UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN

GRADO EN INGENIERÍA DE TECNOLOGÍAS Y SERVICIOS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Design and Implementation of a Social Marketplace for a Startup based on a Content Management System. Application of Sentiment and Emotion Analysis for Sensing Customer Feedback.

ÁLVARO GUILLERMO GERICKE PARGA

2017

TRABAJO FIN DE GRADO

Título:	Diseño e Implementación de un Marketplace para una Startup basado en un Sistema de Gestión de Contenidos. Aplicación de un Sistema de Análisis de Sentimientos y Emociones para Conocer y Mejorar la Experiencia de Usuario.
Título (inglés):	Design and Implementation of a Social Marketplace for a Startup based on a Content Management System. Application of Sentiment and Emotion Analysis for sensing customer feedback.
Autor:	Álvaro Gericke Parga
Tutor:	Carlos A. Iglesias Fernández
Departamento:	Ingeniería de Sistemas Telemáticos

MIEMBROS DEL TRIBUNAL CALIFICADOR

Presidente:

Vocal:

Secretario:

Suplente:

FECHA DE LECTURA:

CALIFICACIÓN:

UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR DE
INGENIEROS DE TELECOMUNICACIÓN

Departamento de Ingeniería de Sistemas Telemáticos
Grupo de Sistemas Inteligentes

TRABAJO FIN DE GRADO

DESIGN AND IMPLEMENTATION OF A SOCIAL MARKETPLACE FOR A STARTUP BASED ON A CONTENT MANAGEMENT SYSTEM. APPLICATION OF SENTIMENT AND EMOTION ANALYSIS FOR SENSING CUSTOMER FEEDBACK.

Álvaro Gericke Parga

Junio de 2017

Resumen

Esta memoria es el resultado de un proyecto cuyo objetivo ha sido desplegar y desarrollar un marketplace de alquiler de equipamiento deportivo. A su vez, se ha desarrollado un sistema de análisis de sentimientos y emociones en comentarios para Wordpress basado en Senpy.

Para desplegar el marketplace, hemos utilizado el CMS Wordpress, el cual ha agilizado y facilitado muchas labores. Se han utilizado varios plugins de Wordpress existentes para cubrir diversas funcionalidades. Por su parte, el sistema de análisis de sentimientos y emociones ha sido desarrollado mediante un plugin para Wordpress. Se han desarrollado dos versiones para dicho plugin. Una versión se puede descargar desde el repositorio oficial de Wordpress y la otra es específica para el marketplace desplegado.

Para realizar el análisis de sentimientos y emociones hemos utilizado principalmente Senpy. El plugin permite configurar todos los parámetros que hay que pasar a Senpy para realizar el análisis del comentario. Toda esa información se guarda en la base de datos para su posterior visualización. El usuario podrá configurar cómo se verán los comentarios.

Para el caso del marketplace, el sistema de visualización muestra a cada tienda y al administrador del marketplace un resumen de los sentimientos y emociones que se expresan en los comentarios relativos a sus productos. Para ello nos hemos ayudado del plugin de Wordpress wpDataTables el cuál utiliza las bibliotecas Google Charts, HighCharts y Charts.js para renderizar los datos en tablas y gráficos .

Por último, se han empleado varios schemas de Schema.org para añadir información semántica al marketplace utilizando Microdata y mejorar así el posicionamiento de éste y sus tiendas en los buscadores.

Como resultado, este proyecto nos permitirá realizar un amplio análisis de los comentarios de los usuarios, pudiendo transmitir de forma clara esta información a las tiendas y así éstas puedan tomar decisiones al respecto.

Palabras clave: Wordpress, Kibiwoo, Sentimientos, Emociones, Análisis, Marketplace, Senpy, Surf, Bicicletas, Ski, Snow.

Abstract

This thesis is the result of a project whose objective has been to develop and deploy a sports equipment rental marketplace. Also we have developed an emotion and sentiments analysis system in comments for Wordpress based on Senpy.

For deploying the marketplace, we have used the CMS Wordpress which has accelerated and facilitated many tasks. Several existing Wordpress plugins have been used to cover various functionalities. For the emotions and sentiments analysis system, a Wordpress plugin has been developed. Two versions of this plugin have been programmed. The first one can be downloaded from Wordpress's official plugin repository and the other one is specific for the marketplace.

For carrying out the emotion and sentiments analysis we have used mainly Senpy. The Wordpress plugin lets the user to configure all of the Senpy's parameters for making the comments analysis. All this information is stored in Wordpress's database for later viewing. The user will be able to configure the comment's view.

For the marketplace, the visualization system shows to each shop and to Kibiwoo's administrator an overview of the sentiments and emotions expressed by users on comments about their products. For this issue, we have used wpDataTables Wordpress plugin that uses Google Charts, HighCharts and Charts.js coding libraries for rendering data in charts and tables.

Finally, several Schema.org's schemas have been used to add semantic information to the marketplace using Microdata. This way we improve the marketplace and its shops online visibility among search engines.

As a result, this project will allow us to carry out an extensive analysis of the user's comments, being able to transmit this information to the stores in a user-friendly way so that they can make decisions about it.

Keywords: Wordpress, Kibiwoo, Sentiments, Emotions, Analysis, Marketplace, Senpy, Surf, Bicycles, Ski, Snow

Agradecimientos

En primer lugar quiero agradecer a mi tutor Carlos Á. Iglesias por haberme ayudado tanto en este trabajo fin de grado.

También quisiera dar las gracias a los miembros del GSI, en especial a Manu por haberme ayudado cada vez que se lo he solicitado.

Por último, agradecer a mis padres y a mis hermanos por todo el apoyo que me han dado durante todos estos años y por apoyarme en todo esos proyectos que he llevado a cabo en combinación con los años de Universidad.

Contents

List of Figures

CHAPTER 1

Introduction

The chapter presents the main ideas of the project and the objectives it pursues. It makes a general overview of the context in which is developed. Finally we make a brief description of the content of each chapter.

1.1 Context

Nowadays, digitalization within nearly all areas is a fact. However, there are several market niches where new technologies do not coexist with traditional non Internet based methods. This is the case of retail stores specially, as hardly some of them have only a simple web page. They see this transformation really expensive and difficult, but the real problem is the lack of knowledge in the available tools that practically do all the hard work for them.

Moreover, although the number of web applications that allow customers to compare related products or services from different shops has increased year by year lately, there are none of them that compare the offer in so popular sports in Spain such as surf, skiing and cycling. What is more, today the most valuable thing on internet for shops are their customers reviews and comments. Then, it is really important to know at first hand their customer's sensations and experiences related to their products or services, *what customers*

think should be improved and what do they like the most.

It is obvious we are living a transition on the form customers interact with each other and the way they interact their smartphones and internet. The result of it is what we call the WEB 3.0, in which the Semantic Web has an important role. Today, it is not enough to have a really good presentation of your content or services on Internet. You need to provide additional value to that information or services, analyse and interprete all the user's data left on the Web.

So the aim of this project is to provide the necessary tools for these shops so they can understand what their clients are saying on internet.

1.2 Project goals

The final purpose of this end-of-degree project is the design and development of a renting Marketplace centered in ski, surf and cycling products that implements a sentiment and emotion analysis tool to provide shops with value information in order to improve their services's quality and customers exprience. This main goal will be achieved after all the following subgoals are completed:

- Design and deployment of a Social Marketplace based on Wordpress CMS that allows shops to manage their products and bookings, and users to compare similar products from nearby shops and book the one that fits them best.
- Development of a Wordpress plugin for emotion and sentiment analysis on customers reviews. Development of a semantic dictionary for sentiment aspect-based classification oriented for shops.
- Representation of the analysis information on a user-friendly way so that shops can interprete the data and take action.
- Introduce semantic content in the web using Schema's vocabularies in order to improve the SEO and the web positioning of the shops that take part in the Marketplace.

1.3 Structure of this document

In this section we introduce the different chapters presented on this document:

Chapter 1 introduces the project and its context. We provide the goals we want to reach with it and finally explains briefly the structure of this document.

Chapter 2 provides a description of the main technologies used on this project to achieve its goals.

Chapter 3 makes a requirement analysis which will enable a more complete vision of the system, listing the use cases of the system.

Chapter 4 explains in detail the architecture of the project, identifying the different modules involved, explaining each of them in detail and the way they interact. Here we introduce the different versions of the plugin developed for the different types of users that can make use of it.

Chapter 5 describes in detail how shops can use the system developed and in which ways they benefit from it.

Chapter 6 is a final overview of the project, the development process and the achieved goals. Also future work for extending the system functionality is proposed.

CHAPTER 2

Enabling Technologies

In this chapter, we are going to describe the different technologies used to carry out this project. We will start with Wordpress, the CMS used to deploy the Marketplace application and the plugins applied. Then, we will give a brief view overview of Senpy, a tool developed by the Intelligent Systems Group (GSI) which we have used to cover the sentiment and emotion analysis part. Finally, we will describe Schema.org vocabulary used for semantic purposes and the different formats available to add information to our webpage.

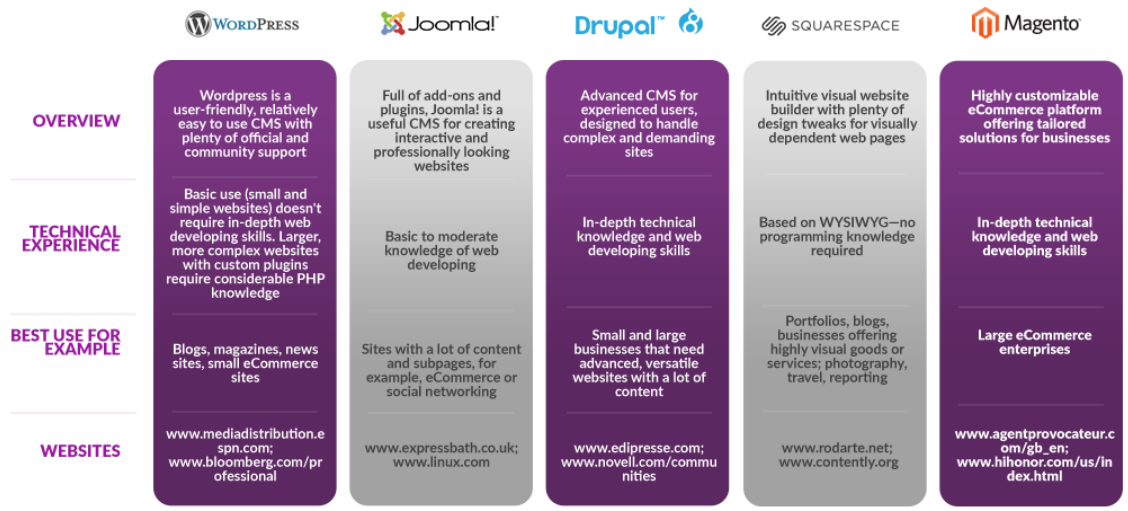
2.1 Content Management Systems

A content management system (CMS) is a software application or set of related programs that are used to create and manage digital content. CMSes are typically used for enterprise content management (ECM) and web content management (WCM). An ECM facilitates collaboration in the workplace by integrating document management, digital asset management and records retention functionalities, and providing end users with role-based access to the organization's digital assets.[1]

CMSes provide lots of tools and functionalities already developed that makes deploying

[1] http://searchcontentmanagement.techtarget.com/definition/content-management-system-CMS

a webpage much more faster. Moreover, they usually are opens source so that developers can integrate their programmes with the CMS's core. We can see in the following table a comparison of the three main CMS options:



Source: https://www.polcode.com/wp-content/uploads/2017/03/CMS-Comparison1.png

Figure 2.1: CMS Comparison Table

In our case, we have selected Wordpress because its the most extended one with the biggest community, it has lots of plugins, it is free and really easy to install and has one of the most popular and complete E-commerce plugins, WooCommerce.

2.1.1 Wordpress

Wordpress [10] is a free and open-source content management system (CMS) based on PHP and MySQL. As it is based on PHP, it is installed on a web server. It is released under the GPLv2 (or later) license from the Free Software Foundation.

It was initially created for blogging services, but while the community was growing, new software packages and plugins were being developed and it turned rapidly into the most popular CMS in the world.

Nowadays, Wordpress supports 23,4% of worldwide websites, being the world's most popular CMS. It has over 12.000 available themes and more than 34.000 plugins to extend Wordpress functionality.

Themes

Wordpress has a high variety of predefined themes. A theme determines the visual part of a website. Each theme has different levels of visual and functionality personalization. It is important that the theme only provides the visual part as you will add and implement functionality by extending your Wordpress site with plugins.

Moreover, theme's code can be modified. This action should be done by creating a child theme, which is a theme that derives from a parent theme, and enables us to add or modify the HTML, CSS, JavaScript or PHP of the parent one. It is necessary to create a child theme in order to avoid losing all of our changes when the theme is updated.

There are free and premium themes. What is more, you can develop your own custom Wordpress Theme that fits perfectly your necessities and styling preferences respecting Wordpress's rules, publish it or sell it.

In my case I will use a **personalized theme** as the Startup requires it.

