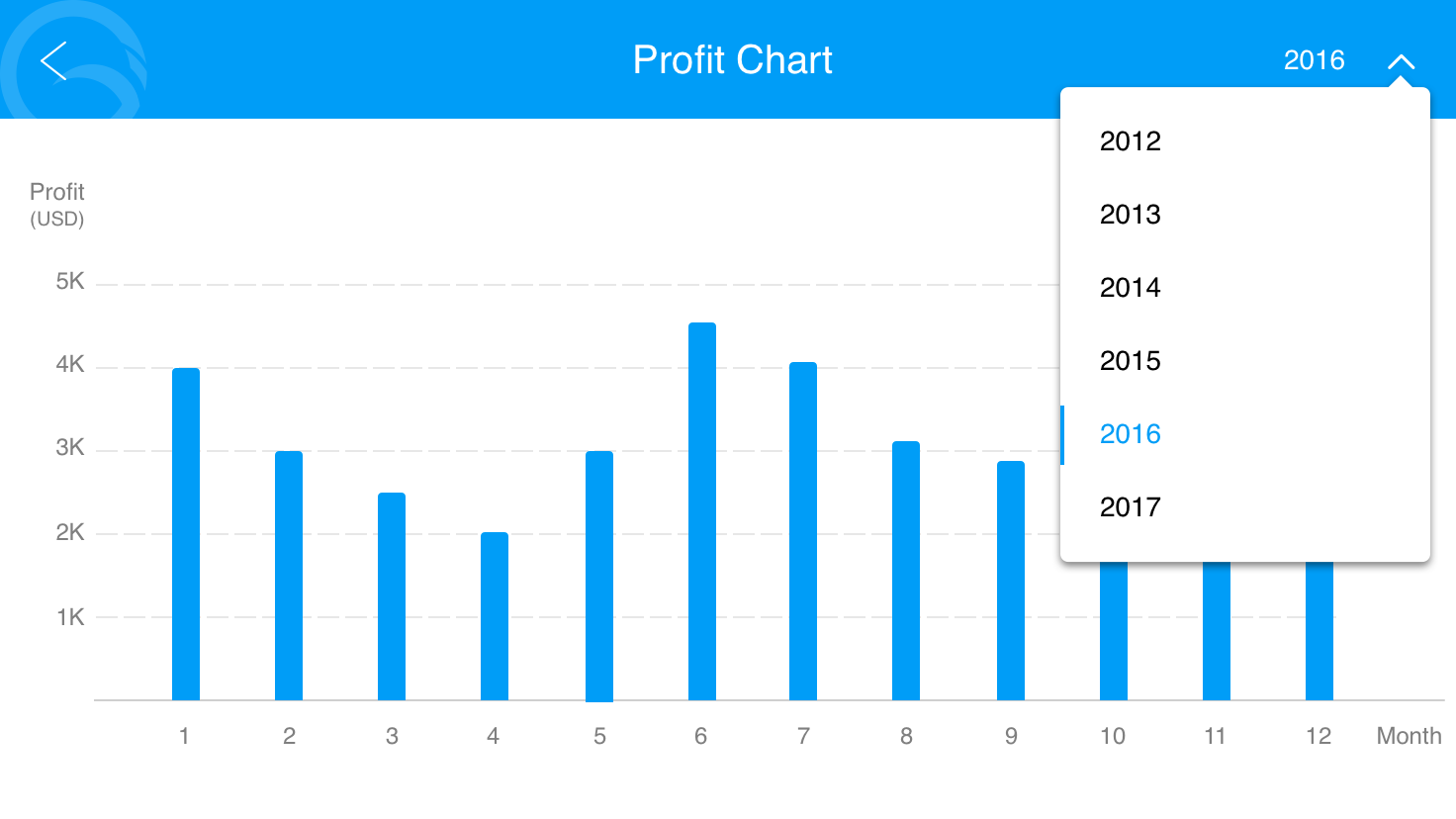


*Gito Profit trên Web*

Profit giúp người dùng thống kê tổng số lợi nhuận theo tháng hoặc năm. Biểu diễn trực quan



