

# Brand Repository

## Functional Specifications

# 1 Content

2	Document Information.....	3
2.1	Versions.....	3
3	Data .....	4
3.1	Definitions .....	4
3.2	Perimeter.....	4
3.3	Schema .....	4
3.4	Data Model.....	4
3.4.1	BRAND table .....	5
3.4.2	BRAND_TYPE table .....	5
3.4.3	BRAND_OWNER table .....	5
4	General Functionalities.....	5
4.1	Users.....	5
4.2	Log .....	5
5	Admin Functionalities.....	6
5.1	Brand management.....	6
5.1.1	List of brands .....	6
5.1.2	Add a brand .....	6
5.1.3	Modify a brand .....	7
5.1.4	Delete a brand .....	7
5.2	Owner management.....	7
5.3	Brand Type management.....	7
6	Public Functionalities.....	8
6.1	Home page .....	8

## 2 Document Information

### 2.1 Versions

Version	Author	Modifications
2013.11.13 #2	P. Plagnol	Creation of the document
2013.11.13 # 3	P. Plagnol	Modifications in blue
2013.11.14 # 1	P. Plagnol	Modifications in blue
2013.11.20 #1	P. Plagnol	Modifications in blue

## 3 Data

### 3.1 Definitions

**BSIN:** The **B**rand **S**tandard Identification **N**umber is a 6-character alphanumeric unique (0 and O excluded) identifier assigned by OKFN. Each brand stored on this repository is given a unique BSIN.

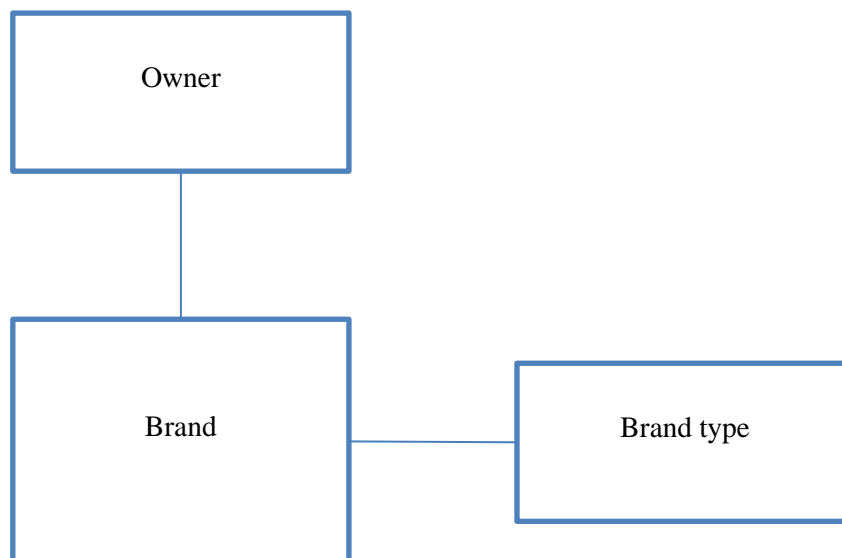
**Owner:** Legal owner of brand

**Type of brand:** Brand owner by manufacturer and retailers are of 2 different types. The repository contains groups of brands when the number of brand is too important and that the number of product is small (Wine, Cheese,...).

### 3.2 Perimeter

The repository excludes brands assigned to books and magazine.

### 3.3 Schema



### 3.4 Data Model

### 3.4.1 BRAND table

Field	PK/FK	Type	Null	Description
BSIN	PK	char(6)	N	BSIN code
BRAND_NM		varchar(255)	N	Name of the brand
GROUP_CD	FK	int	Y	Group owning the brand
BRAND_TYPE_CD	FK	int	N	Type of the brand
BRAND_LINK		varchar(255)	Y	URL of the brand
FLAG_DELETE		tinyint(1)	N	Deletion flag 0: not deleted, 1: deleted
LAST_MODIFIED		datetime	N	Last modification date
COMMENTS		varchar(255)	Y	

