

Integracja programu DEVELOPER System ze stroną WWW

Dotyczy możliwości udostępniania danych przez program DEVELOPER System w celu ich integracji ze stroną internetową.

Osoba kontaktowa: Marcel Fortuna

Tel.: +48 793 404 009

e-mail: marcel@forbit.net

Historia aktualizacji:

06.11.2024

Dodano nowy parametr ShowOnlyIsExtract do skryptu GetProducts.ashx

Dodano obsługę JSON do skryptu GetProducts.ashx, GetInvestments.ashx, GetAccepts.ashx.

12.03.2024

Dodano nowy rozdział – Użytkownik API

12.12.2023

Dodano skrypt AddClientInterest.ashx

11.12.2023

Dodano skrypt AddClient.ashx

06.07.2023

Dodano nowy skrypt ResetClientPassword.ashx

29.03.2023

Poprawiono opis skryptu MakeForm.ashx.

19.12.2022

Rozszerzono skrypt GetInvestments.ashx

01.07.2021

Dodano nowy skrypt GetProductsWithPicture.ashx.

17.06.2021

Dodano nowy skrypt GetUserPicture.ashx.

18.04.2021

Dodano dodatkowe pola do skryptu AddFaultByClient.ashx.

17.03.2021

Dodano dodatkowe pola do skryptu MakeForm.ashx.

04.03.2021

Aktualizacja dokumentacji Konta klienta.

11.12.2020

Aktualizacja skryptu GetProducts.ashx, dodano możliwość pobierania dodatkowych sekcji Pictures i Attachments.

28.10.2020

Usunięto opis obsługi RAMEK.

20.08.2020

Aktualizacja skryptu GetClientAccount.ashx, dodano możliwość pobierania wartości umowy bez aneksów NIE wpływających na wartość umowy.

23.07.2020

Integracja programu DEWELOPER System ze stroną WWW

Aktualizacja skryptu GetClientAccount.ashx, dodano możliwość pobierania załączników
30.04.2020

Aktualizacja skryptu MakeForm.ashx, dodano możliwość rezerwowania produktów
05.02.2020

Aktualizacja skryptów obsługi newsletter-ów
27.01.2020

Aktualizacja obsługi skryptów RODO
16.11.2019

Rozszerzono zarządzanie opiekunami w skrypcie MakeForm.ashx.
13.11.2019

Dodano skrypt umożliwiający dodanie załączników do klientów i usterek.
08.11.2019

Dodano skrypt umożliwiający zgłoszenie usterki.

Uwaga: Informacja poufna.

Informacje dotyczące firmy FORBIT Olena Fortuna i jej działalności zawarte w niniejszym przekazie elektronicznym stanowią tajemnicę przedsiębiorstwa i przeznaczone są wyłącznie dla adresata.

Wszelkie prawa zastrzeżone © 2024 FORBIT Olena Fortuna.

Szanowni Państwo,

integrację strony WWW z programem DEWELOPER System można wykonać na kilka sposobów w zależności od zakresu integracji strony z systemem:

- prezentacja oferty handlowej (mieszkania, lokali) na stronie www,
- obsługa formularzy WWW, pobierania danych ze strony WWW do DS.,
- obsługa zgód RODO,
- obsługa konta klienta,
- obsługa newsletter-ów,
- obsługa usterek.

SPOSÓBY WYWOŁYWANIA SKRYPTÓW

Istnieją dwie metody wywoływania skryptów: metodą GET i POST.

Uwaga! Wszystkie skrypty zawierają zawsze parametr **key**. Klucz (**key**), który jest nadawany dla każdego dewelopera indywidualnie i jest niepowtarzalny. W podanych przykładach jest używany klucz do bazy demonstracyjnej DEMO. Również adres serwera jest domyślny dla bazy DEMO.

Na chwili obecnej istnieją dwa serwery do obsługi skryptów:

1. <https://www.deweloperserwer.eu> lub <http://www.deweloperserwer.eu>
2. <https://dws1.eu> lub <http://dws1.eu>

Metodą GET

Prosty przykład wywoływania skryptu metodą GET:

<https://www.dws1.eu/scripts/nazwaskryptu.ashx?key=...¶metr1¶metr2 ...>

Metodą POST

Niektóre skrypty przekazują parametry metodą POST w formie pliku XML. Wtedy parametr **key** jest przekazywany w treści pliku XML w polu `<key>...</key>`.

Np.:

```
<?xml version="1.0"?>
<root>
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <firstname>Jan</firstname>
  <lastname >Kowalski</lastname>
  <email>jam.kowalski@poczta.pl</email>
  <phone>+48 000-000-000</phone>
</root>
```

PREZENTACJA OFERTY HANDLOWEJ

Prezentacja oferty handlowej, czyli wyświetlanie listy mieszkań. Prezentacje dokonujemy za pomocą wywoływania odpowiednich skryptów, które zwracają pliki XML z zawartością bazy lub załączniki w formie plików PDF, JPEG itd.

Do obsługi oferty handlowej służą trzy skrypty:

GetMapPicture.ashx – służy do obsługi rysunków mapowanych za pomocą pliku XML

GetProducts.ashx – zwraca listę produktów w formie pliku XML

ShowProduct.ashx – zwraca załączniki dla pojedynczego produktu: plan lokalu PDF, Rzut lokalu w formie JPEG itd.

Przykład strony używająca skrypty do prezentacji oferty handlowej:

<https://www.bardowskiego1.pl/mieszkania/>

Poniżej idzie opis każdego skryptu i sposób wywoływania z przykładami.

Skrypt GetMapPicture.ashx

Wywołanie skryptu, który zwraca plik XML z rysunkiem oraz mapą odwołań do pozostałych rysunków:

<https://www.dws1.eu/scripts/getmappicture.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Skrypt wywołujemy metodą GET.

Parametry: (oprócz key)

OwnerKind – rodzaj produktu, którego dotyczy rysunek;

ID_Owner - identyfikator produktu;

Code – kod rysunku z mapy;

Wartości dla pola **OwnerKind**

1	Firma - wartość domyślna
14	Inwestycja
15	Etap
16	Produkt (lokal mieszkalny, dom, itd.)

Jeżeli nie prześlemy żadnego parametru, skrypt zwróci rysunek zaznaczony jako domyślny dla firmy o **ID_Owner** = 1. Jeżeli znamy ID inwestycji, możemy uzyskać domyślny rysunek poprzez wywołanie komendy np.:

https://www.dws1.eu/scripts/getmappicture.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&OwnerKind=14&ID_Owner=38

Skrypt zwraca plik XML, który zawiera rysunek oraz kolekcję obszarów mapowanych. Pole <Data> zawiera zawartość pliku graficznego w formie Base64. Sekcja <Map> zawiera kolekcję regionów (obszarów) do przejścia na kolejny rysunek.

Wartości poniższych pól używamy do wywoływania skryptu GetMapPicture.ashx w celu przejścia na kolejny rysunek:

<Link_Kind> - odpowiada parametrowi **OwnerKind**.

<Link_ID> - odpowiada parametrowi **ID_Owner**.

<Link_Code> - odpowiada parametrowi **Code**.

Jeżeli zwrócony plik XML w polu <Link_Kind> zawiera wartość **16** to znaczy, że obszar wskazuje na konkretny produkt – np. lokal mieszkalny. Po kliknięciu na ten obszar pokazujemy kartę lokalu za pomocą skryptu **ShowProduct.ashx** z parametrem **ID_Product=Link_ID**.

Oto przykład zwracanego pliku.

```
<?xml version="1.0"?>
<root>
  <Error>0</Error>
  <ID_Picture>1</ID_Picture>
  <FileName>a.jpg</FileName>
  <Data>Plik graficzny zakodowany w formacie - Base64 </Data>
  <Map>
    <Regions>
      <Region>
        <Link_Kind>14</Link_Kind>
        <Link_ID>38</Link_ID>
        <Link_Code/>
        <Link_Title>InwestycjaYC</Link_Title>
        <Points>
          <Point>
            <x>161</x>
            <y>71</y>
          </Point>
          <Point>
            <x>359</x>
            <y>27</y>
          </Point>
          <Point>
            <x>727</x>
            <y>203</y>
          </Point>
          <Point>
            <x>723</x>
            <y>240</y>
          </Point>
          <Point>
            <x>363</x>
            <y>116</y>
          </Point>
          <Point>
            <x>161</x>
            <y>138</y>
          </Point>
        </Points>
      </Region>
      <Region>
        ...
      </Region>
    </Regions>
  </Map>
</root>
```

Skrypt GetProducts.ashx

Wywołanie skryptu, który zwraca plik XML z listą produktów (lokale, domy itd.):

<https://www.dws1.eu/scripts/getproducts.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Skrypt może zawierać parametry, które ograniczają zawartość bazy. Skrypt wywołujemy metodą GET i POST.

W przypadku metody POST parametry przekazujemy w formie pliku XML. Na przykład:

```
<?xml version="1.0"?>
<root>
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>
  <id_investment>1</id_investment>
  <format>json</format>
</root>
```

Parametry: (oprócz key)

ID_Investment - identyfikator inwestycji

ID_Stage - identyfikator etapu

ID_Product - identyfikator produktu

ID_ProductKind – rodzaj produktu

ID_AttachmentKind – rodzaj załącznika (stara wersja, patrz sekcji **Attachments** i **Pictures**)

ExtraData – czy mają zostać pobrane również „Ekstra Dane” (0/1)

ShowAll– czy mają zostać pobrane również produkty które nie mają włączoną opcje „Pokaż na WWW” (0/1). Opcja dotyczy również inwestycji i etapów w których znajdują się produkty.

ShowOnlyIsExtract– czy mają zostać pobrane tylko produkty które nie mają włączoną opcje „Wyodrębnij na WWW” (0/1).

Format – format zwracanego pliku (XML, JSON). Domyślnie format to plik XML.

Wartości dla **ID_ProductKind**:

- 1 Lokal mieszkalny,
- 2 Dom,
- 3 Lokal usługowy,
- 4 Lokal biurowy;

Wartości dla **ID_AttachmentKind**:

- 1 Wzór umowy
- 2 Standard robót
- 3 Plan lokalu (Format PDF)
- 4 Oferta handlowa
- 5 Opis inwestycji (Format PDF)

Parametr: **ID_AttachmentKind** - został aby zachować kompatybilność ze starszymi wywoływanymi. W nowych wersjach plik XML zawiera dodatkowe sekcji: **Attachments** i **Pictures**.

Na przykład:

Integracja programu DEWELOPER System ze stroną WWW

https://www.dws1.eu/scripts/getproducts.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Investment=1

- wywoła listę produktów dla inwestycji o identyfikatorze 1.

