Android Lens Credit Cards

Adding Lens SDK to your project

1. Add the Maven repository details to the `dependencyResolutionManagement.repositories` section of your `settings.gradle` as shown in this minimalistic example:

```
dependencyResolutionManagement {
    repositories {
        google()
        mavenCentral()

        // Maven repository for VeryfiLens
        maven {
            url "https://nexus.veryfi.com/repository/maven-releases/"
            credentials {
                username = 'USERNAME'
                password = 'PASSWORD'
            }
            authentication {
                basic(BasicAuthentication)
            }
        }
    }
}
```

2. Add Veryfi Lens to the dependencies in your `build.gradle (:app)` file (replace "X.X.X" with the Lens Credit Cards SDK version you're currently using):

```
implementation 'com.veryfi.lens:veryfi-lens-credit-cards-sdk:X.X.X'
```

Configuring your project to use Lens Credit Cards SDK

1. Ensure your `build.gradle (:app)` file includes at a minimum the following configuration, plus any other required settings

```
plugins {
    id 'kotlin-kapt'
}

android {
    defaultConfig {
        minSdkVersion 21

        ndk {
            // Specifies the ABI configurations of your native app
            abiFilters "armeabi-v7a", "arm64-v8a", "x86", "x86_64"
        }

        javaCompileOptions {
            annotationProcessorOptions {
                arguments += [
                    "room.schemaLocation":"$projectDir/schemas".
toString(),
                    "room.incremental":"true",
                    "room.expandProjection":"true"]
            }
        }
    }

    androidResources {
        noCompress "tflite"
    }

    buildFeatures {
        dataBinding true
    }

    // Required if minSdkVersion is set to lower than 26
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}
```

2. Ensure your `build.gradle (:app)` file includes at a minimum the following dependencies (replace "X.X.X" with the Lens SDK version you're currently using) as well as the latest **AndroidX** libraries:

```
dependencies {
    implementation 'androidx.appcompat:appcompat:X.X.X'
    implementation 'com.veryfi.lens:veryfi-lens-credit-cards-sdk:X.X.X.
X'
    implementation "org.jetbrains.kotlin:kotlin-stdlib:$kotlin_version"
}
```

3. Ensure your main `build.gradle` file includes the required repositories and set Android Gradle Plugin to version 7:

```
allprojects {
    repositories {
        google()
        mavenCentral()
    }
}

buildscript {
    dependencies {
        classpath 'com.android.tools.build:gradle:7.0.2'
    }
}
```

4. Remove/disable `repositoriesMode` in your `settings.gradle` file :

```
dependencyResolutionManagement {
    // Disable the following line:
    // repositoriesMode.set(RepositoriesMode.FAIL_ON_PROJECT_REPOS)

    repositories {
        google()
        mavenCentral()
    }
}
```

5. If your app uses the `android:allowBackup` tag, include the following settings in the Activity section of your `AndroidManifest.xml` file :

```
<application
    tools:ignore="AllowBackup,GoogleAppIndexingWarning"
    tools:replace="android:allowBackup">
```

Initializing Lens SDK to your project

1. Import required classes from Lens SDK:

```
import com.verifi.lens.VerifiLens
import com.verifi.lens.VerifiLensCredentials
import com.verifi.lens.VerifiLensSettings
import com.verifi.lens.VerifiLensDelegate
```

2. Configure your authentication credentials:

```
val verifiLensCredentials = VerifiLensCredentials()
verifiLensCredentials.clientId = "XXX"  // replace XXX with your
assigned Client Id
verifiLensCredentials.username = "XXX"  // replace XXX with your
assigned Username
verifiLensCredentials.apiKey = "XXX"    // replace XXX with your
assigned API Key
verifiLensCredentials.url = "XXX"       // replace XXX with your
assigned Endpoint URL
```

3. Mandatory settings to use Lens Credit Cards. More available settings later in this section.

```
        val verifiLensSettings = VerifiLensSettings()
        verifiLensSettings.autoCaptureIsOn = true
        verifiLensSettings.autoRotateIsOn = true
        verifiLensSettings.autoSubmitDocumentOnCapture = true
        verifiLensSettings.documentTypes = arrayListOf("credit_card")
        verifiLensSettings.galleryIsOn = false
        verifiLensSettings.moreMenuIsOn = false
        verifiLensSettings.showDocumentTypes = true
```

4. Initialize Lens:

```
VerifiLens.configure(this, verifiLensCredentials, verifiLensSettings)
```

5. Launch the Lens camera

```
VerifiLens.showCamera()
```

Available settings:

`docDetectFillUIColor`: card detection rectangle fill color (default: *"#9653BF8A"*)

`docDetectStrokeUIColor`: card detection rectangle stroke color (default: *null*)

Communicating with Lens

1. Use the `VeryfiLens.setDelegate` function to define your delegate functions that will be responsible for handling events triggered by Veryfi Lens

```
VeryfiLens.setDelegate(object : VeryfiLensDelegate {
    override fun veryfiLensClose(json: JSONObject) {
        Log.d(TAG, json.toString(2)) // do something with the JSON
response here
    }

    override fun veryfiLensError(json: JSONObject) {
        Log.d(TAG, json.toString(2)) // do something with the JSON
response here
    }

    override fun veryfiLensSuccess(json: JSONObject) {
        Log.d(TAG, json.toString(2)) // do something with the JSON
response here
    }

    override fun veryfiLensUpdate(json: JSONObject) {
        Log.d(TAG, json.toString(2)) // do something with the JSON
response here
    }
})
```

`veryfiLensClose` - the Veryfi Lens camera has been closed, either as a result of submitting a credit card for processing, or the user closed the camera without submitting a card.

```
{
    "status": "close",
    "queue_count": 1,
    "session_scan_count": 1,
    "framework-version": "1.4.0",
    "framework-build": "1"
}
```

`veryfiLensUpdate` - during the processing of a credit card, this delegate will be fired multiple times.

```
{
    "status": "start",
    "package_id": "edc8653e4c2b4ef1"
}
```

`veryfiLensSuccess` - this delegate fires once a credit card has finished processing.This delegate provides the response from the data extraction.

```
{
      "status": "done",
      "msg": "results",
      "data": {
        "card_number": "5143 3201 1937 4709",
        "card_name": "ALEJANDRO URIBE",
        "card_type": "mastercard",
        "card_dates": [
          "1\/10",
          ""
        ]
      },
      "package_id": "f7d3d73ba7264742"
    }
```


`veryfiLensError` - if an error occurs during processing a submitted card, an error object is sent. If a general exception or crash is caught in Veryfi Lens, an exception object is sent instead


```
{
    "status": "error",
    "package_id": "f7d3d73ba7264742",
    "error": "Error message"
}
```


Please note: adding Android Lens Credit Cards SDK to your app will increase your final app size by up to 12.6 MB. This is due to machine learning models, support libraries, etc included in the SDK.