Technology Architecture PWA Serverless Notes

VueJS PWA

Steps to create new Vue JS PWA:

Install vue cli and create a new application:

```
npm install -g vue
vue create vue-pwa-demo
cd vue-pwa-demo
```

Add the Vue PWA plugin to the application

```
vue add pwa
```

Build the production optimized application in the dist directory:

```
npm run build
```

Use serve to serve the dist folder over HTTP:

```
npx serve -s dist
```

Open the application in a Chrome browser and you will see the install option!

Click install and the application will be installed in chrome! Check below how to view installed apps on Chrome and see screenshots.

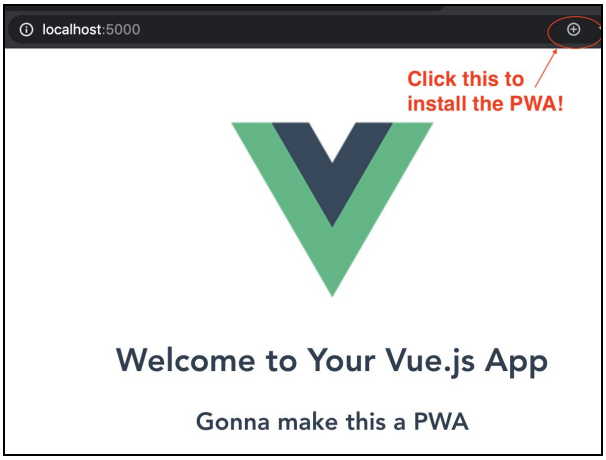
View installed applications

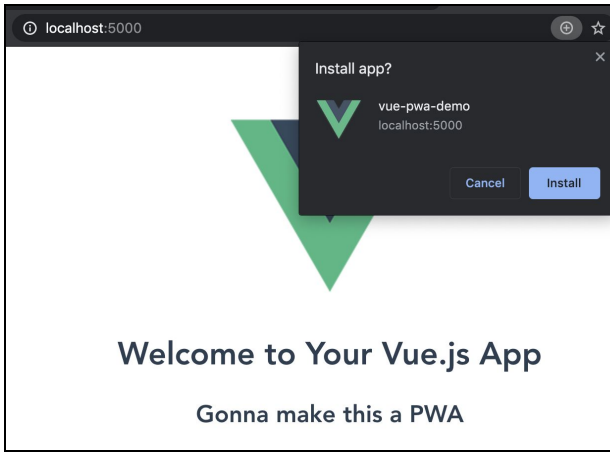
Via Chrome in the address bar type `chrome://apps/`
In the Terminal (Mac): type `ls -l ~/Applications/Chrome\ Apps.localized`
In the Mac Applications explorer: `Press F4 key and navigate to the app icon`

