UNIVERSITÀ DEGLI STUDI
DELL' AQUILA

TESI DI LAUREA

"Enhancing e-Commerce applications through Internet of Things and real usage data mining"

Corso di Laurea Specialistica

Candidato:
Fabio Righi

Relatore:
Dott. Romina Eramo

Abstract

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

Acknowledgements

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

Contents

Introduction

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

Chapter 1

Background

1.1 The Internet of Things and the Web

During the last few years, and in an increasing manner, we have been standing at the forefront of a world where virtually everything can be connected directly to the Internet. This has resulted in an increasingly connected world where emerging technologies enable objects to be linked among themselves through the use of new devices and sensors.

This new era of the Internet has become visible in our everyday lives: from souped-up gadgets tracking our every move to environments capable of predicting our actions and emotions, the list keeps growing every day.

The Internet, consisting of things, rather than just computers, is becoming more central to society than the web as we once knew it. This does not necessarily mean that the traditional web is expected to die, but rather that its role will be reduced to that of a language used for displaying content on devices which are supposed to be more ubiquitous but are not as necessary.

The first “pioneer species” within this ecosystem of ”invisible buttons” has certainly been the smartphone. Its increasing usage among the masses has made it the perfect catalyst for such a revolutionary change. One of the many uses that has helped us better understand this pioneering technology is the way that it beams information about our location and speed when we take it with us in a car resulting in a real-time traffic information accessible by everyone. In such a scenario, the actual gathering of real traffic data happens without the user ever knowing of the data transmission and without a need for interaction such as a click on a button or navigating to a particular web page.

This sort of awareness, especially as it relates to the physical world, leads to areas in space that are listening to the environment and triggering events depending on certain conditions in an entirely automatic way, like a smartphone getting into the range. There are currently applications in which the smartphone can be placed as close as two centimeters away from the top of a credit card reader to enable touch payments, or where the smartphone is detected in a large space, such as a room, triggering an event indicating that a user has entered or exited so that the lights can bed switched on or off.

It is important to differentiate these interconnected objects from being simple on-off switches; they would not be very useful if this was the case. However, because the possible actions they can trigger can be affected by an endless list of other variables such as the time of the day, our personal preferences or the actions of others, they can quickly be scaled to reate a better program that creates more efficient interactions in our physical world.

Leveraging the use of a smartphone acting as a proximity sensor is just an artifact of the current state of technology. The same result could be accomplished with any number of sensors directly connected to the Internet so long as those sensors are capable of dealing with motion, sound, light temperature, humidity, and other variables.

Companies like Apple have been embracing the idea of invisible buttons since the beginning of this new era of sensors and devices.

While the company has been embracing the technology by foreseeing its potential from the beginning, it recently rolled out a new line of devices called iBeacon.

In a nutshell, the iBeacon allows any newer iPhone or Android phone to know its position in space with centimeter precision. Similarly to a more precise Global Positioning System (GPS) that works indoors, an iBeacon allows the developers to take advantage of the technology to define “invisible buttons” of just about any dimension.

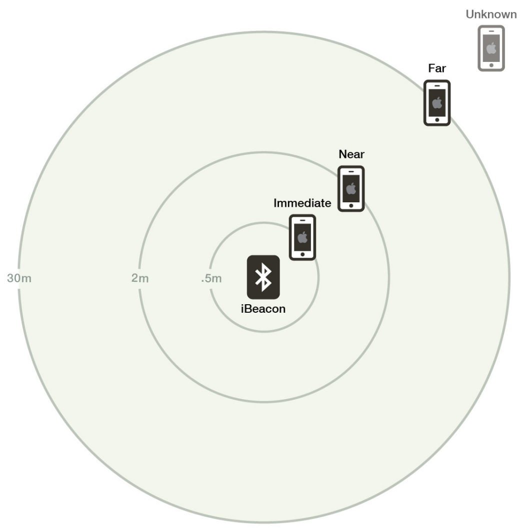


Figure 1.1: Distance categories for iBeacon

In fact, nothing is stopping this technology from being squeezed into something as small as a typical credit card, or from being embedded in any clothing or other discrete wearable devices such fitness sensors, wristwatches or even temporary tattoos. The opportunities are indeed countless. Whether we wear those sensors or use them in our homes and businesses (see smart thermostats, lighting and security systems for example), they can all be prepared and trained to cooperate in sophisticated and unexpected ways once the Internet knows that we are present nearby and what our intentions might be. Imagine a smart home capable of knowing when we wake up based on the activity monitor on our wrist and begin warming up the house, brewing a pot of coffee and switching off the security system. It is evident that with such a capacity for sensing and responding to our needs, the Internet of things is slowly shaping a brand new world capable of being alive in ways that it has never been before.

Figure 1.2: A smart home ecosystem

1.2 Data and web mining

Very often, company management requires selecting the most adequate Business Intelligence solutions that will fit its needs in order to perform crucial strategic decisions.

One of the tools used for this particular goal is a technique called data mining, which is the result of a continuous evolution that has been occuring during the last thirty years of data review. Up until the late 1970s, Business Intelligence decisions carried out their role through the use of standardized reports which contained simple summarized data and analysis.

In the early 1980s, companies began to query data in more detailed and complex ways. This made it easier to detect patterns based, for example, on an individual product or geographic area.

Currently, the advanced software available on the market for data mining is capable of performing real time pattern detection on a vast quantity of data thrown at it. This expedites a company's decision making processes and the creation of robust long-term strategies.

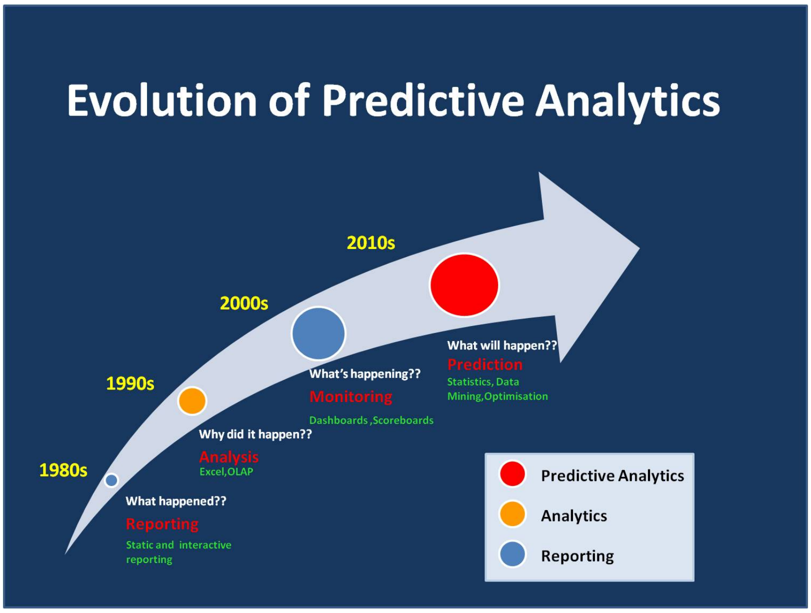


Figure 1.3: Evolution of Predictive Analysis over the last 40 years

Thorough interpretation and analysis of the available data allos the data mining process to create a better overall understanding, and helps in making better decisions.

In fact, thanks to advanced examination techniques, it is possible to find hidden information, create analytical models and data groups, and identify relationships among activities while also correcting errors.

All of this certainly leads to real advantages for a company leveraging these processes, both in terms of revenue and cost. For example, on the side of income, data mining allows companies to identify and classify the best, real and potential customers, discover additional sales opportunities, increase economic productivity and find new ways and new solutions to grow. In parallel, regarding the aspect of cost, the process could maintain clientele by identifying customer loyalty elements, reducing exposure to non-payment risks and distributing resources more efficiently.

For an organization, the reasons behind using data mining may be different. The unifying point, however, is the need to derive insight from the data that will guide the transformation, reorganization or innovation of business processes. It is evident that decisions based on accurate and reliable knowledge are always the best. Data mining, in fact, provides exactly this type of information. While Enterprise Resource Planning (ERP) systems improve operational efficiency, they do not provide the strategic drive for business growth or business change. Warehousing systems can efficiently store data, but they lack the tools to transform those figures into valuable information focused on reporting and answering mostly static questions such as areas where the company has sold the most. On the other side, data mining tools try to present a solution to a wider range of more interesting problems, such as why sales are not taking off as expected, why customers prefer competitors or which previous marketing campaigns had the best outcomes.

Understanding the answers to these questions means taking the right measures to improve the business's performance.

Besides data visualization techniques, one of the most popular data mining processes on the market is based on the simplified transposition of the neural networks and the neural process of the human brain: when presented with models, the brain understands that some patterns are associated with other desired results. Similarly, artificial neural networks are capable, by learning about sets of historical data called learning sets, to generate patterns and validate them on other subsets of data called test sets. They operate iteratively by modifying patterns from time to time to reach an optimal solution, and they have the ability to evaluate and provide feedback on unknown data, thus making them very useful for forecasting and classifying knowledge. They are very often presented as a "black box" approach to data mining, and they turn out to be very useful when creating parametric models that are difficult to construct and when the emphasis is on forecasting rather than explaining complex patterns.

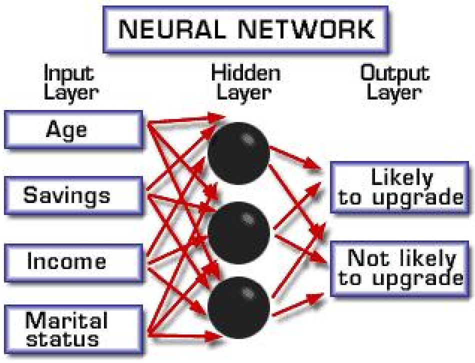


Figure 1.4: Neural networks learn to predict outputs after proper training and weighting.

After discovering what data mining means, who uses it the most, and how it is usually implemented, it is important to focus our attention on the utilization of this tool on the web and its characteristics in such an environment.

In fact, web data mining can detect behavioral models of website visitors, generate reports and implement actions based on those identified patterns. This process is possible because website visitors often unknowingly provide information about themselves and how they respond to the content presented. Monitoring what links they click, where most of their time is spent, what search terms are entered and when the visitors leave the website are just a few appetizers of the endless stream of possible data inputs.

Some visitors also provide information about their lifestyle or personal information such as names and addresses.

Because of all of this, a thorough and adaptive analysis of this considerable amount of stored information is fundamental, and this is exactly where web data mining kicks in by helping to design the web shopper behavioral model and making valuable predictions.

One of the features that unquestionably contributes to the strength of data mining is the ability to combine emerging traffic data to the site with those related to the transactions and the profile of the buyers.

Through these combinations and by highlighting the patterns that are uncovered, it is possible to both gather complex and strategic considerations and generate predictions that may be indispensable and vital for managing a website that wants to improve its business.

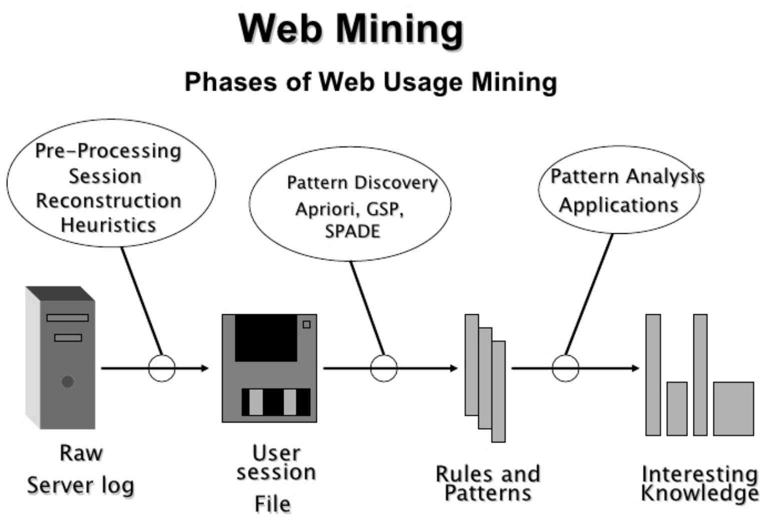


Figure 1.5: Phases of Web Usage Mining

Due to the heterogeneity nature of the source data, Web Mining is far from simply being an application of traditional data mining techniques. In fact, it can be categorized into three main types:

- **Web structure mining**: This focuses on analyzing and discovering useful knowledge from hyperlinks representing the structure of the Web. For example, these links allow us to detect relevant Web pages in a way similar to what search engines are already doing. Alternatively, it is possible to determine shared common interests among users and so on.

