

**İSTANBUL TEKNİK ÜNİVERSİTESİ  
ELEKTRİK-ELEKTRONİK FAKÜLTESİ**

**“SMART”  
AKILLI MARKET SİSTEMİ**

**Bitirme Ödevi**

**Ali Ulvi Kasapođlu 040020384  
Aslı Filiz 040020364**

**Bölüm : Bilgisayar Mühendisliđi  
Anabilim Dalı: Bilgisayar Bilimleri**

**Danışman : Yrd. Doç. Dr. Feza BUZLUCA**

Haziran 2006

**İSTANBUL TEKNİK ÜNİVERSİTESİ  
ELEKTRİK-ELEKTRONİK FAKÜLTESİ**

**“SMART” AKILLI MARKET SİSTEMİ**

**Bitirme Ödevi**

**Ali Ulvi Kasapođlu 040020384  
Aslı Filiz 040020364**

**Bölüm : Bilgisayar Mühendisliđi  
Anabilim Dalı : Bilgisayar Bilimleri**

**Danışman : Yrd. Doç. Dr. Feza BUZLUCA**

Haziran 2006

## **Özgünlük Bildirisi**

1. Bu çalışmada, başka kaynaklardan yapılan tüm alıntıların, ilgili kaynaklar referans gösterilerek açıkça belirtildiğini,
2. Alıntılar dışındaki bölümlerin, özellikle projenin ana konusunu oluşturan teorik çalışmaların ve yazılım/donanımın benim tarafımdan yapıldığını bildiririm.

İstanbul, 22 Mayıs 2006

Ali Ulvi Kasapoğlu

Aslı FİLİZ

# “SMARKET” AKILLI MARKET SİSTEMİ

## ( ÖZET )

Avon, Oriflame, Tupperware gibi birçok firma doğrudan satış yöntemini tercih etmektedir. Bu yöntemde, firma tarafından sorumluluk verilen satış görevlisi ile müşteri birebir temas kurar ve satış görevlisi elinde bulunan basılı haldeki katalogdan müşterisine satışa sunulmuş ürünleri gösterir. Müşteri bir veya birden fazla ürünü almaya karar verdiğinde satış görevlisi bu siparişi alır ve telefon veya faks aracılığı ile firmaya iletir. Daha sonra sipariş, firma tarafından müşterinin adresine teslim edilir.

Bu birbirine bağlı prosedürler sırasında insan faktörüne bağlı olarak birçok hata ile karşılaşmaktadır. Örneğin, telefon veya faks ile iletilen bilgilerin kaybolması, yanlış gönderilmesi veya firma tarafından yanlış alınması mümkün olabilmektedir.

Projenin amaçlarından biri, yukarıda bahsedilen süreçleri elektronik ortama taşıyarak sözü edilen bu hataları ortadan kaldırmak, en azından minimum seviyeye düşürmek için bahsedilen satış süreçlerinin otomasyonunu sağlamak ve bu noktada harcanan işgücünü mümkün olduğunca azaltmaktır. Ayrıca böylece tüm süreç kaydı tutulabilir hale gelmektedir. Bu noktada, satış görevlisinde bulunan ürün kataloğuna, ürüne ait tanıtıcı resimler ve bilgilerle fiyat bilgisinin yanına ürünün kod numarası (ID) da eklenir. Satış görevlisi siparişe ilişkin bu kod numarasını ve ilgili adedi cep telefonunda belli bir arayüz aracılığı ile firmaya iletir. Bu bilgiler web arayüzü tarafından okunarak veri tabanının ilgili kısımlarına işlenir, daha sonra bu veriler firma görevlileri tarafından okunarak müşterinin adresine yollanır.

Projenin bir diğer önemli amacı da ürünleri ve müşterileri, siparişler üzerinden belli sınıflara ayırarak bazı promosyonları firma yönetiminin onayına sunmaktır. Bu sınıflandırma ve bunun aracılığı ile belirlenecek promosyonlar, veri madenciliği yapılarak oluşturulacaktır. Burada kullanılan madencilik algoritması, Jiawei Han, Jian Pei ve Yiwen Yin tarafından önerilen ilişkilendirme algoritması[4] üzerinde bu probleme özel bazı değişiklikler ve sadeleştirmeler yapılarak elde edilmiştir. Projede, zaman kısıtından dolayı sadece birlikte tüketilen ürünlerin ilişkileri göz önünde tutulmuş, birlikte sipariş edildiği belirlenen ürünlerden birinin sipariş edilmesi durumunda diğer(ler)i için promosyon önerilmiştir. Madencilik algoritması, iki kısımdan oluşmaktadır: İlk kısımda ilişkilerin bulunması için gerekli olan FP-ağacı veri tabanı taranarak oluşturulur, ikinci kısımda ise bu ağaç kullanılarak yaygın öğeler elde edilir. Bu sonuç alındığında, siparişlerden birinde yaygın öğeler kümesinden bir elemanın yer alması durumunda diğer(ler)i için promosyon önerisi hazırlanır.