### 3.4.2 BRAND\_TYPE table

Field	PK/FK	Type	Null	Description
BRAND_TYPE_CD	PK	int	N	Type code
BRAND_TYPE_NM		varchar(255)	N	1: Manufacturer-brand 2: Retailer Brand 3: Group of brands

### 3.4.3 BRAND\_OWNER table

Field	PK/FK	Type	Null	Description
OWNER_CD	PK	int	N	Owner code
OWNER_NM		varchar(255)	N	Owner name
OWNER_LINK		varchar(255)	Y	URL of the owner website
OWNER_WIKI_EN		varchar(255)	Y	URL of the English wiki (end of URL)

### 3.4.4 Visitor\_proposal

Structure to determined (including the date, status, user assigned to validate etc.....)

## 4 General Functionalities

### 4.1 Users

Type of users, authorizations, access,...

### 4.2 Log

Each change applied on the database table must be stored in a log table which indicates:

- the user who did the change
- the date and time
- actions (table, fields,...)

Question : it's possible and easy with Django ?

## 5 Admin Functionalities

### 5.1 Brand management

#### 5.1.1 List of brands

The interface allow visitor to browse the list of BSIN (elements to display will be defined in a next version of this specification).

Members of the repository can edit directly a BSIN when they the list is displayed.

#### 5.1.2 Validate a brand

When a visitor submits a new brand, a control must be done to check that the brand doesn't exist and that all elements are corrects.

Cases:

- The brand already exists : a mail is sent to inform the visitor and provide the BSIN
- The brand doesn't exist but is not correct : a mail is sent to ask more precisions

It's critical to check that the brand doesn't already exists : a tool helping to search must be provided (to be discussed).

The moderator in charge of the check must be able to modify the name, type and url of the brand and to load another picture.

When brand is validated, the statuts change in the visitor proposal table and a BSIN is created in brand table. A mail is sent to the visitor to announce the creation of BSIN.

#### 5.1.3 Add a brand

##### 5.1.3.1 Interface

A text field is used to get the brand name.

1 type of brand is selected among all the types stored in brand\_type.

A owner is selected among all the owner of the brand\_owner table

A tool propose to upload the logo and automatically put him to a format of 150 \* 150 pixel

##### 5.1.3.2 Controls

These fields are required: Brand name and brand type.

Remark: The brand must be unique so it's very important to ensure that the brand is not already used. This could be auto

#### 5.1.3.3 Functional rules

When a brand is added in the repository, The BSIN is randomly generated. The BSIN is composed of 6 alpha numeric characters in **upper case** and **excluding 'O' and '0'** (zero) to avoid confusion.

The picture uploaded is resized to 150px\*150px and stored locally (the storage on amazon will be done in a future release).

#### 5.1.4 Modify a brand

BSIN is a master data and is stable. Name of a brand is linked to the BSIN and so the field BRAND\_NM can't change easily. This critical action must be checked by several users before to be done.

- If a brand is not spelled correctly ("Coca cola" instead of "Coca-Cola"), it's possible to change the field
- If a brand change of name, a new BSIN code must be created (with in comments the link between the 2 BSIN) the change of name is forbidden.
- In all other cases, it's not possible to change the brand name

All the other elements can be modified.

#### 5.1.5 Delete a brand

When a brand is deleted, clients of the repository must be informed of it and must have the reason and the potential BSIN replacing the one deleted.

So, the deletion needs to be logic:

In the Brand table:

- the field *flag\_deleted* becomes true
- the explanation is required in the *comment* field
- the date of deletion is indicated in the field *Last\_modified*

## 5.2 Owner management

... coming soon

## 5.3 Brand Type management

... coming soon

## 6 Public Functionalities

### 6.1 Home page

The home page display very clearly:

- A browser to give to the user an overview of the brand repository
- A link to collaborate entering a brand which doesn't exist
- List of the 10 (or 20) last BSIN added

### 6.2 Form to add a brand

A form is displayed to let visitors propose a brand not existing:

- The name
- The type of brand (only 1 and 2)
- The URL
- A logo to upload
- An email address (with confirmation)

The email address is used to inform the visitor of the creation of a BSIN for this brand

A captcha can be added to avoid attacks