

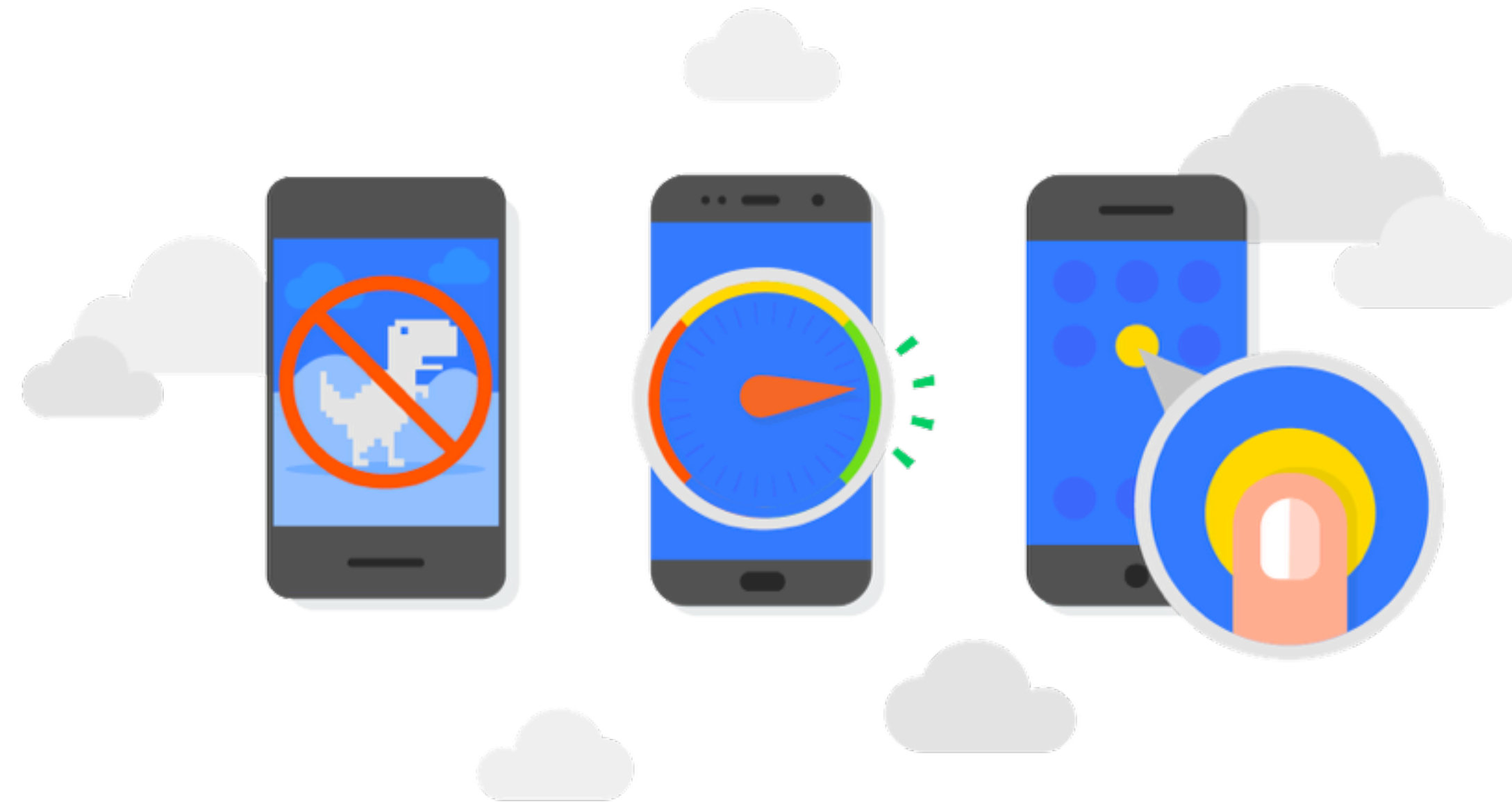
Progressive web apps

Prateek Bhatnagar

@_prateekbh 



What are PWAs



Network Resilient

Responds Fast

Engaging for users

What are you missing

- **Imagine a game is losing its place on homescreen**
- **Multiple businesses are missing notifications**(e.g. sale notifications, delivery notifications, doctor's appointment and lot more)
- **Your web is responding slowly and your apps are already heavy to download**
- **Your web presence goes on and off**

Are you getting a fair chance

80%

OF TIME SPENT IS
IN USERS' TOP 3 APPS

HOW CAN PWAs HELP YOU

Network Resilience

Service workers to build reliable first class offline experiences

Responds Fast

Responding from cache with a fine grained control over it, gives capability to load your content within 2-3 seconds

Engaging for users

Push Notifications and home screen icons help your users engage with your web app