Oto wynik wywołania skryptu z opisem każdego pola:

```
<root>
  <Error>0</Error>
  <Products>
    <Product>
      <ID_Product>14</ID_Product>
      <ID_ProductKind>1</ID_ProductKind>
      <ProductsKindTitle>Lokal mieszkalny</ProductsKindTitle>
      <ID_ProductStatus>3</ID_ProductStatus>
      <ID_Investment>1</ID_Investment>
      <InvestmentTitle>Inwestycja Słonecznik</InvestmentTitle>
      <ID_Stage>1</ID_Stage>
      <StageTitle>Budynek A</StageTitle>
      <Number>IA-A-LM-04</Number>
      <Title>Lokal mieszkalny - 2 pokoje z kuchnią</Title>
      <TitleSupplement>Garaż, (Balkon)</TitleSupplement>
      <Stairway>A</Stairway>
      <Flight>1</Flight>
      <Floor>Parter</Floor>
      <Floors>1</Floors>
      <Rooms>2</Rooms>
      <Kitchen>0</Kitchen>
      <w_IsShow>1</w_IsShow>
      <IsNorth>1</IsNorth>
      <IsEast>1</IsEast>
      <IsSouth>0</IsSouth>
      <IsWest>0</IsWest>
      <w_IsShowPrice>1</w_IsShowPrice>
      <DateStatus>2012-01-11T11:44:57</DateStatus>
      <ProjectNumber />
      <Vertical />
      <FloorNumber>0</FloorNumber>
      <BudgetNetto>0.0000</BudgetNetto>
      <BudgetBrutto>0.0000</BudgetBrutto>
      <BudgetArea>0.0000000000000000e+000</BudgetArea>
      <ExtraData />
      <Area>5.5000000000000000e+001</Area>
      <RealArea>5.5000000000000000e+001</RealArea>
      <TotalSupplementArea>4.0000000000000000e+000</TotalSupplementArea>
      <UnitPriceNetto>6944.4400</UnitPriceNetto>
      <UnitPriceBrutto>7500.0000</UnitPriceBrutto>
      <Netto>381944.4400</Netto>
      <Vat>8</Vat>
      <Brutto>412500.0000</Brutto>
      <TotalNetto>381944.4400</TotalNetto>
      <TotalBrutto>412500.0000</TotalBrutto>
      <TotalOfferNetto>406944.4400</TotalOfferNetto>
      <TotalOfferBrutto>443250.0000</TotalOfferBrutto>
      <ID_Contract>6</ID_Contract>
      <ID_ContractStatus>3</ID_ContractStatus>
      <ConArea>5.5000000000000000e+001</ConArea>
      <ConRealArea>5.5000000000000000e+001</ConRealArea>
      <ConTotalSupplementArea>4.0000000000000000e+000</ConTotalSupplementArea>
      <ConUnitPriceNetto>6481.4800</ConUnitPriceNetto>
      <ConUnitPriceBrutto>7000.0000</ConUnitPriceBrutto>
      <ConNetto>356481.4800</ConNetto>
      <ConVat>8</ConVat>
      <ConBrutto>385000.0000</ConBrutto>
      <ConTotalNetto>356481.4800</ConTotalNetto>
      <ConTotalBrutto>385000.0000</ConTotalBrutto>
      <ConTotalOfferNetto>356481.4800</ConTotalOfferNetto>
      <ConTotalOfferBrutto>385000.0000</ConTotalOfferBrutto>
```

```
<ID_Attachment>7</ID_Attachment>
<ProductRooms>
  <Room>
    <ID_ProductRoom>97</ID_ProductRoom>
    <ID_RoomKind>4</ID_RoomKind>
    <ID_Product>14</ID_Product>
    <Number>1</Number>
    <Title>Hall</Title>
    <Area>4.000000000000000e+000</Area>
    <WithWindow>0</WithWindow>
    <RoomLevel>1</RoomLevel>
  </Room>
  <Room>
    <ID_ProductRoom>98</ID_ProductRoom>
    <ID_RoomKind>1</ID_RoomKind>
    <ID_Product>14</ID_Product>
    <Number>2</Number>
    <Title>Salon</Title>
    <Area>2.100000000000000e+001</Area>
    <WithWindow>1</WithWindow>
    <RoomLevel>1</RoomLevel>
  </Room>
</ProductRooms>
<ProductSupplements>
  <Supplement>
    <ProductsKindTitle>Balkon</ProductsKindTitle>
    <Number />
    <Title>Balkon</Title>
    <Area>4.000000000000000e+000</Area>
    <RealArea>4.000000000000000e+000</RealArea>
    <UnitPriceNetto>0.0000</UnitPriceNetto>
    <UnitPriceBrutto>0.0000</UnitPriceBrutto>
    <Netto>0.0000</Netto>
    <Vat>8</Vat>
    <Brutto>0.0000</Brutto>
    <Floor>Parter</Floor>
    <Flight>0</Flight>
    <FloorNumber>0</FloorNumber>
  </Supplement>
</ProductSupplements>
<Pictures>
  <Picture>
    <ID_Picture>91</ID_Picture>
    <ID_PictureKind>4</ID_PictureKind>
  </Picture>
  <Picture>
    <ID_Picture>92</ID_Picture>
    <ID_PictureKind>5</ID_PictureKind>
  </Picture>
</Pictures>
</Product>
</Products>
</root>
```

Opis pól:

Dane podstawowe:

<ID_Product>	identyfikator produktu
<ID_ProductKind>	rodzaj produktu
<ProductsKindTitle>	nazwa rodzaju produktu
<ID_ProductStatus>	status produktu (1 wolny, 2 rezerwacja, 3 sprzedany)
<DateStatus>	data statusu np. kiedy był rezerwowany lub sprzedany

<ID_Investment>	identyfikator inwestycji, do której należy produkt
<InvestmentTitle>	nazwa inwestycji
<ID_Stage>	identyfikator etapu (budynku)
<StageTitle>	nazwa etapu (budynku)
<Number>	numer produktu
<ProjectNumber>	numer projektowy
<Title>	nazwa produktu
<TitleSupplement>	nazwa dodatków (balkony, tarazy itd.)

Lokalizacja produktu:

<Stairway>	numer klatki schodowej
<Flight>	kondygnacja
<Floor>	piętro
<FloorNumber>	numer piętra
<Floors>	ilość pięter (poziomów) np. dla domków
<Rooms>	ilość pokoi
<Kitchen>	czy zawiera kuchnię (1 tak, 0 aneks kuchenny)
<Vertical>	pion w którym znajduje się produkt (opcjonalnie)

<w_IsShow>	czy pokazywać na stronie www (1 tak, 0 nie)
<w_IsShowPrice>	czy pokazywać cenę produktu na stronie www
<w_IsExtract>	czy wyodrębnić na stronie www (np. Promocja)

<IsNorth>	strony świata (1 zawiera tą stronę, 0 nie)
<IsEast>	
<IsSouth>	
<IsWest>	

Wartości i powierzchnie produktu wg budżetu:

<BudgetNetto>	wartość budżetowa netto
<BudgetBrutto>	wartość budżetowa brutto
<BudgetArea>	powierzchnia budżetowa

Wartości i powierzchnie produktu:

<Area>	powierzchnia
<RealArea>	powierzchnia rzeczywista
<TotalSupplementArea>	powierzchnia wszystkich dodatków

<UnitPriceNetto>	cena jednostkowa netto
<UnitPriceBrutto>	cena jednostkowa brutto
<Netto>	wartość netto (samego produktu)
<Vat>	stawka VAT
<Brutto>	wartość brutto (samego produktu)
<TotalNetto>	wartość netto z dodatkami
<TotalBrutto>	wartość brutto z dodatkami
<TotalOfferNetto>	wartość netto oferty np. mieszkanie z garażem i komórką
<TotalOfferBrutto>	wartość brutto oferty

Wartości i powierzchnie produktu wg umowy:

<ID_Contract>	identyfikator umowy
<ID_ContractStatus>	status umowy. Oto wartości: 1 Wprowadzona 2 Rezerwacyjna 3 Przedwstępna 4 Sprzedaży 5 Anulowana 6 Rozliczona 7 Deweloperska
<ConArea>	powierzchnia wg umowy
<ConRealArea>	powierzchnia rzeczywista wg umowy
<ConTotalSupplementArea>	powierzchnia wszystkich dodatków wg umowy
<ConUnitPriceNetto>	cena jednostkowa netto wg umowy

<ConUnitPriceBrutto>	cena jednostkowa bruto wg umowy
<ConNetto>	wartość netto (samego produktu) wg umowy
<ConVat>	stawka VAT wg. umowy
<ConBrutto>	wartość brutto (samego produktu) wg umowy
<ConTotalNetto>	wartość netto z dodatkami wg umowy
<ConTotalBrutto>	wartość brutto z dodatkami wg umowy
<ConTotalOfferNetto>	wartość netto oferty wg umowy
<ConTotalOfferBrutto>	wartość brutto oferty wg umowy

Dodatkowe informacje:

<ID_Attachment>	identyfikator pierwszego załącznika
<ExtraData />	ekstra dane

Sekcja <ProductRooms>

zawiera informacje dotyczące każdego pokoju:

```
<ProductRooms>
  <Room>
    <ID_ProductRoom>      identyfikator pokoju
    <ID_RoomKind>         rodzaj pokoju (pokój, komunikacja, łazienka itd.)
    <ID_Product>         identyfikator produktu, do którego należy pokój
    <Number>              numer
    <Title>               nazwa
    <Area>                powierzchnia
    <WithWindow>         czy zawiera okno (1 tak, 0 nie)
    <RoomLevel>          numer poziomu, na którym znajduje się pokój
  </Room>
</ProductRooms>
```

Sekcja < ProductSupplements >

zawiera informacje dotyczące każdego dodatku:

```
<ProductSupplements>
  <Supplement>
    <ProductsKindTitle>  nazwa rodzaju
    <Number>              numer
    <Title>               nazwa
    <Area>                powierzchnia
    <RealArea>           powierzchnia rzeczywista
    <UnitPriceNetto>     cena jednostkowa netto
    <UnitPriceBrutto>    cena jednostkowa brutto
    <Netto>               wartość netto
    <Vat>                stawka VAT
    <Brutto>             wartość brutto
    <Floor>              nazwa piętra
    <Flight>             kondygnacja
    <FloorNumber>        numer pietra
  </Supplement>
</ProductSupplements>
```

Sekcja < Pictures >

zawiera informacje dotyczące każdego rysunku podpisanego pod produkt:

```
<Pictures>
  <Picture>
    <ID_Picture>91</ID_Picture>
    <ID_PictureKind>4</ID_PictureKind>
  </Picture>
  <Picture>
    <ID_Picture>92</ID_Picture>
    <ID_PictureKind>5</ID_PictureKind>
  </Picture>
```

```
</Pictures>
```

Sekcja < Attachments >

zawiera informacje dotyczące każdego rysunku podpiętego pod produkt:

```
<Attachments>
```

```
  <Attachment>
```

```
    <ID_Attachment>91</ID_Attachment>
```

```
    <ID_AttachmentKind>4</ID_AttachmentKind>
```

```
  </Attachment>
```

```
  <Attachment>
```

```
    <ID_Attachment>92</ID_Attachment>
```

```
    <ID_AttachmentKind>5</ID_AttachmentKind>
```

```
  </Attachment>
```

```
</Attachments>
```

Skrypt GetProductsWithPicture.ashx

Wywołanie skryptu, który zwraca plik XML z listą produktów wraz z rysunkami (lokale, domy itd.):

<https://www.dws1.eu/scripts/getproductswithpicture.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Skrypt może zawierać parametry, które ograniczają zawartość bazy. Skrypt wywołujemy metodą GET.

