

**İSTANBUL TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ**

Mobil Kent Bilgi Sistemi

Bitirme Ödevi

**Çağlar Çoban-Yekzan Aslan
040000652 - 040000657**

**Bölüm : Bilgisayar Mühendisliği
Anabilim Dalı : Bilgisayar Bilimleri**

Danışman : Yrd. Doç. Dr. Feza Buzluca

Ocak 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,

Ad Soyad

İmza

Ad Soyad

İmza

Bu projenin geliřtirilmesi srecinde her trl bilgi ve desteęini bizden esirgemeyen deęerli hocamız Feza Buzluca'ya ve proje srecinde bize destek veren arkadaşlarımız Grkem Gnay, Kamil Akbul ve Serkan akır'a en iten teřekkrlerimizi sunarız.

Mobil Kent Bilgi Sistemi

(ÖZET)

Ülkemizde ve dünyada kentleşme her yıl giderek artmaktadır. Şehirlerin nüfusları artmakla beraber şehirler durmadan gelişmekte ve genişlemektedir. Bunun sonucu olarak, insanlar için çeşitli problemler ve sıkıntılar ortaya çıkmaktadır. İnsanlar buldukları şehirle ilgili bilgilere ulaşmakta zorlanmaktadır. Bu bilgilere ulaşmak için gerekli kaynakları bulamamaktadırlar.

Bugün büyükşehirlerde insanlar için trafik, adres bulamama, acil işlerde gereken yerlere ulaşma gibi problemler bulunmaktadır. Aynı şekilde insanların eğlence, gezi veya boş zamanlarını değerlendirmek amacıyla ihtiyaç duydukları yerlerle ilgili bilgilere ulaşmaları da güçleşmektedir. Bugün bu sorunlar için verilen hizmetlerin en başında harita kullanımı gelir. Ancak haritanın kullanımı çeşitli zorluklar doğurur. Bu zorluklar, taşınmasının zor olması, güncelleştirme olmaması, üzerinde verilen hizmetlerin kısıtlı olması, kullanıcıdan ve değişikliklerden bağımsız olması olarak sıralanabilir.

Hazırlanan projenin amacı insanları bu konular hakkında bilgi sahibi yapmak, onları buldukları ya da istedikleri yerler hakkında bilgi sahibi yapacak çeşitli hizmetleri web ve mobil hizmet olarak sunmaktır.

Proje, bilgilerin alındığı bir veritabanı, sunucu ve istemcilerden oluşmuştur. İstemciler mobil veya web browser olabilmektedir. Hem mobil hem de web browser için istekler tek sayfaya yapılmakta ve aynı sayfadan her iki platforma birlikte hizmet verilmektedir. Projenin geliştirilmesi için Java'nın jsp, servlet, applet ve j2me teknolojilerinden yararlanılmıştır.

Projede çeşitli yerlerin haritaları bulunmaktadır. Kullanıcı bilgi edinmek istediği yeri seçer ve haritasını web ve mobil ortamda alır. Haritalar ile ilgili dört temel hizmet örnek olarak eklenmiştir. Bunlar eczane, sinema, hotel ve durak hizmetleridir. Eczane hizmeti kullanıcıya o yerdeki eczaneleri harita üzerinde göstermekte ve istenildiği takdirde o gün nöbetçi olan eczane bilgisi verilmektedir. Sinema hizmetinde de o yerde bulunan sinemalar harita üzerinde gösterilmekte, sinemalarda oynayan filmler ve seansları ile bilgi verilmekte ve seçilen bir filmin hangi sinemalarda gösterimde olduğu bilgisi verilmektedir. Kullanıcı hotel hizmetini seçtiği takdirde harita üzerinde o yerde bulunan hotelleri görecektir ve hoteller ile ilgili telefon numarası, kalitesi gibi bilgileri alacaktır. Durak hizmeti seçildiğinde yine haritada duraklar gösterilecek, kullanıcıya istediği durak hakkında otobüs ve hat bilgileri gösterilecek ve ayrıca gitmek istediği yere en yakın durak ve oraya giden otobüsler bilgisi verilecektir.

