Further Work before Production

Yan Zhang
MOBILE APP DEVELOPER
@flamesoftab www.flamesoft.se

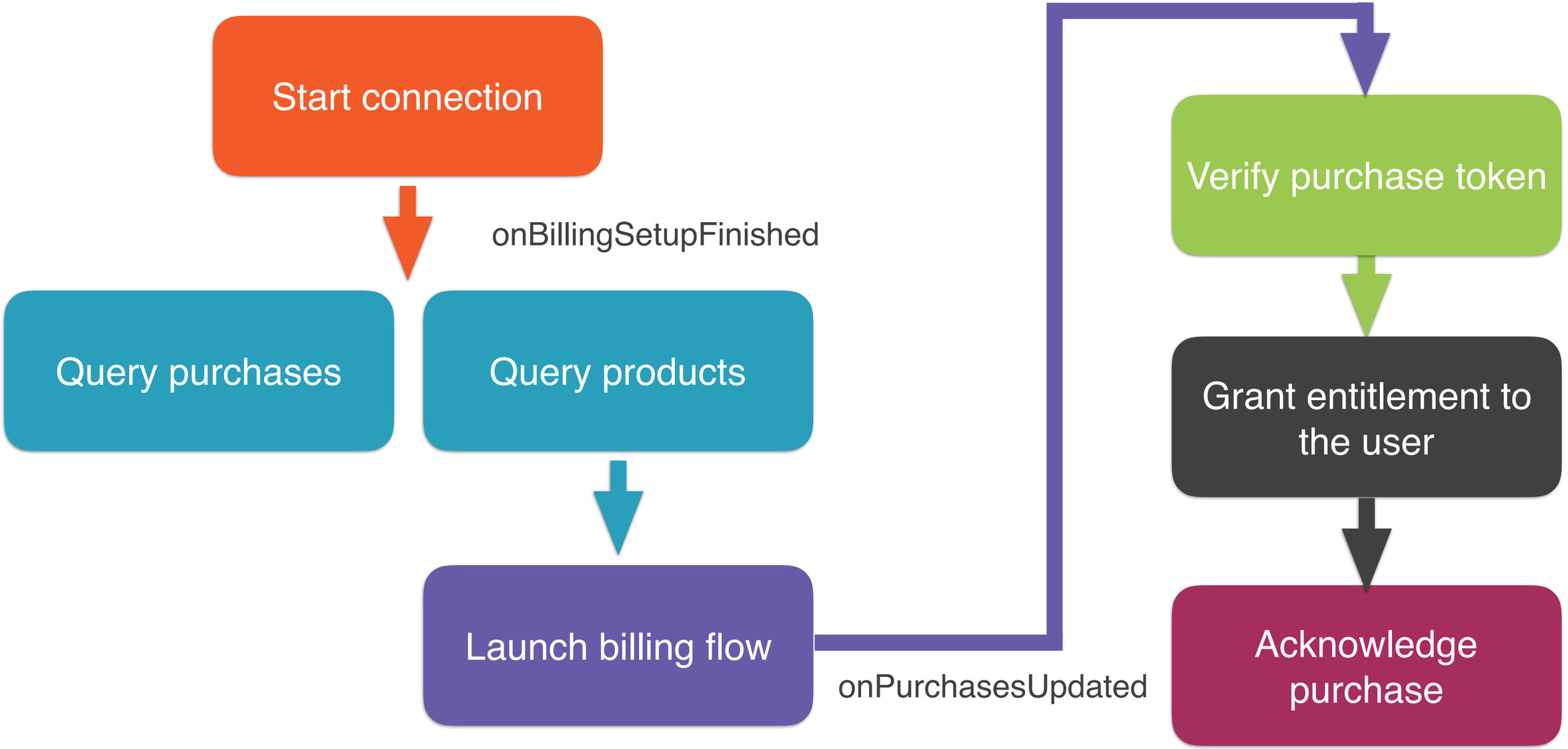
Overview

Backend system

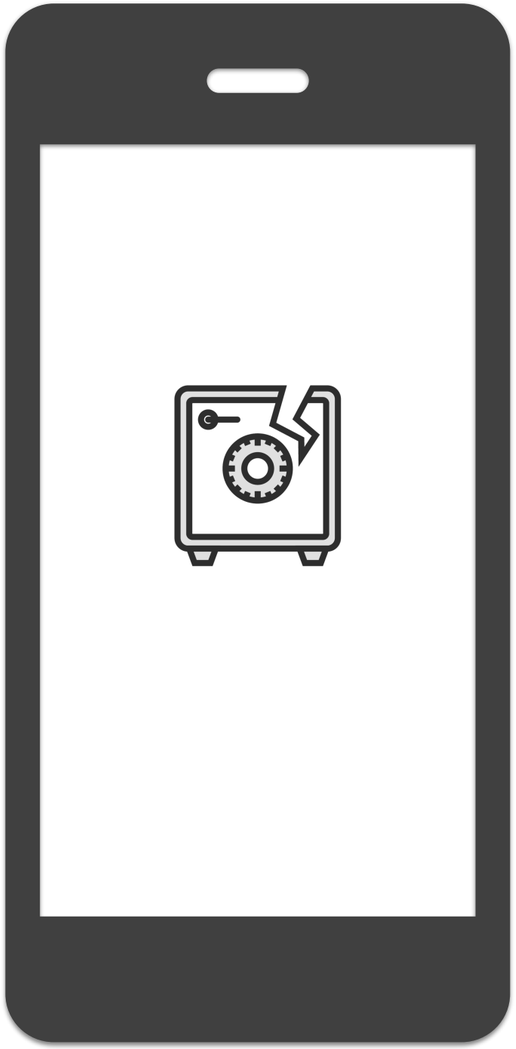
Security consideration

Design for good user experience

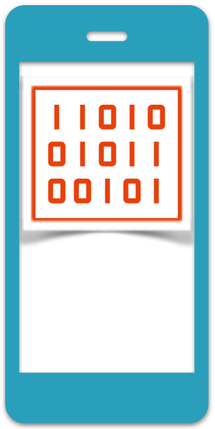
Purchase Flow



Insecure Local Purchase Verification



Verify Purchase on Your Server



How can we verify a purchase on a backend server?