Wynik skryptu jest identyczny jak dla GetProducts.ashx i ma dodatkową sekcję `Pictures` dla każdego produktu:

```
<Pictures>
  <Picture>
    <ID_Picture>16</ID_Picture>
    <ID_PictureKind>4</ID_PictureKind>
    <Picture>Qk3mbgsAA....</Picture>
  </Picture>
</Pictures>
```

Zwracany obraz znajduje się pod kluczem `<Picture>Qk3mbgsAA....</Picture>` w formacie Base64.

Parametry: (oprócz key)

`ID_Investment` - identyfikator inwestycji

`ID_Stage` - identyfikator etapu

`ID_Product` - identyfikator produktu

`ID_ProductKind` – rodzaj produktu

`ExtraData` – czy mają zostać pobrane również „Ekstra Dane” (0/1)

`ShowAll`– czy mają zostać pobrane również produkty które nie mają włączoną opcje „Pokaż na WWW” (0/1). Opcja dotyczy również inwestycji i etapów w których znajdują się produkty.

`ID_PictureKind` – rodzaj rysunku

Wartości dla `ID_ProductKind`:

- 1 Lokal mieszkalny,
- 2 Dom,
- 3 Lokal usługowy,
- 4 Lokal biurowy;

Wartości dla `ID_PictureKind`:

- 1 Ikona
- 2 wizualizacja 1
- 3 wizualizacja 2
- 4 Plan lokalu
- 5 Umiejscowienie
- 6 Mapa 1
- 7 Mapa 2
- 8 Mapa 3
- 12 Ogłoszenia
- 14 Logo
- 15 Logo dodatkowe
- 16 Logo do faktury

- 17 Plan osiedla
- 18 Karta produktu
- 19 WWW Informacje
- 20 WWW Informacje Ikona
- 21 Umieszczenie w budynku

Jest to lista domyślnych rysunków. Lecz każdy deweloper może posiadać dodatkowe rysunki o unikalnych numerach ID.

Na przykład:

https://www.dws1.eu/scripts/getproducts.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Investment=1

- wywoła listą produktów dla inwestycji o identyfikatorze 1.

Skrypt GetInvestments.ashx

Wywołanie skryptu, który zwraca plik XML z listą inwestycji:

<https://www.dws1.eu/scripts/getinvestments.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Skrypt może zawierać parametry, które ograniczają zawartość bazy. Skrypt wywołujemy metodą GET.

Parametry: (oprócz key)

ID_Investment - identyfikator inwestycji. Jeżeli brak - zostanie zwrócona lista wszystkich inwestycji.

ExtraData – czy mają zostać pobrane również „Ekstra Dane” (0/1)

Format – format zwracanego pliku (XML, JSON). Domyślnie format to plik XML.

Oto wynik wywołania skryptu z opisem każdego pola:

```
<?xml version="1.0"?>
<root>
  <Error>0</Error>
  <Investments>
    <Investment>
      <ID_Investment>1</ID_Investment>
      <ID_Company>1</ID_Company>
      <ID_InvestmentRegion>1</ID_InvestmentRegion>
      <Title>Inwestycja Słonecznik</Title>
      <Short>V9</Short>
      <OrderWWW>0</OrderWWW>
      <DateStart>2014-08-01T00:00:00</DateStart>
      <DateEnd>2015-02-28T00:00:00</DateEnd>
      <IsArchival>0</IsArchival>
      <w_IsShow>1</w_IsShow>
      <RegionTitle>Warszawa</RegionTitle>

      <AddressToInvestment>
        <Address>
          <ID_Address>2</ID_Address>
          <ID_AddressKind>6</ID_AddressKind>
          <Street>ul. Szeroka</Street>
          <House>5</House>
          <Door/>
          <POB/>
          <PIN>60-123</PIN>
          <City>Łódź</City>
          <State>wielkopolskie</State>
          <Country>Polska</Country>
          <AddressKind>Lokalizacji</AddressKind>
        </Address>
      </Investment>
    </Investments>
  </root>
```

Aktualizacja 19-12-2022:

Do ww. skryptu dodano oprócz danych inwestycji dane etapów dla każdej inwestycji. Do pliku XML dodana nowa sekcja <StagesToInvestment>. Oto przykład:

```
<StagesToInvestment>
  <Stage>
    <ID_Stage>1</ID_Stage>
    <ID_Investment>1</ID_Investment>
    <Title>Osiedle Promienna I</Title>
    <Short>PI</Short>
    <StageKind>Budynek mieszkalny</StageKind>
    <DateStart>2015-06-05T00:00:00</DateStart>
    <DateEnd>2016-08-31T00:00:00</DateEnd>
    <w_IsShow>0</w_IsShow>
    <IsOpened>0</IsOpened>
    <ExtraData><?xml version="1.0" encoding="windows-1250"?>
      <Values></Values> </ExtraData>
  </Stage>
</StagesToInvestment>
```

Skrypt ShowProduct.ashx

Kolejny skrypt zwraca załączniki dla pojedynczego produktu: plan lokalu PDF, Rzut lokalu w formie JPEG itd. Skrypt wywołujemy metodą GET.

Parametry: (oprócz key)

ID_Product – identyfikator produktu
FileKind – rodzaj pliku (załącznik lub rysunek)
FileType – rodzaj załącznika lub rysunku
Width – szerokość dla rysunku
Height – wysokość dla rysunku

Na przykład:

https://www.dws1.eu/scripts/showproduct.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Product=43

- zwraca plik PDF dla produktu o ID=17

Skrypt **ShowProduct** może zwrócić i inne załączniki przypisane do produktu w zależności od podanych parametrów.

Wartości dla **FileKind**:

1	Zwróć załącznik - wartość domyślna
2	Zwróć rysunek

Wartości dla **FileType**:

Dla załączników:

1	Wzór umowy
2	Standard robót
3	Plan lokalu (Format PDF) - wartość domyślna
4	Oferta handlowa

Dla rysunków:

1	Ikona
2	wizualizacja 1
3	wizualizacja 2
4	Plan lokalu
5	Umiejscowienie
6	Mapa 1
7	Mapa 2
8	Mapa 3
12	Ogłoszenia
14	Logo
15	Logo dodatkowe
16	Logo do faktury
17	Plan osiedla
18	Karta produktu
19	WWW Informacje
20	WWW Informacje Ikona
21	Umiejscowienie w budynku

Na przykład:

https://www.dws1.eu/scripts/showproduct.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Product=43&FileKind=2&FileType=4

- zwraca rysunek JPEG (plan lokalu) dla produktu o ID=17

Przy pobraniu rysunków można określić rozmiar w pikselach zwracanego rysunku za pomocą parametrów **Width** i **Height**.

Na przykład:

https://www.dws1.eu/scripts/showproduct.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Product=43&FileKind=2&FileType=4&Width=200&Height=600

- zwraca rysunek JPEG (plan lokalu) dla produktu o ID=17 o wymiarach 400x600px.

INTEGRACJA FORMULARZA

Istnieje możliwość przekazywania danych z formularzy ze strony www do bazy danych programu DEWELOPER system. Dane są przekazywane za pomocą skryptu metodą POST po przekształceniu do formatu XML.

Skrypt MakeForm.ashx

Skrypt wykonuje kilka czynności w zależności od treści przekazywanego pliku XML. Podstawowa czynność to sprawdzenie, czy klient istnieje w bazie dewelopera po adresie **email** i jeżeli nie dodanie go do bazy. Kolejną czynnością skryptu do tworzenie dokument typu „**Zainteresowanie klienta**” lub „**Rezerwacja**” po stronie bazy DEWELOPER System. I na koniec rejestruje zgody **RODO** zaznaczonych przez klienta.

Skrypt wywołujemy metodą POST.

Oto przykład pliku z danymi:

```
<?xml version="1.0"?>
<root>
  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Zainteresowanie czy rezerwacja -->
  <editmode>1</editmode>

  <!-- Dane klienta -->
  <firstname>Jan</firstname>
  <lastname >Kowalski</lastname>
  <email>jam.kowalski@poczta.pl</email>
  <phone>+48 000-000-000</phone>

  <!-- Dane opiekuna →
  <id_keeper>1,2,3</id_keeper>

  <!-- Dane inwestycji oraz produktu -->
  <id_investment>1</id_investment>
  <id_product>1,2,3</id_product>
  <id_interestsource>1</id_interestsource>
  <sourcedescription>opis</sourcedescription >

  <!-- Dane dotyczące rezerwacji -->
  <cost>100</cost>
  <days>3</days>

  <!-- Wiadomość -->
  <subject>Temat wiadomości ...</subject >
  <message>Tekst wpisany przez klienta ...</message>

  <!-- ZgodyRODO
  <accepts>1,2,3...</accepts >
</root>
```

POLE OBOWIĄZKOWE:

email adres e-mail.

POZOSTAŁE POLA DLA KLIENTA (opcjonalne):

firstname	imię
lastname	nazwisko
secondname	drugie imię
iscompany	czy klient jest firmą (0-nie/1-tak)
companytitle	nazwa firmy
nip	NIP firmy
regon	regon firmy
pesel	pesel
krs	KRS firmy
phone	numer telefonu
description	opis do klienta
ylprofile	profil YouLead, dodaje się do opisu (<i>description</i>)
id_source	źródło pochodzenia klienta (dostępne z poziomu programu DS.) Menu Opcje -> Słowniki -> Źródło pochodzenia klientów. np.: RynekPirwotny, Agencja, Obido itd. Jednym słowem skąd pochodzi klient.
id_category	kategoria klienta (dostępne z poziomu programu DS.)
id_keeper	identyfikator użytkownika programu DS. – opiekun.

Na podstawie adresu e-mail system weryfikuję czy dany klient, jest już w bazie DEWELOPER System. Jeżeli go nie ma, tworzony jest nowy klient z flagą „Klient niezweryfikowany” – co daje możliwość łatwej filtracji nowych klientów.

Pole *id_source* jest używane w przypadku, kiedy deweloper chce wskazać źródła pochodzenia klientów z różnych formularzy. Źródła muszą zostać dodane do programu DEWELOPER System przed wywołaniem formularza. Menu Opcje -> Słowniki -> Źródło pochodzenia klientów. Po dodaniu zestawu źródeł, identyfikatory zdefiniowanych źródeł trzeba przekazać osobom, które opracowują formularz na stronie www.

Pole *ylprofile* jest używane w przypadku, kiedy deweloper współpracuje z systemem YouLead. W tym przypadku do opisu klienta można przypisać jego profil w systemie YouLead.

ZARZĄDZANIE OPIEKUNEM (opcjonalne):

Opiekuna klienta i zainteresowania można ustawić na dwa sposoby. Pierwsza metoda to przekazać go bezpośrednio w parametrze *id_keeper*. Druga możliwość jest przyporządkowanie użytkowników do inwestycji bezpośrednio w programie DEWELOPER System. Przyporządkowanie dokonujemy w menu Widok -> Dział WWW -> panel Inwestycje. Jeżeli nie przekazemy parametr *id_keeper* wówczas skrypt przyporządkuje automatycznie kolejnego użytkownika z okna Dział WWW -> panel Inwestycje.

Uwaga! Aby druga metoda zadziałała niezbędne jest przekazanie parametru *id_investment*, aby skrypt mógł znaleźć odpowiednich opiekunów powiązanych z inwestycją *id_investment*.

Opiekunami można również zarządzać samemu. Skrypt *GetKeepersForInvesment.ashx*. zwraca listę opiekunów dla wybrane inwestycji *id_investment*.