Projenin temel özelliği bu örnek hizmetlere bağlı kalmaması ve güncelleştirmeye açık olmasıdır. Proje ile ilgili bilgiler bir veritabanında tutulmaktadır. Veritabanında yapılan bir değişiklik projeye yansımaktadır. Yeni bir harita eklendiğinde, harita üzerinde yeni bir durak, sinema vs. eklendiğinde yeni bir hizmet eklendiğinde kullanıcı kısmına yeni bir kod eklenmeden yeni girilen bilgiler, farklı bir hizmet olarak uygulamaya yansımaktadır.

Mobile City Information System

(SUMMARY)

Urbanization is increasing rapidly every year in our country. Population is increasing in cities and cities are developing and becoming larger. This comes with new problems and difficulties for people. The information of the cities are becoming more difficult for people to get. People can not find resources for these information.

There are many problems in big cities. One of the biggest is the traffic. The roads are very crowded and people waste their time on the roads. This problem can not be solved immediately but a service that tells the situation of the roads should be given to people. In this case people can route their ways according the situation of the roads. Another is the difficulty for finding address. People solve this problem by asking people where they want to go. The other one is the necessary places in emergency. People can need to know the night-duty pharmacist at night in an emergency situation. In addition, people want to know the places and their activities for entertainment. Getting this information are difficult today. The services for these are limited.

One of the classic solution is using a map. But map comes with new problems. Firstly, it is difficult to carry. If you want to go out and visit some places in a big city, you should get a big map of the city or the maps of the parts. You will need a big place to carry these big map and maps. Also it is difficult to use the map. Finding a place in a big map or searching the parts can be very disturbing. Also a the services on a map is limited you can not give all the services on a single map. At this time maps can be very confusing and complex. Maps are static and you can not add new services on a map. You have to buy a new map of this service. As a result if your solution is the map, you will need a lot of maps. As you can see map is not the right solution.

The right solution is getting these information from mobile devices and from the internet. It is easy to carry and get the information, easy use, many maps with many services, fast access in emergency situations and load one and use permanent.

The aim of the project is to give these information with some services to people. The services are the bus stops, pharmacists, cinemas and hotels. People can get these from a mobile phone or from the internet. People can get the maps of the places and can get these services on the map. The services are not limited. The software can be updated and new services and new places can be added.

The project is composed of a database that the information is stored, a server and clients. The clients can be web browser and a device that supports java mobile technology. The requests are sent to the same web page and this same web page services for both of the platforms. When the request comes to the server from client, server first recognizes the platform of the request. It then directs the request to another page according to the source of the request. The server makes the connection to the database and gets the recent information from the database and sends them to client. And the program continue between the requests from client that the user wants and the server responses. For the development of the project java, jsp, servlet, applet and j2me technologies are used.

Microsoft Access is used for database of the project. It was developed by Microsoft Corporation. The project uses JDBC (Java Database Connectivity) API for accessing database. For accessing Jdbc-Odbc Bridge is used. The Jdbc-Odbc Bridge is a Jdbc driver that implements jdbc operations by translating them to odbc operations. To odbc it appears as a normal application program. The bridge implements jdbc for any database for which an odbc driver is available. The bridge is implemented as the sun.jdbc.odbc Java package and contains a native library used to access odbc [1].

Apache Tomcat Web Server 5.5.12 is used in the server. Apache Tomcat is the servlet container that is used in the Official Reference Implementation for the Java Servlet and Java Server Pages Technologies. Apache Tomcat is developed in an open and participatory environment and released under the Apache Software License. Apache Tomcat is intended to be collaboration of the best-of-breed developers around the world. Apache Tomcat powers numerous large scale, mission-critical web applications across a diverse range of industries and organizations[2].

Servlets are the Java platform technology of choice for extending and enhancing Web servers. Servlets provide a component-based, platform-independent method for building Web-based applications, without the performance limitations of CGI programs. And unlike proprietary server extension mechanisms (such as the Netscape Server API or Apache modules), servlets are server- and platform-independent. This leaves you free to select a "best of breed" strategy for your servers, platforms, and tools[1].