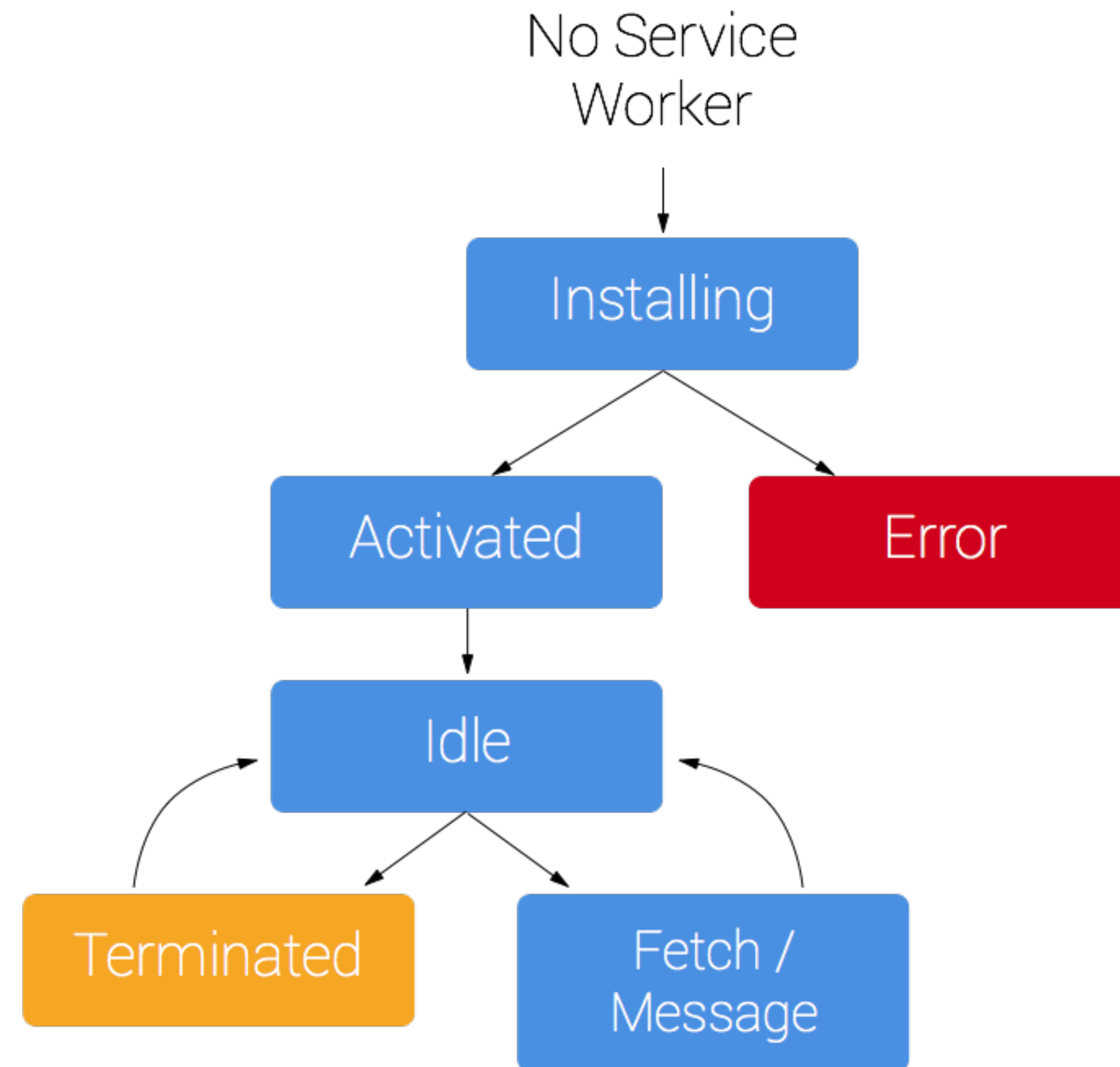
SERVICE WORKERS

The mothership

A service worker is a script that your browser runs in the background, separate from a web page, opening the door to features that don't need a web page or user interaction.

SERVICE WORKERS

The mothership



SERVICE WORKERS

The mothership

```
1 //In index.html
2 if ('serviceWorker' in navigator) {
3     navigator.serviceWorker.register('/sw.js', {scope: '/'}).then(reg=>{
4         console.log("Service worker registered");
5     });
6 }
7
8 //in sw.js
9 // sw does not have access to dom or window
10 // thus self becomes default this
11 self.addEventListener('fetch', (event) => {
12     event.respondWith(new Response('Hello!'));
13 });
14
15
```


Mobile Offline first - Part 1/2

```
1  const CACHE_NAME = 'my-site-cache-v1';
2  const urlsToCache = [
3    '/',
4    '/styles/main.css',
5    '/script/main.js'
6  ];
7
8  self.addEventListener('install', (event) => {
9    // Perform install steps
10   event.waitUntil(
11     caches.open(CACHE_NAME)
12       .then((cache) => {
13         return cache.addAll(urlsToCache);
14       })
15   );
16 });
17
18
```

Mobile Offline first - Part 2/2

```
1 // sw does not have access to dom or window
2 // thus self becomes default this
3 self.addEventListener('fetch', (event) => {
4     event.respondWith(
5         caches.match(event.request)
6         .then((response) => {
7             // Cache hit - return response
8             if (response) {
9                 return response;
10            }
11            return fetch(event.request);
12        })
13    );
14 });
15 });
16
17 // Have the service worker take control as soon as possible.
18 self.addEventListener('install', () => self.skipWaiting());
19 self.addEventListener('activate', () => self.clients.claim());
20
21
22
23
```

Helper tools?

```
1 //sw-toolbox sample
2 //import the library
3 importScripts('/node_modules/sw-toolbox/sw-toolbox.js');
4
5 //allot a strategy to a path or file
6 toolbox.router.get('/', toolbox.cacheFirst);
7
8 //add more options to cache
9 toolbox.router.get('/', toolbox.cacheFirst, {
10     cache: {
11         name: 'googleapis',
12         maxEntries: 10,
13         maxAgeSeconds: 86400
14     },
15     origin: /\.googleapis\.com$/,
16 });
17
18
19
```

So what else then offline shell?

- Your user's last searches
- Their last ride details
- An entire game
- A product page(without buying option may be)
- Your social network's last updated feed
- Your last updated inbox

Manifest.json

Manifest.json is a small description about app in a json file format

```
1 // in your website's head section
2 <link rel="manifest" href="/manifest.json">
3
4 // in manifest.json
5 {
6   "name": "Guess",
7   "short_name": "Guess",
8   "description":
9   "An images based game where you have to guess what is common among all the images.",
10   "theme_color": "#333333",
11   "background_color": "#333333",
12   "display": "fullscreen",
13   "orientation": "portrait",
14   "scope": "/",
15   "start_url": "/?start=home",
16   "gcm_sender_id": "103953800507",
17   "icons": [
18     {
19       "src": "/images/logo.png",
20       "sizes": "192x192",
21       "type": "image/png"
22     }
23   ],
24   "splash_pages": null
25 }
```