*Profit Mobile dạng danh sách*

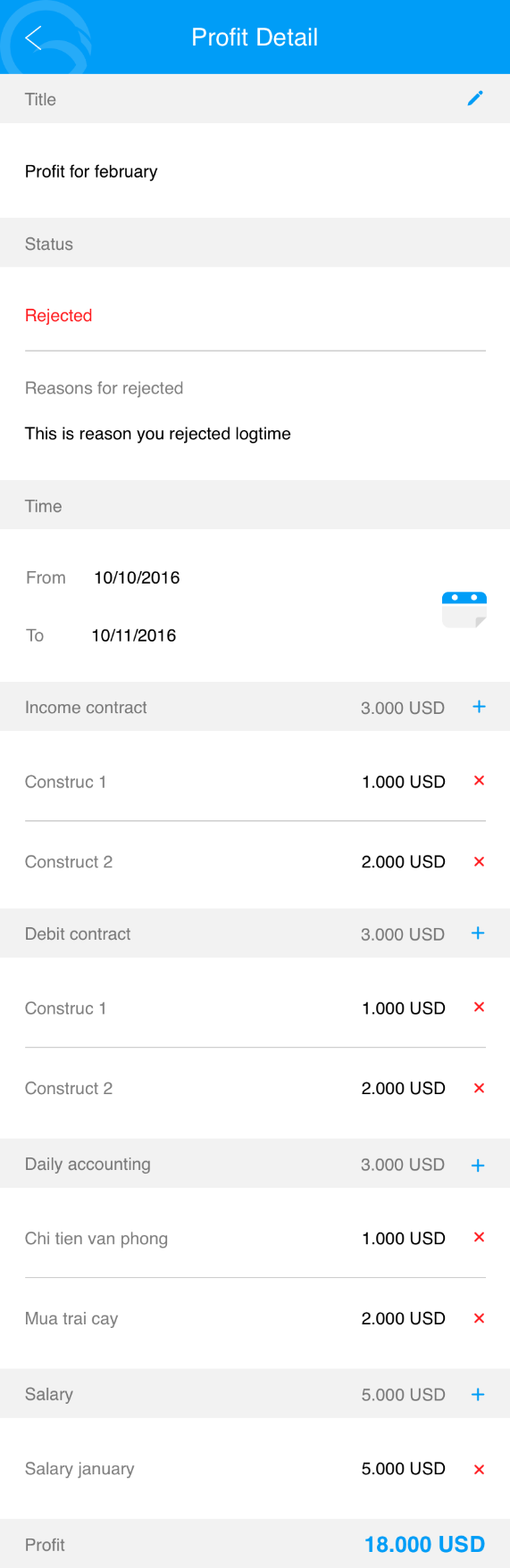
*Profit Mobile dạng biểu đồ*

Tại đây, người dùng có thể tự tạo một Profit, sau đó Profit mới này sẽ ở trạng thái New và chờ được cấp trên Approved hoặc Rejected tùy vào độ chính xác của nội dung.



Phía dưới sẽ là thanh Filter theo trạng thái. Khi người dùng ấn vào các nút này sẽ Filter tương ứng, cho phép Filter nhiều trạng thái cùng một lúc. Hình dưới đây là thông tin một Profit đã bị Rejected trong tháng 10/2016. Con số tổng Profit sẽ được căn cứ theo các thành phần đó là : Income Contract, Debit Contract, Dally Accounting và Salary. Công thức tính Profit sẽ là :

*Profit = Income Contract – Debit Contract – Dally Accounting – Salary (Nghiệp vụ từ anh Toàn CEO cung cấp)*



*Thông tin chi tiết một Profit*

## 3. Thực hiện

Trong khoảng thời gian một tháng, em đã thực hiện được những công việc như sau:

* Đọc thiết kế, xem API từ Postman để hiểu hơn về chức năng, nghiệp vụ.
* Tiến hành xây dựng chức năng Profit, tham khảo từ những màn hình khác của những người đi trước.
* Nhận bug trên Github issues và Fix bug, đồng thời bảo trì Code.
* Nhận thông tin cập nhật (Thanh Filter theo trạng thái) và tiến hành cập nhật theo bản Design mới.

