Приложение Е.1.

к Техническому заданию

№\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

от «\_\_\_» \_ \_\_\_\_\_\_\_\_\_\_ 2014 г.

Оглавление

[Общая информация 2](#_Toc393187234)

[***Название сервиса*** 2](#_Toc393187235)

[***Статус данной спецификации*** 2](#_Toc393187236)

[***Назначение сервиса*** 2](#_Toc393187237)

[Методы, предоставляемые сервисом 3](#_Toc393187238)

[***Получение списка муниципалитетов*** 3](#_Toc393187239)

[***Получение списка отделений МФЦ*** 4](#_Toc393187240)

[***Получение списка услуг*** 5](#_Toc393187241)

[***Получение списка категорий услуг*** 6](#_Toc393187242)

[***Получение списка ведомств*** 7](#_Toc393187243)

[***Получение списка документов*** 8](#_Toc393187244)

[***Получение дерева жизненных ситуаций*** 9](#_Toc393187245)

# Общая информация

***Название сервиса***

Сервис управления справочниками.

***Статус данной спецификации***

Данная спецификация носит рекомендательный характер и может быть изменена и/или дополнена в результате согласования с уполномоченным представителем второй стороны, участвующей в разработке или принятии работ по разработке сервиса согласно данной спецификации.

***Назначение сервиса***

Предоставление методов создания, чтения, изменения и удаления справочной информации. Под справочной информации в данной спецификации подразумеваются следующие данные:

1. Список муниципалитетов Пермского края
2. Список филиалов пермского краевого МФЦ с привязкой к муниципалитету
3. Список услуг, оказываемых пермским краевым МФЦ с возможностью группировки по
   1. Ведомствам
   2. Категориям
   3. Жизненным ситуациям
4. Документы
   1. Нормативные
      1. Законы
      2. Регламенты
      3. Соглашения
   2. Бланки документов, необходимых для получения услуг с привязкой к услуге
5. Расписание работы филиалов
   1. Долгосрочное расписание
   2. Расписание на каждый день с учетом занятого и свободного времени для предварительной записи

# Методы, предоставляемые сервисом

***Получение списка муниципалитетов***

array getMunicipalities()

Метод получает список муниципальных районов и метаданные о каждом из них.

***Атрибуты:***

Нет.

***Результат:***

Массив, содержащий информацию о муниципальных районах.

|  |  |  |
| --- | --- | --- |
| ***Название*** | ***Описание*** | ***Тип (длина)*** |
| id | Уникальный идентификатор муниципального района | Целое число |
| name | Название муниципального района | Строка |
| metadata | Объект, содержащий метаданные о муниципальном районе | Объект |
| metadata.coords | Массив, содержащий 2 элемента — широту и долготу административного центра муниципального района | Массив |

***Получение списка отделений МФЦ***

array getMFCDepartments()

Метод получает список отделений МФЦ с привязкой к муниципальному району.

***Атрибуты:***

Нет.

***Результат:***

Объект, содержащий информацию о филиалах МФЦ.

|  |  |  |
| --- | --- | --- |
| ***Название*** | ***Описание*** | ***Тип (длина)*** |
| id | Уникальный идентификатор филиала МФЦ | Целое число |
| municipalityId | Идентификатора муниципального района, к которому принадлежит филиал | Целое число |
| name | Название филиала | Строка |
| metadata | Объект, содержащий метаданные о филиале | Объект |
| metadata.coords | Массив, содержащий 2 элемента — широту и долготу филиала. | Массив |
| metadata.address | Адрес филиала | Строка |
| metadata.chief | Руководитель филиала | Строка |
| metadata.phone | Объект, содержащий информацию о телефоне | Объект |
| metadata.phone.main | Основной номер телефона филиала | Строка |
| metadata.phone.add | Добавочный номер | Строка |
| metadata.capacity | Количество окон в филиале | Целое число |
| metadata.openingDate | Дата открытия филиала | Unix timestamp |
| metadata.schedule | Массив, содержащий расписание работы филиала на каждый день с сегодняшней даты на 2 месяца вперед с учетом предзаписи. | Массив |

***Получение списка услуг***

array getServices()

Метод получает список государственных и муниципальных услуг, оказываемых МФЦ.

***Атрибуты:***

Нет.

