

**Faculty of Natural and
Mathematical Sciences**
Department of Informatics

The Strand
Strand Campus
London WC2R 2LS
Telephone 020 7848 2145
Fax 020 7848 2851



7CCSMPRJ

Final Project Report

Name: Tudor Zugravu
Student Number: 1671159
Degree Programme: Advanced Software Engineering with Management
Project Title: Domain Specific Sales System Generator with Augmented Reality
Supervisor: Andrew Holyer
Word Count: 0

RELEASE OF PRODUCT

Following the submission of your project, the Department would like to make it publicly available via the library electronic resources. You will retain copyright of the project.

☐ I agree to the release of my project

☐ I do not agree to the release of my project

Signature:

Date: August 20, 2017

Department of Informatics
King's College London
WC2R 2LS London
United Kingdom

Domain Specific Sales System Generator with Augmented Reality

Tudor Zugravu

Student Number: 1671159

Course: Advanced Software Engineering with Management

Supervisor: Andrew Holyer



Thesis submitted as part of the requirements for the award of the MSc in Web
Intelligence.

7CCSMPRJ - MSc Individual Project - 2016

Abstract

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Contents

1	Introduction	1
1.1	Project Aims, Objectives and Introduction	1
1.2	Background and Literature Survey	1
2	Background Theories	2
3	Main Result	4
3.1	System Analysis and Planning	4
3.1.1	The Scoping Process	4
3.1.2	The Planning Process	8
3.2	System Design	10
3.2.1	Business Concept Model	11
4	Conclusion	12
	References	14
A	Review of stochastic calculus	15
A.1	Riemann integration	15
A.2	The Itô integral	15

List of Figures

List of Tables

Acknowledgements

I would like to thank my supervisor.....

1 Introduction

1.1 Project Aims, Objectives and Introduction

It gives a basic background of the work. The problems and project objectives should be clearly stated. The techniques and approaches used to deal with the problem should be stated with reasons, and the contributions and main results achieved should be stated clearly. The structure of the report can be described briefly at the end see subsection 1.2.

1.2 Background and Literature Survey

It gives an overall picture about the work with a clear review of the relevant literature. The background of the project should be given. What have been done to deal with the problem should be stated clearly. The pros and cons of various existing algorithms and approaches should be stated as well. Differences between your proposed method and the existing ones should be briefly described.

The following links may help on the literature review: IEEE Xplore digital library: a resource for accessing IEEE published scientific and technical publications (You must be with King's network to get access to the digital library) ScienceDirect.com: an electronic database offering journal papers not published by IEEE (You must be with King's network to get access to the database)

2 Background Theories

2.1. Too Good To Go One such existing software solution is the "Too Good To Go" mobile application. It targets restaurants and bars that are legally required to throw away excess food at the end of the day and provides a platform through which customers can buy the remaining food at very low prices. This benefits both the customers and the restaurants by increasing sales, offering a cheap food alternative and lowering the great amount of food that is thrown away. Even though "Too Good To Go" targets only unsold food, it has shown great business value, expanding from only a London-targeted application to over 10 European cities. The great success of this application is the main incentive for this project's target. This is but one of numerous niche markets that benefit from the design of such a software solution. The rest of the markets will be more easily available using this project's solution. The "Too Good To Go" consists of a mobile application, both on Android and iOS, which displays the offers either in a list or on a map based on proximity and stock. Filtering of the restaurants based on category has also been added in a recent update to their mobile solutions. The customers have the option to instantly buyout any specific offer and can only redeem it at the restaurant between closing hours. Selection of the favourite eating places is available for the customers. Another functionality of this mobile application is the saving of the credit card details for more seamless future transactions, as well as increased security. In addition, it offers the possibility to rate previous experiences for future reference of other customers. The creators of the system did not add many functionalities, as they focused more on the scope, thus providing a simple and quick mobile solution.

