

## API Schemas

The schemas below are representative solution API patterns and are subject to final detailed design and alignment with the selected onboarding and verification vendors.

### APIs for Digital Onboarding with OCR/eKYC

#### Group Mobile/Web and OEM Layer

##### *Purpose*

This group defines the APIs exposed by the OEM Layer to customer-facing channels such as:

- Mobile App
- Web Portal
- Assisted onboarding channels

These APIs are responsible for:

- creating the onboarding journey
- receiving document and selfie uploads
- submitting OCR/eKYC requests
- polling status
- confirming extracted data
- continuing the onboarding flow

These are solution APIs, not Curity-native APIs.

##### *Common request pattern for Mobile/Web APIs*

For all APIs from Mobile/Web to OEM Layer, device information should be passed as a single object.

##### **Example request pattern**

```
{  
  "channel": "mobile",  
  "locale": "en-US",  
  "device": {  
    "deviceId": "sample-device-id",  
    "appInstanceId": "sample-app-instance-id",  
    "platform": "iOS",
```

```
"platformVersion": "18.2",
"appVersion": "6.2.0",
"telemetry": {},
"network": {},
"security": {},
"additionalAttributes": {}
}
}
```

### **Device object meaning**

The device object is a container for all collected device-related information. Its internal schema may be extended depending on:

- device intelligence requirements
- anti-fraud requirements
- app integrity checks
- device binding requirements
- regulatory or bank-specific requirements

### *API Create Onboarding Journey*

#### **Purpose**

Creates a new onboarding journey and registration case.

#### **Endpoint**

POST /api/v1/onboarding/journeys

#### **Request**

```
{
  "channel": "mobile",
  "journeyType": "service-registration",
  "customerRef": null,
  "locale": "en-US",
  "device": {
    "deviceId": "sample-device-id",
```

```

"appId": "sample-app-id",
"appName": "Sample App",
"appInstanceId": "sample-app-instance-id",
"platform": "iOS",
"platformVersion": "18.2",
"appVersion": "6.2.0",
"telemetry": {},
"network": {},
"security": {},
"additionalAttributes": {}
}
}

```

### Request fields

Field	Type	Required	Meaning
channel	string	yes	Source channel such as mobile or web
journeyType	string	yes	Business onboarding journey type
customerRef	string / null	no	Existing customer reference if any
locale	string	no	Language and regional setting
device	object	yes	Aggregated device context

### Response

```

{
  "code": "0000",
  "message": "Success",
  "data": {
    "journeyId": "jrny_100001",
    "registrationId": "reg_100001",
    "status": "INITIATED",
    "nextAction": "UPLOAD_DOCUMENT"
  },
  "errors": [],
}

```

```

"meta": {
  "requestId": "req_100001",
  "timestamp": "2025-10-14T10:15:00Z"
}
}

```

### *API Upload Document*

#### **Purpose**

Uploads a document image and associates it with the current onboarding case.

#### **Endpoint**

POST /api/v1/onboarding/journeys/{journeyId}/documents

Content-Type: multipart/form-data

#### **Multipart fields**

Field	Type	Required	Meaning
registrationId	string	yes	Registration case identifier
documentType	string	yes	Document type, for example passport or national_id
documentSide	string	yes	front, back, or full
captureMode	string	no	camera or gallery
countryCode	string	no	Country code
file	binary	yes	Uploaded document file
device	json string	yes	Device object serialized as JSON

#### **Example device JSON part**

```

{
  "deviceId": "sample-device-id",
  "appInstanceId": "sample-app-instance-id",
  "deviceFingerprint": "sample-device-fingerprint",
  "platform": "Android",
  "platformVersion": "15",

```

```
"appVersion": "7.0.1",
"telemetry": {},
"network": {},
"security": {},
"additionalAttributes": {}
}
```

## Response

```
{
  "code": "0000",
  "message": "Document uploaded",
  "data": {
    "journeyId": "jrny_100001",
    "registrationId": "reg_100001",
    "documentRef": "doc_front_100001",
    "status": "DOCUMENT_UPLOADED",
    "nextAction": "SUBMIT_OCR"
  },
  "errors": [],
  "meta": {
    "requestId": "req_100002",
    "timestamp": "2025-10-14T10:16:10Z"
  }
}
```

## *API Submit OCR Request*

### **Purpose**

Submits uploaded document references for OCR extraction.

