Сервис учета продуктов на кухне.

Макет.

**Продукты. Кухня**

Логин:

Пароль:

**Продукты. Кухня**

Продукты в наличии:

Молоко

Название:

Производитель:

Количество:

Ед.изм.:

Срок годности:

Место хранения:

**Продукты. Кухня**

Добавление продукта:

Название:

Производитель:

Количество:

Ед.изм.:

Срок годности:

Место хранения:

**Продукты. Кухня**

Удалить продукт:

Молоко

WS3

Проверка запасов в WS. Ответ:

{ "id": "namber",

"name\_product": "varchar(100)",

“storage\_in”: “Kitchen cabinet” ,

“unit”: "XX."

}

Удаление продукта. Ответ

{“Продукт удален”}

Удаление продукта

DELETE {{http\_schema}}://{{host}}/delete \_object/

Создание/добавление/обновление продукта

POST {{http\_schema}}://{{host\_ws\_1}}/creating\_object/

**POST {{http\_schema}}://{{host\_ws\_2}}/creating\_object/**

POST {{http\_schema}}://{{host\_ws\_3}}/creating\_object/

Выбор WS выполняется в соответствии с параметром **“storage\_in”**

WS2

Проверка запасов в WS. Ответ:

{ "id": "namber",

"name\_product": "varchar(100)",

“storage\_in”: "Freezer compartment",

“unit”: "XX."

}

Создание/добавление/обновление продукта Ответ:

{ "id": "namber",

"name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "ХХ",

"expiration date": "2010-10-10",

"storage\_in": "Fridge"

}

Функционал (Создание/добавление/обновление) продукта выбирается по результатам проверки параметров в БД

Удаление объекта

DELETE {{http\_schema}}://{{host}}/delete \_object/

Body { "**id**": "number",

"**storage\_in": "Kitchen cabinet"**

}

Обновление продукта

PATCH {{http\_schema}}://{{host}}/creating \_object/

Body { "**name\_product**": "varchar(100)",

"**maker**": "varchar(50)",

"quantity": "namber",

"**unit**": "гр.", "кг.", "л.", "мл.", "шт.", "уп."

"**expiration date**": "2010-10-10",

"**storage\_in**": "Fridge", "Freezer compartment", "Kitchen cabinet"

}

Добавление объемов продукта

PATCH {{http\_schema}}://{{host}}/creating \_object/

Body { "**name\_product**": "varchar(100)",

"**maker**": "varchar(50)",

"quantity": "namber",

"**unit**": "гр.", "кг.", "л.", "мл.", "шт.", "уп."

"**expiration date**": "2010-10-10",

"**storage\_in**": "Fridge", "Freezer compartment", "Kitchen cabinet"

}

Создание продукта

POST {{http\_schema}}://{{host}}/creating\_object/

Body { "name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "гр.", "кг.", "л.", "мл.", "шт.", "уп."

"expiration date": "2010-10-10",

"storage\_in": "Fridge", "**Freezer compartment**", "Kitchen cabinet"

}

Проверка запасов для выпадающего меню

GET {{http\_schema}}://{{host}}/creat\_object

Проверка запасов выбранного продукта

GET {{http\_schema}}://{{host}}/creat\_object/get\_method?ID=1 , storage\_in="Fridge"

WS1

Проверка запасов в WS. Ответ:

{ "id": "namber",

"name\_product": "varchar(100)",

“storage\_in”: "Fridge",

“unit”: "XX."

}

Проверка запаса в WS. Ответ:

{ "id": "namber",

"name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "ХХ",

"expiration date": "2010-10-10",

"storage\_in": "Fridge"

}

Проверка запаса в WS

{{http\_schema}}://{{host}}/creat\_object/get\_method?ID=1 , storage\_in="Fridge"

Проверка запасов в WS

GET {{http\_schema}}://{{host\_ws1}}/creat\_object

GET {{http\_schema}}://{{host\_ws2}}/creat\_object

GET {{http\_schema}}://{{host\_ws3}}/creat\_object

Object

id: auto\_increment primary key

name\_product: varchar(100)

maker: varchar(50)

quantity: "namber"

unit: "гр."/"кг."/"л."/"мл."/"шт."/"уп."

expiration date: "date”

storage\_in: "Fridge", "Freezer compartment", "Kitchen cabinet"

User

user\_id: auto\_increment primary key

login: varchar(20)

password: varchar(20)

Ответ

{token: “token” }

При успешной валидации с БД

Авторизация

POST {{http\_schema}}://{{host}}/users/obtain-token

Body { "username": " varchar(20)",

"password": "varchar(20)",

}

БД

WS

Proxy

Прокси-сервер производит запросы на WEB-сервисы в соответствии с местом размещения продуктов:

WS1 - "Fridge",

WS2 - "Freezer compartment",

WS3 - "Kitchen cabinet".