2.2. Appsmakerstore The "Appsmakerstore" system is a web platform that provides the means to create a customizable retail mobile application for both the iOS and the Android mobile operating systems. Even though it does not target multiple vendors based upon a specific type of product or service, its relevance derives from the feature of deploying mobile software solutions only by using a functionalities filtering system and a graphical interface for the clients. This is very similar to this project's solution, the only differences being the

overview design of the generated mobile applications and the list of functionalities that the client can choose from. As the "Appsmakerstore"'s generated applications fall under the retail category, they provide targeted functionalities that can be included in the client's tailored application. The list includes displaying the available products, managing the customer's shopping cart, real-time order and requests processing, displaying promotions, photo galleries, push notifications and loyalty systems. In addition, one of the more interesting features is GEO auto-PUSH messaging that has the scope to attract customers when they are in the proximity. The system also allows the main client to manage and update the mobile applications' content remotely, further enhancing the possibilities of seamless customization. However, this is a feature made possible through the "Appsmakerstore"'s hosting and maintenance package, through which they even publish the clients' solutions to the AppStore and Google Play. However useful this option is, this project's system aims to generate the software solutions and to offer the clients complete control over their product.

3 Main Result

This chapter provides a detailed description of the process through which the proposed system has been generated. Starting from the requirements elicitation process until the implementation and testing phases, the step by step workflow of the project is reported in the following sections.

3.1 System Analysis and Planning

The scope of this project is to develop a platform that provides customisable software products tailored for correlating vendors and customers in the form of both a web platform and a mobile application. To ensure that both the designing and the implementation phases converge to the successful attainment of the project's goals, project management techniques have been applied. As a result, a number of deliverables have been created in order to help identify the specific requirements of both the main and generated systems, as well as to plan and organise the development phase. These deliverables consist of the Conditions of Satisfaction, the Project Overview Statement, the Requirements Breakdown Structure, the Work Package Breakdown and the project's development schedule.

3.1.1 The Scoping Process

The analysis of the project's scope lead to the identification of the Conditions of Satisfaction, which provide the means of establishing when the project has reached the end stage. The COS outlined for this project are:

1. A fully-functional solution is generated in less than 10 minutes.
2. The customers are provided with two different means of using the system.
3. The system customisation options cover most types of vendors.

Having determined the success criteria of the project, the properties of the end system could be established. Derived from the scope of the project, the core functionalities of the system comprise the high-level requirements, which form a necessary and sufficient set for the attainment of the goal [1].

1. A complete system is delivered based upon specific requirements and features.
2. The solution presents a list of offers for the chosen product or service categories based upon distance, rating score or offer value.
3. The chosen offer can be purchased and redeemed at the location.

This provided a sufficient base for generating an initial Requirements Breakdown Structure. The initial step for identifying the basic features of the system was decomposing the high-level requirements into necessary functions and sub-functions. However, due to the complexity and dynamic nature of the end-solution, the set of system requirements has been constantly adapted throughout the development process. The final form of the Requirements Breakdown Structure is:

1. A complete system is delivered based upon specific requirements and features.
 - 1.1. A customisation platform is provided to the client.
 - 1.1.1. The client can choose a system based upon location, product or service.
 - 1.1.2. The system must be suitable for offers of different categories.
 - 1.1.2. The client can customise the appearance of the end-system.
 - 1.2. iOS and Web applications, server and database templates are created.
 - 1.2.1. Database templates matching any combination of customisation criteria are designed.
 - 1.2.2. A server component template is created.
 - 1.2.3. An iOS application template is created.
 - 1.2.4. A Web application template is created.
 - 1.3. The necessary and sufficient set of features and functionalities for the generated

system are identified.

1.3.1. Any type of solution provides rating, directions and offer description functionalities.

1.3.2. Product or service based solutions offer the possibility of offer purchasing.

1.3.3. Service based solutions provide an appointment management system.

2. The solution presents a list of offers for the chosen product or service categories based upon distance, rating score or offer value.