Google Play Developer API

Server-to-server APIs

- Subscriptions and In-App Purchases API

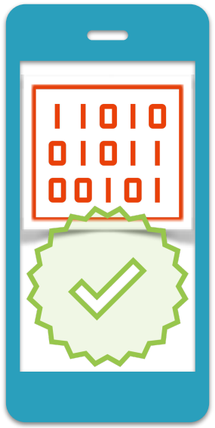
Complement the Google Play Billing Library

- Securely verifying purchases
- Get the latest subscription state

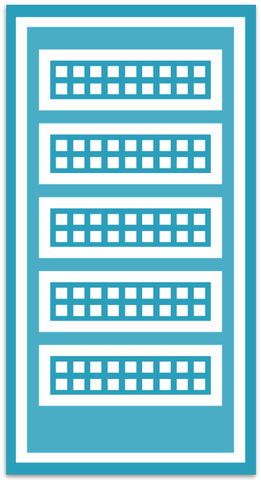
```
GET https://androidpublisher.googleapis.com/androidpublisher/v3/
applications/{packageName}/purchases/subscriptions/{subscriptionId}/
tokens/{token}
```

Get SubscriptionPurchase

Server-side Purchase Verification Flow



```
{
  "resource": {
    object (SubscriptionPurchase)
  }
}
```



How can we install a backend server?

Server

Maintain it yourself

Need to keep it secure

Costly

Function as a Service (FaaS)

Backend functions

Deploy

Flexible

Scalable

FaaS Providers

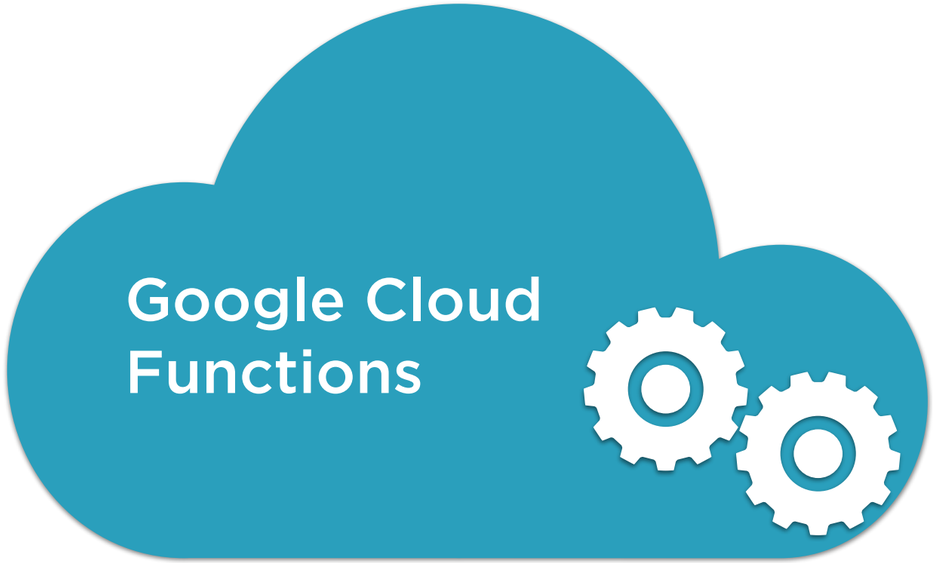
AWS Lambda

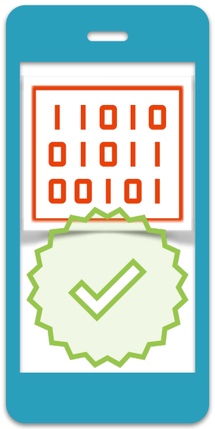
Google Cloud Functions

Microsoft Azure Functions

IBM Cloud

Server-side Purchase Verification Flow





Google Cloud Functions

Cloud Functions for Google Cloud Platform

Cloud Functions for Firebase

Cloud Functions for Google Cloud Platform

Cloud Functions for Google Cloud Platform serve as a connective layer allowing you to weave logic between Google Cloud Platform services by listening for and responding to events.

Cloud Functions for Firebase

Cloud Functions for Firebase provides a way to extend the behavior of Firebase and integrate Firebase features through the addition of server-side code.