Servlets have access to the entire family of Java APIs, including the JDBC API to access enterprise databases. Servlets can also access a library of HTTP-specific calls and receive all the benefits of the mature Java language, including portability, performance, reusability, and crash protection[1].

Today servlets are a popular choice for building interactive Web applications. Third-party servlet containers are available for Apache Web Server, Microsoft IIS, and others. Servlet containers are usually a component of Web and application servers, such as BEA WebLogic Application Server, IBM WebSphere, Sun Java System Web Server, Sun Java System Application Server, and others[1].

JavaServer Pages (JSP) technology enables Web developers and designers to rapidly develop and easily maintain, information-rich, dynamic Web pages that leverage existing business systems. As part of the Java technology family, JSP technology enables rapid development of Web-based applications that are platform independent. JSP technology separates the user interface from content generation, enabling designers to change the overall page layout without altering the underlying dynamic content[1].

The Java 2 Platform, Micro Edition (J2ME) provides a robust, flexible environment for applications running on consumer devices, such as mobile phones, PDAs, and TV set-top boxes, as well as a broad range of embedded devices. Like its counterparts for the enterprise (J2EE), desktop (J2SE) and smart card (Java Card) environments, J2ME includes Java virtual machines and a set of standard Java APIs defined through the Java Community Process, by expert groups whose members include leading device manufacturers, software vendors, and service providers[1].

J2ME delivers the power and benefits of Java technology to consumer and embedded devices. It includes flexible user interfaces, a robust security model, a broad range of built-in network protocols, and extensive support for networked and offline applications that can be downloaded dynamically. Applications based on J2ME specifications are written once for a wide range of devices, yet exploit each device's native capabilities[1].

The J2ME platform is deployed on millions of devices, supported by leading tool vendors, and used by companies worldwide. In short, it is the platform of choice for today's consumer and embedded devices[1].

The J2ME architecture comprises a variety of configurations, profiles, and optional packages that implementers and developers can choose from, and combine to construct a complete Java runtime environment that closely fits the requirements of a particular range of devices and a target market. Each combination is optimized for the memory, processing power, and I/O capabilities of a related category of devices. The result is a common Java platform that takes full advantage of each type of device to deliver a rich user experience[1].

Configurations comprise a virtual machine and a minimal set of class libraries. They provide the base functionality for a particular range of devices that share similar characteristics, such as network connectivity and memory footprint. Currently, there are two J2ME configurations: the Connected Limited Device Configuration (CLDC) and the Connected Device Configuration (CDC) [1].

To provide a complete runtime environment for a specific device category a configuration must be combined with a profile, a set of higher-level APIs that further define the application life-cycle model, the user interface, and access to device-specific properties. A profile supports a narrower category of devices within the framework of a chosen configuration. A widely adopted example is to combine CLDC with the Mobile Information Device Profile (MIDP) to provide a complete Java application environment for cell phones and other devices with similar capabilities[1].

The J2ME platform can be extended by adding various optional packages to a technology stack that includes either CLDC or CDC and an associated profile. Created to address very specific application requirements, optional packages offer standard APIs for using both existing and emerging technologies such as database connectivity, wireless messaging, multimedia, Bluetooth, and web services. Because optional packages are modular, developers can avoid carrying the overhead of unnecessary functionality by including only the packages an application actually needs[1].

In the project, various places' maps are stored. For the web part of the project, firstly the places that are supported listed in a combo box. The user can choose one of them. When clicked, the map of the selected place is shown on the screen with the most known places of that place. At the same time the services are shown at the bottom of the map. The user can choose one of cinemas, hotels, bus stops, pharmacies and hotels. When the user chooses one of them he/she is direct to a new page about that service for this place.