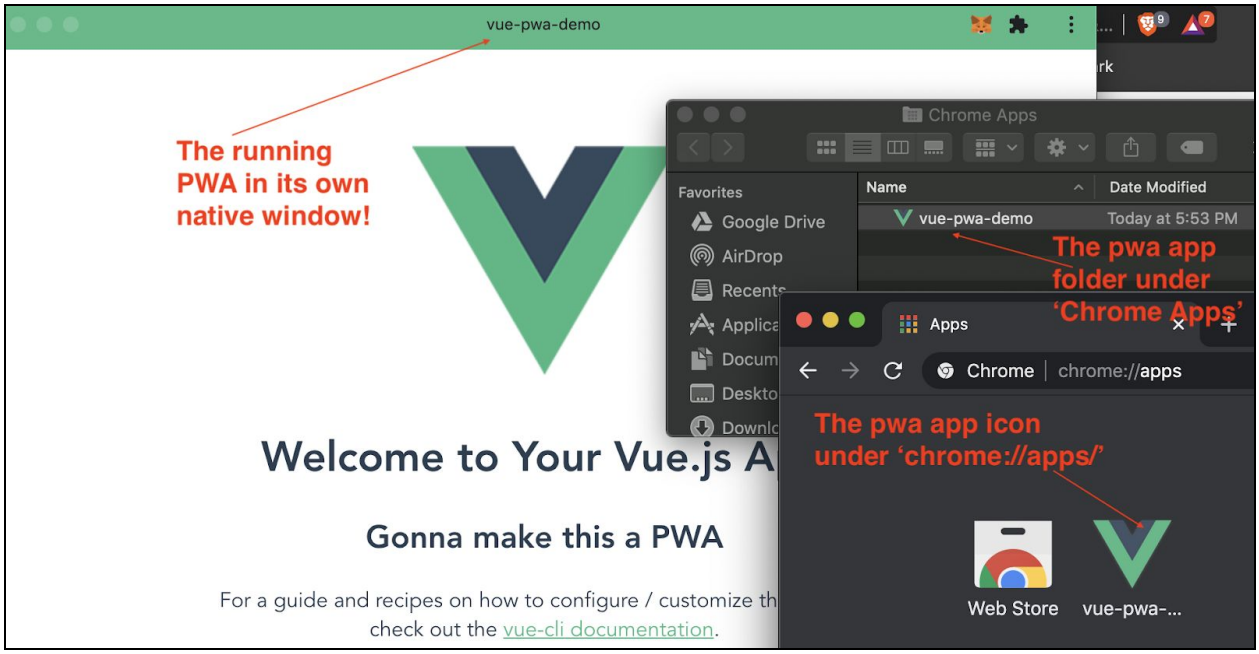
Uninstall the application

From chrome://apps/ find the application icon, right click and select ‘Remove from Chrome’

Installing PWA via localhost in Chrome Screenshots







Installing the PWA on a mobile (iPhone) when running via localhost in dev environment

In this scenario the PWA is running on the laptop and you want to connect to it from a browser running on a separate mobile device (iPhone). In this case the browser needs to access the PWA endpoint via a secure HTTPS connection. This can be achieved by the following steps:

Generate your own CA certificate

Reference

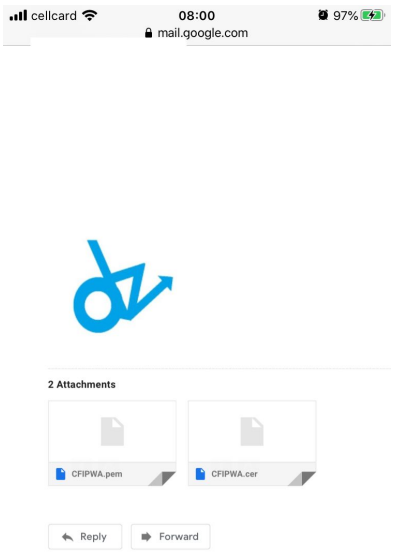
```
openssl genrsa -des3 -out myCA.key 2048
openssl req -x509 -new -nodes -key myCA.key -sha256 -days 1825 -out myCA.pem
```

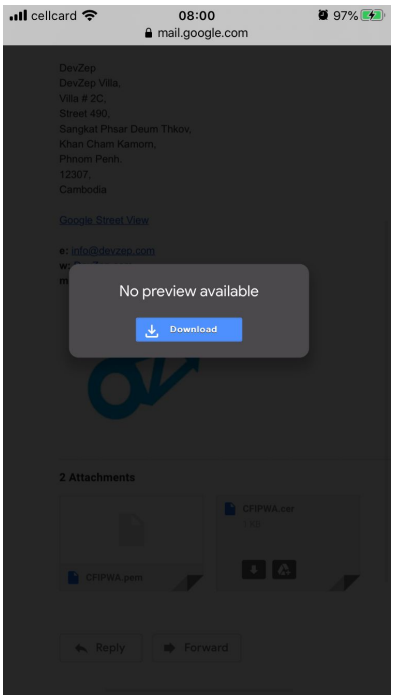
Install the CA certificate onto your Macbook keychain