Projeyi test etmek amacıyla belli sonuçlara yönelik veriler hazırlanarak algoritmanın adımları tek tek test edilmiştir. Her adımın doğru çalıştığı kesinleştiğinde, büyük miktarda rasgele veriler hazırlanarak sisteme girilmiş ve veri madenciliği programı çalıştırılmıştır. Elde edilen yaygın öğeler ve programın ara sonuçları incelendiğinde elde edilen çıkış değerlerinin teorik beklentilerle uyum içinde olduğu görülmüştür.

Projede kullanılan madencilik algoritması, yukarıda da belirtildiği gibi, uygun değişiklikler yapılarak farklı ilişkiler için de çalışır hale getirilebilir. Bu durumda ticari bakımdan kullanılabilirliği daha yüksek olacaktır.

# **“SMARKET” SMART MARKETING SYSTEM**

## **( SUMMARY )**

A lot of companies like Avon, Oriflame and Tupperware prefer the method of outright sale. In this method, there are neither warehouses, nor stores, nor customers walking around, but sales representatives assigned by the company who establish a one-to-one connection with the customer in their houses or wherever they are. Sales representatives carry the product catalogues with them which consist of information about the products like their photo, some introducing details and their price. They expose this to the customers and if the customer makes an order, the according sales representative notifies the company of this order via phone or fax. Later on, the order of the customer will be delivered to his/her address.

All these procedures depend on each other and they involve a high rate of human work so there appear many errors. For instance, the information transmitted over phone or fax can be lost, mixed, received incorrectly by the company or sent incorrectly by the sales representative.

One of the aims of the project is to transfer the procedures explained above to the electronic environment. This automation of the sale procedure enables the total removal of the errors and mistakes named in the previous paragraph in some certain situations or at least the reduction of these to an acceptable minimum. Besides, this way the whole procedure becomes recordable and tracable.

To achieve that, an identification number (ID) is added to the other details of the products of the named catalogue of products. The sales representative transmits this identification number and the according count of the product requested to the company using a GSM interface created specifically for this application. These information is received by the web interface and entered into the relevant fields of the company database. Afterwards, these data is conveyed to the workers charged to post the orders.

Another crucial aim of the project is classifying the products and the customers using their orders. These classification is used to determine new promotions for the company and table hem to the resolution of the company management.

The classes and the promotions determined over them are computed with data mining. The data mining algorithm involved here is based on the association algorithm developed by Jiawei Han, Jian Pei ve Yiwen Yin [4], but there have been some changes and simplifications made specifically for the structure of this certain problem. Because of the time constraints, only the associations between the products consumed together are taken into account. So, if one of the products between which an association is found is ordered, a promotion for the other product or products is suggested.

The data mining algorithm consists of two steps:

The first step scans the database and constructs the frequent pattern tree, known as FP-tree. This is a tree composed of three parts. The first constituent is the *root* of the tree which has the information “null” and its children are the *item prefix subtrees*, which make up the second constituent. Nodes in these subtrees include the *item-name* which is the name of the product in our project, the *count* which is the number of orders containing the relevant product, and the *node-link* which points to the following node that has the same item-name or is null if there is no such node. Third constituent of the FP-tree is the *frequent-item header table* with the fields the *item-name* which is the product name as explained previously, and the *head of node-link* which points to the first node with the same item-name.

The main method that constructs the FP-Tree is “build” of class “FP\_Builder”. This method first scans the database once and reads all the orders in the database. All of these will be examined and the *frequent items* will be specified. These are products that are included in orders more than a certain times, which is called the *threshold*. The value of the threshold is to be set manually by the user, the smaller it is, the more frequent patterns will be obtained, and the greater it is, the less frequent patterns will be obtained. After specifying the frequent items, the orders will be re-examined and translated to a list consisting of only frequent items, each ordered by descending support value, or by descending *frequency*, which is the number of the appearance of that product in all orders. Next step is creating the header-table which is explained above in detail. Finally, all the orders, called *transactions* in the algorithm and code, will be inserted to FP-Tree as a branch from root to a leaf. If that branch already exists, the support values of the products on that branch will be incremented, else a new branch is created.

The second step is *pattern fragment growth*, based on the procedure FP-growth which uses the FP-tree calculated in the previous step. The procedure FP-growth takes the frequent items and using those, finds the frequent item sets in the database. The output of the algorithm is the list of frequent patterns.

