

Flutterwave

Flutterwave est un fournisseur de services de paiement qui offre divers services pour envoyer et recevoir des paiements. Si vous êtes propriétaire d'une entreprise, vous pouvez collecter les paiements de vos clients en utilisant la variété d'options de Flutterwave :

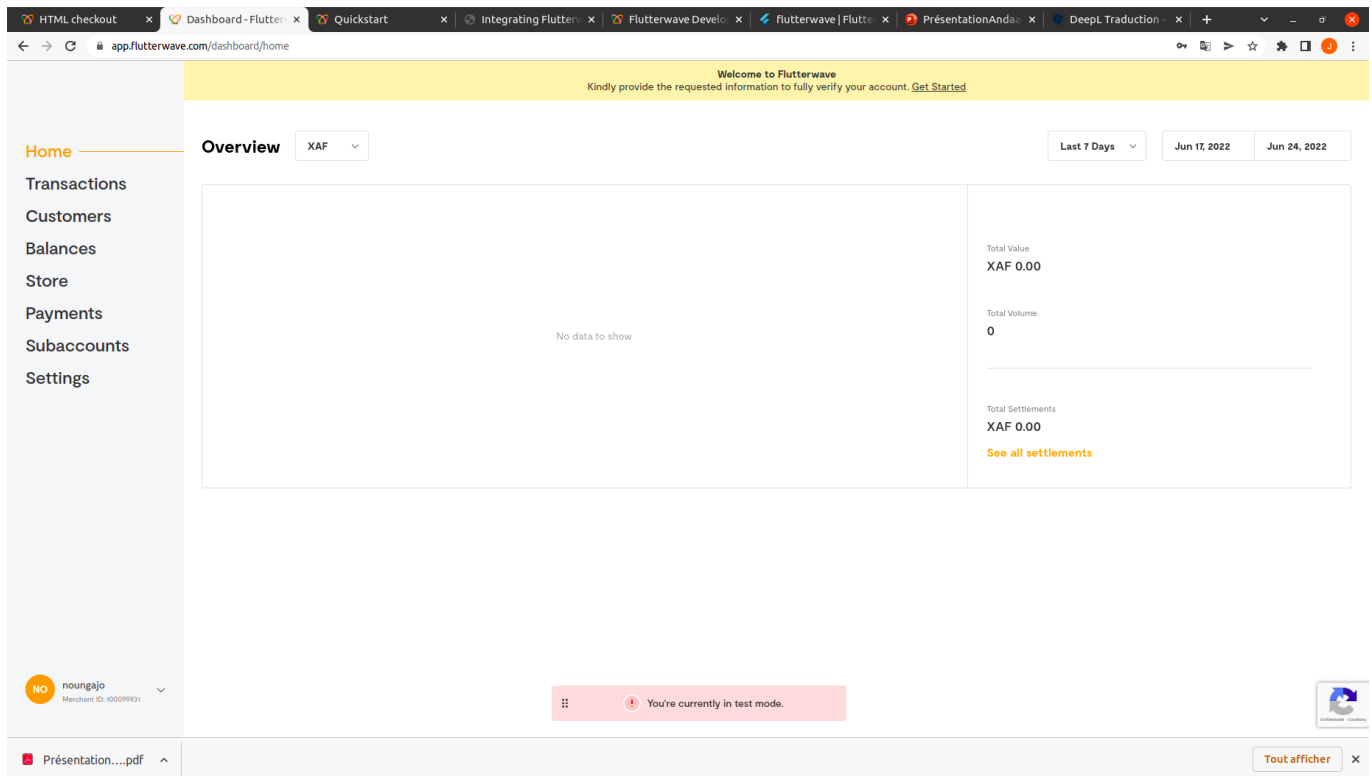
- Bank transfers pour envoyer des fonds à un compte bancaire
- Mobile money (exemple de MTN money)
- Debit and credit cards pour les cartes visa, mastercard ...
- POS :point de vente mobile
- Bank account pour collecter l'argent directement depuis le compte du client
- Visa QR
- USSD
- Mpesa(Kenya).

