**I. Firebase**

1. Set up project

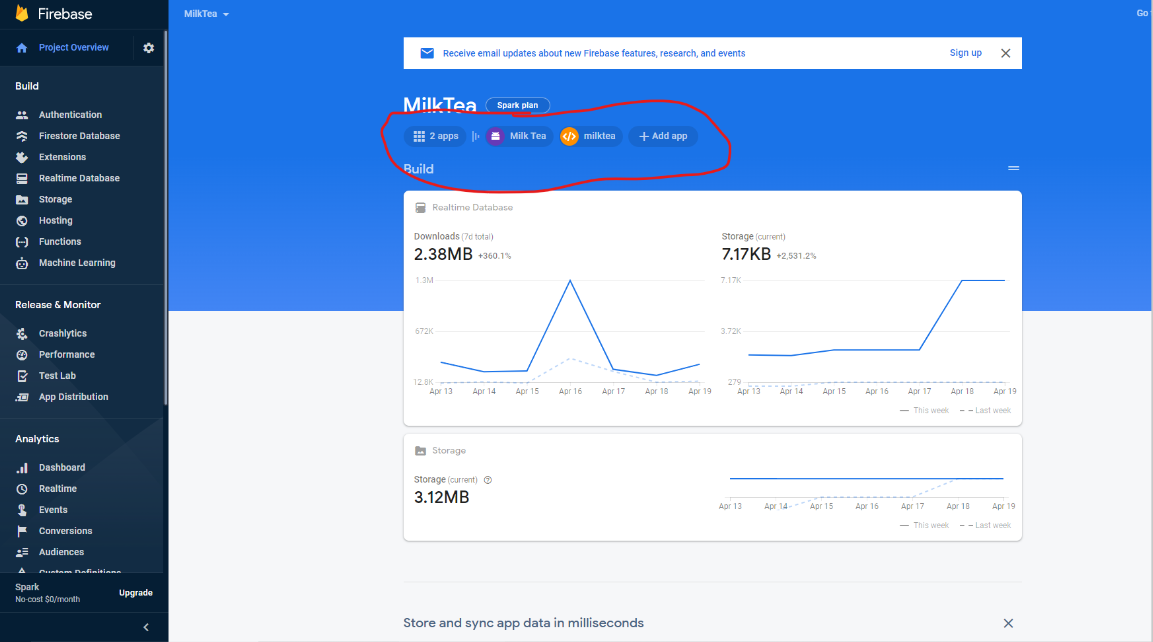
a. Set up trên Firebase

Đăng nhập và tạo tài khoản trên Firebase, sau đó tạo project MilkTea, ta sẽ có giao diện như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình

Mô tả được tạo tự động

Trong đó để có thể cho Firebase biết được chúng ta đang xây dựng trên nền tảng nào: Web, Android, ...



Nhấn Add App để thêm các nền tảng mà nhà phát triển muốn có thêm.

Mỗi nền tảng sẽ có những yêu cầu cần phải config riêng để phù hợp. Để biết cần phải config như thế nào thì ta nhấn vô các icon đó, Web sẽ đưa ta đến 1 trang để ta có thể xử lý các vấn đề trên.

VD: với android, ta có thể nhấn vào ô **See SDK instructions** để xem cần phải config như thế nào. Tương tự với IOS và Web.

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình

Mô tả được tạo tự động

b. Set up trên Flutter

Tải thư viện <https://pub.dev/packages/firebase_core>, sau đó vô phần code để set up.

Trên các thiết bị client, thêm các dòng sau vào trong phần main để bổ sung các dòng như bên dưới (phần BlocOverides chỉ để theo dõi các trạng thái của bloc trong app, có thể có hoặc không cũng được) để cho Firebase bắt đầu khởi tạo:

Ảnh có chứa văn bản

Mô tả được tạo tự động

Trên server, vì Web cần có những config khác nên ta phải làm như bên dưới để Firebase có thể khởi tạo.

