

Mobilne systemy webowe

Sebastian Łasisz

Dominik Janusiewicz

31 maja 2016

1 ReShP

Aplikacja pozwala użytkownikowi na tworzenie dowolnych przepisów. Dane przepisy będą mogły być prywatne bądź publiczne. Wszystkie przepisy publiczne będą dostępne w API które zostanie udostępnione. Ponadto użytkownik będzie mógł dodawać własne listy zakupowe oraz własne listy produktów. Będą one zawierały najważniejszą informację, którą użytkownik będzie miał możliwość posiadać podczas przygotowywania posiłków. Dodatkowo użytkownik będzie mógł ustalać i planować własne posiłki. Ponadto dzięki integracji z zewnętrznymi serwisami użytkownik będzie miał dostęp do szczegółowej analizy składników oraz będzie mógł dodać swoje posiłki do kalendarza dzięki czemu zyska lepszą kontrolę czasu jak i dostęp do powiadomień. Przykładowe zapytania do API obejmują:

- Wszystkie publiczne przepisy
- Własne przepisy
- Własne listy zakupowe
- Własne listy posiadanych produktów
- Listę planowanych posiłków

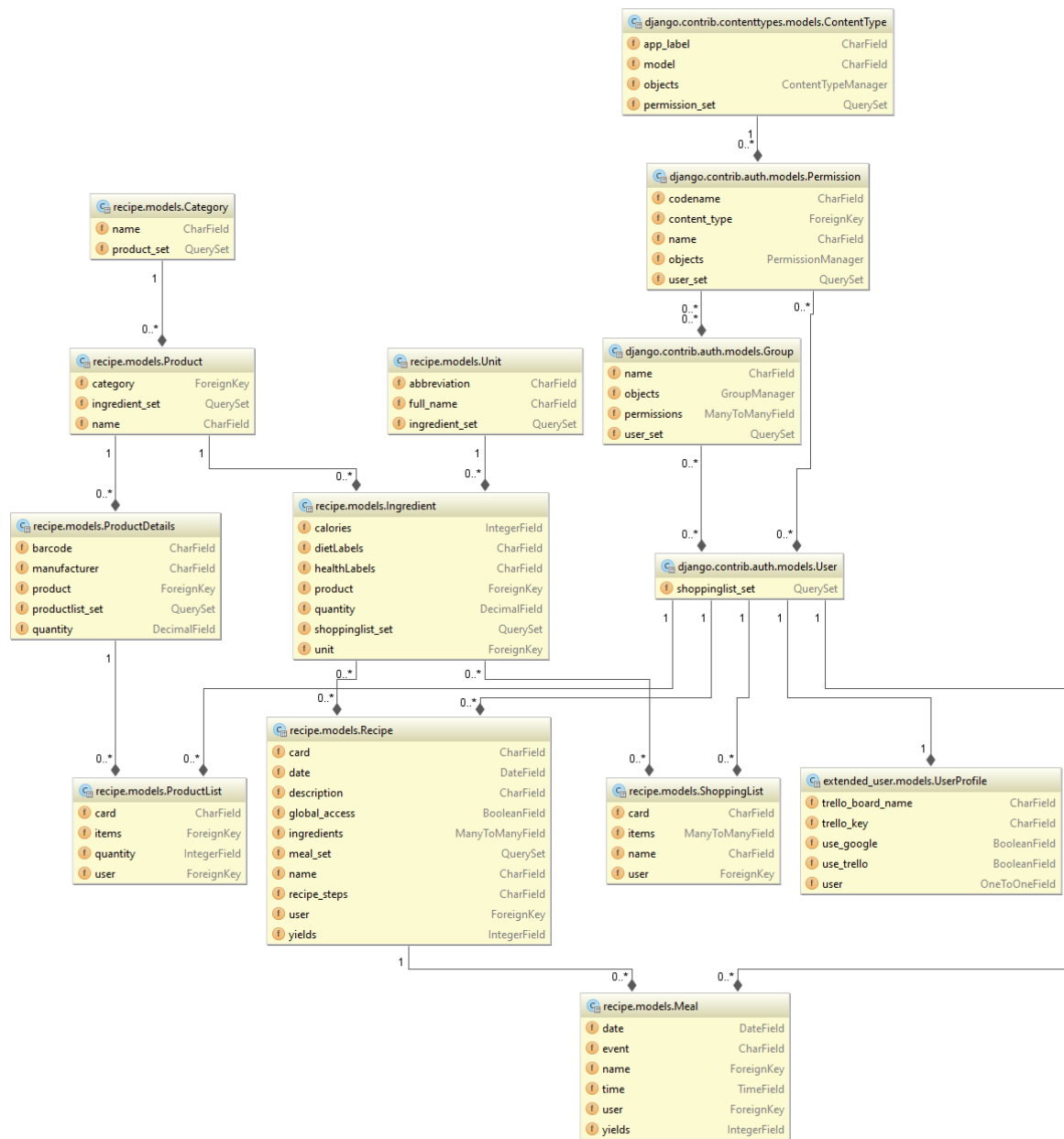
Ponadto wszystkie przepisy będą miały szczegółowe informacje o składnikach oraz o autorze danego przepisu. Do stworzenia aplikacji zostały wykorzystane następujące technologie:

- Python 2.7.11
- Django 1.9.4
- HTML 5
- CSS 3
- JavaScript

Ponadto do stworzenia aplikacji zostały wykorzystane następujące biblioteki:

- Twitter Bootstrap 3
- Django Summernote
- Django REST Swagger
- Django REST Framework

1.1 Diagramy



Powered by yFiles

Rysunek 1: Diagram klas

1.2 Opis wykorzystywanych serwisów zewnętrznych

Aplikacja wykorzystuje trzy zewnętrzne serwisy (Edaman, Trello, Google Calendar) do pobierania oraz udostępniania danych i informacji. Korzystanie z Edamam jest obligatoryjne, jednakże korzystanie z połączenia z Trello i Google Calendar jest dobrowolne i można zadeklarować chęć korzystania z tych serwisów poprzez odpowiednie ustawienia w profilu.

1.2.1 Edamam

Edamam jest to system, który umożliwia wyszukiwanie przepisów posiłków oraz umożliwia analizę składników w czasie rzeczywistym. W ramach działania serwisu zostały udostępnione cztery różne API:

- Recipe Analysis and Nutrition API
- Nutrition Data API
- Diet Recommendations API
- Recipe Search API

Przykład 1: Edaman API Example

```
{
  "title": "Fresh Ham Roasted With Rye Bread and Dried Fruit Stuffing",
  "prep": "1. Have your butcher bone and butterfly the ham and score the
          fat in a diamond pattern. ...",
  "yield": "About 15 servings",
  "ingr": [
    "1 fresh ham, about 18 pounds, prepared by your butcher (See Step 1)",
    "7 cloves garlic, minced",
    "1 tablespoon caraway seeds, crushed",
    "4 teaspoons salt",
    "Freshly ground pepper to taste",
    "1 teaspoon olive oil",
    "1 medium onion, peeled and chopped",
    "3 cups sourdough rye bread, cut into 1/2-inch cubes",
    "1 1/4 cups coarsely chopped pitted prunes",
    "1 1/4 cups coarsely chopped dried apricots",
    "1 large tart apple, peeled, cored and cut into 1/2-inch cubes",
    "2 teaspoons chopped fresh rosemary",
    "1 egg, lightly beaten",
    "1 cup chicken broth, homemade or low-sodium canned"
  ]
}
```

1.2.2 Trello

Trello jest to system, który umożliwia tworzenie tablic wypełnionych kartami. Każda karta może być notatką/listą zadań/etc... Trello API umożliwia tworzenie nowych tablic/notatek jak i dodawanie nowych, bądź modyfikowanie czy usuwanie starych. Przykładowe zapytanie zwrócenia informacji o karcie:

Przykład 2: Trello API Example

```
{
  "id": "4eea503d91e31d174600008f",
  "name": "Learn about the Trello API",
  "idList": "4eea4ffc91e31d174600004b"
}
```

1.2.3 Google Calendar API

Przykład 3: Google Calendar API Example

```
{
  "kind": "calendar#event",
  "etag": etag,
  "id": string,
  "status": string,
  "htmlLink": string,
  "created": datetime,
  "updated": datetime,
  "summary": string,
  "description": string,
  "location": string,
  "colorId": string,
  "creator": {
    "id": string,
    "email": string,
    "displayName": string,
    "self": boolean
  },
  "organizer": {
    "id": string,
    "email": string,
    "displayName": string,
    "self": boolean
  },
  "start": {
    "date": date,
    "dateTime": datetime,
    "timeZone": string
  },
  "end": {
    "date": date,
    "dateTime": datetime,
    "timeZone": string
  },
  "attendees": [
    {
      "id": string,
      "email": string,
      "displayName": string,
      "organizer": boolean,
      "self": boolean,
      "resource": boolean,
      "optional": boolean,
      "responseStatus": string,
      "comment": string,
      "additionalGuests": integer
    }
  ],
  ...
}
```

