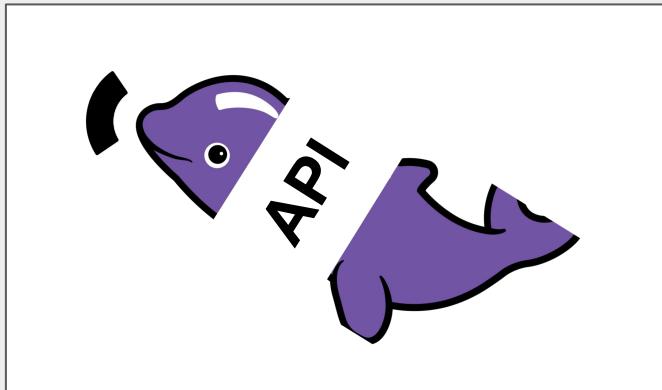
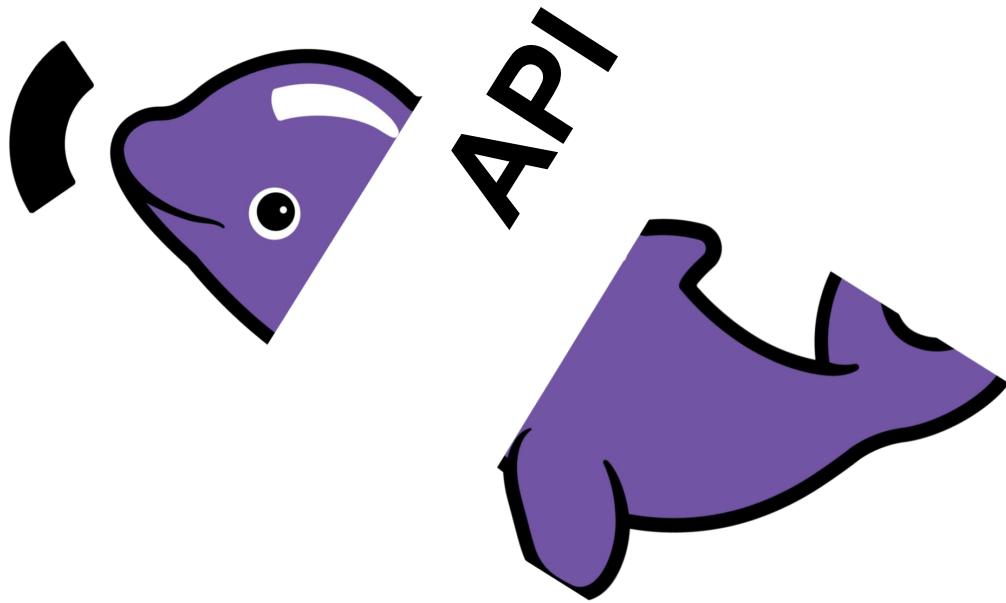


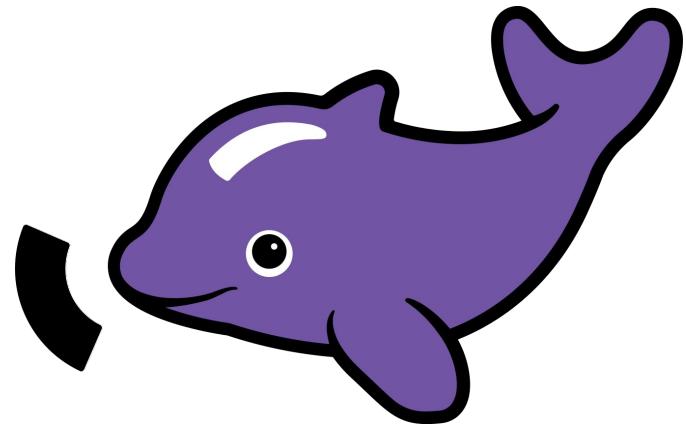
API

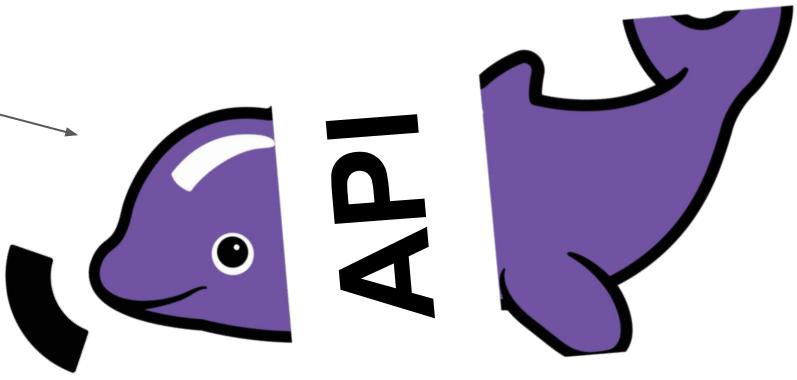
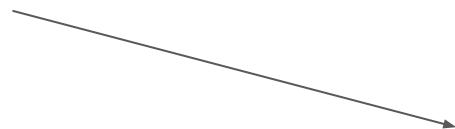
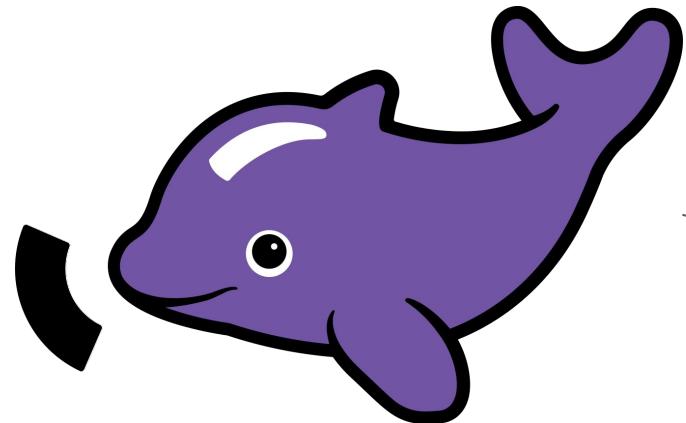
MY AWESOME CI INFRA



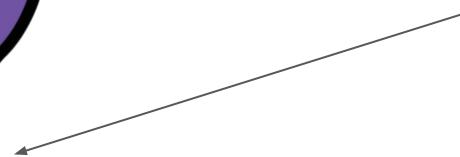
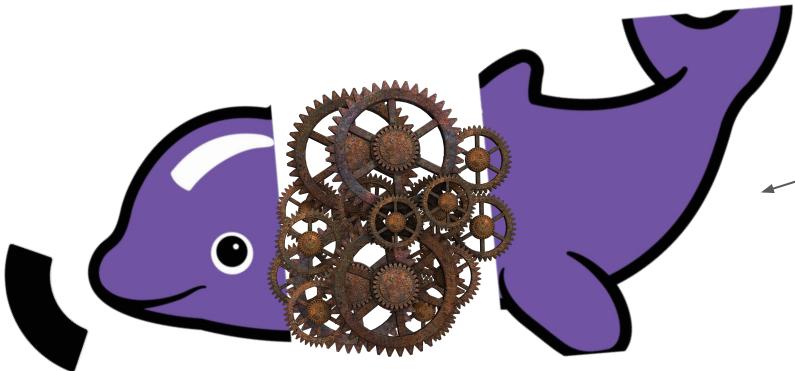
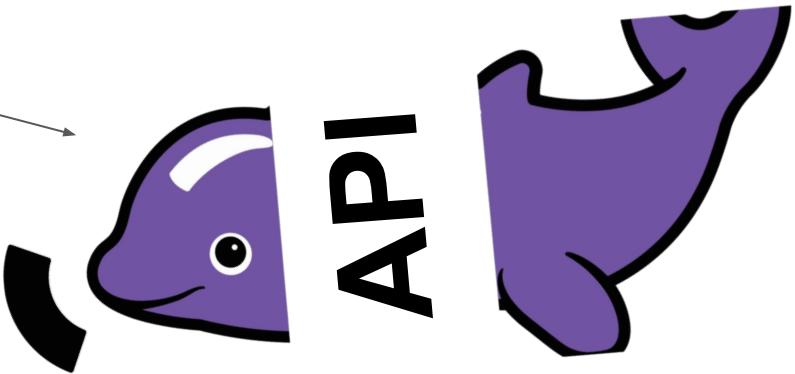
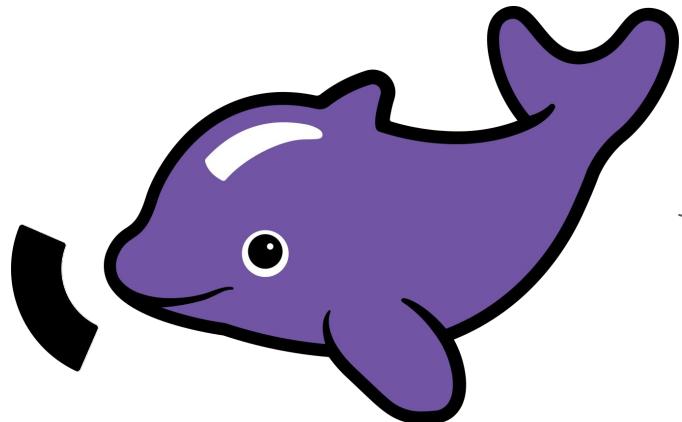








API



FLIPPER IS ...

Flipper is a platform for debugging iOS, Android and React Native apps. Visualize, inspect, and control your apps from a simple desktop interface. Use Flipper as is or extend it using the plugin API. Flipper aims to be your number one companion for mobile app development on iOS and Android. Therefore, we provide a bunch of useful tools including a log viewer, interactive layout inspector, and network inspector. Flipper is built as a platform. In addition to using the tools already included, you can create your own plugins to visualize and debug data from your mobile apps. Flipper takes care of sending data back and forth, calling functions, and listening for events on the mobile app. Both Flipper's desktop app and native mobile SDKs are open-source and MIT licensed. This enables you to see and understand how we are building plugins, and of course join the community and help improve Flipper. We are excited to see what you will build on this platform. Desktop App Flipper helps you debug in the following environments: Android and iOS. Web apps running in an emulator/simulator. Connected physical development devices. Your browser. Flipper consists of two parts: The desktop app. The native mobile SDKs for Android and iOS, the client for JavaScript, or even a third-party client you could implement yourself or find on the web. Once you start Flipper and launch an emulator/simulator or connect a device, you'll start to see the device logs (and any other device-level plugins that work with your device). Currently, there are no plugins available for web apps. To see app-specific data, you need to integrate the Flipper SDK into your app (see the 'Adding Flipper to your app' within the 'Getting Started' section of the SideBar). Installation NOTE The desktop part of Flipper doesn't need a setup. Simply download the latest build for Mac, Linux or Windows and launch it. If you're on macOS, you can run brew install --cask flipper to let homebrew manage installation and upgrades (simply run brew upgrade to upgrade when a new version is released, although it might take a few hours up to a day for the package to be upgraded on homebrew). To work properly with mobile apps, Flipper requires the following: Working installation of Android development tools [Where applicable] Working installation of iOS development tools OpenSSL binary on your \$PATH. A compatible OpenSSL for Windows can be downloaded from slproweb.com or from Chocolatey with choco install openssl. If you are hacking a JS app, you should be good to go without any extra dependencies installed. RMATION [Experimental] Alternatively, it is possible to run a browser based version of Flipper directly from NPM by using npx flipper-server. Troubleshooting If you run into problems, take a look at the troubleshooting page. Failing that, check GitHub Issues. Adding Flipper to Android apps with Gradle To set up Flipper for Android, you need to add the necessary dependencies to your app, initialize the Flipper client and enable the plugins you want to use. Optionally, you can hook up the diagnostics Activity to help you troubleshoot connection issues. Dependencies Flipper is distributed via Maven Central: add the dependencies to your build.gradle file. You should also explicitly depend on SoLoader instead of relying on transitive dependency resolution, which is getting deprecated with Gradle 5. There is a 'no-op' implementation of some oft-used Flipper interfaces, which you can use to make it easier to strip Flipper from your release builds: repositories { mavenCentral() } dependencies { debugImplementation 'com.facebook.flipper:flipper:0.145.0' debugImplementation 'com.facebook.soloader:soloader:0.10.1' releaseImplementation 'com.facebook.flipper:flipper-noop:0.145.0' } WARNING The flipper-noop package provides a limited subset of the APIs provided by the flipper package and does not provide any plugin stubs. It's recommended that you keep all Flipper instantiation code in a separate build variant to ensure it doesn't accidentally make it into your production builds. To see how to organise your Flipper initialization into debug and release variants. see thos sample app. Alternatively, have a look at the third-party flipper-android-no-op repository, which provides empty implementations for several Flipper plugins. Application setup Now you can initialize Flipper in your Application's onCreate method, which involves initializing SoLoader (for loading the C++ part of Flipper) and starting a FlipperClient. Kotlin Java import com.facebook.flipper.android.AndroidFlipperClient import com.facebook.flipper.android.utils.FlipperUtils import com.facebook.flipper.core.FlipperClient import com.facebook.flipper.plugins.inspector.DescriptorMapping import com.facebook.flipper.plugins.inspector.InspectorFlipperPlugin class MyApplication : Application { override fun onCreate() { super.onCreate() SoLoader.init(this, false) if (BuildConfig.DEBUG && FlipperUtils.shouldEnableFlipper(this)) { val client = AndroidFlipperClient.getInstance(this) client.addPlugin(InspectorFlipperPlugin(this, DescriptorMapping.withDefaults())) client.start() } } } Diagnostics It's recommended that you add the following activity to the manifest, which can help diagnose integration issues and other problems: <activity android:name="com.facebook.flipper.android.diagnostics.FlipperDiagnosticActivity" android:exported="true"/> Android snapshots NOTE Android snapshot releases are published directly off main. You can get the latest version by adding the Maven Snapshot repository to your sources and pointing to the most recent -SNAPSHOT version. repositories { maven { url 'https://oss.sonatype.org/content/repositories/snapshots/' } } dependencies { debugImplementation 'com.facebook.flipper:flipper:0.145.1-SNAPSHOT' debugImplementation 'com.facebook.soloader:soloader:0.10.1' releaseImplementation 'com.facebook.flipper:flipper-noop:0.145.1-SNAPSHOT' } Enabling plugins Finally, you need to add plugins to your Flipper client. Above, the Layout Inspector plugin has been added to get you started. See the Network Plugin and Layout Inspector Plugin pages for information on how to add them, and also enable Litho or ComponentKit support. For examples of integrating other plugins, take a look at the sample apps in the GitHub repo.

DevTools - docs.google.com/presentation/d/1JXF9Lfn ds2ihEgCCv1gPea6ODh3BZZSkLgcIWE09aGU/edit

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Facebook Slog Redux

Preserve log Disable cache No throttling Invert Hide data URLs All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies Blocked Requests 3rd-party requests

Filter 5 ms 10 ms 15 ms 20 ms 25 ms 30 ms 35 ms 40 ms 45 ms 50 ms 55 ms 60 ms 65 ms 70 ms 75 ms 80 ms 85 ms 90 ms 95 ms 100 ms 105 ms 11

Recording network activity...