2.1. An updated list of offers is presented, specifying the distance from the customer's current position.

2.1.1. The system can locate the customer's current position.

2.1.2. Customers can view all the offers within a 50 kilometre radius.

2.2. The customer can filter the list of offers.

2.2.1. The list can be sorted by proximity, rating score or offer value.

2.2.2. Customers can manage their favourite locations list.

3. The chosen offer can be viewed, purchased and redeemed at the location.

3.1. The solution provides details about the vendor and the location.

3.1.1. The customers can view the description, rating mark and available offers for any location.

3.1.2. The system provides directions to the location via GPS or through Augmented Reality.

3.1.3. Customers can provide a rating mark after purchasing an offer.

3.2. Customers can purchase offers with credit and debit cards.

3.2.1. Customers can add credit to their accounts.

3.2.2. The vendors are transferred the value of an offer when purchased.

3.2.3. The customers receive receipts for their purchases.

Following this set of requirements ensured the generation of autonomous systems suitable for any type of product or service. The presentation components provide all the important functionalities needed in customers' search for the best suited offer, as well as

a platform for localisation by enabling them to select their favourite vendor locations. In addition, this Requirements Breakdown Structure helped in identifying the customisation criteria necessary for covering most of a potential client's requests. As a result, a Project Overview Statement was written in order to record and maintain a reference to the initial decisions for the project.

The next step towards identifying the best suited approach to developing the system was prioritising the Scope Triangle. The scope of the project is the most critical aspect, as it must target the needs of a large number of both customers and vendors, thus making it prone to revisions. The system's quality is of close importance, as it is essential to acquiring and maintaining a good customer base. Having less than three months for all of the phases of development, the time for creating such an extensive project is short, proving to be the third most critical aspect. Despite that, the priority is developing an extensive and reliable system. The resource availability and the cost could prove to be an issue in the development of a complete solution. However, these aspects are least likely to change as they are clearly outlined from the beginning.

Variable / Priority	Critical (1)	(2)	(3)	(4)	Flexible (5)
Scope	x				
Quality		x			
Time			x		
Cost				x	
Resource availability					x

While the project started with a clear goal, the solution was not yet fully identified at the time. Even though a minimum set of requirements was identified, both the main and the generated systems were not clearly outlined. In addition, further functionalities for such a retail system could and have been identified further on. Based upon these factors, the analysis conducted previously, as well as on the importance of the project

and the amount of time available, the chosen approach towards developing the systems was the Agile Project Management. Thus, the Adaptive Project Management Life Cycle was adopted, outlining the conclusion of the project's scoping phase.

3.1.2 The Planning Process

Having identified the initial requirements for the system, a project plan was outlined in order to organise the following stages of the development process. This provided a clear roadmap for the sequence of steps that needed to be followed as to minimise uncertainty and increase efficiency. However, as the Adaptive lifecycle was chosen as the PMLC, identifying clear tasks and creating an overall schedule was hindered by the large complexity. The approach of solving this issue and maintaining awareness of the progress and results is described in this section.

As the depth of the Requirements Breakdown Structure was not known after the features level, the solution could not be clearly defined yet. Thus, the planning process started with documenting and creating an initial plan using a board and sticky notes. This task provided a good initial approach, as it helped in transforming the features of the RBS into work packages and to estimate the duration and complexity of each and every one of them. This was followed by organising the identified work packages using a software tool in order to maintain persistency and to enhance the visual aspect of the schedule.

Not having an explicit description of the solution, the work packages were arranged into different development phases, in accordance to Agile development guidelines [4]. All phases ended with a feedback loop to assess their completion, helping to determine if the next phase could start or if a new cycle of the current step should be launched to further shape the work packages involved. Following this model aided in managing the complexity and uncertainty of the project, every cycle converging to a complete solution. Five development stages were planned, each starting with a partial perception of

it's expected result and based upon certain assumptions about the input received and completion level of the previous cycles, thus establishing the five major milestones of the project. Aiming both to fulfil already existing tasks and to identify new functionalities for the system resulted in the definition and completion of the end system.