- **Web content mining**: The main goal of this technique is to extract valuable information or data from the content of the web pages. After doing so, it is possible to automatically classify and group this information according to topic area. While these tasks are apparently similar to those in traditional data mining, we still can discover relevant behavioral models using product descriptions, forum posts, customer reviews and much more.

- **Web usage mining**: This technique usually refers to the identification of user access patterns from Web usage logs once the sanitization and preprocessing of the clickstream data has occurred.

Although the web mining process is similar to traditional data mining techniques, the data gets collected in an entirely different way. In traditional data mining, the data is often already available and stored in a data warehouse, whereas for the web counterpart, the effort for the data acquisition can be a cumbersome task, especially for web structure and content mining. This is due to the potentially high number of links and large quantity of pages to crawl.

1.3 Model-driven techniques

Software development techniques are continuously evolving while also trying to solve the principal problems that still affect the building and maintenance of software: time, costs and susceptability to errors.

On this topic, one of the latest research trends in software engineering is the Model Driven Engineering (MDE) technique, which was born as an extension of more specific approaches such as the Model-Driven Architecture (MDA) of the Object Management Group (OMG).[1]

MDE's primary goal is to define the methodologies and techniques to support the process related to the entire lifecycle of software development through the manipulation of models.

Before proceeding further, it is beneficial to explain the difference between MDA, Model-Driven Development (MDD), MDE and Model-Based Engineering (MBE).

The first is, by all means, an OMG standard focused on software development and using a set of defined languages utilized for a specific purpose (e.g. UML[2]). On the other hand, the focus of MDD is still on software development, but is independent of mandatory language constraints to perform its tasks. MDE, as a superset of MDD, does not only drop the software-related restrictions, but it unties itself from a particular development process, therefore expediting the definition of model driven processes to facilitate a complete software engineering process. Finally, we use MBE to refer to a softer version and a superset of MDE where models still play an important role, but are not the central artifacts of the development process. (e.g. blueprints or sketches of the system handed out to programmers directly without automatic code generation) (Figure 1.6).

[1] The Object Management Group (OMG) is an international, open membership, not-for-profit technology standards consortium, founded in 1989. OMG standards are driven by vendors, end-users, academic institutions and government agencies.

[2] The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering, that is intended to provide a standard way to visualize the design of a system.

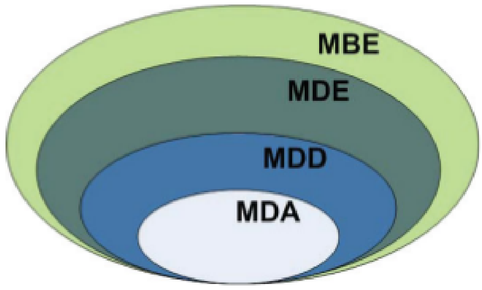


Figure 1.6: Relationship between the different MD* acronyms.

The model is at the center of any MDE process and, to be considered as such, the modeling language that generated it needs to have well-formalized syntax and semantics. This is a fundamental condition for the automatic transformation of models. In fact, the model represents the system and constitutes an abstract and conforming formalization to a particular language. Such a language is often tailored to a certain domain and is often called Domain-Specific Language (DSL).

The advantages of using the model driven approach rely on the consideration of the generating language as a type of scheme that is also fully modelable through a formal and abstract definition known as the meta-model. In fact, the model must represent an abstraction of the concepts of the system that needs to be built, while also respecting a meta-model likewise determined by the application domain.

This reasoning can be further iterated up to the determination of a model for the representation of languages used to formalize other languages (the meta-meta model).

Because of the above, model-driven approaches are often defined by models on a stack basis as shown on Figure 1.7.

One of the most commonly used techniques in MDE is the automatic model generation based on the definition of a set of automatic transformations defined by the starting and target languages. In a nutshell, the MDE approach consists of:

- **Models Generation**, based on modeling stacks definition supporting meta-modelling methods where lower-level models comply with what is specified by higher-level models.
- **Models Manipulation** performed through transformations expected to generate destination models which conform to destination meta-models when provided with source models that conform to source meta-models.

In summary, MDE techniques allow, on the one hand, for the management of the complexity of a system by ensuring increased productivity, and for quality improvements by promoting a higher level of abstraction and reuse, on the other. This result is achieved through a process of engineering, both for the languages and the transformations involved: they are, in fact, the basis of a system's development process because they represent the transition from higher-level models

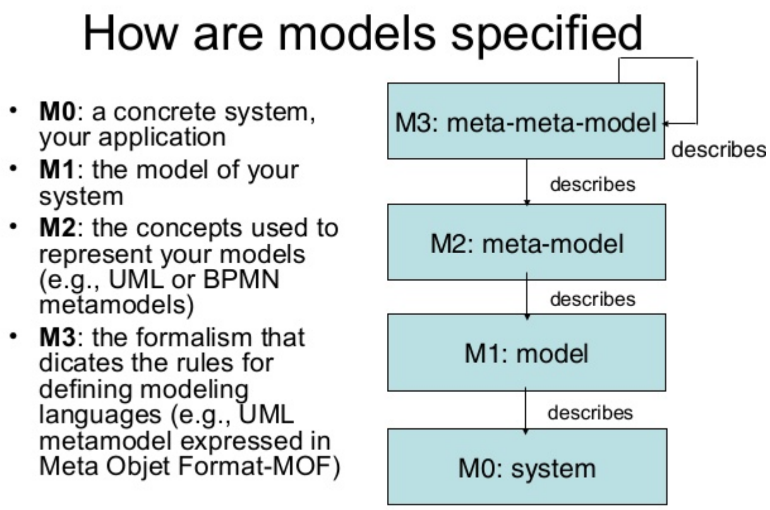


Figure 1.7: Models stack

into lower-level ones until they can be made executable using either automatic code generation or model interpretation.

Chapter 2

State of the art

2.1 Iot Devices : RFID, NFC, Beacons and Sensors

As per introduced in 1.1, The Internet of Things (IoT) describes an ecosystem or network of Internet-connected objects able to collect and transfer data using embedded sensors. This ecosystem is vast but can be organized into a number of major dimensions that are discussed below :

- **Radio Frequency Identification (RFID)** is a technology predominantly applied to one-way inventory tracking and supply chain problems. Packages are affixed with passive RFID tags containing important product information that are detectable to a distance of 100 meters by stationary readers. RFID tags are comprised of a small antenna and a silicon chip capable of storing the product information to facilitate the identification.The tags are remarkably small, and can be effectively integrated with adhesive labels attached to cases and pallets, incorporated into security cards, or even implanted in pets, to name just a few examples.

  The primary difference with the most common electronic barcode mechanism is that barcode readers require "line-of-site"; the barcode reader must see the barcode lines to read the data and can only do that one barcode at a time while the RFID does not require it as tags can be scanned through a variety of materials. From an IoT perspective, RFID readers (that can be handheld or mounted to a specific location such as a dock door) are a powerful mechanism to read and write data across networks of RFID tags and automatic transfer vast amounts of data to any variety of data clouds or backend systems.

- **Near Field Communication (NFC)** comprises a set of close-range wireless communication standards offering functions similar to the most common Bluetooth and RFID technologies. Much like RFID, NFC can detect and access data from special tags but have the additional advantage of being suitable for virtually unlimited applications because information can be easily retrieved from any conventional NFC-enabled device. Similar to Bluetooth technology, NFC supports two-way secure data exchange with a simple tap or wave among devices.

Currently, NFC is already incorporated into over 1 billion devices globally, including an increasing number of tablets, PCs, household appliances, electronic devices, gaming consoles and of course smartphones. For enhanced security and control, NFC operates only when devices are in close proximity (approximately 10 centimeters), thus making this technology optimal for more protected applications like financial transactions and secure login access at a particular location.

- **Beacons** collectively refer to small wireless devices that are capable of transmitting simple radio signals embedded within a unique identification number. At any time, a nearby device such as smartphones using Bluetooth Low Energy technology detects transmitted signals resulting in the reading of the beacon's ID, calculation of the distance to the beacon and, depending on the result, may trigger an action in a beacon compatible mobile app.

  Despite the simplistic mechanism, beacons represent a substantial technological advancement and have opened new opportunities to allow more precise tracking for indoor positioning and behavior compared with standard GPS technology.

  The simple communication associated with beacons, which importantly is not reliant on significant power consumption, unfolds a wide range of new seamless interactions with application in numerous different areas. The retail sector, for example, is currently one of the most popular fields in which beacon technology is utilized because it offers an unprecedented way to track in-store interactions between customers and product displays, ultimately resulting in more personalized offers and a better retail experience based on the acquired data. Additional applications of the beacon technology range from supporting and improving physical navigation in large spaces such as museums, airports and stadiums, to assisting impaired people in public transportation.

Figure 2.1: RFID, NFC and Beacon Tags

- **Sensors** are physical pieces of hardware responsible for monitoring processes, taking measurements and collecting data. There are literally infinite measurements that can be gathered by sensor arrays, ranging from temperature to proximity, from pressure to water quality, and from gas detection to liquid level tracking. A wide variety of devices currently include these sensors, and based on current trends we are witnessing a move towards multi-sensors platforms capable of simultaneously incorporating different sensing elements. In the retail sector, for example, it is possible to make the store shelf ”smarter” by providing visibility from the product’s arrival to final sale thanks to a combination of store shelf sensors, smart displays, digital price tags and high-resolution cameras. In fact, sensor-based technology now enables the retail businesses to precisely determine product volume on store shelves and in stock rooms, leading to more efficient product ordering and restocking. A final illustration of sensor application is the next-generation of personalized self-tracking products in the form of wearables and smartwatches. Examples include combinations of accelerometers, GPS, temperature and heart rate sensors that provide a comprehensive data insight on the activity of the user.

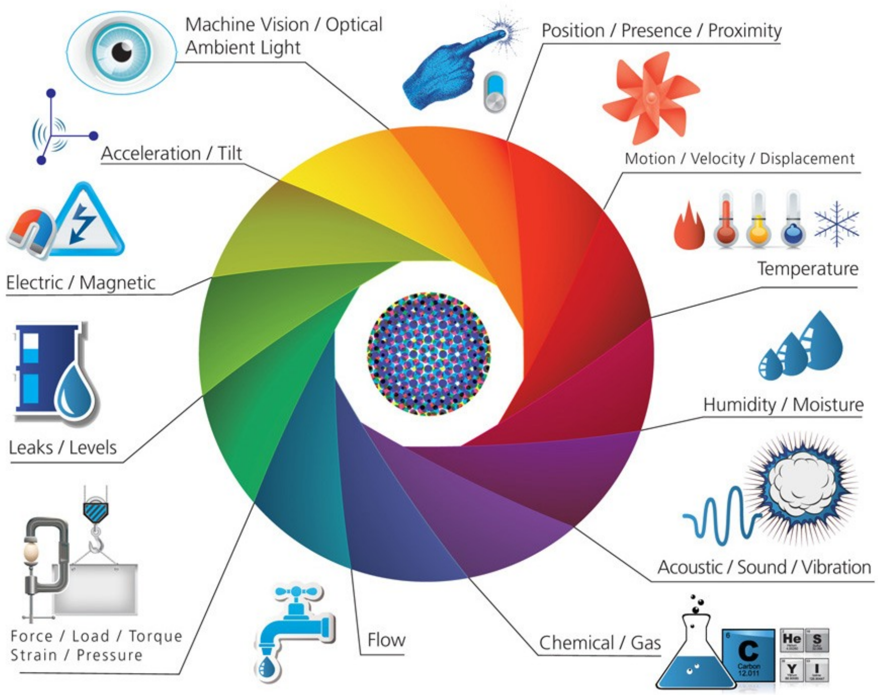


Figure 2.2: Sensors ecosystem

2.2 Big Data and Predictive Analysis applications

The concept of Big Data relates directly to the Internet of Things. Simply, IoT devices are capable of collecting terabytes of data over very short time periods, thus emphasizing the importance of being able to support the efficient processing and interpretation of data as it is collected.