Manifest.json

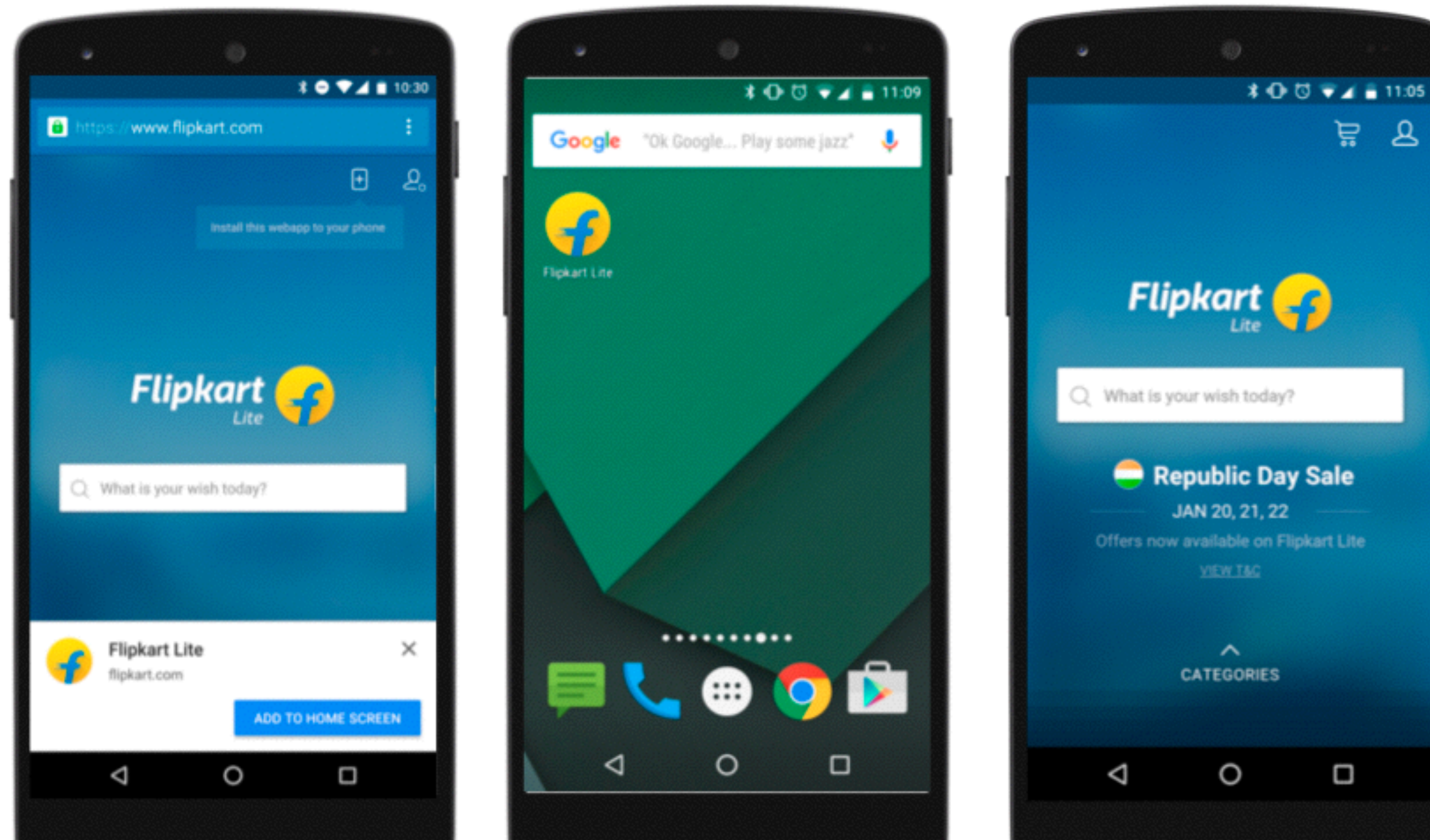
Either follow this spec: <https://w3c.github.io/manifest/>