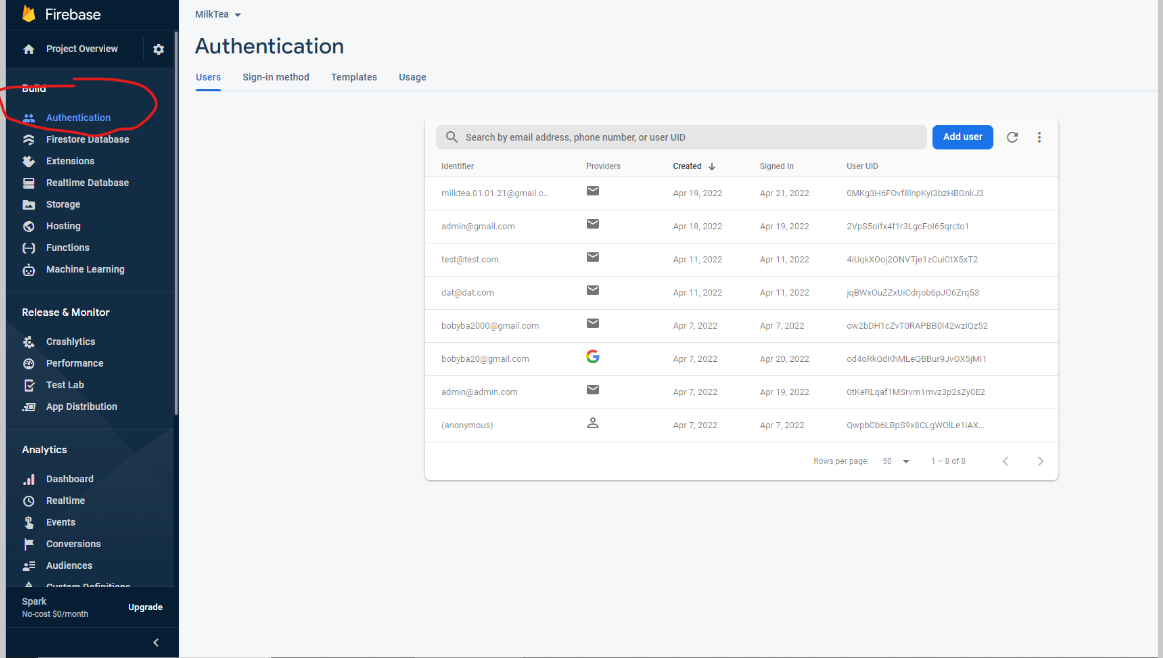
Ảnh có chứa văn bản

Mô tả được tạo tự động

2. Đăng nhập với firebase

a. Trên Firebase

Vô firebase Web, chọn Authentication, vô phần sign in method để chọn các phương thức đăng nhập vào trong app của mình.



b. Trên Flutter

Cài đặt thư viện <https://pub.dev/packages/firebase_auth>, trong code đã trình bày rất rõ quy trình đăng nhập cũng như đăng ký tài khoản trong file ***lib\page\login\_page.dart*** , có thể lên trang <https://firebase.flutter.dev/docs/auth/start/> để tìm hiểu thêm.

3. Load file

File sẽ được lưu trên firebase storage.

a. Trên firebase.

Cần sửa lại rules để có thể cho phép truy cập từ nhiều nguồn khác nhau, Ảnh có chứa văn bản

Mô tả được tạo tự động

b. Trên Flutter

Cài thư viện <https://pub.dev/packages/firebase_storage>, trong code đã trình bày rất rõ ở file ***lib\utils\storage\_service.dart***, có thể lên trang <https://firebase.flutter.dev/docs/storage/overview>, để tìm hiểu thêm

4. Lưu data

Data sẽ được lưu trên Firebase Realtime Database

a. Trên firebase

Config lại để có thể ghi hoặc đọc data ở bất kỳ thời điểm nào



b. Trên Flutter

Cài đặt thư viện <https://pub.dev/packages/firebase_database>, trong code đã có trình bày ở các phần sau:

* Get toàn bộ menu

Ảnh có chứa văn bản

Mô tả được tạo tự động

* Get toàn bộ Order

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, màu bạc

Mô tả được tạo tự động

Có thể tham khảo thêm trên trang <https://firebase.flutter.dev/docs/database/overview>.

Vì database là dạng NOSQL, nên ta cần phải lưu dưới dạng JSON

Để có thể parse từ các Object trong code sang dạng JSON để lưu trên database, và ngược lại, code đã có xử lý trong các file nằm trong thư mục model

VD: MenuItemModel

Ảnh có chứa văn bản

Mô tả được tạo tự động

**II. API**

API được lập trình đơn giản và đẩy lên trên heroku với đường link <https://milk-tea-api.herokuapp.com>, API này chỉ có một nhiệm vụ duy nhất là tính tổng giá trị các sản phẩm trong đơn hàng. API sẽ sử dụng thư viện DIO (<https://pub.dev/packages/dio>), phần gọi API được trình bày rõ trong file lib\service\api\_service.dart.

