

Структури оперативного інформаційного обміну

1. Терміни та скорочення

Транспортний конверт - умовна назва http-запиту до програмного комплексу ПФУ типу POST.

Прикладна система - будь-яка прикладна система учасників обміну.

Запитувач - зовнішній учасник обміну.

Провайдер - прикладна підсистема внутрішнього учасника обміну.

Пакет - умовна назва блока інформації що є предметом обміну (структура пакету може будь якої та узгоджується між користувачами обміну окремо).

Учасники обміну - в поточному документі:

- *Внутрішній* – Пенсійний фонд України;
- *Зовнішній* – Надавач житло-комунальних послуг.

Програмний комплекс ПФУ - програмний комплекс електронних сервісів та інформаційної взаємодії веб-порталу електронних послуг Пенсійного фонду України.

Унікальний номер електронної облікової картки застрахованої особи – це 10-ти значний номер персональної електронної облікової картки застрахованої особи в реєстрі застрахованих осіб Державного реєстру загальнообов’язкового державного соціального страхування, до якої включаються відомості про застрахованих осіб, інформація про набуття прав на одержання страхових виплат за всіма видами загальнообов’язкового державного соціального страхування та інформація про виплати за всіма видами загальнообов’язкового державного соціального страхування. Унікальний номер електронної облікової картки формується автоматично шляхом додавання одиниці до останнього наявного унікального номера електронної облікової картки.

(ask) – ознака розділу, що описує вхідну інформацію до ПФУ.

(answer) – ознака розділу, що описує вихідну інформацію від ПФУ.

2. Запити до програмного комплексу ПФУ

Запити до програмного комплексу ПФУ є http-запитами типу POST, значущою частиною яких є xml визначеної в `simple_request` структури в кодуванні Unicode (UTF-8). Результатом запиту є xml такої ж структури. В подальшому ця структура буде називатись «транспортний конверт».

2.1. Транспортний конверт - `simple_request`

Структура `simple_request`

Опис атрибутів транспортного конверту:

- **VER** – версія транспортного конверту (поточний документ описує версію 0001);
- **CODE** – код запиту до програмного комплексу ПФУ (можливі значення: `get_request_status`, `get_request_data`, `get_ask_query_list`, `get_query_data`, `post_response` ...);
- **NUM** – ідентифікатор запиту з прикладної системи зовнішнього учасника обміну;
- **FROM** – ідентифікатор зовнішнього учасника обміну в програмному комплексі ПФУ;

- TO – ідентифікатор провайдера запиту (унікальний реєстраційний код зовнішнього учасника обміну в ПФУ, до 20 символів латиниці та/або цифри);
- DATA – дані запиту, повинні бути зашифровані за допомогою функції EnvelopDataToRecipients модуля криптографічних перетворень ІТ або сумісних (зі сторони ПФУ використовуються функції EnvelopDataToRecipients та DevelopData). Дані записуються в base64-кодуванні;
- ECP – електронний цифровий підпис на поля транспортного конверта.
Не використовується для прикладних потреб. Інформаційний об'єкт для обчислення ЕЦП визначається за формулою: VER + '#' + CODE + '#' + NUM + '#' + FROM + '#' + TO + '#' + DATA (формула записана на С#). Для обчислення ЕЦП повинен використовуватись модуль криптографічних перетворень ІТ або сумісний (зі сторони ПФУ використовуються функції RawSignData та RawVerifyData);
- ECODE – код помилки, якщо реєстрація запита успішна (формат, ЕЦП, шифрування – успішні), тоді ECODE поверне 0;
- EMESSAGE – при ECODE≠0 містить діагностичне повідомлення.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="request">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="ver"/>
        <xs:element type="xs:string" name="code"/>
        <xs:element type="xs:string" name="num"/>
        <xs:element type="xs:string" name="from"/>
        <xs:element type="xs:string" name="to"/>
        <xs:element type="xs:string" name="ecp"/>
        <xs:element type="xs:string" name="data"/>
        <xs:element type="xs:string" name="ecode"/>
        <xs:element type="xs:string" name="emessage"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Приклад simple_request