Cloud Functions for Firebase

Access to mobile-centric services

- Authentication
- Realtime database

Node.js Runtime

- JavaScript
- TypeScript

Setup for Purchase Verification

Configure Google Developer API

Set up Cloud Functions for Firebase

- Install Firebase cli
- Write functions
- Deploy functions

Add code for Firebase in the app

Demo

Configure Google Developer API

- Create a new API project
- Create a service account to access the API

Access Types

OAuth clients

Service accounts

Setup Servers for Getting Subscription State Updates

Configure Google Developer API

Set up Cloud Functions for Firebase

- Install Firebase cli
- Write functions
- Deploy functions

Add code for Firebase in the app

Demo

Prepare for Cloud Functions for Firebase

- Create a Firebase project
- Set up Node.js and the Firebase CLI
- Initialize your project

Set up Node.js and the Firebase CLI

Node.js

https://nodejs.org/en/

Node Version Manager

https://github.com/nvm-sh/nvm/blob/master/README.md

Firebase CLI

https://firebase.google.com/docs/cli#setup_update_cli

Demo

Cloud Functions project

- Add functions and the key file
- Deploy functions

Billing

Set up billing account

- https://console.developers.google.com/billing

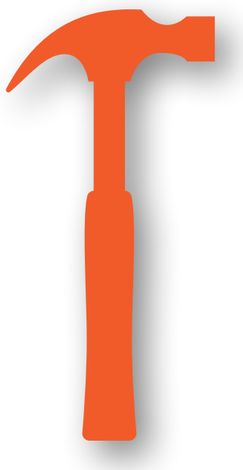
Enable billing for the project

- https://console.developers.google.com/billing/project

Demo

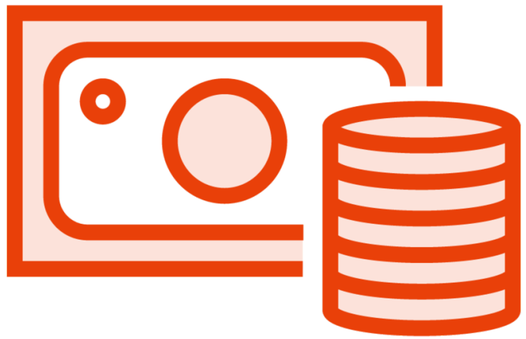
Call Cloud Functions for Firebase in the app