It is important to note that in addition to the fact that IoT devices are capable of generating profile data at an incredibly high rate, the modern Web represents another essential data source in this paradigm (see chapter 1.2). Social networks, for example, are capable of producing an endless stream of data describing user preferences and behaviors at any time. This vast volume of "Big Data", available in a whole array of formats and growing constantly, provides unparalleled opportunities for addressing any number of socially-important questions.

To obtain economic value from Big Data, companies are investing in advancements in artificial intelligence and machine learning algorithms to efficiently process data, create products, and implement solutions that extend well beyond traditional systems currently used for managing and storing information. This includes novel approaches supporting high calculation inclination that are efficient at sorting data in a structured and meaningful manner to detect relationships and patterns in complex data streams. This new approach to data management differs from what companies used to do in the past when priorities were bound to an IT level governance only and they were solely accessed by a restricted set of users. Moreover, it also becomes vital for a company to identify new sources of big data when they become available on the market, and quickly incorporating them into the data management platform following a constant continuous improvement logic.

Figure 2.3: The five fundamental Vs of big data

Admittedly, it is challenging to forecast the added value brought by Big Data analysis to the various application disciplines in which they can be applied. However, it is clear that they certainly have improved the quality of the forecasts that contribute to more prudent decisions supported by more robust empirical evidence. In fact, Big Data analysis constitutes a fantastic instrument in the field of the decision-making by minimizing the risks and reducing the consequences caused by inadvertent human errors. For example, the marketing department of an organization could potentially leverage big data for increasing marketing intelligence predicting customer interests. At the same time, Big Data could be utilized to provide better forecasts of product stock replenishment and to optimize production requests.

In summary, the potential applications mentioned above represent the core of the predictive analysis notion: the practice of extracting information from Big Data with the goal of determining patterns and predicting future outcomes and possible trends.

It is also important to remark that this kind of analysis does not offer a precise assessment of what will occur in the future but rather they forecast the likelihood of particular outcomes with an acceptable level of reliability. This often includes what-if scenarios and risk assessments.

Focusing on the Marketing intelligence example mentioned above, the benefits of using Predictive Analysis are important to fully understand. Recently, digital companies are increasingly embracing this idea because it offers a better and more constructive experience to the customers on every possible point of contact between the business itself and the clients. This can increase customer loyalty and, by extension, economic returns.

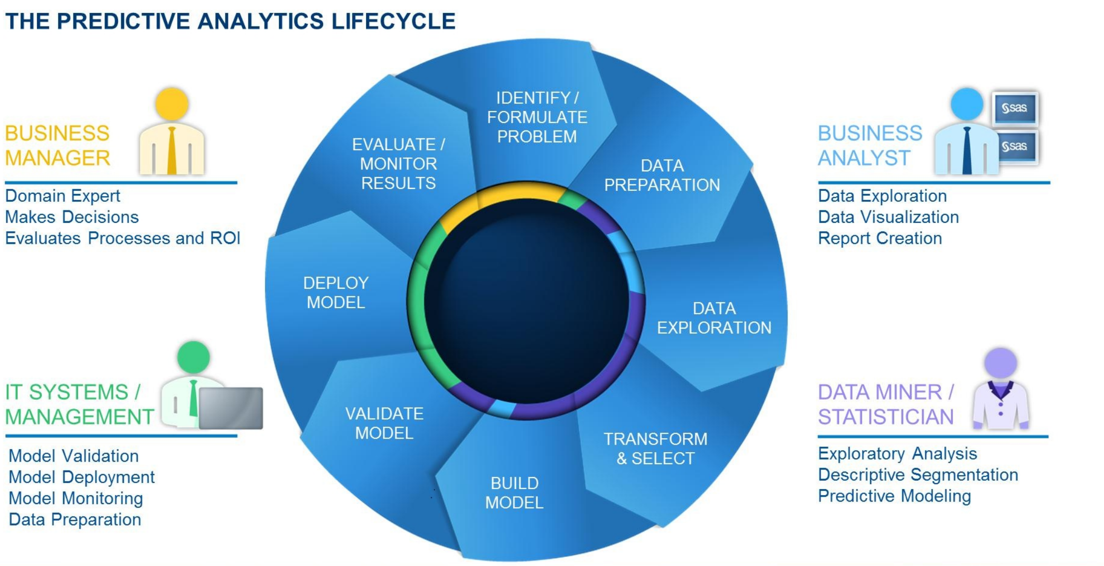


Figure 2.4: The predictive analysis lifecycle

In more detail, this outcome can be achieved using sophisticated algorithms and mathematical models on top of the big datasets of customers activity accessible from Big Data sources; eventually, this data gets sanitized, structured and filtered and finally grouped in a meaningful way.

The behaviors and patterns of interaction detected by this analysis process can indicate, for example, a more appealing product offer with a major chance of conversion for a particular customer profile segment.

Different typologies and techniques are used to perform enhanced analysis for behavioral prediction. Here, I discuss a few of the major approaches.

- **Clustering or Unsupervised learning**: This methodology seeks to group similar individuals and identifies with high precision based upon the enterprise's customer base. These algorithms can process hundreds of attributes with the goal of identifying those characteristics that best discriminate individuals according to their behavior. The underlying notion is that, statistically, distinct groups of individuals behave in similar ways.

  Group clusters obtained with this process are similar to groups determined through a priori classification; the only fundamental difference between the two methodologies is that clustering or unsupervised learning allows individuals to be categorized into different groups based solely on data and not pre-conceived attribute differences.

  Clusters can be of different types depending mostly on the criteria by which customers are grouped. Product-based clusters, for example, collate all customers who tend to buy products or combinations of several products in the same category. By contrast, brand-based clusters focus on grouping customers who prefer certain brands instead of others, and behavior-based clusters combine consumers with similar buying behaviors, helping the marketing manager to identify the most appropriate way to address each of them.

- **Propensity models or Supervised learning**: This family of techniques is based on probabilistic models using advanced machine learning techniques such as neural networks, logistic regression, random forest, and genetic algorithms. The main purpose of these procedures is to predict future customer behavior based on past examples. Over time, these algorithms become more efficient, improving predictions when more data is collected.

- **Reinforcement learning and Collaborative filtering**: This increasingly popular technique has been demonstrated in a number of major applications, one of the most famous being the ability of companies to recommend products to purchase. The recommendations are targeted and tailor-made for the client and they are defined by considering the entire relationship between customer and brand. For this reason, they can upsell for higher value products, cross-sell to same category items or dynamically link to other products based on the modeled associations.

Chapter 3

Combining data acquisition channels : the real usage data

The evolution of the Internet and the amount of data available from web usage mining and IoT devices has led to an enormous proliferation of the accessible information as per described in the previous sections. In this chapter, we focus on describing the possible channels of adquisition of customer behavioral data both in the virtual and the physical world. This valuable information represent the milestone of the journey towards the enhancement of the eCommerce experience for its usesr and aims to address some of the weaknesses of the standard approaches for web personalization.

Before we can efficiently represent content visualized on user interfaces, navigation paths and user-triggered events on a web application, we need to take a little detour briefly introudicing a standard modeling language capable of modeling such interactions which will be analysied with more detail in the next chapter.

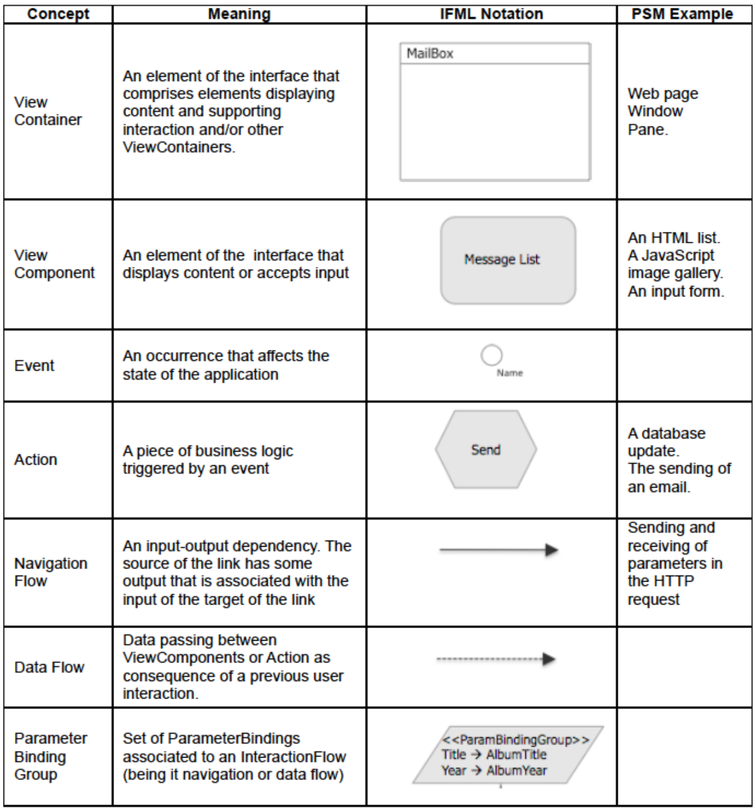
3.1 The IFML language

The Interaction Flow Modeling Language (IFML)[1, 2] is designed for describing and controlling the behavior of front-end software applications, it brings several advantages to the development process such as promoting the separation of concerns between roles and increasing the overall understanding of the product for non-technical stakeholders. To achieve so IFML supports formal specification for interface composition, user interaction and event management independently of the implementation platform and it was adopted as a standard by the Object Management Group (OMG) in March 2013.

IFML supports the following concepts:

- **The view structure** describes *ViewContainers*, their nesting relationships, their visibility and their reachability.

- **The view content** manages *ViewComponents*, i.e., content and data entry elements contained within ViewContainers.
- **The events** defines the *Events* that may affect the state of the user interface. *Events* can be produced by the user's interaction, by the application, or by an external system;
- **The actions** triggered by the user's events. The effect of an *Event* is represented by an *InteractionFlow* connection, which connects the event to the *ViewContainer* or *ViewComponent* affected by the *Event*. The *InteractionFlow* expresses a change of state of the user interface: the occurrence of the event triggers a change in the state that produces a transition in the user interface.
- **The navigation flow** indicates the effect of an Event on the user interface.
- **The data flow** indicates the data passed between *ViewComponents* and *Actions*
- **The parameter binding** illustrates the input-output dependencies between *ViewComponents* and between *ViewComponents* and *Actions*.



Concept	Meaning	IFML Notation	PSM Example
View Container	An element of the interface that comprises elements displaying content and supporting interaction and/or other ViewContainers.	MailBox	Web page Window Pane.
View Component	An element of the interface that displays content or accepts input	Message List	An HTML list. A JavaScript image gallery. An input form.
Event	An occurrence that affects the state of the application	Name	
Action	A piece of business logic triggered by an event	Send	A database update. The sending of an email.
Navigation Flow	An input-output dependency. The source of the link has some output that is associated with the input of the target of the link		Sending and receiving of parameters in the HTTP request
Data Flow	Data passing between ViewComponents or Action as consequence of a previous user interaction.		
Parameter Binding Group	Set of ParameterBindings associated to an InteractionFlow (being it navigation or data flow)	<<ParamBindingGroup>> Title → AlbumTitle Year → AlbumYear	

Figure 3.1: Main IFML concepts and notations.

3.2 Navigational modeling for the Web

eCommerce website front ends are usually built using shared and reusable components (forms, list views, detail views, etc.) which have a specific and expected behavior. For example, Product lists and grids show record details for the user to view and interact with an action on these, ”Add To Cart” call-to-action buttons are presented within product pages to trigger a different response and so on. All these interactive operations can be represented using the IFML notation.