or use an online manifest generator

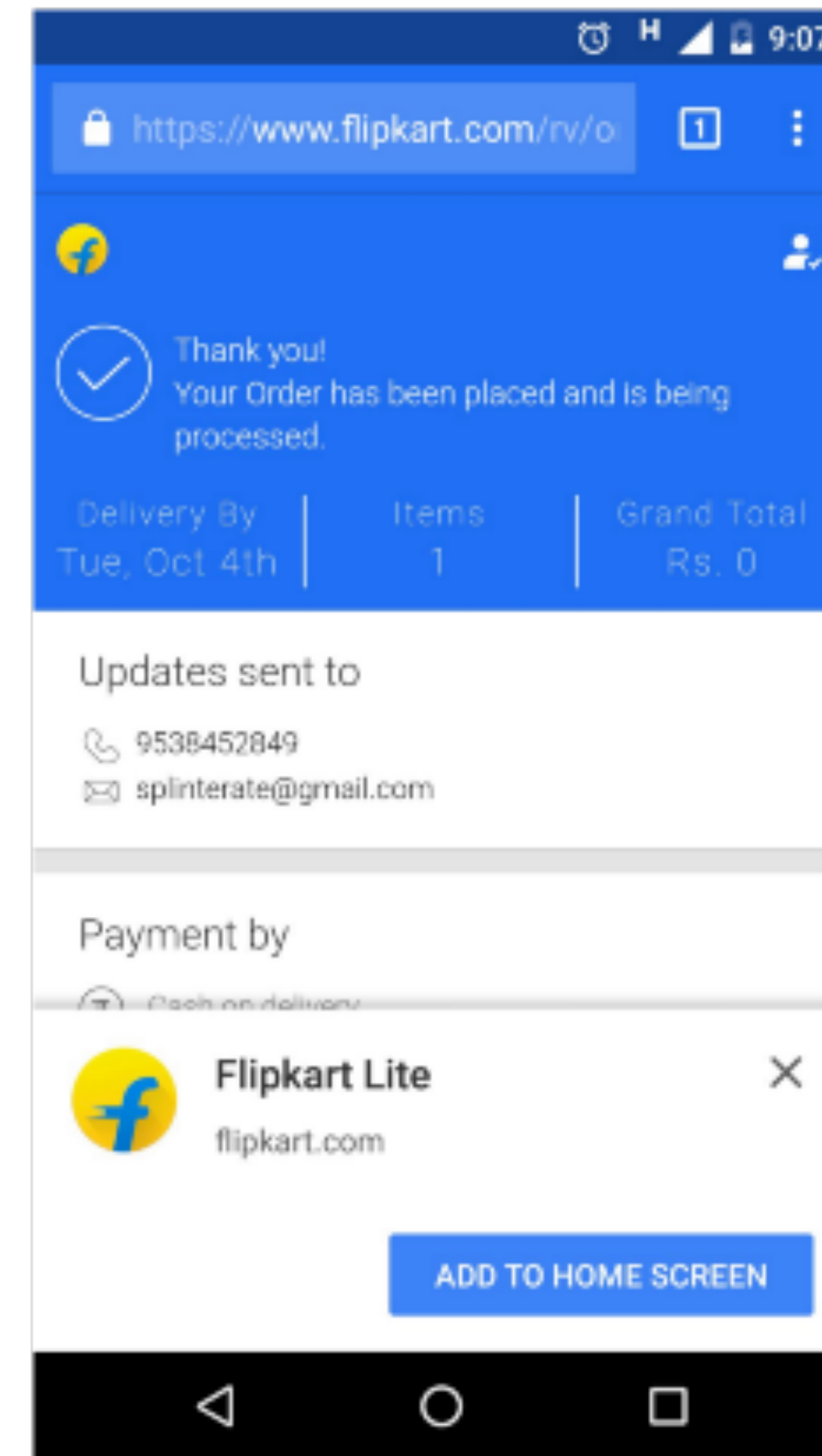
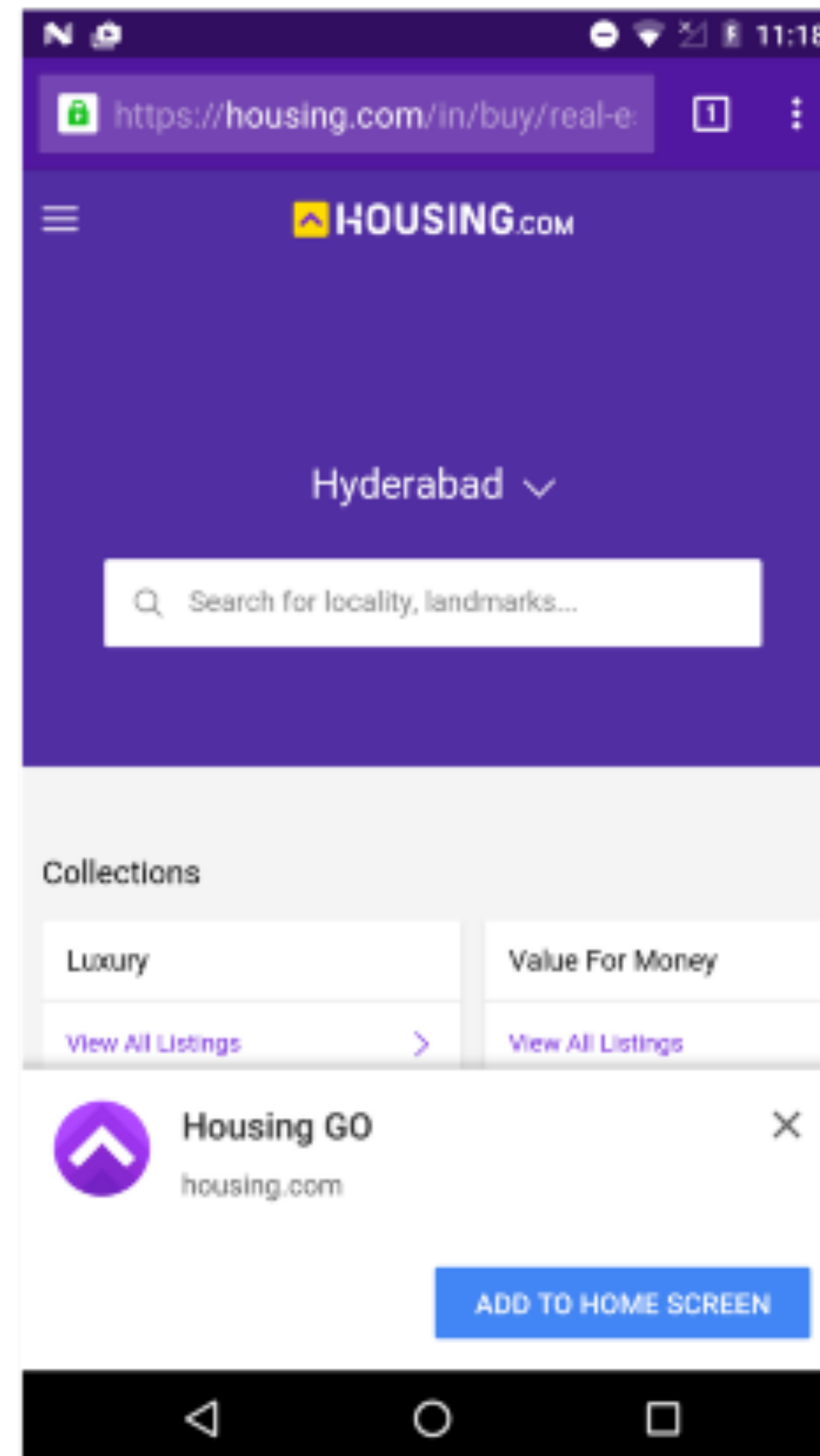
<https://app-manifest.firebaseapp.com/>

<https://tomitm.github.io/appmanifest/>

Homescreen



Homescreen



Homescreen

// in manifest.json

```
{
  "name": "Guess",
  "short_name": "Guess",
  "description":
    "An images based game where you have to guess what is common among all the images.",
  "theme_color": "#333333",
  "background_color": "#333333",
  "display": "fullscreen/standalone/minimal-ui/browser",
  "orientation": "portrait/landscape/...",
  "scope": "/",
  "start_url": "/?start=home",
  "gcm_sender_id": "103953800507",
  "icons": [
    {
      "src": "/images/logo.png",
      "sizes": "192x192",
      "type": "image/png"
    }
  ],
  "splash_pages": null
}
```

Homescreen API

WAIT WHAT NO API?