Авторизация:

Authorization: Token {{token}}', где {{token}} - это токен

Получение токена осуществляется через метод /users/obtain-token

Запрос - POST-запрос с инфой: "username" -- логин пользователя "password" — пароль

Валидация учетных данных в БД “User”

В БД созданы User\_1(12345) и User\_2(4321)

Каждый запрос, как ожидает, так и отвечает application/json

ПРОВЕРКА ЗАПАСОВ

Для отображения в выпадающем меню проводится запрос названий и ID имеющихся позиций в БД

GET {{http\_schema}}://{{host}}/creat\_object

Proxy пересылает запрос

* {{http\_schema}}://{{host\_ws1}}/creat\_object
* {{http\_schema}}://{{host\_ws2}}/creat\_object
* {{http\_schema}}://{{host\_ws3}}/creat\_object

В ответе: { "id": "namber",

"name\_product": "varchar(100)",

"storage\_in": "XXX",

"unit": "XXX"

}

При выборе определенного названия из выпадающего меню идет запрос с включенным ID и “storage\_in”

GET {{http\_schema}}://{{host}}/creat\_object/get\_method?ID= , storage\_in=

В ответе: { "id": "namber",

"name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "гр."/"кг."/"л."/"мл."/"шт."/"уп.",

"expiration date": "2010-10-10",

"storage\_in": "Fridge", "Freezer compartment", "Kitchen cabinet" }

Проверка БД при обращении к ней:

Если "expiration date" <= "actual date" - open pop-up window "the expiration date has ended.

Вложить тело ответа по ID".

Кнопка «ОК» скрывает сообщение.

Кнопка «Изъято» формирует запрос DELETE с ID в теле. (см.ниже)

СОЗДАНИЕ ОБЪЕКТА

POST {{http\_schema}}://{{host}}/creating\_object/

Body { "name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "гр.", "кг.", "л.", "мл.", "шт.", "уп."

"expiration date": "2010-10-10",

"storage\_in": "Fridge", "Freezer compartment", "Kitchen cabinet"

}

Если "storage\_in": "Fridge": {{http\_schema}}://{{host\_ws1}}/creat\_object/

Если "storage\_in": "Freezer compartment" {{http\_schema}}://{{host\_ws2}}/creat\_object/ Если "storage\_in": "Kitchen cabinet" {{http\_schema}}://{{host\_ws3}}/creat\_object/

В ответе: { "id": "namber",

"name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "гр."/"кг."/"л."/"мл."/"шт."/"уп.",

"expiration date": "2010-10-10",

"storage\_in": "Fridge", "Freezer compartment", "Kitchen cabinet"

}

Добавить ID в список ежедневной проверки по "expiration date"

ДОБАВЛЕНИЕ ОБЪЕМОВ ОБЪЕКТА

Если полное соответствие параметров "name\_product", "maker", "unit", "storage\_in", "expiration date" и "quantity" >= 0 суммируем "quantity" к объему в БД

PATCH {{http\_schema}}://{{host\_ws1}}/creating\_object/{{id}}

Необходимые параметры: "quantity" и "expiration date"

В ответе получаем: { "id": "namber",

"name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "гр."/"кг."/"л."/"мл."/"шт."/"уп."

"expiration date": "2010-10-10",

"storage\_in": "Fridge", "Freezer compartment", "Kitchen cabinet"

}

ОБНОВЛЕНИЕ ОБЪЕКТА

Если полное соответствие параметров "name\_product", "maker", "unit", "storage\_in" и "quantity" = 0 обновляем "quantity" и "expiration date"

PATCH {{http\_schema}}://{{host\_ws1}}/creating\_object/{{id}}

Необходимые параметры: "quantity" и "expiration date"

В ответе получаем: { "id": "namber",

"name\_product": "varchar(100)",

"maker": "varchar(50)",

"quantity": "namber",

"unit": "гр."/"кг."/"л."/"мл."/"шт."/"уп."

"expiration date": "2010-10-10",

"storage\_in": "Fridge", "Freezer compartment", "Kitchen cabinet"

}

УДАЛЕНИЕ ОБЪЕКТА

DELETE {{http\_schema}}://{{host\_ws1}}/delete\_object/

Необходимые параметры: "id" и “storage\_in”

В ответе получаем: pop-up window «Удален»