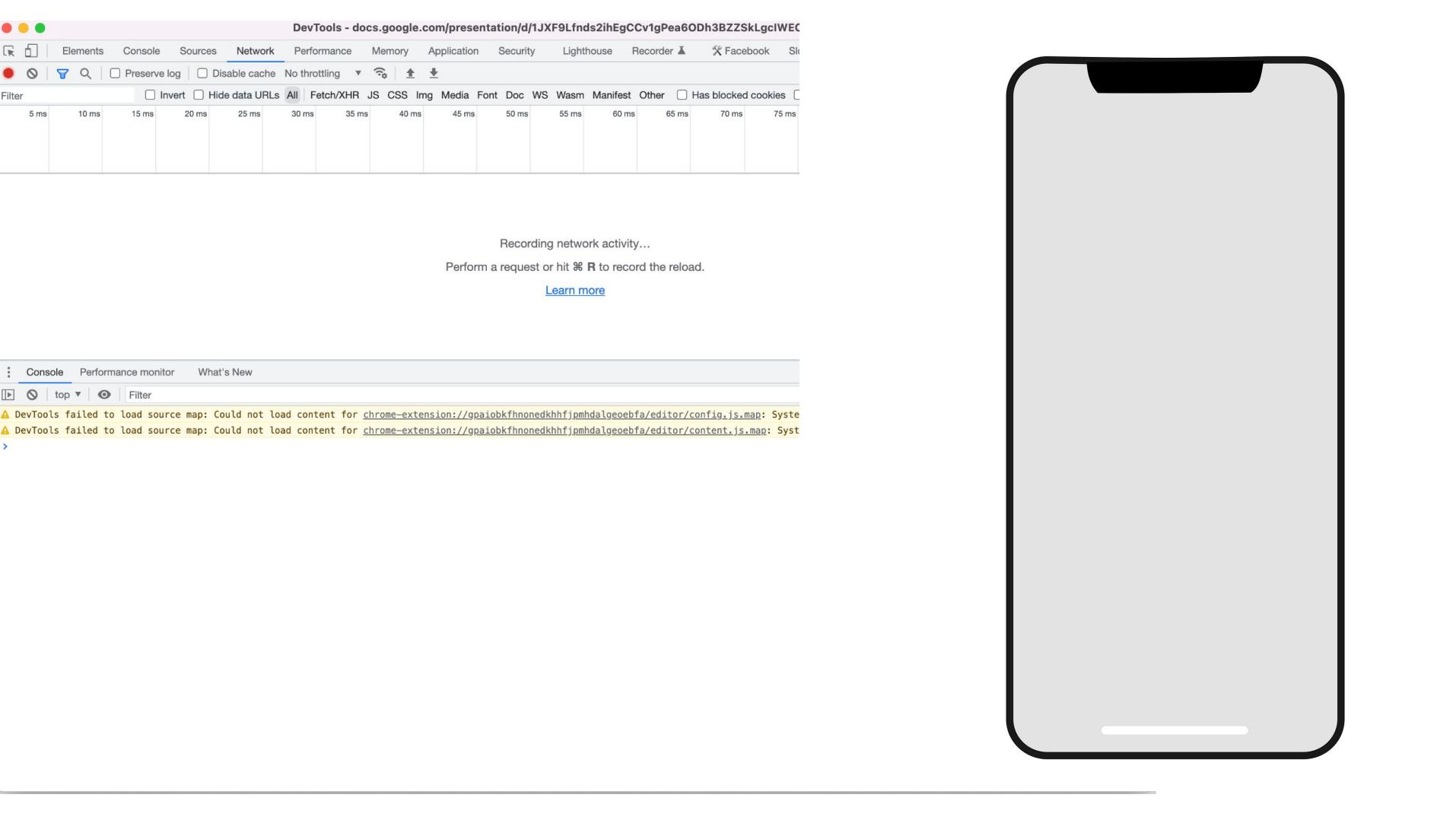
Perform a request or hit ⌘ R to record the reload.

[Learn more](#)

Console Performance monitor What's New

All levels 8 Issues: 1 7

⚠ DevTools failed to load source map: Could not load content for [chrome-extension://gpaiobkfhnonedkhhfjpmhhalgeoebfa/editor/config.js.map](#): System error: net::ERR_BLOCKED_BY_CLIENT
⚠ DevTools failed to load source map: Could not load content for [chrome-extension://gpaiobkfhnonedkhhfjpmhhalgeoebfa/editor/content.js.map](#): System error: net::ERR_BLOCKED_BY_CLIENT



APP INSPECT

ReactNativeFlipp... Pixel_4a_API_32

Troubleshooting Guide

PLUGINS

Device 3

- Crash Reporter
- Logs
- Mobile Builds

Enabled 3

- Layout
- Network
- React Native Tic Tac Toe

Detected in App 6

Unavailable plugins 99+

Open With

Big Grep

ViewGroup

- clipChildren:
- clipToPadding:
- layoutMode: LAYOUT_MODE_CLIP_BOUNDS

View

```

alpha: 1
background: (not set)
bounds: {bottom: 2296, left: 0, right: 1080, top: 0}
elevation: 0
foreground: (not set)
height: 2296
keyedTags: {}
layoutDirection: LAYOUT_DIRECTION_LTR
layoutParams: {gravity, height: MATCH_PARENT, margin, width: MATCH_PARENT}
padding: {bottom: 0, left: 0, right: 0, top: 66}
pivot: {x: 540, y: 1148}
position: {x: 0, y: 0, z: 0}
positionOnScreenX: 0
positionOnScreenY: 0
rotation: {x: 0, y: 0, z: 0}
scale: {x: 1, y: 1}
state: {activated: false, enabled: true, focused: false, selected: false}
tag: undefined
textAlignment: TEXT_ALIGNMENT_GRAVITY
textDirection: TEXT_DIRECTION_FIRST_STRONG
translation: {x: 0, y: 0, z: 0}
visibility: VISIBLE
width: 1080

```

Theme

APP INSPECT

ReactNativeFlipp... Pixel_4a_API_32

Troubleshooting Guide

PLUGINS

Device 3

- Crash Reporter
- Logs
- Mobile Builds

Enabled 3

- Layout
- Network**
- React Native Tic Tac Toe

Detected in App 6

Unavailable plugins 99+

Search...

Request Time	Domain	Method	Status	Request Size	Response Size	Time
21:54:51.667	status.npmjs.org/	GET	200	0 B	5.1 kB	261ms
21:54:51.915	status.npmjs.org/	GET	200	0 B	5.0 kB	76ms
21:54:52.787	status.npmjs.org/	GET	304	0 B	0 B	66ms

Request

Key	Value
Full URL	https://status.npmjs.org/
Host	status.npmjs.org
Path	/
Query String	

Request Headers

Key	Value
accept	application/json
Accept-Encoding	gzip
Connection	Keep-Alive
Host	status.npmjs.org
If-None-Match	W/"gz[4bf89424df6bab0edee52568a23a4a61]"
User-Agent	okhttp/4.9.3

Response Headers

Key	Value
date	Tue, 14 Jun 2022 19:54:52 GMT
etag	W/"gz[4bf89424df6bab0edee52568a23a4a61]"

Response Body

(empty)

Options

APP INSPECT

ReactNativeFlipp... Pixel_4a_API_32

Troubleshooting Guide

1

PLUGINS

Device 3 ^

- Crash Reporter**
- Logs**
- Mobile Builds**

Enabled 3 ^

- Layout**
- Network**
- React Native Tic Tac Toe**

Detected in App 6 ^

- Databases**
- Headless-demo**
- Images**
- Sea Mammals**
- Shared Preferences Viewer**

Trace

Unavailable plugins 99+ ^

Search...

Time	PID	Tag	Message
21:55:28.281	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=2 ai_socktype=1 ai_protocol=6
21:55:28.281	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=10 ai_socktype=1 ai_protocol=6
21:55:28.281	342	resolv	explore_numeric_scope
21:55:28.281	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=2 ai_socktype=1 ai_protocol=6
21:55:28.281	342	resolv	explore_numeric_scope
21:55:30.293	342	resolv	GetAddrInfoHandler::run: {101 101 101 983141 10149 0}
21:55:30.293	342	resolv	resolv_getaddrinfo: explore_fqdn(): ai_family=0 ai_socktype=1 ai_protocol=6
21:55:30.293	342	resolv	GetAddrInfoHandler::run: {101 101 101 983141 10149 0}
21:55:30.293	342	resolv	resolv_getaddrinfo: explore_fqdn(): ai_family=0 ai_socktype=1 ai_protocol=6
21:55:30.293	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=10 ai_socktype=1 ai_protocol=6
21:55:30.293	342	resolv	explore_numeric_scope
21:55:30.293	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=2 ai_socktype=1 ai_protocol=6
21:55:30.293	342	resolv	explore_numeric_scope
21:55:30.293	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=10 ai_socktype=1 ai_protocol=6
21:55:30.293	342	resolv	explore_numeric_scope
21:55:30.293	342	resolv	explore_numeric_scope
21:55:30.293	342	resolv	GetAddrInfoHandler::run: {101 101 101 983141 10149 0}
21:55:32.303	342	resolv	resolv_getaddrinfo: explore_fqdn(): ai_family=0 ai_socktype=1 ai_protocol=6
21:55:32.303	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=10 ai_socktype=1 ai_protocol=6
21:55:32.303	342	resolv	explore_numeric_scope
21:55:32.303	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=2 ai_socktype=1 ai_protocol=6
21:55:32.303	342	resolv	explore_numeric_scope
21:55:32.303	342	resolv	GetAddrInfoHandler::run: {101 101 101 983141 10149 0}
21:55:32.303	342	resolv	resolv_getaddrinfo: explore_fqdn(): ai_family=0 ai_socktype=1 ai_protocol=6
21:55:32.303	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=10 ai_socktype=1 ai_protocol=6
21:55:32.303	342	resolv	explore_numeric_scope
21:55:32.303	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=2 ai_socktype=1 ai_protocol=6
21:55:32.303	342	resolv	explore_numeric_scope
21:55:34.306	342	resolv	GetAddrInfoHandler::run: {101 101 101 983141 10149 0}
21:55:34.306	342	resolv	resolv_getaddrinfo: explore_fqdn(): ai_family=0 ai_socktype=1 ai_protocol=6
21:55:34.306	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=10 ai_socktype=1 ai_protocol=6
21:55:34.306	342	resolv	explore_numeric_scope
21:55:34.306	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=2 ai_socktype=1 ai_protocol=6
21:55:34.306	342	resolv	explore_numeric_scope
21:55:34.306	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=10 ai_socktype=1 ai_protocol=6
21:55:34.306	342	resolv	explore_numeric_scope
21:55:34.306	342	resolv	android_getaddrinfofornetcontext: explore_numeric: ai_family=2 ai_socktype=1 ai_protocol=6

APP INSPECT

ReactNativeFlipp... Pixel_4a_API_32

Troubleshooting Guide

1

PLUGINS

Device 3

- Crash Reporter
- Logs
- Mobile Builds

Enabled 3

- Layout
- Network

React Native Tic Tac Toe

Detected in App 6

- Databases
- Headless-demo
- Images
- Sea Mammals
- Shared Preferences Viewer

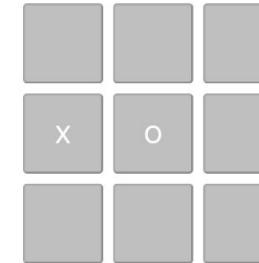
Trace

Unavailable plugins 99+

This plugin demonstrates how to create pure JavaScript Flipper plugins for React Native. Find out how to create a similar plugin at fbflipper.com.

Flipper Tic-Tac-Toe

Mobile players turn..

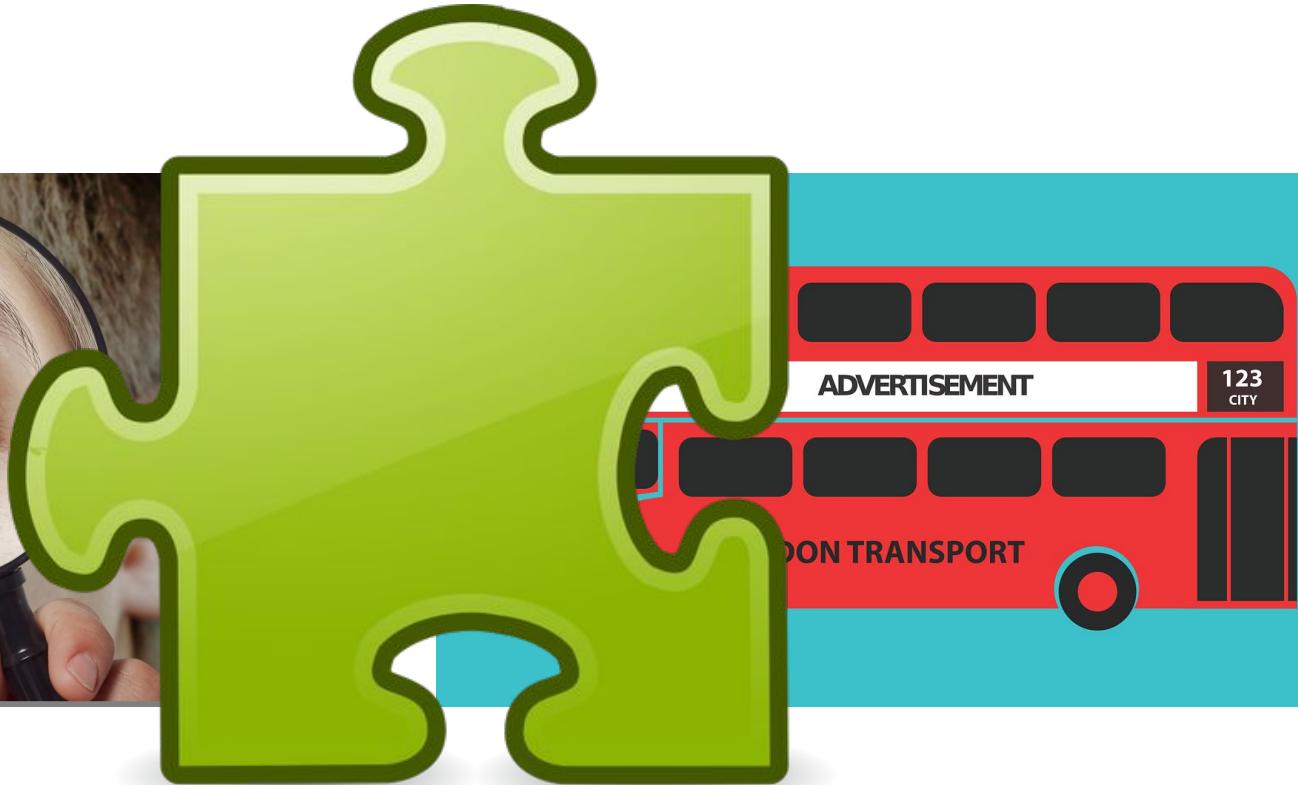


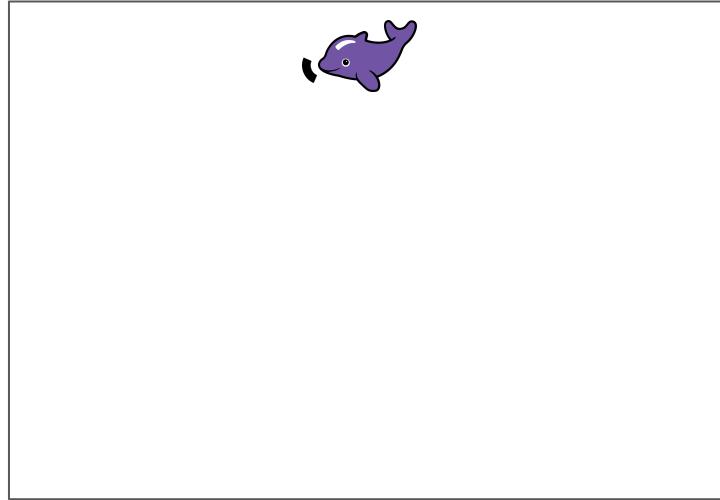
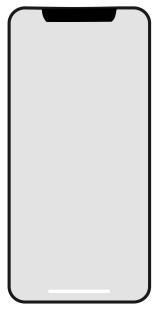
[Start new game](#)

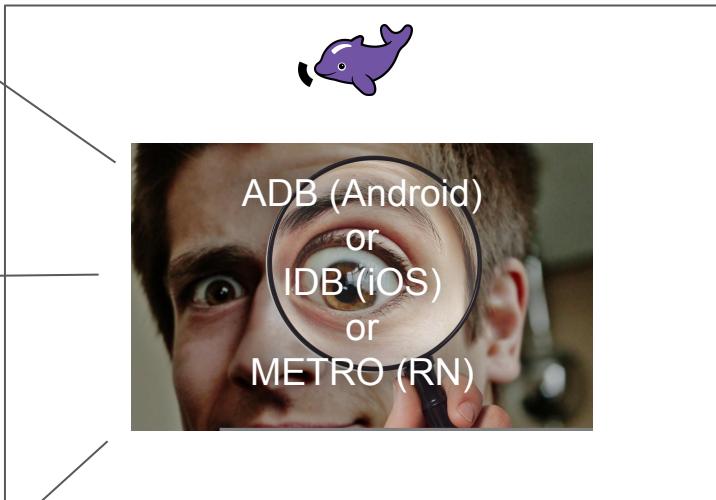
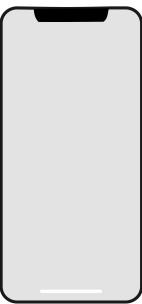
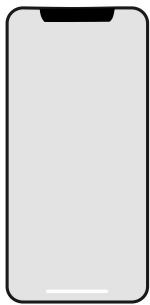