### **Endpoint**

POST /api/v1/onboarding/journeys/{journeyId}/ocr

### **Request**

```

{
  "registrationId": "reg_100001",
  "mode": "synchronous",
  "documents": [
    {
      "documentRef": "doc_front_100001",
      "documentType": "passport",
      "documentSide": "front"
    }
  ],
  "device": {
    "deviceId": "sample-device-id",
    "appInstanceId": "sample-app-instance-id",
    "deviceFingerprint": "sample-device-fingerprint",
    "platform": "iOS",
    "platformVersion": "18.2",
    "appVersion": "6.2.0",
    "telemetry": {},
    "network": {},
    "security": {},
    "additionalAttributes": {}
  }
}

```

### Request fields

Field	Type	Required	Meaning
registrationId	string	yes	Registration case ID
mode	string	yes	synchronous or asynchronous
documents	array	yes	List of document references

Field	Type	Required	Meaning
device	object	yes	Aggregated device context

### Response

```
{
  "code": "0000",
  "message": "OCR completed",
  "data": {
    "journeyId": "jrny_100001",
    "registrationId": "reg_100001",
    "verificationId": "ver_ocr_100001",
    "status": "OCR_COMPLETED",
    "ocrResult": {
      "documentType": "passport",
      "fullName": "JOHN WILLIAM CARTER",
      "documentNumber": "P12345678",
      "dateOfBirth": "1991-08-17",
      "issueDate": "2023-02-01",
      "expiryDate": "2033-02-01",
      "nationality": "USA",
      "confidenceScore": 0.97
    },
    "nextAction": "CONFIRM_EXTRACTED_DATA"
  },
  "errors": [],
  "meta": {
    "requestId": "req_100003",
    "timestamp": "2025-10-14T10:16:50Z"
  }
}
```

## *API Poll Verification Result*

### **Purpose**

Retrieves verification progress and current status for asynchronous OCR/eKYC processing.

### **Endpoint**

GET /api/v1/onboarding/cases/{registrationId}/status

### **Response**

```
{
  "code": "0000",
  "message": "Verification status retrieved",
  "data": {
    "registrationId": "reg_100010",
    "verificationId": "ver_ekyc_100010",
    "status": "PENDING_VERIFICATION",
    "progress": {
      "documentUpload": "DONE",
      "ocr": "DONE",
      "documentAuthenticity": "DONE",
      "watchlist": "IN_PROGRESS"
    }
  },
  "errors": [],
  "meta": {
    "requestId": "req_100010",
    "timestamp": "2025-10-14T10:20:00Z"
  }
}
```

## *API Confirm Extracted Data*

### **Purpose**

Allows the customer to confirm or correct OCR-extracted data before the onboarding flow continues.

### Endpoint

POST /api/v1/onboarding/journeys/{journeyId}/confirm-ocr

### Request

```
{  
  "registrationId": "reg_100001",  
  "verificationId": "ver_ocr_100001",  
  "confirmed": true,  
  "correctedData": {  
    "fullName": "JOHN WILLIAM CARTER",  
    "documentNumber": "P12345678",  
    "dateOfBirth": "1991-08-17"  
  },  
  "device": {  
    "deviceId": "sample-device-id",  
    "appInstanceId": "sample-app-instance-id",  
    "deviceFingerprint": "sample-device-fingerprint",  
    "platform": "iOS",  
    "platformVersion": "18.2",  
    "appVersion": "6.2.0",  
    "telemetry": {},  
    "network": {},  
    "security": {},  
    "additionalAttributes": {}  
  }  
}
```

### Response

```
{
```

```

"code": "0000",
"message": "OCR data confirmed",
"data": {
  "journeyId": "jrny_100001",
  "registrationId": "reg_100001",
  "status": "CONFIRMED",
  "nextAction": "CONTINUE_ONBOARDING"
},
"errors": [],
"meta": {
  "requestId": "req_100004",
  "timestamp": "2025-10-14T10:17:20Z"
}
}

```

### *API Upload Selfie / Video Selfie*

#### **Purpose**

Uploads selfie evidence for face match and liveness verification.