Np.: https://www.dws1.eu/scripts/GetKeepersForInvesment.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&id_investment=1

ZAINTERESOWANIE CZY REZERACJA

O tym czy będzie stworzona rezerwacja czy zainteresowanie produktem decyduje flaga `editmode`.

Oto wartości dla `editmode`:

1 – zostanie stworzone zainteresowanie produktem (wartość domyślna).

2 – zostanie stworzona rezerwacja dla produktu `id_product`. W tym przypadku pole `id_product` jest obowiązkowe.

ZAINTERESOWANIE PRODUKTEM (opcjonalne):

Jeżeli plik XML zawiera dodatkowe pola dotyczące produktu lub inwestycji, wówczas skrypt tworzy dodatkowy dokument: „Zainteresowanie klienta” po stronie bazy Deweloper System i zapisuje ww. dane.

`id_investment` identyfikator inwestycji
`id_product` identyfikator produktu lub identyfikatory dla wielu produktów, jako separator używamy znak przecinka: -,-. Np. 120,130,146

`id_interestkind` identyfikator rodzaju kontaktu (znajduje się w bazie DS.)
`id_interestsource` identyfikator źródła zainteresowania (znajduje się w bazie DS.)
Menu Opcje -> Słowniki -> Opcje zainteresowań.
Np.: RynekPirwotny.pl, Obido.pl itd.
Jednym słowem skąd pochodzi zainteresowanie.

`sourcedescription` opis źródła zainteresowania. **Uwaga – pole nie może zawierać znaki Enter lub Tab (/n lub /r lub /t).**

`subject` temat -pole tekstowe na formularzu wypełniane przez klienta
`message` opis - pole tekstowe na formularzu wypełniane przez klienta

`area` pole tekstowe na formularzu wypełniane przez klienta
`rooms` pole tekstowe na formularzu wypełniane przez klienta

wartości dla pola `area` określa jakim metrażem jest zainteresowany klient. Przykładowe wartości dla pola `area`: = 50, > 30, < 90, 30 ... 60 lub 30 – 60.

wartości dla pola `rooms` określa jaką ilość pokoi jest zainteresowany klient. Przykładowe wartości dla pola `rooms`: = 2, > 2, < 5, 3 ... 5 lub 3 – 5.

Zamiast pól `area` i `rooms` można użyć po dwa pola dla każdej wartości typu INTEGER:

`areafrom` pole określa wartość metrażu od. Np. 60
`areato` pole określa wartość metrażu do. Np. 83
`roomsfrom` pole określa wartość ilości pokoje od. Np. 2
`roomsto` pole określa wartość ilości pokoje do. Np. 4
`datecreated` data dodania zainteresowania

Jeżeli jedno z pól (`id_investment`, `id_product`) zawiera wartość większą od zera, wtedy następuje dodanie nowego zainteresowania produktem do klienta w bazie DEWELOPER System. Wartości tych pól można uzyskać za pomocą skryptu `GetInvestments.ashx` i `GetProducts.ashx`.

Np.: <https://www.dws1.eu/scripts/GetInvestments.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Wartość pola `id_interestskind` (rodzaj kontaktu podczas tworzenia zainteresowania (telefon, wizyta, mail itd.) ?) można uzyskać za pomocą skryptu `GetInterestKind.ashx`. Dla stałych przypadków, identyfikator może zostać podany przez użytkownika programu DEWELOPER System. Przykładowy skrypt, który zwraca listę identyfikatorów źródła można wywołać pod adresem:

<https://www.dws1.eu/scripts/GetInterestKind.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Wartość pola `id_interestsource` (skąd się dowiedziałeś o naszej ofercie ?) można uzyskać za pomocą skryptu `GetInterestSource.ashx`. Dla stałych przypadków, identyfikator może zostać podany przez użytkownika programu DEWELOPER System. Przykładowy skrypt, który zwraca listę identyfikatorów źródła można wywołać pod adresem:

<https://www.dws1.eu/scripts/GetInterestSource.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

REZERWACJA PRODUKTU (opcjonalne):

Jeżeli plik XML zawiera pole `editmode` o wartości 2 wówczas zamiast zainteresowania zostanie stworzona rezerwacja produktu o identyfikatorze `id_product`. Domyślna wartość pola jest `editmode 1`, co oznacza zainteresowanie.

UWAGA !

Pole `id_product` jest obowiązkowe i nie może zawierać identyfikatory dla wielu produktów lecz tylko dla jednego. Ze względu na możliwość kolejkowania, rezerwacja zawsze dotyczy tylko jednego produktu.

Pozostałe pola:

<code>cost</code>	jest to koszt rezerwacji - wartość netto. Wysokość VAT zostanie pobrana z produktu który będzie rezerwowany.
<code>days</code>	długość rezerwacji w dniach. Początek rezerwacji ustawiany jest na dzień dokonania rezerwacji lub jeżeli produkt jest już zarezerwowany, to na koniec okresu ostatniej rezerwacji.

Oto przykład :

```
<editmode>2</editmode>
<id_product>123</id_product>
<cost>100</cost>
<days>3</days>
```

ZGODY RODO:

accepts identyfikatory zgód rozdzielone przecinkiem. Identyfikatory można uzyskać za pomocą skryptu [GetAccepts.ashx](#).

W świetle nowych przepisów, deweloper jest zobowiązany do rejestrowania zgód klienta pod kątem rodzaju i zakresu przetwarzanych danych osobowych. Zgody te mogą być wyświetlane pod formularzem kontaktowym na stronie internetowej. Aby pobrać listę dostępnych zgód z bazy dewelopera i pokazać je pod formularzem na stronie internetowej używamy skryptu: [GetAccepts.ashx](#). Zwraca on listę zgód z treścią i identyfikatorami. Przykładowy skrypt który zwraca listę zgód można wywołać pod adresem:

<https://www.dws1.eu/scripts/GetAccepts.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Uwaga!

Skrypt [MakeForm.ashx](#) nie edytuje wartości związanych ze zgodami, tylko dodaje nowe zgody zaznaczone przez klienta.

Trudno wymagać od klienta, aby zawsze zaznaczał wszystkie poprzednio wyrażone przez niego zgody. Jeżeli klient chce anulować Swoje zgody, musi się zgłosić do administratora danych z prośbą o ich anulowanie. Istnieje skrypt do edycji zgód, ale szerzej zostanie omówimy w dalszej dokumentacji.

Dodatkowo istnieje funkcja (opcja do wykupienia), która pozwala po wypełnieniu przez klienta formularza kontaktowego, wysłać automatycznie zwrotnego emaila z informacją o otrzymaniu przez dewelopera danych kontaktowych i najbliższym kontakcie. Do wysłania automatycznego emaila można użyć skrzynki pocztowej dewelopera lub w przypadku jej braku można użyć skrzynkę serwisową firmy FORBIT.

Poniżej znajduje się dwa linki z przykładowym kodem w języku C# ASP.NET oraz JavaScript.

Kod z przykładem można pobrać pod tym linkiem:

www.deweloperserwer.eu/scripts/doc/example1.txt

www.deweloperserwer.eu/scripts/doc/example2.txt

Skrypt [AddClient.ashx](#)

Skrypt dodaje nowego klienta do bazy danych. Podstawowa czynność to sprawdzenie, czy klient istnieje w bazie dewelopera po adresie **email** i jeżeli nie dodanie go do bazy.

Skrypt wywołujemy metodą POST.

Oto przykład pliku z danymi:

```
<?xml version="1.0"?>
<root>
  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Dane klienta -->
  <firstname>Jan</firstname>
  <lastname >Kowalski</lastname>
  <email>jam.kowalski@poczta.pl</email>
  <phone>+48 000-000-000</phone>
  <id_keeper>1</id_keeper>
```

</root>

POLE OBOWIĄZKOWE:

email adres e-mail.

POZOSTAŁE POLA DLA KLIENTA (opcjonalne):

firstname	imię
lastname	nazwisko
secondname	drugie imię
iscompany	czy klient jest firmą (0-nie/1-tak)
companytitle	nazwa firmy
nip	NIP firmy
regon	regon firmy
pesel	pesel
krs	KRS firmy
phone	numer telefonu
description	opis do klienta
ylprofile	profil YouLead, dodaje się do opisu (description)
id_source	źródło pochodzenia klienta (dostępne z poziomu programu DS.) Menu Opcje -> Słowniki -> Źródło pochodzenia klientów. np.: RynekPirwotny, Agencja, Obido itd. Jednym słowem skąd pochodzi klient.
id_category	kategoria klienta (dostępne z poziomu programu DS.)
id_keeper	identyfikator użytkownika programu DS. – opiekun klienta.
datefirstcontact	data pierwszego kontaktu z klientem.
id_investment	identyfikator inwestycji, aby można było określić opiekuna w przypadku gdy go nie przekazujemy bezpośrednio w polu id_keeper .

Na podstawie adresu e-mail system weryfikuję czy dany klient, jest już w bazie DEWELOPER System. Jeżeli go nie ma, tworzony jest nowy klient z flagą „Klient niezweryfikowany” – co daje możliwość łatwej filtracji nowych klientów.

Pole **id_source** jest używane w przypadku, kiedy deweloper chce wskazać źródła pochodzenia klientów z różnych formularzy. Źródła muszą zostać dodane do programu DEWELOPER System przed wywołaniem formularza. Menu Opcje -> Słowniki -> Źródło pochodzenia klientów. Po dodaniu zestawu źródeł, identyfikatory zdefiniowanych źródeł trzeba przekazać osobom, które opracowują formularz na stronie www.

Pole **ylprofile** jest używane w przypadku, kiedy deweloper współpracuje z systemem YouLead. W tym przypadku do opisu klienta można przypisać jego profil w systemie YouLead.

ZARZĄDZANIE OPIEKUNEM (opcjonalne):

Opiekuna klienta i zainteresowania można ustawić na dwa sposoby. Pierwsza metoda to przekazać go bezpośrednio w parametrze **id_keeper**. Druga możliwość jest przyporządkowanie użytkowników do inwestycji bezpośrednio w programie DEWELOPER System. Przyporządkowanie dokonujemy w menu Widok -> Dział WWW -> panel Inwestycje. Jeżeli nie prześlemy parametr **id_keeper** wówczas skrypt przyporządkuje automatycznie kolejnego użytkownika z okna Dział WWW -> panel Inwestycje.

Uwaga! Aby druga metoda zadziałała niezbędne jest przekazanie parametru `id_investment`, aby skrypt mógł znaleźć odpowiednich opiekunów powiązanych z inwestycją `id_investment`.

Skrypt AddClientInterest.ashx

Skrypt dodaje nowego nowego zainteresowanie do klienta. Podstawowa czynność to sprawdzenie, czy klient istnieje w bazie dewelopera po adresie `email` i jeżeli tak to dodajemy do niego zainteresowanie. Skrypt wywołujemy metodą POST.

POLE OBOWIĄZKOWE:

`email` adres e-mail.

POZOSTAŁE POLA DLA ZAINTERESOWANIA:

Jeżeli plik XML zawiera dodatkowe pola dotyczące produktu lub inwestycji, wówczas skrypt tworzy dodatkowy dokument: „Zainteresowanie klienta” po stronie bazy Deweloper System i zapisuje ww. dane.