Plugins

A Worpress Plugin [14] is a tool used to add functionality to your web. Wordpress itself is a simple blog creator and manager. Plugins are the ones which allow you to use Wordpress as anything you want, from a social network to an E-commerce platform.

In my case, I used Wordpress in order to develop a Marketplace which is focused on renting third party sports products. This Marketplace is the technological sustenance base of a Startup called Kibiwoo. I have used several plugins to implement all the functionalities needed to deploy this Marketplace. Some of them are existent ones but other focused on providing more specific services have been developed

I have used the following existing plugins:

- ***WooCommerce.***[2] This is the most important plugin in the project. It enables us to turn our Wordpress's blog site into a complete E-commerce platform.
  We acquire the following functionalities with this plugin:
  - Products Management.
  - User Management (roles, user products, account settings).
  - Shopping Cart.
  - Taxes and Delivery Fees Management.

[2]https://woocommerce.com

  - Orders Resume & Graphics.
  - Discounts & Coupons.

- ***WooCommerce Product Vendors.***[3] Plugin to turn our simple E-commerce site into a Multi-Vendor Marketplace that allows vendors to manage their own products and earn commissions. We can set different types of commissions depending on the vendor or product, choose different types of delivery fees, manage vendor accounts, categorize products. This way we can take our commission as customers pay directly to Kibiwoo, and then Kibiwoo will pay their vendors after taking their commission.

- ***WooCommerce Bookings.***[4] Another fundamental plugin for my project. Without this plugin, we can only sell our products or third party ones. WooCommerce Bookings enables to add booking functionality to the products, such as sell time or date-based bookings to any product in our Marketplace. It has multiple services and allows a high personalization of product prices, availability and even add complements to each product.

  We can select in each product whether it is bookable or not, create booking slots, enable multi-person bookings or even offer a per-person discount. We add also a very complete booking management functionalities as well as very deep product booking configuration (costs, days, availability, email notifications).

- ***Yoast SEO.***[5] SEO is fundamental if we aim to increase our customers as faster as possible and obtain better image and reputation than our competitors. Yoast SEO offers a wide range of characteristics and tools to improve our platform SEO.

  Moreover, not only offers really useful tools but it also provides with suggestions and continuously analyses all parts or your web in order to find possible things that can be harmful for your SEO.

  You can acquire a really good position in search engines even with no knowledge in the area. Meta elements and links, XML site maps, RSS optimization, social integration and lots more of functionalities are included in this plugin.

- ***Loco Translate.***[6] There are lot of plugins and themes which not include several languages. This is a really bad characteristic when taking about SEO if your web is not in the same language as the country's tongue where it operates. Moreover, if you edit the files directly, when the themes updates you will lose all of your personalization.

[3] https://woocommerce.com/products/product-vendors/
[4] https://woocommerce.com/products/woocommerce-bookings/
[5] https://yoast.com/wordpress/plugins/seo/
[6] https://es.wordpress.org/plugins/loco-translate/

So this plugin provides in-browser editing of Wordpress translation files. This means, translations are in a separate file so you will never lose your translations. It is not necessary to translate all of the strings, just the ones you want. In addition, you can add as many languages files as you want, so if you have activity in different countries you can select a translation file depending on the area.

Furthermore, it provides additional localization tools for developers such as extracting strings and generating templates.

- ***wpDataTables.***[7] Really useful plugin for extracting information from Wordpress Databases and rendering it in a table or chart. It has a high level of configuration and different chart available. It also has its own Wordpress Hooks so that developers can personalize their charts and tables at a rally high level.

  We will use it as the vehicle to show shops information gathered through the emotion and sentiment plugin in a user-friendly way. Charts will be responsive and interactive.

- ***Emotion and Sentiment Plugin for Wordpress*** The Wordpress plugin we will develop to adapt Senpy so that Wordpress users can use it. It will allow us to analyse the comments left on our platform and extract really value information for shops.

  We will deploy two versions of this plugin, the "public version" which will implement basically what you can do with the senpy cluster, and the personalized version for Kibiwoo, which will implement more advanced techonologies and configurations.

- ***WooCommerce PayPal Express Checkout Gateway.*** Paypal plugin to integrate Paypal Gateway in order to allow users to pay via Paypal. It also creates an express Checkout button to pay faster.

2.2 Sentiment And Emotion Analysis Technologies

Knowing what customers are expressing on a single comment is easy, but when it comes to extracting data and analysing information from within hundreds or thousands of comments it is not that easy. Over the last years, information has become the most valuable active for the majority of enterprises. But there is so many data on internet that is necessary to rely on software specialized on extracting value information from that data.

In our case, shops who know exactly what clients are opining and demanding are one step forward from its competitors.

[7] https://wpdatatables.com/

There are multiple available services and softwares that perform a sentiment and emotion analysis on the information you provide them. Some examples are Microsoft Azure Cognitive Services, AlchemyLanguage, Meaningcloud, Senpy, text2data...

But what we are seeking for is not only an application that outputs a simple number between 0 or 1 expressing the sentiment or just a word describing the emotion, we want something that uses a standard semantic language that allows to interact with other semantic services.

That is why we finally used Senpy in our project, as you can integrate any of the services in Senpy with a simple plugin and it provides its data in defined model languages such as Marl or Onyx.

2.2.1 Senpy

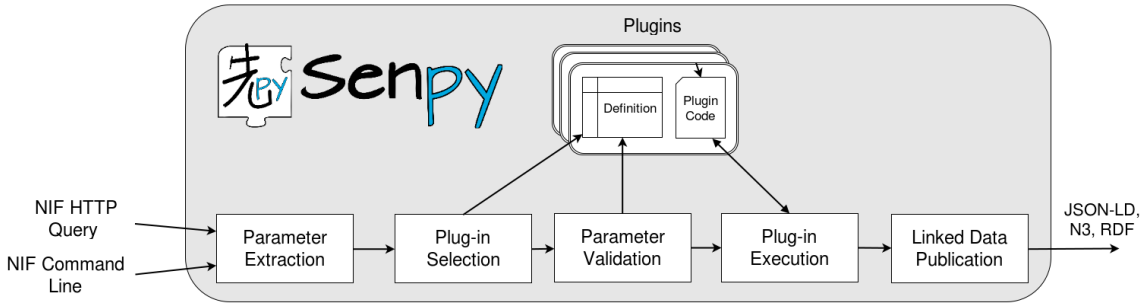
Senpy [12] is an open source framework to build semantic and emotion analysis services. It is a linked data model that makes use of different output and input formats of data such as turtle, JSON-LD and XML-RDF. This model uses Marl, Onyx and NIF semantic vocabularies to annotate its results. This technology has been developed by the GSI of ETSIT-UPM.

Senpy proposes a modular and dynamic architecture so that services of different providers are interoperable using a common interface. It makes it easy to develop and publish your own analysis algorithms. In our project we will focus on Senpy's adaptation to a Wordpress plugin so that we extent its usability to Wordpres's users. In fact, we will develop two different Wordpress plugins based on Senpy:

- *General purpose Wordpress plugin:* This part will be focused on adapting the actual NIF API to be used via a Wordpress Plugin, giving the possibility of choosing between the different plugins and parameters available for Senpy and showing the information in a widget.
- *Kibiwoo purpose Wordpress plugin:* This part will be focused on developing a Wordpress plugin as a software solution for renting shops that uses Senpy as its base on the sentiment and emotion analysis part before interpreting and rendering the data in a way it generates value for those shops.

The framework consists of two main modules: Senpy core, which is the building block of the service, and Senpy plug-ins, which consists in a number of NLP algorithms. In Fig. 2.2

we can see a simplified version of the processes involved in an analysis with the Senpy framework.



Source: http://senpy.readthedocs.io/en/latest/_images/senpy-architecture.png

Figure 2.2: Senpy Architecture

The available Senpy plugins are the following[8]:

- *emotion-wnaffect:* Emotion classifier using WordNet-Affect to calculate the percentage of each emotion. This plugin classifies among 6 emotions: anger, fear, disgust, joy, sadness or neutral. The only available language is English (en)
- *sentiment-140:* Sentiment classifier using rule-based classification for English and Spanish. This plugin uses sentiment140 data to perform classification. For more information: http://help.sentiment140.com/for-students/
- *sentiment-tass:* Sentiment classifier using rule-based classification based on English and Spanish.
- *sentiment-meaningCloud:* Sentiment analysis with meaningCloud service. To use this plugin, you need to obtain an API key from meaningCloud[9]. When you had obtained the meaningCloud API Key, you have to provide it to the plugin, using param apiKey. Example request: http://senpy.cluster.gsi.dit.upm.es/api/?algo=meaningCloud&language=en&apiK
- *emotion-anew:* This plugin consists in an emotion classifier using ANEW lexicon dictionary to calculate VAD (valence-arousal-dominance) of the sentence and determinate which emotion is closer to this value. Each emotion has a centroid, calculated according to this article [7]. The plugin is going to look for the words in the sentence that appear in the ANEW dictionary and calculate the average VAD score for the sentence. Once this score is calculated, it is going to seek the emotion that is closest to this value.

[8]http://test.senpy.cluster.gsi.dit.upm.es/#about
[9]https://www.meaningcloud.com/developer/login

- *affect:* Sentiment Analysis and Emotion Recognition. This plugin uses emotion and sentiment plugins to return an unified response with the sentiment analysis and the emotion Recognition. You have to choose as parameters one sentiment plugin and one emotion plugin.

- *sentiment-vader:* Sentiment classifier using vaderSentiment module. The parameter accepted is language. you can select between English or Spanish. The output uses Marl ontology developed at GSI UPM for semantic web.

- *sentiment-basic:* Sentiment classifier using rule-based classification for Spanish. Based on English to Spanish translation and SentiWordNet sentiment knowledge. This is a demo plugin that uses only some features from the TASS 2015 classifier. To use the entirely functional classifier you can use the service in: http://senpy.cluster.gsi.dit.upm.es.

2.3 Semantic Vocabularies

We have been talking about analysing sentiments and emotions but the retrieved information needs to follow a standard and be expressed using worldwide vocabularies. On the semantic web, vocabularies define the concepts and its relationships and help data integration.

W3C offers a large palette of techniques to describe and define different forms of vocabularies in a standard format. These include RDF and RDF Schemas [4], Simple Knowledge Organization System (SKOS) [9] and Web Ontology Language (OWL) [5]. The choice among these different technologies depend on the complexity and rigour required by a specific application.

In our area of concern, comments will talk about shops and its services, their products, their experiences and valuations. So we will use the schemas defined by Schema.org, as they have a Product Schema, a Review Schema, a Business Schema and much more.

2.3.1 Schema.org

Schema.org [1] is a collaborative, community activity with a mission to create, maintain, and promote schemas for structured data on the Internet, on web pages, in email messages, and beyond.

A shared vocabulary makes it easier for webmasters and developers to decide on a schema and get the maximum benefit for their efforts. It is in this spirit that the founders,

together with the larger community have come together - to provide a shared collection of schemas. Since April 2015, the W3C Schema.org Community Group is the main forum for schema collaboration, and provides the public-schemaorg@w3.org mailing list for discussions. Schema.org issues are tracked on GitHub.

Most webmasters are familiar with HTML tags on their pages. Usually, HTML tags tell the browser how to display the information included in the tag. For example, <h1>Avatar </h1> tells the browser to display the text string "Avatar" in a heading 1 format. However, the HTML tag doesn't give any information about what that text string means —"Avatar" could refer to the hugely successful 3D movie, or it could refer to a type of profile picture — and this can make it more difficult for search engines to intelligently display relevant content to a user.

Schema.org vocabulary can be used with many different encodings, including RDFa [2], Microdata [11] and JSON-LD [3]. These vocabularies cover entities, relationships between entities and actions, and can easily be extended through a well-documented extension model. Over 10 million sites use Schema.org to markup their web pages and email messages. Many applications from Google, Microsoft, Pinterest, Yandex and others already use these vocabularies to power rich, extensible experiences.

Founded by Google, Microsoft, Yahoo and Yandex, Schema.org vocabularies are developed by an open community process, using the public-schemaorg@w3.org mailing list and through GitHub. Schema.org provides a collection of shared vocabularies webmasters can use to mark up their pages in ways that can be understood by the major search engines: Google, Microsoft, Yandex and Yahoo!

For our project, we will use the Product [10], Review [11] and Store [12] schemas in order to describe elements inside Kibiwoo platform and improve Kibiwoo's site and the shops that form part of it SEO.

[10] http://schema.org/Product
[11] http://schema.org/Review
[12] http://schema.org/Store

CHAPTER 3

Requirement Analysis

This chapter is aimed at making a requirement analysis of the project. This will help us to detect all the necessities and have a complete vision of the system to be developed.

3.1 Introduction

A requirement analysis is made to identify the needs or conditions that the project has to meet. It is not about going deeply in each of the necessities and functionalities of the project but making a broadly identification of them, at a level of detail sufficient for making the system design.

This analysis is critical to the success or failure of a systems or software project. It must be documented. In our case, we will use the Unified Modeling Language (UML)[1] for this issue. UML is a graphical language for visualizing, specifying, constructing and documenting a system.