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <ver>0001</ver>
  <code>get_query_data</code>
  <num>233</num>
  <from>z</from>
  <to>x</to>
  <ecp>ksdljfklsjflkasjdfk</ecp>
  <data>MIIС8AYJKoZIh... ASPm24+sV5BIAJP2gE/pSg89B5</data>
  <ecode></ecode>
  <emessage></emessage>
</request>
```

Програмний комплекс ПФУ надає можливість виконувати 2 типи запитів: сервісні та прикладні. Прикладні запити є запитами реєстрації задач для обробки прикладними системами та виконуються асинхронно. Результатом прикладного запиту, який повертає програмний

комплекс ПФУ в http-сесії, є інформація про ідентифікатор зареєстрованого запиту. Сервісні запити є запитами безпосередньо до ядра програмного комплексу ПФУ, призначені для перевірки стану запиту, отримання результату обробки прикладних запитів та виконуються синхронно. Сервісні та прикладні запити формуються в кодуванні Unicode (UTF-8).

3. Сервісні запити

3.1. Сервіс «Отримання стану запиту» (get_request_status)

Призначений для отримання стану запиту. Дані запиту є xml визначеної в файлі get_request_status (ask) структури. Результатом запиту є xml визначеної в файлі get_request_status (answer) структури.

Запит - get_request_status (ask)

Структура get_request_status (ask)

Опис атрибутів запиту get_request_status:

- id – ідентифікатор запиту в програмному комплексі ПФУ. Отримується в структурі відповіді підчас реєстрації прикладних запитів. Обов'язковий параметр.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="filter">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="id"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Приклад get_request_status (ask)

```
<?xml version="1.0" encoding="utf-8"?>
  <filter>
    <id>777</id>
  </filter>
```

Результат запиту - get_request_status (answer)

Структура get_request_status (answer)

Опис атрибутів відповіді get_request_status:

- rq_id – ідентифікатор запиту в програмний комплекс ПФУ. Використовується в сервісних запитах «Отримання стану запиту» (get_request_status) та «Отримання результату запиту» (get_request_data);
- rq_st – ідентифікатор стану запиту (можливі значення: R – запит зареєстровано; S – запит оброблено та надіслано; D – запит скасовано);

- `parts_cnt` – має значення при `rq_st=S`, та зазначає кількість частин відповіді (якщо розмір відповіді, перетвореної в `base64`, перевищує допустиме значення, відповідь, закодована в `base64`, розподіляється на частини);
- `rq_st_name` – назва стану запиту.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="requeststatus">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="rq_id"/>
        <xs:element type="xs:string" name="rq_st"/>
        <xs:element type="xs:string" name="rq_st_name"/>
        <xs:element type="xs:string" name="parts_cnt"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Приклад `get_request_status` (answer)

```
<?xml version="1.0" encoding="utf-8"?>
  <requeststatus>
    <rq_id>777</rq_id>
    <rq_st>S</rq_st>
    <rq_st_name>Зареєстровано</rq_st_name>
    <parts_cnt>17</parts_cnt >
  </requeststatus>
```

3.2. Сервіс «Отримання результату запиту» (`get_request_data`)

Призначений для отримання результату обробки прикладного запиту. Дані запиту є `xml` визначеної в файлі `get_request_data` (ask) структури. Результатом запиту є `xml` визначеної в файлі `get_request_data` (answer) структури.

Zanum - get_request_data (ask)

Структура `get_request_data` (ask)

Опис атрибутів запиту `get_request_data`:

- `id` – ідентифікатор запиту в програмному комплексі ПФУ. Отримується в структурі відповіді під час реєстрації прикладних запитів. Обов'язковий параметр;
- `part` – частина відповіді.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="filter">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="id"/>
        <xs:element type="xs:string" name="part"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Приклад get_request_data (ask)