In the bus stop service, the map is stored as an applet. The applet for that service is composed of three parts. One part is a panel that holds the map and listens for the mouse events, the other part is another panel that holds a combo box and a button. The third part is a panel that holds a textarea that shows the informations about the buses. User clicks on the

map to show his/her coordinates that he/she presents. At the bottom of the applet there is the places that user can go from that place by buses. They are listed in a combo box. When the user selects one of them and clicks the button the applet shows the nearest bus stop he can go that selected place that has a bus going to that place and the names of the buses at the bottom of the applet. At the same time the nearest bus stop is shown by a green arrow on the map and users can easily see the nearest bus stop according to the place that they click on the applet.

In the cinema service, map is on the applet too. The applet for that service is composed of four parts. Each part is a panel. One of them is the panel that holds the map. In this map cinemas are shown that places in this place. On one of the panels, there is a combo box that lists the cinemas on that place. When the user selects one of the cinemas from the combo box and clicks the button, he can see the movies that are shown in that cinemas on the text area at the bottom of the applet. In another panel there is service for movies. There is also a combo box at that panel that lists the movies that are shown on the cinemas at that place. When the user selects a movie from the combo box and presses the button, he can see the cinemas that the selected movie is shown on the text area.

In the hotel service, the user can see the map with the the hotels that are in that places on the map. User can select one of the hotels from the combo box and he gets the phone number and the quality of the hotel on the text area. To show the user the place of the hotel a green arrow is used.

In the pharmacy service, the map comes with the pharmacies on it to the screen. User can see all of the pharmacies in that place on the map. This service also shows the pharmacy on night-duty.

In the mobile platform, the same services are included. At the beginning of the program client makes a connection to the server and get the names of the maps and services from the database. User can select one of them. According to the selection the map of the place and the information about the service comes and shown on the map on the mobile phone. There are two types of menus. One is the general menu that does not depend on any selection from the map and a specific menu that depends on the item selected on the map. In the mobile platform you can see the names of cinemas, hotels, pharmacies and bus stops when you are on it on the map. You can click anywhere to see general menu or you can come on a item and get the specific menu. The information about the selected menu item is shown on a separate browser screen.

Although the project includes only these services, the most important feature of the project is it can easily be updated and new services can be added without need of new code on the client side. All of the information is read from database and these names are used in the code. Services are also read from the database. Thus when a new service or new information is added to the database, the project need not to be rewritten. The new part is immediately used when the project restarted.

İÇİNDEKİLER

1- GİRİŞ.....	1
2- PROJENİN TANIMI VE PLANI.....	2
2.1. Projenin Tanımı	1
2.2. Projenin Planı.....	1
3- KURAMSAL BİLGİLER.....	1
4- ANALİZ VE MODELLEME.....	2
4.1. Analiz.....	1
4.2. Modelleme	1
5- TASARIM, GERÇEKLENME VE TEST.....	2
5.1. Tasarım ve Gerçekleme.....	1
5.2. Test.....	1
6- DENEYSEL SONUÇLAR.....	1
7- SONUÇ VE ÖNERİLER.....	3
7.1. Performans.....	1
7.2. Fiyat.....	1
7.3. Öneriler.....	1
8- KAYNAKLAR.....	1

1 GİRİŞ

İstanbul gibi dünyadaki bütün büyükşehirlerdeki, belli bir adrese ulaşma veya kent üzerindeki bir bilgiye –bu bilgi ve adresler duraklar, eczaneler, hoteller, tarihi yerler, sinema filmleri olabilir- sorunu, insanları bu karmaşıklığı çözmeye yöneltmiş ve bunun sonucunda kent bilgilerini mümkün olduğunca içeren kent haritaları yapılmıştır. Fakat insanların bir kentle ilgili ihtiyaç duydukları veriler arttıkça, bu haritalar da işlevselliğini yitirmiştir. Özellikle bu tür haritaların taşınmasının zor olması, ihtiyaç duyulan bilginin türüne göre farklı haritaların edinilme zorunluğu, belli bir süre sonra şehirlerdeki hızlı değişimlerden kaynaklı olarak haritaların kullanılabilirliğinin azalması gibi sorunlar, yazılım geliştiricileri günlük hayatta kullanılacak dijital harita temelli sistemler geliştirmeye itmiştir. Fakat bu sistemler, çoğunlukla bilgisayarlar veya belli bir maliyet ve taşıma sorunu olan GPS ve el bilgisayarları için yapılmıştır. Projemizde bu sistemin (Mobil Kent Bilgi Sistemi), insanların birçoğunun günün her saati yanında buldukları cep telefonları için geliştirilmesi hedeflenmiş, bunun yanında diğer sistemlerde de çalışabilmesini sağlayacak alt yapıyı hazırlamak amaçlanmıştır.