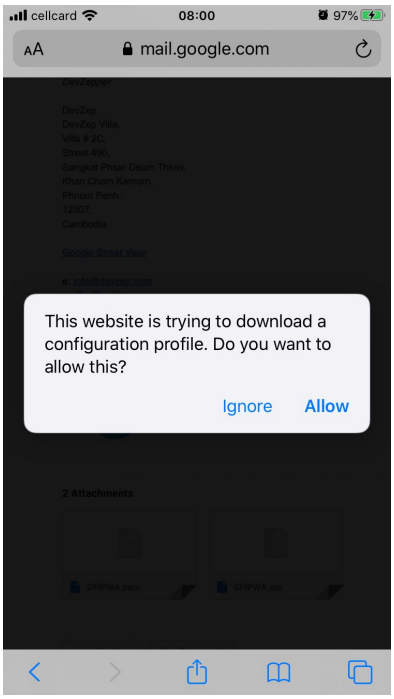
```
sudo security add-trusted-cert -d -r trustRoot -k "/Library/Keychains/System.keychain" myCA.pem
```

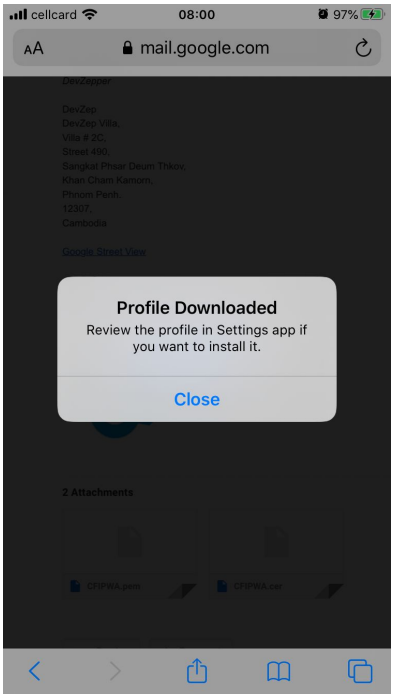
Install the CA certificate onto your iPhone

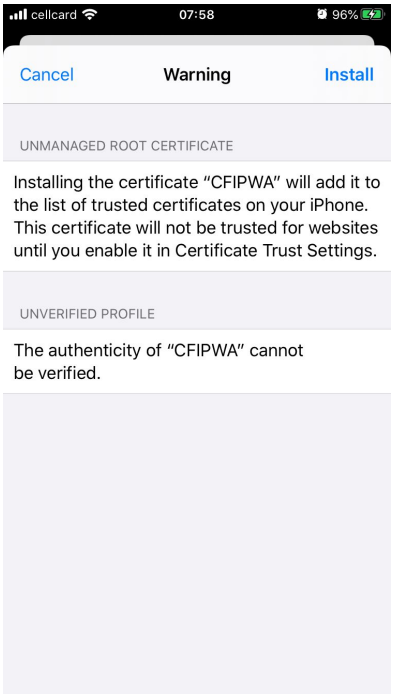
1. Email the `myCA.pem` file to your email as an attachment.
2. Log into your email client **using the Safari browser** (**very important** - the install will NOT work otherwise!) on your iPhone.
3. Open the email and click on the attachment.
4. Click to download the certificate.
5. You will be prompted to install it. Don't forget to trust it after under **"Certificate Trust Settings".** Below are the screenshots for reference:

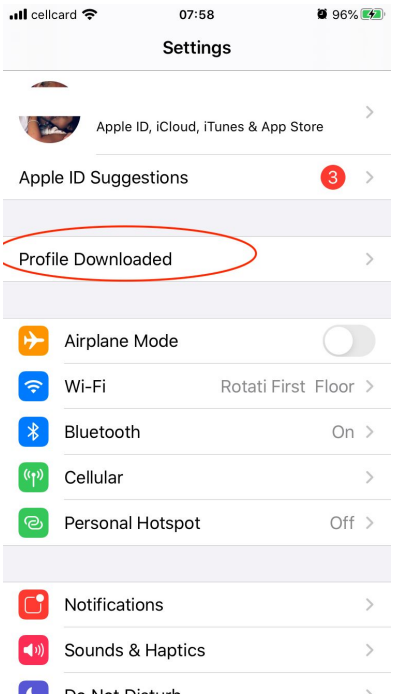


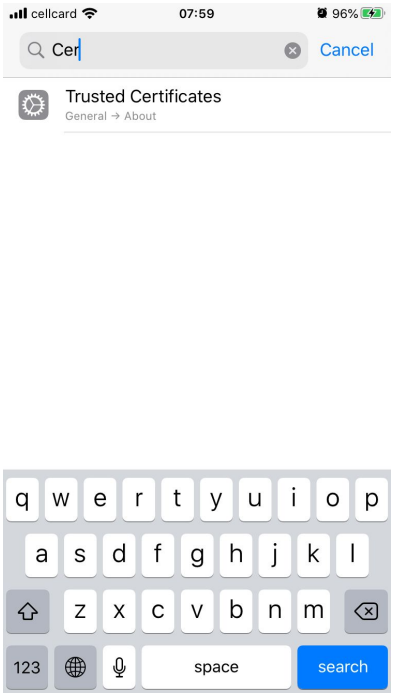


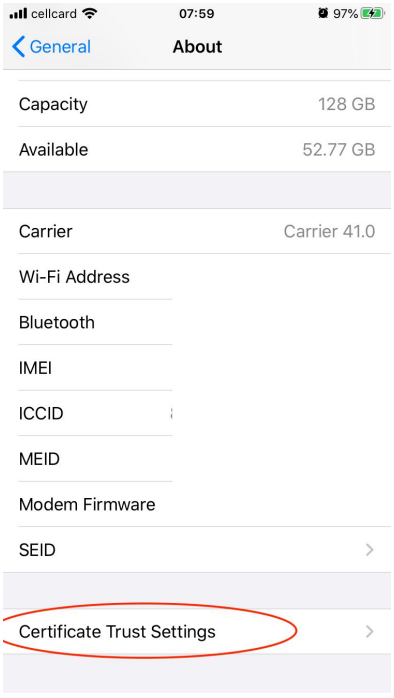


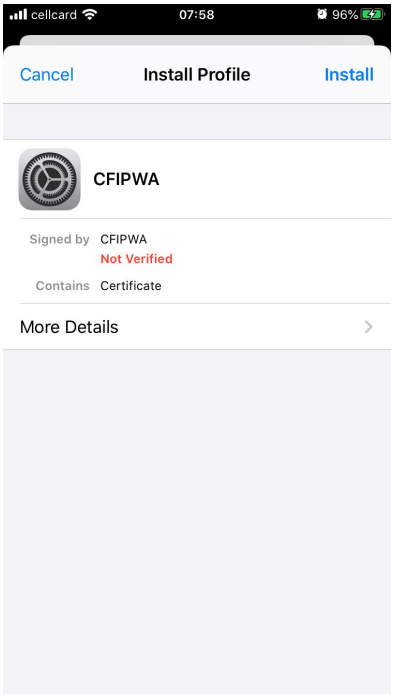

cellcard 08:00 97%
AA
mail.google.com
Profile Downloaded
Review the profile in Settings app if you want to install it.
Close

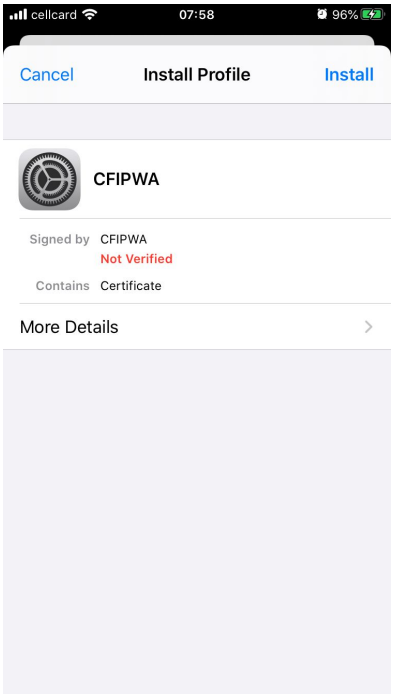

cellcard 07:58 96%
Cancel
Warning
Install
UNMANAGED ROOT CERTIFICATE
Installing the certificate "CFIPWA" will add it to the list of trusted certificates on your iPhone. This certificate will not be trusted for websites until you enable it in Certificate Trust Settings.
UNVERIFIED PROFILE
The authenticity of "CFIPWA" cannot be verified.