To demonstrate the versatility and adaptability of this modeling language we introduce a real-life example which we will use as a reference from this chapter forward: an online boutique website called *”Madison Island”* specialized in fashion items running on an eCommerce platform.

Madison Island presents all the features of a typical eCommerce digital store including navigation and browsing of its catalog, product searching, customer account section, shopping cart and order processing.

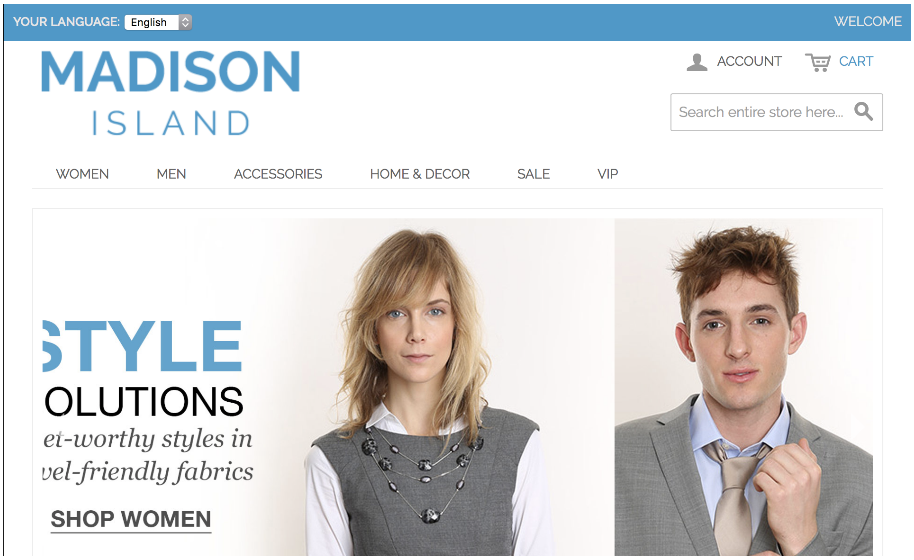


Figure 3.2: Madison Island digital store homepage

Figure 3.2 shows the home page of the website. In this section, the user can select one of the product categories, access his customer area, switch the language of the website, search for an item or go directly to the shopping cart.

In the following subsections, we analyze a couple of scenarios related to users navigational behaviors exposing both their representation in IFML notation and the associated server log entries in the application server.

3.2.1 The product page journey

Starting from the homepage the user can interact with the navigation menu to select from a set of categories available. (Figure 3.3)

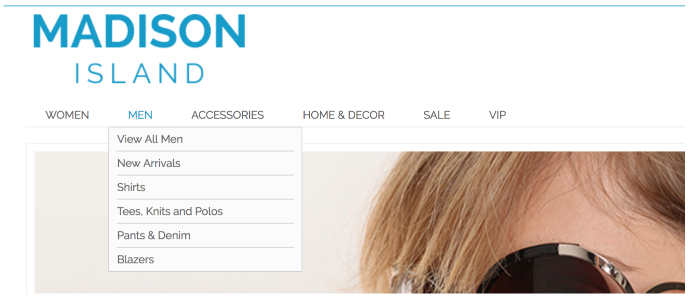
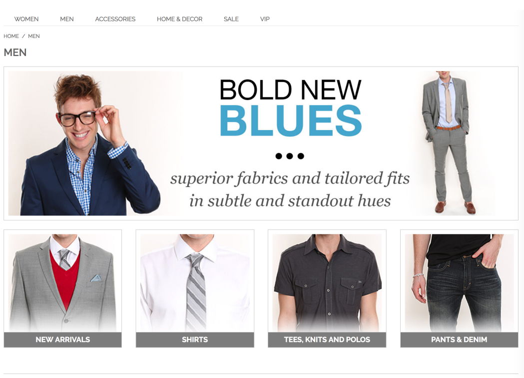
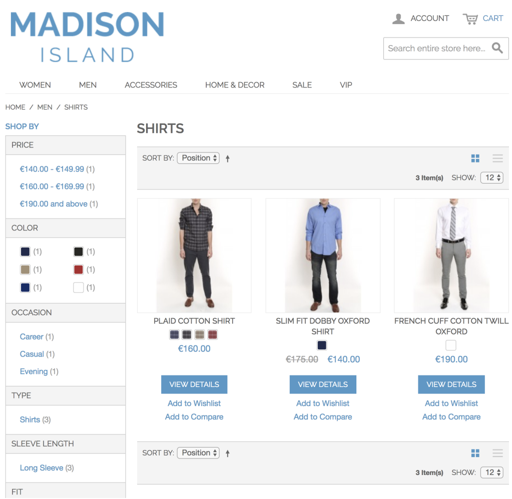


Figure 3.3: Navigation menu

Depending on the category display mode, a category page can either list CMS content showing a list of links to children categories serving as an intermediary transitional page or directly present its products to the user. (Figure 3.4)



(a) Category listing



(b) Product listing

Figure 3.4: Different category page view modes

Finally, from the product listing screen, the user can potentially access any of the product detail pages clicking on the call to action *te"View Details"* below the selected product image thumbnail, let's assume he chooses the *"Plaid Cotton Shirt"* in this case. (Figure 3.5)

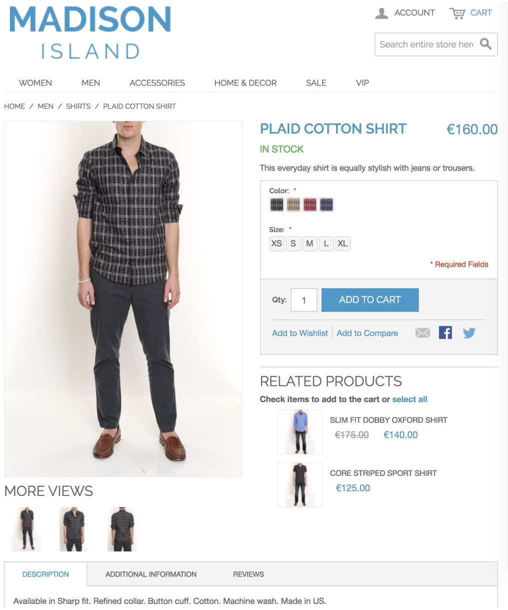


Figure 3.5: Product detail page

The end to end interaction from the homepage to the product detail page can be represented with a model using the following IFML notation:

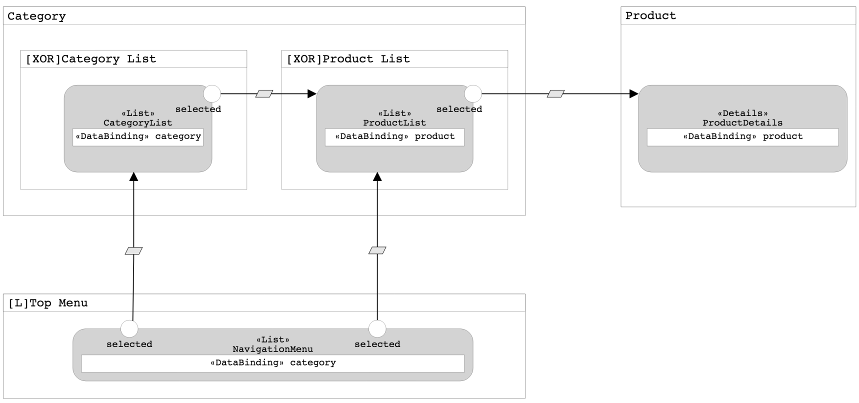


Figure 3.6: IFML representation of the Product Detail interaction

The very same sequence of actions can be expressed as a stream of records in the application server access logs which purpose is to track all the requests processed by the web platform. Here's an example of the very same user journey we just outlined above :

Application Server Access Log			
ID	**Page**	**IFML**	**Log Entry**
1	Home Page	Home	*[29/Nov/2017:06:30:45 +0000] "GET /" 200 0 - 29505*
2	*"View All Men"* Category Page	Category	*[29/Nov/2017:06:49:38 +0000] "GET /men.html /" 200 0 - 29505*
3	*"Shirts"* Category Page	Category	*[29/Nov/2017:07:04:15 +0000] "GET /men/shirts.html" 200 0 - 29505*
4	*"Plaid Cotton Shirt"* Product Page	Product	*[29/Nov/2017:07:08:40 +0000] "GET /men/shirts/plaid-cotton-shirt-476.html" 200 0 - 29505*

As noticeable from this table, both entries 2 and 3 record a *"Category Page"* pageview action logging their associated URLs. The entry 4, on the other side, tracks a *"Product Page"* visit by logging a specific URL into the system concatenating the product URL key and the category path for the product itself to flag its direct access from a Category Page.

3.2.2 Products association and correlation

In this scenario, we analyze the set of actions that can possibly generate page views among different product pages on the Madison Island website.

Using as a starting point the same product page for the *"Plaid Cotton Shirt"* article of the previous navigational behavior analysis,the customer is presented with a *"Related Products"* widget just below the *"Add To Cart"* segment. The principal purpose of this section is to provide the user with some suggestions about products that can be linked in some way to the current one and generate customer's interest.(Figure 3.7)

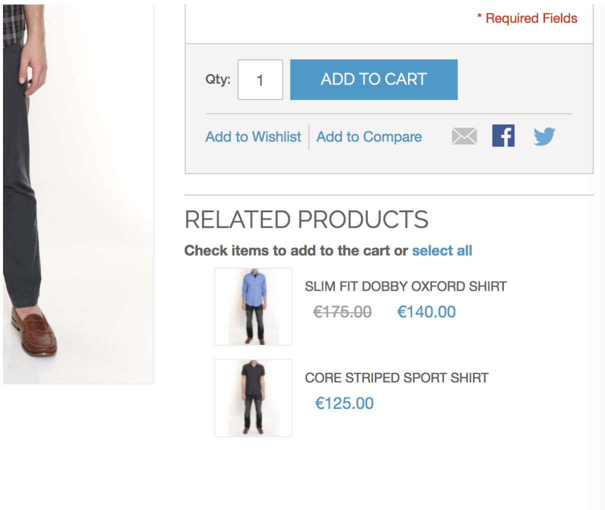


Figure 3.7: Related products section

Clicking either on the product name or its thumbnail the user is brought to the related product page. In this case, we assume his interest goes for the *"Core Striped Sport Shirt"* item presented in the list. (Figure 3.8)

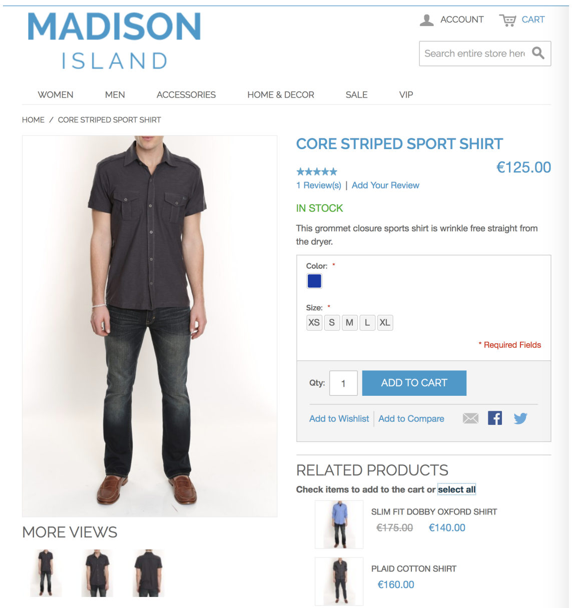


Figure 3.8: A related product page

Besides the simple navigational shortcut offered by the related products section, the user can potentially reach a product page in many other alternative ways on the website, the tracking of these interactions could potentially bring benefits for establishing a correlation pattern among the items. For instance, he can quickly go back with the back button of his browser to the previous category listing page and choose a different item to check, use the navigatoin menu to browse another category or use the search bar on the top bar to perform a global product search. For the sake of this example, we pretend he searches for blazers writing the *"blazer"* word on the previously mentioned search bar from within the current product page. (Figure 3.9)