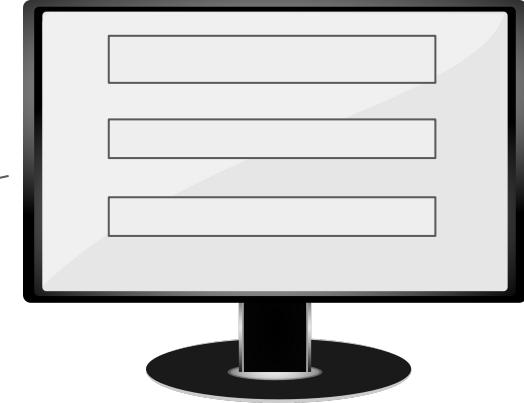
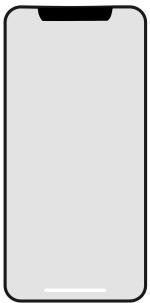
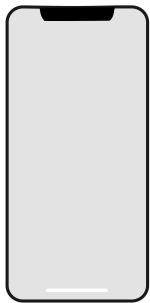


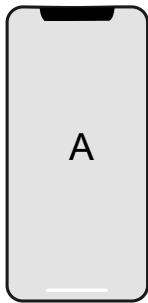




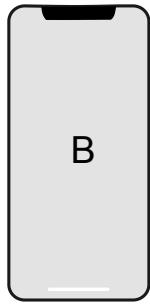




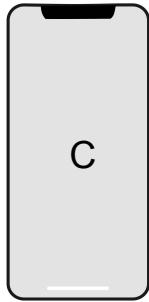




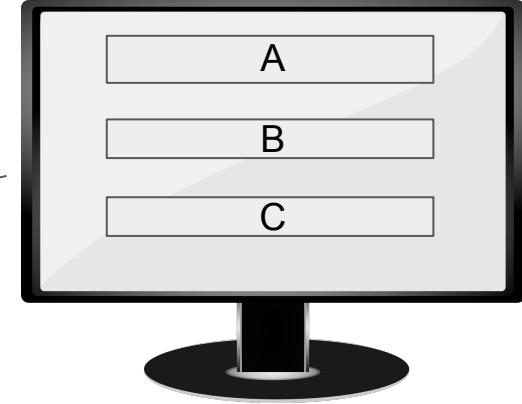
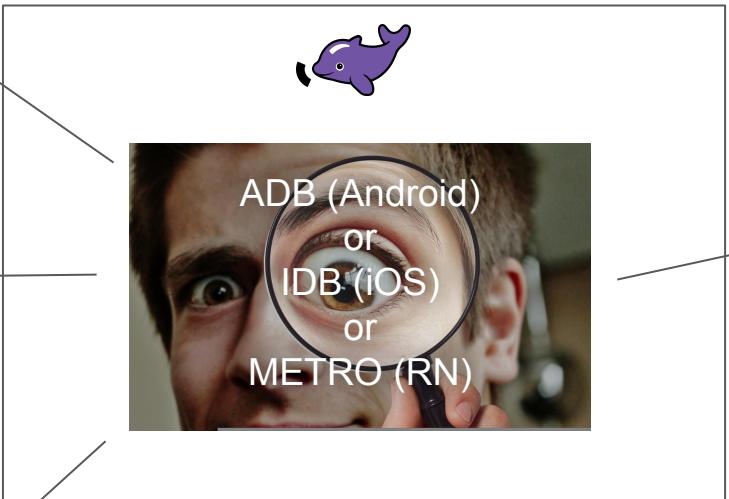
A



B



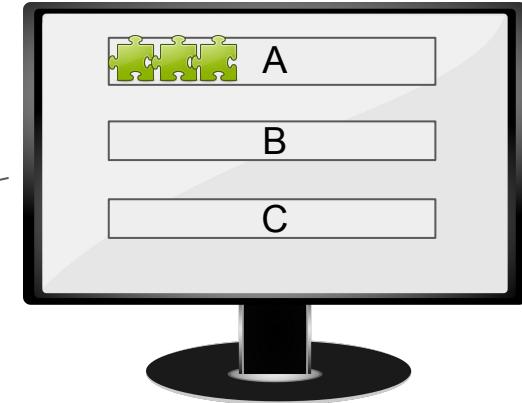
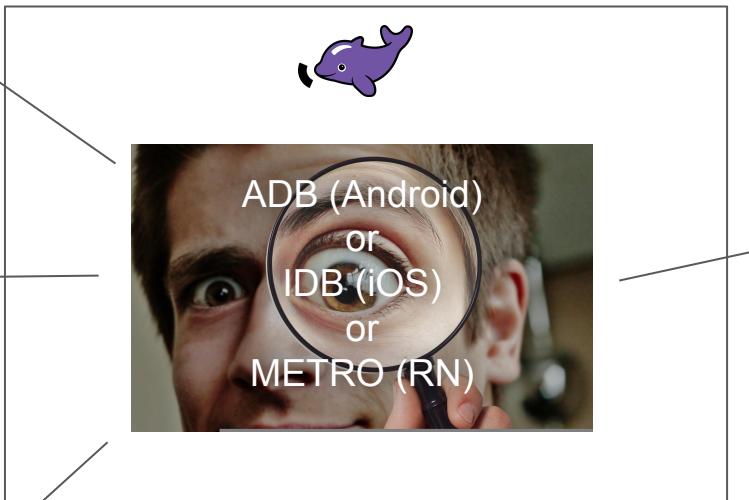
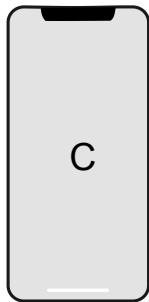
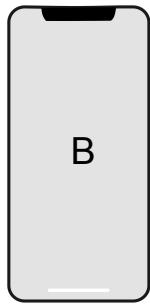
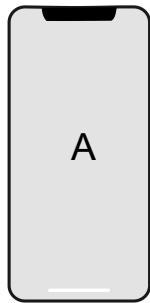
C

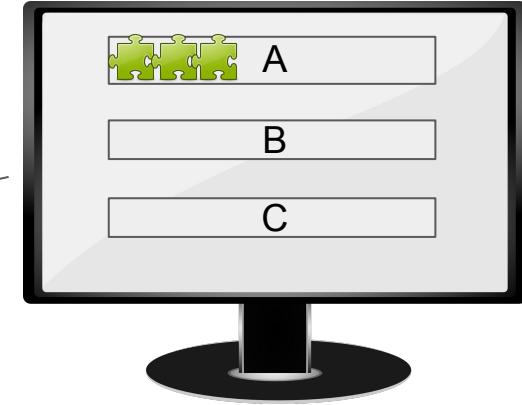
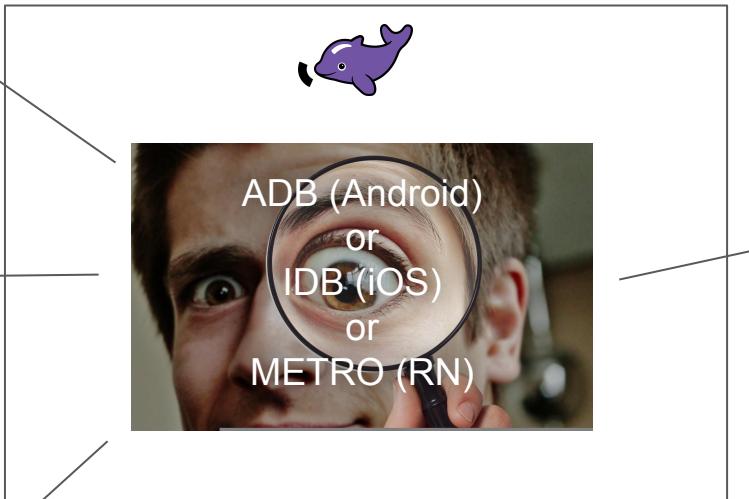
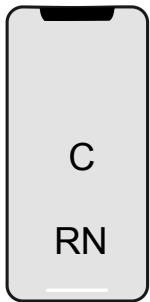
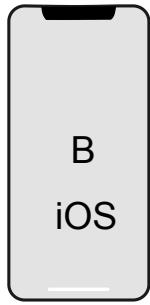
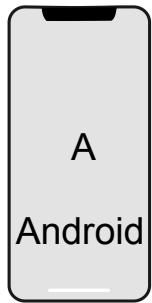


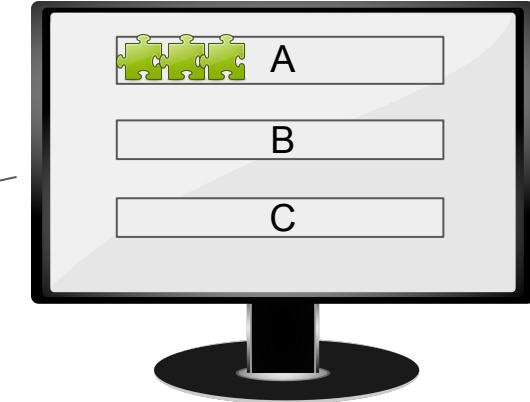
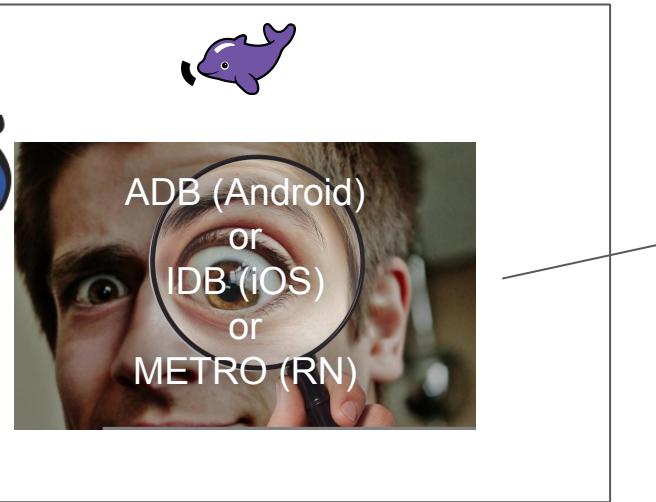
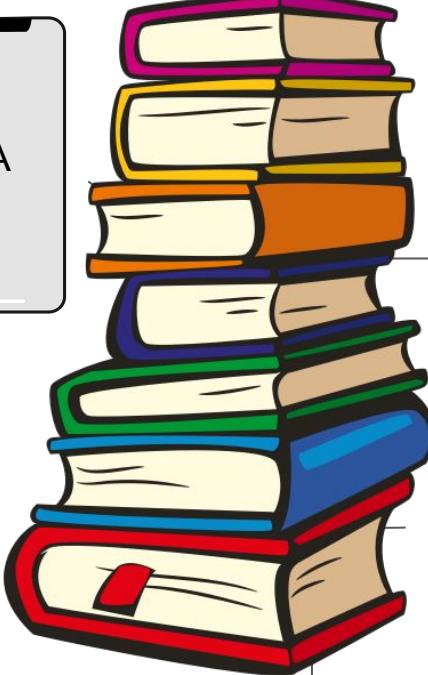
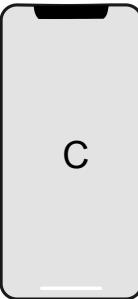
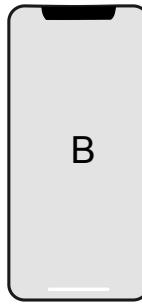
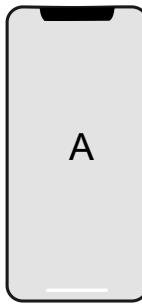
A

B

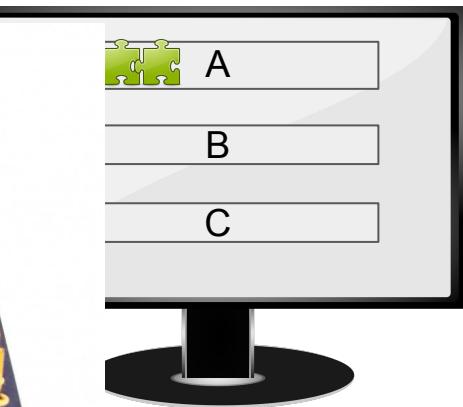
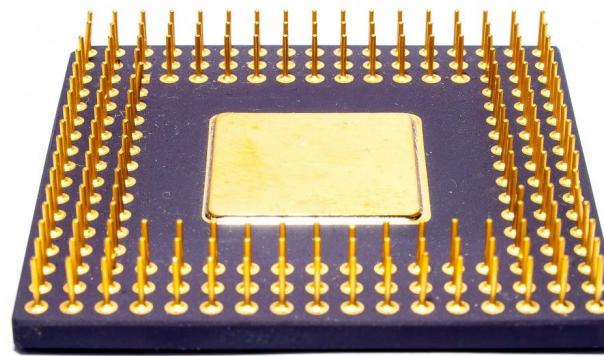
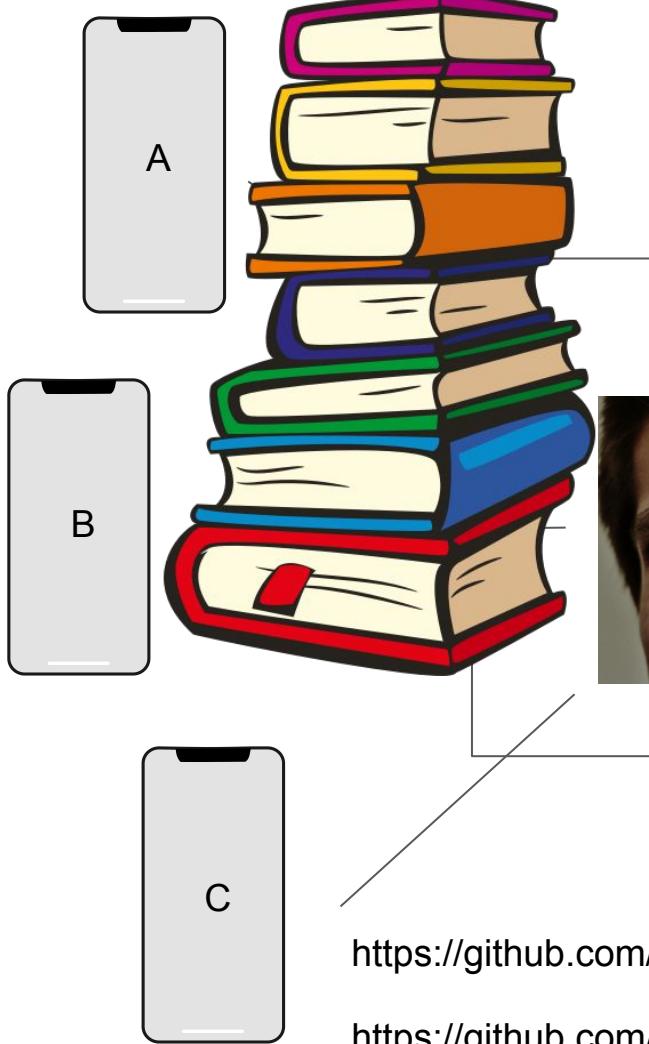
C







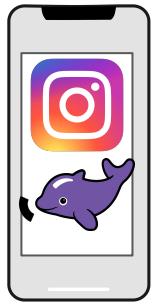
<https://github.com/facebook/flipper/tree/main/desktop/plugins/public/logs>

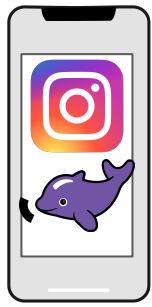


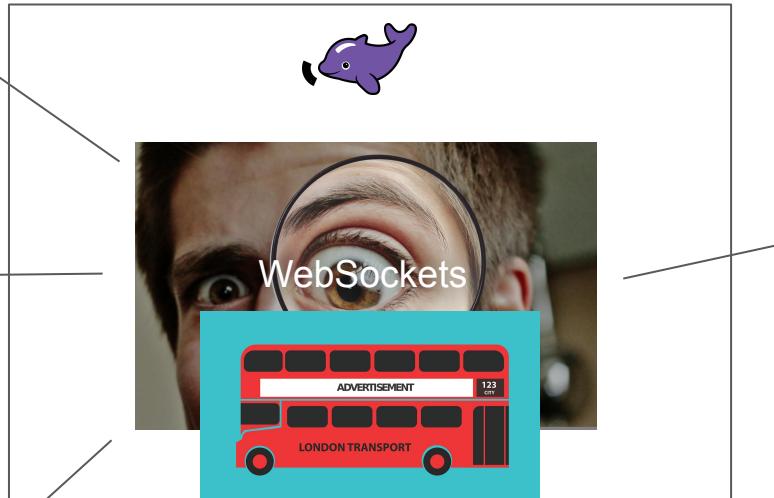
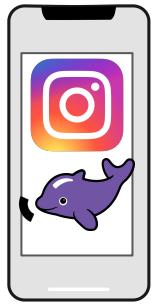
<https://github.com/facebook/flipper/tree/main/desktop/plugins/public/logs>

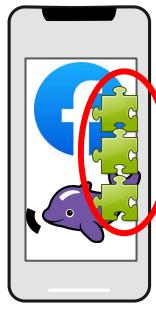
<https://github.com/facebook/flipper/tree/main/desktop/plugins/public/cpu>