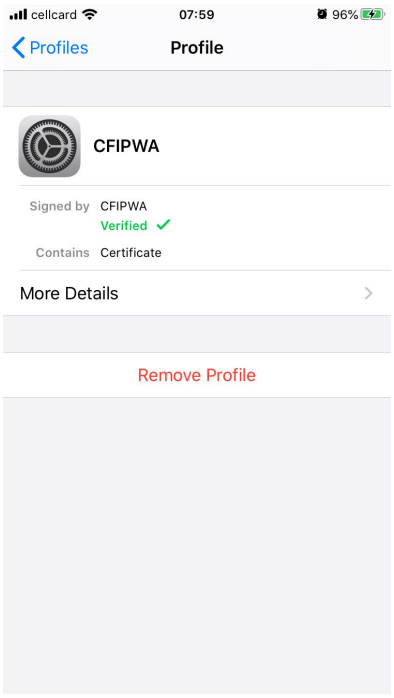

cellcard 07:58 96%
Settings
Apple ID, iCloud, iTunes & App Store
Apple ID Suggestions 3
Profile Downloaded
Airplane Mode
Wi-Fi Rotati First Floor
Bluetooth On
Cellular
Personal Hotspot Off
Notifications
Sounds & Haptics

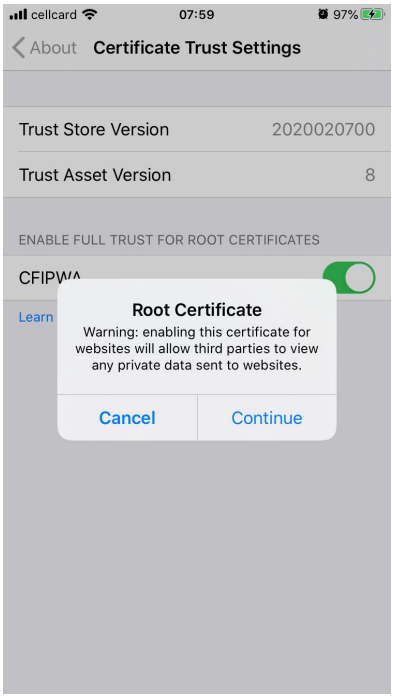

cellcard 07:59 96%
Cer
Cancel
Trusted Certificates
General → About
q w e r t y u i o p
a s d f g h j k l
z x c v b n m
123
space
search


cellcard 07:59 97%
General
About
Capacity 128 GB
Available 52.77 GB
Carrier Carrier 41.0
Wi-Fi Address
Bluetooth
IMEI
ICCID
MEID
Modem Firmware
SEID
Certificate Trust Settings


cellcard 07:58 96%
Cancel
Install Profile
Install
CFIPWA
Signed by CFIPWA
Not Verified
Contains Certificate
More Details







Now that you have your CA certificate created and installed on both your MacBook and your iPhone you need to create a specific domain Certificate using your new CA certificate as the authority

Use the below shell script to do that! The script can be found in this repo here and was taken from [this article reference]. Naturally, ensure this script is executable, then choose a local development domain that you want to use for development. I tend to use the format: `projectname.dev` for my local domain testing. In this example is `cfipwa.dev`.

```
#!/bin/sh

if [ "$#" -ne 1 ]
then
 echo "Usage: Must supply a domain"
 exit 1
fi

DOMAIN=$1

cd ~/certs

openssl genrsa -out $DOMAIN.key 2048
```