Flutterwave integration paiement

- Flutterwave Inline est le moyen le plus simple de collecter des paiements sur votre site Web [lien](#)
- Flutterwave Standard est notre flux de paiement "standard". Voici comment ça fonctionne: [lien](#)
- Direct Charge API : chaque type de facturation directe a ses propres exigences et flux d'autorisation. les types de facturations(cards; bank accounts Nigeria,UK, US/South Africa; Mobile money M-pesa, Rwanda, Ghana, Uganda, Zambia, Francophone countries; USSD; Bank transfer) [lien](#)
- Payment Links : permettent aux commerçants d'accepter les paiements sur leur site sans avoir à l'intégrer. Cette solution est idéale pour les particuliers et les commerçants qui ne disposent pas de ressources en matière de développement.
- HTML Checkout : est semblable à flutterwave inline, seulement qu'il est réalisé uniquement avec html [lien](#)

Pour débiter

- Créer un compte flutterwave via le lien : [lien](#)
- Se connecter : pour voir le tableau de bord. Le tableau de bord contient divers éléments de détails du compte que vous pouvez explorer comme les transactions, les transferts, les sous-comptes, les options de facturation, les plans de paiement, les factures, etc.



actuellement nous sommes en mode test. Une fois que tout est en place il est possible de passer en mode live. Dans cet article, nous utiliserons le plugin flutter + flutterwave dont l'implémentation imite la méthode d'intégration des paiements en ligne de Flutterwave(Flutterwave Inline).

flutter + flutterwave

ici nous allons supposer que tous les concepts de base de flutter sont acquis. Sinon voici le lien pour débuter avec flutter .

créer un simple projet flutter

dans ce projet supprimer le code généré et faire:

Le package flutterwave

Flutterwave's Flutter SDK est le sdk officiel de Flutterwave pour intégrer le paiement Flutterwave dans votre application Flutter. Il est livré avec une interface utilisateur Drop In prête à l'emploi. Les méthodes de paiement actuellement prises en charge sont les suivantes :

- Cards,
- USSD,
- Mpesa,
- GH Mobile Money,
- UG Mobile Money,
- ZM Mobile Money,
- Rwanda Mobile Money,
- Franc Mobile Money and
- Nigeria Bank Account

pour l'ajouter au dans le fichier pubspec.yml faire :

```
# ajouter la dépendance
flutterwave: 1.0.1
# exécuter la commande
run flutter pub get
```

importer le package flutterwave

```
import 'package:flutterwave/flutterwave.dart';
```

Créer une instance flutterwave en appelant `Flutterwave.forUIPayment()` Le constructeur accepte une instance obligatoire des détails suivants : Context , publicKey, encryptionKey, amount, currency, email, fullName, txRef, isDebugMode and phoneNumber . Et ceci retournera une instance de Flutterwave .

Puis mettre sur pied une interface qui demande à l'utilisateur d'entrer des détails nécessaire tel que : *email, amount, full name, and phone number*

NB: Si vous attendez des paiements Flutterwave d'utilisateurs ayant des devises différentes, vous pouvez ajouter le champ de saisie de la devise dans l'interface utilisateur.

Utilisation de flutterwave

Se logger puis aller sur setting, puis API keys [lien](#) pour générer de nouvelles clés.

