This video contains a great summary of how to enable Google Sign In for a cordova app: <https://youtu.be/TSax0iCRccc>

First create the cordova app:

cordova create journalapp com.cloudmatica.journal Journal

Add the android and browser platforms

cordova platform add android

cordova platform add browser

Create Keystore

cat cloudmatica-javakeys.sh

#https://alvinalexander.com/java/java-using-keytool-genkey-private-key-keystore/

#Used Cloudmatica codex password for keystore and foo private key

keytool -genkey -alias foo -keystore privateKey.store

Get fingerprint

cat get\_fingerprint.sh

keytool -keystore privateKey.store -list -v

Configure OAuth2 Client for Android

​

Install cordova plugin

cat install\_cordova\_plugin.sh

CLIENT\_ID=893193964658-8s0u87dfga7lr2352iemqa4hdum7114s.apps.googleusercontent.com

REVERSED\_CLIENT\_ID=com.googleusercontent.apps.893193964658-8s0u87dfga7lr2352iemqa4hdum7114s

cordova plugin add https://github.com/EddyVerbruggen/cordova-plugin-googleplus \

--save --variable REVERSED\_CLIENT\_ID=$REVERSED\_CLIENT\_ID \

--variable WEB\_APPLICATION\_CLIENT\_ID=$CLIENT\_ID

Add the following lines to www/index.html

<button onclick="gplusLogin()">Login</button>

<script>

document.addEventListener('deviceready', main)

function main() {

console.log('device is ready')

//ref = cordova.InAppBrowser.open('https://cloudmatica.com', '\_blank', 'location=yes');

}

function gplusLogin() {

window.plugins.googleplus.login({},

function(result) {alert(JSON.stringify(result))},

function(error) {alert(JSON.stringify(error))})

}

</script>

Now you should be able to test it by executing:

cordova run android --emulate

​