Các công việc trên được hoàn thành là nhờ một phần giúp đỡ nhiệt tình từ các thành viên khác trong Android Team.

## 4. Kế hoạch chi tiết

* Giai đoạn Alpha:
  + Xem document trên hệ thống Github
  + Đọc Postman về API Profit
  + Xem design bằng Figma và Google Drive
  + Tạo class Profit Activity và kết nối vào Menu chính bằng các kế thừa lại Class DrawerActivity.
  + Tạo giao diện XML theo bản mẫu Design
  + Xử lý load danh sách (Tham khảo chức năng Survey của anh Tuấn)

Kết quả:

* + Hiểu được nghiệp vụ của chức năng được giao.
  + Biết cách tạo một Class và tích hợp vào Framework của công ty.
  + Là bước cơ sở để tiện cho công việc trong giao đoạn Beta.
* Giai đoạn Beta:
  + Tạo XML và xử lí logic cho Create Profit
  + Tạo XML và xử lí logic cho Detail Profit
  + Tạo XML và xử lí logic cho Add Income Contract
  + Tạo XML và xử lí logic cho Add Debit Contract
  + Tạo XML và xử lí logic cho Add Debit Daily Accounting
  + Tạo XML và xử lí logic cho Add Debit Salary
  + Tham khảo thư viện và thiết kế biểu đồ Profit Chart.
  + Tính toán Profit và kiểm tra lại Server
  + Xử lí Bug từ Tester Team
  + Cập nhật chức năng Filter theo trạng thái.
  + Bảo trì, nâng cấp Code theo yêu cầu của Android Leader Team

Kết quả:

* + Hoàn thành chức năng Profit.
  + Tự mình đi theo quy trình phát triển một chức năng từ đầu tới cuối một cách chuyên nghiệp.
  + Tiền đề để phát triển các chức năng khác được giao một cách nhanh chóng.

# **TỔNG KẾT**

* Sau 3 tháng thực tập tại Horical, em đã học được nhiều kinh nghiệm bổ ích về lập trình android. Đã hoàn thành chức năng Profit (100%) và hiện thực ứng dụng Gito Testcase (70%). Nhờ đó, em đã hiểu được quy trình phát triển một ứng dụng Android, đồng thời hiểu được trải nghiệm làm dự án thực tế, tăng kĩ năng giao tiếp, xử lí tình huống.
* Chân thành cám ơn quý công ty Horical đã giúp đỡ tận tình cho em trong suốt 3 tháng thực tập tại công ty. Đặc biệt là anh Toàn giám đốc công ty và anh Vinh leader team android đã hướng dẫn tận tình cho em để em hoàn thành công việc được giao.

# **TÀI LIỆU THAM KHẢO**

## Tổng quan

MVP Framework ▶▶▶▶▶

Retrofit ▶▶▶▶▶

Dagger 2 ▶▶▶▶▶

Glide ▶▶▶▶▶

Services Android ▶▶▶▶▶

Design Patterns- Builder ▶▶▶▶▶

Git Console ▶▶▶▶▶

## Link chi tiết

* MVP Framework:
  + [https://viblo.asia/p/mvp-pattern-for-android-1qm6RWzOveJE](https://viblo.asia/p/mvp-pattern-for-android-1qm6RWzOveJE%20)
* Retrofit :
  + [http://square.github.io/retrofit/](http://square.github.io/retrofit/%20)
* Dagger 2:
  + [http://www.vogella.com/tutorials/Dagger/article.html](http://www.vogella.com/tutorials/Dagger/article.html%20)
* Glide:
  + [https://github.com/bumptech/glide](https://github.com/bumptech/glide%20)
* Services Android:
  + [https://developer.android.com/guide/components/services](https://developer.android.com/guide/components/services%20)
* Design Patterns- Builder:
  + [https://viblo.asia/p/design-patterns-builder-pattern-DXOkRZARkdZ](https://viblo.asia/p/design-patterns-builder-pattern-DXOkRZARkdZ%20)
* Git Console:
  + [https://git-scm.com/book/en/v2/Getting-Started-The-Command-Line](https://git-scm.com/book/en/v2/Getting-Started-The-Command-Line%20)