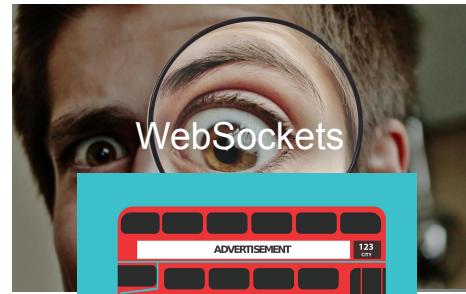
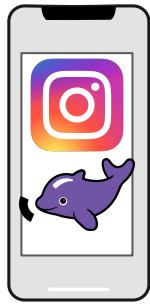
```
public void sendNavigationEvent(  
    @Nullable String keyURI, @Nullable String className, @Nullable Date date) {  
    if (mConnection != null) {  
        FlipperObject sendObject =  
            new FlipperObject.Builder()  
                .put("uri", keyURI)  
                .put("date", date != null ? date : new Date())  
                .put("class", className)  
                .build();  
        mConnection.send("nav_event", sendObject);  
    }  
}
```

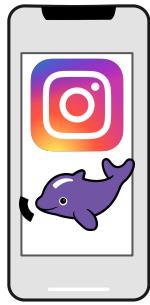










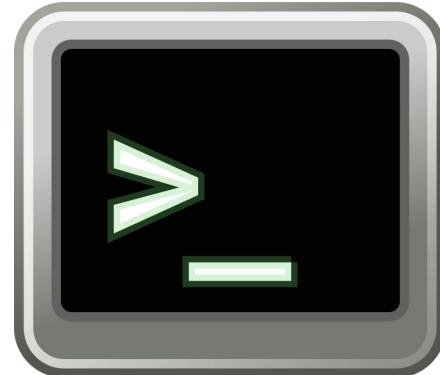
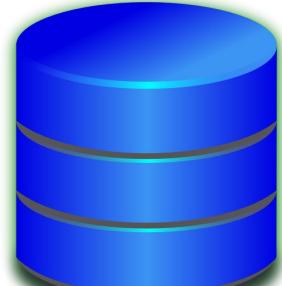
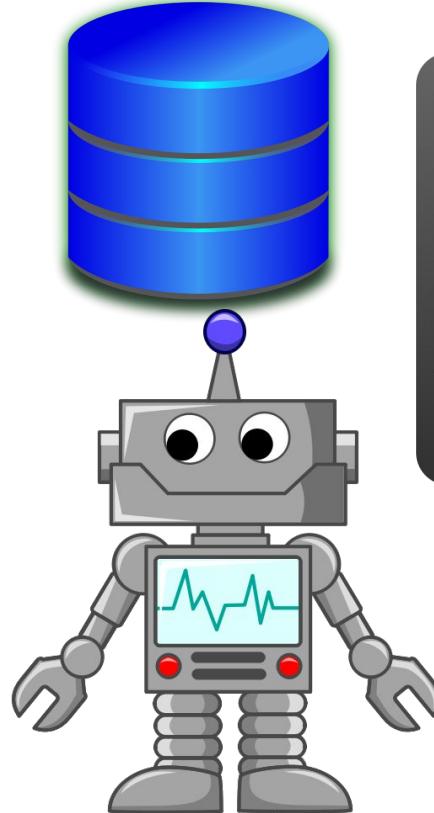


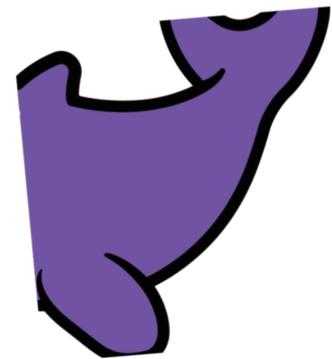
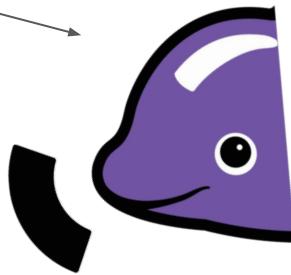
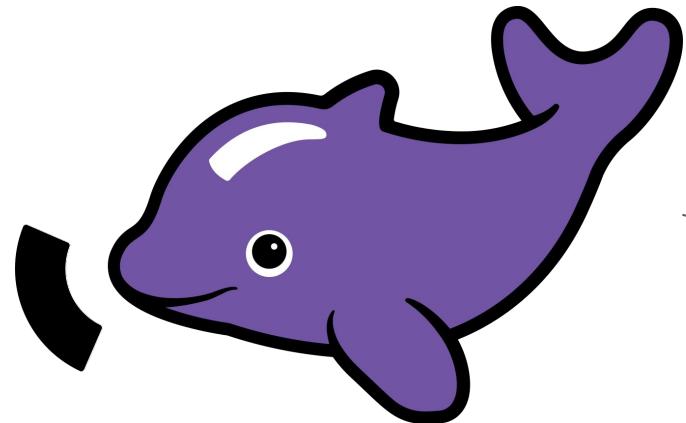


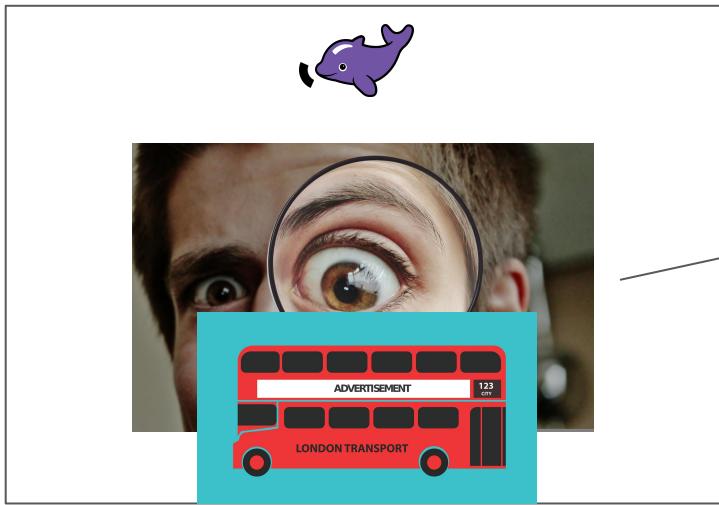








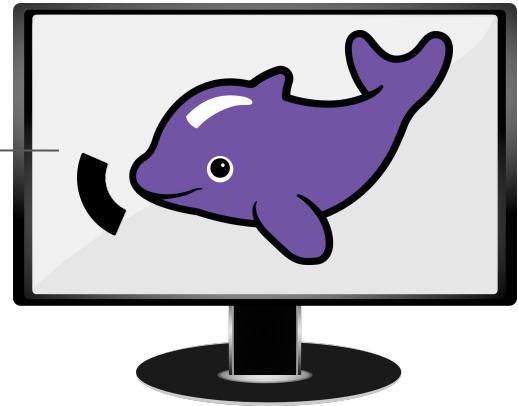
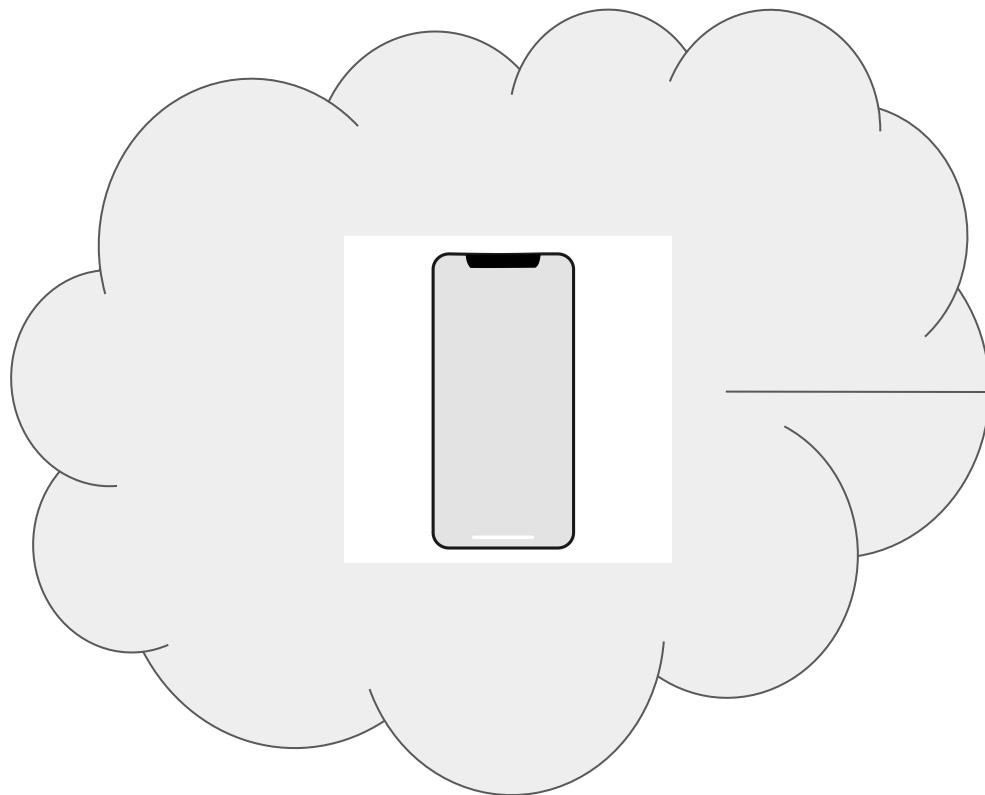