Homescreen API

Heuristics for Add to home screen

- A valid service worker
- Manifest.json
- User spend sometime on your app

Homescreen API

But what if it doesn't make sense for me?

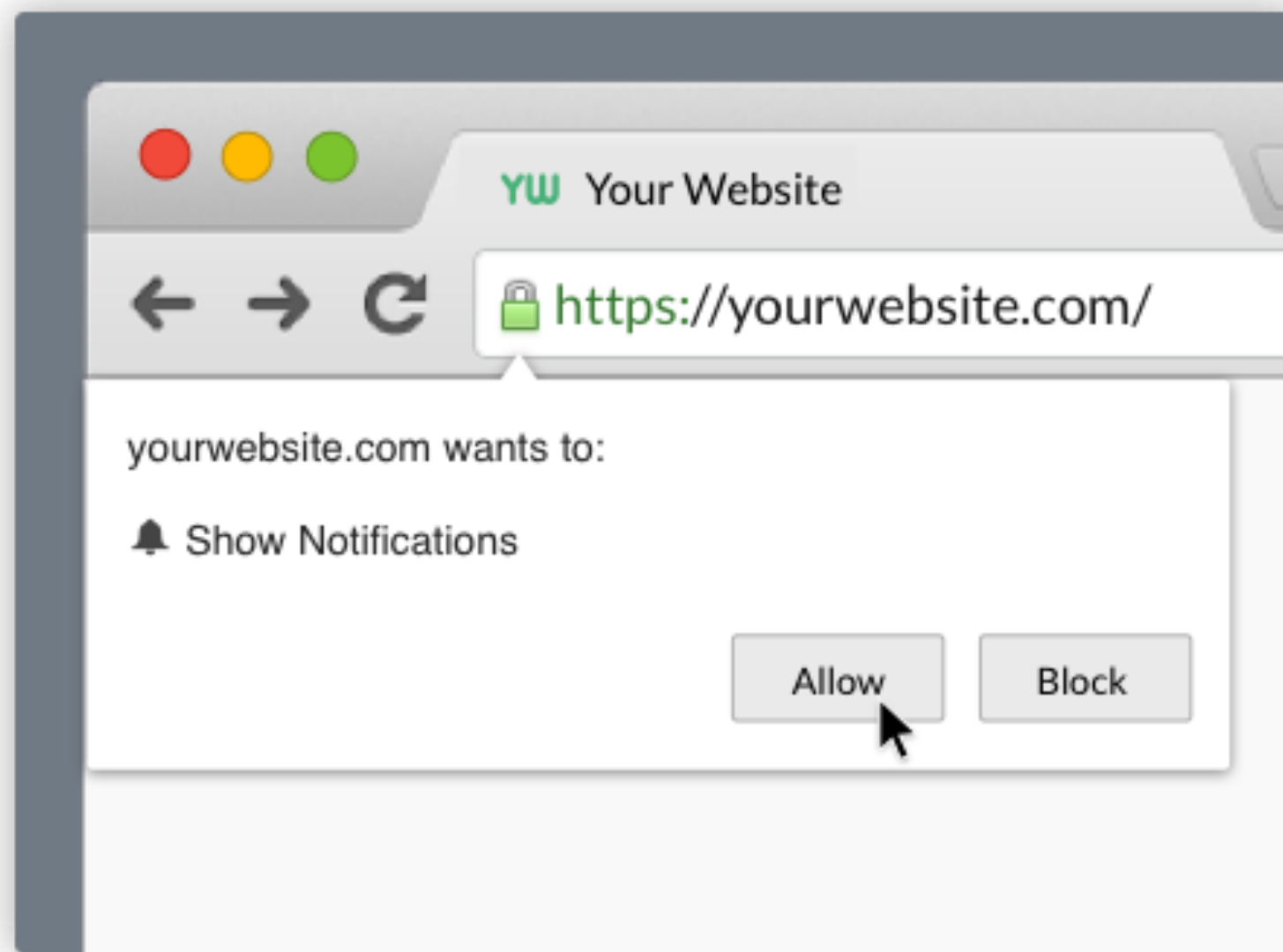
Push notifications



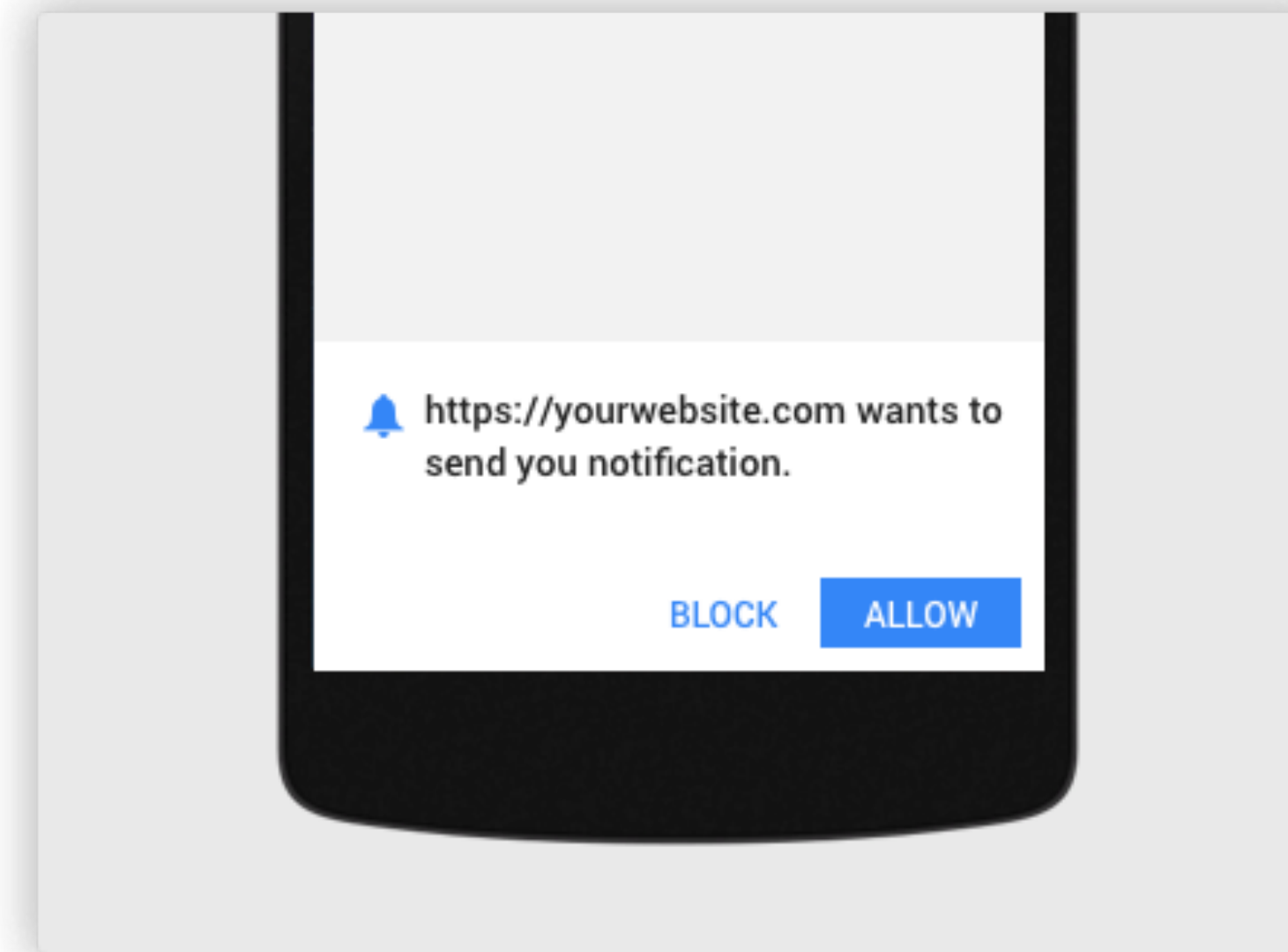
Easily the most awaited feature for web

Push notifications

Guarded by User permissions modal



Desktop Opt-in



Mobile Website Opt-in

Push notifications

Push notification powered by service worker

Push notifications

```
1 // in your sw.js
2 self.addEventListener('push', (event)=> {
3   const title = 'Push Codelab';
4   const options = {
5     body: 'Yay it works.',
6     icon: 'images/icon.png',
7     badge: 'images/icon.png',
8     actions: [
9       {action: 'like', title: 'Like', icon: 'https://example/like.png'},
10      {action: 'reply', title: 'Reply', icon: 'https://example/reply.png'}],
11     image: 'images/banner.png',
12     tag: 'promotion'
13   };
14   event.waitUntil(self.registration.showNotification(title, options));
15 });
16
17
18
```


Push notifications

Firebase your cross browser, push notification friend

```
1 // in your sw.js
2 importScripts('https://www.gstatic.com/firebasejs/3.6.9/firebase-app.js');
3 importScripts('https://www.gstatic.com/firebasejs/3.6.9/firebase-messaging.js');
4
5 firebase.initializeApp({
6   'messagingSenderId': '678578574774'
7 });
8
9 const messaging = firebase.messaging();
10
11 messaging.setBackgroundMessageHandler(function(payload) {
12   console.log('[firebase-messaging-sw.js] Received background message ', payload);
13   // Customize notification here
14   const notificationTitle = 'My App!';
15   const notificationOptions = {
16     body: 'Firebase notif says: ' + payload.data.msg,
17     icon: '/images/icon.png'
18   };
19
20   return self.registration.showNotification(notificationTitle,
21     notificationOptions);
22 });
23
```