General Bank Accounts Users **API** Webhooks Account Settings

Test API Keys

Public Key

FLWPU... X Copy key

Secret Key

FLWSECK_... 17310-X Copy key

Encryption Key ⓘ

F... Copy key

. copier *Public key* et *Encryption key*

sur l'image ci-dessous, la *public key* et *encryption key** sont en mode test.

instance complet de flutterwave

```

//Add a method to make the flutter wave payment
//This Method includes all the values needed to create the Flutterwave
Instance
void _makeFlutterwavePayment(BuildContext context, String fullname, String
phone, String email, String amount) async {
  try {
    Flutterwave flutterwave = Flutterwave.forUIPayment(
      //the first 10 fields below are required/mandatory
      context: this.context,
      fullName: fullname,
      phoneNumber: phone,
      email: email,
      amount: amount,
      //Use your Public and Encryption Keys from your Flutterwave
account on the dashboard
      encryptionKey: "Your Encryption Key",
      publicKey: "Your Public Key",
      currency: currency,
      txRef: DateTime.now().toIso8601String(),
      //Setting DebugMode below to true since will be using test mode.
      //You can set it to false when using production environment.
      isDebugMode: true,
      //configure the the type of payments that your business will
accept
      acceptCardPayment: false,
      acceptUSSDPayment: false,
      acceptAccountPayment: false,
      acceptFrancophoneMobileMoney: false,
      acceptGhanaPayment: false,
      acceptMpesaPayment: true,
      acceptRwandaMoneyPayment: false,
      acceptUgandaPayment: false,
      acceptZambiaPayment: false
    );
  }
}

```

Note: sur cette démo le paiement flutterwave est réalisé avec Mpesa. Il est nécessaire d'avoir un numéro de téléphone kenyan qui accepte les paiements par Mpease, et c'est pourquoi `acceptMpesaPayment`, est mis à `true` à la ligne 120 de la méthode ci-dessus. Cependant il est possible de mettre à `true` n'importe quel méthode de paiement d'un pays souhaité et le type de paiement acceptable. Par exemple si `CardPayment` veut être testée, il faudra la mettre à `true`.

Ensuite ajouter une réponse immédiatement sous l'instance flutterwave qui renvoie ue instance *flutterwave* sur laquelle nous appelons ensuite la méthode asynchrone `.initializeForUiPayments()`. le code est celui-ci:

```

final response = await flutterwave.initializeForUiPayments();
if (response == null) {
  // user didn't complete the transaction.
}

```

```

        print("Transaction Failed");
    } else {

        if (response.status == "Transaction successful") {
            print(response.data);
            print(response.message);

        } else {
            print(response.message);
        }
    }
} catch (error) {
    print(error);
}
}