In the referenced algorithm, this procedure operates on the FP-Tree from its beginning to the end. But it is implemented slightly differently to make the computation easier, since working on a tree is easy for the human eye, but arrays and similar structures sometimes give better results when the subject is a computer. So, the tree is scanned and all the paths are stored as arrays. This is done by handling each node of the tree, following the links to its children since a leaf (a node without any child) is reached. Then, paths containing a selected frequent item are found and using these paths, the *conditional frequent pattern base* for each frequent item is calculated. This is a list including the IDs of the frequent items which are ordered together with the specified frequent item more than the threshold value. So, the list of all possible combinations of these items give us the the list of frequent patterns, the result of mining.

Once these are in hand, the orders are checked if they include a product in the list of frequent items. If such a situation is encountered, a promotion for the other product(s) in the same frequent pattern with the said product is suggested to company management.

With appropriate changes and additions, the algorithm can be converted to an algorithm finding other possible associations in the database. This would increase the usability of the project for companies in real world.

The project is made of four interfaces: The database, web, GSM and data mining interfaces.

The database interface is written using EasyPHP 1-8. The database consists of six tables which are explained below in detail. The text file *database.txt* includes all the necessary commands to create the database, the tables, and enter some data to be entered to some of the tables.

The table “customers” contains information about all the customers registered to the system, which means any customer who bought anything from this company at any date of time. The saved data items are an identification number (ID) assigned to the customer by the system, customer’s name, password, address, phone, e-mail address, the area he or she is found in, the identification number of the sales representative who is responsible for this customer and boolean value which shows if the customer accepts to receive SMS’ for promotions or not. The content of this table has no importance for the data mining algorithm, so there are only a few symbolic values entered to it. This is done during the creation of the database.

The next table is “orders” which saves data of any order made any time via the system. An order is specified by an identification number, the identification number of the customer who made the order, the date at which the order was made, if it has been sent or not, and finally “products” which is a string containing the identification numbers and the number of the product ordered in a specific format. This table is crucial for data mining. The more data it contains, the better results can be produced by the mining, so it is created using a program. This program creates a random number of orders which contain random products with random count.

“products” is the next large table. It contains the identification number of a product, its name, type, cost, price, the number of items sold within the last month and the number of items in the reserve at the moment. This is also a very important table for data mining, since the orders are created dependent on this table. Because of that, the products in this table is numerous. These are sample data taken from a country-wide active market chain and entered to table automatically after its creation.

The table “sales\_reps” contains information about the sales representatives. Additional to the identity informations like identification number, name, password, address, area, phone and e-mail address, also other information like the representative’s average sale amount in a month, sale amount within the last month and number of customers the representative is responsible of. Since the project is limited with mining on orders only, this table has no effect on the data mining algorithm, neither, so it contains data for only a small number of sales representatives.

The table “types” consists of only two fields, the identification number and name of a type. This table shows different classifications of products and is thought as a help for the association algorithm. The algorithm implemented in the project is concentrated on finding frequent patterns only, which means it examines the orders and finds out the products which are usually ordered together, so the table “types” has no role in this algorithm. But in case of extending the algorithm in such a way that it also consideres some certain grouping of the products, this table is assumed to be very helpful. The data in this table depends on the data in the table “products”. Products are grouped into six types, so there are six entries in the table “types”.

The most important indicator of data mining results is the table “promotions”. Using the association algorithm, some promotions are determined and saved and monitored with this table. In the table there are the identification number and name of the promotion, start and end dates of the interval in which the promotion is valid, the area and the customers’ identification numbers for which it is set, number of discounted items, discount percentage, the profit made in the month before the promotion was valid and the profit in the month of profit. The last two serve as an indicator that monitors if the promotion has been successful or not. To bind the results of the mining algorithm and the web interface, a special program is written. When activated, this program takes the orders made after a specific date, runs the data mining algorithm on these orders which gives the list of frequent patterns as output. Promotions are set for a product(s) in a pattern at least another element of which is included in an order. These promotions are monitored to the company management and an approval or denial is waited for each promotion. If a promotion is approved, it is entered to the table “promotions” and the user is informed of that situation.

The web interface is designed using MacroMedia DreamWeaver and coded with PHP programming language. There are two main modules: With the module for administrator, the company management can monitor the products in the reserve, register a new sales representative, view the situation of current promotions and suggested promotions, approve or deny them. The sales representative module enables to the representative add new customers to the system, send an order to the company, and view his or her own personal details.

Deriving from the fundamental concept “customer may be anywhere at anytime”, Direct Sales, which is adopted as a sales model in this project, necessitates ubiquitous access to all tools of sales. This necessity may emerge when the sales representative comes across with a potential customer who tends to have great interest in the products offered by the company or in any other unexpected way.