[1] http://www.uml.org/

3.2 Use cases

In this section, we will identify the main use cases of the Sentiment and Emotion Analysis Plugin for Wordpress. We will focus on Kibiwoo's version of the plugin as this version is for a business.

We will identify the main user actions when interacting with the system. This will guide us on defining the specifications of the system.

A description of the different actors involved will be presented in Sect. 3.2.1. Then, we will show in Sect. 3.2.2 the use cases identified and the actors that are involved in each of them. We will use an UML diagram for representing the use cases. Finally, in Subsect. 3.2.2.1 and Subsect. 3.2.2.2 we will present sequence diagrams for the different use cases. Again, these diagrams are developed with UML.

3.2.1 System Actors

First of all, we need to determine all the actors that are involved and interact with the system. The actors of our system are:

***Kibiwoo's Admin*:** The Kibiwoo's Administrator. He will be on charge of configuring the parameters of the plugin for analysing the comments. He can also have a general overview of all the results from all the comments analysed and filter them by the different shops associated to Kibiwoo. He is the main actor.

***Shop's Admin*:** The administrator of each shop. He can filter the analysis results. They can select between different options such as viewing the results of the comments analysed for each product, see a general analysis view of their shops or view the users that have generate those comments.

***Kibiwoo's Users*:** The users that rent the products on Kibiwoo's platform. They will be the ones that generate and leave the comments that will be analysed.

***Senpy*:** This is a secondary actor. It will be the software in charge of analysing the comment's text.

***Wordpress*:** This is a secondary actor. It is the CMS which supports the platform and the one that provides the different user interfaces for interacting with the plugin.

***wpDataTables*:** Another secondary actor. It is a Wordpress plugin that will be on charge of generating charts and tables from the data you provide him.

3.2.2 Use Cases

Next a use case UML diagram is presented. In this graphic it is shown the main use cases identified and the interconnection with the actors of the system.

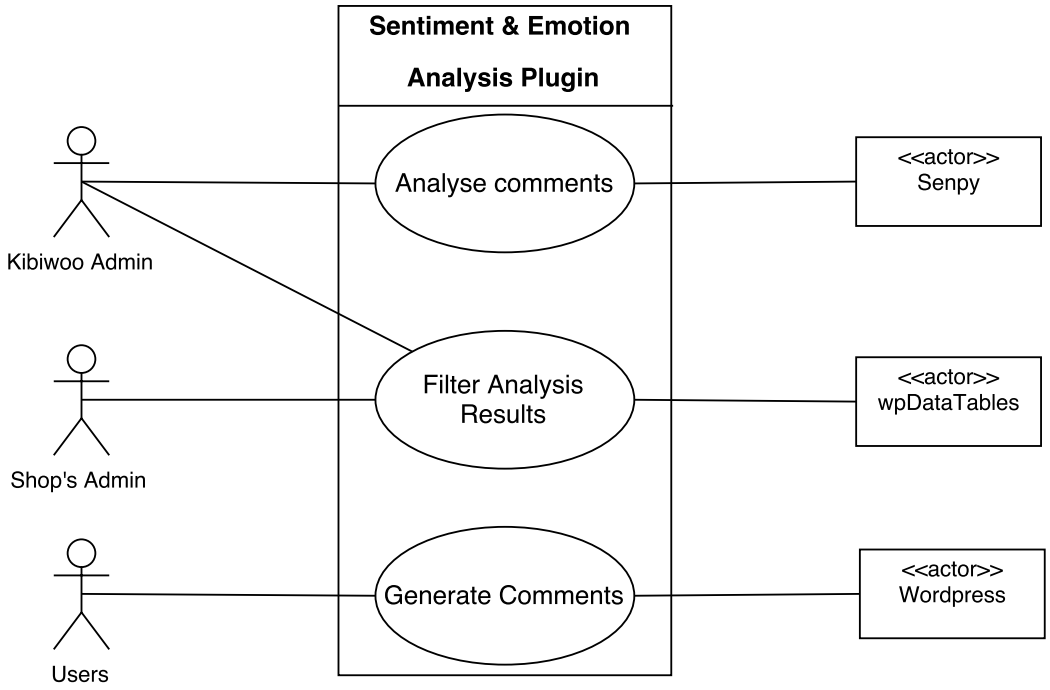


Figure 3.1: Use case diagram

3.2.2.1 Filter analysis results

The actor of this use case is Kibiwoo's admin or the shop's admin. In this case, the actor selects what information he wants to see in a chart or table. That information is extracted from the Wordpress Database and sent to wpDataTables plugin. With this information, this plugin generates a chart and outputs it through the user interface. This is an interactive chart where the actor can add filters to the information and the chart will be automatically updated.

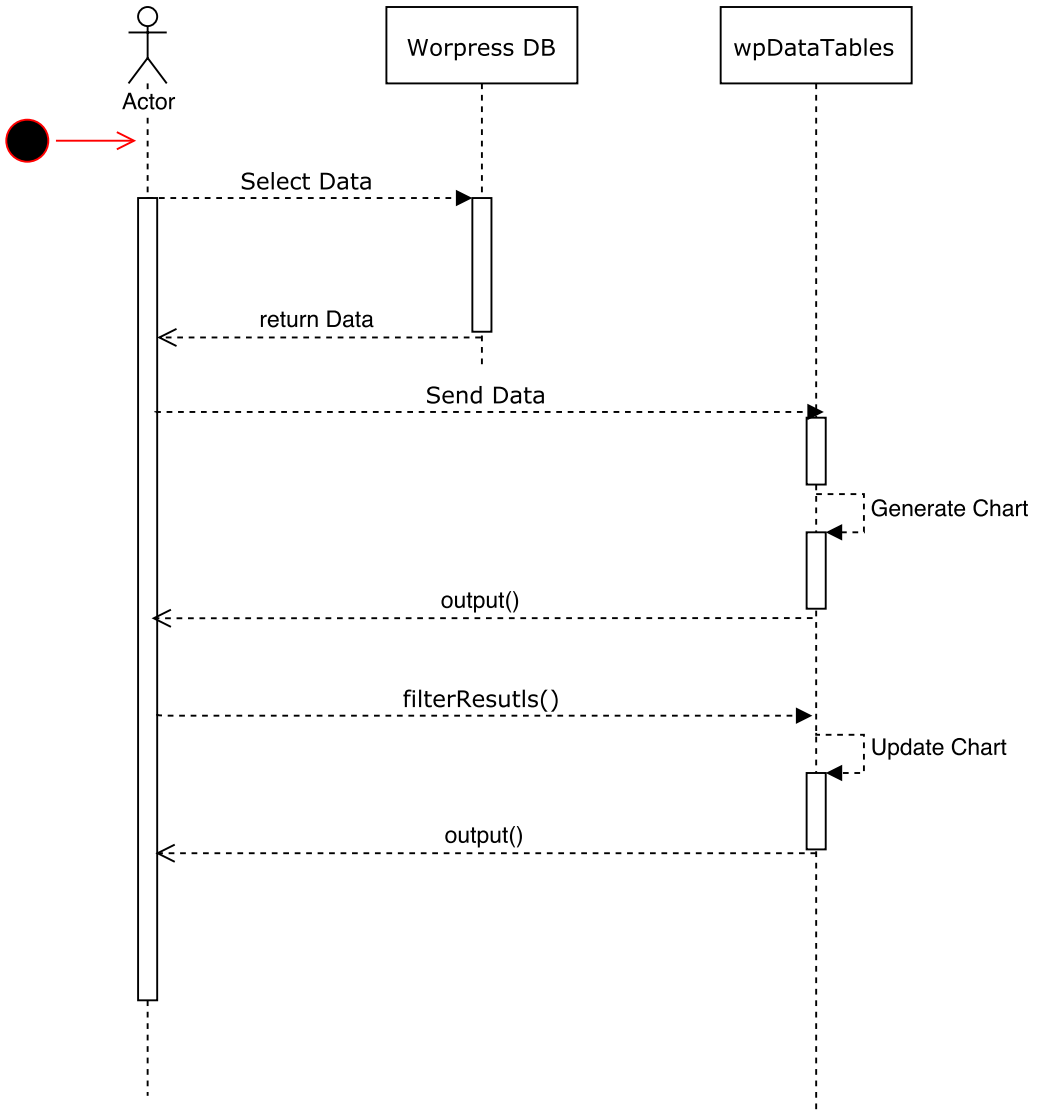


Figure 3.2: Filter Results Use Case

3.2.2.2 Analyse Comments

The actor of this use case is Kibiwoo's administrator. He selects the Senpy parameters. Then the comment to be analysed is sent to Senpy, that returns the analysis results. Only required information is extracted from the results and then saved to the table 'comment-meta' of Wordpress.

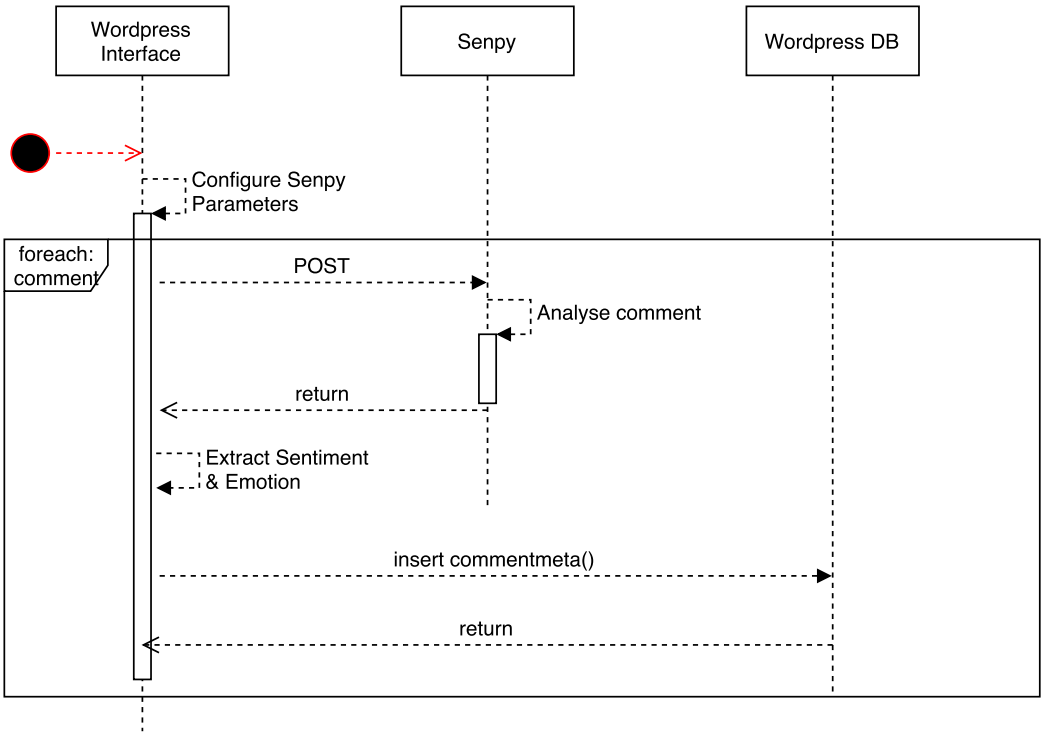


Figure 3.3: Analyse Comment Use Case

CHAPTER 4

Architecture

In this chapter we are going to explain the architecture of this project including the design phase and implementation details. We can divide the architecture of the project into two different systems, the emotion and sentiment analysis system, and the visualization system. First of all, we will give a general overview of the global architecture and how these different main modules interact. Then, we will describe the sentiment and emotion analysis system in detail, explaining the different processes involved. Afterwards, we will explain the visualization system developed. Finally, we will explain how data has been tagged using the Schema semantic vocabularies under Microdata encoding.

4.1 Overview

In this chapter we will give an introduction to each of the systems developed and how their functionality is integrated in order to generate value for users, Kibiwoo and its shops.

We will develop two versions of each part. A general version oriented for Wordpress's general users, and another version oriented for Kibiwoo. The first one will be available for download in the Wordpress's official plugins repository. Additionally, this plugin version will be programmed under Wordpress's coding standards and will be developed for supporting

various languages. Plugin security will also be taken into count seriously.

The analysis system will allow us to extract all the sentiments and emotions expressed on comments and save them in our databases. In Kibiwoo's version, additionally, a classification for the different stores and their products will be made. This will enable shops to filter the results for each of their products and compare their results with their competitors overall score. Sentiments and emotions from comments will be related to each product and hence also to the shop.

Each system will have some configuration parameters which will be defined and selected by the user in the menu that will be available in the admin's area. Depending on these parameters different Senpy plugins will be used and visualization details will defer. On the comment's analysis for extracting sentiment and emotions part, a slight visualization configuration will be needed for configuring the comment view.