#### **Endpoint**

POST /api/v1/onboarding/cases/{registrationId}/selfie

Content-Type: multipart/form-data

#### **Multipart fields**

Field	Type	Required	Meaning
captureType	string	yes	selfie_image or selfie_video
file	binary	yes	Selfie image or video
registrationId	string	yes	Registration case ID
device	json string	yes	Serialized device object

#### **Response**

```
{
```

```
"code": "0000",
"message": "Selfie uploaded",
"data": {
  "registrationId": "reg_100020",
  "selfieRef": "selfie_100020",
  "status": "DOCUMENT_UPLOADED",
  "nextAction": "RUN_BIOMETRIC_CHECK"
},
"errors": [],
"meta": {
  "requestId": "req_100020",
  "timestamp": "2025-10-14T10:22:00Z"
}
}
```

### *API Trigger Biometric Verification*

#### **Purpose**

Triggers face match and liveness verification using document and selfie references.

#### **Endpoint**

POST /api/v1/onboarding/cases/{registrationId}/biometric-check

#### **Request**

```
{
  "documentRef": "doc_front_100020",
  "selfieRef": "selfie_100020",
  "checks": ["face-match", "liveness"],
  "device": {
    "deviceId": "sample-device-id",
    "appInstanceId": "sample-app-instance-id",
    "deviceFingerprint": "sample-device-fingerprint",
```

```
"platform": "Android",
"platformVersion": "15",
"appVersion": "7.0.1",
"telemetry": {},
"network": {},
"security": {},
"additionalAttributes": {}
}
}
```

### **Response**

```
{
"code": "0000",
"message": "Biometric verification completed",
"data": {
"registrationId": "reg_100020",
"verificationId": "ver_bio_100020",
"status": "BIOMETRIC_COMPLETED",
"biometricResult": {
"faceMatchScore": 0.98,
"livenessScore": 0.95,
"decision": "APPROVED"
},
"nextAction": "CONTINUE_ONBOARDING"
},
"errors": [],
"meta": {
"requestId": "req_100021",
"timestamp": "2025-10-14T10:22:35Z"
}
}
```

```
}
```

## *API Continue Onboarding*

### **Purpose**

Continues the onboarding flow after OCR confirmation and optional biometric verification.

### **Endpoint**

POST /api/v1/onboarding/journeys/{journeyId}/continue

### **Request**

```
{  
  "registrationId": "reg_100001",  
  "steps": [  
    "reconcile-customer",  
    "evaluate-policy",  
    "synchronize-with-curity"  
  ],  
  "device": {  
    "deviceId": "sample-device-id",  
    "appInstanceId": "sample-app-instance-id",  
    "deviceFingerprint": "sample-device-fingerprint",  
    "platform": "iOS",  
    "platformVersion": "18.2",  
    "appVersion": "6.2.0",  
    "telemetry": {},  
    "network": {},  
    "security": {},  
    "additionalAttributes": {}  
  }  
}
```

### **Response**

```
{
  "code": "0000",
  "message": "Onboarding continuation accepted",
  "data": {
    "journeyId": "jrny_100001",
    "registrationId": "reg_100001",
    "status": "PENDING_VERIFICATION",
    "nextAction": "WAIT"
  },
  "errors": [],
  "meta": {
    "requestId": "req_100005",
    "timestamp": "2025-10-14T10:18:10Z"
  }
}
```

## Group OEM Layer and OCR/eKYC Engine

### *Purpose*

This group defines the backend connector APIs between the OEM Layer and the OCR/eKYC Engine. These APIs are used for:

- creating verification jobs/cases
- submitting files or references
- querying status
- receiving callback results
- submitting biometric verification
- receiving biometric callback results

These are integration APIs, not Curity-native APIs.

### *API Create Verification Case*

#### **Purpose**

Creates a verification case in the external OCR/eKYC engine.