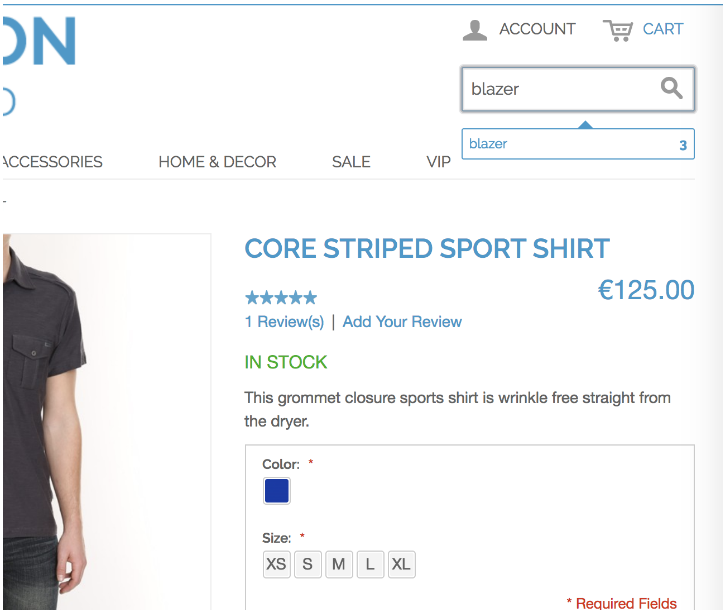


Figure 3.9: Product search bar

When the search is performed, the user is taken to the search result page which shows a list of products matching the word he searched for. From within this screen, the user can freely browse to any of the available product pages similarly to what we already described for the category listing page. (Figure 3.10)

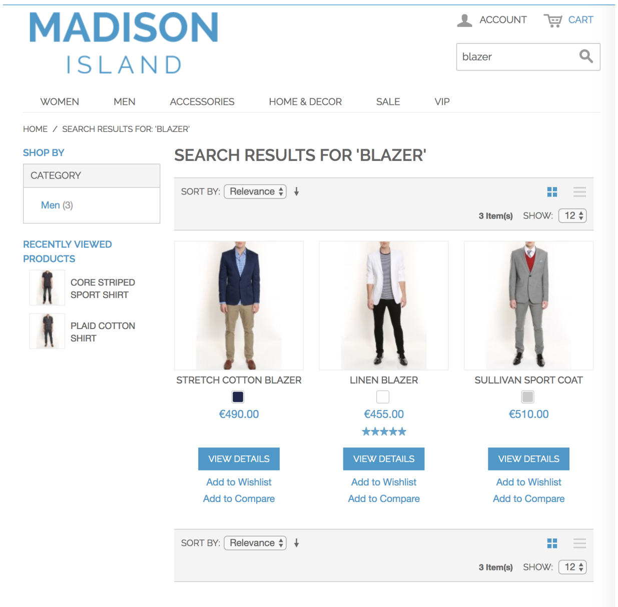


Figure 3.10: Search results page

At this point, we can update and extend the IFML model shown previously in Figure 3.6 accordingly to the new notions and interactions that have been described just above.

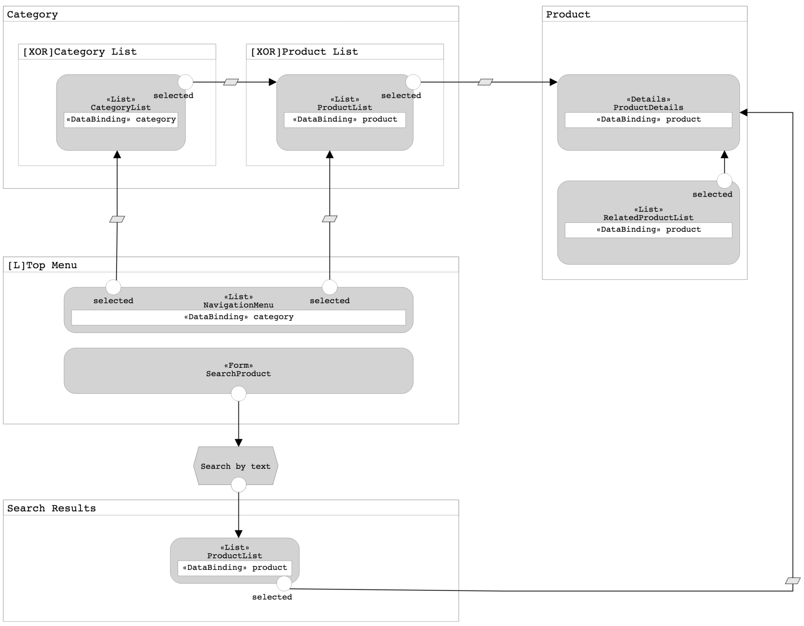


Figure 3.11: Updated IFML representation of the navigational behavious

The same new navigational paths discovered are represented in the application server access log in the following form :

Application Server Access Log			
ID	**Page**	**IFML**	**Log Entry**
1	*"Plaid Cotton Shirt"* Product Page	Product	*[04/Dec/2017:06:37:06 +0000] "GET /men/shirts/plaid-cotton-shirt-476.html" 200 0 - 29505*
2	*"Core Striped Sport Shirt"* Product Page	Product	*[04/Dec/2017:06:37:15 +0000] "GET /core-striped-sport-shirt-551.html" 200 0 - 29505*
3	*"Plaid Cotton Shirt"* Product Page	Product	*[04/Dec/2017:06:37:21 +0000] "GET /men/shirts/plaid-cotton-shirt-476.html" 200 0 - 29505*
5	*"Tees Knits And Polos"* Category Page	Category	*[04/Dec/2017:06:38:06 +0000] "GET /men/tees-knits-and-polos.html" 200 0 - 29505*
6	*"Blazer"* Search By Term	Search Results	*[04/Dec/2017:06:38:20 +0000] "GET /catalogsearch/result/?q=blazer" 200 0 - 29505*
7	*"Stretch Cotton Blazer"* Product Page	Product	*[04/Dec/2017:06:38:43 +0000] "GET /stretch-cotton-blazer-587.html" 200 0 - 29505*

The sequence of actions recorded from the access logs reveal the user browsed from one product to another taking advantage of the related product links available (ID 2); in fact, the target URL does not include any category path beside the URL key associated with the product indicating a direct access. The entries 3 and 4 illustrate the journey of the user clicking on its back button on the browser and performing the very same actions again. The last three actions recorded show a direct access to a category page through the usage of the navigational menu, a search by the *"blazer"* term as per the previous example and the related redirection to the product page respectively.

3.3 IoT behavioral modeling

With mobile surpassing desktop as the most critical influencer for customers to make purchase decisions and track behaviors, location and proximity tracking is proving to be a valuable tool for brands and stores. In these two next subsections, we analyze two possible scenarios of customer interaction in the real world based on IoT devices recording and reporting capabilities. The data coming from IoT devices would then be collected in conjunction with the web-based one described previously to form a combined behavioral data stream to leverage for generating tailored customizations of the Madison eCommerce webportal.

3.3.1 Apple iBeacon technology and Estimote Beacons overview

The IoT device of choice to illustrate the scenarios related to the behavioral modeling would be the Estimote Beacons which are using Apple iBeacon technology and are compatible with iBeacon-enabled Apple products and applications.

The company from Cupertino jumped first on the beacon bandwagon publishing a detailed specification (IDs, transmission intervals, etc.) in 2013 developing the iBeacon protocol and unlocking vendors, such as Estimote, to ship iBeacon-compatible hardware transmitters.

Figure 3.12: An Estimote iBeacon compatible device

As per previously described in 2.1 a beacon can simply be seen as a lighthouse broadcasting information in certain intervals and at a defined power leveraging Low Energy Bluetooth connection. In the case of iBeacons the information sent to listening mobile devices would contain:

- The Universal Unique ID (UUID) is globally unique. Example: de2b45ae-ed98-11e4-3432-78616d6f6f6d
- The Major ID uniquely identifies our customer's system: e.g. 51314
- The Minor ID reports the exact location or object (in our case, the spot): e.g. 23369

To receive this 1-way data stream the customer needs to have an app installed on his phone unlike QR and NFC communication. Technically, the app obeys only to ”its” iBeacons, those with fitting UUID, Major and Minor IDs looking for a match, when this happens the app can react accordingly. This mechanism ensures that only the installed app can track users as they passively walk around the transmitters.

In detail, the phone OS will keep listening for beacons at all times—even if the app is not running or it was closed, and even if the phone is locked or rebooted. Once an “enter” or “exit” happens, the OS will launch the app into the background (if needed) and let it execute some code for a few seconds to handle the event.

3.3.2 Proximity Marketing

Proximity Marketing is an efficient tool to involve and discover new potential customers or target better the already existing ones. This marketing technique operates in a given physical location leveraging technologies offered by the IoT ecosystem to promote the sale of products and services. This communication channel acts on a clear consumer target: all the customers who are in the vicinity or within a given area, covered by the diffusion devices.

This innovative form of relationship marketing aims to activate and involve users by capturing attention at the right time and in the right place, making it experience a more intense and stimulating shopping experience.

In the context of this thesis work, we will focus on a basic scenario where the customer proximity recognition happening in the real world does not directly trigger any immediate action to grab customer attention but it limits itself to silently track the event collecting and sending a communication to a listening web server.

We start defining a possible allocation of items for a ”Madison Island” retail store which resembles the catalog presented on its website.

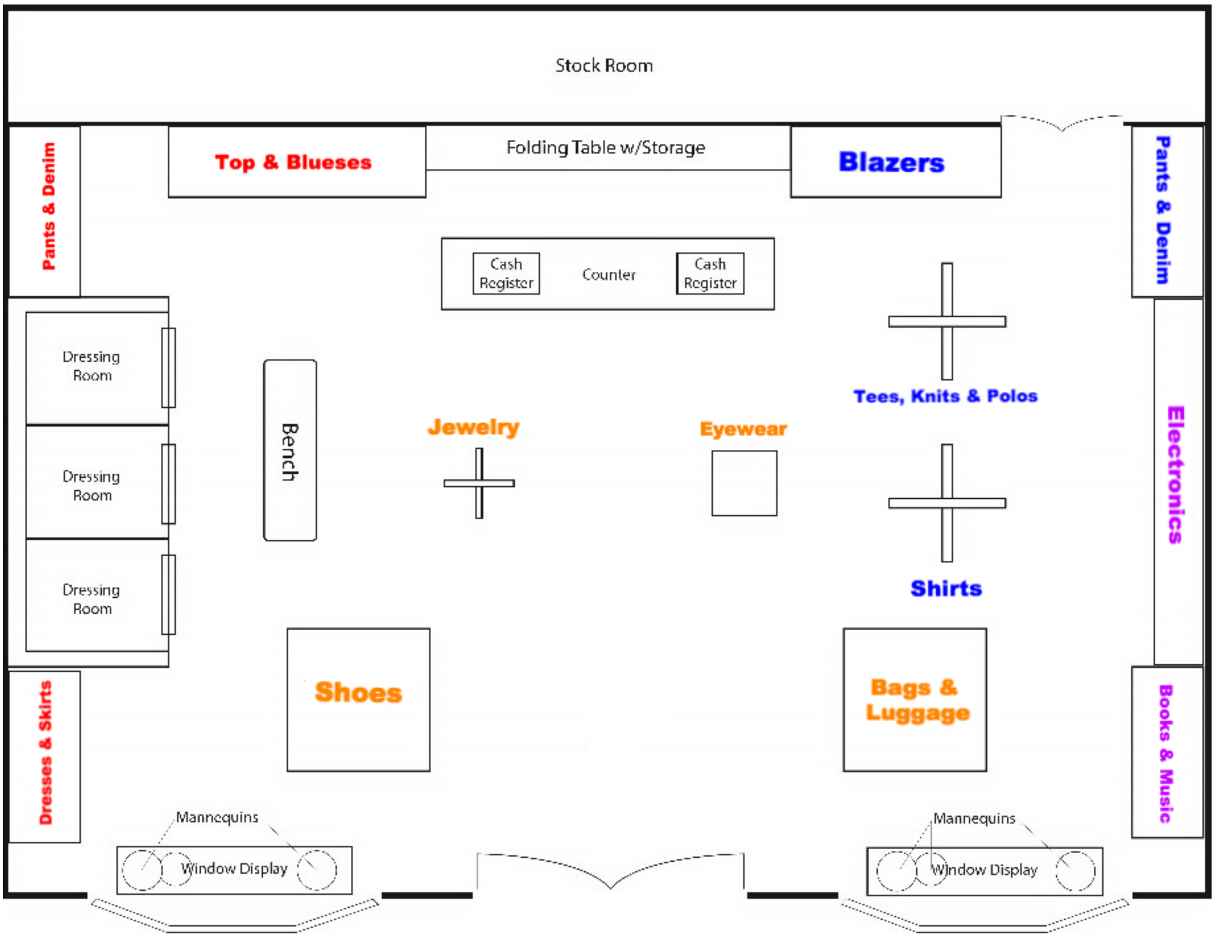


Figure 3.13: Madison Island retail store map

Each label described in 3.13 represents a specific category of the website whereas the color of the label itself indicates the parent category the items belong to. In more detail :

- Red categories belong to the **Women** category mapping the content available at /**women.html**.
- Blue categories belong to the **Men** category mapping the content available at /**men.html**.
- Purple categories belong to the **Home & Decor** category mapping the content available at /**home-decor.html**.
- Orange categories belong to the **Accessories** category mapping the content available at /**accessories.html**.

