Docent Jeroen de Kort, Semester 4, 2020/2021

A close up of a logo

Description automatically generated

**Mario Pizzeria Casus Analysis**

by Lex van Dalen and Slawomir Twardowski

**Table of Contents**

[Introduction 3](#_Toc55067864)

[General requirements 4](#_Toc55067865)

[Mario Pizzerias data types 10](#_Toc55067866)

[Extra ingredienten.csv 11](#_Toc55067867)

[MarioOrderData01\_10000.csv - MarioOrderData04\_10000.csv 11](#_Toc55067868)

[Overige producten.xlsx 12](#_Toc55067869)

[pizza\_ingredienten.xlsx 12](#_Toc55067870)

[pizzabodems.xlsx 13](#_Toc55067871)

[Winkels Mario.txt 13](#_Toc55067872)

[Postcode tabel.mdb 14](#_Toc55067873)

[Summary 15](#_Toc55067874)

Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.Error! Hyperlink reference not valid.Error! Bookmark not defined.

# **Introduction**

This document concerns the analysis of the Super Mario Pizzeria business case.

At first, we’re going to analyze the general requirements of the whole ordering system, then will go over the Domino’s website with its functionality concerning the use of (dynamic) data fetched from databases, on which the Mario Pizza system will be based on. We will use screenshots and diagrams to help visualize the layout and actions taking place on the site.

In the next step we will analyze the data files available as part of the current Super Mario Pizzeria operation. Those files consist of information from different stores, in several different database formats and the data contained will have to be cleaned up and transferred to a new all-encompassing database of our own design. The data base, as per client’s requirements, should be ready to migration to a cloud-based environment in the future.

# **General requirements**

* Automated ordering system ala Domino’s Pizza
* Incorporated existing data handling
* Growth oriented/ready solution
* Possible cloud migration in the future
* Compatible with existing business

**Diagram

Description automatically generated**

Fig. Mario Pizza web store solution diagram

**Domino’s website analysis**

Website in question can be found under this link: <https://www.dominos.nl/>

This is an overview of the main functionality used by the Domino’s Pizza ordering page. This will make the foundation for further decisions undertaken in the data base design.

Possibility to choose right away whether an order should be delivered or taken away; typing in postcode automatically suggest a list of nearby pizza shops:

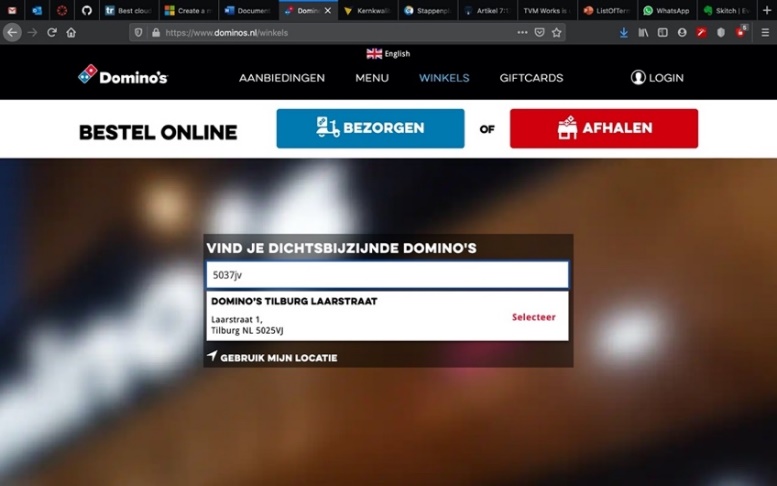


Fig. Domino's website main

Login possibility to trace orders and recall addresses. Orders are being stored and linked to the account for further use, like ordering the same meals etc.:

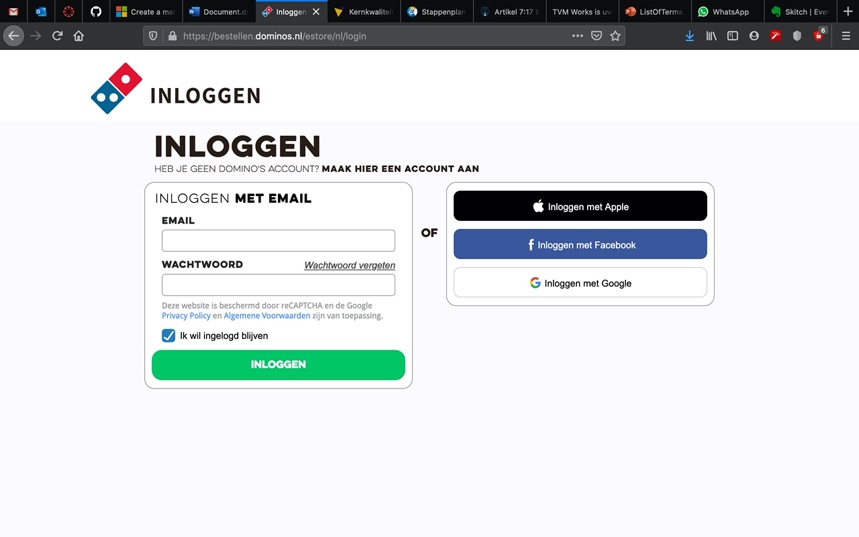
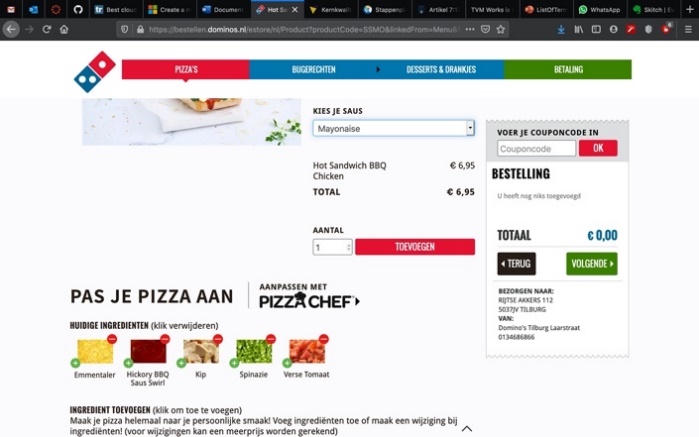


Fig. Domino's website login page

Menu display + option to Add product to the order; order details like total price, quantity, products to buy etc. are dynamically updated in a sidebar; possibility to add extra ingredients directly from the main page:

 Graphical user interface, website

Description automatically generated

Fig. Domino's website ordering process 1

Products can be filtered by selecting a specific filtering option; a coupon code can be entered for promotion benefits:

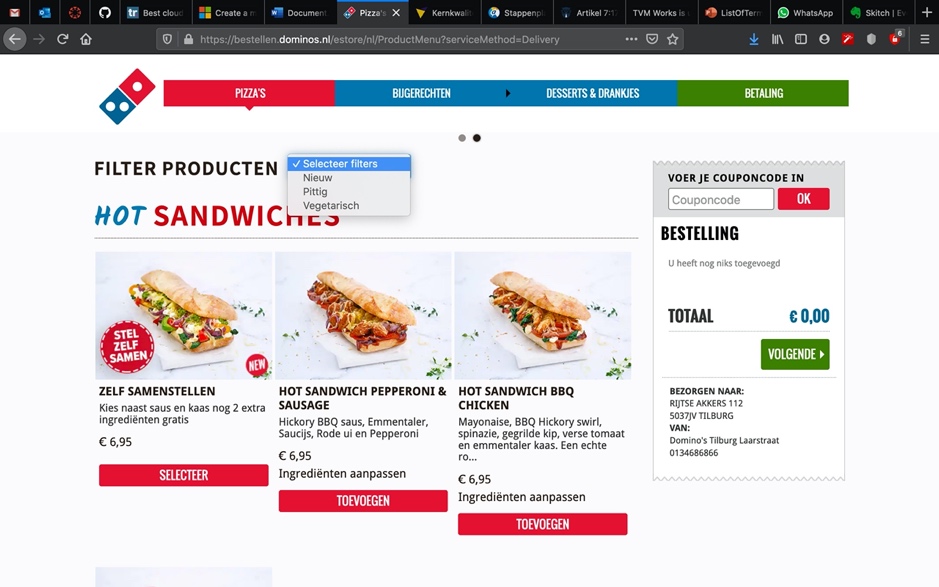


Fig. Domino's website product filtering options

The site keeps recently used addresses in cache for convenience even without the use of an account:

A screenshot of a cell phone

Description automatically generated

Fig. Domino's website recent location caching

Selecting specific pizzas is realized by using drop down lists and the fee is being calculated dynamically as the user selects given ingredients:

A screenshot of a computer

Description automatically generated

Fig. Domino's website pizza selection dropdown

If given pizza shop is closed at the moment it is still possible to place an order on a specified date and time:

A screenshot of a computer

Description automatically generated

Fig. Domino's website ordering on a specific date/time

Checkout requires consent to store the user data and lets the user use the option to leave a notice to the delivery person:

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

Fig. Domino's website payment page

At checkout payment method can be selected; it is possible to tip the delivery person at this point (at predefined or chosen by user amount) and the specified tip is added to the total cost dynamically:

A screenshot of a computer screen

Description automatically generated

Fig. Domino's website payment page 2

# **Mario Pizzerias data types**

There are a few different data file types used in the Mario Pizzerias current system. The files use different formats, are organized in different ways and contain errors due to manual input. This presents a challenge in aligning the databases with one another and cleaning up the provided data. Moreover, in the transition period data will still be supplied in current un-ordered manner, so the import process will have to be repeatable and automatized.

On the following pages is a list of provided files with a short description of the contents and possible obstacles.

## ***Extra ingredienten.csv***

Contains a list of ingredients (like: ananas, bacon, salami etc.) in one row and price in second row. While the cost of ingredients seems to be correct the price fields contain numbers as well as alphabetical symbols. File consists of 29 rows.

A picture containing table

Description automatically generated

Fig. Extra ingredienten.csv - Price field containing numerical value and a alphabetical symbol (letter "d")

## ***MarioOrderData01\_10000.csv - MarioOrderData04\_10000.csv***

Those 4 files contain a list of all orders placed by clients. A complete orders history is stored here (approx.). Columns used:

*Winkelnaam, Klantnaam, TelefoonNr, Email, Adres, Woonplaats, Besteldatum, AfleverType, AfleverDatum, AfleverMoment, Product, PizzaBodem, PizzaSaus, Prijs, Bezorgkosten, Aantal, Extra Ingredi√´nten, Prijs Extra Ingredi√´nten, Regelprijs, Totaalprijs, Gebruikte Coupon, Coupon Korting, Te Betalen.*

Each order consists of as many rows as there are products/ingredients in the order and each order is separated by an empty row. Additionally extra ingredients are listed in just one field.

A close up of a screen

Description automatically generated

*Fig.* 12 *MarioOrderData.csv - first 5 rows contain redundant information; an empty row is used as a delimiter separating orders from each other*

A picture containing text

Description automatically generated

Fig. 13 MarioOrderData.csv - Extra Ingredients are all listed in one field, this will have to be parsed during import to be able to treat it as separate products in the newly designed database.

## ***Overige producten.xlsx***

Contains a list of products other than pizzas (ie. Desserts, drinks etc.). Column names:

*categorie, subcategorie, productnaam, productomschrijving, prijs, spicy, vegetarisch*

Not all products have a description (empty *productomschrijving* field). Last 2 columns are binary values. Row 23 seems to be overlapping two different categories of products (corrupted).