1.3 Komuniakcja z ReShP

Aplikacja będzie udostępniała dane za pomocą interfejsu REST, który umożliwi na komunikowanie się z pomiędzy serwisami. Następujące zapytania zostały udostępnione:

- POST /api-token-auth
- GET /api/own_recipe_list/
- GET /api/recipe_list/
- POST /api/add_recipe
- PUT /api/recipe/id/
- GET /api/recipe/id/
- DELETE /api/recipe/id/
- GET /api/shopping_lists/
- PUT /api/shopping_list/id/
- GET /api/shopping_list/id/
- DELETE /api/shopping_list/id/
- POST /api/add_shopping_list/
- GET /api/product_lists/
- PUT /api/product_lists/id/
- GET /api/product_list/id/
- DELETE /api/product_list/id/
- POST /api/product_list/id/
- GET /api/meals/
- POST /api/add_meal/
- PUT /api/meal/id/
- GET /api/meal/id/
- DELETE /api/meal/id/

1.3.1 Przykładowe zapytania

Przykład 4 ukazuje w jaki sposób można pobrać token do komunikowania się z ReShP. Aby otrzymać token wystarczy podać nazwę konta oraz hasło dostępu.

Przykład 4: Pobieranie tokenu

```
curl --data "username=admin&password=admin" http://127.0.0.1:8000/api-token-auth/
```

Przykład 5 ukazuje w jaki sposób można dodać posiłek do bazy danych wykorzystując REST. Wystarczy podać nazwę przepisu, ilość porcji oraz datę i godzinę.

Przykład 5: Dodawanie posiłku

```
curl -X POST -H "Content-Type: application/json" -d '{
  "name": "Chicken with fries",
  "yields": 4,
  "date": "2016-05-20",
  "time": "15:00"
}' http://127.0.0.1:8000/api/add_meal/ -H 'Authorization: Token 647
a398cc3918952b93a368593e9ee97472b294b '
```

Przykład 6 ukazuje w jaki sposób można dodać przepis do bazy danych wykorzystując REST. Wystarczy podać nazwę przepisu, krótkie podsumowanie, w jaki sposób przygotować posiłek, ilość porcji oraz jakie składniki zawiera.

Przykład 6: Dodawanie przepisu

```
curl -X POST -H "Content-Type: application/json" -d '{
  "name": "Chicken with fries",
  "description": "My very own first recipe. Chicken with fries. Very
    tasty. Much wow.",
  "recipe_steps": "1. First step, 2. Second step. 3. Third step",
  "yields": "4",
  "ingredients":
  [
    {"product": {"name": "Chicken", "category": {"name": "Red Meat"}},
      "quantity": "1", "unit": {"full_name": "kilogram", "
        abbreviation": "kg"}},
    {"product": {"name": "Potato", "category": {"name": "Vegetable"}},
      "quantity": "3", "unit": {"full_name": "kilogram", "
        abbreviation": "kg"}},
    {"product": {"name": "Tomato", "category": {"name": "Vegetable"}},
      "quantity": "2", "unit": {"full_name": "pieces", "abbreviation":
        "p"}}
  ]
}' http://127.0.0.1:8000/api/add_recipe/ -H 'Authorization: Token 647
a398cc3918952b93a368593e9ee97472b294b '
```

Przykład 7 ukazuje w jaki sposób można pobrać daną listę zakupową. Wystarczy znać unikalny numer listy zakupów, aby pobrać wszystkie potrzebne informacje.

Przykład 7: Pobieranie listy zakupów

```
curl -X GET http://127.0.0.1:8000/api/shopping_list/11/ -H 'Authorization:
Token 647a398cc3918952b93a368593e9ee97472b294b '
```

Przykład 8 ukazuje w jaki sposób można usunąć dany produkt z listy posiadanych produktów. Wystarczy znać unikalny numer produktu, aby go całkowicie usunąć z bazy danych.