The visualization part will be in charge of rendering the data extracted by the analysis system in a user-friendly way. We will use an existing Wordpress plugin for this task combined with some programming. Between the different Wordpress plugins available for making charts and tables we have selected WpDataTables[1]. This plugin enables us to:

- Pull table data from any source – Excel to MySQL.
- Create Responsive tables and charts.
- Hook my own functions to the filters and action hooks defined by wpDataTables developers.
- Use different library charts such as Goggle Charts, Highcharts or Chart.js
- Enable Advanced filters and searches.
- Customize the table's and chart's design.
- Select between 11 different column types.

So as from our point of view is the most complete Wordpress plugin for this task and it has a high level of personalization and configuration for developers, this will be the selected plugin although it is not free. The free options considered do not cover all of our necessities for this project which makes it worthy paying 35 $ for it.

[1] https://wpdatatables.com/

4.2 Emotion and Sentiment Analysis Plugin

As we described on Chap. 2: Enabling Technologies, Wordpress has a series of available APIs to interact with Wordpress's Core. The Plugin API provides a set of hooks that enable plugins to access specific parts of Wordpress.

Wordpress defines a distinguishing class to describe comments, the WP_Comments class[2]. This module will make use of hooks which are specifically oriented to interact with comments. Its main purpose is to analyse each comment and save the results as comment's meta-data. Additionally, it will generate a personalized visualization of the comment, rendering also the analysis information.

Wordpress saves comments and its meta-data in two different data tables, the wp_comments table and the wp_commentmeta table. The wp_commentmeta table has a comment_id column which stores a foreign key with the comment_id that identifies the comments in the wp_comment table. You can add any meta-data using the add_comment_data() function[3]. This function requires the comment_id, a meta_key, which needs to be a string value, a meta_value, which can be anything, and another optional boolean parameter named unique, which determines whether or not you want the key to stay unique.

We have developed two versions of this module. The first one is oriented for general use of Wordpress users. It basically implements a Senpy adaptation for Wordpress, adding a high level of personalization and data representation. The resultant plugin of this version will be published on Wordpress's official plugin repository, so it will be available for public download. From now on, we will identify this plugin version under the name of *"General Comment Plugin"*.

The other version is oriented for the startup called Kibiwoo. In it, we will implement also a sentiment and emotion analysis on users comments. In this case, Kibiwoo's administrator will be the one who configures all the senpy parameters. In particular, we will use the MeaningCloud Senpy plugin. For using it, we have needed to register a developer account on MeaningCloud platform. Meta-data associated to each comment will also be related to a product and a shop, as afterwards that information will be rendered to the shop owner. This version will only be available for Kibiwoo's owners use, and they will be on charge of explaining its usage to the shops that participate on their Marketplace. From now on, we will identify this plugin version under the name of *"Kibiwoo Comment Analysis Plugin"*.

[2]https://developer.wordpress.org/reference/classes/wp_comment/
[3]https://codex.wordpress.org/Function_Reference/add_comment_meta

4.2.1 General Comment Analysis Plugin

We can see in Fig. 4.1 a general overview of the architecture implemented by the General Comment Plugin.

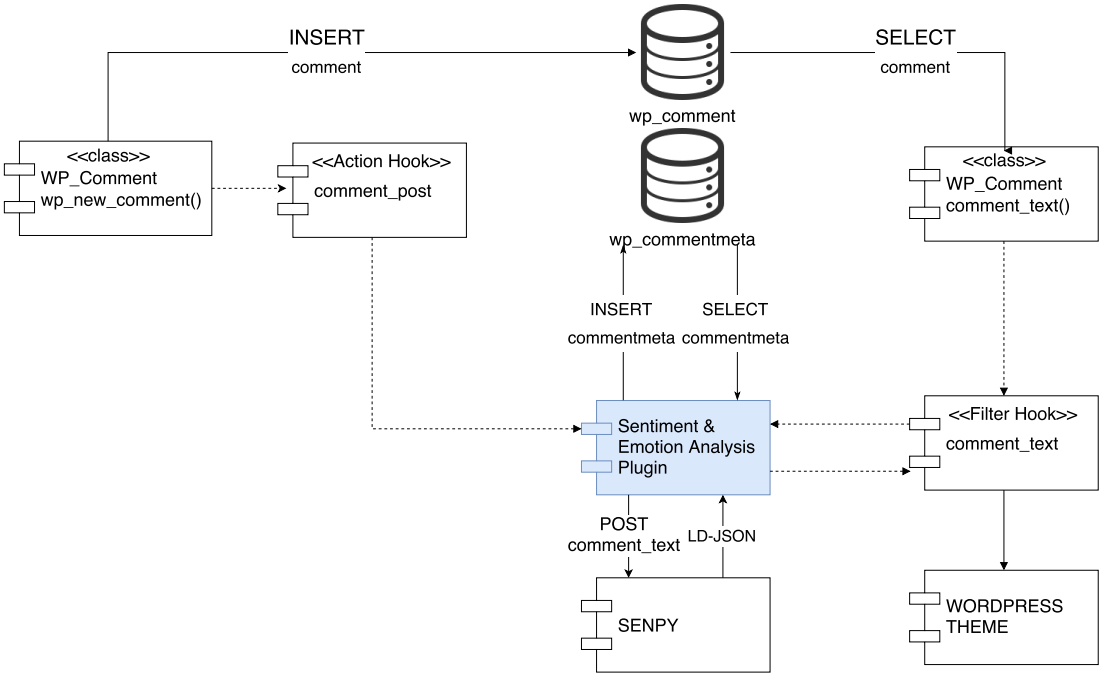


Figure 4.1: General Analysis Plugin Architecture

As we can see, this module uses the action hook *"comment_post"*[4]. This action is triggered immediately after a comment is inserted into the database, when the function wp_new_comment()[5] is called. It has two properties, the comment_ID and the boolean or int value comment_approved that indicates whether the comment has been approved or not. After a comment is inserted in the database, this action hook is fired and all the functions hooked to it will be executed.

We will obtain the comment_text value which contains the text the user has added as a comment and we will make a POST request to Senpy configuring the required parameters. The user will be able to configure several parameters in the plugin menu displayed in the admin dashboard. Sentiment and emotion senpy plugins will be selected by the user between the senpy available plugins. Furthermore, depending on the selected plugin a language parameter will also be chosen by the user and, if needed, a key for some plugins will be required. In Fig. 4.2 we can see how this options menu looks like on the admin's Wordpress dashboard.

[4]https://codex.wordpress.org/Plugin_API/Action_Reference/comment_post
[5]https://developer.wordpress.org/reference/functions/wp_new_comment/

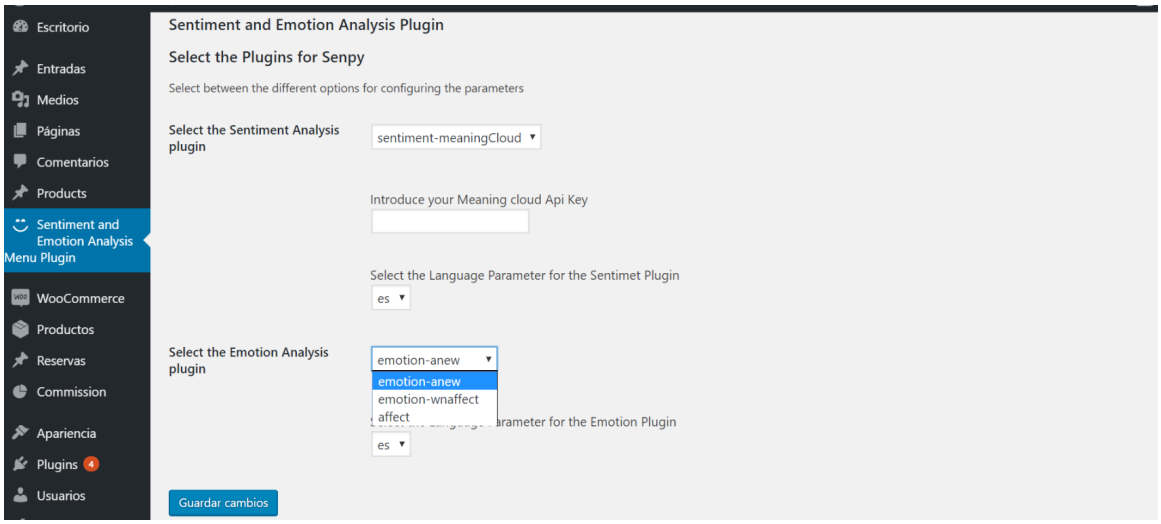


Figure 4.2: General Analysis Plugin Menu

Senpy will return a JSON-LD formatted object containing the comment evaluation. We will extract the comment's sentiment and emotion, and store it into the wp_commentmeta table under the keys "marl:hasPolarity" and "onyx:hasEmotionCategory". As Senpy information is described using the semantic vocabularies specified in Sect. 2, we will save what we want with the vocabulary category in order to provide standardization to the plugin information and facilitate its integration and communication with other systems.

We can see in Diag. 4.3 an overview of this first process, when a comment is analysed.

Then we have the comment's view part, where we will show the meta-data saved previously in a user-friendly way.

For this purpose, we will hook another function. This time we will use a filter hook to change the comment text that is normally returned, and add the information we want to be retrieved as well. When a comment is going to be showed, firstly, Wordpress selects the comment from the wp_comment database. Before the comment text is displayed, we can hook our function to the "comment_text" Filter Hook[6], make any changes we want and then return the comment text altered to be displayed. This hook is triggered by the comment_text()[7] Wordpress's function.

The user will have a page on its admin dashboard to select how the analysis information of the comment will be displayed. Its important to notice that only the comment text part can be changed, as there is only a hook available for that reason. If you want to change the whole comment's template, you need to alter the theme's files that define the comments

[6]https://developer.wordpress.org/reference/hooks/comment_text/
[7]https://developer.wordpress.org/reference/functions/comment_text/

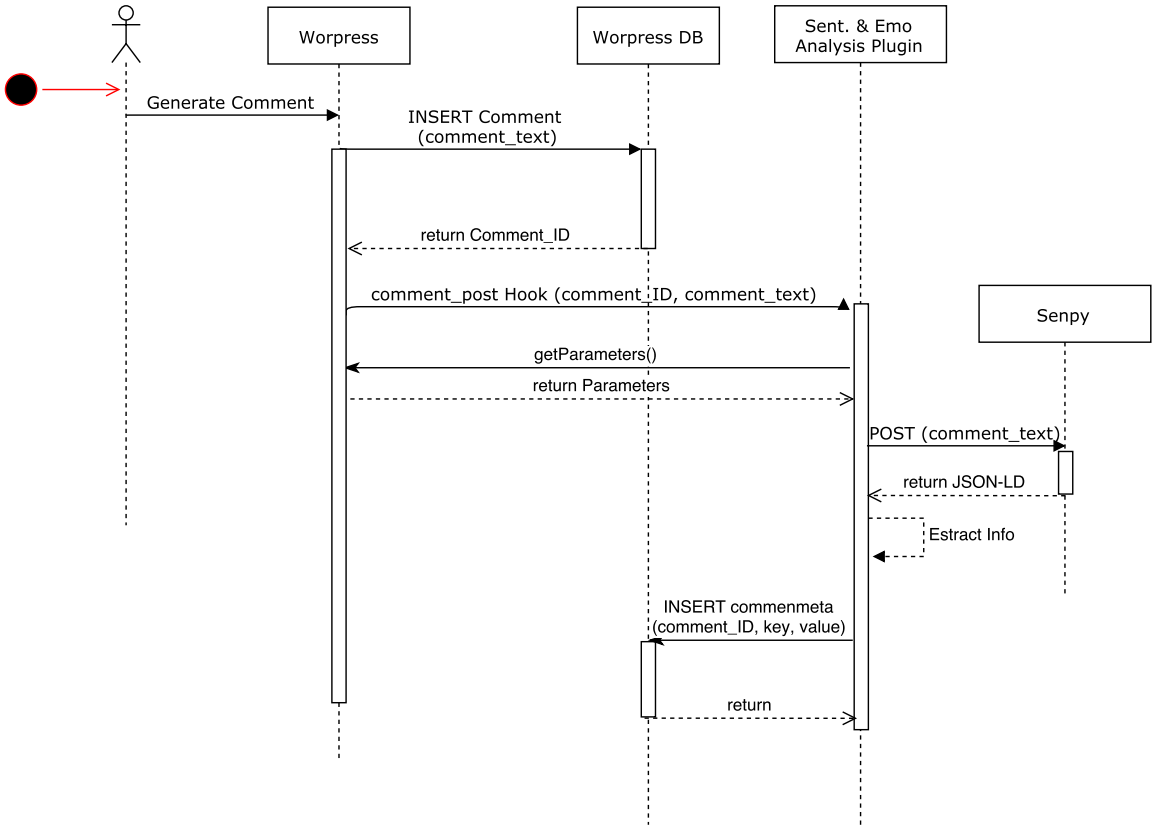


Figure 4.3: Analyse Comment Diagram

view [13]. As we want our plugin to easily integrate with other systems and themes, we will only make use of available hooks so that our plugin can interact with all Wordpress's themes. In Fig. 4.4 we can see an example of the comment's view with two different themes. In concrete, you can see the Storefront Theme[8] and the Zerif Lite Theme[9].