```

réaliser un paiement

Maintenant qu'une instance flutterwave et une réponse ont été créées, il faut appeler la méthode **_makeFlutterwavePayment** créée ci-dessus sur la fonction onPressed() du bouton pour exécuter le paiement lorsque le bouton est pressé. code :

```

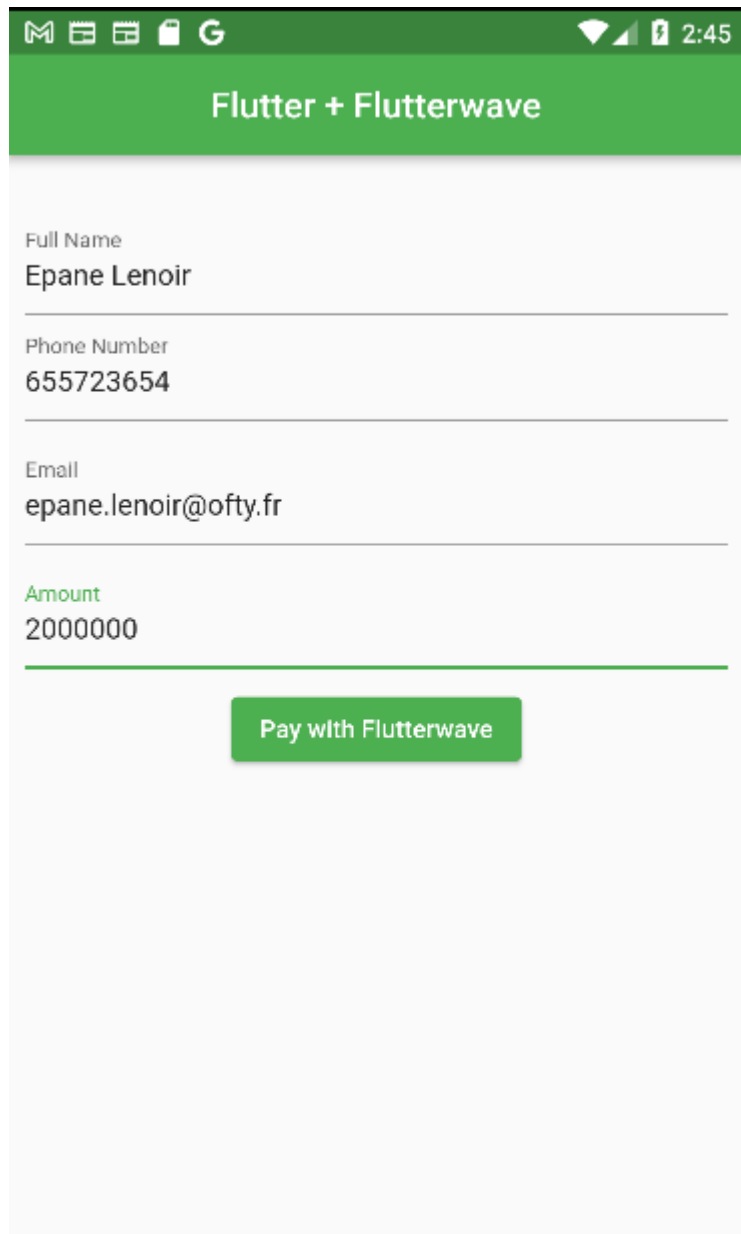
ElevatedButton(
  child: const Text('Pay with Flutterwave'),
  onPressed: () {
    final name = fullname.text;
    final userPhone = phone.text;
    final userEmail = email.text;
    final amountPaid = amount.text;

    if (formKey.currentState!.validate()) {
      _makeFlutterwavePayment(context, name, userPhone, userEmail, amountPaid);
    }
  },
),

```

Confirmation de paiement flutterwave

Exécutez maintenant l'application flutterwave construite ci-dessous et remplissez les détails de l'interface utilisateur en tant qu'utilisateur de test.



A mobile application interface for Flutter + Flutterwave. The top status bar shows icons for mail, calendar, and Google, along with signal, Wi-Fi, and battery indicators, and the time 2:45. Below the status bar is a green header with the text "Flutter + Flutterwave". The main form area is white and contains four input fields with labels and values: "Full Name" with "Epane Lenoir", "Phone Number" with "655723654", "Email" with "epane.lenoir@ofty.fr", and "Amount" with "2000000". A green button labeled "Pay with Flutterwave" is positioned below the "Amount" field.