<code>id_keeper</code>	identyfikator użytkownika programu DS. – opiekun zainteresowań.
<code>id_investment</code>	identyfikator inwestycji
<code>id_product</code>	identyfikator produktu lub identyfikatory dla wielu produktów, jako separator używamy znak przecinka: -, -. Np. 120,130,146
<code>id_interestkind</code>	identyfikator rodzaju kontaktu (znajduje się w bazie DS.)
<code>id_interestsource</code>	identyfikator źródła zainteresowania (znajduje się w bazie DS.) Menu Opcje -> Słowniki -> Opcje zainteresowań. Np.: RynekPirwotny.pl, Obido.pl itd. Jednym słowem skąd pochodzi zainteresowanie.
<code>sourcedescription</code>	opis źródła zainteresowania. Uwaga – pole nie może zawierać znaki Enter lub Tab (/n lub /r lub /t).
<code>subject</code>	temat -pole tekstowe na formularzu wypełniane przez klienta
<code>message</code>	opis - pole tekstowe na formularzu wypełniane przez klienta
<code>area</code>	pole tekstowe na formularzu wypełniane przez klienta
<code>rooms</code>	pole tekstowe na formularzu wypełniane przez klienta

wartości dla pola `area` określa jakim metrażem jest zainteresowany klient. Przykładowe wartości dla pola `area`: = 50, > 30, < 90, 30 ... 60 lub 30 – 60.

wartości dla pola `rooms` określa jaką ilością pokoi jest zainteresowany klient. Przykładowe wartości dla pola `rooms`: = 2, > 2, < 5, 3 ... 5 lub 3 – 5.

Zamiast pól `area` i `rooms` można użyć po dwa pola dla każdej wartości typu INTEGER:

<code>areafrom</code>	pole określa wartość metrażu od. Np. 60
<code>areato</code>	pole określa wartość metrażu do. Np. 83
<code>roomsfrom</code>	pole określa wartość ilości pokoje od. Np. 2
<code>roomsto</code>	pole określa wartość ilości pokoje do. Np. 4
<code>datecreated</code>	data dodania zainteresowania

Jeżeli pole `id_investment` zawiera wartość większą od zera, wtedy następuje dodanie nowego zainteresowania inwestycją do klienta w bazie DEWELOPER System. Wartości tego pól można uzyskać za pomocą skryptu `GetInvestments.ashx`.

Np.: <https://www.dws1.eu/scripts/GetInvestments.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Wartość pola `id_interestskind` (rodzaj kontaktu podczas tworzenia zainteresowania (telefon, wizyta, mail itd.) ?) można uzyskać za pomocą skryptu `GetInterestKind.ashx`. Dla stałych przypadków, identyfikator może zostać podany przez użytkownika programu DEWELOPER System. Przykładowy skrypt, który zwraca listę identyfikatorów źródła można wywołać pod adresem:

<https://www.dws1.eu/scripts/GetInterestKind.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Wartość pola `id_interestsource` (skąd się dowiedziałeś o naszej ofercie ?) można uzyskać za pomocą skryptu `GetInterestSource.ashx`. Dla stałych przypadków, identyfikator może zostać podany przez użytkownika programu DEWELOPER System. Przykładowy skrypt, który zwraca listę identyfikatorów źródła można wywołać pod adresem:

<https://www.dws1.eu/scripts/GetInterestSource.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

OBSŁUGA ZGÓD RODO

W świetle nowych przepisów, deweloper jest zobowiązany pobrać od klienta różnego rodzaju zgody na przetwarzania danych osobowych, jak również dać mu możliwość ich zmiany. DEWELOPER System zawiera kilka skryptów do obsługi zgód RODO. Skrypt wywołujemy metodą GET.

Skrypt GetAccepts.ashx

Skrypt służy do pobrania listy istniejących zgód w bazie dewelopera. Przykładowy skrypt, który zwraca listę zgód można wywołać pod adresem:

<https://www.dws1.eu/scripts/GetAccepts.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8>

Skrypt wywołujemy metodą GET.

Parametry: (oprócz key)

Format – format zwracanego pliku (XML, JSON). Domyślnie format to plik XML.

Oto struktura pliku XML ze zgodami zwracany przez skrypt:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <Error>0</Error>
  <Accepts>
    <Accept>
      <ID_Accept>1</ID_Accept>
      <IsActive>1</IsActive>
      <Body>Treść zgody</Body>
      <AcceptOffer>0</AcceptOffer>
      <AcceptOfferEmail>0</AcceptOfferEmail>
      <AcceptOfferPhone>1</AcceptOfferPhone>
      <AcceptServiceEmail>0</AcceptServiceEmail>
      <AcceptProfiling>0</AcceptProfiling>
      <IsWWW>1</IsWWW>
      <AcceptMustChecked>1</AcceptMustChecked>
    </Accept>
  </Accepts>
</root>
```

Pole:

ID_Accept – identyfikator zgody

IsActive – czy zgoda jest aktywna

Body – treść zgody

AcceptOffer – rodzaj zgody (oferta handlowa)

AcceptOfferEmail – rodzaj zgody (oferta handlowa elektroniczna - email)

AcceptOfferPhone – rodzaj zgody (oferta handlowa bezpośrednia - telefon)

AcceptServiceEmail – rodzaj zgody (obsługa elektroniczna - email)

AcceptProfiling – rodzaj zgody (profilowanie)

IsWWW - czy pokazywać na WWW

AcceptMustChecked - czy ma być zaznaczony przez klienta na WWW

Integracja programu DEWELOPER System ze stroną WWW

Na stronie (np. w formularzu kontaktowym) pokazujemy tylko aktywne zgody i z flagą **IsWWW** =1.

Skrypt AddAcceptsByEmail.ashx

Niniejszy skrypt dodaje tylko zaznaczone zgody z formularza na stronie internetowej do bazy DEWELOPER System. Dane przekazujemy za pomocą pliku XML. Identyfikacja klienta następuje po polu email. Skrypt dotyczy formularzy, które pobierają tylko zgody klientów. Na przykład po wykonaniu mailingu. Skrypt wywołujemy metodą POST.

Oto struktura pliku:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>
  <email>mail@mail.pl</email>
  <accepts>1,2,3</accepts>
</root>
```

Pole **accepts** zawiera identyfikatory zgód, które zostały zaznaczone przez klienta. Identyfikatory zgód można uzyskać za pomocą skryptu [GetAccepts.ashx](#).

Skrypt SetAcceptsByEmail.ashx

Niniejszy skrypt modyfikuje zgody wg zaznaczonych zgody z formularza na stronie internetowej do bazy DEWELOPER System. Dane przekazujemy za pomocą pliku XML. Identyfikacja klienta następuje po polu email. Skrypt wywołujemy metodą POST.

Oto struktura pliku:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>
  <email>mail@mail.pl</email>
  <accepts>1,2,3</accepts>
</root>
```

Pole **accepts** zawiera identyfikatory zgód, które zostały zaznaczone przez klienta. Identyfikatory zgód można uzyskać za pomocą skryptu [GetAccepts.ashx](#). Nie zaznaczone zgody nie przekazujemy i zostaną one usunięte z bazy DS.

To może Ci zainteresować:

Dodatkowej informacji na temat aktualizacji danych RODO za pomocą strony www można znaleźć na stronie: <https://www.deweloperserwer.eu/rodo>

OBSŁUGA KONTA KLIENTA

Do obsługi konta klienta służą dwa skrypty:

GetClientAccount.ashx – skrypt pobiera pełne dane dotyczące konta klienta.

SetClientPassword.ashx – skrypt daje możliwość zmiany hasła do konta klienta.

ResetClientPassword.ashx – skrypt daje możliwość resetowania hasła do konta klienta.

Dodatkowo do obsługi zgłaszania usterek przez klienta za pomocą formularza służy skrypt:

AddFaultByClient.ashx

Skrypt GetClientAccount.ashx

Skrypt pobiera pełne dane dotyczące konta klienta. Jako parametr podajemy login i hasło do konta klienta. Skrypt wywołujemy metodą POST.

Oto przykład pliku z parametrami:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Parametry -->
  <login>Demo</login>
  <password>Demo</password>
</root>
```

POLE OBOWIĄZKOWE:

login login do konta klienta.
password hasło do konta klienta.

Na podstawie loginu i hasła system weryfikuje dane klienta. Jeżeli dane są prawidłowe zwracany jest plik XML z zawartością konta klienta.

Oto struktura pliku XML z danymi klienta zwracany przez skrypt:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <Error>0</Error>
```

DANE KLIENTA

```
<Client>
  <ID_Client>2</ID_Client>
  <IsCompany>0</IsCompany>
  <IsVerify>1</IsVerify>
  <Code />
  <Title>Kowalski Zbigniew</Title>
  <DateCreated>2011-12-07T10:02:17</DateCreated>
  <Login>demo</Login>
  <IsFirstPassword>0</IsFirstPassword>
```

```
<DateLogin>2018-10-11T10:51:46.653</DateLogin>
<FirstName>Zbigniew</FirstName>
<SecondName />
<LastName>Kowalski</LastName>
<ParentsName />
<MaidenName />
<DateBirth>1899-12-30T00:00:00</DateBirth>
<Sex>0</Sex>
<Pesel>69010100879</Pesel>
<NIP />
<REGON />
<KRS />
<CompanyTitle />
<CompanyPartners />
<DebtNow>640204.5000</DebtNow>
<DebtDays>2111</DebtDays>
<Wedlock />
<IsLoginActive>1</IsLoginActive>
<ID_User>4</ID_User>
<ID_ClientInterestStatus>1</ID_ClientInterestStatus>
<Description />
```

ZAŁĄCZNIKI (dla klienta)

```
<BoxDocuments>
  <BoxDocument>
    <FileGUID>656732F5-83A9-4519-A105-B26023455322</FileGUID>
    <FileName>1.rtf</FileName>
    <FileSize>36456</FileSize>
  </BoxDocument>
</BoxDocuments>
</Client>
```

** patrz skrypt GetBoxDocument.aspx*

DANE OPIEKUNA

```
<UsersToClient>
  <UserToClient>
    <ID_User>3</ID_User>
    <FName>Marcel</FName>
    <LName>Fortuna</LName>
    <KeeperKind>Podstawowy</KeeperKind>
    <PhonesToUser>
      <Phone>
        <Number>+48 793 404 009</Number>
        <ID_PhoneKind>1</ID_PhoneKind>
        <PhoneKind>Telefon</PhoneKind>
      </Phone>
      <Phone>
        <Number>marcel@forbit.net</Number>
        <ID_PhoneKind>6</ID_PhoneKind>
        <PhoneKind>e-Mail</PhoneKind>
      </Phone>
    </PhonesToUser>
  </UserToClient>
</UsersToClient>
```

PODSUMOWANIA WSZYSKICH UMÓW

```
<ContractsToClientSum>
  <Brutto>996204.5000</Brutto>
  <BruttoToPay>996204.5000</BruttoToPay>
  <Income>356000.0000</Income>
  <DebtNow>640204.5000</DebtNow>
  <DebtInterest>0.0000</DebtInterest>
  <EscrowPayment>49431.0000</EscrowPayment>
```