As we can see, the area surrounded by the blue rectangle is the part that users will be able to customize.

We have two tabs for personalizing the view of the comment. The first one is a selection part only, where users will select what they want to be shown with comment text. The possibilities are to show the general sentiment, the emotion expressed, both of them or neither of them.The second tab, is focused for more experienced users with some HTML and CSS knowledge. Here they can configure more deeply the html of the comment text, and add whatever they want. Helpful tips will be available in order to teach users how to output the meta-data analysis information.

[8]https://woocommerce.com/storefront/
[9]https://themeisle.com/themes/zerif-lite/

Figure 4.4: Comment View Examples

As we can see on the right part of the architecture in Fig. 4.1, when a comment is going to be output, Wordpress's themes call the comment_text() function. This function applies the comment_text filter hook so all functions hooked to it will be fired.

In our case, we receive the comment's text. First of all, we obtain the plugin's visualization options defined by the user. According to what has been selected, we form the HTML and CSS necessary to display the comment as selected. Then, we return the comment's text with all our HTML and CSS added. Finally, the theme receives what we have returned and outputs it.

The main reason and advantage of using the comment_text filter Hook, is that the comment's text is only changed when is going to be showed in a browser, but that information will not be altered in the database. Although this can be detrimental in terms of page's time of charge, when changing the selection of plugins, themes, or any other configuration, it will make it easier to integrate it as we are separating the comment text itself saved in the wp_comment table and the analysis information that is saved in the wp_commentmeta table. Also if we want to delete the analysis information or change it for another one, if we saved this data within the comment in the wp_comment table we will need to remove the entire comment including the comment's text itself so no further analysis will be able to be carried on.

We can see in Diag. 4.5 an overview of the process carried out when outputting a comment.

4.2.2 Kibiwoo Comment Analysis Plugin

Kibiwoo is a renting Marketplace of sports equipment. It enables physical shops to rent their products through the Marketplace platform. As an additional tool for shops, they will be able to see a summary of users sentiments and emotions among their products and

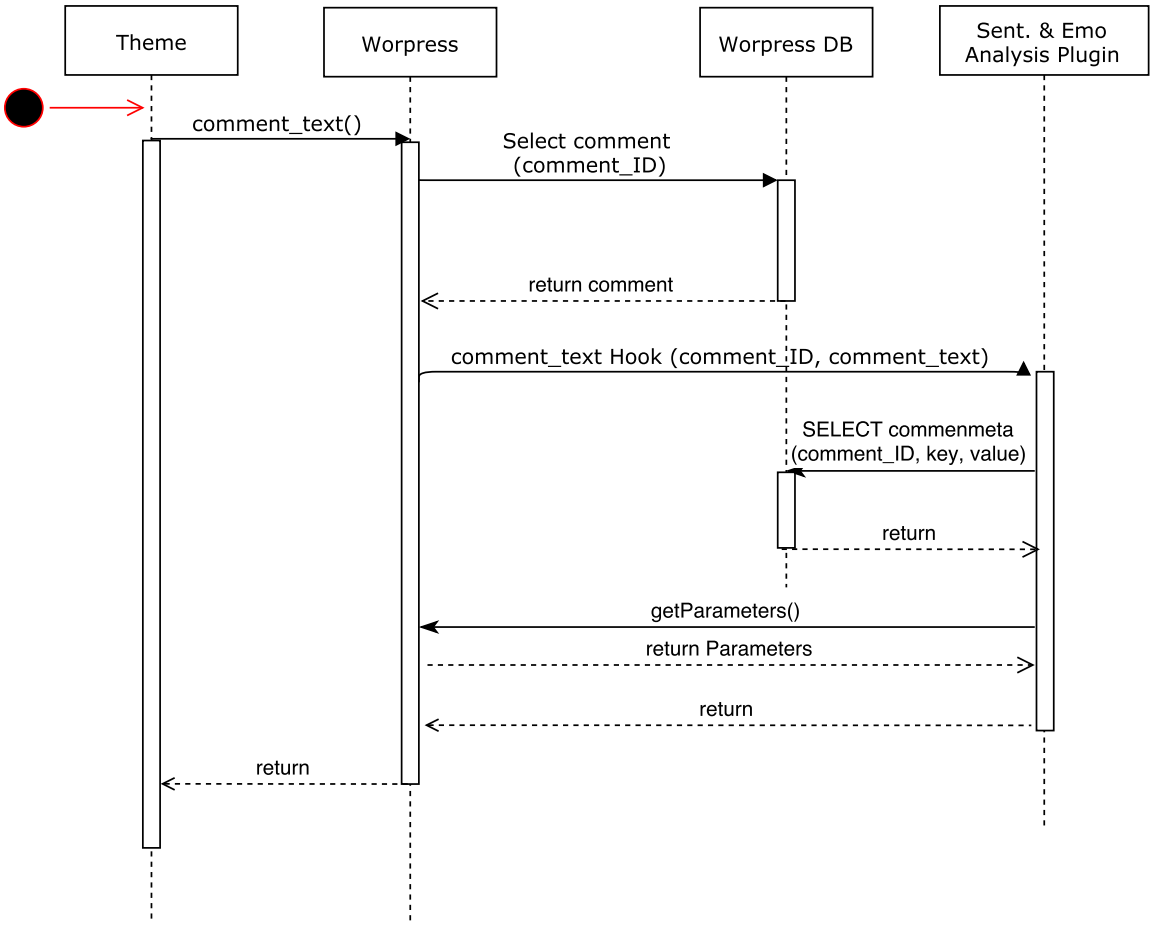


Figure 4.5: View Comment Diagram

services.

So for reaching that necessities and requirements we will develop an aspect based sentiment and emotion classification system that will render the data generated in different charts that will be defined by Kibiwoo's admin. Moreover, this data will be related to a product and as that product is related to a shop, the data will also be associated to a shop. This way shops will be able to:

- See a general overview of the sentiments and emotions expressed by users.
- Identify users that are happier or annoyed with their services and products.
- See all these classifications filtered by the different products they have

This tool will enable Kibiwoo to create a differentiation value among its competitors.

This plugin version has the same architecture as shown in Fig. 4.1, but configuration and the way data is stored is different

For analysing the information, we will make use of Senpy as well, but instead of selecting between different plugins, we will use the meaningCloud Senpy plugin for this issue. This plugin can divide a comment into different and independent simple sentences that may express distinct emotions or sentiments. Each sentence is evaluated separately. This is a fundamental characteristic as in the future we will be able to extract topics from the simple independent sentences and associate specific opinions to each topic detected on these not complex sentences, that may differ from the general emotion or sentiment detected for the whole comment.

In Listing 4.1 we can see an example of Senpy's output using meaningCloud.

Listing 4.1: meaningCloud Output Example


```
{
  "@context": "http://test.senpy.cluster.gsi.dit.upm.es/api/contexts/Results.jsonld",
  "@id": "_:Results_1496400624.04",
  "@type": "results",
  "analysis": [
    "plugins/sentiment-meaningCloud_1.0"
  ],
  "entries": [
    {
      "@id": "_:Entry_1496400624.04",
      "@type": "entry",
      "emotions": [],
      "entities": [],
      "sentiments": [
        {
          "@id": "Opinion0",
          "@type": "sentiment",
          "marl:hasPolarity": "marl:Negative",
          "marl:opinionCount": 1,
          "marl:polarityValue": -1
        },
        {
          "@id": "Opinion1",
          "@type": "sentiment",
          "marl:aggregatesOpinion": "Opinion0",
          "marl:hasPolarity": "marl:Negative",
          "marl:polarityValue": -1,
          "nif:anchorOf": "La%tabla era buena, sin20embargo la atenci\u00f3n fue mala",
          "nif:beginIndex": "0",
          "nif:endIndex": "52"
        }
      ],
```

```
      "suggestions": [],
      "text": "La%tabla era buena, sin20embargo la atenci\u00f3n fue mala",
      "topics": []
    }
  ]
}
```


In this version, different type of users will interact with the system, and different type of permission will be associated to each user. We can see an overview of different user's actions in Diag. 4.6

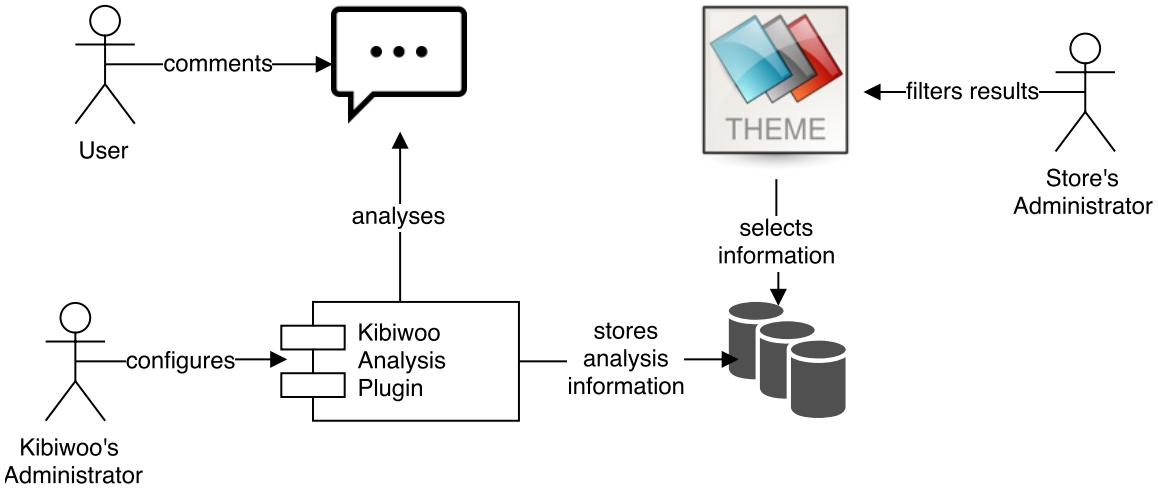


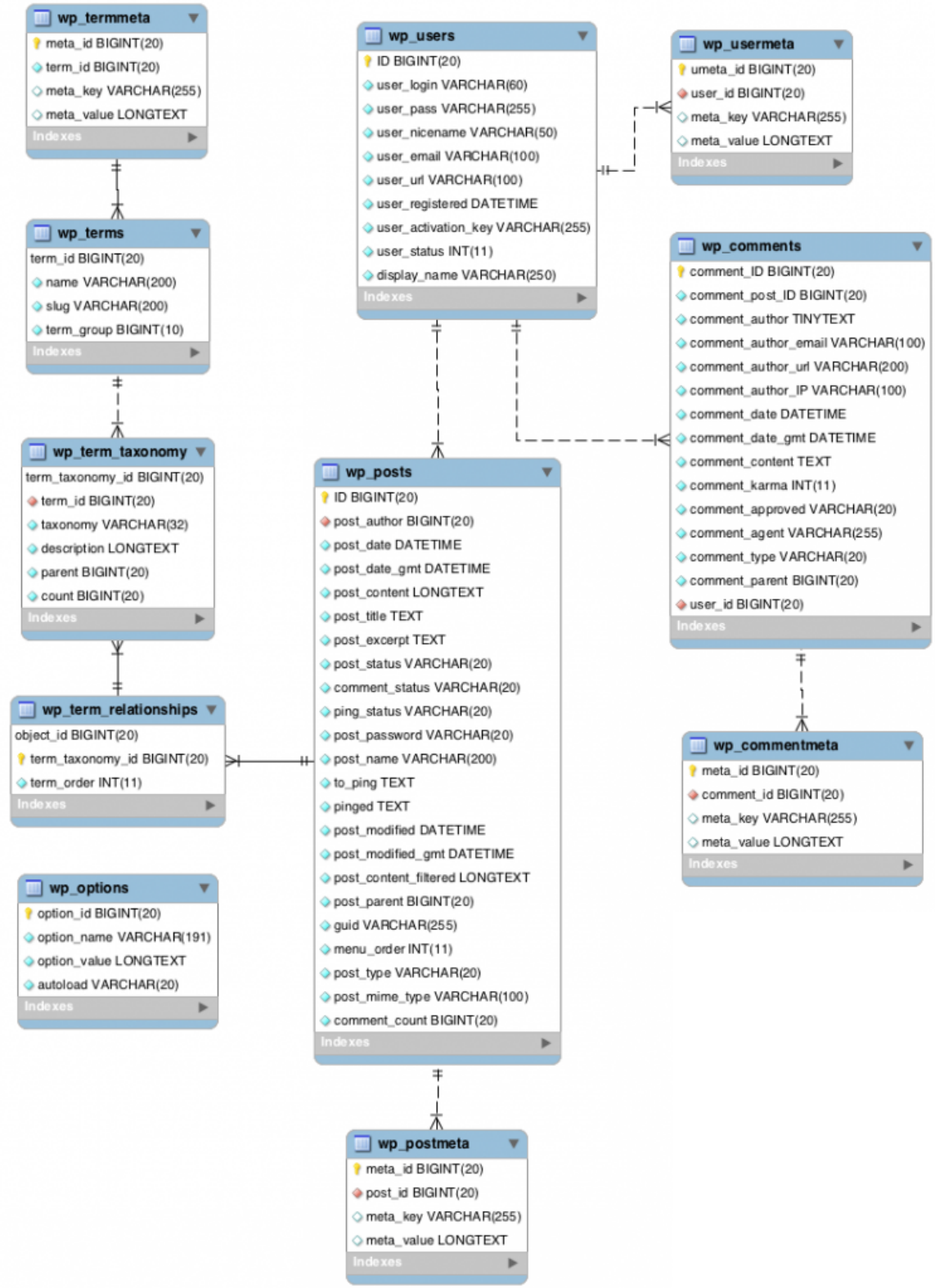
Figure 4.6: User's Actions Diagram

First we have Kibiwoo's administrator, who will be the one in charge of configuring the parameters of the plugin and what information will be stored and which not.