Przykład 8: Usuwanie produktu

```
curl -X DELETE http://127.0.0.1:8000/api/product_list/1/ -H 'Authorization:
Token 647a398cc3918952b93a368593e9ee97472b294b '
```

1.3.2 Przykładowe wyniki

Do komunikacji z ReShP aplikacja zewnętrzna potrzebuje token, który będzie wykorzystywany jako autoryzacja. W celu pobrania tokenu użytkownik musi przesłać dane logowania, a w zamian otrzymuje token. Przykładowy token jest widoczny na Przykładzie 9 poniżej.

Przykład 9: Pobranie tokenu

```
{"token":"647a398cc3918952b93a368593e9ee97472b294b"}
```

W ramach korzystania z aplikacji użytkownik ma możliwość pobrania posiłku, który został przez niego wcześniej dodany. Posiłek zawiera dane o przepisie, który zostanie wykorzystane, liczbie porcji, dacie oraz godzinie o której planowany jest posiłek. Przykładowy wynik zapytania może wyglądać tak jak zostało to przedstawione na Przykładzie 10 poniżej.

Przykład 10: Pobranie posiłku

```
[{"name":{"name":"Chicken with fries","description":"My very own first recipe. Chicken with fries. Very tasty. Much wow.","recipe_steps":"1. First step, 2. Second step. 3. Third step","ingredients":[{"product":{"name":"Chicken","category":{"id":10,"name":"Red Meat"}}, {"quantity":1.00,"unit":{"full_name":"kilogram","abbreviation":"kg"},"calories":2150,"dietLabels":["LOW_CARB","LOW_SODIUM"],"healthLabels":["SUGAR_CONSCIOUS","LOW_SUGAR","KIDNEY_FRIENDLY","PALEO","SPECIFIC_CARBS","DAIRY_FREE","GLUTEN_FREE","WHEAT_FREE","EGG_FREE","MILK_FREE","PEANUT_FREE","TREE_NUT_FREE","SOY_FREE","FISH_FREE","SHELLFISH_FREE","PORK_FREE","RED_MEAT_FREE","CRUSTACEAN_FREE","CELERY_FREE","MUSTARD_FREE","SESAME_FREE","LUPINE_FREE","MOLLUSK_FREE","ALCOHOL_FREE","NO_OIL_ADDED","NO_SUGAR_ADDED","KOSHER"]}], [{"product":{"name":"Potato","category":{"id":11,"name":"Vegetable"}}, {"quantity":3.00,"unit":{"full_name":"kilogram","abbreviation":"kg"},"calories":2310,"dietLabels":["HIGH_FIBER","LOW_FAT","LOW_SODIUM"],"healthLabels":["FAT_FREE","LOW_FAT_ABS","SUGAR_CONSCIOUS","VEGAN","VEGETARIAN","PESCATARIAN","DAIRY_FREE","GLUTEN_FREE","WHEAT_FREE","EGG_FREE","MILK_FREE","PEANUT_FREE","TREE_NUT_FREE","SOY_FREE","FISH_FREE","SHELLFISH_FREE","PORK_FREE","RED_MEAT_FREE","CRUSTACEAN_FREE","CELERY_FREE","MUSTARD_FREE","SESAME_FREE","LUPINE_FREE","MOLLUSK_FREE","ALCOHOL_FREE","NO_OIL_ADDED","NO_SUGAR_ADDED","KOSHER"]}], [{"yields":4,"date":"2016-05-28"}], [{"yields":4,"date":"2016-05-28","time":"12:11:00"}]}
```

W ramach korzystania z aplikacji użytkownik ma możliwość pobrania przepisu, który został przez niego wcześniej dodany. Posiłek zawiera dane o składnikach, krótkie podsumowanie, opis wykonania, przewidzianą liczbę porcji oraz datę dodania. Przykładowy wynik zapytania może wyglądać tak jak zostało to przedstawione na Przykładzie 11 poniżej.