**III. Một số thư viện khác được sử dụng**

font\_awesome\_flutter: ^9.2.0: load các icon Google

email\_validator: ^2.0.1: sử dụng để kiểm tra tính hợp lệ của mail

google\_sign\_in: ^5.2.4 để có thể kết hợp với Firebase Auth và đăng nhập trên Google.

iconsax: ^0.0.8 một số icon.

url\_launcher: ^6.0.20 để mở 1 số trang như Youtube, Google Map

  flutter\_bloc: ^8.0.1 dùng để quản lý state

  flutter\_easyloading: ^3.0.3 dùng để show dialog báo loading

  cached\_network\_image: ^3.2.0 Load image từ url

  file\_picker: ^4.5.1 chọn file từ thiết bị

  flutter\_form\_builder: ^7.1.1 Build các ô để điền nội dung vào

  fluttertoast: ^8.0.9 Show toast để thông báo cho người dùng.

  easy\_debounce: ^2.0.2+1 Dùng để tạo độ trễ trong phần search

  scrollable\_positioned\_list: ^0.2.3 Dùng để scroll và nhảy tới các vị trí của list

  flutter\_braintree: ^2.3.1 Dùng để thanh toán bằng paypal

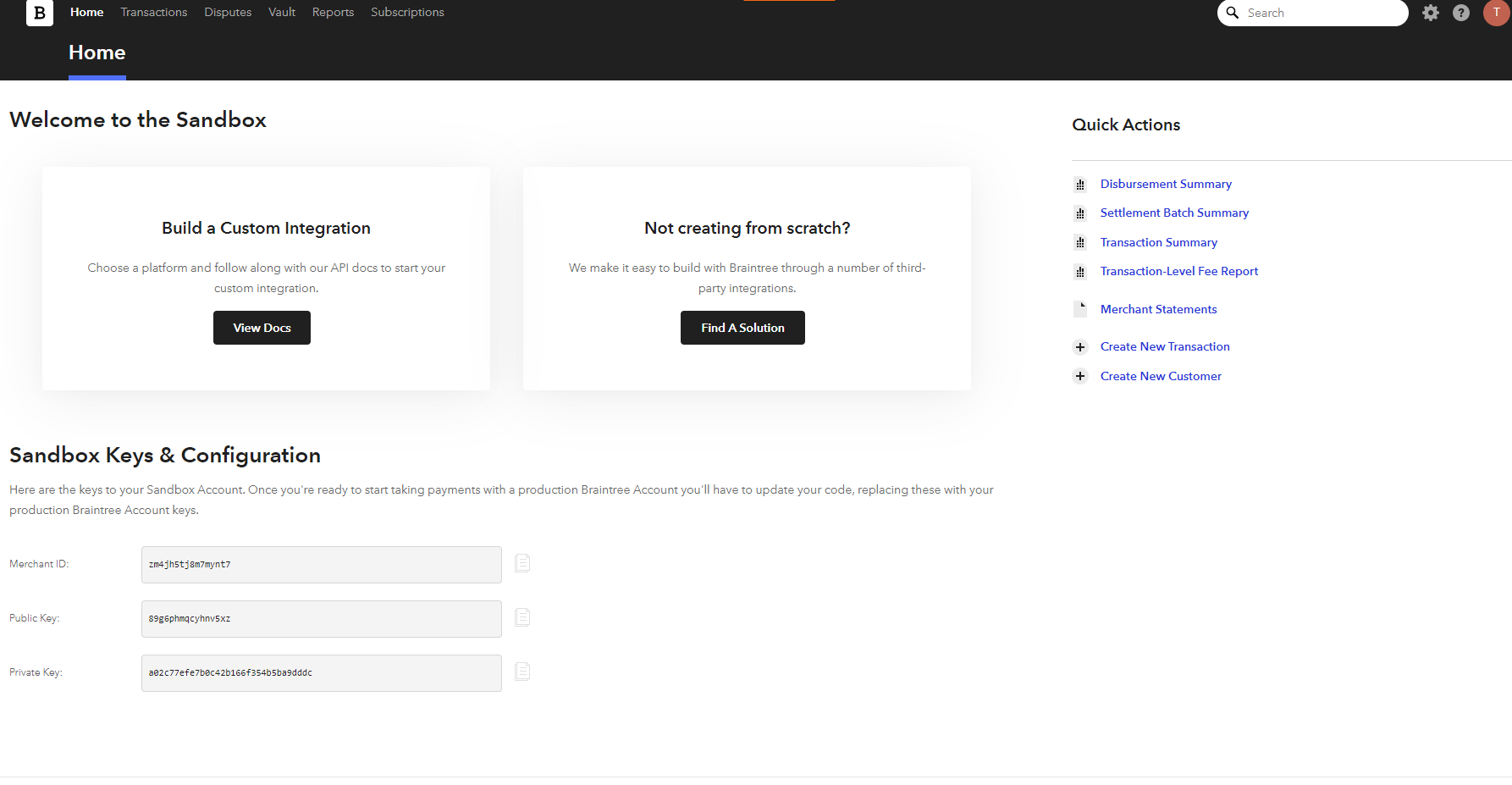
**IV. Thanh toán bằng Paypal**

Tạo tài khoản paypal,

Ảnh có chứa văn bản, người, ảnh chụp màn hình, máy tính xách tay

Mô tả được tạo tự động

Tạo tài khoản Braintree (chọn vị trí là US để đỡ handle việc phải xác nhận nơi ở).