The mobile application component of the project is designed and implemented to serve this purpose. Initially, setting out for a user-friendly and functional mobile web service, the keywords are “simplicity”, “functionality” and “low-maintenance applications”. Since, mobile devices, which usually employ very limited features on processor, memory and power supply, implies easy-to-handle software applications; the keywords “low-maintenance” and “simplicity” become even more crucial.

The mobile application consists of two layers, client (mobile phone) side and server (Servlet applications) side. The client side, which is implemented in Java Micro Edition with MIDP (Mobile Information Device Profile) 1.0 and CLDC (Connected Limited Device Configuration) 1.0, is designed to handle duties in the simplest way in order to optimize the speed and the performance. Thus, complex jobs, such as parsing parameter contents, connecting to databases, is redirected to the corresponding Servlet application working in the server part of the application. The MIDlet application (the small mobile ‘applet’ running on the mobile device) simply prompts request for user input such as identification data, order strings, confirmation messages, etc. This kind of approach when adjusting the division of labour brings with efficient usage of constrained resources of the mobile device.

The server side of the application, actually, constitutes the backbone of the mobile web service, since it handles most of the essential duties in the process. The Servlets

running on a Tomcat server, basically, receives the user data, which is transmitted from the MIDlet application on the mobile phone via radio packages over http; and verifies its consistency-syntax before it performs the appropriate action. Since MIDlets cannot directly establish a connection to databases, the communication between the mobile phone owner (the sales representative, in this case) is provided over these servlets, which have direct access to the company database over JDBC (Java Database Connector).

Regarding the server-side of the application, there are fundamentally, two actions that are handled by the servlets: ID verification and database querying. ID verification stands for the 'stricted access'. This feature is essential since an order request is only expected and permitted to be entered by an authorized sales representative. Serving this purpose, verification servlets receives the identification data from the GPRS-connected mobile device and looks it up in the corresponding database table. The user access to the rest of the web service is only given to clients who provide a valid password-ID number pair. Additionally, the servlets receive and parse the 'raw' order data arriving from the MIDlet application. The order strings are initially considered raw and potentially faulty, since there may be bad intentioned intrusion or abuse attempts to the system. Thus, the user input is first checked for consistency and right syntax before using it in database queries. For instance, the order string is checked whether it contains well-formatted order items. In case an invalid input is caught by the parser at servlet, that input is discarded, preventing it from causing exceptions in the system.

The resulting queries or messages are posted from the Servlet to the mobile device application in order to establish the sales-representative \ database interaction. This sent data is perfectly formatted in order to necessitate the minimum amount of word-processing, since exhausting the limited resources of the mobile device is highly prevented.

With the cooperating client-server structure, the mobile web service fort he sales representative, is designed to provide a simple yet powerful for ubiquitous purchase & order actions. However, it may still be in security aspects of the operation performed.

The data mining interface is based on the association algorithm explained above in detail and another program that enters sample data to the system for the mining algorithm. The algorithm is coded using Java programming language and written as servlets. The algorithm that enteres random data to the database is also an important compound of this interface. When the data necessary is entered to the database, the data mining algorithm is run, the promotions are determined and if approved, added to the database and the user is informed of them.

The division of labour is as follows:

- Web and Database Design: Ali Ulvi Kasapoğlu, Aslı Filiz
- Database Implementation: Aslı Filiz
- Web Design: Ali Ulvi Kasapoğlu
- Web Implementation (PHP coding): Ali Ulvi Kasapoğlu, Aslı Filiz
- Web – Database Integration Testing: Ali Ulvi Kasapoğlu, Aslı Filiz
- GSM Interface Design and Implementation: Ali Ulvi Kasapoğlu
- Web – Database – GSM Integration Testing : Ali Ulvi Kasapoğlu
- Data Mining Research: Aslı Filiz
- Association Algorithm Design and Implementation: Aslı Filiz
- Association Algorithm Testing: Ali Ulvi Kasapoğlu, Aslı Filiz

- Random Data Creation Algorithm: Ali Ulvi Kasapođlu
- Promotion Determining Algorithm: Ali Ulvi Kasapođlu
- Data Mining Algorithms Integration Testing: Ali Ulvi Kasapođlu, Aslı Filiz
- Web – Database – GSM – Mining Integration Testing : Ali Ulvi Kasapođlu, Aslı Filiz