Przykład 11: Pobranie przepisu

```
[{"name":"Chicken with fries","description":"My very own first recipe. Chicken with fries. Very tasty. Much wow.","recipe_steps":"1. First step, 2. Second step. 3. Third step","ingredients":[{"product":{"name":"Chicken","category":{"id":10,"name":"Red Meat"}}, {"quantity":1.00,"unit":{"full_name":"kilogram","abbreviation":"kg"},"calories":2150,"dietLabels":["LOW_CARB","LOW_SODIUM"],"healthLabels":["SUGAR_CONSCIOUS","LOW_SUGAR","KIDNEY_FRIENDLY","PALEO","SPECIFIC_CARBS","DAIRY_FREE","GLUTEN_FREE","WHEAT_FREE","EGG_FREE","MILK_FREE","PEANUT_FREE","TREE_NUT_FREE","SOY_FREE","FISH_FREE","SHELLFISH_FREE","PORK_FREE","RED_MEAT_FREE","CRUSTACEAN_FREE","CELERY_FREE","MUSTARD_FREE","SESAME_FREE","LUPINE_FREE","MOLLUSK_FREE","ALCOHOL_FREE","NO_OIL_ADDED","NO_SUGAR_ADDED","KOSHER"]}], [{"product":{"name":"Potato","category":{"id":11,"name":"Vegetable"}}, {"quantity":3.00,"unit":{"full_name":"kilogram","abbreviation":"kg"},"calories":2310,"dietLabels":["HIGH_FIBER","LOW_FAT","LOW_SODIUM"],"healthLabels":["FAT_FREE","LOW_FAT_ABS","SUGAR_CONSCIOUS","VEGAN","VEGETARIAN","PESCATARIAN","DAIRY_FREE","GLUTEN_FREE","WHEAT_FREE","EGG_FREE","MILK_FREE","PEANUT_FREE","TREE_NUT_FREE","SOY_FREE","FISH_FREE","SHELLFISH_FREE","PORK_FREE","RED_MEAT_FREE","CRUSTACEAN_FREE","CELERY_FREE","MUSTARD_FREE","SESAME_FREE","LUPINE_FREE","MOLLUSK_FREE","ALCOHOL_FREE","NO_OIL_ADDED","NO_SUGAR_ADDED","KOSHER"]}], [{"yields":4,"date":"2016-05-28"}], [{"yields":4,"date":"2016-05-28","time":"12:11:00"}]}
```



```

kilogram", "abbreviation": "kg"}, "calories": 2310, "dietLabels": "[u'
HIGH_FIBER', u'LOW_FAT', u'LOW_SODIUM']", "healthLabels": "[u'FAT_FREE',
u'LOW_FAT_ABS', u'SUGAR_CONSCIOUS', u'VEGAN', u'VEGETARIAN', u'
PESCATARIAN', u'DAIRY_FREE', u'GLUTEN_FREE', u'WHEAT_FREE', u'EGG_FREE
', u'MILK_FREE', u'PEANUT_FREE', u'TREE_NUT_FREE', u'SOY_FREE', u'
FISH_FREE', u'SHELLFISH_FREE', u'PORK_FREE', u'RED_MEAT_FREE', u'
CRUSTACEAN_FREE', u'CELERY_FREE', u'MUSTARD_FREE', u'SESAME_FREE', u'
LUPINE_FREE', u'MOLLUSK_FREE', u'ALCOHOL_FREE', u'NO_OIL_ADDED', u'
NO_SUGAR_ADDED', u'KOSHER']"]], "yields": 4, "date": "2016-05-28"]

```

W ramach korzystania z aplikacji użytkownik ma możliwość pobrania listy zakupowej, która została przez niego wcześniej dodana. Lista zakupowa posiada szczegółowe informacje o pozycjach, które są do kupienia. Przykładowy wynik zapytania może wyglądać tak jak zostało to przedstawione na Przykładzie 12 poniżej.