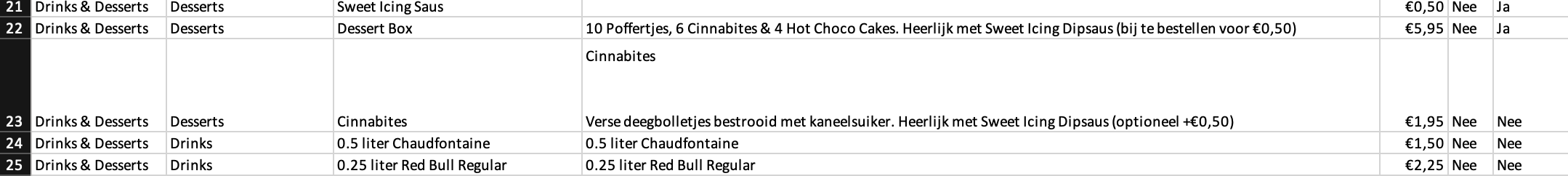


Fig. OverigeProducten.xlsx - overlapping/corrupted row

## ***pizza\_ingredienten.xlsx***

This file contains all the pizza ingredients found on any particular pizza (similar to a recipe). One type of pizza takes as many rows as there are ingredients on it. Fields:

*categorie, subcategorie, productnaam, productomschrijving, prijs, bezorgtoeslag, spicy, vegetarisch, beschikbaar, aantalkeer\_ingredient, ingredientnaam, pizzasaus\_standaard*

Table, Excel

Description automatically generated

Fig. pizza\_ingredienten.xlsx - first row used for table names; any particular pizza type takes as many rows as there are ingredients on the pizza; the separating factor will be pizza name ("productnaam" table)

## ***pizzabodems.xlsx***

A list of available pizza bottoms. Contains product name, description, price, availability. Price field will have to be parsed to get rid of the currency symbol. Availability column will be converted into a *Boolean* type. Columns contained:

*naam, diameter, omschrijving, toeslag, beschikbaar*

A picture containing text

Description automatically generated

Fig. pizzabodem.xlsx - again first row used for column names

## ***Winkels Mario.txt***

This file contains a list of all the shops. Shops are separated by an empty line. The data seems to be organized in the following manner (top to bottom for a shop):

* *Place* (this can be a city or district, it’s inconsistent)
* *Street*
* *House number*
* *City*
* *Country code*
* *Post code*
* *Phone number*

As this document has been typed in by an employee there are some typing errors etc. which would need cleaning up.

Text

Description automatically generated

Fig. Winkels Mario.txt - containing store addresses typed in by an employee; contains address inconsistencies

## ***Postcode tabel.mdb***

This database consists of all the postcodes and corresponding address breakdown in POSTCODES table; and municipality data in GEMEENTEN table. On this basis an address can be deducted from the postcode alone. It has the following columns:

*POSTCODES table columns:*

*A13\_POSTCODE, A13\_REEKSIND, A13\_BREEKPUNT\_VAN, A13\_WOONPLAATS, A13\_STRAATNAAM, A13\_GEMEENTECODE*

From conducted research it seems the POSTCODE table is incomplete. According to [www.poscodeafstanden.nl](http://www.poscodeafstanden.nl) as of writing (mid-2020) there are 460 478 unique zip codes in the Netherlands. The file contains only 304 573 unique records, however. This will pose a challenge, as it’ll be potentially impossible to bind some of the store or clients addresses. This issue would have to be taken up with the Mario Pizza business owner to look for a fitting solution.

*GEMEENTEN table columns:*

*N42\_GEM\_KODE, N42\_GEM\_NAAM*

# **Summary**

As can be seen from the supplied analysis material the scope of work to be done is significant and requires custom solutions, developed specifically for the needs of the client’s business case. Considering the nature of supplied data files and in some cases its incompleteness, in other its inconsistencies, a custom software solution will have to be programmed to fulfill the importing and data cleaning duties. Furthermore, consultation with the client will have to be undertaken to resolve some of the issues - like the missing postcodes or faulty addresses contained within the files. Some of those problems ask for a systemic solution on the client’s side and would have to be addressed at the source, otherwise the timeframe of the project could extend dramatically and, what follows, the cost as well. It seems that in many cases the process could be streamlined if the supplied data files would be kept in better order. This is object of further consultation and at this stage the complete solution cannot be developed without proper assessment and engagement from the business owner.

A close up of a logo

Description automatically generated

2020/2021