افزونه EKYC

- امضای الکترونیک
- احراز هویت تصویری

امضای الکترونیک

افزونه امضای شامل چندین عملکرد است که امکان مدیریت و استفاده از قابلیت‌های امضا را فراهم می‌کند. این عملکردها به شرح زیر هستند:

1. signmanager/initialize

این عملکرد برای فعال سازی عملکرد امضا برای مینی اپ استفاده می‌شود. برای استفاده از متد های امضا این متد حتما باید صدا شده باشد در غیر این صورت متد های دیگر اجرا نمی شوند. این عملکرد نیاز به 9 پارامتر دارد که شامل کد ملی، شماره موبایل، تاریخ تولد، شماره سریال کارت، نام، نام خانوادگی، نام انگلیسی، نام خانوادگی انگلیسی و ایمیل می‌باشد.

فراخوانی در اسکریپت با استفاده از `Appsan.runAction()`:

```
Appsan.runAction("signmanager/initialize(\"1234567890\",\"09123456789\",\"1370/01/01\",\"12345\",\"نام\",\"نام‌خانوادگی\",\"FirstName\",\"LastName\",\"email@example.com\")");
```

فراخوانی در XML:

```
<Button onclick="signmanager/initialize('1234567890','09123456789','1370/01/01','12345','نام','نام‌خانوادگی','FirstName','LastName','email@example.com')" />
```

2. signmanager/isIdentityProvided

این عملکرد بررسی می‌کند که آیا اطلاعات هویتی توسط کاربر فراهم شده است یا خیر. نتیجه این بررسی به صورت یک مقدار بولین برگردانده می‌شود. **فراخوانی در اسکریپت با استفاده از** `Appsan.runAction()`:

```
Appsan.runAction("signmanager/isIdentityProvided()");
```

فراخوانی در XML:

```
<Button onclick="signmanager/isIdentityProvided()" />
```

3. signmanager/getRegistrationObject

این عملکرد شیء ثبت نامی را که توسط SignManager مدیریت می‌شود، برمی‌گرداند. **فراخوانی از اسکریپت با استفاده از** `Appsan.runAction()`:

```
Appsan.runAction("signmanager/getRegistrationObject()");
```

فراخوانی در XML:

```
<Button onclick="signmanager/getRegistrationObject()" />
```

4. signmanager/jsonStamp

این عملکرد یک رشته ی base64 شده را دریافت کرده و نسخه مهر شده‌ی آن را به صورت json برمی‌گرداند. **فراخوانی از اسکریپت با استفاده از** `Appsan.runAction()`:

```
Appsan.runAction("signmanager/jsonStamp('{\"key\":\"value\"}')");
```

فراخوانی در XML:

```
<Button onclick="signmanager/jsonStamp('{\"key\":\"value\"}')" />
```

5. signmanager/setRegistrationObjectResponse

این عملکرد برای تنظیم پاسخ شیء ثبت نامی در SignManager استفاده می‌شود. **فراخوانی از اسکریپت با استفاده از** `Appsan.runAction()`:

```
Appsan.runAction("signmanager/setRegistrationObjectResponse('{\"registrationObject\":\"value\"
}')");
```

فراخوانی در XML:

```
<Button
onclick="signmanager/setRegistrationObjectResponse('{\"registrationObject\":\"value\"}')" />
```

6. signmanager/getIdentityKeySpec

این عملکرد کلید مشخصات هویتی را که توسط SignManager مدیریت می‌شود، برمی‌گرداند.
فراخوانی در اسکریپت با استفاده از `Appsan.runAction()`:

```
Appsan.runAction("signmanager/getIdentityKeySpec()");
```

فراخوانی در XML:

```
<Button onclick="signmanager/getIdentityKeySpec()" />
```

7. signmanager/docDeCipher

این عملکرد یک سند را دریافت کرده و آن را رمزگشایی می‌کند و نتیجه به صورت رشته‌ای که شامل دو مقدار است، برگردانده می‌شود. **فراخوانی در اسکریپت با استفاده از** `Appsan.runAction()`:

```
Appsan.runAction("signmanager/docDeCipher('encryptedDocument')");
```

فراخوانی در XML:

```
<Button onclick="signmanager/docDeCipher('encryptedDocument')" />
```

8. signmanager/docToCipher

این عملکرد یک سند را دریافت کرده و آن را رمزنگاری می‌کند و نسخه رمزنگاری شده را به صورت یک رشته برمی‌گرداند. **فراخوانی در اسکریپت با استفاده از** `Appsan.runAction()`:

```
Appsan.runAction("signmanager/docToCipher('documentToEncrypt')");
```

فراخوانی در XML:

```
<Button onclick="signmanager/docToCipher('documentToEncrypt')" />
```

این نمونه‌ها نشان می‌دهند که چگونه می‌توانید هر یک از عملکردهای افزونه "امضا" را با استفاده از دو روش `Appsan.runAction()` و فراخوانی مستقیم در XML فراخوانی کنید.