- Add Firebase support in the project
- Call Cloud Functions



Firebase Assistant

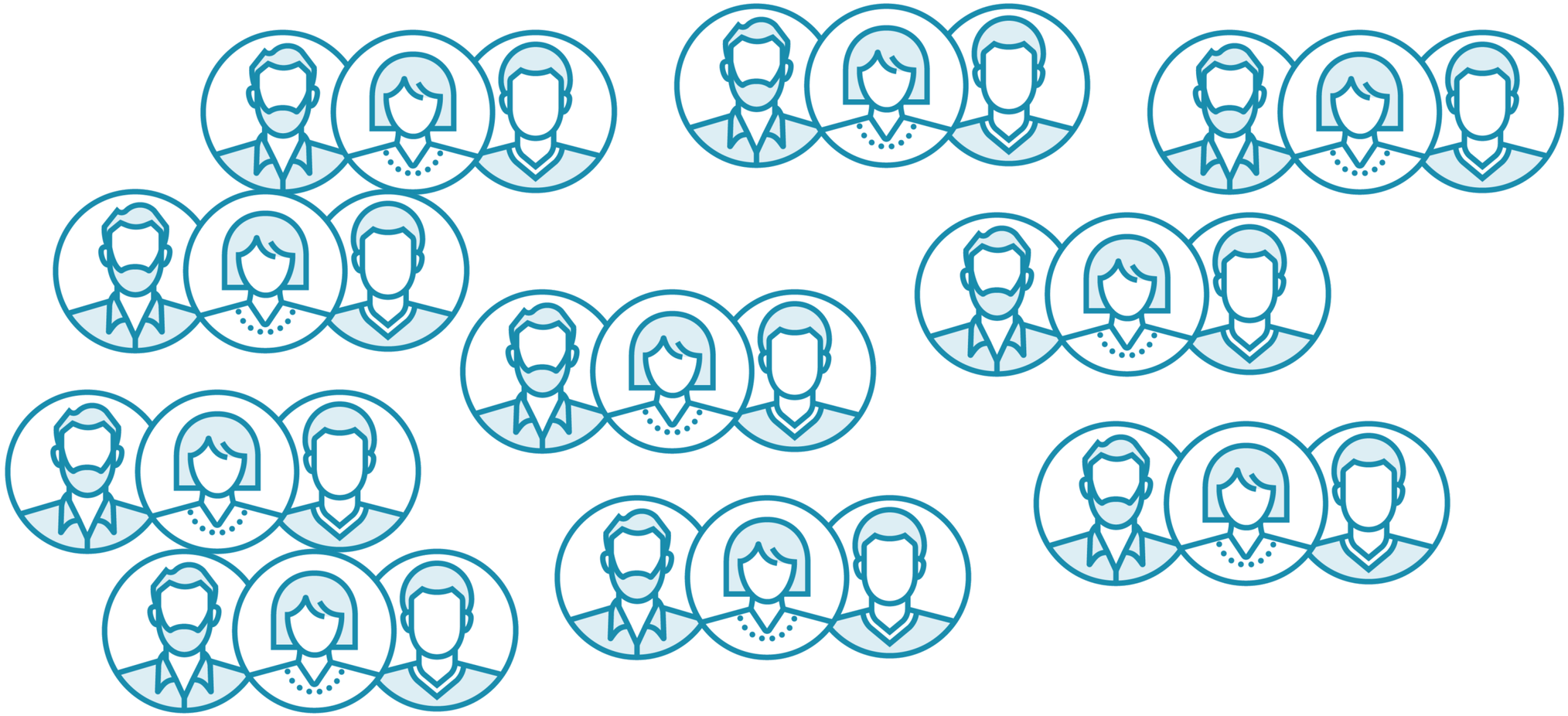
Tools -> Firebase



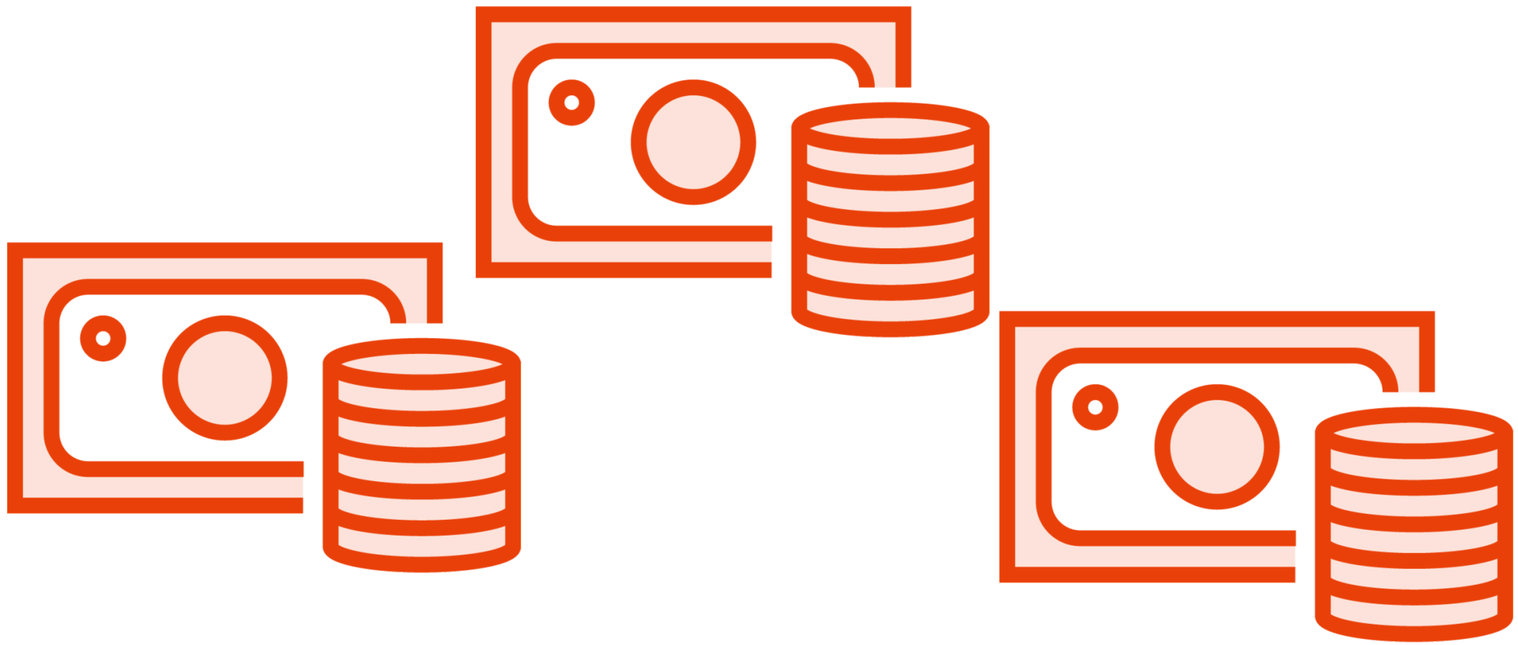
Google Play Developer API Quota

< 200,000 quotas/day per application

When Your App Becomes Popular

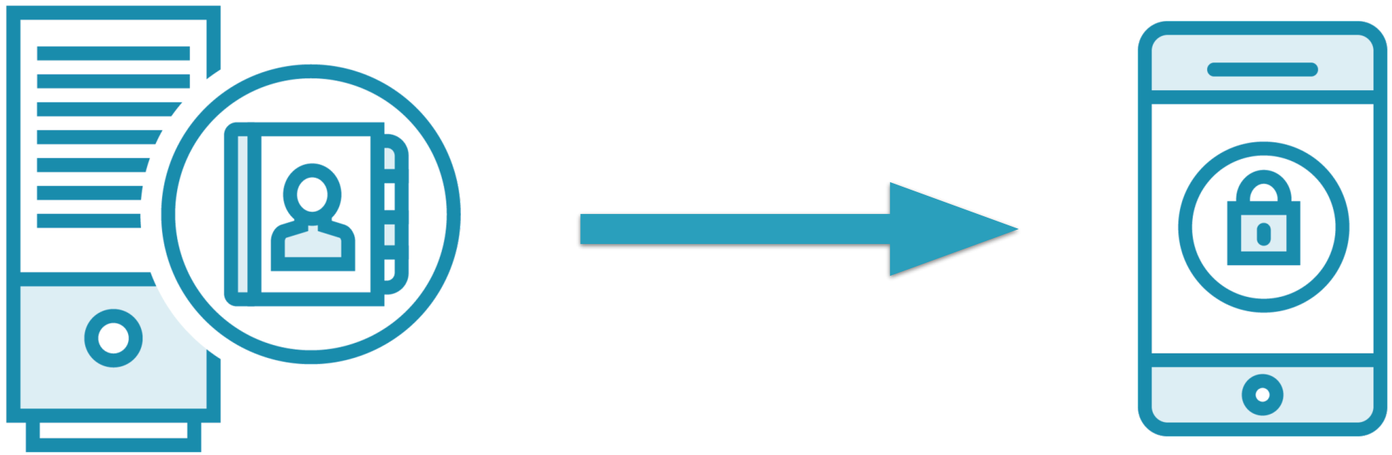


Increase Quota

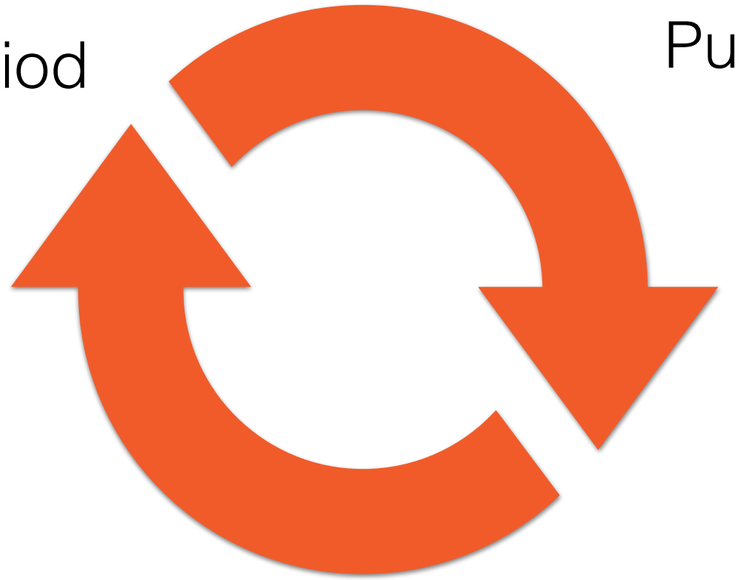


Saving is a Virtue