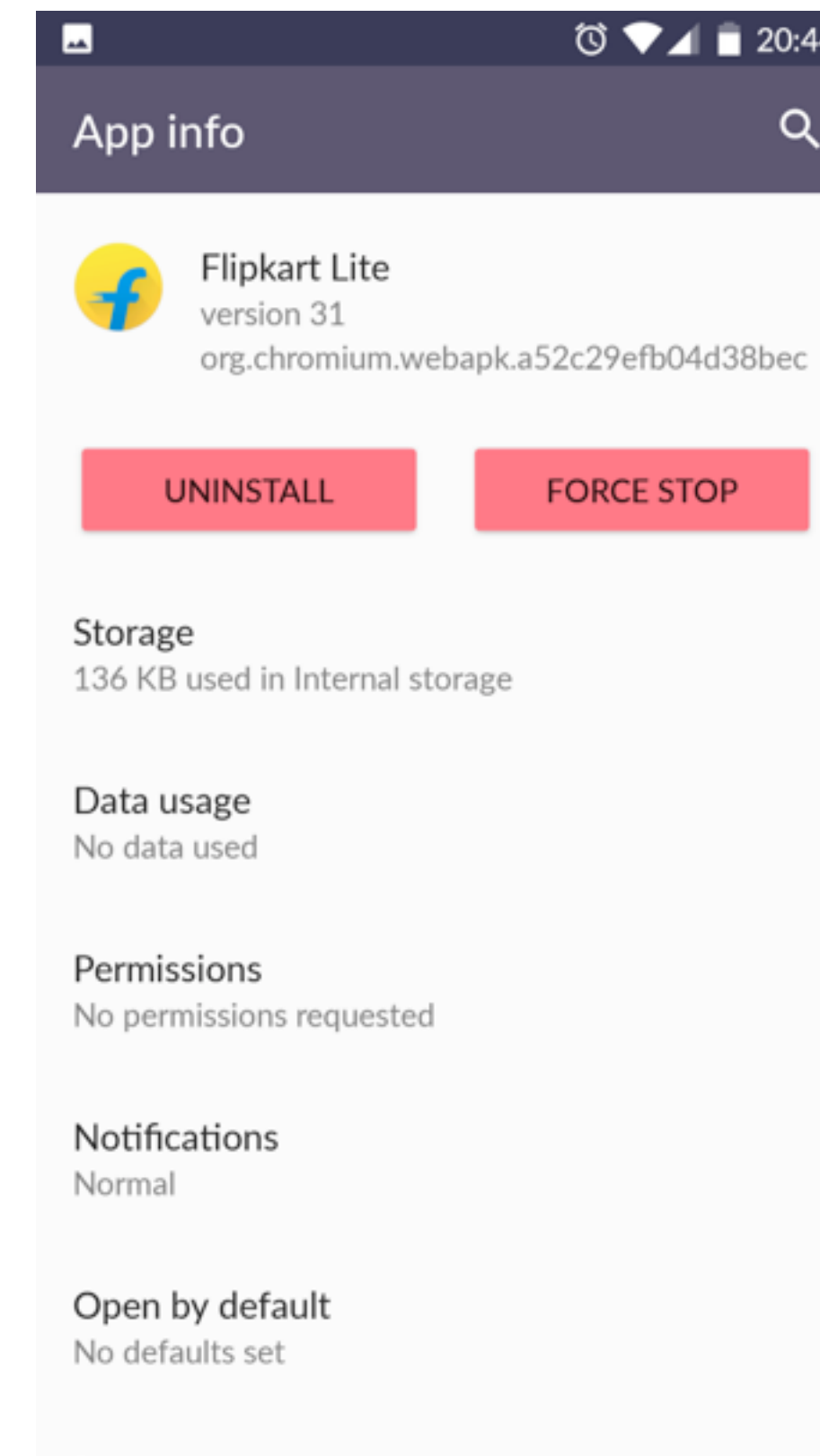
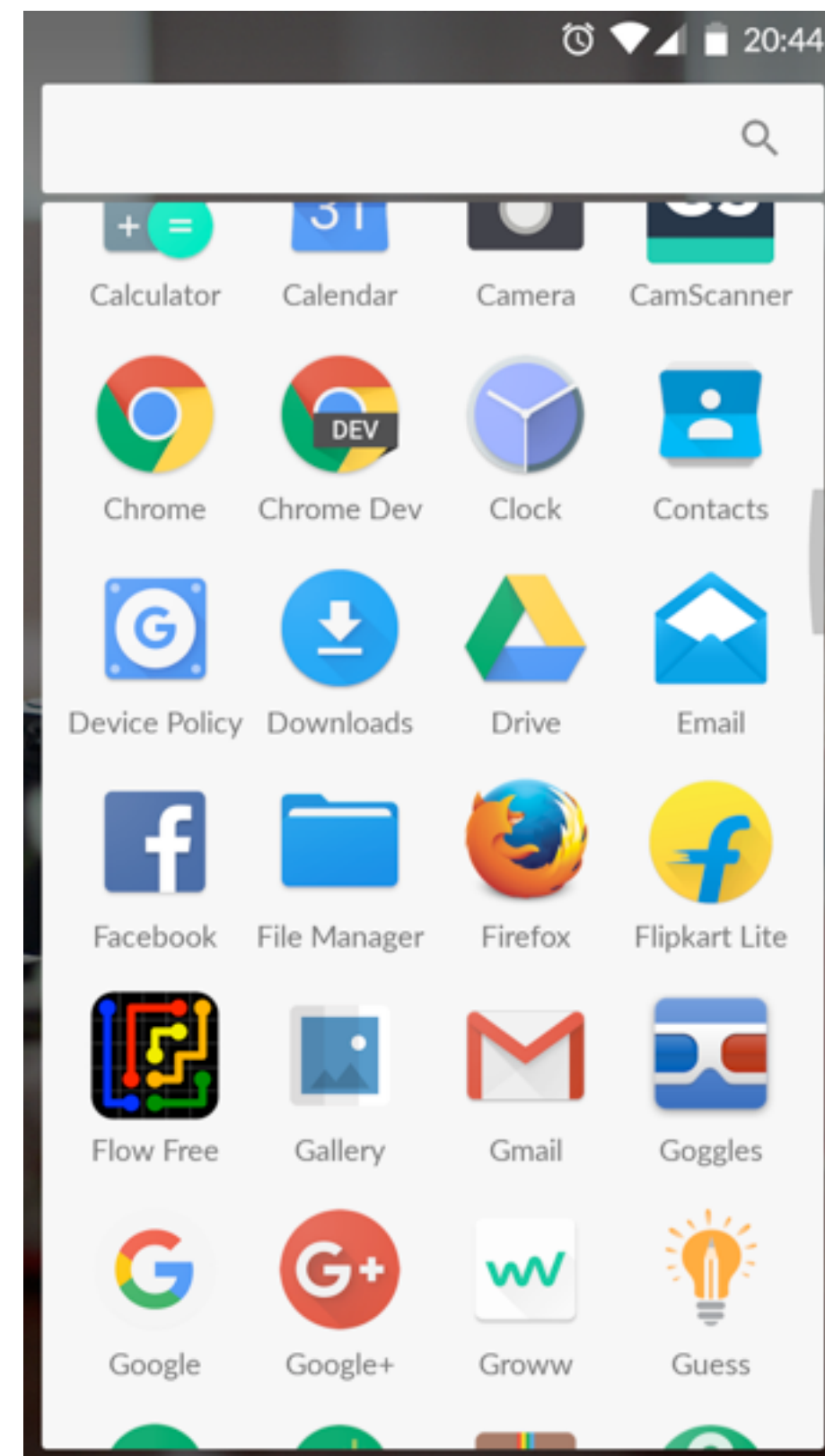
Push notifications