Przykład 12: Pobranie listy zakupowej

```

[{"name": "My very own shopping list", "items": [{"product": {"name": "Chicken",
"category": {"id": 10, "name": "Red Meat"}}, "quantity": "1.00", "unit": {"full_name": "kilogram", "abbreviation": "kg"}, "calories": 2150, "dietLabels": "[u'LOW_CARB', u'LOW_SODIUM']", "healthLabels": "[u'SUGAR_CONSCIOUS', u'LOW_SUGAR', u'KIDNEY_FRIENDLY', u'PALEO', u'SPECIFIC_CARBS', u'DAIRY_FREE', u'GLUTEN_FREE', u'WHEAT_FREE', u'EGG_FREE', u'MILK_FREE', u'PEANUT_FREE', u'TREE_NUT_FREE', u'SOY_FREE', u'FISH_FREE', u'SHELLFISH_FREE', u'PORK_FREE', u'RED_MEAT_FREE', u'CRUSTACEAN_FREE', u'CELERY_FREE', u'MUSTARD_FREE', u'SESAME_FREE', u'LUPINE_FREE', u'MOLLUSK_FREE', u'ALCOHOL_FREE', u'NO_OIL_ADDED', u'NO_SUGAR_ADDED', u'KOSHER']"}, {"product": {"name": "Potato", "category": {"id": 11, "name": "Vegetable"}}, "quantity": "3.00", "unit": {"full_name": "kilogram", "abbreviation": "kg"}, "calories": 2310, "dietLabels": "[u'HIGH_FIBER', u'LOW_FAT', u'LOW_SODIUM']", "healthLabels": "[u'FAT_FREE', u'LOW_FAT_ABS', u'SUGAR_CONSCIOUS', u'VEGAN', u'VEGETARIAN', u'PESCATARIAN', u'DAIRY_FREE', u'GLUTEN_FREE', u'WHEAT_FREE', u'EGG_FREE', u'MILK_FREE', u'PEANUT_FREE', u'TREE_NUT_FREE', u'SOY_FREE', u'FISH_FREE', u'SHELLFISH_FREE', u'PORK_FREE', u'RED_MEAT_FREE', u'CRUSTACEAN_FREE', u'CELERY_FREE', u'MUSTARD_FREE', u'SESAME_FREE', u'LUPINE_FREE', u'MOLLUSK_FREE', u'ALCOHOL_FREE', u'NO_OIL_ADDED', u'NO_SUGAR_ADDED', u'KOSHER']"}, {"product": {"name": "Tomato", "category": {"id": 11, "name": "Vegetable"}}, "quantity": "2.00", "unit": {"full_name": "pieces", "abbreviation": "p"}, "calories": 163, "dietLabels": "[u'HIGH_FIBER', u'LOW_FAT', u'LOW_SODIUM']", "healthLabels": "[u'LOW_FAT_ABS', u'VEGAN', u'VEGETARIAN', u'PESCATARIAN', u'PALEO', u'SPECIFIC_CARBS', u'DAIRY_FREE', u'GLUTEN_FREE', u'WHEAT_FREE', u'EGG_FREE', u'MILK_FREE', u'PEANUT_FREE', u'TREE_NUT_FREE', u'SOY_FREE', u'FISH_FREE', u'SHELLFISH_FREE', u'PORK_FREE', u'RED_MEAT_FREE', u'CRUSTACEAN_FREE', u'CELERY_FREE', u'MUSTARD_FREE', u'SESAME_FREE', u'LUPINE_FREE', u'MOLLUSK_FREE', u'ALCOHOL_FREE', u'NO_OIL_ADDED', u'NO_SUGAR_ADDED', u'KOSHER']"}]}]

```

W ramach korzystania z aplikacji użytkownik ma możliwość pobrania produktów, które wcześniej dodał. Produkt zawiera informacje o ilości, nazwie producenta, ilości w opakowaniu oraz o kodzie kreskowym. Przykładowy wynik zapytania może wyglądać tak jak zostało to przedstawione na Przykładzie 13 poniżej.

Przykład 13: Pobranie produktów

```
[{"quantity":5,"items":{"product":{"name":"Tomato","category":{"id":7,"name":"{u'name': u'meat'}"}}, "barcode":"test_code","manufacturer":"Polska", "quantity":"5.00"}]}
```

2 Foodster

Aplikacja mobilna jest częściowym odwzorowaniem aplikacji webowej „ReShP”. Pozwala na przeglądanie przepisów oraz list zakupowych zalogowanego użytkownika. Może on również stworzyć nową listę zakupową, lub edytować wcześniej zdefiniowaną. Aplikacja została jednak okrojona o możliwość dodawania/edycji przepisów, ze względu na zakładane nieznaczące wykorzystanie tych funkcji na urządzeniach mobilnych.

Dodatkową funkcją niedostępną w aplikacji webowej jest możliwość zeskanowania kodu kreskowego produktu i na jego podstawie otrzymania szczegółowych informacji o danym produkcie. Umożliwi to użytkownikowi sprawniejsze wprowadzanie zakupionych produktów do bazy danych, co wpłynie na lepsze gospodarowanie zasobami. Wszystkie dane wykorzystywane w aplikacji są przechowywane i udostępniane przez aplikację webową „ReShP” za pomocą API.