The first phase was the most crucial as it's main purpose was to identify and perform the abstraction of all the business entities involved in the system. In addition, the data structures had to be clearly defined in order to start designing a coherent database structure. This was particularly challenging as the elements this system targeted were subjective and based upon customer preferences, so the requirements analysis performed had to be extensive. Further on, the server component had to be designed and implemented as it was responsible for all the generated system's business logic, acting as a middle point between the database and both the iOS and Web clients. The first milestone was considered reached when the server, the database and a rudimentary iOS mobile application were created and the communication between them was functional, these components representing the deliverables of the initial stage.

The second phase comprised the work packages related to the iOS application template. The launching of this stage implied that the server was active and that the system's functionalities were already implemented. Thus, the iOS client development stage would revolve around presenting the information received from the server and the specific mobile features of the system. However, this was expected to be a time consuming and of a high importance process as this component is the main interaction point with potential customers. In addition, the augmented reality feature increased the level of uncertainty for this cycle. The deliverable of this phase was a fully functional iOS mobile application template, upon the completion of which the second milestone was considered to be reached.

The third phase of the project schedule plans the work packages responsible for the

creation of the Web application and the generation platform. This stage was expected to be less complex than the iOS application development, as a number of features were not included in this client platform, as they had to make use of particular hardware functionalities not available to a browser, such as geometric sensors and video camera usage. However, the main generation system proved to be more complex than expected, requiring an extra cycle for the completion of this phase. The milestone was reached when all of the established features were implemented, the deliverable being a functional client Web interface template and a generation system.

The fourth and fifth phases consisted of deploying, debugging and testing the overall system, thus being dependent on the deliverables from the previous cycles. Extra time was allocated for each work package involved, as neither of the components were yet extensively tested. Thus, a time margin was granted for fixing any system or logic issue that became visible. Reaching the milestone of the fifth stage implied that the project's solution was identified and created, this being asserted by evaluating the system based upon the success criteria of the Conditions of Satisfaction.

3.2 System Design

As an Agile project, the functionalities, features and even requirements may change throughout the development process. In order to increase the efficiency and reliability of the software solution, Model Driven Development has been chosen as the main approach for designing and implementing both the main and generated systems. Through model creation and transformation, the complexity of the project will be reduced. As the initial step towards developing the solution is the abstraction of the components and interactions, this will aid in maintaining the core functionalities throughout the usage of multiple types of programming languages, as well as in the case of changing an already used technology. In addition, following MDD techniques provides clear steps in case of changing or reusing the components involved [2]. This section aims to present the steps taken for outlining the resulted overall system design.

The first step towards identifying and defining the main functionalities and components of the system is to create a Platform Independent Model. The system model regarding the core domain notions and software independent structures helps to convey the high-level functionalities of the system, as well as provide flexibility and reusability across both the iOS and the Web platforms. This has been achieved by outlining the Business Concept Model based upon the business requirements identified in the previous section, in addition to the system requirements resulted from establishing the Use Cases for the two systems described by this project [3].

3.2.1 Business Concept Model

The representations of the domain elements and the associations between them form the Business Concept Model, an initial design of the entities used and handled in real life problems that a system tackles. The Unified Model Language was chosen to present the elements of all the design and architecture components of this project. Thus, the resulted model for the proposed system are presented in fig.

4 Conclusion

It is a chapter to sum up the main points of the work, such as the aims and objectives of the project, the contributions and results you have achieved. Future plan and development can be mentioned in this section.

References

- [1] Andrew Holyer. How to scope a tpm project, 2016.
- [2] Kevin Lano. *Model-driven software development with UML and Java*. Cengage Learning EMEA, 1 edition, 2009.
- [3] Dr. Nishanth Sastry. Model driven development, 2017.
- [4] Robert K Wysocki. *Effective Project Management: Traditional; Agile; Extreme;*. Wiley, 7 edition, 2013.

A Review of stochastic calculus

A.1 Riemann integration

A.2 The Itô integral