**Для інтеграції у веб. портал**

Зміст

[Для інтеграції у веб.портал 1](#_Toc91078968)

[1. Створюємо вікно, в якому запитуємо логін та пароль користувача. 1](#_Toc91078969)

[2. Підписання документу 3](#_Toc91078970)

**1. Створюємо вікно, в якому запитуємо логін та пароль користувача.**   
Отриманий логін та пароль передаємо у метод   
/api/v1/{clientId}/auth/login

Метод може повернути помилку,   
*{*

*"Code": "Authorize",*

*"CodeNumber": 4,*

*"DetailCode": "TwoFactorRequired",*

*"DetailCodeNumber": 1005,*

*"Message": "Введіть код двофакторної автентифікації",*

*"Source": "DepositSign"*

*}*  
Це означає, що у користувача активовано другий фактор. При такій помилці вікно має відобразити додаткове поле введення для коду другого фактору. Користувачу цей код прийде у мобільний додаток DepositSign  
Отриманим кодом, необхідно зашифрувати пароль та ще раз викликати метод   
/api/v1/{clientId}/auth/login  
Шифрування, та приклади є на <https://depositsign.com/swagger-api-doc/index.html> у секції   
*Public RSA key for password encryption:*

У разі вдалої авторизації, метод login поверне json структуру:  
  
*{*

*"Token": "string",*

*"KeysInfo": [*

*{*

*"KeyName": "string",*

*"IsRegistrationAdministratorKey": true,*

*"IsVerifierKey": true,*

*"CertificatesInformation": [*

*{*

*"ExpireDate": "2021-12-22T13:14:37.569Z",*

*"StartDate": "2021-12-22T13:14:37.569Z",*

*"CertificateBinary": "string",*

*"KeyUsage": "string",*

*"Thumbprint": "string",*

*"Edrpou": "string",*

*"Drfo": "string",*

*"FullName": "string",*

*"OrgUnit": "string",*

*"Org": "string",*

*"Address": "string",*

*"Phone": "string",*

*"EMail": "string",*

*"SubjectTitle": "string",*

*"Issuer": "string",*

*"IssuerCN": "string",*

*"Serial": "string",*

*"ExtKeyUsage": "string",*

*"Upn": "string",*

*"PowerCert": true,*

*"QSCD": true,*

*"CommonName": "string"*

*}*

*]*

*}*

*]*

*}*

1. Нам тут цікаві декілька речей:  
   Token – це токен, який необхідно буде передавати при виклику методів підписання та інше. Приклад як це робити, можна переглянути на <https://depositsign.com/swagger-api-doc/index.html> у секції Postman colection: - там власне посилання на колекцію постмену з прикладами.
2. Список KeysInfo –– це список ключів та опис їх сертифікатів, який потрібно відобразити користувачу для вибору (вибір для дії – скажімо підписання). Після вибору користувача нам потрібно запам'ятати KeyName, його потрібно буде передавати у потрібний метод (підписання).

# **2.** **Підписання документу**

Для накладання підпису вам потрібно визвати метод /api/v1/{clientId}/sign/file

в параметрах:

*{*

*"Password": "string", - пароль від ключа (секрет користувача)*

*"KeyName": "string", - ім'я ключа, котре обрав користувач в п.1*

*"SignType": - вибір з варіантів (1 - CADES\_BES, 4 - CADES\_T, 8 - CADES\_C, 16 - CADES\_X\_LONG, 128 - CADES\_X\_LONG\_TRUSTED) гадаю, Вам самий підходящій 16 - CADES\_X\_LONG.*

*"AppendDataTsp": true, - можете не передавати, за замовчуванням true, чи додавати електронну позначку часу від даних. Гадаю, вам потрібно це робити.*

*"FileData": "string", - base64 строка, бінарні дані файлу для підпису*

*"AppendData": true – ось цей параметр відповідає за зовнішній та внутрішній підпис. Якщо true, то внутрішня, якщо false - зовнішня*

*}*

У випадку, якщо у користувача активовано другий фактор, також потрібно запитати у користувача поточний код другого фактору, та Password зашифрувати як при login.