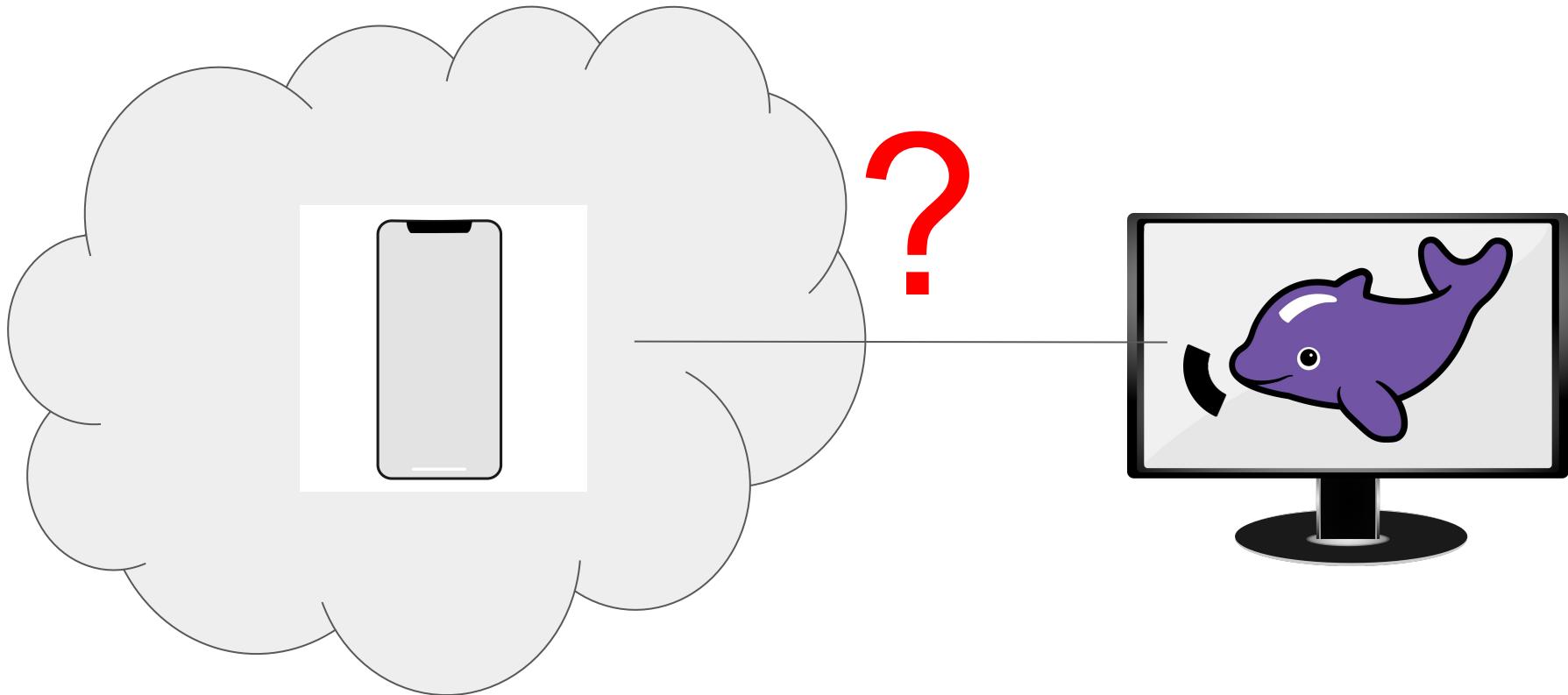


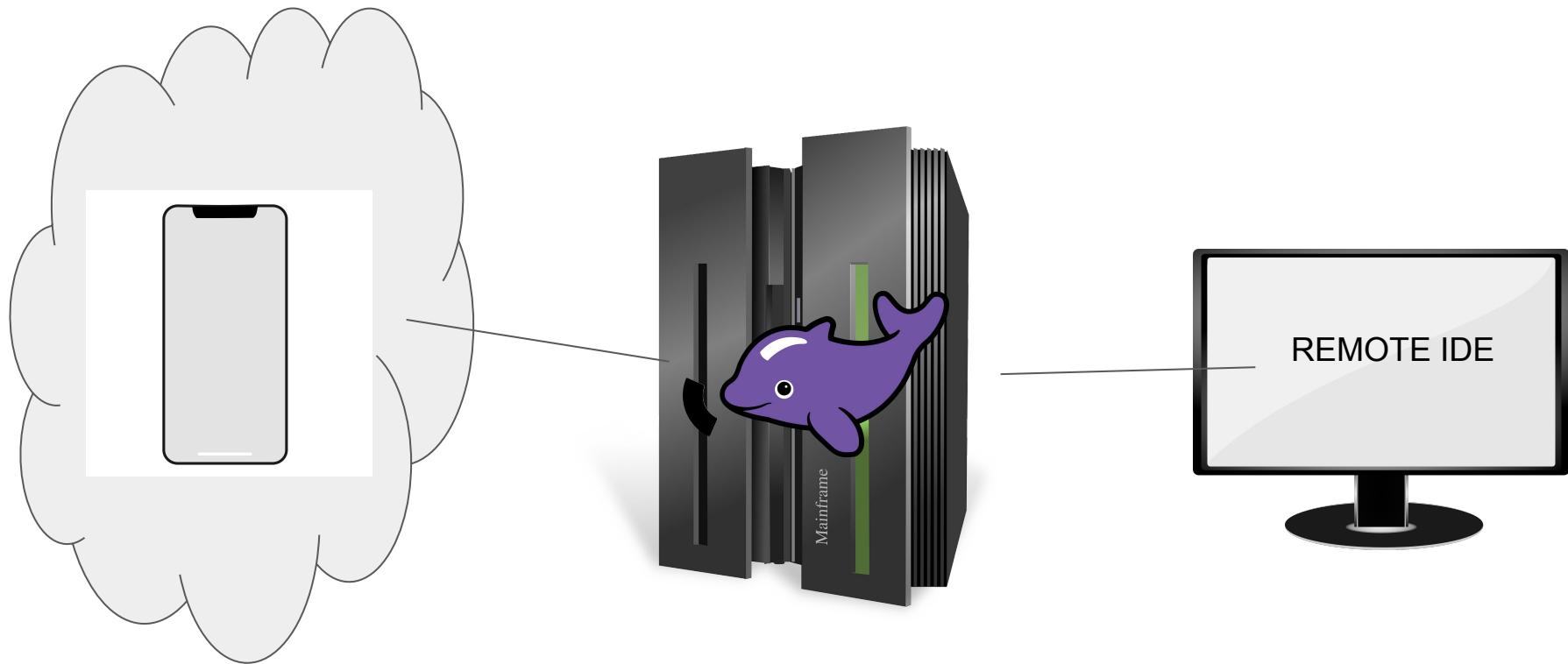


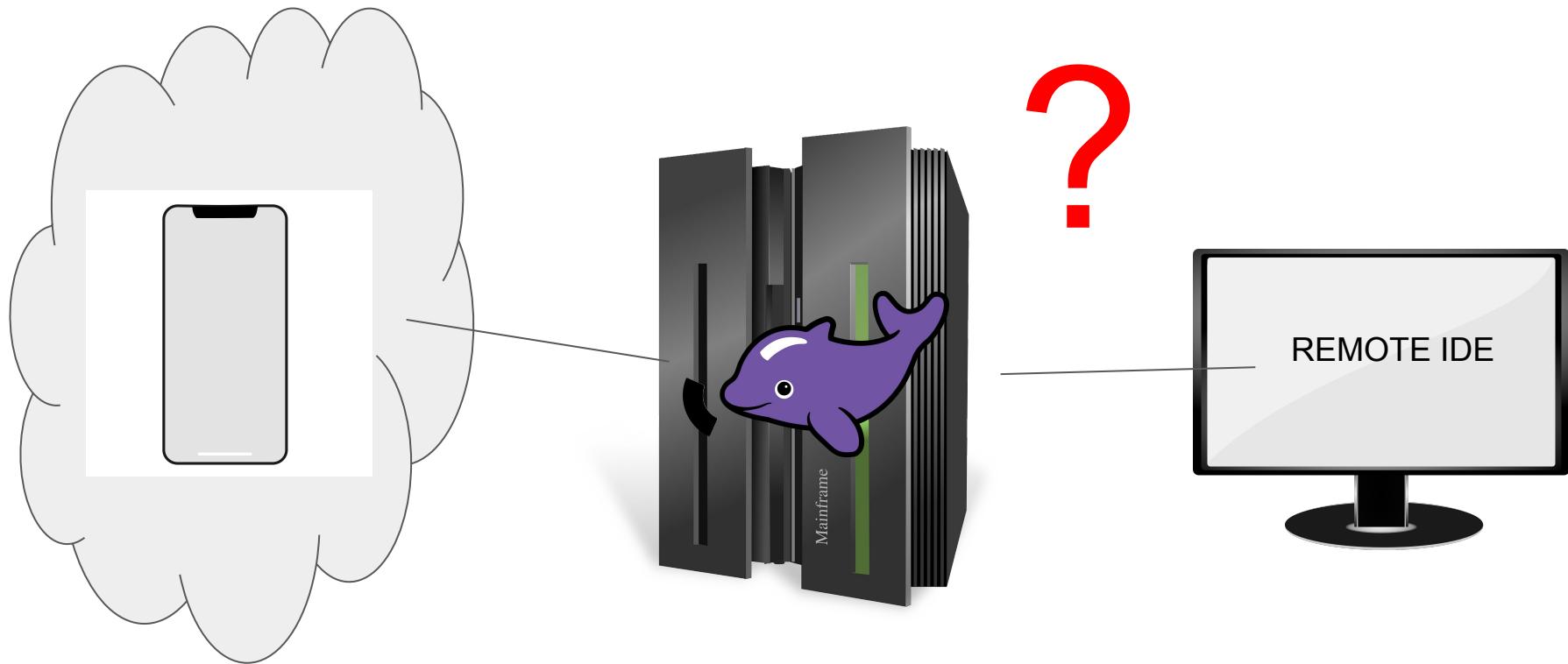
WebView

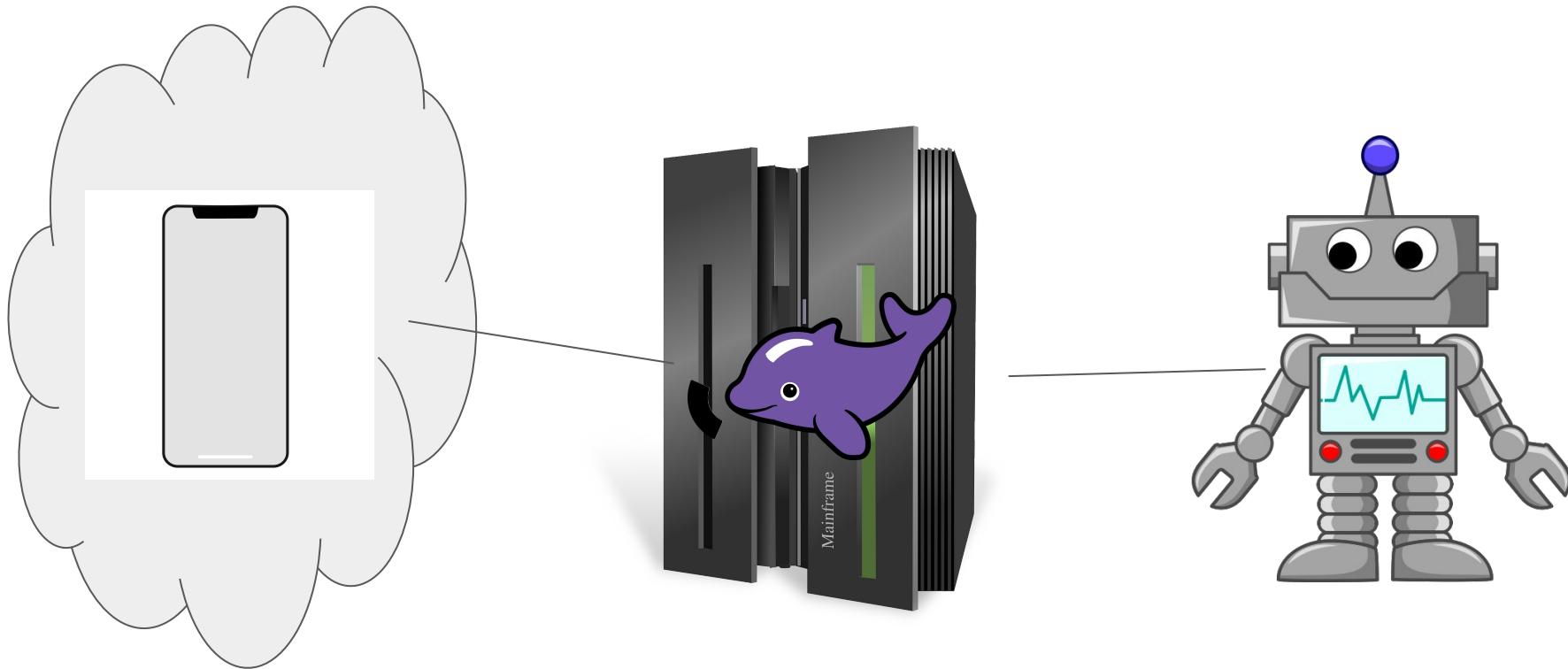


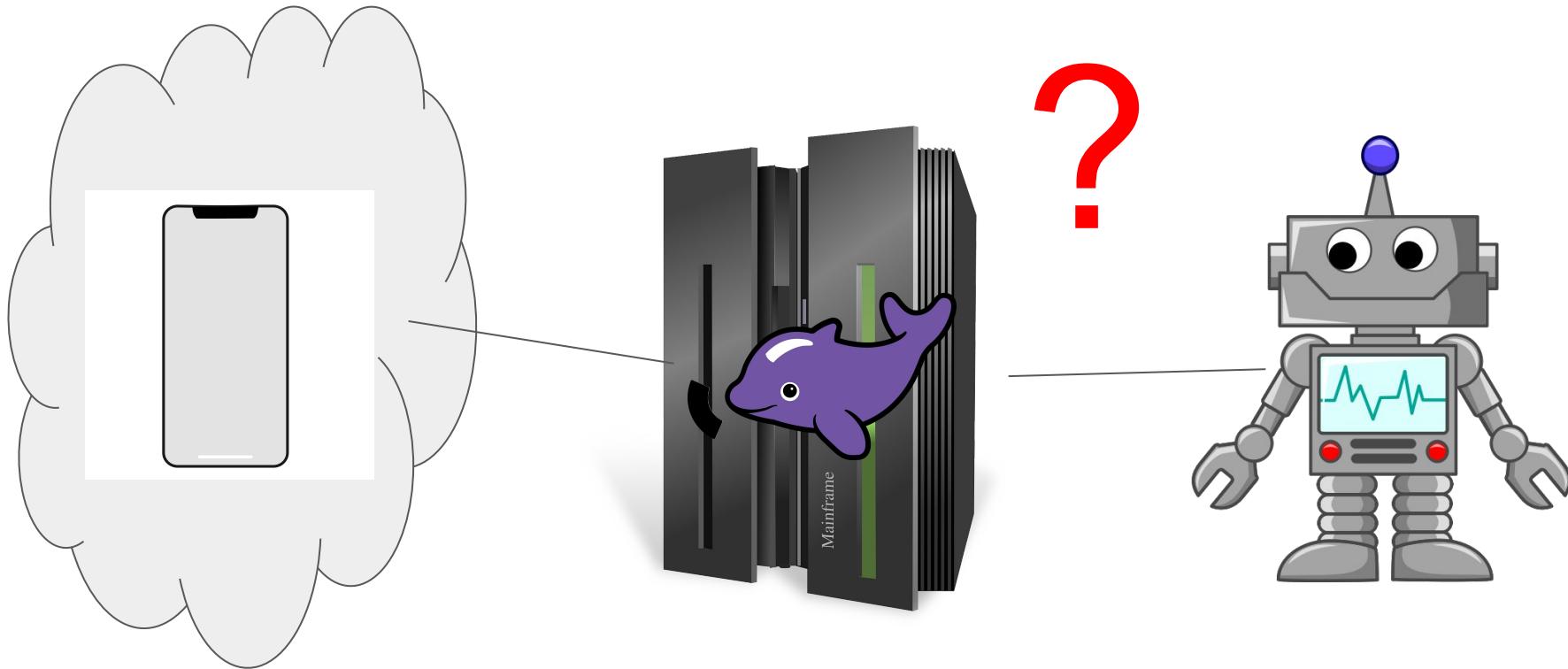


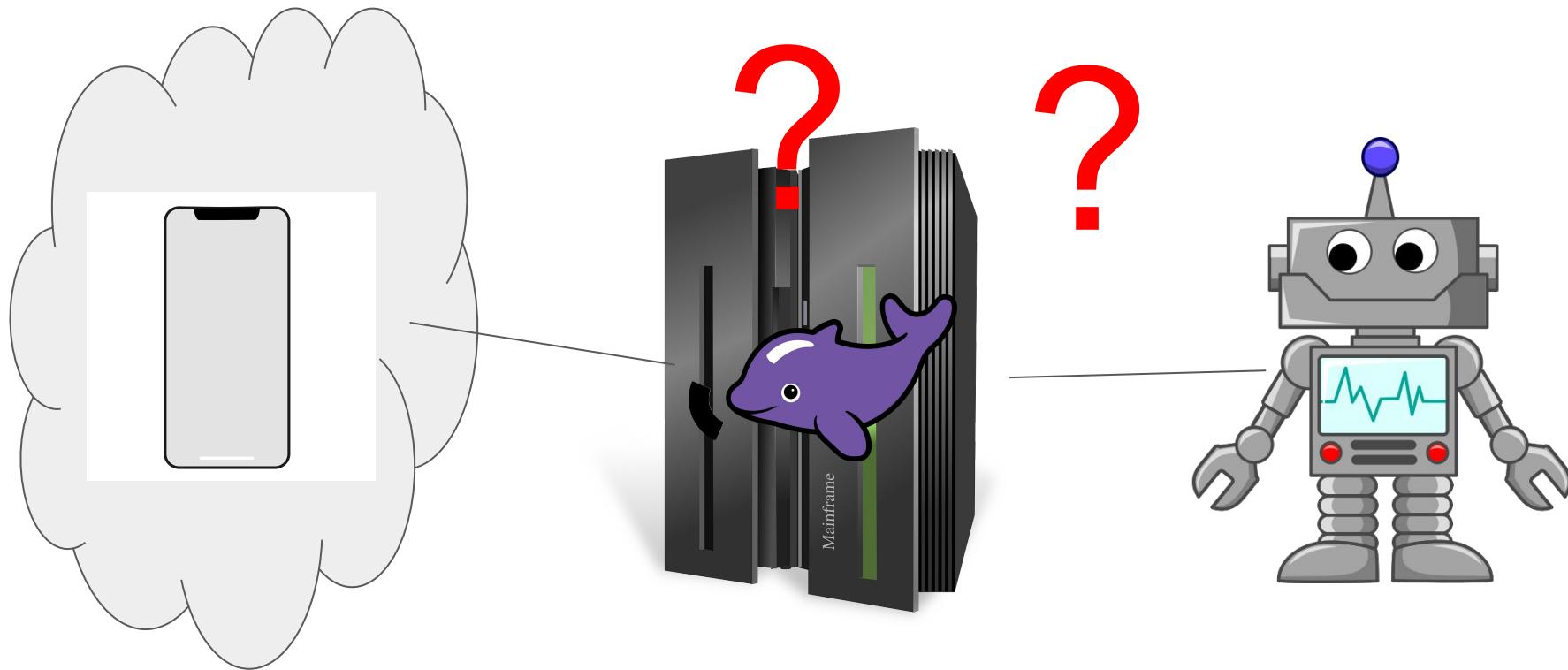


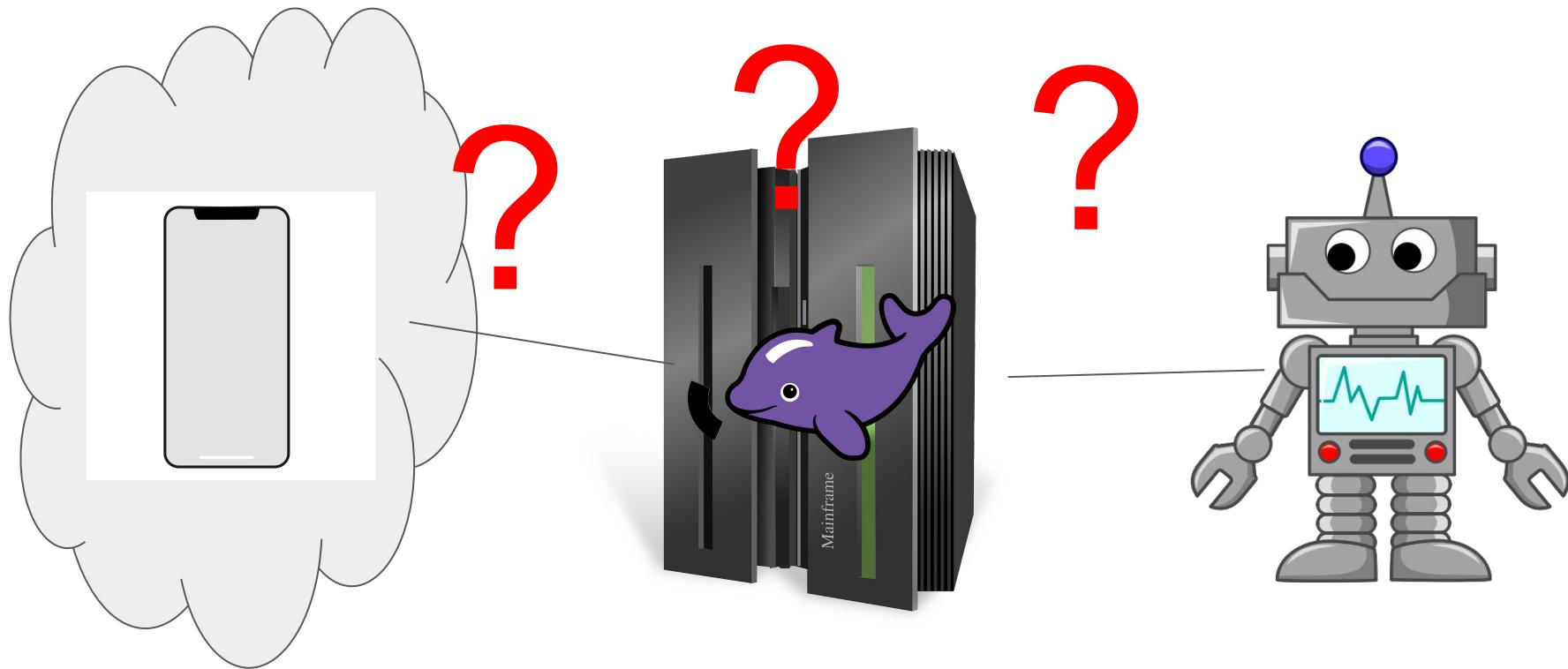


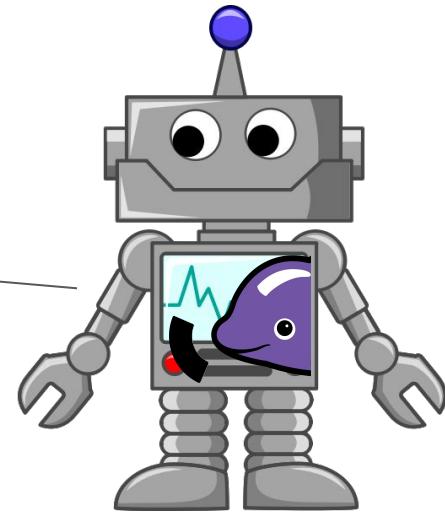
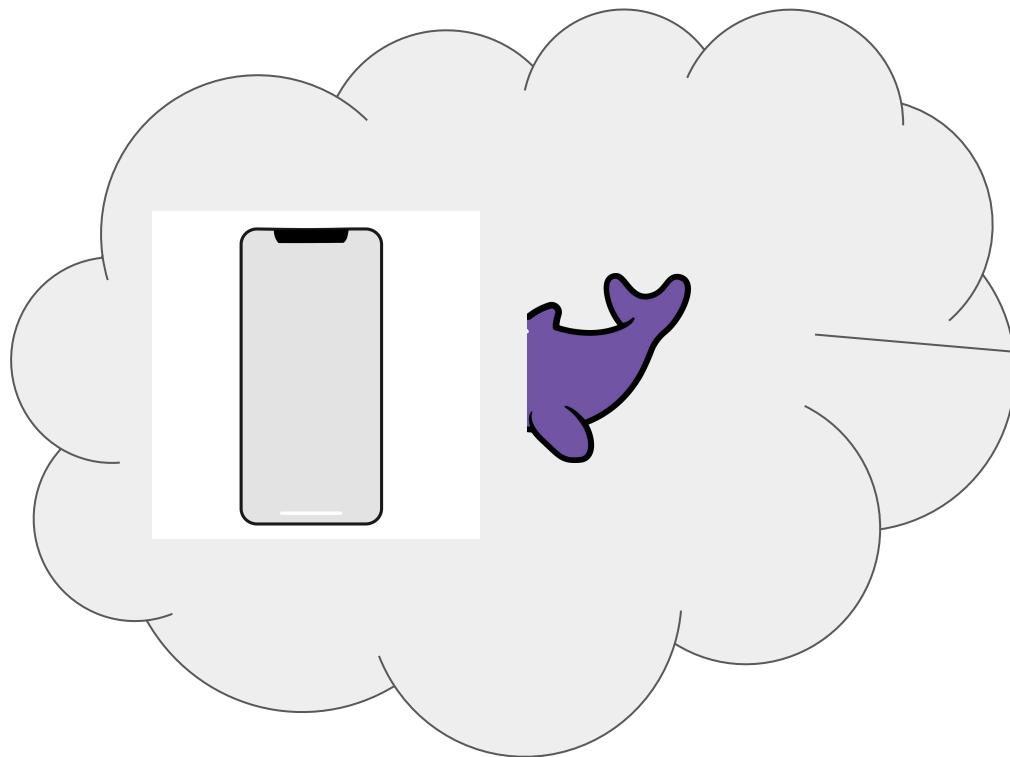