Flutter + Flutterwave

Full Name
Epane Lenoir

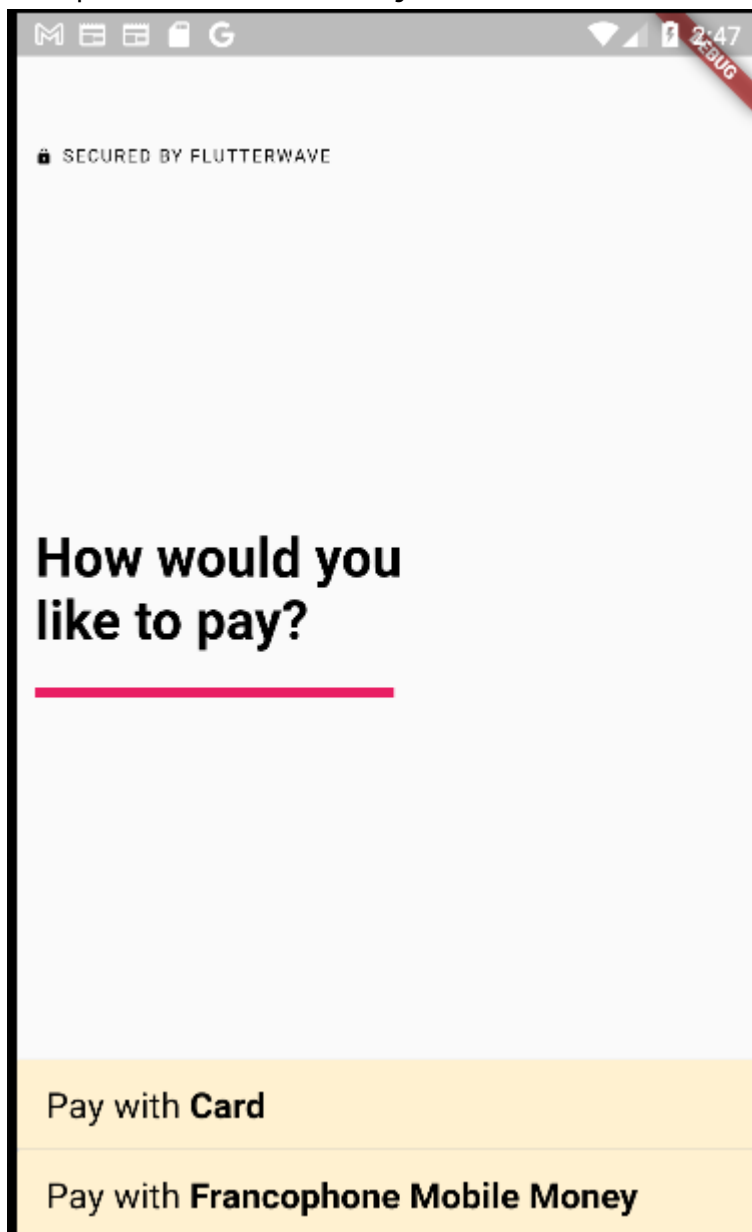
Phone Number
655723654

Email
epane.lenoir@ofty.fr

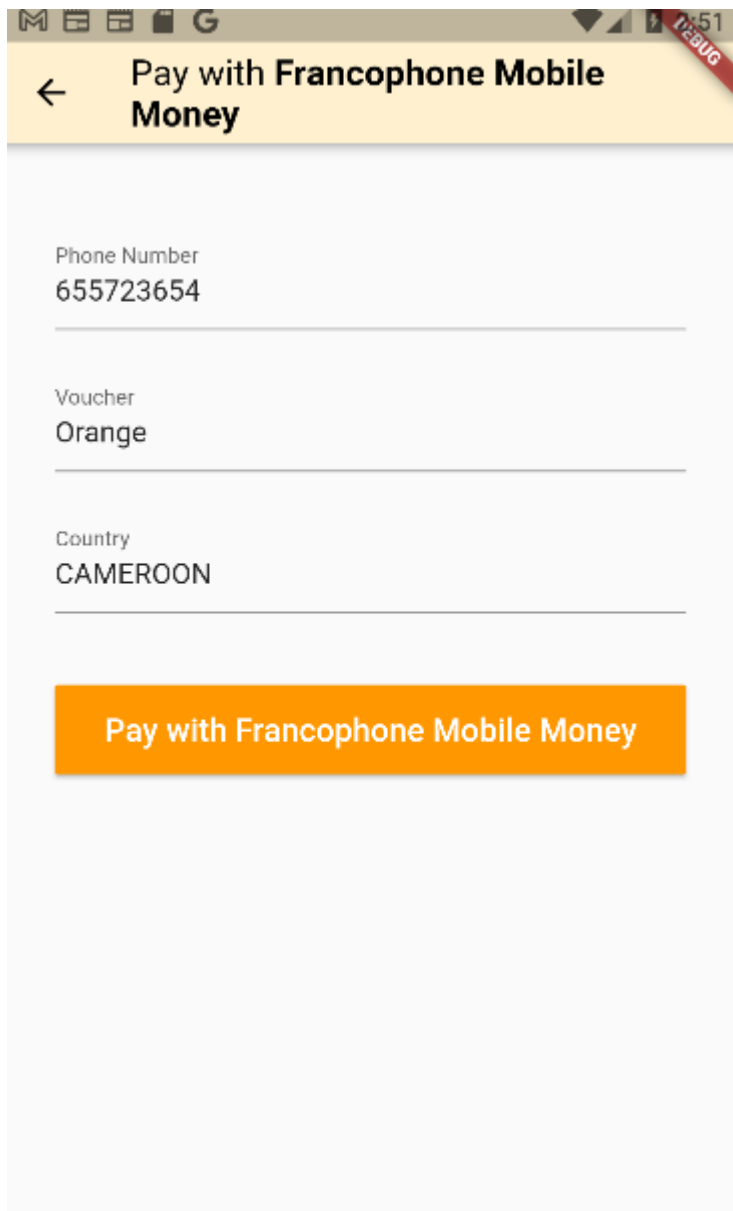
Amount
2000000

Pay with Flutterwave

alors presser sur le bouton **Pay with Flutterwave**. Il sera alors possible d'initier le paiement



il faudra choisir la méthode de paiement. Pour le test choisissons Francophone Mobile Money. Nous sommes invité à fournir des informations complémentaires



The screenshot shows a mobile application interface for paying with Francophone Mobile Money. At the top, there is a status bar with icons for signal, Wi-Fi, battery, and time (12:51). Below the status bar is a yellow header bar with a back arrow icon on the left and the text "Pay with Francophone Mobile Money" in bold. A red ribbon with the word "Debug" is visible in the top right corner of the app. The main content area is white and contains three input fields, each with a label and a value: "Phone Number" with "655723654", "Voucher" with "Orange", and "Country" with "CAMEROON". Each input field has a horizontal line below it. At the bottom of the form, there is a large orange button with the text "Pay with Francophone Mobile Money" in white.