Then we have the shops administrators, who will be able to filter the results. Also he will be able to select and add a widget with information about comment's analysis in his dashboard.

Finally we have the customer, the user that only rents products. This type of users will not be able to configure any parameters of the plugin. They will only see the analysis results on each product and by shop while renting through Kibiwoo's Marketplace.

In Diag. 4.7 we can see an overview of Kibiwoo's database architecture.



Source: https://codex.wordpress.org/Database_Description

Figure 4.7: Kibiwoo Database

4.3 Data Visualization System

For this module, we will make use of wpDataTables plugin combine with Wordpress Plugin API and Shortcode API[10]. Shortcode API enables us to create shortcodes. Shortcodes are Wordpress-specific code that enables you to nifty things with little effort, such as embed content or create objects that would normally require lots of complicated, ugly code [14]. Before inserting shortcodes in our pages or posts, we need to define them using the add_shotcode()[11] function. We can see in Listing 4.2 an example of a shortocode definition.

Listing 4.2: "Shortocode Definition Example"

```
<?php
  //Register a new shortcode: [kibiwoo shop="kibiwoo"]link
     title[/kibiwoo]
  add_shortcode( 'kibiwoo', 'ag_aseg_shop_shortcode');

  //Callback function that will replace [kibiwoo]
  function ag_aseg_shop_shortcode( $attr, $content ) {
    $shop = $attr['shop'];
    $content = esc_html( $content );

    //Link to Kibiwoo's shop page.
    return "<a href='http://kibiwoo.com/$shop'>$content</a>";
  }
 ?>
```

This way, we can insert the shortcode [kibiwoo shop="shop"]My shop[/kibiwoo] in any part of a page, post or widget and it would be replaced with a link to the specified shop page.

The way wpDataTables works when creating tables and charts is shown in Diag. 4.8.

As we can see, wpDataTables stores the tables and charts it generates in Wordpress's database. In particular, it defines 3 new tables, the wp_kbw_wpdatables, where tables parameters are stored, the wp_kbw_wpdatatables_columns, where columns for each table are piled and has a foreign_key for associating the columns to each table, and the

[10] https://codex.wordpress.org/Shortcode_API
[11] https://developer.wordpress.org/reference/functions/add_shortcode/

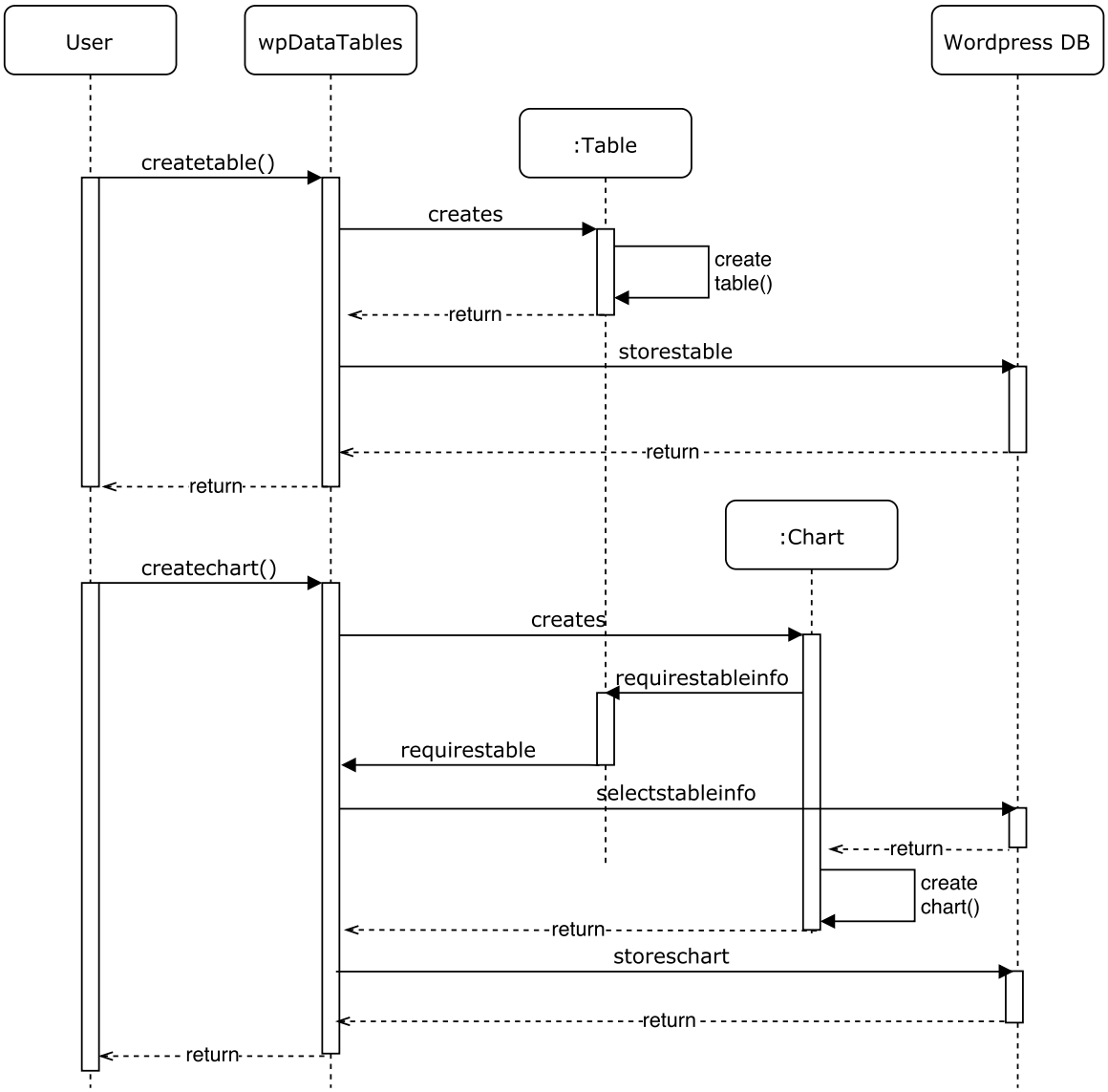


Figure 4.8: wpDataTables Creating Tables or Charts Diagram

wp_kbw_wpdatacharts, that stores the charts and is related to tables by its table_id.

For inserting the tables or charts in the stores dashboard we will use shortcodes. We can see in Listing 4.3 and Listing 4.4 some examples.

Listing 4.3: "Table Shortcode Example"

```
//id of the table
[wpdatatable id=1]
```

Listing 4.4: "Chart Shortcode Example"

```
//id of the table
[wpdatachart id=1]
```

This shortcodes are replaced with the html code defined in the functions of each shortcode, as shown in Listing 4.2, when the page, post or widget is going to be rendered.

We can see the data visualization system architecture in Fig. 4.9.

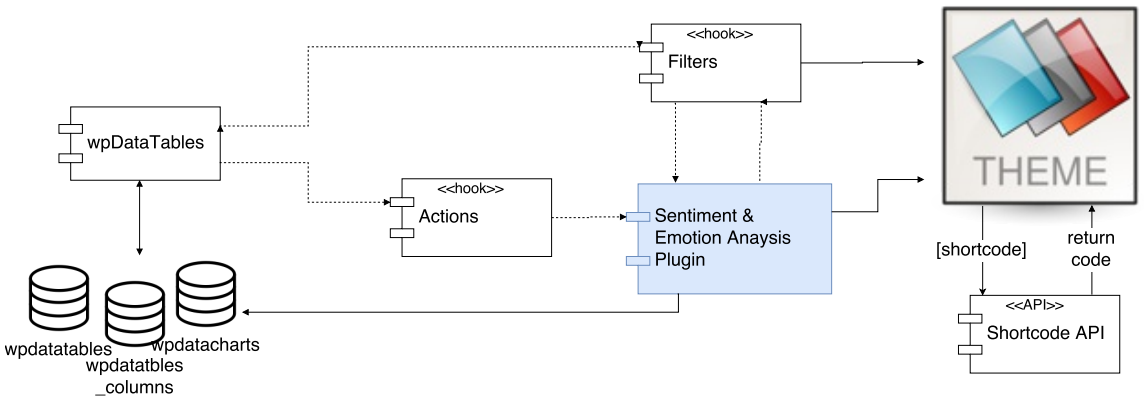


Figure 4.9: Data Visualization System Architecture

When introducing tables or charts, we only introduce the shortcode within the page. wpDataTables defines the shortcodes so that the Shortcode API replaces the shortcode with its html code. It is important to know that these charts and tables are dynamic. this means that when new data (products, users, reviews...) is introduced to the database, these tables automatically update with the new data and modify their view.

We will interact with the tables and charts via the filters and actions available. Our functions will be hooked to select the appropriate data depending on the user.

4.3.1 Gathering Data For Each Shop

In Kibiwoo's Marketplace, each shop will be able to manage products, payments and bookings. Each shop will have at least one administrator user, that will be on charge of the shop's Kibiwoo account. A shop can have as many administrators as it wants, but normally they will be only one of them. the relation between products, shops, users and bookings is

stored in the database.

Products are stored as posts table with *post_type='product'*. Shops are stored in terms and term_taxonomy tables with *taxonomy='wcpv_product_vendors'*. Products are related to shops in term_relationship table, where the post_id is linked to the term_id. And users are associated to shops in the termmeta table where the user_id is associated to each shop by its term_id.

For the comments part, each comment is related to a product in comments table where the post_ID is linked to the comment_ID. The analysis information for each comment is stored in commentmeta table.

So knowing the relationships between the different tables,we know can build the appropriate sql statement for extracting the data for each shop, and send it to wpdatatables plugin so that it outputs the information in a table. We have to notice that each shop and administrator has a different ID so we need to extract the ID depending on the user that is logged in.

In Listing 4.5 we can see the SQL statement for showing the overall sentiments among all products of a specific shop, in this case a shop called Kibiwoo.

Listing 4.5: "SQL Statement For extracting Data about a Specific Shop"

```
SELECT t.name, commeta.meta_key, commeta.meta_value, COUNT(com.
   comment_ID)
FROM wp_kbw_terms AS t
INNER JOIN wp_kbw_term_taxonomy AS tax
ON t.term_id=tax.term_id
INNER JOIN wp_kbw_term_relationships AS r
ON r.term_taxonomy_id = tax.term_taxonomy_id
INNER JOIN wp_kbw_posts AS post
ON r.object_id = post.ID
INNER JOIN wp_kbw_comments AS com
ON post.ID = com.comment_post_ID
INNER JOIN wp_kbw_commentmeta AS commeta
ON commeta.comment_id = com.comment_ID
WHERE tax.taxonomy='wcpv_product_vendors'
AND commeta.meta_key='marl:polarity'
AND t.term_id = 15
```

```
GROUP BY commeta.meta_value
```

This way we extract all the comments related to the products of one shop and group them by the sentiment they express. We can see the resultant table in Fig. 4.10

Show 10 entries Search:

name	meta_key	meta_value	COUNT(com.comment_ID)
Kibiwoo	marl:polarity	marl:Neutral	1
Kibiwoo	marl:polarity	marl:Positive	3
Kibiwoo	marl:polarity	marl:Negative	2
name	meta_key	meta_value	COUNT(com.comment_ID)

Figure 4.10: global Sentiments For a Single Shop

It is important to notice that this table is responsive and you can filter the results on the same page. As user filters the results, the table automatically updates in the moment. Also when databases are updated or modified, also the data for the table is updated. this way we have a completely up to date table.

For analysing each product separately, we will build a table containing all the products for one shop, and the sentiments associated to each shop. The SQL statement for generating that table is shown in Listing 4.6

Listing 4.6: "SQL Statement For extracting Data about a Specific Products of a Shop"

```
SELECT post.post_title, t.name, commeta.meta_key, commeta.
   meta_value, COUNT(com.comment_ID)
FROM wp_kbw_terms as t
inner join wp_kbw_term_taxonomy as tax
on t.term_id=tax.term_id
inner join wp_kbw_term_relationships as r
on r.term_taxonomy_id = tax.term_taxonomy_id
inner join wp_kbw_posts as post
on r.object_id = post.ID
inner join wp_kbw_comments as com
on post.ID = com.comment_post_ID
inner join wp_kbw_commentmeta as commeta
on commeta.comment_id = com.comment_ID
where tax.taxonomy='wcpv_product_vendors'
```

```
AND commeta.meta_key='marl:polarity'
AND t.term_id = 15
GROUP BY post.ID, commeta.meta_value
```

The table we obtain is shown in Fig. 4.11. In this table we can filter the results for viewing only one particular product. we can filter the products by its name. An example is shown in Sect. 4.3.2.