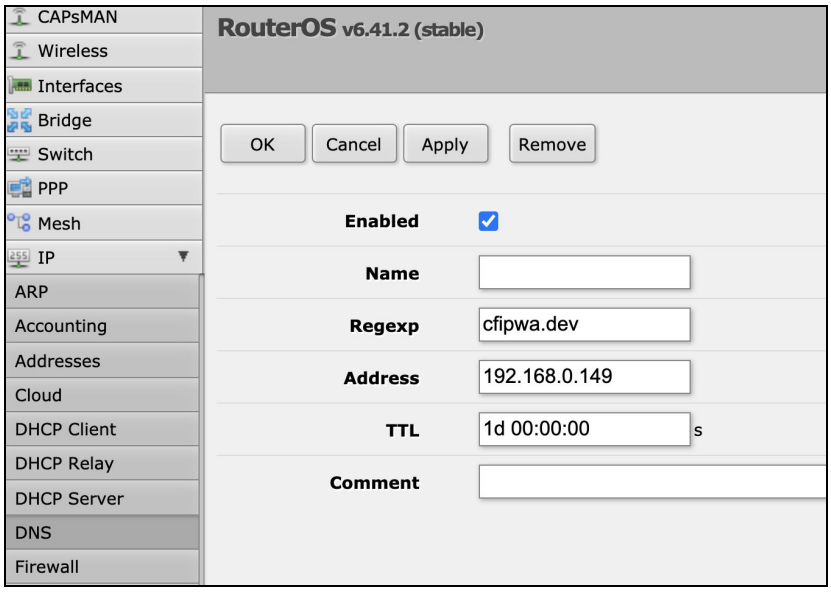
```
openssl req -new -key $DOMAIN.key -out $DOMAIN.csr

cat > $DOMAIN.ext << EOF
authorityKeyIdentifier=keyid,issuer
basicConstraints=CA:FALSE
keyUsage = digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment
subjectAltName = @alt_names
[alt_names]
DNS.1 = $DOMAIN
EOF

openssl x509 -req -in $DOMAIN.csr -CA ../myCA.pem -CAkey ../myCA.key -CAcreateserial \
-out $DOMAIN.crt -days 825 -sha256 -extfile $DOMAIN.ext
```

Update your /etc/hosts on your Mac to add DNS mapping to the dev domain and add:

`127.0.0.1 cfipwa.dev`

(where `cfipwa.dev` is the your local testing domain that you want to use).

Update your DNS connection between your phone and your Macbook so that the domain is routed to your IP address. One way is to add a static DNS entry to your local Router:



An alternative solution is to either use a DNS proxy server. Unfortunately, you cannot edit your /etc/hosts file directly on your iPhone so alternatives like these are required for this scenario.

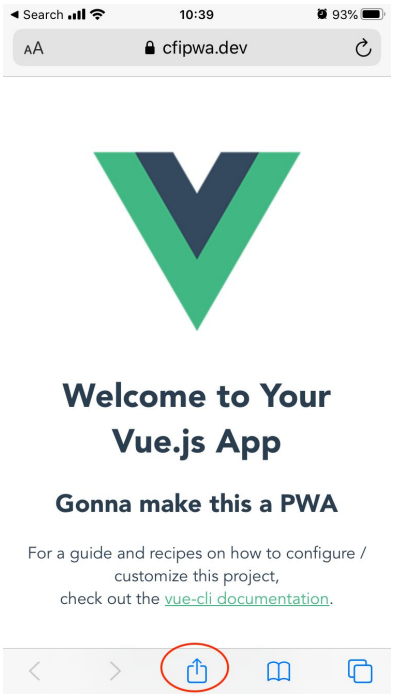
Run the node serve app passing in the new SSL certificate reference

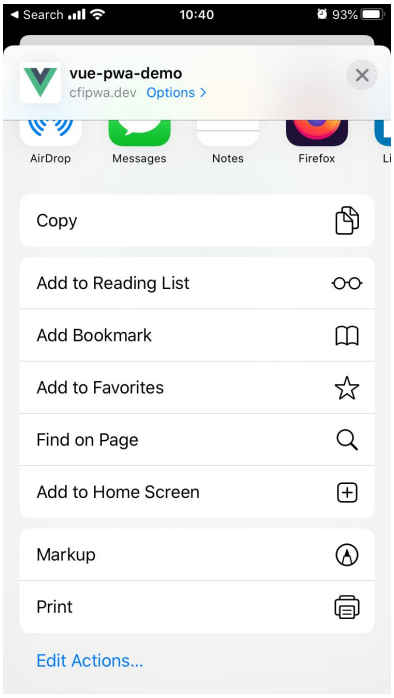
```
npx serve --ssl-cert cfipwa.dev.crt --ssl-key cfipwa.dev.key
```