Bu konuda, özellikle son 2-3 yılda Türkiye’de ve son 5-6 yılda dünyada bazı firmalar çalışma yapmışlardır.

Infotech: Mobil GIS (Coğrafi Bilgi Sistemi) teknolojisini bir WAP sayfası olarak mobil telefon kullanıcıları sunmaktadır.

Turkcell ve Telsim: Kullanıcılarına sayısal haritaları bir WAP Browser aracılığıyla Göstererek bazı hizmetler vermektedir.

Sayısal Grafik: Her yıl düzenlediği GIS seminerlerinde Mobil GIS konusu da ele alınmaktadır.

Bunların dışında dünyada birçok firma, bu konularda son yıllarda çözümler geliştirmiştir.

Geliştirilen proje, kullanıcıların mobil telefonunda veya internet bağlantısı olan el bilgisayarları, dizüstü bilgisayarları ve benzeri taşınabilir cihazlarda, istediği şehre ait istediği semtle ilgili belirlenmiş hizmetlerden faydalanmalarını sağlamaktadır. Sistemin güncellenebilir olması ve internet bağlantısı olan birçok cihazda kullanılabilir olması, projenin ulaştığı önemli hedeflerdendir. Buna göre kullanıcı, seçime bağlı olarak o anda sistemde bulunan son verilere göre, bir ilin istediği semtini seçebilir ve o anda verilen bütün hizmetlerden yararlanabilir. Örneğin, İstanbul-Beşiktaş’ta bulunan biri, o anda Beşiktaş’taki hotellerle ilgili bilgilere –konum, özellik, telefon vb.- ihtiyaç duyuyorsa, programdaki seçimleri buna göre yapıp, ihtiyaç duyduğu verilere ulaşabilir. Eğer sistemde bulunmayan bir ilçenin de, sisteme dahil olması isteniyorsa, bunun için o ilçeye ilgili bütün gerekli verilerin, sunucunun veri tabanına eklenmesi yeterlidir. Yani, kullanıcının bu ilçe için, tekrardan programı yüklemesi veya güncellemesine gerek yoktur. Bunun sağladığı en önemli fayda, kullanıcıya ek maliyet çıkarmamak ve program kodlarına müdahale edilmeden zaman kazanmaktır. Bu da, günümüz yazılım geliştiriciliğinin önemli bir sorununu çözen bir yaklaşımdır.

Projenin genel bir tanımı, projenin bölümleri ve bu bölümlerin projeye başlanırken öngörülen bitiş zamanları Bölüm-2’de;

Proje hazırlanmadan önce projenin yapılabilmesi için gerekli kuramsal bilgilerin neler olduğu ve seçilen bazı geliştirme yöntem ve araçlarının neden seçildiği Bölüm-3’te;

Projenin bir problem olarak bileşenleriyle beraber açıklandığı ve bu probleme uygun bir çözümün modellenmesi Bölüm-4'te;

Problemin çözülmesi için tasarlanmış olan bütün bileşenlerin birbirleriyle ilişkileri, nasıl işledikleri ve bunların şekilsel gösterimleri Bölüm-5'te;

Sistemin gerçekleşmesinden sonra yapılan denemelerden ne gibi sonuçların çıktığı Bölüm-6'da;

Performans ve fiyat gibi kriterlerin, gerçekleşen sistem için yapılan değerlendirilmeleri

Bölüm-7'de;

Başvurulan kaynakların listesi ise Bölüm-8'de işlenecektir.