As the customer walks around the shop, the Madison Island mobile app will scan for a predefined set of beacon regions and register proximity data whenever the device enters and exits from each one of them efficiently tracking which aisles (regions) customers visited and which not.

Considering the above, the following would be a suitable Estimote beacon allocation for the store :

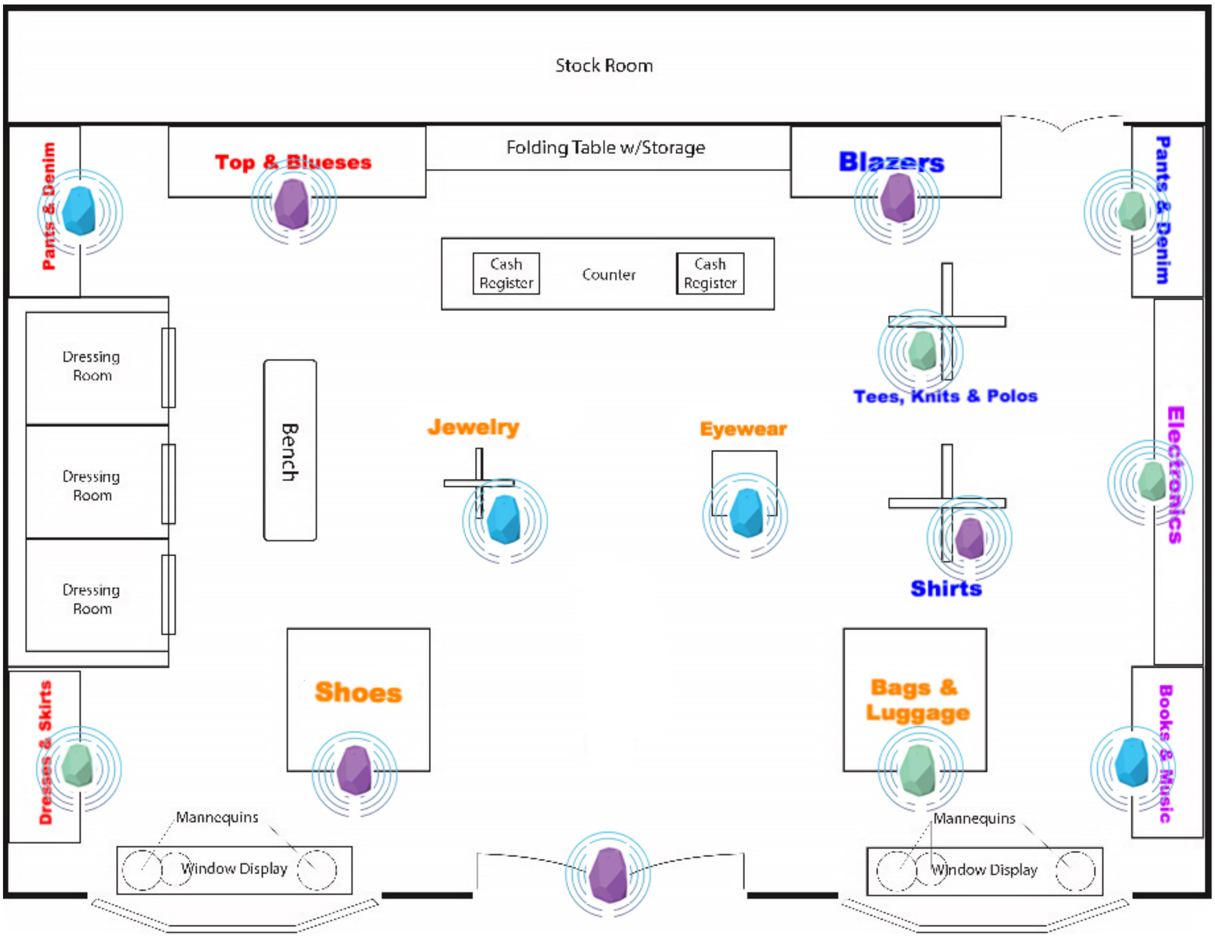


Figure 3.14: Madison Island retail store with Estimote beacons allocation

Depending on the business use case, the mobile app can wether send an event to the listening server whenever the user enters within each region boundary or submit a single event with a full list of regions detected during the staying along with their minimum proximities.

Due to the behavioral profiling nature of this activity, we assume the app behaves like the latter and detects all the events entering or leaving each virtual fence activating a *ranging* procedure aimed to detect proximity information based on the strenght of the Bluetooth signal. [3].

Once the shoppers leaves the retail store the app performs a single data push to a REST API endpoint with all the tracked data.

From a technical perspective, all these operations are accomplished leveraging the Estimote iOS-SDK [4] which allows the developer to define regions and ranges for each beacon quickly and facilitates the ranging process to track the actual proximity to them.

The JSON payload sent from the app to the REST endpoint for customer 3045678 (e.g. **POST /users/3045678/sessions**) for analysis would have this structure:

```
{
```

```
  "data":{
    "customerId":3045678,
    "storeId":8784,
    "storeLabel" :"Madison1",
    "sessionId" :"89376f84-065b-11e8-ba89-0ed5f89f718b"

    "sessionRegions":[
          {
            "regionId: :156,
            "regionLabel":"store-entrance",
            "detectionCount":2,
            "maxSecondsInRegion": 5,
            "maxProximity":"unknown",
            "firstDetectionTimestamp":"2018-02-21T18:09:07Z",
            "lastDetectionTimestamp":"2018-02-21T18:16:02Z",
            "beaconData" :{
              "uuid":"0686a88e-fed6-11e7-8be5-0ed5f89f718b",
              "majorId":2553,
              "minorId":79
            }
          },
          {
            "regionId: :645,
            "regionLabel":"shoes",
            "detectionCount":1,
            "maxSecondsInRegion": 24,
            "maxProximity":"near",
            "firstDetectionTimestamp":"2018-02-21T18:09:20Z",
            "lastDetectionTimestamp":"2018-02-21T18:09:20Z",
            "beaconData" :{
              "uuid":"0686a88e-fed6-11e7-8be5-0ed5f89f718b",
              "majorId":19029,
              "minorId":49
            }
          },
          {
            "regionId" :6875,
            "regionLabel":"jewelry",
```

```
        "detectionCount":1,
        "maxSecondsInRegion": 15,
        "maxProximity":"far",
        "firstDetectionTimestamp":"2018-02-21T18:10:15Z",
        "lastDetectionTimestamp":"2018-02-21T18:10:15Z",
        "beaconData" :{
          "uuid":"0686a88e-fed6-11e7-8be5-0ed5f89f718b",
          "majorId":38415,
          "minorId":59
        }
      },
      {
        "regionId" :2563,
        "regionLabel":"blazers",
        "detectionCount":1,
        "maxSecondsInRegion": 195,
        "maxProximity":"immediate",
        "firstDetectionTimestamp":"2018-02-21T18:11:01Z",
        "lastDetectionTimestamp":"2018-02-21T18:11:01Z",
        "beaconData" :{
          "uuid":"0686a88e-fed6-11e7-8be5-0ed5f89f718b",
          "majorId":25911,
          "minorId":27
        }
      },
      {
        "regionId" :456,
        "regionLabel":"tees-knits-polos",
        "detectionCount":1,
        "maxSecondsInRegion": 10,
        "maxProximity":"far",
        "firstDetectionTimestamp":"2018-02-21T18:14:56Z",
        "lastDetectionTimestamp":"2018-02-21T18:14:56Z",
        "beaconData" :{
          "uuid":"0686a88e-fed6-11e7-8be5-0ed5f89f718b",
          "majorId":42037,
          "minorId":36
        }
```

```
            },
            {
               "regionId" :998,
               "regionLabel":"bags-and-luggage",
               "detectionCount":1,
               "maxSecondsInRegion": 7,
               "maxProximity":"far",
               "firstDetectionTimestamp":"2018-02-21T18:15:12Z",
               "lastDetectionTimestamp":"2018-02-21T18:15:12Z",
               "beaconData" :{
                 "uuid":"0686a88e-fed6-11e7-8be5-0ed5f89f718b",
                 "majorId":37931,
                 "minorId": 85
               }
            }
         ]
   }
}
```


The above example session shows an evident preference in ”Blazer” items by the customer and a slight interest in the ”Shoes” items. More precisely, the ”Blazer” region registered a session lasted more than 3 minutes and the highest proximity to a beacon.

3.3.3 Customer Rewards

Besides Proximity based marketing, Beacon technology can also be used to reward customers for particular actions based on geolocalization data improving the overall quality of the brand loyalty program which is key to cultivating lasting customer relationships.

Achieving such result is possible by expanding the set of actions that enable customers to earn bonuses and discounts on the website (newsletter subscriptions, minimum order amount, etc..) to additional activities performed in the real world, including the simple act of visiting and walking around the store.

For example, the brand can rank customers by the amount of time spent at each Madison Retail shop and reward them with tailored offers every month depening on their purchases, or it can focus on offering special offers on the website to those customers that visited a retail store in a specific time span such as Christmas Time.

For our Madison Island example, we are considering a use case scenario where customers get rewarded if they manage to scan three products QR codes from the store to earn a fixed amount of reward points on their online account. Specifically, when the beacon detects their entrance into the store, the mobile app pushes a notification to the lock screen presenting a CheckPoints message to the customer inviting him to scan codes of actual items available in the shop. Once the valid product scans are collected within a session, the mobile app pushes the information to the same REST API used in the previous chapter which is responsible for storing the data.