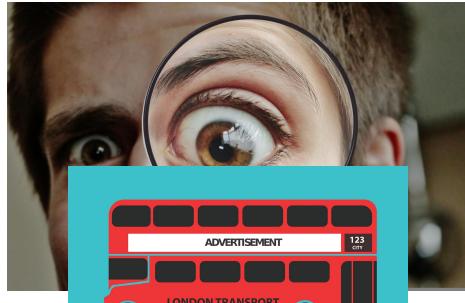






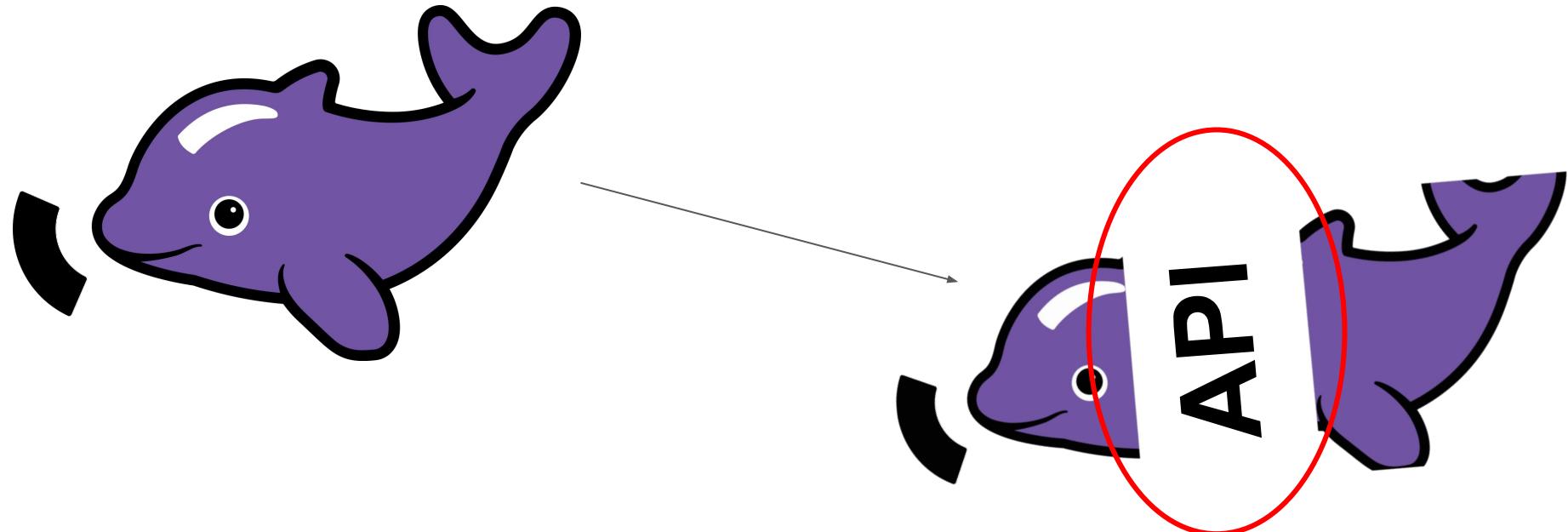


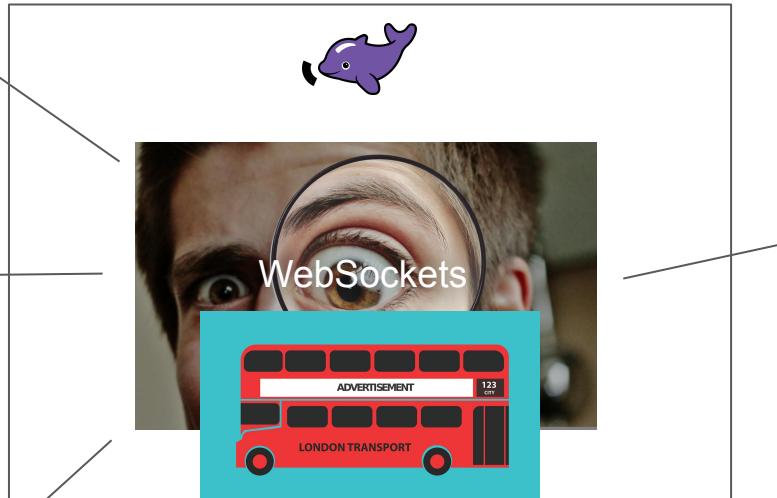
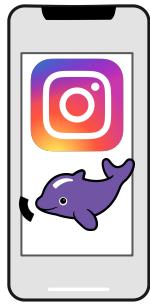


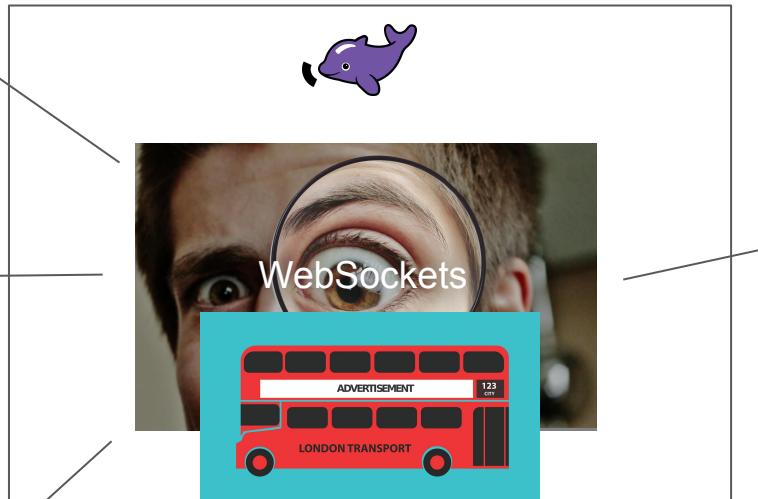
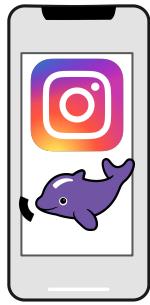


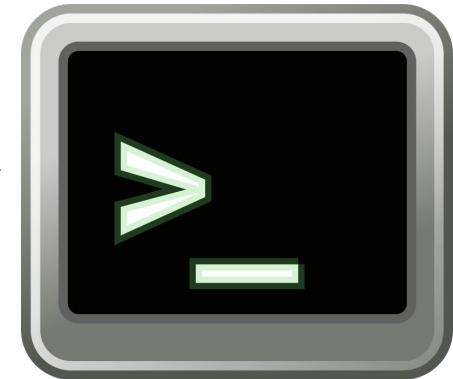
WebSockets

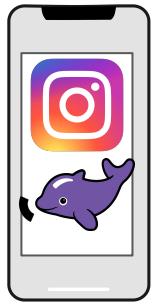


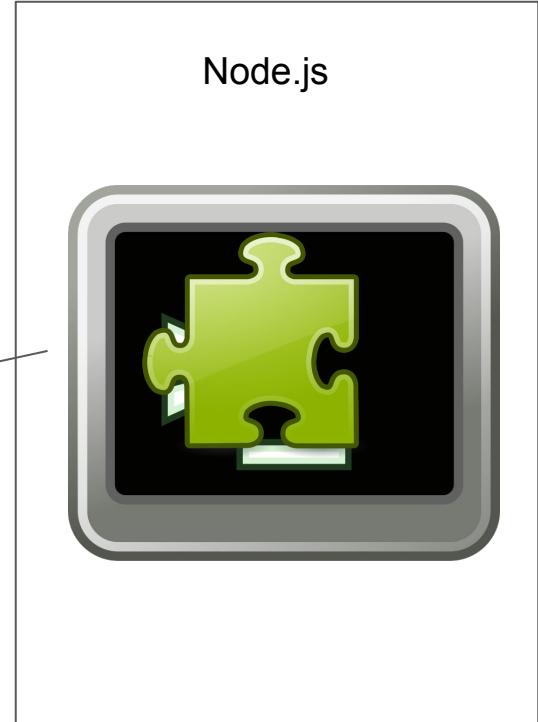
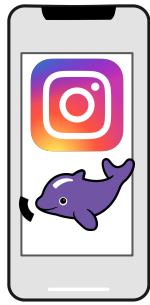


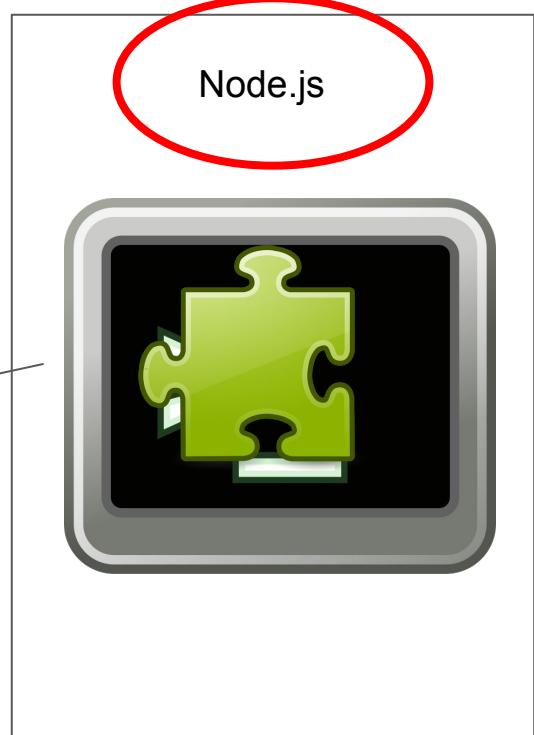
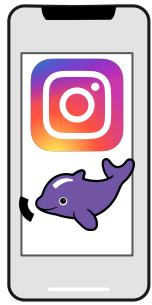


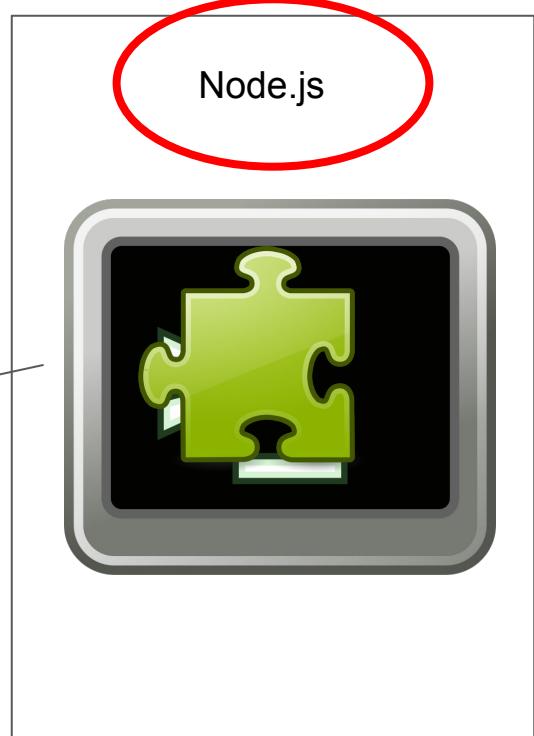
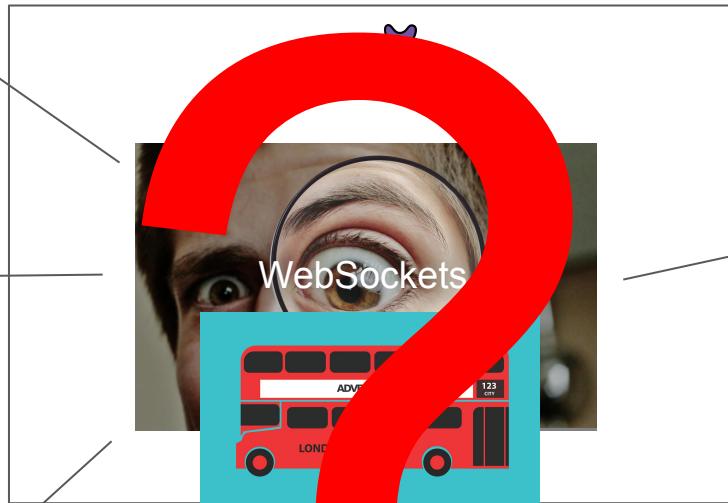
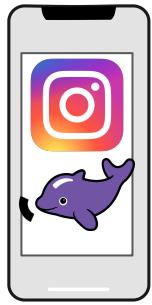


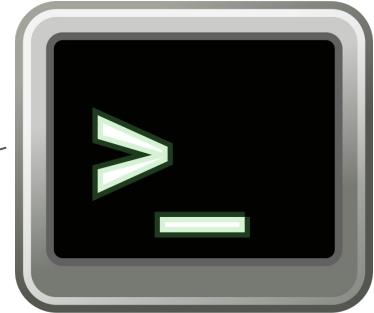
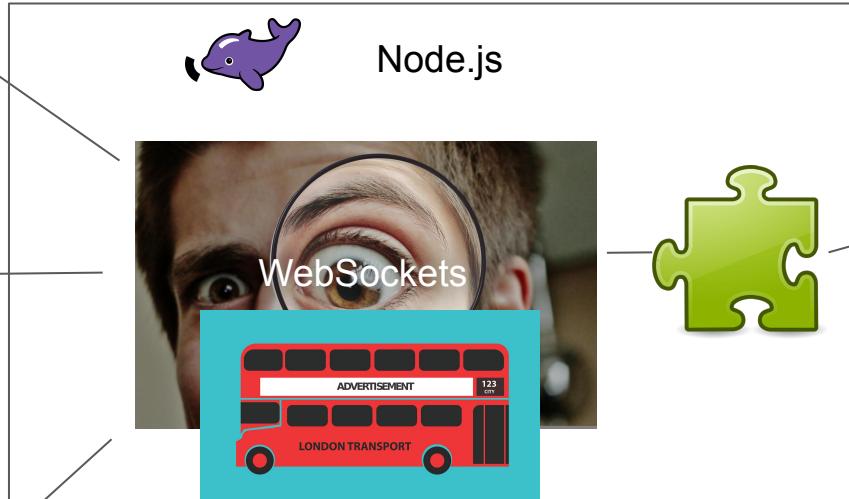
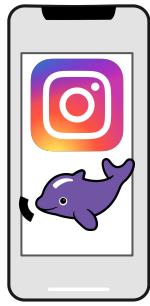












APP INSPECT

ReactNativeFlipp... Pixel_4a_API_32

Troubleshooting Guide

1

PLUGINS

Device 3

- Crash Reporter
- Logs
- Mobile Builds

Enabled 3

- Layout
- Network

React Native Tic Tac Toe

Detected in App 6

- Databases
- Headless-demo
- Images
- Sea Mammals
- Shared Preferences Viewer

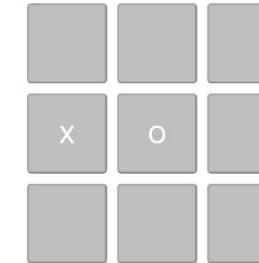
Trace

Unavailable plugins 99+

This plugin demonstrates how to create pure JavaScript Flipper plugins for React Native. Find out how to create a similar plugin at fbflipper.com.