Update User Entitlement Correctly



Subscription Status



In grace period

Purchased

Paused

Expired

Canceled

On hold

Renewed

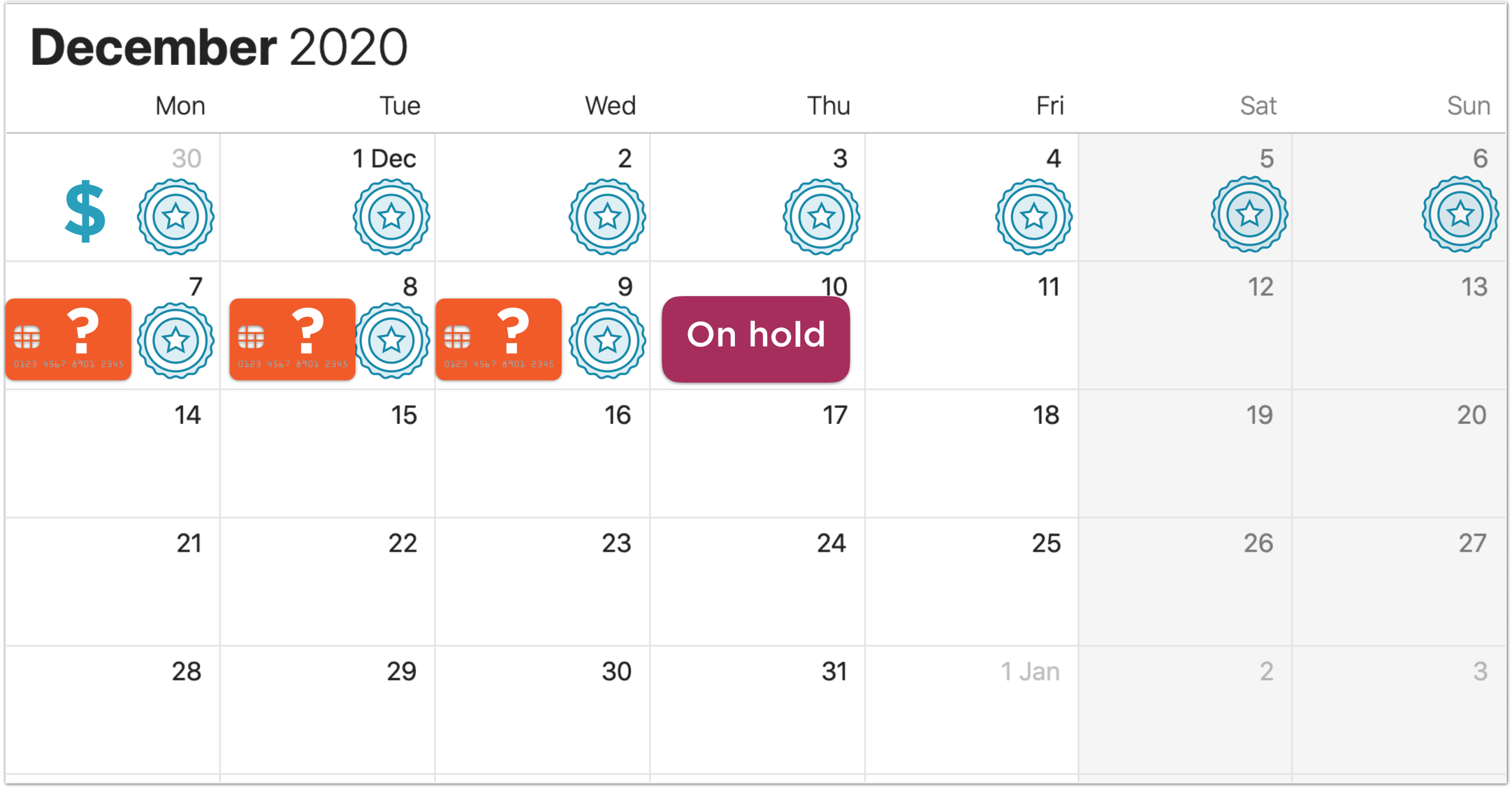
Restarted

Recovered

Deferred

Price change confirmed

Grace Period -> on Hold



Reflect Subscription State Change

Google Play Billing Library

Real-time developer notifications

Real-time Developer Notifications

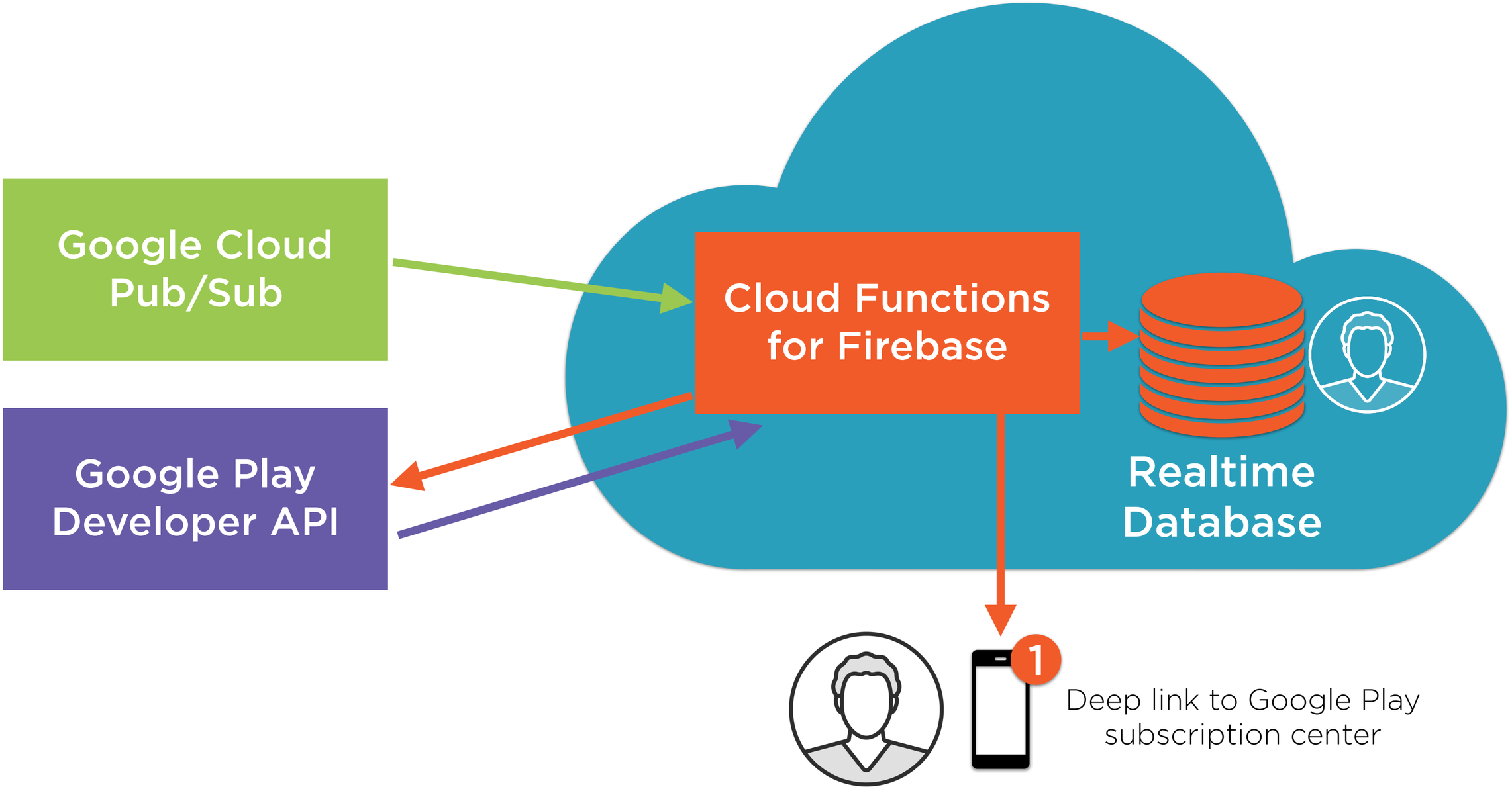