# İÇİNDEKİLER

Şekiller.....	12
1. Giriş.....	13
2. Projenin Tanımı ve Planı.....	14
2. 1. Proje Tanımı .....	14
2. 2. İş Planı .....	14
2. 2. 1. Web ve Veritabanı Arayüzlerinin Gerçeklenmesi.....	14
2. 2. 1. 1. Web Arayüzü ve Veritabanı Tasarımı.....	14
2. 2. 1. 2. Veritabanı Kurulumu.....	14
2. 2. 1. 3. Web Arayüzü Görsel Tasarımı.....	14
2. 2. 1. 4. Web Arayüzü Kodlaması.....	14
2. 2. 1. 5. Web ve Veritabanı Arayüzlerinin Test Edilmesi.....	14
2. 2. 2. Mobil Arayüzün Gerçeklenmesi.....	15
2. 2. 2. 1. SMS Kullanma.....	15
2. 2. 2. 2. Web, Veritabanı ve Mobil Arayüzlerin Test Edilmesi....	15
2. 2. 3. Veri Madenciliği ve Promosyon Algoritması.....	15
2. 2. 3. 1. Ön Araştırma.....	15
2. 2. 3. 2. Algoritma Tasarımı.....	15
2. 2. 3. 3. Uygulama.....	16
3. Kuramsal Bilgiler.....	16
3. 1. Mobil Telefon Uygulaması: Java2ME, MIDP ve CLDC Kavramları.....	16
3. 1. 1. Açıklama.....	16
3. 1. 2. Bileşenler.....	16
3. 1. 2. 1. Konfigürasyon.....	16
3. 1. 2. 2. Profil.....	17
3. 1. 2. 3. Seçime Bağlı Özellikler.....	17
3. 1. 3. Uygulama.....	17
3. 2. Veri Madenciliği Uygulaması: Veri Madenciliği ve İlişkilendirme.....	18
3. 2. 1. Veri Madenciliği.....	18
3. 2. 2. İlişkilendirme Algoritmaları.....	19
3. 2. 2. 1. Kavramlar.....	19
3. 2. 2. 2. İlişkilendirme Algoritmaları Hakkında Genel Bilgi.....	20
3. 2. 2. 3. Yaygın Örüntü Ağacı (FP-ağacı) Yaklaşımı.....	20
4. Analiz ve Modelleme.....	22
4. 1. Doğrudan Satış Yöntemi ve Promosyon Planı.....	22
4. 1. 1. İş Akışı.....	22
4. 2. Sipariş ve Kayıt Otomasyon Sistemi.....	23
4. 2. 1. Genel Nitelikler.....	23
4. 2. 2. Sistemde Kimlikler.....	23
4. 2. 3. Promosyon Oluşturma ve Yönetimi.....	24
4. 2. 4. Proje Bileşenleri Arasında İlişkiler.....	25
4. 3. Modelleme.....	26
4. 3. 1. Sipariş İş Akışı.....	26
4. 3. 2. Sipariş / İstatistik Otomasyon Sistemi Modeli.....	26
4. 3. 2. 1. Sipariş Görevlisi Arayüzü.....	27

4. 3. 2. 2. Sipariş Görevlisi Mobil Telefon Uygulaması.....	29
4. 3. 2. 3. Otomasyon Firma (Yönetici) Bileşeni.....	30
5. Tasarım, Gerçekleme ve Test.....	31
5. 1. Satış Görevlisi Web Arayüzü.....	31
5. 1. 1. Temel Sınıf ve Metodlar.....	31
5. 1. 1. 1. Veritabanı Sınıfı.....	32
5. 1. 1. 2. Genel Fonksiyonlar.....	32
5. 1. 1. 3. Kayıt Sınıfı.....	33
5. 1. 2. Tasarım ve Uygulama.....	33
5. 1. 2. 1. Kimlik Doğrulama / Oturum Açma Gerçeklemesi.....	34
5. 1. 2. 2. Sipariş Gönderme İşlemi Gerçeklemesi.....	35
5. 1. 2. 3. Yeni Müşteri Kaydı Gerçeklemesi.....	38
5. 1. 2. 4. Satış Görevlisi İstatistik Görüntüleme.....	39
5. 1. 2. 5. Oturumun Sonlandırılması.....	39
5. 2. Satış Görevlisi Mobil Telefon Uygulaması.....	39
5. 2. 1. Mobil Telefon Bileşeni (J2ME).....	40
5. 2. 1. 1. Oturum Açılması.....	41
5. 2. 1. 2. Sipariş İçeriğinin Girilmesi.....	43
5. 2. 2. Mobil Uygulama Sunucu Bileşeni (Java Servlet).....	44
5. 2. 2. 1. Oturum Yöneten Servlet.....	45
5. 2. 2. 2. Sipariş İletilerinin Yönetilmesi.....	45
5. 3. Veri Madenciliği Algoritması.....	47
5. 3. 1. Temel Sınıf ve Metodlar.....	47
5. 3. 1. 1. Item Sınıfı.....	48
5. 3. 1. 2. Node Sınıfı.....	48
5. 3. 1. 3. Tree Sınıfı.....	49
5. 3. 1. 4. DescendingComparator Sınıfı.....	50
5. 3. 1. 5. HeaderTable Sınıfı.....	51
5. 3. 1. 6. FP_Builder Sınıfı.....	51
5. 3. 1. 7. FPGrowth Sınıfı.....	53
6. Deneysel Sonuçlar.....	56
7. Sonuç ve Öneriler.....	59
7. 1. Web Hizmet Bileşeni.....	59
7. 2. Mobil Telefon Uygulaması.....	59
8. Kaynaklar.....	61