```
</ContractsToClientSum>
```

SEKCJA ADRESY

```
<AddressesToClient>  
  <Address>  
    <ID_Address>1049</ID_Address>  
    <ID_AddressKind>2</ID_AddressKind>  
    <Street>ul. Krucza</Street>  
    <House>12</House>  
    <Door>3</Door>  
    <POB/>  
    <PIN>09-234</PIN>  
    <City>Warszawa</City>  
    <State/>  
    <Country>Polska</Country>  
    <AddressKind>Zameldowania</AddressKind>  
  </Address>  
</AddressesToClient>
```

SEKCJA KONTAKY (Telefony, adresy e-mail)

```
<PhonesToClient>  
  <Phone>  
    <ID_Phone>3</ID_Phone>  
    <ID_PhoneKind>1</ID_PhoneKind>  
    <Number>601454545</Number>  
    <IsMaster>1</IsMaster>  
    <PhoneKind>Telefon</PhoneKind>  
  </Phone>  
</PhonesToClient>
```

SEKCJA UMOWY

```
<ContractsToClient>  
  <Contract>  
    <ID_Contract>1</ID_Contract>  
    <ID_Stage>2</ID_Stage>  
    <ID_Investment>2</ID_Investment>  
    <InvestmentTitle>Eskadra</InvestmentTitle>  
    <StageTitle>Etap C</StageTitle>  
    <Number>IA-A-2012/5</Number>  
    <Brutto>351428.5000</Brutto>  
    <BruttoToPay>351428.5000</BruttoToPay>  
    <Income>117000.0000</Income>  
    <DebtNow>234428.5000</DebtNow>  
    <InterestValue>0.0000</InterestValue>  
    <DebtInterest>0.0000</DebtInterest>  
    <EscrowPayment>0.0000</EscrowPayment>
```

SEKCJA HARMONOGRAMY

```
<SchedulesToContract>  
  <Schedule>  
    <ID_Schedule>10</ID_Schedule>  
    <Number>IA-A-2012/5/H1</Number>  
    <Brutto>351428.5000</Brutto>  
    <Income>117000.0000</Income>  
    <DebtNow>234428.5000</DebtNow>
```

SEKCJA RATY DLA HARMONOGRAMU

```
<RatesToSchedule>  
  <Rate>  
    <DatePay>2012-02-01T00:00:00</DatePay>  
    <PercentOfSchedule>1.3650000000000000e+001</PercentOfSchedule>  
    <Brutto>48000.0000</Brutto>
```

```
    <Income>48000.0000</Income>
  </Rate>
  <Rate>
    <DatePay>2012-03-31T00:00:00</DatePay>
    <PercentOfSchedule>1.7080000000000000e+001</PercentOfSchedule>
    <Brutto>60000.0000</Brutto>
    <Income>60000.0000</Income>
  </Rate>
  <Rate>
    <DatePay>2012-12-30T00:00:00</DatePay>
    <PercentOfSchedule>5.4630000000000000e+001</PercentOfSchedule>
    <Brutto>192000.0000</Brutto>
    <Income>9000.0000</Income>
  </Rate>
  <Rate>
    <DatePay>2013-03-23T00:00:00</DatePay>
    <PercentOfSchedule>5.1220000000000000e+001</PercentOfSchedule>
    <Brutto>51428.5000</Brutto>
    <Income>0.0000</Income>
  </Rate>
</RatesToSchedule>
</Schedule>
</SchedulesToContract>
```

SEKCJA PRODYKTY W UMOWIE

```
<Products>
  <Product>
    <ID_Product>16</ID_Product>
    <ID_ProductKind>1</ID_ProductKind>
    <Number>SKX-A-LU-01</Number>
    <ProjectNumber />
    <Title>Lokal usługowy - 1 pokój</Title>
    <Area>1.2000000000000000e+002</Area>
    <RealArea>1.2000000000000000e+002</RealArea>
    <Netto>1080000.0000</Netto>
    <Brutto>1328400.0000</Brutto>
    <Vat>23</Vat>
    <TotalNetto>1080000.0000</TotalNetto>
    <TotalBrutto>1328400.0000</TotalBrutto>
  </Product>
  <Product>
    <ID_Product>93</ID_Product>
    <ID_ProductKind>5</ID_ProductKind>
    <Number>SKX-A-G-02</Number>
    <ProjectNumber />
    <Title>Nowy produkt</Title>
    <Area>3.0000000000000000e+001</Area>
    <RealArea>0.0000000000000000e+000</RealArea>
    <Netto>20491.8000</Netto>
    <Brutto>25000.0000</Brutto>
    <Vat>22</Vat>
    <TotalNetto>20491.8000</TotalNetto>
    <TotalBrutto>25000.0000</TotalBrutto>
  </Product>
</Products>
```

ZAŁĄCZNIKI (dla umowy)

```
<BoxDocuments>
  <BoxDocument>
    <FileGUID>39D4D94D-56E2-4AE2-AD1C-CA5D5A261E0D</FileGUID>
    <FileName>1.bmp</FileName>
    <FileSize>774</FileSize>
    <ID_BoxDocumentFolder>1</ID_BoxDocumentFolder>
    <FolderTitle>Test2</FolderTitle>
  </BoxDocument>
</BoxDocuments>
</Contract>
</ContractsToClient>
```

* patrz skrypt *GetBoxDocument.aspx*

SEKCJA WPŁATY POWIERNICZE

```
<EscrowPaymentForClient>
  <EscrowPayment>
    <ID_EscrowPayment>1</ID_EscrowPayment>
    <ID_EscrowAccount>1</ID_EscrowAccount>
    <ID_User>3</ID_User>
    <Date>2015-09-23T00:00:00</Date>
    <Quantity>50000.0000</Quantity>
    <Kind>0</Kind>
    <InvoicesCount>0</InvoicesCount>
    <ID_Contract>29</ID_Contract>
    <ID_User>3</ID_User>
    <ID_Company>1</ID_Company>
    <ID_Stage>1</ID_Stage>
    <ID_Status>2</ID_Status>
    <Number>IA-A-2013/6</Number>
    <Code />
    <DateCreated>2013-10-11T11:47:38</DateCreated>
    <DateReserved>2013-10-11T00:00:00</DateReserved>
    <DateSign>2013-10-11T00:00:00</DateSign>
    <DateCancel>1899-12-30T00:00:00</DateCancel>
    <DateFinal>1899-12-30T00:00:00</DateFinal>
    <Netto>593062.6900</Netto>
    <ValueVat>51713.3100</ValueVat>
    <Brutto>644776.0000</Brutto>
    <NettoToPay>593062.6900</NettoToPay>
    <ValueVatToPay>51713.3100</ValueVatToPay>
    <BruttoToPay>644776.0000</BruttoToPay>
    <Income>239000.0000</Income>
    <DebtNow>405776.0000</DebtNow>
    <DebtDays>1745</DebtDays>
    <InterestValue>0.0000</InterestValue>
    <InvoicesNetto>92197.3800</InvoicesNetto>
    <InvoicesValueVat>7802.6200</InvoicesValueVat>
    <InvoicesBrutto>100000.0000</InvoicesBrutto>
    <IsValorizationActive>0</IsValorizationActive>
    <DateValorizationStart>1899-12-30T00:00:00</DateValorizationStart>
    <DateValorizationEnd>1899-12-30T00:00:00</DateValorizationEnd>
    <ValorizationNetto>0.0000</ValorizationNetto>
    <ValorizationValueVat>0.0000</ValorizationValueVat>
    <ValorizationBrutto>0.0000</ValorizationBrutto>
    <IsInterest>0</IsInterest>
    <IsContractInterest>0</IsContractInterest>
    <IsContractValorization>0</IsContractValorization>
    <Warnings />
    <DateExpire>1899-12-30T00:00:00</DateExpire>
    <DateReceipt>1899-12-30T00:00:00</DateReceipt>
    <DateAct>1899-12-30T00:00:00</DateAct>
    <DebtInterest>0.0000</DebtInterest>
```

```
<DatePreSign>1899-12-30T00:00:00</DatePreSign>
<ID_EscrowAccount>1</ID_EscrowAccount>
<EscrowPayment>49431.0000</EscrowPayment>
<ExtraData>
  <?xml version="1.0" encoding="windows-1250"?>
  <Values>
</ExtraData>
<ID_ClientToContract>62</ID_ClientToContract>
<ID_Client>2</ID_Client>
<ID_Contract>29</ID_Contract>
<ID_PartInContract>48</ID_PartInContract>
<IsMaster>0</IsMaster>
<Description />
<Priority>0</Priority>
</EscrowPayment>
</EscrowPaymentForClient>
```

SEKCJA WPŁATY

```
<PaymentsToClient>
  <Payment>
    <ID_Payment>1</ID_Payment>
    <ID_PaymentKind>1</ID_PaymentKind>
    <ID_Client>2</ID_Client>
    <ID_User>4</ID_User>
    <Number>071211/2</Number>
    <Assign>Wpłata za mieszkanie</Assign>
    <DatePay>2011-12-07T00:00:00</DatePay>
    <Payment>80000.0000</Payment>
    <PaymentRest>0.0000</PaymentRest>
    <PaymentInCurrency>0.0000</PaymentInCurrency>
    <Content />
    <Description />
    <Refund>0.0000</Refund>
    <InvoicesCount>1</InvoicesCount>
    <ID_Company>2</ID_Company>
    <DateImport>1899-12-30T00:00:00</DateImport>
    <ImportCode />
    <AccountNumber />
    <AccountBank />
    <Kind>0</Kind>
  </Payment>
</PaymentsToClient>
```

SEKCJA REZERWACJE

```
<ReservationsToClient>
  <Reservation>
    <ID_Reservation>1</ID_Reservation>
    <ID_ReservationStatus>6</ID_ReservationStatus>
    <ID_Client>2</ID_Client>
    <ID_User>4</ID_User>
    <ID_Product>41</ID_Product>
    <ID_Contract>1</ID_Contract>
    <IsVerify>1</IsVerify>
    <Number>ES-C-LM-01/07-12-11/1</Number>
    <DateCreated>2011-12-07T10:06:11</DateCreated>
    <DateStart>2011-12-07T00:00:00</DateStart>
    <DateEnd>2011-12-09T00:00:00</DateEnd>
    <Days>3</Days>
    <Cost>0.0000</Cost>
    <Vat>8</Vat>
    <Priority>1</Priority>
    <DateToPay>1899-12-30T00:00:00</DateToPay>
    <Description />
```

```
<Income>0.0000</Income>
<DebtNow>0.0000</DebtNow>
<DebtDays>0</DebtDays>
<IsSign>0</IsSign>
<ProductNumber>ES-C-LM-01</ProductNumber>
<ProductTitle>Lokal mieszkalny - 2 pokoje z kuchnią</ProductTitle>
</Reservation>
</ReservationsToClient>
```

SEKCJA USTERKI