Receive notifications from Google

Google Cloud Pub/Sub

Real-time messaging service

Publish push notifications on topics

System with Real-time Developer Notification



System Configuration

Configure Real-time developer notifications

- Setup Cloud Pub/Sub
- Enable Real-time developer notifications for the app

Handle notifications in cloud functions

Demo

Configure Real-time developer notifications

- Setup Cloud Pub/Sub
- Configure Real-time developer notifications for the app

Demo

Handle notifications in cloud functions

To-do List

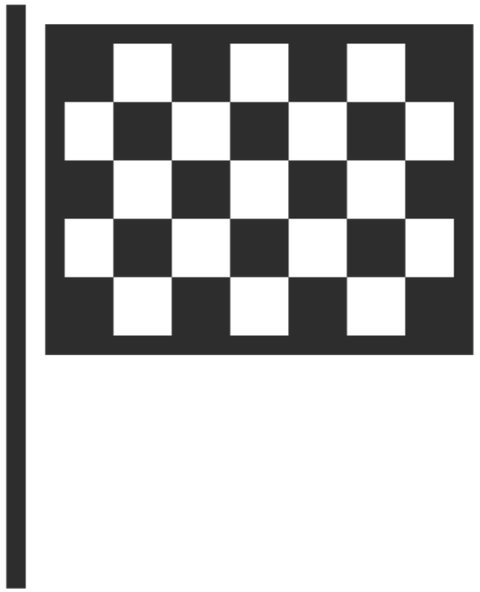
Get updated subscription purchase

Find the user who has made the purchase

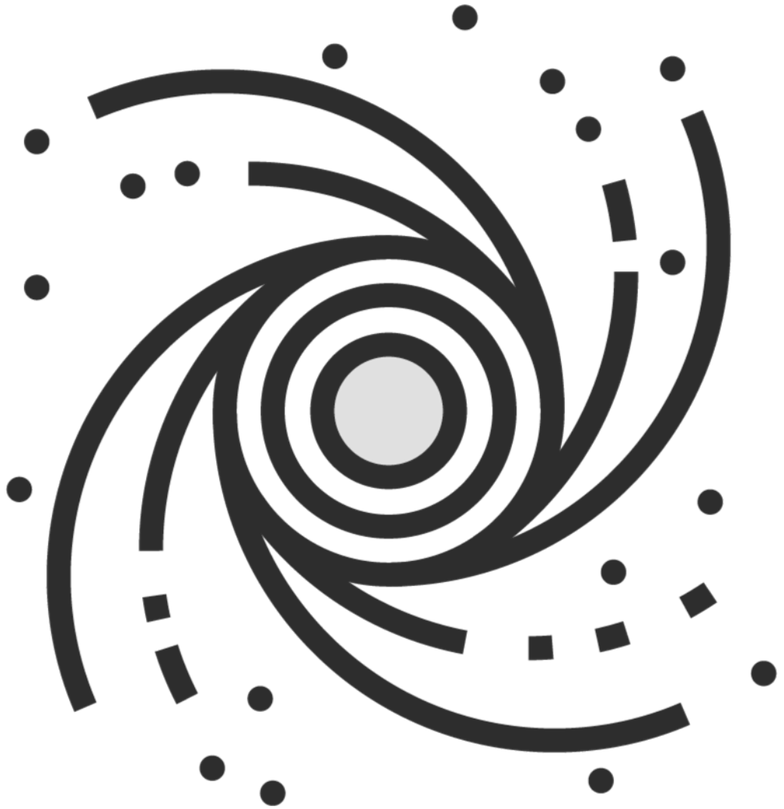
Update the user’s entitlement

Inform the user about the subscription change

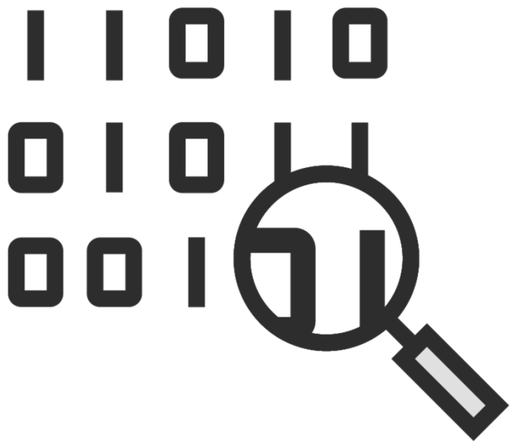
Are We Done?