Figure 3.15: Madison Island mobile app push notification example

The JSON payload sent to the server (e.g. **POST /users/3045678/scans**) after each succesful scan would then have a structure similar to this :

```
{
  "data":{
     "customerId":3045678,
     "storeId":8784,
     "storeLabel" :"Madison1",
     "sessionId" :"89376f84-065b-11e8-ba89-0ed5f89f718b"
```

```
      "sessionDuration" :456,
      "sessionActions": [{
         "userAgent" :"Iphone 6s",
         "scannedItems":
           [{
             "barcode":"042100005264",
             "name" :"Elizabeth Knit Top-Red-S"
             "sku": "wbk012c-Red-S"
           },
           {
             "barcode":"042100005931",
             "name" :"Plaid Cotton Shirt-Khaki-L"
             "sku": "msj006c-Khaki-L"
           },
           {
             "barcode":"042100007717",
             "name" :"Broad St Saddle Shoes"
             "sku": "shm00110"
           }
           ]
      }]
   }
}
```

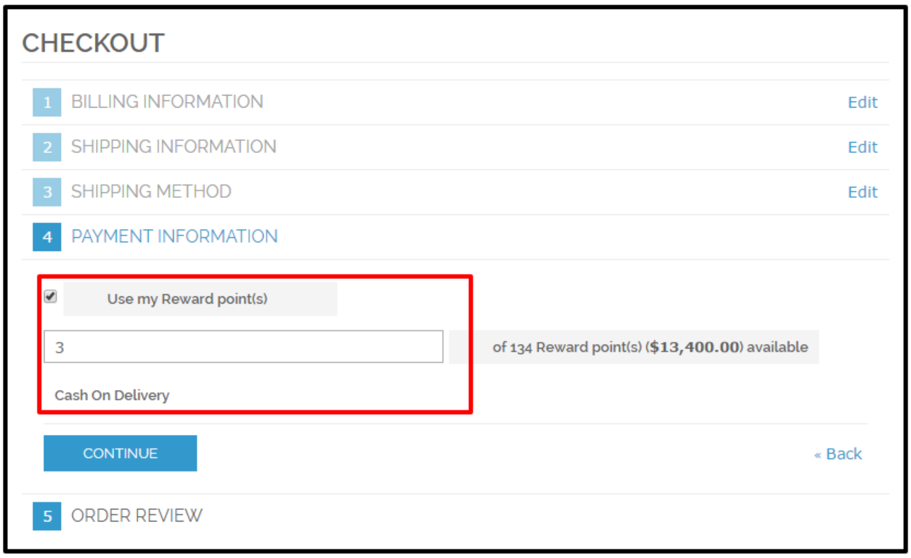


Figure 3.16: Madison Island loyalty program usage during checkout

Similarly to the process outlined in 3.3.2 the data collected by the server about each one of the single product scans performed by the customers during their sessions in the retail store will eventually convert in reward points attributions to use on the website for the customers.

Chapter 4

Our approach

After summarily describing three different streams of real usage data obtained both from the physical and the virtual world in the last chapter, we now focus on expanding those representations in a more detailed way with the help of the Model Driven Engineering techniques briefly described in 1.3. Concretely, the first two section objectives are to illustrate the defining languages (metamodels) for both the real usage data and the eCommerce platform interactions used in the previous examples and generate actual models based upon them representing the very same information. Finally, in the last section of this chapter, we will be using the very same generation for updating the previously instantiated models leveraging model transformation techniques based upon usage pattern detection resulting from the Big Data analysis.

4.1 Representing Real Usage Data

4.1.1 Metamodel

The representation of the real usage data starts from the definition of the metamodel which defines the languages and processes from which to form a model without making statements about its content. In fact, a metamodel is itself a model that is used to describe another model using a modeling language and at a different level of abstraction.

The figure in 4.1 describes the processed metamodel as a UML Class Diagram accordingly to the data retrieved for our real usage data analysis.

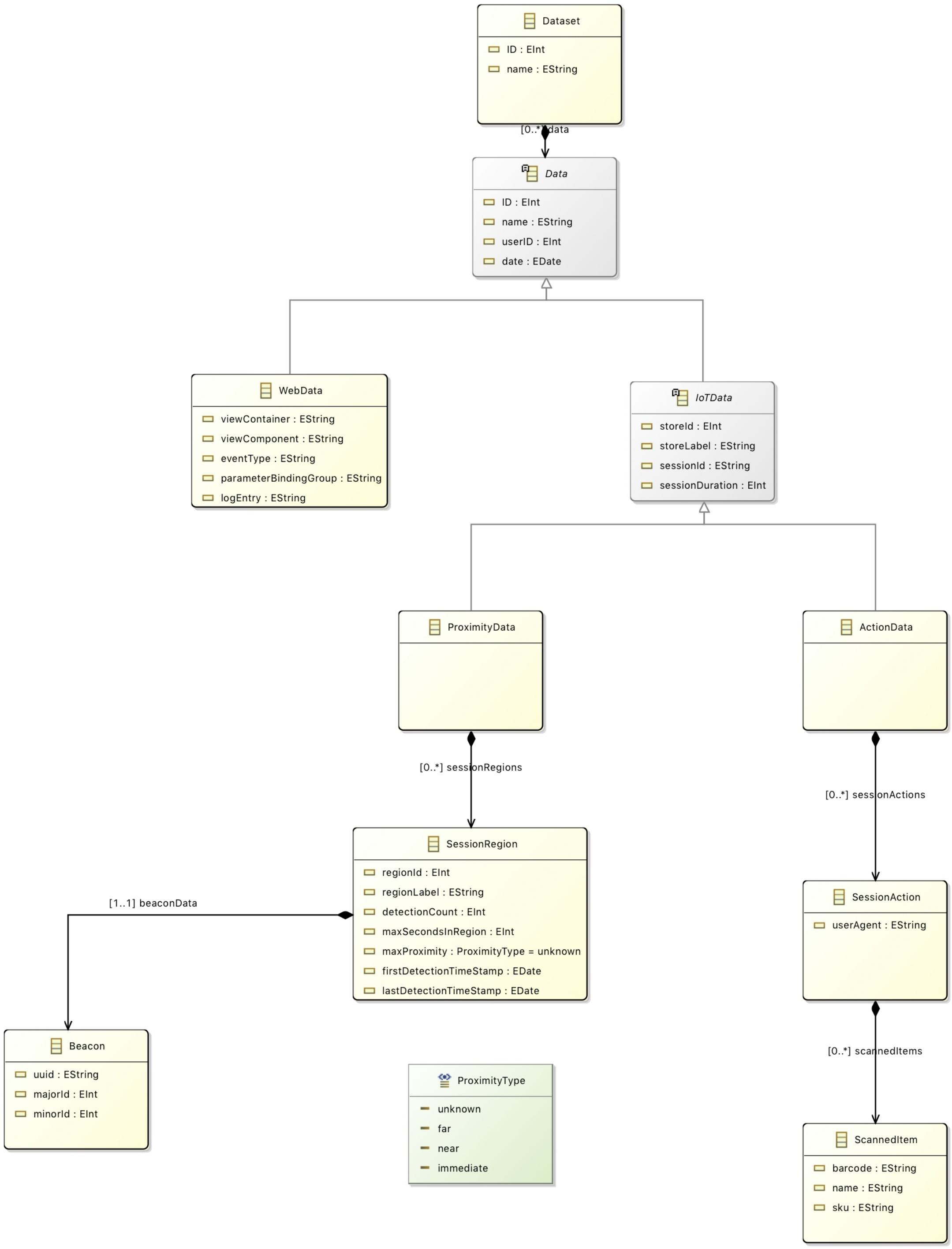


Figure 4.1: Real Usage Data Metamodel Diagram Class

4.1.2 Model

The RealUsageData metamodel defined above allow us to create dynamic instances which precisely map the real usage data collected from the web mining process and the IoT devices tracking. Figure 4.2 illustrates this processed model in its eCore representation form in Eclipse and it is followed by the corrisponding XMI file content.