Firebase your cross browser, push notification friend

```
1 // in your index.html
2 const messaging = firebase.messaging();
3
4 messaging.requestPermission()
5 .then(function(currentToken) {
6   console.log('Notification permission granted. ');
7   messaging.getToken()
8   .then(function(currentToken) {
9     if (currentToken) {
10       console.log(currentToken);
11     } else {
12       // Show permission request.
13       console.log('No Instance ID token available. Request permission to generate one. ');
14     }
15   })
16   .catch(function(err) {
17     console.log('An error occurred while retrieving token. ', err);
18   });
19 })
20 .catch(function(err) {
21   console.log('Unable to get permission to notify.', err);
22 });
23
24
```

What more can you expect

Improved Add to home-screen



What more can you expect

Replace your images for webP

What more can you expect

Background sync

Some more tools



Some more tools

Lighthouse
Version: 1.6.5

BETA

Progressive Web App

Best Practices

Performance

Fancier stuff

> Results for: <https://playguess.herokuapp.com/>
Generated on: 5/4/2017, 7:40:39 PM GMT+5:30

Export...

97
100

Progressive Web App

These audits validate the aspects of a Progressive Web App. They are a subset of the [PWA Checklist](#).

✓

App can load on offline/flaky connections >

✓

Page load performance is fast ▾

Users notice if sites and apps don't perform well. These top-level metrics capture the most important perceived performance concerns.

90

First meaningful paint: **2052.5ms** (target: 1,600ms) ?

26

Perceptual Speed Index: **8719** (target: 1,250) ?

First Visual Change: **8710ms**

Last Visual Change: **10113ms**

100

Estimated Input Latency: **16ms** (target: 50ms) ?

92

Time To Interactive (alpha): **2279.6ms** (target: 5,000ms) ?

✓

Site is progressively enhanced >

✓

Network connection is secure >

✓

User can be prompted to Add to Homescreen >

✓

Installed web app will launch with custom splash screen >

✓

Address bar matches brand colors >

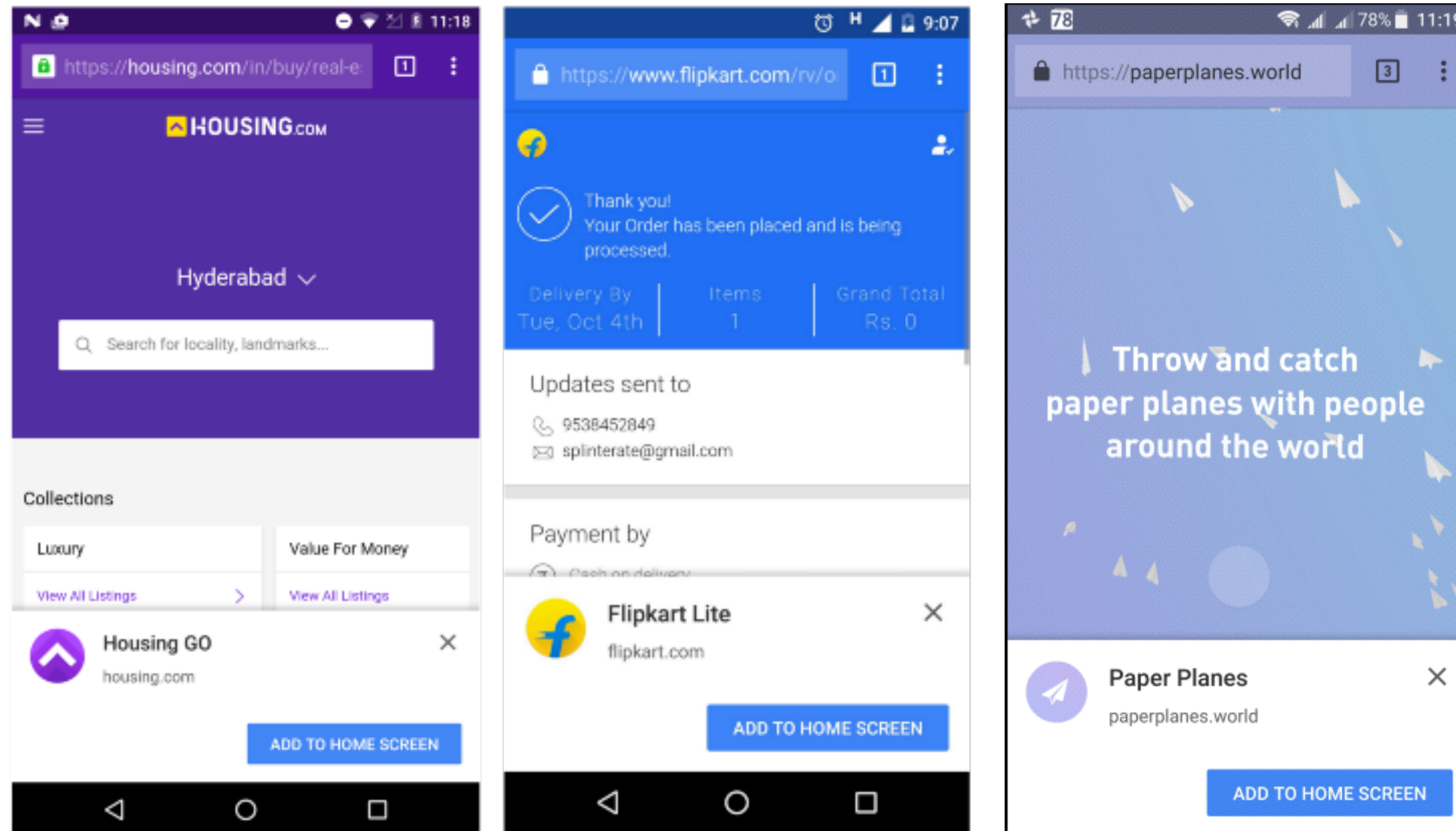
✓

Design is mobile-friendly >

Browser Support



Some examples I know



<http://outweb.io/>

Thank you