```
<FaultsForClient>
  <Faults> * patrz skrypt AddFaultByClient.ashx
    <ID_Fault>1</ID_Fault>
    <ID_User>4</ID_User>
    <ID_Client>2</ID_Client>
    <ID_Product>41</ID_Product>
    <ID_FaultState>1</ID_FaultState>
    <Number>2</Number>
    <DateCreate>2011-12-07T10:39:53</DateCreate>
    <DateApplication>2011-12-07T00:00:00</DateApplication>
    <DateInspection>2011-12-09T00:00:00</DateInspection>
    <DateToRemoved>2011-12-16T00:00:00</DateToRemoved>
    <DateRemoved>2011-12-17T00:00:00</DateRemoved>
    <DateReceipt>1899-12-30T00:00:00</DateReceipt>
    <ResponsiblePerson />
    <Comments>Test programu</Comments>
    <ID_Investment>2</ID_Investment>
    <ID_Stage></ID_Stage>
    <ID_Product></ID_Product>
    <CommonPart>0</CommonPart>
    <BoxDocuments> * patrz skrypt GetBoxDocument.ashx
      <BoxDocument>
        <FileGUID>786CD53C-ECA0-4721-A7F2-05A55626D976</FileGUID>
        <FileName>1.txt</FileName>
        <FileSize>936289</FileSize>
      </BoxDocument>
    </BoxDocuments>
  </Faults>
</FaultsForClient>
```

SEKCJA PRODUKTY

```
<ProductsForClient>
  <Product>
    <ID_Product>16</ID_Product>
    <ID_ProductKind>1</ID_ProductKind>
    <ProductsKindTitle>Lokal mieszkalny</ProductsKindTitle>
    <Number>IA-A-LM-06</Number>
    <ProjectNumber />
    <Title>Lokal mieszkalny - 3 pokoje z kuchnią</Title>
    <ID_Investment>1</ID_Investment>
  </Product>
  <Product>
    <ID_Product>93</ID_Product>
    <ID_ProductKind>5</ID_ProductKind>
    <ProductsKindTitle>Garaż</ProductsKindTitle>
    <Number>IA-A-G-07</Number>
    <ProjectNumber />
    <Title>Garaż</Title>
    <ID_Investment>1</ID_Investment>
  </Product>
</ProductsForClient>
```

SEKCJA FAKTURY

```
<InvoicesForClient>
  <Invoice>
    <ID_Company>4</ID_Company>
    <ID_Contract>4429</ID_Contract>
    <ID_Investment>5</ID_Investment>
    <Number>2/06/2009/SKX</Number>
    <DateInvoice>2009-06-26T00:00:00</DateInvoice>
    <DateSales>2009-06-26T00:00:00</DateSales>
    <PayNetto>2000.0000</PayNetto>
    <PayValueVat>440.0000</PayValueVat>
    <PayBrutto>2440.0000</PayBrutto>
    <Income>0.0000</Income>
    <FileGUID>786CD53C-ECA0-4721-A7F2-05A55626D976</FileGUID>
    <BoxDocuments> * patrz skrypt GetBoxDocument.ashx
      <BoxDocument>
        <FileGUID>786CD53C-ECA0-4721-A7F2-05A55626D976</FileGUID>
        <FileName>1.txt</FileName>
        <FileSize>936289</FileSize>
      </BoxDocument>
    </BoxDocuments>
  </Invoice>
</InvoicesForClient>
</root>
```

UWAGA

Pole `<IsFirstPassword>` wskazuje na to czy hasło jest pierwotne: 1 TAK, 0 NIE.
Jeżeli hasło jest pierwotne trzeba zaproponować zmianę hasła.

Skrypt SetClientPassword.ashx

Skrypt zmienia hasło do konta klienta. Jako parametr podajemy login i stare hasło do konta klienta.
Skrypt wywołujemy metodą POST.

Oto przykład pliku z parametrami:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Parametry -->
  <login>Demo</login>
  <password>Demo</password>
  <newpassword>NewDemo</newpassword>

</root>
```

POLE OBOWIĄZKOWE:

login	login do konta klienta.
password	hasło do konta klienta.
newpassword	nowe hasło do konta klienta.

Na podstawie loginu i hasła system weryfikuje dane klienta. Jeżeli dane są prawidłowe zmienia hasło do konta klienta na nowe (newpassword).

Skrypt ResetClientPassword.ashx

Skrypt zmienia hasło do konta klienta. Jako parametr podajemy klucz i adres e-mail klienta. Skrypt wywołujemy metodą GET.

Przykład wywołania skrypt, który resetuje hasło wywołać pod adresem:

<https://www.dws1.eu/scripts/ResetClientPassword.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&email=demo@demo.pl>

POLE OBOWIĄZKOWE:

key	klucz.
email	e-mail klienta.

Na podstawie emaila system weryfikuję dane konto klienta. Jeżeli dane są prawidłowe zmienia hasło do konta klienta na nowe (przypadkowe) i wysyła go na podany adres email.

Skrypt AddFaultByClient.ashx

Skrypt służy do zgłaszania przez klienta nowej usterki. Skrypt wywołujemy metodą POST.

Oto przykład pliku z parametrami:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Parametry -->
  <id_client>2</id_client>
  <id_investment>41</id_investment>
  <id_product>12</id_product>
  <comments>uwagi klienta dotyczące usterki</comments>
  <notification>opis do zgłaszanej usterki</notification>
  <commonpart>0</commonpart>
</root>
```

POLE OBOWIĄZKOWE:

Id_client identyfikator klienta (uzyskujemy za pomocą [GetClientAccount.ashx](#))
Id_investment identyfikator inwestycji której dotyczy usterka.

Pozostałe pola nie są obowiązkowe, lecz uzupełniają dane usterki:

Id_product identyfikator produktu którego dotyczy usterka.
Id_stage identyfikator etapu którego dotyczy usterka.
commonpart jeżeli usterka dotyczy części wspólnych to 1 inaczej 0.
comments uwagi klienta dotyczące usterki.
notification opis do zgłaszanej usterki.

Identyfikator inwestycji oraz produktu można uzyskać ze skryptu [GetClientAccount.ashx](#). W zwracanym pliku XML znajduje się sekcja `<ProductsForClient>`. Zawiera ona wszystkie produkty kupione przez klienta. W oparciu o tę kolekcję można stworzyć listę produktów z której klient wybiera produkt, którego dotyczy usterka. Również sekcja `<FaultsForClient>` pozwala na wyświetlenie listy poprzednio zgłaszanych usterek przez klienta.

Wynikiem skryptu jest **ID_Fault** – czyli identyfikator nowej usterki.

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <Error>0</Error>
  <ID_Fault>47</ID_Fault>
</root>
```

OBSŁUGA KAMPANIJ EMAIL

Do obsługi newsletteru służą trzy skrypty:

RegisterOpenEmail.ashx – skrypt rejestruje otwarcie listu przez klienta.

RegisterClickEmail.ashx – skrypt rejestruje kliknięcie przez klienta na otwartym liście.

Oba skrypty oprócz klucza potrzebują tylko jednego parametru: ID_Email.

Trzeci skrypt **RemoveFromNewsletter.ashx** służy do wypisywania się klienta z newsletterów.

W tym skrypcie oprócz kucza podajemy jako parametr adres email: Email.

Skrypt RegisterOpenEmail.ashx

Skrypt rejestruje otwarcie listu przez klienta. Skrypt wywołujemy metodą GET. Jako parametr podajemy klucz oraz identyfikator listu ID_Email:

https://www.dws1.eu/scripts/registeropenemail.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Email=3779

POLE OBOWIĄZKOWE:

ID_Email identyfikator listu.

Skrypt RegisterClickEmail.ashx

Skrypt rejestruje kliknięcie na otwartym przez klienta liście. Skrypt wywołujemy metodą GET. Jako parametr podajemy klucz oraz identyfikator listu ID_Email:

https://www.dws1.eu/scripts/registerclickemail.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Email=3779

POLE OBOWIĄZKOWE:

ID_Email identyfikator listu.

Dane rozwiązanie działa pod warunkiem że klient poczty otwiera list jako przeglądarka HTML.

Większość programów pocztowych ze względu na bezpieczeństwo blokuje wykonanie JavaScript-ów.

Skąd pobrać identyfikator listu ID_Email. Otóż podczas generowania treści maila program Deweloper System automatycznie podmienia pola zawarte w szablonie newsletteru. Tworząc szablon listu używamy pole [EMAIL.ID], które podczas generowania wysyłki zostanie zamienione odpowiednią wartością.

Skrypt RegisterClickUrl.ashx

Jest to alternatywa do poprzedniego skryptu który nie wymaga użycia kodu JavaScript. Składnia jest identyczna lecz jako dodatkowy paramet występuje adres URL na który ma przejść klient klikając na link.

Oto przykład wywołania skryptu:

https://www.dws1.eu/scripts/registerclickurl.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&ID_Email=3779&gotourl=https://www.dewelopererwer.eu

POLE OBOWIĄZKOWE:

ID_Email identyfikator listu.
gotourl adres URL na który zostanie przeniesiony klient po kliknięciu na link.

Przykład szablonu do listu email można znaleźć pod adresem:

<https://www.dewelopererwer.eu/scripts/doc/newsletter.html>

Prosimy pobrać plik i przeanalizować jego treść.

Skrypt RemoweFromNewsletter.ashx

Skrypt wyłącza zgodę klienta na otrzymywanie ofert za pomocą poczty elektronicznej. Jako parametr podajemy klucz oraz adres Email:

<https://www.dws1.eu/scripts/remowefromnewsletter.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&Email=jankowalski@poczta.pl>

POLE OBOWIĄZKOWE:

Email adres email.

Skąd pobrać adres email **Email**. Otóż podczas generowania treści maila program Deweloper System automatycznie podmienia pola zawarte w szablonie newsletteru. Tworząc szablon listu używamy pole [KLIENT.EMAIL], które podczas generowania wysyłki zostanie zamienione odpowiednią wartością.

SKRYPTY NARZĘDZIOWE

OBSŁUGA ZAŁĄCZNIKÓW

Kolekcja skryptów zawiera kilka skryptów narzędziowych, które służą do pobrania pomocniczych elementów.

Skrypt `GetBoxDocument.ashx`

Skrypt służy do pobrania załącznika w formie pliku XML wraz z jego zawartością. Jako parametr podajemy identyfikator załącznika. Jest to unikalny ciąg znaków dla wybranego załącznika. Skrypt wywołujemy metodą GET:

<https://www.dws1.eu/scripts/getboxdocument.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&fileguid=491F3EAA-FD93-4639-809B-DAC6AE9F5CED>

POLE OBOWIĄZKOWE:

`FileGuid` identyfikator załącznika (File GUID).