Flipper Tic-Tac-Toe

Mobile players turn..



Start new game

<https://github.com/facebook/flipper/tree/main/desktop/plugins/public/rn-tic-tac-toe>

```
export const plugin = (client: PluginClient<Events, Methods>) => {
  const state = createState(initialState());
  const sendUpdate = () => {};
  const makeMove = (player: 'X' | 'O', move: number) => {};
  const reset = () => {};
  client.onConnect(() => {});
  return {
    makeMove,
    reset,
    state,
  };
};
```

```
const makeMove = (player: 'X' | 'O', move: number) => {
  if (state.get().turn === player && state.get().cells[move] === ' ') {
    state.update(computeNextState(move, player));
    sendUpdate();
  }
};
```

```
const sendUpdate = () => {
  client.send('SetState', state.get());
};
```

```
client.onConnect(() => {

  client.onMessage('XMove', ({move}) => {
    makeMove('X', move);
  });

  client.onMessage('GetState', () => {
    sendUpdate();
  });

  sendUpdate();
});
```

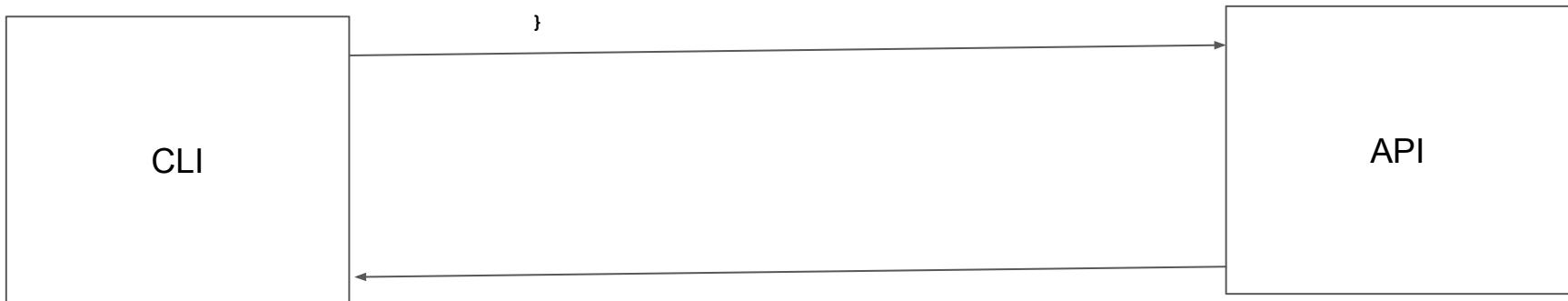
```
export const Component = () => {

  const pluginInstance = usePlugin(plugin);
  const {winner, turn, cells} = useState(pluginInstance.state);

  return (
    <Layout.Container>
      <GameBoard>
        {cells.map((c, idx) => (
          <Cell
            key={idx}
            disabled={c === ' ' || turn !== desktopPlayer || winner !== ' '}
            onClick={() => pluginInstance.makeMove(desktopPlayer, idx)}>
            {c}
          </Cell>
        )));
      </GameBoard>
    </Layout.Container>
  );
};
```

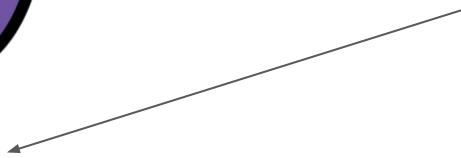
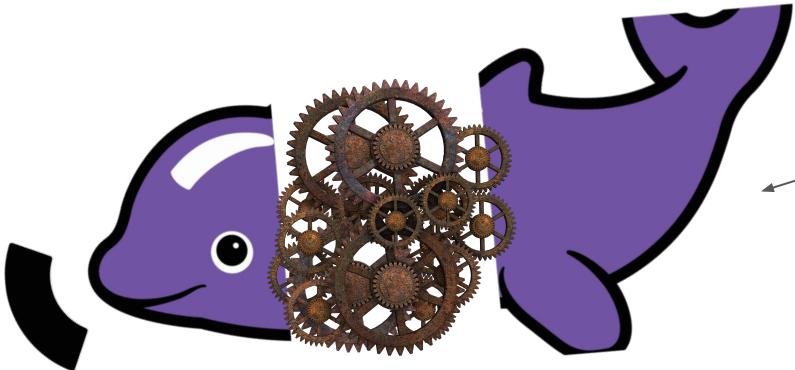
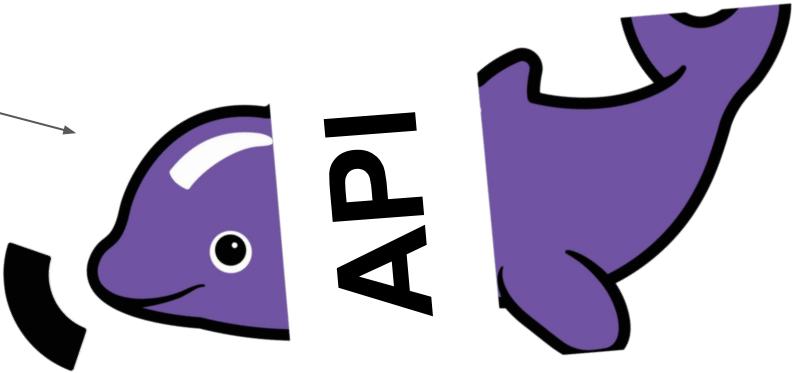
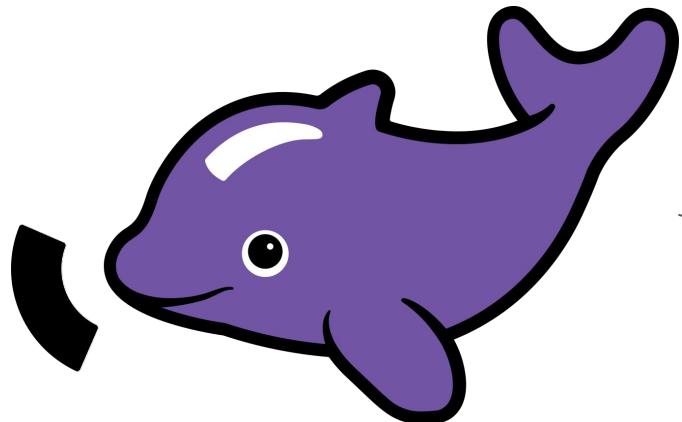
```
export const API = (pluginInstance:  
  FlipperPluginInstance<typeof plugin>) => {  
  
  return {  
  
    makeMove: pluginInstance.makeMove,  
  
    reset: pluginInstance.reset,  
  
    state: pluginInstance.state,  
  
  };  
};
```

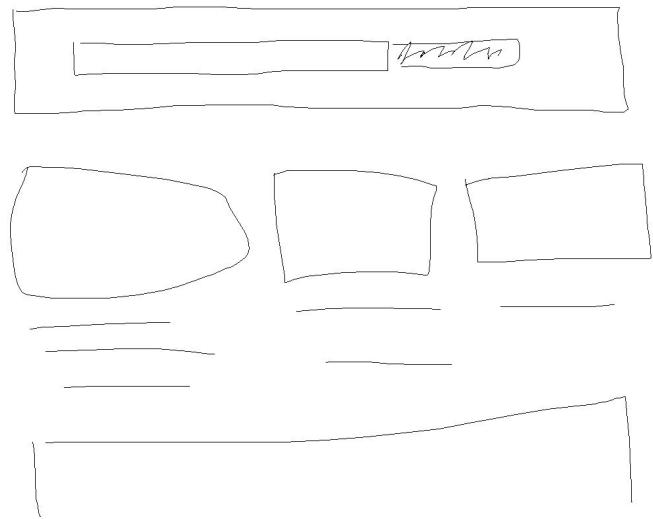
```
{  
  event: 'exec',  
  payload: {  
    event: 'companion-plugin-exec',  
    id: 42,  
    data: {...},  
  },  
},
```

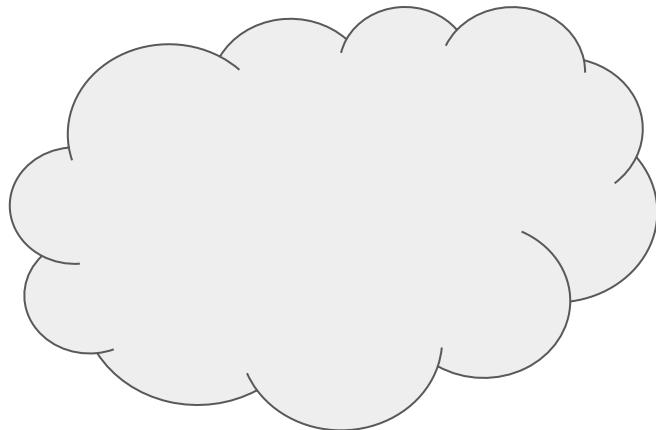
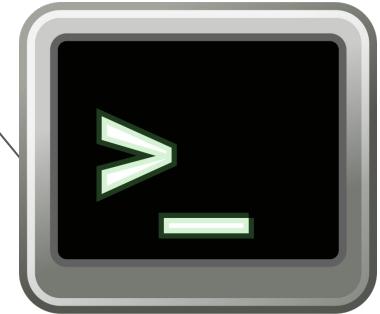
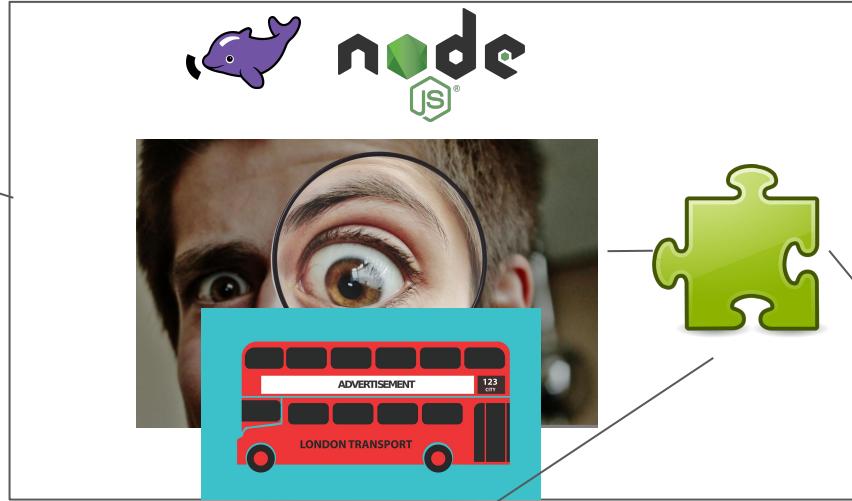


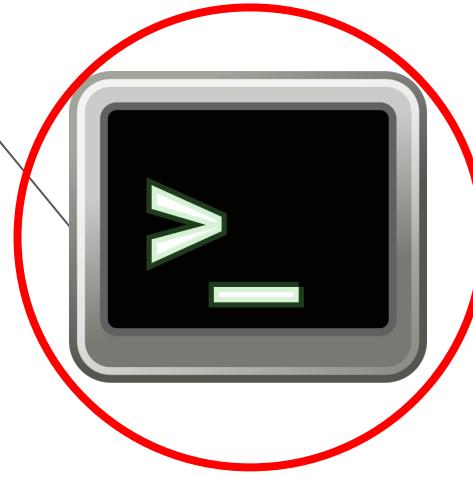
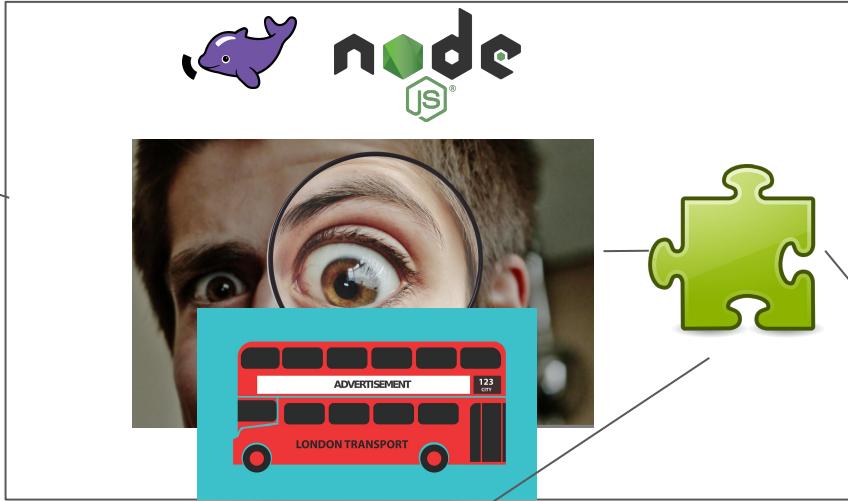
```
{  
  event: 'exec-response',  
  payload: {  
    id: 42,  
    data: {...},  
  },  
},  
}
```

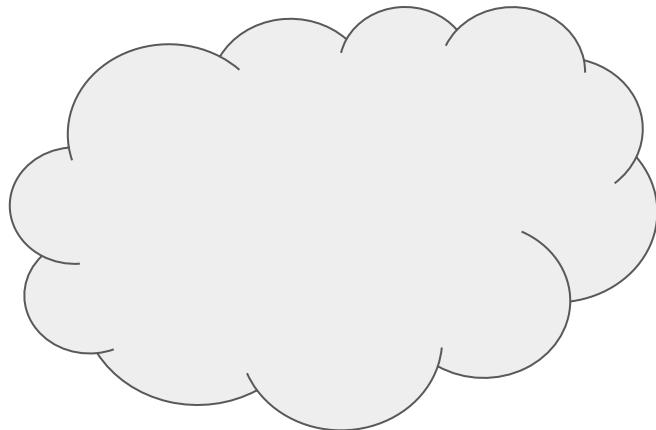
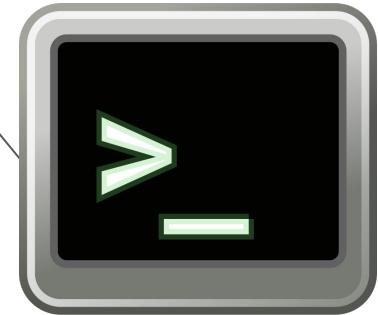
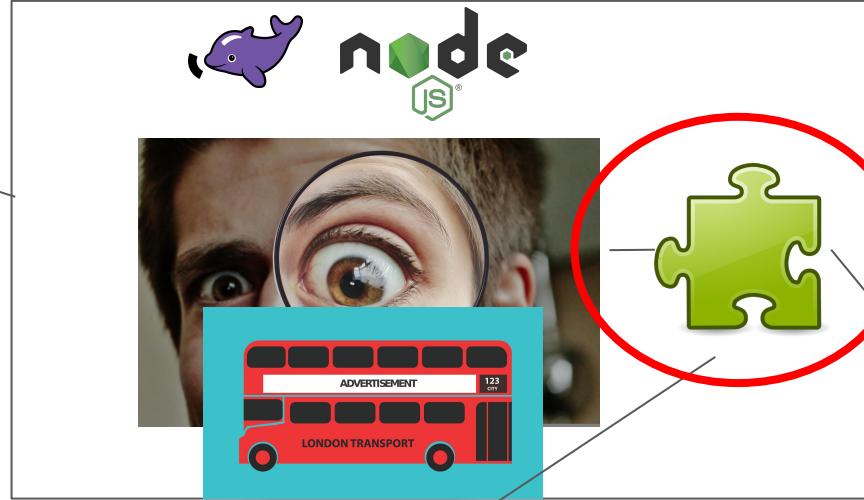