Risks for an App in Production



Risk 1: Pirate Copy



Steps to Hack an App

Download the apk file

Reverse engineering it

- https://github.com/skylot/jadx

Solution 1: Use Proguard

Solution 2: Protect Sensitive Logic

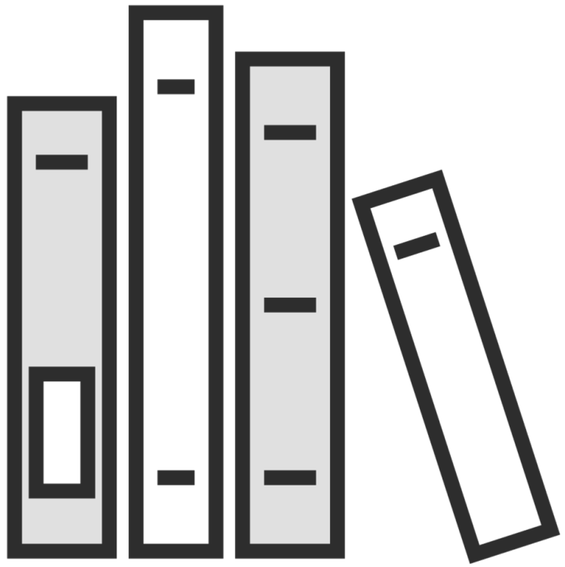
Use NDK

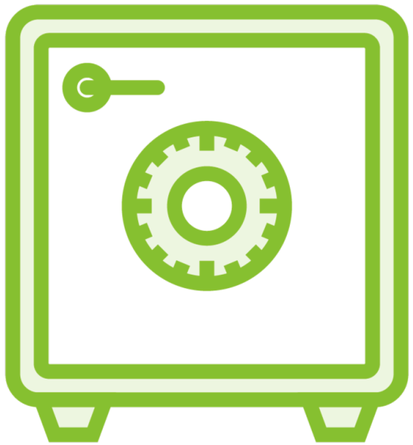
Move the logic to the backend

Risk 2: Break Purchase Verification

Solution : Verify Purchases on Your Backend

Risk 3: Steal Unlocked Content





Solution: Protect Unlocked Content

Store app content on remote servers

Encrypt local unlocked content

Be Secure

Obfuscate the project

- Enable Proguard
- Move sensitive logic to the backend

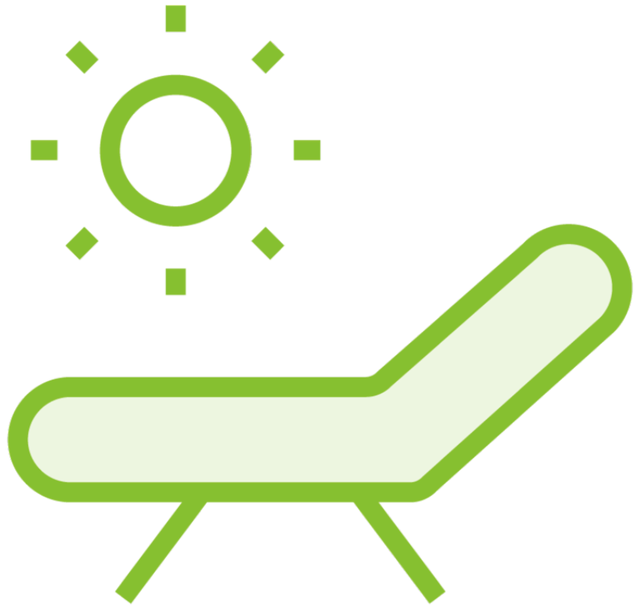
Verify purchase in your backend

Protect unlocked content

- Distribute content on remote servers
- Encrypt unlocked content

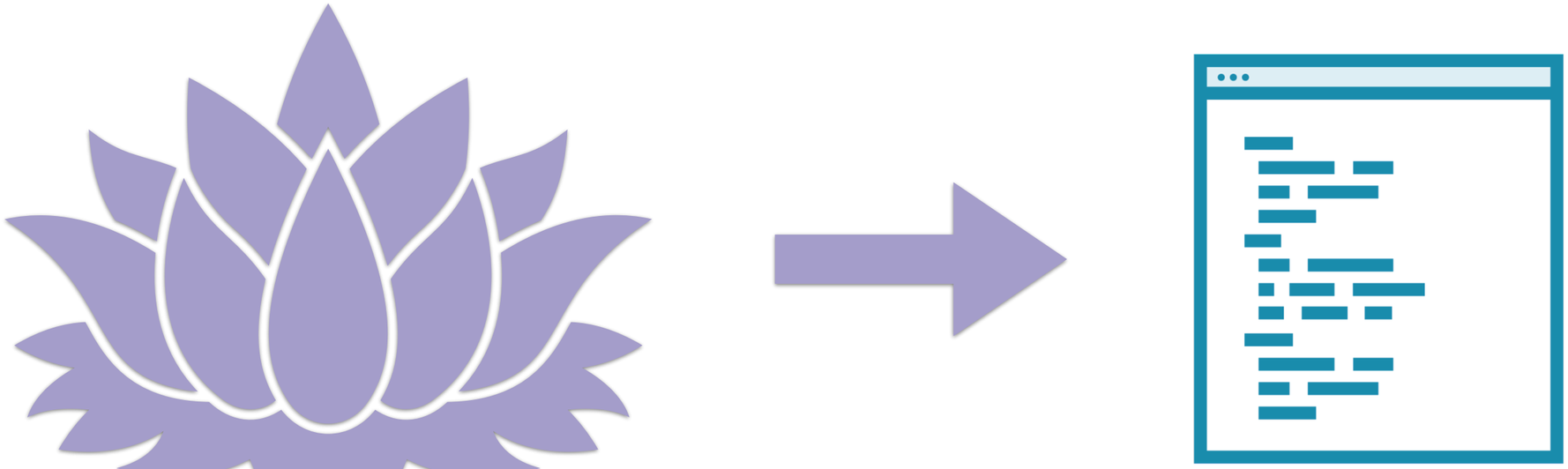
Security vs. Development Cost

Design for Your Fortune



Design a Good User Experience

User Experience First



Marketing Model

A- Attention

Make premium feature obvious

Clear product name and description

I- Interest

Content

Function

User experience design

D- Desire

Basic free features

Enable free trial for premium features

A- Action

Feel at home

Show clear information

Fast delivery of premium features

Good Luck!

Reference - Billing Related

Google Play's Billing System Overview

https://developer.android.com/google/play/billing?hl=ar

REST Resource: purchases.subscriptions

https://developers.google.com/android-publisher/api-ref/rest/v3/purchases.subscriptions

Reference - Cloud Platform Related

Google Play Developer API

https://developers.google.com/android-publisher/getting_started

Real-time developer notifications reference guide

https://developer.android.com/google/play/billing/rtdn-reference

Reference - Firebase Related

Cloud Functions for Firebase

https://firebase.google.com/docs/functions/get-started#create-a-firebase-project

Google Cloud Functions and Firebase

https://cloud.google.com/functions/docs/concepts/functions-and-firebase

Reference - Code Examples

Google’s Sample Code

https://github.com/android/play-billing-samples/tree/master

Google Play IAP Verification Using Cloud Functions

https://medium.com/@msasikanth/google-play-iap-verification-using-cloud-functions-bd8c3a22f9b9

Summary

Backend system

- Verify purchases
- Track subscription state changes

Security consideration

Design for good user experience