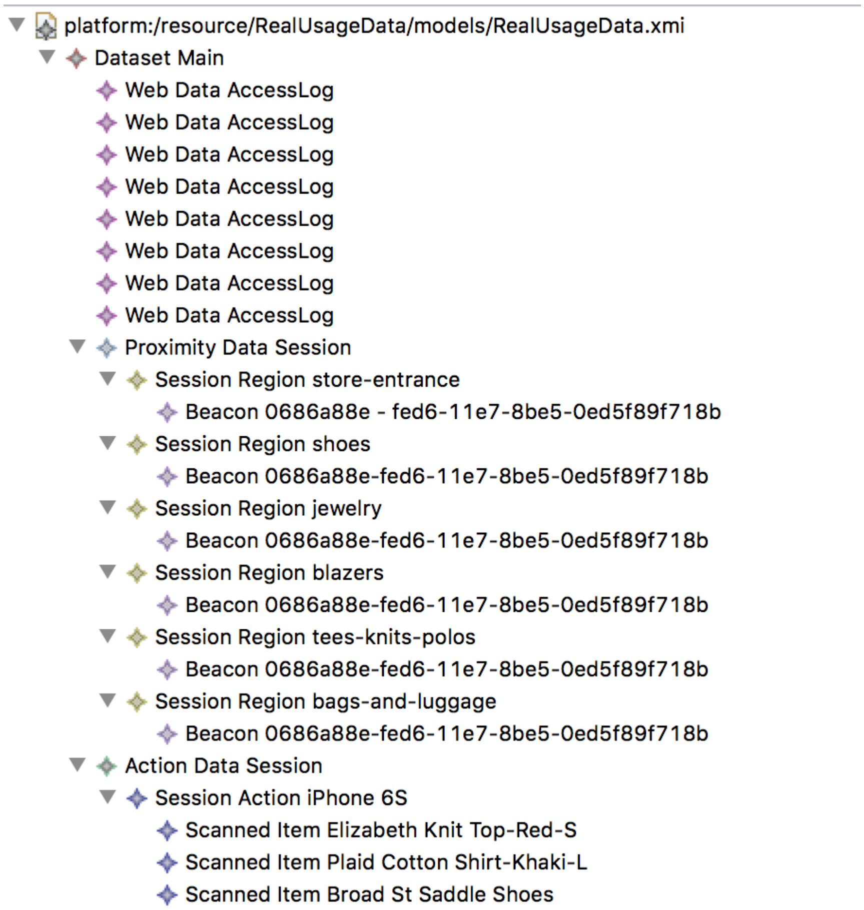


Figure 4.2: Real Usage Data Model

```
<?xml version="1.0" encoding="UTF-8"?>
<RealUsageData:Dataset
    xmi:version="2.0"
    xmlns:xmi="http://www.omg.org/XMI"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:RealUsageData="RealUsageData"
    xsi:schemaLocation="RealUsageData ../metamodels/RealUsageData.ecore"
    ID="1" name="Main">
  <data xsi:type="RealUsageData:WebData"
```

```
    ID="1"
    name="AccessLog"
    userID="3045678"
    date="2017-11-29T17:06:49.000+0100"
    viewContainer="Homepage"
    viewComponent="TopMenu"
    eventType="click"
    parameterBindingGroup="Category/5"
    logEntry="GET /men.html / 200 0 - 29505"/>
<data xsi:type="RealUsageData:WebData"
    ID="2"
    name="AccessLog"
    userID="3045678"
    date="2017-11-29T17:07:04.000+0100"
    viewContainer="Category #5"
    viewComponent="CategoryList"
    eventType="click"
    parameterBindingGroup="Category/15"
    logEntry="GET /men/shirts.html 200 0 - 29505"/>
<data xsi:type="RealUsageData:WebData"
    ID="3"
    name="AccessLog"
    userID="3045678"
    date="2017-11-29T07:08:40.000+0100"
    viewContainer="Category #15"
    viewComponent="ProductList"
    eventType="click"
    parameterBindingGroup="Product/404"
    logEntry="GET /men/shirts/plaid-cotton-shirt-476.html 200 0 - 29505"/>
<data xsi:type="RealUsageData:WebData"
    ID="4"
    name="AccessLog"
    userID="3045678"
    date="2017-12-04T06:37:15.000+0100"
    viewContainer="Product #404"
    viewComponent="RelatedProductList"
    eventType="click"
    parameterBindingGroup="Product/413"
```

```
    logEntry="GET /core-striped-sport-shirt-551.html 200 0 - 29505"/>
<data xsi:type="RealUsageData:WebData"
    ID="5"
    name="AccessLog"
    userID="3045678"
    date="2017-12-04T06:37:21.000+0100"
    viewContainer=""
    viewComponent=""
    eventType="backButton"
    parameterBindingGroup=""
    logEntry="GET /men/shirts/plaid-cotton-shirt-476.html 200 0 - 29505"/>
<data xsi:type="RealUsageData:WebData"
    ID="6"
    name="AccessLog"
    userID="3045678"
    date="2017-12-04T06:38:06.000+0100"
    viewContainer="Product #404"
    viewComponent="TopMenu"
    eventType="click"
    parameterBindingGroup="Category/16"
    logEntry="GET /men/tees-knits-and-polos.html 200 0 - 29505"/>
<data xsi:type="RealUsageData:WebData"
    ID="7"
    name="AccessLog"
    userID="3045678"
    date="2017-12-04T06:38:20.000+0100"
    viewContainer="Category #16"
    viewComponent="TopSearch"
    eventType="submit"
    parameterBindingGroup="SearchText/blazer"
    logEntry="GET /catalogsearch/result/?q=blazer 200 0 - 29505"/>
<data xsi:type="RealUsageData:WebData"
    ID="8"
    name="AccessLog"
    userID="3045678"
    date="2017-12-04T06:38:20.000+0100"
    viewContainer="Search Results"
    viewComponent="ProductList"
```

```
    eventType="click"
    parameterBindingGroup="Product/407"
    logEntry="GET /stretch-cotton-blazer-587.html 200 0 - 29505"/>
<data xsi:type="RealUsageData:ProximityData"
    ID="9"
    name="Session"
    userID="3045678"
    date="2018-02-21T18:16:07.000+0100"
    storeId="8784"
    storeLabel="Madison1"
    sessionId="89376f84-065b -11e8- ba89-0ed5f89f718b"
    sessionDuration="345">
  <sessionRegions
      regionId="156"
      regionLabel="store-entrance"
      detectionCount="2"
      maxSecondsInRegion="5"
      firstDetectionTimeStamp="2018-02-21T18:09:07.000+0100"
      lastDetectionTimeStamp="2018-02-21T18:16:02.000+0100">
    <beaconData
        uuid="0686a88e - fed6-11e7-8be5-0ed5f89f718b"
        majorId="2553"
        minorId="79"/>
  </sessionRegions>
  <sessionRegions
      regionId="645"
      regionLabel="shoes"
      detectionCount="1"
      maxSecondsInRegion="24"
      maxProximity="near"
      firstDetectionTimeStamp="2018-02-21T18:09:20.000+0100"
      lastDetectionTimeStamp="2018-02-21T18:09:20.000+0100">
    <beaconData
        uuid="0686a88e-fed6-11e7-8be5-0ed5f89f718b"
        majorId="19029"
        minorId="49"/>
  </sessionRegions>
  <sessionRegions
```

```
        regionId="6875"
        regionLabel="jewelry"
        detectionCount="1"
        maxSecondsInRegion="15"
        maxProximity="far"
        firstDetectionTimeStamp="2018-02-21T18:10:15.000+0100"
        lastDetectionTimeStamp="2018-02-21T18:10:15.000+0100">
      <beaconData
          uuid="0686a88e-fed6-11e7-8be5-0ed5f89f718b"
          majorId="38415"
          minorId="59"/>
    </sessionRegions>
    <sessionRegions
        regionId="2563"
        regionLabel="blazers"
        detectionCount="1"
        maxSecondsInRegion="195"
        maxProximity="immediate"
        firstDetectionTimeStamp="2018-02-21T18:11:01.000+0100"
        lastDetectionTimeStamp="2018-02-21T18:11:01.000+0100">
      <beaconData
          uuid="0686a88e-fed6-11e7-8be5-0ed5f89f718b"
          majorId="25911"
          minorId="27"/>
    </sessionRegions>
    <sessionRegions
        regionId="456"
        regionLabel="tees-knits-polos"
        detectionCount="1"
        maxSecondsInRegion="10"
        maxProximity="immediate"
        firstDetectionTimeStamp="2018-02-21T18:14:56.000+0100"
        lastDetectionTimeStamp="2018-02-21T18:14:56.000+0100">
      <beaconData
          uuid="0686a88e-fed6-11e7-8be5-0ed5f89f718b"
          majorId="42037"
          minorId="36"/>
    </sessionRegions>
```

```
  <sessionRegions
      regionId="998"
      regionLabel="bags-and-luggage"
      detectionCount="1"
      maxSecondsInRegion="7"
      maxProximity="far"
      firstDetectionTimeStamp="2018-02-21T18:15:12.000+0100"
      lastDetectionTimeStamp="2018-02-21T18:15:12.000+0100">
    <beaconData
        uuid="0686a88e-fed6-11e7-8be5-0ed5f89f718b"
        majorId="37931"
        minorId="85"/>
  </sessionRegions>
</data>
<data xsi:type="RealUsageData:ActionData"
    ID="10"
    name="Session"
    userID="3045678"
    date="2018-02-22T15:27:09.000+0100"
    storeId="8784"
    storeLabel="Madison1"
    sessionId="89376f84-065b-11e8-ba89-0ed5f89f718b"
    sessionDuration="456">
  <sessionActions
      userAgent="iPhone 6S">
    <scannedItems
        barcode="042100005264"
        name="Elizabeth Knit Top-Red-S"
        sku="wbk012c-Red-S"/>
    <scannedItems
        barcode="042100005931"
        name="Plaid Cotton Shirt-Khaki-L"
        sku="msj006c-Khaki-L"/>
    <scannedItems
        barcode="042100007717"
        name="Broad St Saddle Shoes"
        sku="shm00110"/>
  </sessionActions>
```

```
  </data>
</RealUsageData:Dataset>
```

4.2 Representing our eCommerce application with IFML

As per briefly introduced in 3.1, interaction flow models are platform independent-level models which can be used to define the interactions between the users of an application and the application itself. They describe the user interface components required at the front-end of the application, without specifying layout details of these elements enhancing the separation of concerns among developers and UX designers where the latter build the user interface accordingly to an interaction flow model. Besides defining components of the User Interface, the interaction flow model explains how data flows among different sections of the application upon triggering events and introduces the business logic carried out using this data.

4.2.1 IFML Metamodel

The IFML metamodel is organized into three packages: the Core package, the Extension package and the DataTypes package. The Core package contains the concepts that build up the interaction infrastructure of the language concerning InteractionFlowElements, InteractionFlows and Parameters. The Extension package extends the Core package components with more complex behaviors. The DataTypes package contains the custom data types defined by IFML.

By using the primitive data types from the UML metamodel and a UML representation for the IFML Domain Model, the IFML metamodel specifies a set of UML metaclasses as the foundation for the IFML metaclasses.

The following is the structure of the high-level representation of the IFML metamodel and its areas of concern:

- IFML Model
- Interaction Flow Model
- Interaction Flow Elements
- View Elements
- Events
- Specific Events and View Components
- Parameters
- Expressions

- ContentBinding

Figure 4.3 shows an excerpt of the IFML metamodel. As can be seen, IFMLModel is the top-level container of all the model elements and represents an IFML model. It contains an InteractionFlowModel which is the user view of an application, a DomainModel represented in UML and optionally ViewPoints. The concepts extending ViewContainer, ViewComponets, ViewComponentPart, and ViewElementEvent represent the visual elements of an IFML model.

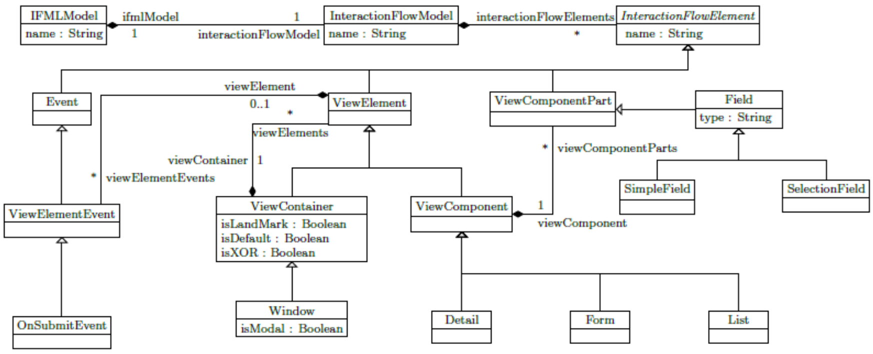


Figure 4.3: Simple Ecore model of an IFML subset.

4.2.2 Model

As per mentioned in the last subsection, interaction flow models are described using the Interaction Flow Modelling Language and, together with the domain model and optionally viewpoints, they form the core of the IFML model.

To complete the picture we may state the domain model objective is essentially offering to the interaction flow references about the content available. An example of a domain model for an e-commerce website is given in figure 4.4

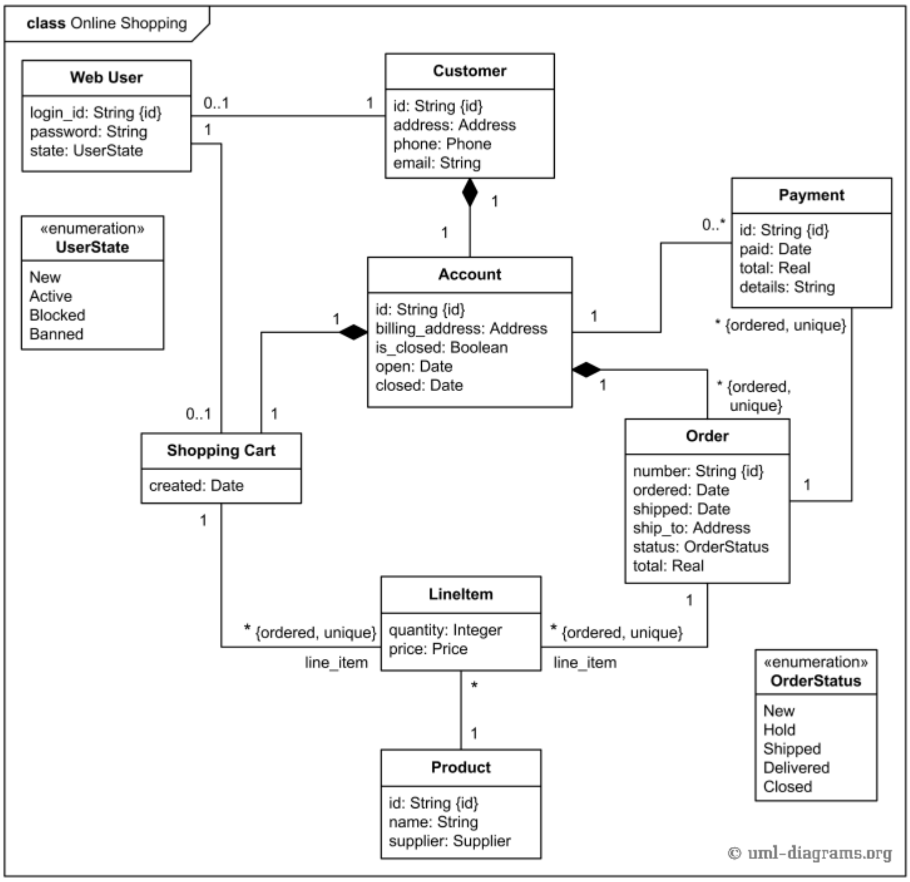


Figure 4.4: Domain Model UML Class Diagram ecommerce example.

Although some partial IFML model representations for the Madison Island eCommerce platform have been already summarily introduced in 3.2, in this subsection we examine them in more detail and with a more global approach not strictly related to the navigational modeling. The final goal is to model, taking advantage of the IFML metamodel described just above, an IFML model which would represent the main pages and interactions of the website on top of which we would perform transformations dictated by the Real Usage Data models illustrated in 4.1.2

4.3 Updating the web models through real usage data

4.3.1 Transformation

4.3.2 Updated Models

Chapter 5

From Models to Code

5.1 Magento eCommerce Platform

5.2 Serializing Models

5.3 Magento extension

Chapter 6

Related work

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborumLorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborumLorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborumLorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborumLorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborumLorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu FINE

Conclusions

Loren ipsum

Bibliography

[1] Marco Brambilla, Eric Umuhoza and Roberto Acerbis - Model-driven development of user interfaces for IoT systems via domain-specific components and patterns

https://jisajournal.springeropen.com/articles/10.1186/s13174-017-0064-1

[2] "IFML Specification document". OMG - Object Management Group. Retrieved 9 April 2013.

[3] Region Monitoring and iBeacon

https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/LocationAwarenessPG/RegionMonitoring/RegionMonitoring.html

[4] GitHub - Estimote/iOS-SDK: Estimote SDK for iOS devices https://github.com/Estimote/iOS-SDK