```
<?xml version="1.0" encoding="utf-8"?>
  <filter>
    <id>777</id>
    <part>5</part>
  </filter>
```

Результат запиту - get_request_data (answer)

Структура get_request_data (answer)

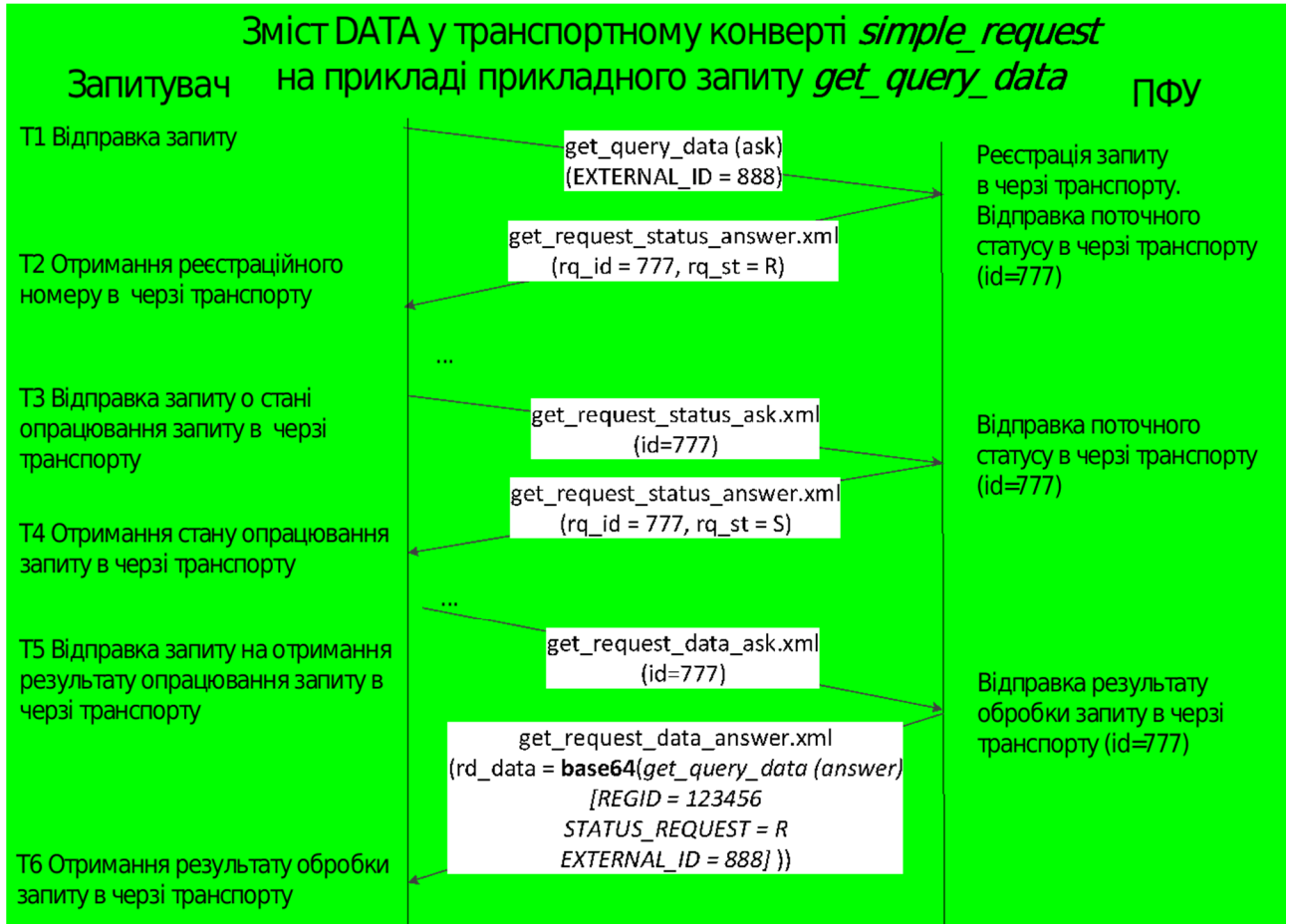
Опис атрибутів відповіді get_request_data:

- rd_id – ідентифікатор запису з даними в програмному комплексі ПФУ. Використовується для технічної підтримки;
- rd_rq – ідентифікатор запиту в програмному комплексі ПФУ;
- part – частина відповіді;
- rd_data – дані обробки прикладного запиту в base64-кодуванні, структура даних визначається в залежності від типу прикладного запиту, дані обробки якого отримуються цим сервісним запитом.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="requestdata">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="rd_id"/>
        <xs:element type="xs:string" name="rd_rq"/>
        <xs:element type="xs:string" name="part"/>
        <xs:element type="xs:string" name="rd_data"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Приклад get_request_data (answer)