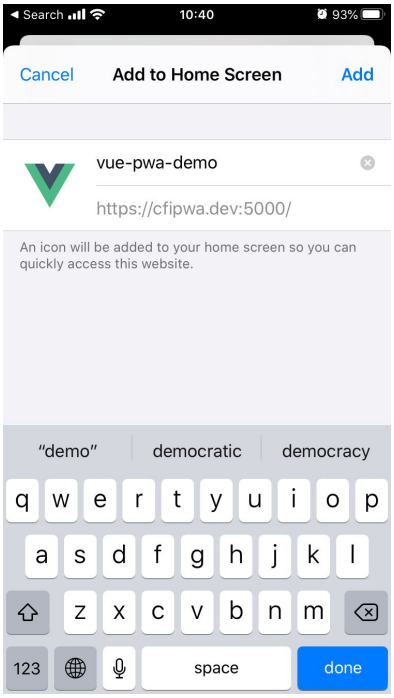
Open the https version in your iPhone Safari browser

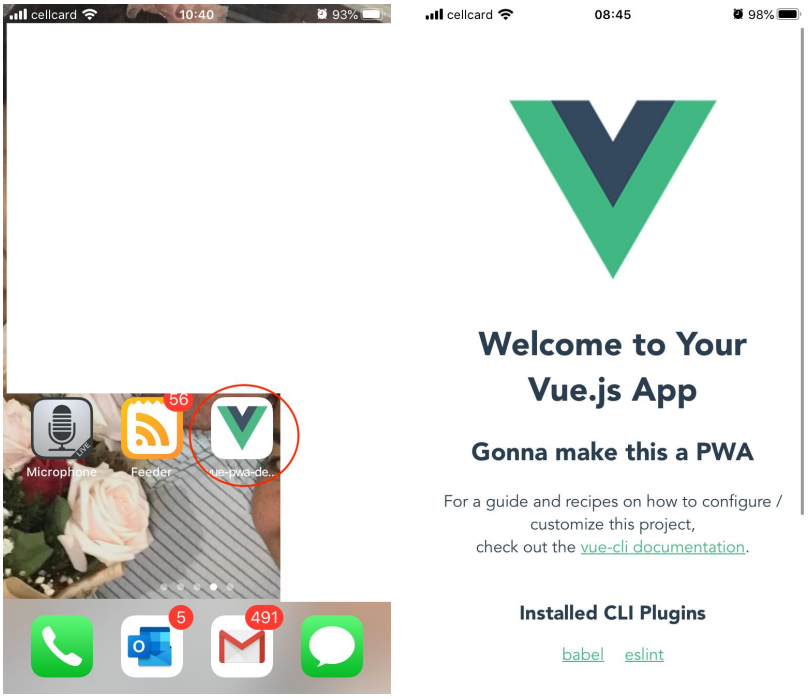
https://cfipwa.dev:5000/

Since you installed and trusted your CA Root certificate to your phone and you have added a static DNS entry for cfipwa.dev to your Router configuration the website will load in your iPhone safari browser using HTTPS! Now you can install this as a PWA by clicking the share button and selecting the *"add to home screen option"*. Check screenshots below:









Tutorials

https://medium.com/javascript-in-plain-english/how-to-create-a-progressive-web-app-with-vue-js-7086cbf87a43
https://www.simicart.com/blog/pwa-add-to-home-screen/
https://www.simicart.com/blog/pwa-app-stores/
https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps/Installable_PWAs
Vue JS PWA Tutorial https://www.youtube.com/watch?v=nTNvLv7Cf2A
VueJS Tutorial https://www.youtube.com/watch?v=BPyniDJ5QOQ

Tools

https://github.com/GoogleChrome/lighthouse
https://www.pwabuilder.com/
https://github.com/pwa-builder/pwa-starter