# ŞEKİLLER

Şekil 3. 1: Java Micro Edition Teknolojisinin Katmanları.....	18
Şekil 3. 2: Eşik Değerine Bağlı Olarak Apriori – FP-ağacı Karşılaştırması.....	21
Şekil 3. 3: Hareket Sayısına Bağlı Olarak Apriori – FP-ağacı Karşılaştırması.....	21
Şekil 4. 1: Doğrudan Satış Yaklaşımı Genel İş Akışı.....	22
Şekil 4. 2: Sipariş Genel Durum Diyagramı.....	25
Şekil 4. 3: Otomasyon Sistemi Bileşenleri ve Erişimler.....	26
Şekil 4. 4: Satış Görevlisi Web Arayüzü.....	27
Şekil 4. 5: Satış Görevlisinin Sipariş İletimi İçin Tasarlanan Mobil Arayüz Diyagramı.....	29
Şekil 4. 6: Otomasyon Sistemi Yönetim Bileşeni Modülleri Durum Diyagramı.....	30
Şekil 5. 1: Veritabanı sınıfı UML diyagramı.....	32
Şekil 5. 2: Ürün kodu ve adedi için fonksiyon.....	33
Şekil 5. 3: Satış Görevlisi Web Arayüzü Modeli.....	33
Şekil 5. 4: oturum açılması ve satış görevlisi ana sayfasına (sg_ana) yönlendirilme.....	34
Şekil 5. 5: Oturum sınıfına ait ac() metodu.....	35
Şekil 5. 6: Sipariş İçeriğinin Oluşturulduğu siparis.php sayfası.....	36
Şekil 5. 7: fiyat bilgileri elde edilmesi ve session parametrelerinin güncellenmesi.....	37
Şekil 5. 8: Her türlü bilgisi tamamlanmış bir sipariş sepeti (siparis_onay.php).....	37
Şekil 5. 9: send() metodu.....	38
Şekil 5. 10: Satış Görevlisi Mobil Telefon Uygulamasında Yer Alan Sınıf Tanımları.....	40
Şekil 5. 11: Oturum açma ekranı.....	41
Şekil 5. 12: Yeni t ipliğinin oluşturulması.....	42
Şekil 5. 13: Kimlik denetleme işlemi.....	42
Şekil 5. 14: Alışveriş sepetinin oluşturulması.....	43
Şekil 5. 15: Alışveriş sepeti.....	44
Şekil 5. 16: JDBC MySQL sürücüsünün uygulamaya eklenmesi.....	45
Şekil 5. 17: Kimlik numarası denetlenmesi.....	45
Şekil 5. 18: smarketServerOrder servletinde çağırılan generateOrderString metodu.....	46
Şekil 5. 19: Mobil Telefon ve Servlet İletişimi / Durum Geçişleri.....	47
Şekil 5. 20: Item sınıfı UML diyagramı.....	48
Şekil 5. 21: Node sınıfı UML diyagramı.....	48
Şekil 5. 22: Tree sınıfı UML diyagramı.....	49
Şekil 5. 23: Ağaçta önceden bulunan bir hareketin ağaca işlenmesi.....	50
Şekil 5. 24: Ağaçta önceden bulunmayan bir hareketin ağaca işlenmesi.....	50
Şekil 5. 25: DescendingComparator sınıfı UML diyagramı.....	51
Şekil 5. 26: HeaderTable sınıfı UML diyagramı.....	51
Şekil 5. 27: FP_Builder sınıfı UML diyagramı.....	52
Şekil 5. 28: getTrans metodu.....	52
Şekil 5.29: build metodu.....	53
Şekil 5. 30: FPGrowth sınıfı UML diyagramı.....	53
Şekil 5. 31: getPaths metodu.....	54
Şekil 5. 32: createBase() metodu.....	54
Şekil 5. 33: mine() metodunun ilgili kısmı.....	55
Şekil 6. 1: Yaygın öğelere ilişkin program çıktısı.....	56
Şekil 6. 2: FP-ağacına ilişkin program çıktısı.....	57
Şekil 6. 3: FP-ağacından elde edilen yol sayısına ilişkin program çıktısı.....	57
Şekil 6. 4: Yaygın örüntülere ilişkin program çıktısı.....	57
Şekil 6. 5: Yaygın örüntülere ilişkin program çıktısı (eşik değeri yükseltilmiştir.).....	58
Şekil 6. 6: Promosyon belirleme sayfası.....	58