***Результат:***

Массив, содержащий информацию об услугах.

|  |  |  |
| --- | --- | --- |
| ***Название*** | ***Описание*** | ***Тип (длина)*** |
| id | Уникальный идентификатор услуги | Целое число |
| categoryId | Идентификатор категории, к которой относится услуга | Целое число |
| departmentId | Идентификатор ведомства, которым оказывается услуга | Целое число |
| name | Объект, содержащий информацию о наименовании услуги | Объект |
| name.simple | Народное наименование услуги | Строка |
| name.official | Официальное наименование услуги | Строка |
| metadata | Объект, содержащий метаданные об услуге | Объект |
| metadata.result | Описание результата оказания услуги | Строка |
| metadata.recipient | Тип получателя, 0 — физическое лицо, 1 — юридическое лицо | Бинарный |
| metadata.price | Стоимость оказания услуги | Строка |
| metadata.term | Срок оказания услуги | Строка |
| metadata.documents | Массив, содержащий информацию о требуемых для получения услуги документов | Массив |
| metadata.documents[].id | Идентификатор документа | Целое число |
| metadata.documents[].type | Тип документа, 0 — подлинник, 1 — копия, 2 — нотариальная копия | Бинарный |
| metadata.documents[].count | Требуемое количество экземпляров данного документа | Целое число |

***Получение списка категорий услуг***

array getServiceCategories()

Метод получает дерево категорий услуг.

***Атрибуты:***

Нет.

***Результат:***

Массив, содержащий дерево категорий услуг. Элементы categories массива могут иметь произвольную длину в зависимости от сложности структуры категорий.

|  |  |  |
| --- | --- | --- |
| ***Название*** | ***Описание*** | ***Тип (длина)*** |
| id | Уникальный идентификатор категории | Целое число |
| name | Название категории | Строка |
| categories | Массив, содержащий информацию о подкатегориях данной категории | Массив |

***Пример возвращаемых данных:***

[

{

"id": 1,

"name": "Гражданство",

"categories": [

{

"id": 101,

"name": "Получение паспорта"

}

]

},

{

"id": 2,

"name": "Здравоохранение",

"categories": [

{

"id": 201,

"name": "Лицензирование, присвоение категории, выплаты"

},

{

"id": 202,

"name": "Медицинская помощь"

}

]

}

]

В данном примере приведен массив данных в формате JSON для следующего дерева категорий услуг:

1. Гражданство
   1. Получение паспорта
2. Здравоохранение
   1. Лицензирование, присвоение категории, выплаты
   2. Медицинская помощь

***Получение списка ведомств***

array getDepartments()

Метод получает список ведомств.

***Атрибуты:***

Нет.

***Результат:***

Массив, содержащий информацию о муниципальных районах.

|  |  |  |
| --- | --- | --- |
| ***Название*** | ***Описание*** | ***Тип (длина)*** |
| id | Уникальный идентификатор ведомства | Целое число |
| name | Название ведомства | Строка |

***Получение списка документов***

array getDocuments()

Метод получает список документов.

***Атрибуты:***

Нет.

***Результат:***

Массив, содержащий информацию о документах.

|  |  |  |
| --- | --- | --- |
| ***Название*** | ***Описание*** | ***Тип (длина)*** |
| id | Уникальный идентификатор документа | Целое число |
| name | Название документа | Строка |
| type | Тип документа | Перечислимый |

***Получение дерева жизненных ситуаций***

object getSituationTree()

Метод получает дерево принятия решений на основе жизненных ситуаций.

***Атрибуты:***

Нет.

***Результат:***

Массив, содержащий дерево жизненных ситуаций. Элементы options массива могут иметь произвольную длину в зависимости от сложности принятия решения.

|  |  |  |
| --- | --- | --- |
| ***Название*** | ***Описание*** | ***Тип (длина)*** |
| id | Уникальный идентификатор жизненной ситуации или услуги | Целое число |
| question | Вопрос, задаваемый пользователю для перехода на следующий уровень дерева. Если атрибут не указан, данный уровень является последним и содержит список услуг | Строка |
| options | Массив, содержащий варианты ответа на вопрос | Строка |

***Пример возвращаемых данных:***

[

{

"id": 1,

"name": "Социальное обеспечение, семья",

"question": "Вам необходимо получить информацию об услугах фонда социального страхования?",

"options": [

{

"id": 2,

"option": "Мне необходимо произвести регистрацию или сняться с регистрационного учета",

"question": "Выберите подходящую услугу для регистрации или снятия с регистрационного учета",

"options": [

{ "id": 10001 },

{ "id": 10002 },

{ "id": 10003 },

{ "id": 10004 }

]

},

{

"id": 4,

"option": "Я должен сдать отчет",

"question": "Выберите отчет, который вы хотите сдать",

"options": [

{ "id": 30001 },

{ "id": 30002 }

]

}

]

}

]

В данном примере приведен массив данных в формате JSON для следующего дерева принятия решений:

1. **Q:** Вам необходимо получить информацию об услугах фонда социального страхования?
   1. **A:** Мне необходимо произвести регистрацию или сняться с регистрационного учета
      1. **Q:** Выберите подходящую услугу для регистрации или снятия с регистрационного учета
         1. **A:** 10001 (идентификаторы конечных услуг )
         2. **A:** 10002
         3. **A:** 10003
         4. **A:** 10004
   2. **A:** Я должен сдать отчет
      1. **Q:** Выберите отчет, который вы хотите сдать
         1. **A:** 30001 (идентификаторы конечных услуг)
         2. **A:** 30001