Show 10 entries Search:

post_title	name	meta_key	meta_value	COUNT(com.comment_ID)
Tabla de LongBoard	Kibiwoo	marl:polarity	marl:Positive	1
Tabla Surf	Kibiwoo	marl:polarity	marl:Neutral	1
Tabla Surf	Kibiwoo	marl:polarity	marl:Positive	2
Tabla Surf	Kibiwoo	marl:polarity	marl:Negative	2
post_title	name	meta_key	meta_value	COUNT(com.comment_ID

Figure 4.11: Sentiments For Each Product of a Shop

4.3.2 Showing Information To Shops

After building the tables, we create charts to show the information to the shop's administrator in a friendly-way. Charts gather data from the tables we have created. This way, as tables update when any modification to the information or the database is made, the charts will also update as their source of information are the tables we have created.

We have multiple options of charts styles and render engines. We can select between Google Charts, Highcharts [8] or Chart.js [6] rendering engine.

In Fig. 4.12 we can see an overview of all the sentiments expressed on Kibiwoo's shop products by users. This chart is synchronized with table shown in Fig. 4.10

In Appendix A we can see an example of how this results are shown to a shop and the filters it can make to them.

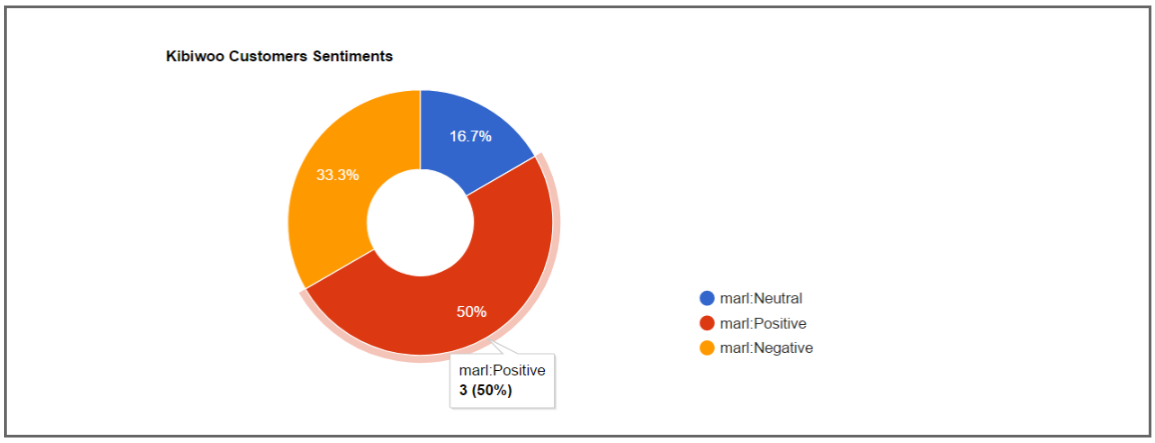


Figure 4.12: General Sentiments Overview Chart For a Shop

4.4 Semantic Identification with Microdata

Giving our page semantic identification will improve our SEO. WooCommerce plugin introduces some slight Microdata to pages but very little.

As we explained in Sect. 4.2, we want our General Plugin to work with any Wordpress theme. So as we need to alter the theme's files to introduce Microdata, in this version we will not introduce any semantic mark-up.

But as Kibiwoo uses a personalized theme, we will introduce microdata in products, reviews and stores. For that we will need to alter the theme's files. We will use the schemas for Products, Reviews and Stores.

Introducing microdata will improve Kibiwoo's SEO and hence shop's SEO also. This will make a difference among Kibiwoo and its competitors.

CHAPTER 5

Case study

This chapter, describes the case study oriented to Kibiwoo. We make an overview of the tools described through the project and the way they are integrated into a real business case.

5.1 Kibiwoo

We have explained what Kibiwoo is through this project, now we are going to show visually how it looks. In Fig. 5.1 we can see Kibiwoo's landing page. Here we can select between the different categories available and search for any product we want.

When you make a search, all the products from the different shops which match that search will be shown. Users can compare all the products. All of them will have the overall score extracted from the reviews that users have made about that specific product. Users will be able to leave a comment on a product only if they have rented that product and the rent has finished already. In the product page, the comments will be shown the same way as in Fig. 4.4.

Users will have the possibility to pay with Credit Card or PayPal. Each type of user will have a different dashboard and possibilities.

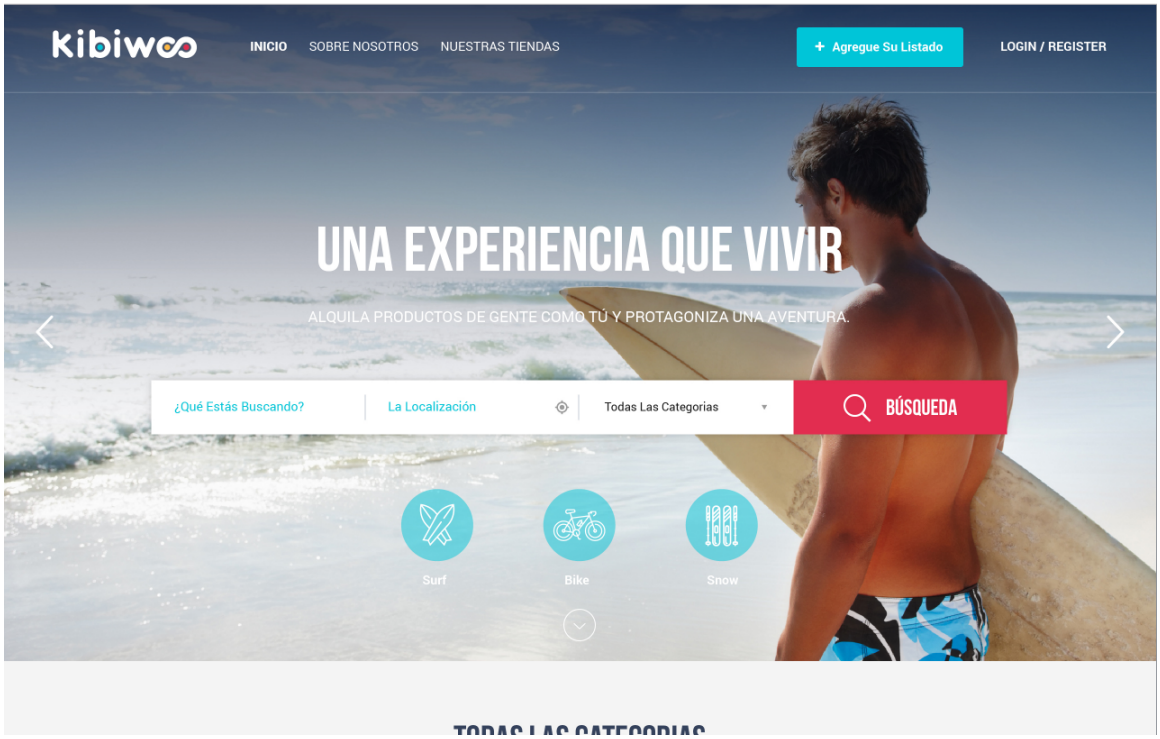


Figure 5.1: Kibiwoo's Landing Page

5.1.1 Kibiwoo's Admin

Kibiwoo's Admin will be able to configure all the parameters for the analysis plugin, manage users, products, reviews, payments, commissions and all the aspects of Kibiwoo.

Each time a user is registered, makes a booking, leaves a review, uploads a new product or pays, Kibiwoo's admin is notified. From its dashboard he will be able to control all of these aspects. Moreover, he can see an overview of Kibiwoo revenues by month or year, the revenues generated by shop, product or category and have a lot more useful filters. We can see in Fig. 5.2 Kibiwoo's admin dashboard. In Fig. 5.3 we can see an overview of Kibiwoo's year sales.

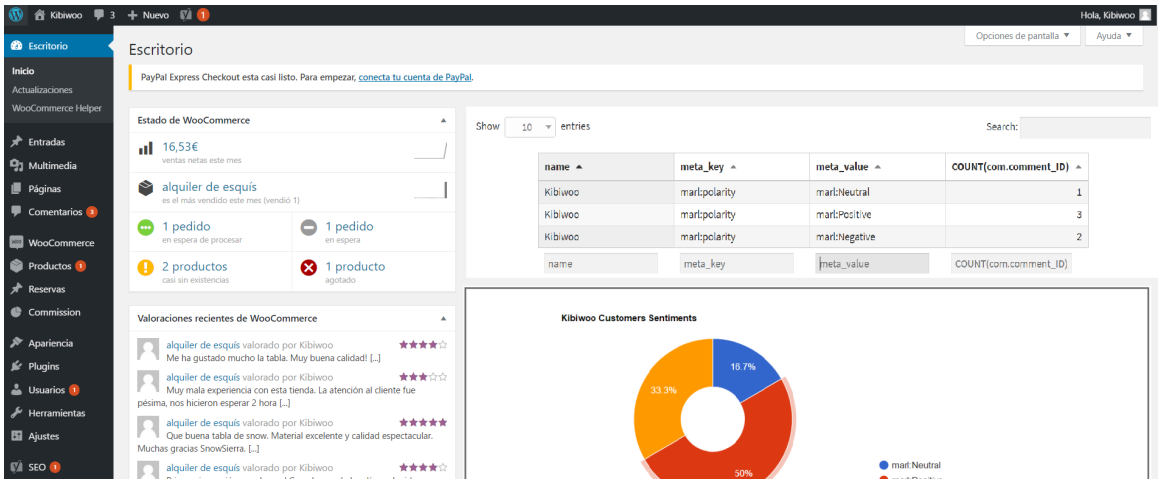


Figure 5.2: Kibiwoo's Admin Dashboard

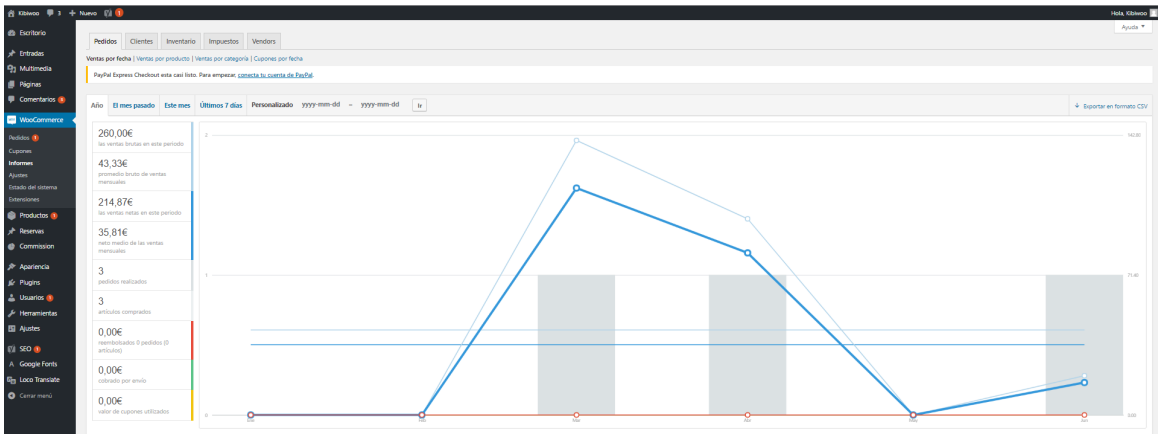


Figure 5.3: Kibiwoo's Year Sales Overview

5.1.2 Shops

As we explained in Chap. 4, shops will be able to filter results by product. Shops will have a different dashboard to Kibiwoo's Admin dashboard. Shops will be able to manage their products, see the bookings of their products, payments, configure their account and see an overview of the sentiments and emotions expressed by users among their products. They will also be able to see a sales overview similar to the shown in Fig. 5.3, but only those related to the sales associated with their products.

We can see in Fig. 5.4 the profile page of a store's administrator.

In Fig. 5.5 we can see a shop's dashboard and all the different options it has.