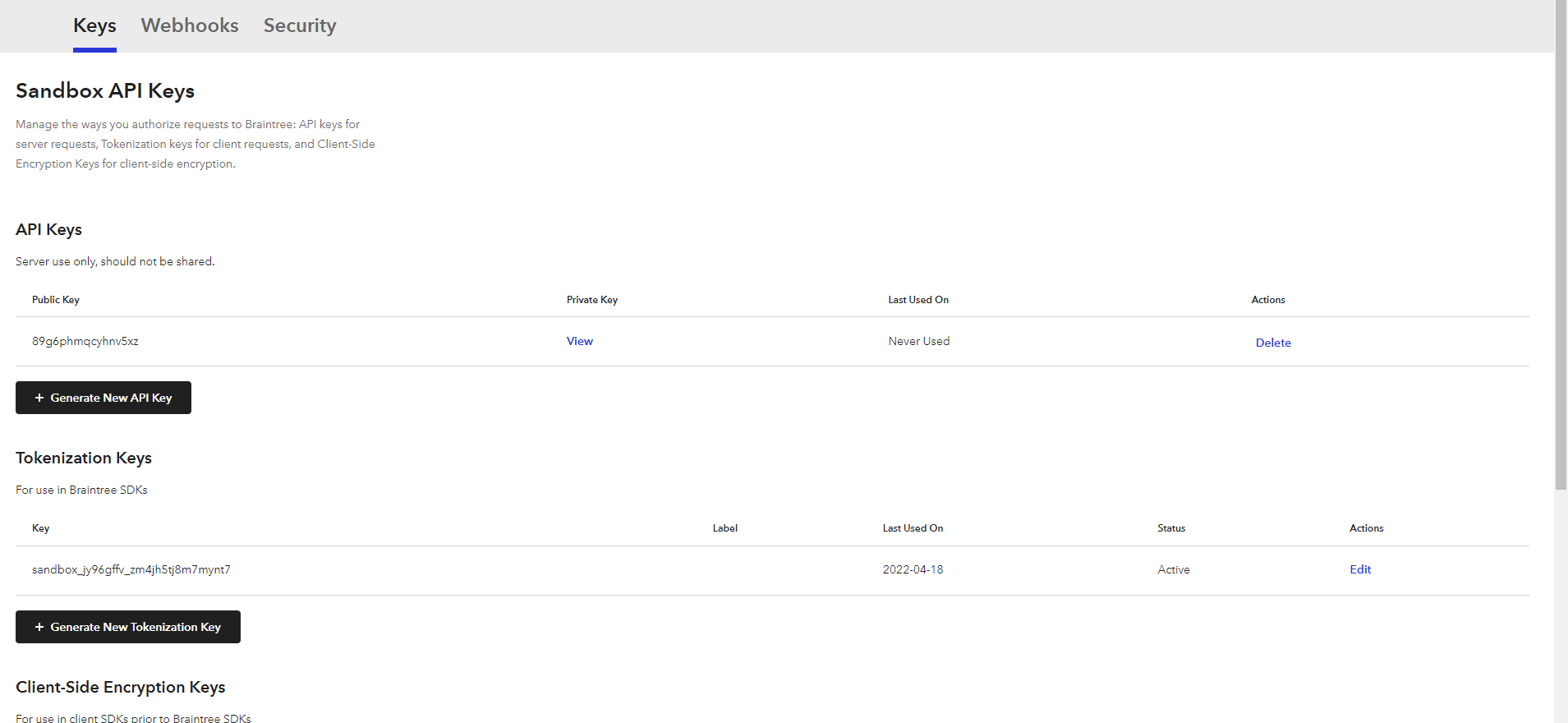


Vô phần cài đặt, chọn API

Ảnh có chứa văn bản

Mô tả được tạo tự động

Tạo một key token để sử dụng cho việc thanh toán



Xong chọn tab Processing, vô phần Payment Methods, chọn Paypal (mặc định là được chọn), link tài khoản với một tài khoản testing. Hoặc có thể làm theo hướng dẫn

Ảnh có chứa văn bản

Mô tả được tạo tự động

Hướng dẫn chi tiết cho việc tạo tài khoản paypal admin và tạo tài khoản paypal customer.

<https://developer.paypal.com/braintree/docs/guides/paypal/testing-go-live/#linked-paypal-testing>