نمونه کد تست

```
<page title="نمایش نتایج روی تستم" scrollable="false">
    <button text=\"init\" onclick=\"signmanager/initialize('1111111111', '09123456789',
'1370/01/01',
    '0x123456789', ' اسم', 'فامیل ', 'First','last', 'example@gmail.com')"/>
    <button text="isIdent" onclick="signmanager/isIdentityProvided->script/s:res($data)"/>
<script id=\"s\">function res(data) {console.log(data)}"</script>
</page>
```

احراز هویت تصویری

افزونه احراز هویت تصویری شامل چندین عملکرد است که امکان مدیریت و استفاده از قابلیت‌های eKYC را فراهم می‌کند. این عملکردها به شرح زیر هستند:

1. ekyc/generateToken

این عملکرد برای تولید توکن احراز هویت تصویری استفاده می‌شود. این متد باید فراخوانی شود تا عملیات احراز هویت شروع شود. این عملکرد نیاز به 2 پارامتر دارد: دامنه، شماره سریال کارت.

فراخوانی در اسکریپت با استفاده از `Appsan.runAction()`:

```
Appsan.runAction("ekyc/generateToken('SCOPE', 'CARD_SERIAL_NO')");
```

فراخوانی در XML:

```
<Button onclick="ekyc/generateToken('SCOPE', 'CARD_SERIAL_NO')" />
```

در ورودی scope می توانید از مقدار 0300000 استفاده نمایید و ورودی دوم سریال پشت کارت ملی می باشد.

نمونه خروجی

```
{
   "status": "success",
   "orderId": "{{orderId}}"
}
```

2. ekyc/getTokenData

این عملکرد اطلاعات مرتبط با توکن تولید شده را برمی‌گرداند و برای دریافت داده‌های احراز هویتی استفاده می‌شود. این متد به سه پارامتر نیاز دارد: نام اپلیکیشن، نوع داده احراز هویت، و نوع استعلام.

فراخوانی در اسکریپت با استفاده از `Appsan.runAction()`:

```
Appsan.runAction("ekyc/getTokenData('authenticationType', 'inquiryType')");
```

فراخوانی در XML:

```
<Button onclick="ekyc/getTokenData('authenticationType', 'inquiryType')" />
```

ورودی اول نوع پاسخ را مشخص می‌کند و یکی از مقادیر زیر به صورت عدد داخل string می باشد.

```
public enum AuthenticationData {
    @SerializedName("1")
    AUTHENTICATION_RESULT((byte) 1),

    @SerializedName("2")
    SELFIE_PICTURE((byte) 2),

    @SerializedName("3")
    VIDEO((byte) 3),

    @SerializedName("4")
    SPEECH((byte) 4),

    @SerializedName("5")
    SIGNATURE((byte) 5);
}
```

ورودی دوم نوع داده را مخص می کند

```
enum InquiryType {
  GET_TOKEN_DATA = 0;
  GET_CITIZEN_TOKENS_BY_NATIONAL_ID = 1;
  GET_CITIZEN_LAST_AUTHENTICATED_DATA = 2;
//  REFRESH_AND_GET_TOKEN_DATA = 3;
}
```

این نمونه‌ها نشان می‌دهند که چگونه می‌توانید هر یک از عملکردهای افزونه "احراز هویت تصویری" را با استفاده از دو روش `Appsan.runAction()` و فراخوانی مستقیم در XML فراخوانی کنید.

نمونه خروجی

```
{
  "status": "وضعیت کلی پاسخ (مثلاً موفق یا ناموفق)",
  "responseCode": "کد پاسخ (مثلاً 200 برای موفقیت)",
  "responseMessage": "پیام توضیحی درباره نتیجه درخواست",
  "acceptanceStatus": "وضعیت پذیرش درخواست (مثلاً پذیرفته‌شده یا ردشده)",
  "data": "داده‌های بازگردانده‌شده در پاسخ",
  "dataType": "نوع داده موجود در فیلد data",
  "authenticationErrorCode": "کد خطای احراز هویت (در صورت وجود)",
  "orderId": "شناسه سفارش مربوط به درخواست",
  "insertDateTime": "زمان ثبت یا درج اطلاعات"
}
```

نمونه کد تست

```
<page title="احراز هویت تصویری" scrollable="false">
    <button text="Generate Token" onclick="ekyc/generateToken('SCOPE', '123456')" />
    <button text="Get Token Data" onclick="ekyc/getTokenData('1','GET_TOKEN_DATA')-
>script/s:result2($data)" />
    <script id="s1">
        function res(data) {
            console.log(data);
        }
        function result2(res) {
            console.log(res);
        }
    </script>
</page>
```

> برای اطلاع از ورودی و خروجی دقیق متد ها به مستندات ekyc اصلی مراجعه شود.