```
<?xml version="1.0" encoding="utf-8"?>
  <requestdata>
    <rd_id>906</rd_id>
    <rd_rq>777</rd_rq>
    <part>5</part>
    <rd_data>akfjdkafjklsjdfkjasklfdjalsfkda</rd_data>
  </requestdata>
```



Малюнок 1 Робота транспортного рівня ПФУ на прикладі опрацювання сервісу «Отримання запиту» (*get_query_data*)

<i>simple_request (ask)</i>
CODE =get_query_data DATA =EnvelopDataToRecipients(Base64 (get_query_data (ask))) ECP =RawSignData(VER+'#' +CODE+'#' +NUM +'#' +FROM +'#' +TO+'#' +DATA)
<i>get_query_data (ask)</i>
EXTERNAL_ID =888
...

Малюнок 2 Схематичне зображення запиту від Запитувача

<i>simple_request (answer)</i>
DATA =EnvelopDataToRecipients(Base64 (get_query_data (answer.xml))) ECP =RawSignData(VER+'#' +CODE+'#' +NUM +'#' +FROM +'#' +TO+'#' +DATA)
<i>get_query_data (answer)</i>
REGID =123456
STATUS_REQUEST =R
EXTERNAL_ID =888
...

Малюнок 3 Схематичне зображення відповіді ПФУ

4. Прикладні запити

4.1. Сервіс «Запит на отримання статусу опрацювання прикладного запиту» (get_status_app)

Призначений для отримання статусу опрацювання запитів прикладних процесів. Запит формується в xml структурі get_status_app (ask). Результатом запиту є xml визначеної в get_status_app (answer) структури.

Zanum - get_status_app (ask)

Опис атрибутів даних запиту:

- REGID – ідентифікатор запиту в прикладній системі ПФУ.

Приклад get_status_app (ask)

```
<?xml version="1.0" encoding="UTF-8"?>
<FILTER>
  <REGID>123456</REGID>
</FILTER>
```

Результат запиту - get_status_app (answer)

Опис атрибутів даних відповіді:

- REGID – ідентифікатор запиту в прикладній системі ПФУ.
- STATUS_REQUEST – статус опрацювання запиту прикладними системами ПФУ (можливі значення: R - зареєстровано; V - оброблено; S - надано відповідь; E - помилка);
- STATUS_DATA – дата встановлення статусу (формат: DDMMYYYY);
- EXTERNAL_ID - ідентифікатор запиту Запитувача;
- ERRMESSAGE – інформаційне сповіщення.

Приклад get_status_app (answer)