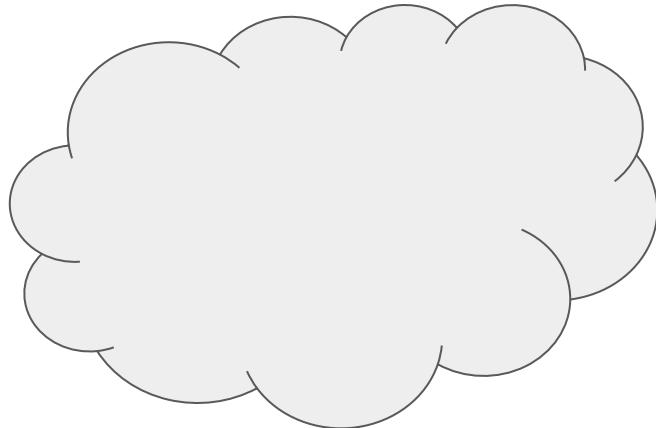
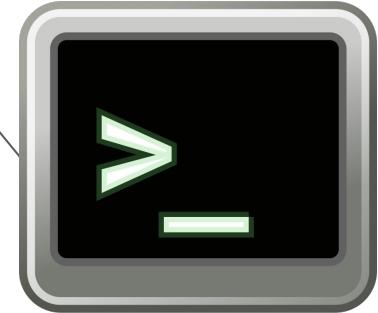
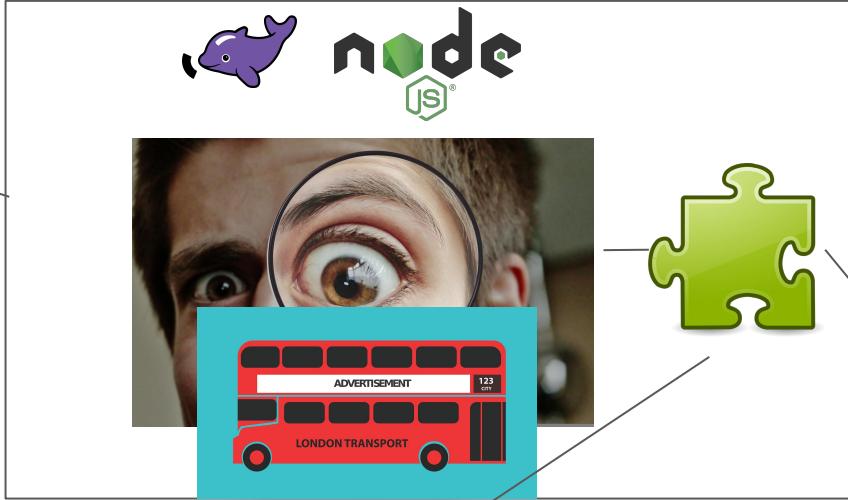


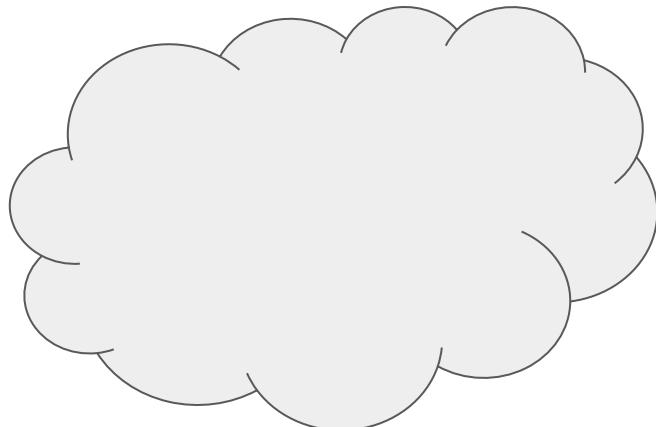
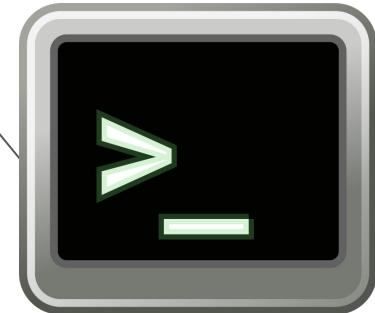
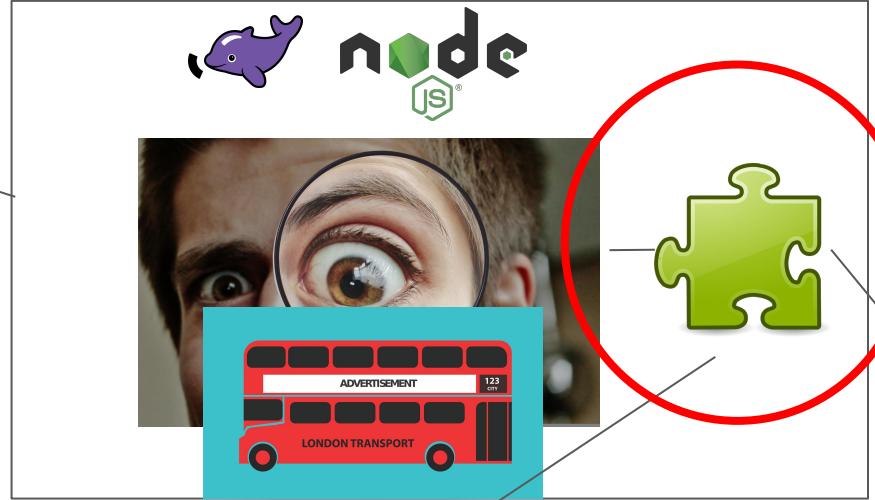


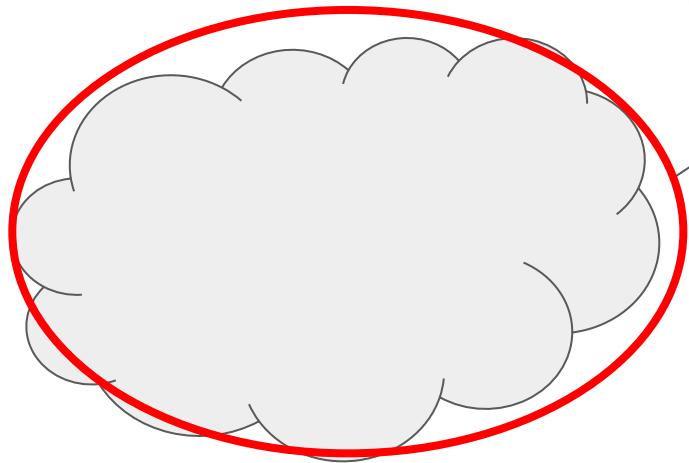
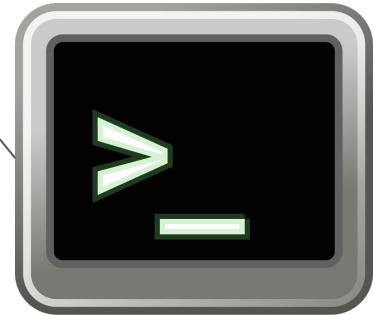
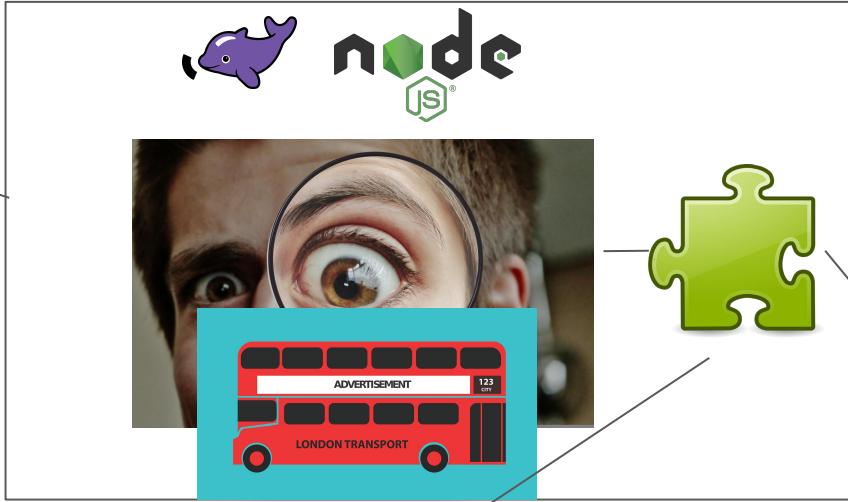






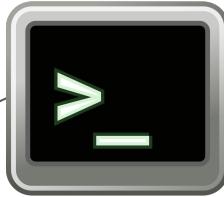
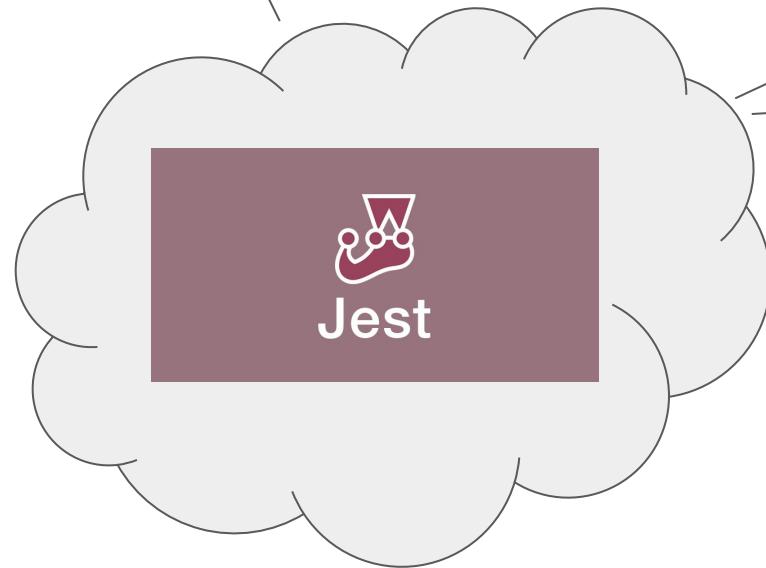


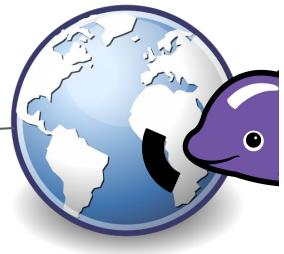
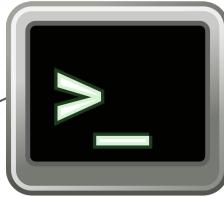
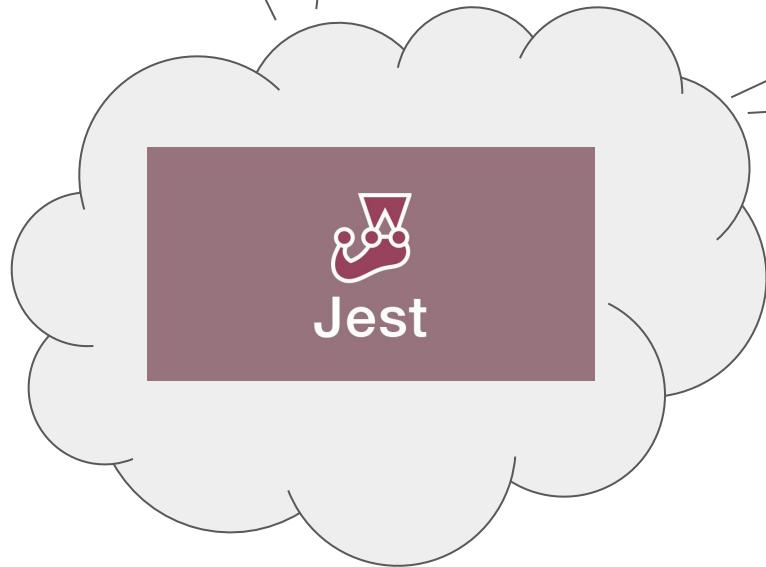
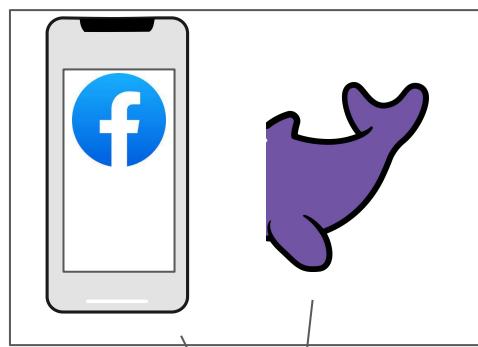


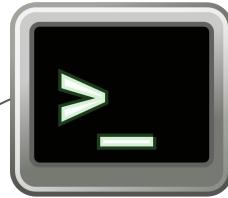
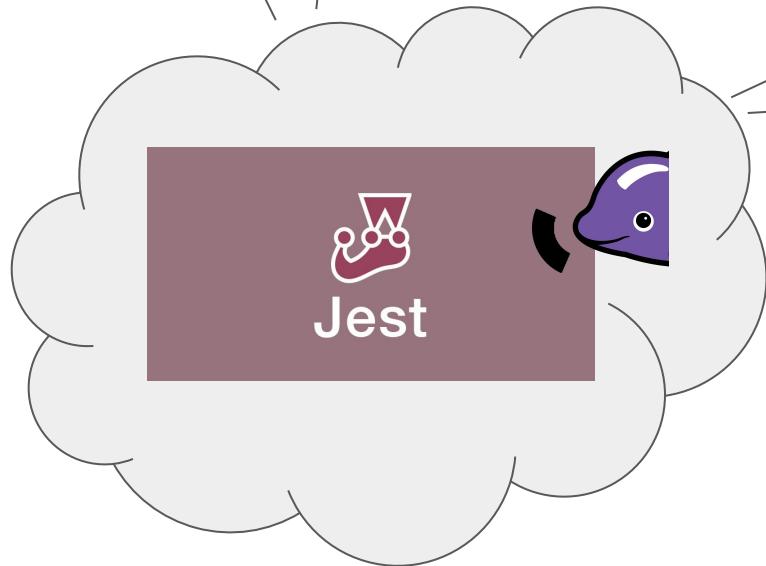
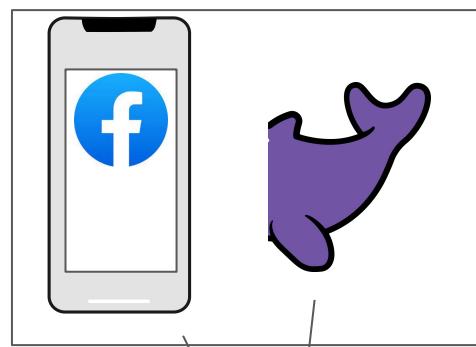


Testing











publicdomainvectors.org

<https://github.com/facebook/flipper/tree/main/desktop/examples/headless-tic-tac-toe>

<https://github.com/facebook/flipper/tree/main/desktop/examples/headless-demo>



publicdomainvectors.org

<https://github.com/facebook/flipper/tree/main/desktop/examples/headless-tic-tac-toe>
<https://github.com/facebook/flipper/tree/main/desktop/examples/headless-demo>



flipper_fb



facebook/flipper



publicdomainvectors.org

<https://github.com/facebook/flipper/tree/main/desktop/examples/headless-tic-tac-toe>
<https://github.com/facebook/flipper/tree/main/desktop/examples/headless-demo>



flipper_fb



facebook/flipper



ai_goncharov



aigoncharov

- <https://pixabay.com/photos/magnifying-glass-detective-looking-5965372/>
- <https://pixabay.com/vectors/london-bus-london-bus-transport-1510821/>
- <https://pixabay.com/illustrations/iphone-x-iphone-x-mockup-iphone-3365689/>
- <https://pixabay.com/vectors/monitor-screen-display-tv-155565/>
- <https://publicdomainvectors.org/en/free-clipart/Plugin-icon/59375.html>
- <https://pxhere.com/en/photo/1342296>
- <https://pixabay.com/illustrations/envelope-paper-letters-post-black-5431870/>
- <https://freesvg.org/blue-database-icon-vector-image>
- <https://freesvg.org/gps>
- <https://freesvg.org/command-line>
- <https://pixabay.com/vectors/computer-mainframe-server-icon-98401/>
- <https://freesvg.org/comic-robot>
- <https://publicdomainvectors.org/en/free-clipart/Vector-image-of-internet-web-browser-icon/31567.html>
- <https://pixabay.com/vectors/heart-love-heart-rate-ekg-health-6232887/>
- https://commons.wikimedia.org/wiki/File:Not_facebook_dislike_thumbs_down.png
- <https://pixabay.com/photos/scissors-duct-tape-blister-foil-1986602/>
- <https://pixabay.com/vectors/mobile-devices-website-mockup-web-2017978/>
- <https://pixabay.com/illustrations/test-testing-sign-laboratory-670091/>
- <https://publicdomainvectors.org/en/free-clipart/Kids-read-a-book/70498.html>
- <https://freesvg.org/stack-of-books-vector-illustration>
- <https://pixabay.com/vectors/gandalf-tux-animal-bird-conjure-161456/>
- <https://pixabay.com/photos/gear-gear-wheels-steampunk-rusty-1127518/>