Figure 5.4: Shop's Administrator Account

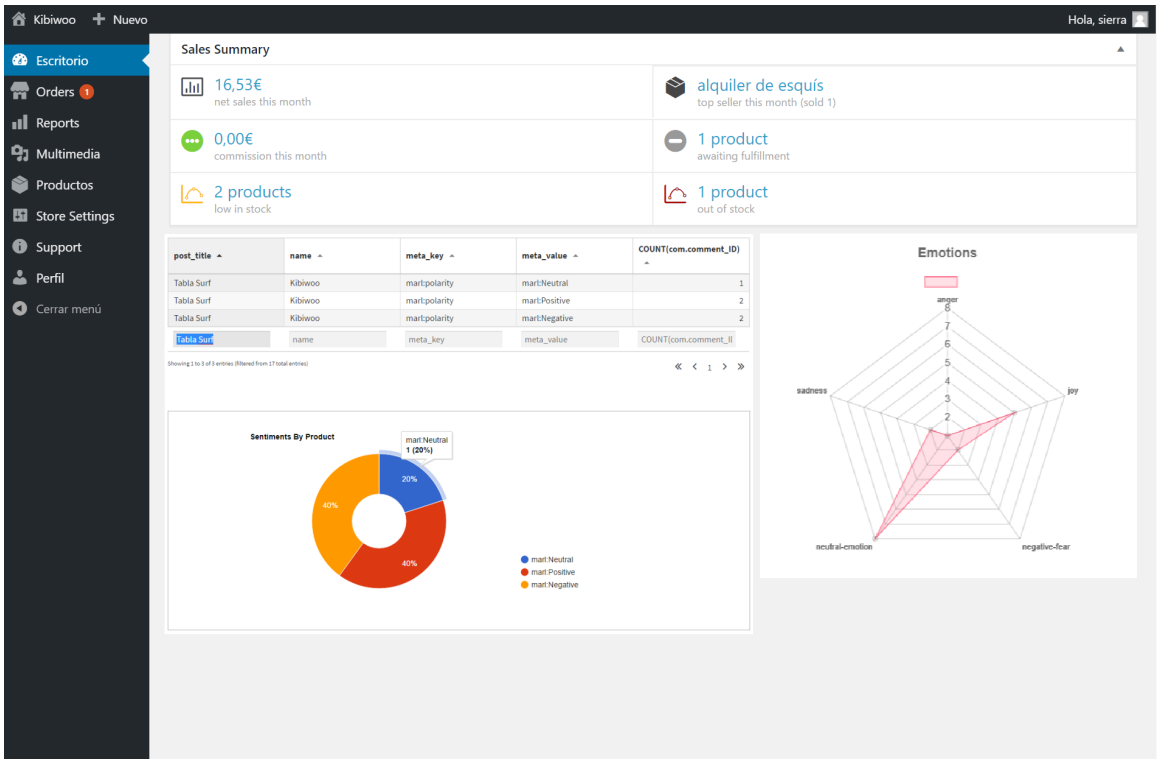


Figure 5.5: Shop's Dashboard

5.1.3 Users

Users will be able to book products and see an overview of their orders and payments. They will also leave comments on products as long as they have rented that specific product. The main page of a user account can be seen in Fig. 5.6. And in Fig. 5.7 we can see the bookings of that user.

It is important to notice that any person can surf through Kibiwoo's Marketplace without being registered. What is more, a non-registered user will be able to make a booking, but when he is going to pay, he will be required to register in Kibiwoo first. Comments can only be left by registered users that have booked a product.

Figure 5.6: User's Account

Figure 5.7: User's Account Bookings

5.2 Conclusions

In this chapter we have shown a few examples on how Kibiwoo's different users an benefit from our system and can introduce themselves in the new technology era. Moreover, no coding knowledge is needed for them.

We have checked that all the modules combined function correctly. The Sentiment and Emotion Analysis part works correctly analysing the comments and storing the information genrated in the Wordpress's database.

We have made a brief overview of the possibilities of each user and what can be done in each case.

CHAPTER 6

Conclusions and future work

In this chapter we describe the conclusions extracted from this project and the future work that is going to be carried.

6.1 Conclusions

In this project we have created two versions of a Sentiment and Emotion Analysis plugin for Wordpress. The first version is oriented to general users of Wordpress and can be downloaded from Wordpress's repository plugins. The other one, has a business orientation. We have shown how can this technologies be integrated in a real business example, in concrete, in a renting market niche.

We have two main parts of the system, the sentiment and emotion analysis system, and the visualization system. The first system, uses Senpy for analysing the comments that are left on the platform and then stores the information in Wordpress's database. The visualization system is in charge of extracting that data from the database and outputting it to the shops. Moreover, it provides responsive and interactive charts and tables that update when the user filters the results.

Finally, we have modified Kibiwoo's template by inserting semantic tags with Microdata

format in order to improve Kibiwoo's SEO and hence their shops SEO so that shops see a strong added value in being register at Kibiwoo's Marketplace.

6.2 Achieved goals

In Chap. 1 we list some goals for the project. Now, once we have finished the project, we can analyse if each of them has been achieved or not.

Design and deployment of a Social Renting Marketplace based on Wordpress CMS that allows shops to manage their products and bookings, and users to compare similar products from nearby shops and book the one that fits them best. This objective has been reached successfully. Now we have shops and users interacting with Kibiwoo's Wordpress Marketplace. This Marketplace is deployed in http://kibiwoo.com/wordpress and a few bookings have been closed through it.

Development of a Wordpress plugin for emotion and sentiment analysis on customers reviews. Development of a semantic dictionary for sentiment aspect-based classification oriented for shops. This objective has been reached partially. Two versions of the sentiment and emotion analysis plugins have been developed as explained in Chap. 4. An aspect-based classification is proposed as future work for extending the functionality provided for shops.

Representation of the analysis information on a user-friendly way so that shops can interprete the data and take action. Responsive and interactive tables and charts have been developed, and what is more, we have carried out several proofs in which shop's administrators have interacted with the system and a process for introducing small changes has been carried out until they have been completely satisfied with the system and its interface. So we can say that complete success in this objective has been reached.

Introduce semantic content in the web using Schema's vocabularies in order to improve the SEO and the web positioning of the shops that take part in the Marketplace. Kibiwoo's theme has been modified for introducing schema vocabulary tags with Microdata format. This was the main objective as adding this function to the general version plugin could result in a poorer integration with the available themes as this aspect needs to be introduce by altering the theme files

directly. So finally we opted for prioritizing integration in this version. So we can say that this objective has been achieved too.

6.3 Future work

In this section i will introduce future work that can be done from my point of view

Aspect-Based Analysis System This is the first thing to be done in the future. It generates a lot of value to shops. The idea is to associate sentiments and emotions to topics related to the shop's services. This topics shoul be price, location, quality, time and customer service. A first work for this aspect has been done as the plugin developed for Kibiwoo separates a full comment into simple sentences and provides an independent analysis for each sentence.Then, only the aspect-based system that identifies these aspects in sentences needs to be done.

Adapting the general version plugin for other languages As Wordpress is the main CMS used in the world, an adaptation for other languages should be done. some work in this aspect has be done as all the outputted strings have been enclosed in Wordpress's functions that depending on what language is selected it will translate it or not. Now, the dictionaries that determine all the sentences translations need to be made and a selector language option needs to be integrate it.

Analysing data from other sources such as RRSS Some shops have few comments in Kibiwoo's platform so extracting information from the comments in other platforms or social sites will enrich the tools provided to the shops.

Integrating a System to follow the situation of a bookingor product The idea is to have a simple interface where you can see easily the state in real time of each booking or product.

Identifying users behaving models The idea is to identify within the information gathered from users comments and bookings users behavioural models for anticipating to them and improving marketing campaigns.

Bibliography

[1] schemaorg/schemaorg.

[2] Ben Adida, Mark Birbeck, Shane McCarron, and Steven Pemberton. Rdfa in xhtml: Syntax and processing. *Recommendation, W3C*, 7, 2008.

[3] World Wide Web Consortium et al. Json-ld 1.0: a json-based serialization for linked data. 2014.

[4] World Wide Web Consortium et al. Rdf 1.1 concepts and abstract syntax. 2014.

[5] Mike Dean, Guus Schreiber, Sean Bechhofer, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L McGuinness, Peter F Patel-Schneider, and Lynn Andrea Stein. Owl web ontology language reference. *W3C Recommendation February*, 10, 2004.

[6] Nick Downie. Chart. js documentation. *Dostopno na: http://www. chartjs. org/-docs (marec 2014)*, 65:66, 2014.

[7] Sunghwan Mac Kim, Alessandro Valitutti, and Rafael A Calvo. Evaluation of unsupervised emotion models to textual affect recognition. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 62–70. Association for Computational Linguistics, 2010.

[8] Joe Kuan. *Learning Highcharts 4*. Packt Publishing Ltd, 2015.

[9] Alistair Miles and Sean Bechhofer. Skos simple knowledge organization system reference. 2009.

[10] Matt Mullenweg, R Boren, M Jaquith, A Ozz, and P Westwood. Wordpress, 2011.

[11] Jason Ronallo. Html5 microdata and schema. org. *Code4Lib Journal*, 16, 2012.

[12] J Fernando Sánchez-Rada, Carlos A Iglesias, Ignacio Corcuera, and Óscar Araque. Senpy: A pragmatic linked sentiment analysis framework. In *Data Science and Advanced Analytics (DSAA), 2016 IEEE International Conference on*, pages 735–742. IEEE, 2016.

[13] Brad Williams, David Damstra, and Hal Stern. *Professional WordPress: Design and Development*. John Wiley & Sons, 2015.

[14] Brad Williams, Ozh Richard, and Justin Tadlock. *Professional WordPress Plugin Development*. Wrox Press Ltd., Birmingham, UK, UK, 1st edition, 2011.

APPENDIX A

Showing Sentiments and Emotions to Shops

We are going to see some examples of a analysis by different products. We will use the data shown in Fig. 4.11. We will show in Fig. A.1 and Fig. A.2 how the chart updates with the table when the table's results are filtered.

post_title	name	meta_key	meta_value	COUNT(com.comment_ID)
Tabla Surf	Kibiwoo	marl:polarity	marl:Neutral	1
Tabla Surf	Kibiwoo	marl:polarity	marl:Positive	2
Tabla Surf	Kibiwoo	marl:polarity	marl:Negative	2
Tabla Surf	name	meta_key	meta_value	COUNT(com.comment_II

Showing 1 to 3 of 3 entries (filtered from 17 total entries)

« ‹ 1 › »

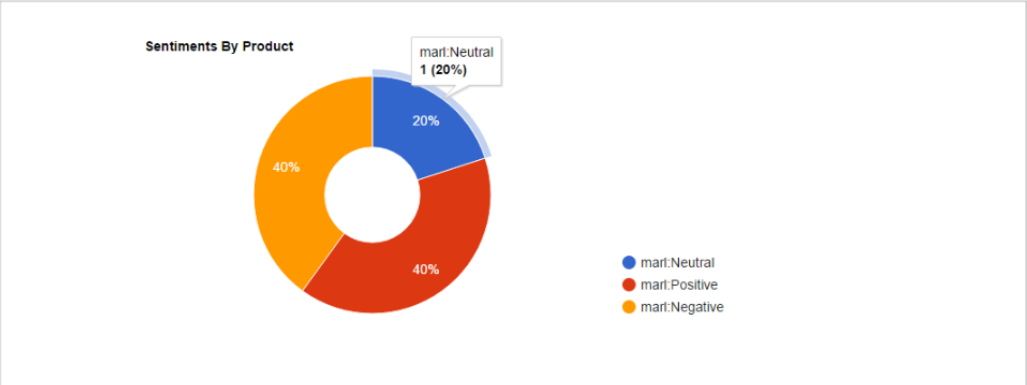


Figure A.1: Sentiments for Tabla Surf

post_title	name	meta_key	meta_value	COUNT(com.comment_ID)
Tabla de LongBoard	Kibiwoo	marl:polarity	marl:Positive	1
Tabla de Longboard	name	meta_key	meta_value	COUNT(com.comment_II

Showing 1 to 1 of 1 entries (filtered from 17 total entries)

« ‹ 1 › »

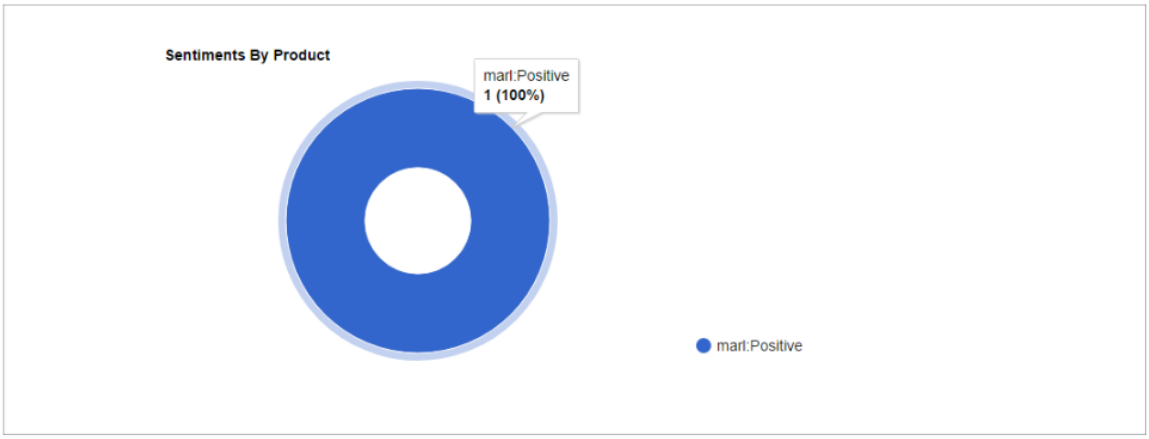


Figure A.2: Sentiments for Tabla de Longboard