```
<?xml version="1.0" encoding="UTF-8"?>
<REQUESTSTATUS>
  <REGID>123456</REGID>
  <STATUS_REQUEST>S</STATUS_REQUEST>
  <STATUS_DATA>18052017035501</STATUS_DATA>
  <EXTERNAL_ID>888</EXTERNAL_ID>
  <ERRMESSAGE></ERRMESSAGE>
</REQUESTSTATUS>
```

4.2. Сервіс «Запит переліку підготовлених запитів» (get_ask_query_list)

Призначений для передачі запиту на формування списку підготовлених запитів. Запит формується в xml визначеному в get_ask_query_list (ask) структури. Результатом запиту є xml визначеної в get_ask_query_list (answer) структури.

Результат обробки запиту прикладною системою необхідно отримувати за допомогою сервісних запитів get_request_status та get_request_data

Запит - get_ask_query_list (ask)

Структура get_ask_query_list (ask)

- EXTERNAL_ID – ідентифікатор запиту Запитувача
- EXTERNAL_DATE – дата формування пакету Запитувача
- DATE_START – дата, початку періоду, у форматі DDMMYYYY
- DATE_STOP – дата, завершення періоду, у форматі DDMMYYYY

Результат запиту - get_ask_query_list (answer)

Структура get_ask_query_list (answer)

№ п/п	Назва поля	Код поля	Тип та значність	Обов'язкове поле (Так, Ні)	Примітка
Загальні атрибути файлу відповіді					
1.	Ідентифікатор пакету Запитувача	EXTERNAL_ID	NUMBER(14)	Так	Значення з get_ask_query_list (ask)
2.	Дата відповіді Провайдера	ANSWER_DATE	DATETIME	Так	
Атрибути переліку звітів					
3.	Ідентифікатор запиту	REGID	NUMBER(14)	Так	
4.	Дата формування запиту	CREATE_DATE	DATETIME	Так	у форматі DDMMYYYYHHMMS S (дата знаходиться в періоді зазначеному в запиті між DATE_START та DATE_STOP)
5.	Поточний статус опрацювання запиту прикладними системами ПФУ	QUERY_ST	VARCHAR2(1)	Так	можливі значення: R – запит сформовано S - запит було отримано Провайдером
6.	Код запиту	QUERY_CODE	VARCHAR2(3)	Так	перелік запитів в Додатку Додаток 2
7.	Початок періоду запиту	BEGIN_DATE	DATE	Так	у форматі DDMMYYYY
8.	Завершення періоду запиту	END_DATE	DATE	Так	у форматі DDMMYYYY

4.3. Сервіс «Отримання запиту» (get_query_data)

Призначений для отримання сформованого запиту за ідентифікатором REGID. Запит формується в xml визначеному в get_query_data (ask) структури. Результатом запиту є xml визначеної в get_query_data (answer) структури

Результат обробки запиту прикладною системою необхідно отримувати за допомогою сервісних запитів get_request_status та get_request_data

Запит get_query_data (ask)

Структура get_query_data (ask)

- REGID - ідентифікатор запиту в системі ПФУ.

Приклад get_query_data (ask).xml