Oto struktura pliku XML z treścią załącznika zwracany przez skrypt:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <Error>0</Error>
  <FileName>1.rtf</FileName>
  <FileSize>31098</FileSize>
  <DateCreated>2011-12-07</DateCreated>
  <Title>Umowa</Title>
  <Data>...</Data>
  <UCPD></UCPD>
  <Description>Wygenerowano na podstawie szablonu.</Description>
</root>
```

Pole `FileName` zawiera nazwę pliku załącznika.

Pole `Data` zawiera zakodowaną w formie Base64 treść pliku załącznika.

Skrypt `GetBoxDocumentInfo.ashx`

Skrypt służy do pobrania informacji o załączniku w formie pliku XML bez jego zawartości. Jako parametr podajemy identyfikator załącznika. Jest to unikalny ciąg znaków dla wybranego załącznika. Skrypt wywołujemy metodą GET:

<https://www.dws1.eu/scripts/getboxdocumentinfo.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&fileguid=491F3EAA-FD93-4639-809B-DAC6AE9F5CED>

POLE OBOWIĄZKOWE:

`FileGuid` identyfikator załącznika. (File GUID).

Oto struktura pliku XML z treścią załącznika zwracany przez skrypt:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <Error>0</Error>

  <FileName>1.rtf</FileName>
  <FileSize>31098</FileSize>
  <DateCreated>2011-12-07</DateCreated>
  <Title>Umowa</Title>
  <UCPD></UCPD>
  <Description>Wygenerowano na podstawie szablonu.</Description>
</root>
```

Skrypt GetBoxDocumentBody.ashx

Skrypt służy do pobrania tylko zawartości załącznika, która jest zwracana jako ciąg znaków zakodowanych w formie Base64. Jako parametr podajemy identyfikator załącznika. Jest to unikalny ciąg znaków dla wybranego załącznika. Skrypt wywołujemy metodą GET:

<https://www.dws1.eu/scripts/getboxdocumentbody.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&fileguid=491F3EAA-FD93-4639-809B-DAC6AE9F5CED>

POLE OBOWIĄZKOWE:

FileGuid identyfikator załącznika. (File GUID).

Skrypt GetDescriptionFor.ashx

Skrypt służy do pobrania opisu dla kilku elementów systemu. Są to elementy: firma, inwestycja, etap i produkt. Jako parametr podajemy rodzaj elementu, jego identyfikator oraz identyfikator rodzaju opisu. Skrypt wywołujemy metodą GET:

<https://www.dws1.eu/scripts/getdescriptionfor.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&OwnerKind=STAGE&OwnerID=1&Kind=2>

Wszystkie paramenty są obowiązkowe.

Parametr **OwenrKind** zawiera następujące wartości:

COMPANY	dla elementu typu firma,
INVESTMENT	dla elementu typu inwestycja,
STAGE	dla elementu typu etap,
PRODUCT	dla elementu typu produkt (lokal, komórka itd.)

Parametr **Kind** zawiera następujące wartości:

- 1 Opis skrócony
- 2 Pełny opis
- 3 Opis lokali usługowych
- 4 Opis lokali biurowych
- 5 Opis lokalizacji
- 6 Zaawansowanie robót
- 7 Sprzedaż prowadzi
- 8 Termin rozpoczęcia
- 9 Termin oddania
- 10 Wzór umowy
- 11 Standard robót
- 12 Opis do stopki w ofercie handlowej

Oto struktura pliku XML z treścią załącznika zwracany przez skrypt:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <Error>0</Error>
  <ID_DescriptionKind>2</ID_DescriptionKind>
  <Description>Kontakt:& lt;b&gt;&lt;br /&gt; 22 001 02 03&lt;br /&gt;
bok@deweloper.pl&lt;/b&gt;&lt;br /&gt;& lt;br /&gt; Zakończenie budowy - uzyskanie
pozwolenia na użytkowanie: &lt;br /&gt; nie p&#243;źniej niż do &lt;b&gt;31 grudnia
2020&lt;/b&gt;&lt;br /&gt;& lt;br /&gt; Termin odbioru technicznego: &lt;br /&gt;
nie p&#243;źniej niż do &lt;b&gt;31 stycznia 2021&lt;/b&gt;&lt;br /&gt;& lt;br
/&gt; Termin przekazania posiadania:&lt;br /&gt; nie p&#243;źniej niż do
&lt;b&gt;31 marca 2021&lt;/b&gt;&lt;br /&gt;& lt;br /&gt; Termin zawarcia Umowy
Przeniesienia:&lt;br /&gt; nie p&#243;źniej niż do &lt;b&gt;30 czerwca
2021&lt;/b&gt;</Description>
</root>
```

Pole `<Description>` zawiera treść opisu w formacie HTML zakodowane do pliku XML za pomocą funkcji `HttpUtility.HtmlEncode`, która konwertuje ciąg na ciąg kodowany w formacie HTML.

Skrypt AddBoxDocumentToClient.ashx

Skrypt służy do dodania zawartości załącznika do klienta. Skrypt wywołujemy metodą POST.

Oto przykład pliku z parametrami:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Parametry -->
  <id_client>2</id_client>
  <filename>MyFile.txt</filename>
  <filebody>PHJvb3Q+PEVycm9yPjA8L0Vycm9yPjwvc9vdD4=</filebody>
  <id_user>1</id_user>
</root>
```

POLE OBOWIĄZKOWE:

id_client	identyfikator klienta (uzyskujemy za pomocą GetClientAccount.ashx)
filename	nazwa pliku załącznika.
filebody	treść załącznika.

Jeżeli chcemy określić konkretnego użytkownika który dodał załącznik używamy pole `id_user`.

UWAGA

1. Pole `filebody` przekazujemy jako typ `string` kodowany Base64,
2. Pole `filename` zawiera tylko nazwę pliku bez ścieżki,
3. Rozmiar pliku załącznika nie może przekroczyć 300 KB.

Wynikiem skryptu jest `ID_BoxDocument` i `FileGUID` – czyli unikalny identyfikator załącznika w formie ciągu znaków GUID.

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <Error>0</Error>
  <ID_BoxDocument>1</ID_BoxDocument>
  <FileGUID>8e79e628-21cc-4a9a-a478-90ab551ab4a6</FileGUID>
</root>
```

Oto przykładowy kod w języku ASP.NET C# jak można wywołać ww. skrypt:

www.deweloperserwer.eu/scripts/doc/example3.txt

Skrypt AddBoxDocumentToFault.ashx

Skrypt służy do dodania zawartości załącznika do usterki. Skrypt wywołujemy metodą POST.

Oto przykład pliku z parametrami:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Parametry -->
  <id_fault>123</id_fault>
  <filename>MyFile.txt</filename>
  <filebody>PHJvb3Q+PEVycm9yPjA8L0Vycm9yPjwvcm9vdD4=</filebody>
  <id_user>1</id_user>
  <id_client>2</id_client>
</root>
```

POLE OBOWIĄZKOWE:

id_fault	identyfikator usterki (zwracany ze skryptu AddFaultByClient.ashx).
filename	nazwa pliku załącznika.
filebody	treść załącznika.

Jeżeli chcemy określić konkretnego klienta który dodał załącznik używamy pole **id_client**. (uzyskujemy za pomocą [GetClientAccount.ashx](#)). Jeżeli chcemy określić konkretnego użytkownika który dodał załącznik używamy pole **id_user**.

UWAGA

1. Pole **filebody** przekazujemy jako typ `string` kodowany Base64,
2. Pole **filename** zawiera tylko nazwę pliku bez ścieżki,
3. Rozmiar pliku załącznika nie może przekroczyć 300 KB.

Wynikiem skryptu jest **ID_BoxDocument** i **FileGUID** – czyli unikalny identyfikator załącznika w formie ciągu znaków GUID.

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <Error>0</Error>
  <ID_BoxDocument>1</ID_BoxDocument>
  <FileGUID>8e79e628-21cc-4a9a-a478-90ab551ab4a6</FileGUID>
</root>
```

Oto przykładowy kod w języku ASP.NET C# jak można wywołać ww. skrypt:

www.deweloperserwer.eu/scripts/doc/example4.txt

OBSŁUGA PLIKÓW I RYSUNKÓW

Skrypt `GetAttachmentFor.ashx`

Skrypt służy do pobrania załącznika dla kilku elementów systemu. Są to elementy: firma, inwestycja, etap i produkt. Jako parametr podajemy rodzaj elementu, jego identyfikator oraz identyfikator rodzaju opisu. Skrypt wywołujemy metodą GET:

<https://www.dws1.eu/scripts/getattachmentfor.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&OwnerKind=STAGE&OwnerID=1&Kind=1>

Wszystkie parametry są obowiązkowe.

Parametr `OwnerKind` zawiera następujące wartości:

<code>COMPANY</code>	dla elementu typu firma,
<code>INVESTMENT</code>	dla elementu typu inwestycja,
<code>STAGE</code>	dla elementu typu etap,
<code>PRODUCT</code>	dla elementu typu produkt (lokal, komórka itd.)

Parametr `Kind` zawiera następujące wartości:

1	Wzór umowy
2	Standard robót
3	Plan lokalu (Format PDF)
4	Oferta handlowa
5	Opis inwestycji (Format PDF)

Skrypt GetPictureFor.ashx

Skrypt służy do pobrania rysunków (JPEG, BMP) dla kilku elementów systemu. Są to elementy: firma, inwestycja, etap i produkt. Jako parametr podajemy rodzaj elementu, jego identyfikator oraz identyfikator rodzaju opisu. Skrypt wywołujemy metodą GET:

<https://www.dws1.eu/scripts/getpicturefor.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&OwnerKind=STAGE&OwnerID=1&Kind=2>

Wszystkie parametry są obowiązkowe.

Parametr `OwnerKind` zawiera następujące wartości:

<code>COMPANY</code>	dla elementu typu firma,
<code>INVESTMENT</code>	dla elementu typu inwestycja,
<code>STAGE</code>	dla elementu typu etap,
<code>PRODUCT</code>	dla elementu typu produkt (lokal, komórka itd.)

Parametr `Kind` zawiera następujące wartości:

1	Ikona
2	wizualizacja 1
3	wizualizacja 2
4	Plan lokalu
5	Umiejscowienie
6	Mapa 1
7	Mapa 2
8	Mapa 3
12	Ogłoszenia
14	Logo
15	Logo dodatkowe
16	Logo do faktury
17	Plan osiedla
18	Karta produktu
19	WWW Informacje
20	WWW Informacje Ikona
21	Umiejscowienie w budynku

Skrypt GetUserPicture.ashx

Skrypt służy do pobrania rysunków (JPEG, BMP) dla użytkownika - zdjęcie. Jako parametr podajemy identyfikator użytkownika `UserID` oraz klucz. Skrypt wywołujemy metodą GET:

<https://dws1.eu/scripts/GetUserPicture.ashx?key=b7409fb0-88d1-4bae-b834-bfaaf7648fd8&UserID=4>

UŻYTKOWNIK API

OPIS

Użytkownik API – pozwala na logowania do API – wersja wzbogacona o dostęp do rozrzedzonych danych, które nie są dostępne w dotychczasowych skryptach. Użytkownik API dodany jest do systemu za pomocą programu Deweloper System z poziomu panelu administracyjnego. Użytkownik API nie może logować się bezpośrednio do programu Deweloper System, lecz dostęp do danych uzyskuje tylko za pomocą API.

WYWOŁANIE PROCEDUR DLA UŻYTKOWNIKA API

Różnica między wywołaniem poprzednich procedur a procedur dla użytkownika API polega na dodaniu dwóch dodatkowych parametrów: loginu i hasła. Wszystkie skrypty dla użytkowników API wywołujemy metodą POST.

Oto przykład pliku:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <!--login i hasło -->
  <login>USERAPI</login>
  <password>test1234</password>

  <!-- klucz -->
  <key>b7409fb0-88d1-4bae-b834-bfaaf7648fd8</key>

  <!-- Parametry -->
  <ID_Investment>1</ID_Investment>
  <ID_Stage>1</ID_Stage>
  <ID_ContractStatus>0</ID_ContractStatus>
</root>
```

Skrypty dla użytkowników API są przeważnie robione na zamówienie dewelopera i z tego powodu nie podajemy ich opis w niniejszej instrukcji.