## Endpoint

POST /vendor-api/v1/verifications/cases

## Request

```
{  
  "requestId": "req_200001",  
  "registrationId": "reg_100010",  
  "flowType": "ocr-ekyc",  
  "callbackUrl": "https://oem.example.com/api/v1/vendor/callbacks/ekyc",  
  "customerRef": null,  
  "countryCode": "US",  
  "documentType": "passport"  
}
```

## Request fields

Field	Type	Required	Meaning
requestId	string	yes	Request ID generated by OEM layer
registrationId	string	yes	Registration case ID
flowType	string	yes	Verification flow type
callbackUrl	string	no	Callback URL for async result
customerRef	string / null	no	Existing customer reference
countryCode	string	no	Country code
documentType	string	yes	Main document type

## Response

```
{  
  "verificationId": "ver_ekyc_200001",  
  "status": "ACCEPTED"  
}
```

*API Submit File or Reference*

## Purpose

Submits files or document references to the external OCR/eKYC engine.

### Endpoint

POST /vendor-api/v1/verifications/{verificationId}/documents

### Request

```
{
  "registrationId": "reg_100010",
  "documents": [
    {
      "documentRef": "doc_front_100010",
      "documentSide": "front",
      "fileName": "passport_front.jpg",
      "contentType": "image/jpeg",
      "contentBase64": "<base64-document-content>"
    },
    {
      "documentRef": "doc_back_100010",
      "documentSide": "back",
      "fileName": "passport_back.jpg",
      "contentType": "image/jpeg",
      "contentBase64": "<base64-document-content>"
    }
  ]
}
```

### Response

```
{
  "verificationId": "ver_ekyc_200001",
  "status": "SUBMITTED"
}
```

### *API Query Verification Status*

#### **Purpose**

Queries the external verification case status.

#### **Endpoint**

GET /vendor-api/v1/verifications/{verificationId}

#### **Response**

```
{
  "verificationId": "ver_ekyc_200001",
  "status": "IN_PROGRESS",
  "stages": {
    "ocr": "DONE",
    "documentAuthenticity": "DONE",
    "watchlist": "IN_PROGRESS",
    "fraudCheck": "PENDING"
  }
}
```

### *API Receive OCR/eKYC Callback Result*

#### **Purpose**

Receives asynchronous callback results from the OCR/eKYC engine.

#### **Endpoint**

POST /api/v1/vendor/callbacks/ekyc

#### **Request**

```
{
  "verificationId": "ver_ekyc_200001",
  "registrationId": "reg_100010",
  "callbackRef": "cb_200001",
  "status": "COMPLETED",
  "decision": "APPROVED",
  "result": {
```

```
"fullName": "EMILY GRACE ROBERTS",
"documentNumber": "P98765432",
"dateOfBirth": "1994-03-12",
"issueDate": "2024-05-15",
"expiryDate": "2034-05-15",
"documentValid": true,
"watchlistHit": false,
"confidenceScore": 0.94
},
"referenceId": "vendor_ref_ekyc_001"
}
```

### **Response**

```
{
  "acknowledged": true
}
```

### *API Submit Biometric Verification*

### **Purpose**

Submits face match and liveness verification to the biometric engine.

### **Endpoint**

POST /vendor-api/v1/biometric/verifications

### **Request**

```
{
  "requestId": "req_200020",
  "registrationId": "reg_100020",
  "documentRef": "doc_front_100020",
  "selfieRef": "selfie_100020",
  "checks": ["face-match", "liveness"],
  "callbackUrl": "https://oem.example.com/api/v1/vendor/callbacks/biometric"
}
```

## Response

```
{  
  "verificationId": "ver_bio_200020",  
  "status": "ACCEPTED"  
}
```

### *API Receive Biometric Callback Result*

## Purpose

Receives asynchronous biometric results from the biometric verification engine.

## Endpoint

POST /api/v1/vendor/callbacks/biometric

## Request

```
{  
  "verificationId": "ver_bio_200020",  
  "registrationId": "reg_100020",  
  "status": "COMPLETED",  
  "result": {  
    "faceMatchScore": 0.98,  
    "livenessScore": 0.95,  
    "decision": "APPROVED"  
  },  
  "referenceId": "vendor_ref_bio_001"  
}
```

## Response

```
{  
  "acknowledged": true  
}
```