```
<?xml version="1.0" encoding="UTF-8"?>
<FILTER>
  <REGID>123456</REGID>
</FILTER>
```

Результат запиту - get_query_data (answer)

Структура get_query_data (answer)

№ п/п	Назва поля	Код поля	Тип та значність	Обов'язкове поле (Так, Ні)	Примітка
Загальні атрибути відповіді					
1.	Ідентифікатор пакету Запитувача	EXTERNAL_ID	NUMBER(14)	Так	
2.	Ідентифікатор відповіді Провайдера	REGID	NUMBER(14)	Так	Значення з get_ask_query_list
3.	Дата відповіді Провайдера	ANSWER_DATE	DATETIME	Так	Якщо запит отримується повторно, то дата останньої відправки
Атрибути звітів					
4.	Дата формування запиту	CREATE_DATE	DATETIME	Так	
5.	Код запиту	QUERY_CODE	VARCHAR2(3)	Так	перелік запитів в Додаток 2
6.	Початок періоду запиту	BEGIN_DATE	DATE	Так	у форматі DDMMYYYY
7.	Завершення періоду запиту	END_DATE	DATE	Так	у форматі DDMMYYYY
8.	Назва файлу-запиту	QUERY_NAME	VARCHAR2	Так	
9.	Зміст запиту	QUERY_DATA	VARCHAR2	Так	base64(zip(файл-запит)) (формати звітів наведено в Додаток 3)

4.4. Сервіс «Надання відповіді» (post_response)

Призначений для передачі звіту на запит. Запит формується в xml визначеному в post_response (ask) структури. Результатом запиту є xml визначеної в post_response (answer) структури

Результат обробки запиту прикладною системою необхідно отримувати за допомогою сервісних запитів get_request_status та get_request_data

Запит post_response (ask)

Структура post_response (ask)

№ п/п	Назва поля	Код поля	Тип та значність	Обов'язкове поле (Так, Ні)	Примітка
Загальні атрибути звіту					
10.	Ідентифікатор Запитувача	EXTERNAL_ID	NUMBER(14)	Так	
11.	Ідентифікатор запиту Провайдера	REGID	NUMBER(14)	Так	Значення з get_query_data, або «-1» якщо звіт сформовано за ініціативою Запитувача
12.	Дата звіту Запитувача	EXTERNAL_DATE	DATETIME	Так	Дата формування звіту Запитувача
Атрибути звіту					
13.	Код звіту	RESPONSE_CODE	VARCHAR2(3)	Так	перелік звітів в Додаток 2
14.	Початок періоду звіту	BEGIN_DATE	DATE	Так	у форматі DDMMYYYY, значення з get_query_data, або
15.	Завершення періоду звіту	END_DATE	DATE	Так	визначені Запитувачем, якщо звіт сформовано за ініціативою Запитувача
16.	Назва файлу-звіту	RESPONSE_NAME	VARCHAR2	Так	
17.	Зміст запиту	RESPONSE_DATA	VARCHAR2	Так	base64(zip(файл-звіту)) (формати звітів наведено в Додаток 3)

Результат запиту - post_response (answer)

Структура post_response (answer)

- REGID - ідентифікатор запиту-звіту в прикладній системі ПФУ

- STATUS_REQUEST – статус опрацювання запиту прикладними системами ПФУ (можливі значення: R - зареєстровано; V - оброблено; S - надано відповідь; E - помилка);
- STATUS_DATA – дата встановлення статусу (формат: DDMMYYYY);
- EXTERNAL_ID - ідентифікатор запиту Запитувача;
- ERRMESSAGE – інформаційне повідомлення.

Довідники*Таблиця 1. Типи запитів/звітів*

№ з/п	Код запити (QUERY_CODE)	Зміст	Код звіту (RESPONSE_CODE)
1	A	Отримання інформації про характеристику житла, про різновид використання житлово-комунальних послуг та розмір внесків\тарифів, а також про зміну особових рахунків та борги	B

Формат та структура електронного обміну

№ п/п	Найменування поля	Тип поля	Довжина поля	Коментар
1	ID	NUMERIC	7	Ідентифікатор рядка
	OZN	NUMERIC	1	Ознака первинний запит чи підтвердження (Можливі значення: 0- первинний; 1 – звірка)
2	CDPR	NUMERIC	12	Код за ЄДРПОУ організації
3	NCARD	NUMERIC	7	Номер картки пільговика
4	OSOB	CHARACTER	25	Номер особового рахунку в організації надавача (згідно з Договором або з квитанції)
5	EIS	CHARACTER	16	Код енергетичної ідентифікації суб'єкта ринку електричної енергії та/або точки комерційного обліку
6	FIO	CHARACTER	50	ПІБ пільговика / субсидіанта
7	IDPIL	CHARACTER	10	Ідентифікаційний номер пільговика / субсидіанта
8	INDEX	NUMERIC	6	Код індексу
9	CDUL	NUMERIC	14	Код вулиці
10	UL	CHARACTER	50	Назва вулиці
11	HOUSE	CHARACTER	7	Номер будинку
12	BLILD	CHARACTER	2	Номер корпусу

13	APT	CHARACTER	4	Номер квартири