# 1. Giriş

Projenin ilk adımdaki hedefi, telefon, faks, e-mail vs. üzerinden kayıt dışı bir şekilde yürüyen ve insan faktörüne bağlı olarak çeşitli hatalara meydan veren sipariş sistemini, elektronik ortamda insandan bağımsız işler hale getirerek hata olasılığını ortadan kaldırmak, prosedürü hızlandırmak ve kaydı tutulabilir hale getirmektir.

Bundan sonra gerçekleştirilecek ana hedef ise, müşterinin daha önceki alımlarına veya potansiyel alımlarına göre, çeşitli promosyonları belirlemek ve bunları ilgili satış görevlisi veya müşteriye SMS yoluyla bildirmektir. Projenin ikinci adımı olan bu kısım, ürünlerin ve müşteri tercihlerinin uygun bir ilişkilendirme algoritması kullanılarak sınıflandırılması ve satış hacmini artırmaya yönelik olarak birlikte tüketildiği belirlenen ürünler için firma yönetimine promosyonlar önerilmesidir.

Yapılan literatür taramasında, birebir benzer bir uygulamaya rastlanmamıştır. Ancak problem çözümünde kullanılan ilişkilendirme algoritmasının ana hatlarını belirleyen yaygın öğeler kümelerinin oluşturulması ile çözülmüş birçok problem bulunmaktadır. Bu projede kullanılan algoritma, Jiawei Han, Jian Pei ve Yiwen Yin tarafından hazırlanan ilişkilendirme algoritması[4] üzerinde gerekli bazı değişiklikler yapılarak elde edilmiştir. Bu algoritmanın kullanılmasının uygun olduğu her problem, bu proje ile de temelde benzerdir.

Proje raporu, sekiz ana bölümden oluşmaktadır. İlk bölüm raporun giriş kısmı olup genel olarak projenin çözmeyi hedeflediği problem açıklanmakta, raporda konuların açıklanışı ele alınmaktadır. İkinci kısım, “Proje Tanımı ve Planı”, projenin kısa bir tanımından sonra iş akışının ne şekilde planlandığını ve görev dağılımını içermektedir. “Kuramsal Bilgiler” kısmında zaten herkes tarafından bilinen programların vs açıklanmaması tercih edilmiş, bunun yerine proje için gerekli olan ancak daha önce bilinmediği için ön araştırma gerektiren konular üzerinde durulmuştur. Bunlar mobil telefon uygulaması ve veri madenciliği başlıkları altında toplanmıştır. Dördüncü bölüm olan “Analiz ve Modelleme” kısmında sistemin temel bileşenleri tesbit edilerek bunlar arasındaki ilişkiler ayrıntılı olarak açıklanmıştır. “Tasarım, Gerçekleme ve Test” bölümü, projenin işleyişine ilişkin en önemli bilgileri barındıran kısımdır. Burada projenin modüler yapısı gösterilmiş, modüller ayrıntılandırılarak birbirleri ile ilişkileri üzerinde durulmuştur. “Deneysel Sonuçlar” başlıklı altıncı bölümde, projenin veri madenciliği ile ilgili kısmı üzerinde durulmuş, ilgili algoritmanın çalışmasında kullanılan veriler, algoritmanın farklı basamaklarında elde edilen sonuçlar ve nihai sonuç açıklanmıştır. “Sonuç ve Öneriler” bölümünde proje bir bütün olarak performans açısından değerlendirilmiş, elde edilen sonuçlar fiyat vb. çevre faktörleri ile bağlantılı olarak ele alınmıştır. Son bölüm olan “Referanslar” başlığında projenin tasarım, gerçekleştirme ve dokümantasyon aşamalarında kullanılan her türlü kaynak listelenmiştir.

## **2. Projenin Tanımı ve Planı**

### **2. 1. Proje Tanımı**

Smarket, bir otomasyon ve akıllı market uygulamasıdır. Doğrudan satış yönteminde gerekli olan sipariş alımı süreci, bunun yanı sıra siparişlerin, ürünlerin, satış görevlilerinin ve müşterilerinin kaydının tutulması, tamamen elektronik ortama taşınmıştır. Ayrıca, veri madenciliği ile müşterilerin tüketim alışkanlıkları incelenerek hangi ürünlerin birlikte tüketildiği belirlenmiş, bu ürünlerden bir veya daha fazlasının bir müşteri tarafından satın alınması durumunda diğer ürün veya ürünler için uygun bir promosyon belirleyerek firma yönetiminin onayına sunan bir modül tasarlanmıştır.