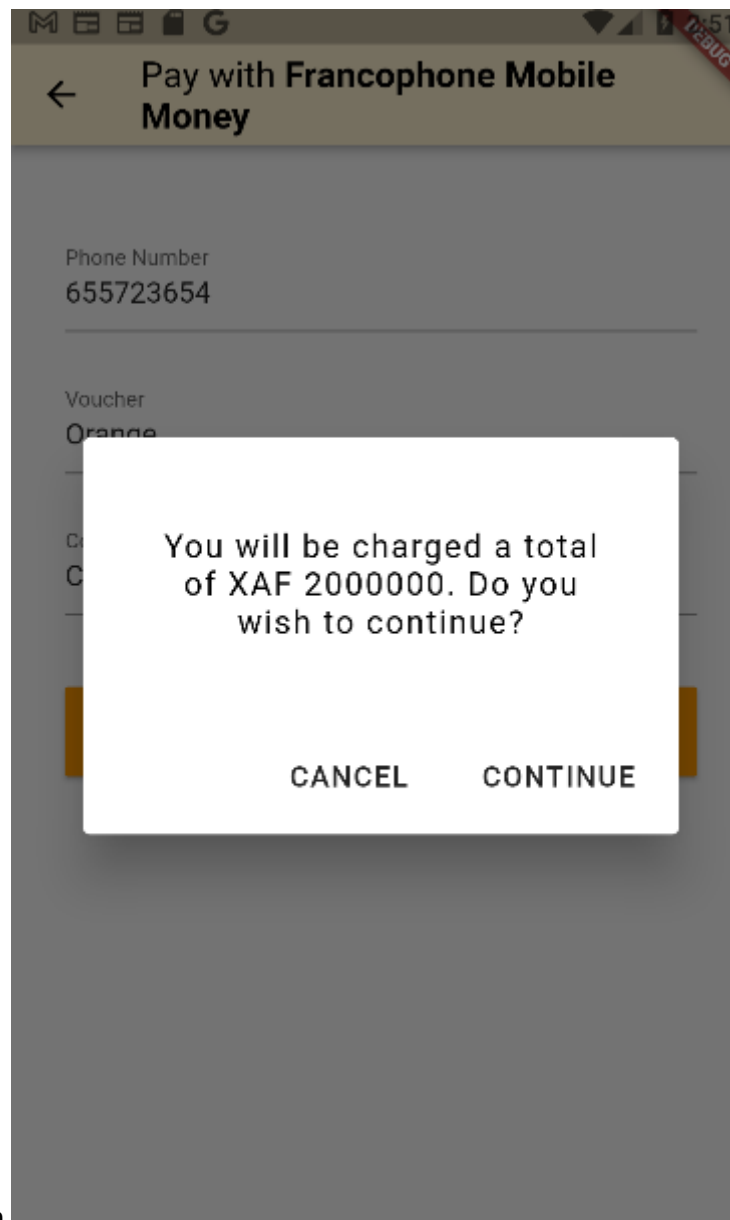
Pay with Francophone Mobile Money

Phone Number
655723654

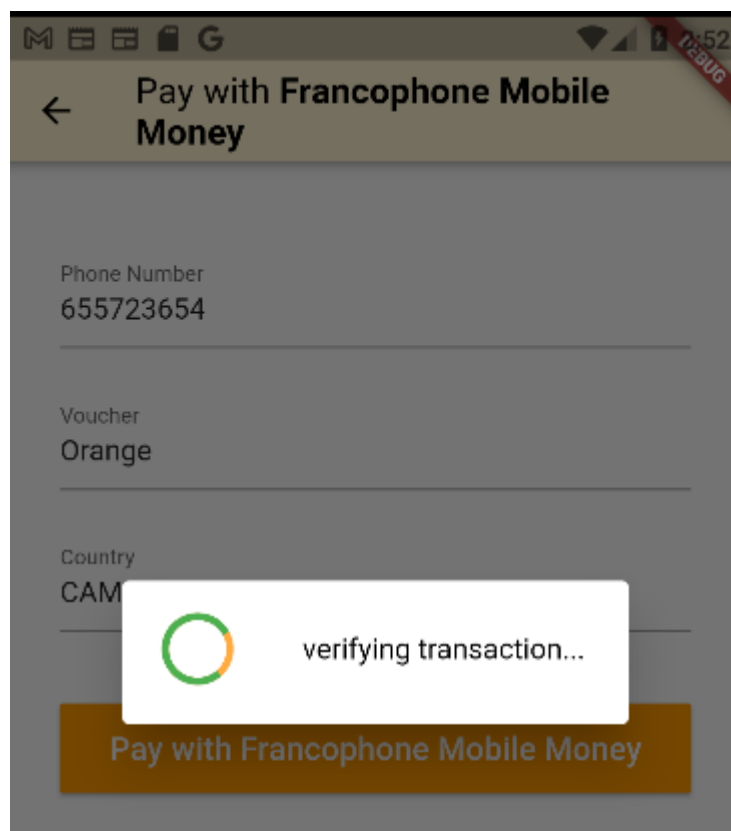
Voucher
Orange

Country
CAMEROON

Pay with Francophone Mobile Money



Vérification de la transaction





Après avoir effectué le paiement, je peux consulter mes mails pour voir que la trasaction est réussit. retour sur le site flutterwave,transaction [lien](#) permet d'observer toute les transactions réalisées.

< Main menu

Transactions

Refunds

Chargebacks

Welcome to Flutterwave

Kindly provide the requested information to fully verify your account. [Get Started](#)

3 transactions shown

Filter applied: 2

Download

Description	Amount	Payment type	Status	Date
Payment from epans.lenoir@ofty.fr	XAF 2,000,000.00	Mobilemoneysn	Successful	2 minutes ago
Payment from caleb.fonyuy@ofty.fr	XAF 10,000.00	Mobilemoneysn	Successful	15 minutes ago
Payment from noutcha.ofty@gmail.com	KES 2,000.00	Mpesa	Successful	5 days ago

Noutcha Ngapi Jonathan OFTY Cameroun