Блок характеристики домогосподарства.(заповнюється організаціями надавачами\ виробниками комунальних послуг а також тими, що здійснюються управління багатоквартирним будинком, витрати на управління багатоквартирним будинком, у якому створено об'єднання співвласників багатоквартирного будинку/житлово-будівельний кооператив)				
14	PLZAG	NUMERIC	6.2	Загальна площа приміщення
15	PLOPAL	NUMERIC	6.2	Опалювальна площа приміщення
16	ZABORG	NUMERIC	6.4	Сума заборгованості по організації, грн.
Масив послуг (коди послуг згідно універсального довідника послуг)				
17	COD	NUMERIC	3.2	Код послуги (згідно Додатку 4)
18	PIDTVERD	NUMERIC	1	Підтвердження про наявність послуги (1 - ТАК/0 - НІ)
19	TARIF	NUMERIC	6.4	Тариф, грн.

Додаток 4

Уніфікований довідник житлово-комунальних послуг на які установлені соціальні норми, в межах яких держава надає громадянам житлові субсидії та пільги за використання житлово-комунальних послуг

Код послуги	Назва послуги
101	Житлова послуга - послуга з управління багатоквартирним будинком; витрати на управління багатоквартирним будинком, у якому створено об'єднання співвласників багатоквартирного будинку/житлово-будівельний кооператив
102	Послуга з постачання природного газу:
102.1	на опалення
102.2	на газову плиту (приготування їжі)
102.3	на газову плиту за відсутності гарячого водопостачання і газового водонагрівача (приготування їжі і підігрів води)
102.4	на газову плиту і водонагрівач
103	Послуга з розподілу природного газу:
103.1	на опалення
103.2	на газову плиту (приготування їжі)
103.3	на газову плиту за відсутності гарячого водопостачання і газового водонагрівача (приготування їжі і підігрів води)
103.4	на газову плиту і водонагрівач
104	Послуга з централізованого водопостачання
104.1	за наявності централіз. постачання гарячої води
104.2	за відсутності централіз. постачання гарячої
105	Послуга з постачання гарячої води

106	Послуга з централізованого водовідведення
107	Централізоване теплопостачання
107.1	За постачання теплової енергії
107.2	За приєднане теплове навантаження
118	Оплата витрат на опалення місць загального користування
108	Послуга з постачання електричної енергії:
108.1	на опалення
108.2	у будинках, не обладнаних стаціонарними електроплитами, за наявності централізованого постачання холодної та гарячої води
108.3	у будинках, не обладнаних стаціонарними електроплитами за наявності централізованого постачання холодної води та за відсутності централізованого постачання гарячої води і газових водонагрівальних приладів
108.4	у будинках, обладнаних стаціонарними електроплитами, за наявності централізованого постачання гарячої води
108.5	у будинках, обладнаних стаціонарними електроплитами, за відсутності централізованого постачання гарячої води
109	Послуга з розподілу електричної енергії:
109.1	на опалення
109.2	у будинках, обладнаних стаціонарними електроплитами, за наявності централізованого постачання холодної та гарячої води
109.3	у будинках, не обладнаних стаціонарними електроплитами за наявності централізованого постачання холодної води та за відсутності централізованого постачання гарячої води і газових водонагрівальних приладів
109.4	у будинках, обладнаних стаціонарними електроплитами, за наявності централізованого постачання гарячої води
109.5	у будинках, обладнаних стаціонарними електроплитами, за відсутності централізованого постачання гарячої води
110	Послуга з поводження з побутовими відходами (твердими, великогабаритними, ремонтними):

110.1	послуга з поводження з твердими відходами
110.2	послуга з поводження з великогабаритними відходами
110.3	послуга з поводження з ремонтними відходами
111	Послуга з поводження з побутовими відходами (рідкими) або вивезення рідких нечистот
112	Плата за абонентське обслуговування за послугою з централізованого водопостачання
113	Плата за абонентське обслуговування за послугою з централізованого водовідведення
114	Плата за абонентське обслуговування за послугою з постачання гарячої води
115	Плата за абонентське обслуговування за послугою з постачання теплової енергії
116	Плата за абонентське обслуговування за послугою з поводження з побутовими відходами