### **2. 2. İş Planı**

#### **2. 2. 1. Web ve Veritabanı Arayüzlerinin Gerçeklenmesi**

##### **2. 2. 1. 1. Web arayüzü ve veritabanı tasarımı**

Sayfalarda kullanılacak ve görüntülenecek verilerin tespiti ile bundan hareketle program içindeki veri akışının belirlenmesi, tutulacak veri tabanının modellenmesi (Aslı Filiz, Ali Ulvi Kasapoğlu)

##### **2. 2. 1. 2. Veritabanı kurulumu**

Yapılan veri tabanı tasarımının gerçekleşmesi ve örnek veriler oluşturulması (Aslı Filiz)

##### **2. 2. 1. 3. Web arayüzü görsel tasarımı**

Web arayüzünün görselliğinin; anlaşılabilirlik, kullanım kolaylığı göz önünde bulundurularak sağlanması (Ali Ulvi Kasapoğlu)

##### **2. 2. 1. 4. Web arayüzü kodlaması**

Veritabanı ve görsel tasarım tamamlandıktan sonra, sayfaların işleyişi için gerekli kodlamaların yapılması. Web katmanının firmaya bağlı kısmının programlamasını Ali Ulvi Kasapoğlu, satış görevlisi ve müşteri ilişkileri kısmını Aslı Filiz gerçekleştirmiştir.

##### **2. 2. 1. 5. Web ve veri tabanı arayüzlerinin birlikte test edilmesi**

Veritabanı ve web arayüzünün birlikte sorunsuz işlediğinin test edilmesi. İki modülün entegrasyonunu beraberinde getiren bu test öncesinde dikkat edilmesi gereken en önemli nokta, birleştirilecek olan web ve veritabanının daha önceden test edilip, sorunsuz çalıştıklarından emin olunmasıdır. (Aslı Filiz, Ali Ulvi Kasapoğlu)

## **2. 2. 2. Mobil Arayüzün Gerçeklenmesi**

### **2. 2. 2. 1. SMS kullanımı**

Web üzerinden SMS alma-gönderme hakkında bilgi edinilmesi ve bu doğrultuda gerekli çalışmaların yapılması (Ali Ulvi Kasapoğlu)

### **2. 2. 2. 2. Web, veri tabanı ve mobil arayüzlerin birlikte test edilmesi**

Her üç arayüzün de birlikte sorunsuz işlediğinin test edilmesi (Ali Ulvi Kasapoğlu, Aslı Filiz)

## **2. 2. 3. Veri Madenciliği (Data Mining) ve Promosyon Algoritması**

Bu kısım, daha önce de açıklandığı gibi “Promosyonlar” üzerinden gerçekleştirilecektir. Eldeki veriler kullanılarak oluşturulmuş olan şartlara uygun bir promosyon üretilmektedir. Promosyonların oluşturulmasında kullanılacak olan sözkonusu veriler, iktisat literatüründe “segmentation” olarak geçen; müşterileri aldıkları ürünlere göre tanımlayıp, uygun ürün sınıflarına yönlendirme işlemini esas alınarak işlenecektir.

### **2. 2. 3. 1. Ön araştırma**

Data mining hakkında bilgi toplanması, projeye ne şekilde uygulanabileceğinin belirlenmesi (Aslı Filiz)

### **2. 2. 3. 2. İlişkilendirme algoritması**

Yapılan ön araştırma neticesinde projenin yapısına ve hedeflerine uygun bir ilişkilendirme algoritması belirlenmesi ve uygulanması (Aslı Filiz)

### **2. 2. 3. 3. Veri giriş algoritması**

Veri madenciliği algoritmasının kullanılabilmesi için gerekli miktarda rasgele veriyi belirleyerek veri tabanına giren bir algoritmanın tasarımı ve uygulanması (Ali Ulvi Kasapoğlu)

### **2. 2. 3. 4. Veri madenciliği algoritmalarının birlikte test edilmesi**

Veri giriş ve ilişkilendirme algoritmalarının bir arada sorunsuz işlediğinin test edilmesi (Ali Ulvi Kasapoğlu, Aslı Filiz)

### **2. 2. 3. 5. Web, veritabanı ve veri madenciliği arayüzlerinin birlikte test edilmesi**

Veri madenciliği algoritmasının web arayüzü üzerinden kullanılabilirliğinin ve sonuçlarının izlenebilirliğinin sağlanması (Ali Ulvi Kasapoğlu)