

**İSTANBUL TEKNİK ÜNİVERSİTESİ  
ELEKTRİK-ELEKTRONİK FAKÜLTESİ**

**ULAŞIM ROTASI BELİRLEME YARDIMCISI**

**Bitirme Ödevi**

**Pınar Cana  
040010417**

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

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

Aralık 2006

**İSTANBUL TEKNİK ÜNİVERSİTESİ  
ELEKTRİK-ELEKTRONİK FAKÜLTESİ**

**ULAŞIM ROTASI BELİRLEME YARDIMCISI**

**Bitirme Ödevi**

**Pınar Cana  
040010417**

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

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

Aralık 2006

## **Özgünlük Bildirisi**

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

İstanbul, 08.01.2007

Pınar Cana

# ULAŞIM ROTASI BELİRLEME YARDIMCISI

## ( ÖZET )

Günümüzde büyük şehirlerde bir yerden bir yere ulaşımında güzergah belirleme git gide daha büyük bir sorun olmaya başlamıştır. Ulaşım rotası belirleme yardımcısı projesindeki amaç bu konuda kullanıcıya alternatifler sunan bir asistan oluşturmaktır. Bütün olasılıkları göz önünde bulundurarak, hızlı sonuç üreten bir yöntem olması nedeniyle de pek çok kişi ve kurum bu tür yazılımları kullanmaktadır. Ulaşım rotası belirleme yardımcısı da şehirci ulaşımında örnek bir otobüs ağı için aktarma bilgisi ile birlikte kullanıcıya en uygun sonuçları belirlemektedir.

Program sunucu istemci mimarisinde, Tomcat Server üzerinde SOAP web servisleri kullanılarak gerçekleştirilmiştir. Web servis teknolojisi uzaktaki bir sistemde var olan hizmete ağ üzerinden erişimi sağlamaktadır. Bu sayede istemci ve sunucu birbirinden bağımsız geliştirilebilmektedir. Simple Object Access Protokoll (SOAP) ile web servisinin sağlamış olduğu fonksiyon uygun parametrelerle uzaktan çağrılabilmekte, istemciye sunucu tarafında elde edilmiş sonuçlar gönderilmektedir. Hem sunucu hem de istemci Java tabanlı uygulamalardır. İstemci tarafını mobil ve web istemcileri oluşturmaktadır.

Projede ulaşım ağını oluşturan hatlar ve duraklar bir graf üzerinde ifade edilmiş, asistan graf üzerinde en kısa yol algoritmaları ( Shortest Path - E.W. Dijkstra) uygulanılarak gerçekleştirilmiştir. Projenin geliştirme sürecinde karşılaşılan sorunlar yazılım projelerinde yaşanan problemlere çözüm üretme yeteneği konusunda büyük katkılar sağlamış, projenin başlangıcındaki proje planı ve analiz sürecinin önemini kavramamı sağlamış ve yazılım mühendisliğindeki vizyonumu genişletmiştir.

# TRANSPORTATION ROUTE PLANER

## ( SUMMARY )

Urban transportation in big cities becomes a more complicated problem in daily life every day. The complexity of transportation networks and existence of several alternative paths confuses the passengers. Transportation route planner project aims to provide an assistance service which gives the user the ability to plan his/her journey with its alternatives using a computer, PDA or a mobile phone. The planner calculates the most suitable journey for given start and end stations in an example urban bus network.

The main constraint of such a project is the level of accessibility and interoperability. Transportation Route Planner has a service oriented / server-client architecture, implementing SOAP xml web services. Using a standard service communication interface gives us the ability to develop clients on various platforms. During the project a web and cellular phone & PDA client using JME is already developed. It is also possible to develop further clients using any language on any platform where it is possible to communicate with a standard RPC style SOAP web service.

The core of the system depends on the representation of the transportation network on a weighted graph, where nodes represent stops, edges represent lines and the weight of an edge stands for the required time to journey from one node to another. The station and line information is stored in a database and the graph is built run-time to make the changes on the network take effect immediately. The famous Shortest Path Algorithm from Edsger Wybe Dijkstra, which is in several routing algorithms widely in use today, is used to determine the shortest journey. The system has a central unique interface which is implemented by a web service and used by the client applications to query the stations and request for a route plan.

In general a web service is defined as an application accessible to other application over the Web. XML & SOAP Web service are a standardized way of integrating Web-based applications using the XML, SOAP, and WSDL open standards over an internet protocol. XML is used to tag the data, SOAP is used to transfer the data, WSDL is used for describing the services available. The power of xml web services depends mostly on the interoperability, availability of open source useful tools and reference implementations and the success of the service oriented architecture.

In this project an Apache Tomcat Application Server with Axis, a powerful SOAP implementation from Apache community, is used as software infrastructure on the server side. The JSP web client uses also Axis API to perform soap requests. On the mobile platforms such as cellular phones or PDA devices, due to memory limitations and abilities of the underlying runtime environment, a lightweight SOAP API, *kSOAP* is preferred.

# İÇİNDEKİLER

1 GİRİŞ.....	6
2 PROJENİN TANIMI VE PLANI.....	7
2.1 Proje Tanımı .....	7
2.2 Proje Planı.....	7
3 KURAMSAL BİLGİLER.....	10
3.1 XML web servisleri .....	10
3.2 Dijkstra En Kısa Yol Algoritması.....	12
4 ANALİZ VE MODELLEME .....	13
4.1 Sistem Mimarisi.....	13
4.2 Ulaşım Ağı Varlık Modeli .....	14
4.3 Uygulama Nesneleri .....	15
5 TASARIM, GERÇEKLEME VE TEST.....	22
5.1 Projenin Gerçeklenmesi .....	22
5.2 Kullanıcı Arayüzü.....	22
6 DENEYSEL SONUÇLAR .....	25
7 SONUÇ ve ÖNERİLER .....	26
8 KAYNAKLAR .....	27

# 1 GİRİŞ

Ulaşım rotası belirleme yardımcısı projesi, bir ulaşım ağındaki duraklar ve hatların bilgisi ile bir yerden bir yere en uygun şekilde nasıl ulaşılacağına dair kullanıcıya öneride bulunan bir uygulamadır. Şehir içi ulaşım için örnek bir otobüs ağı oluşturularak iki durak arasında hangi yolun izlenmesi gerektiğine karar verilecektir.

Yapılan araştırma sonucunda problemin graf problemi olduğuna karar verilmiştir. En kısa yol bulma algoritması olarak, Dijkstra algoritmasının pozitif ağırlıklı kenarlara sahip graflar için kullanılan en yaygın ve en iyi çözüm olduğu araştırmalar sırasında öğrenilmiş olduğundan, bu algoritma üzerinde probleme uygun değişiklikler yapılarak kullanılmıştır. Ulaşım ağını oluşturan hatlar ve duraklar bir graf üzerinde ifade edilmiş, graf üzerinde Dijkstra algoritması baz alınarak geliştirilmiştir.

Proje sonucunda grafın program içinde ne şekilde implemente edildiğinin hem programın efektif olması açısından hem de programcı açısından büyük önem taşıdığı görülmüştür. Web servislerinin kullanımı da projenin geliştirilmesinde büyük kolaylık sağlamıştır.

## 2 PROJENİN TANIMI VE PLANI

### 2.1 Proje Tanımı

Projenin amacı güzergah belirlemek için oluşturulmuş örnek ağ üzerinde en iyi sonuçları hızlı bir şekilde üretmek ve uygulamanın kullanmasında kolaylık sağlamaktır. Bu nedenle proje üç ana parçadan oluşmaktadır. Bunlardan ilki projenin web servis kullanılarak gerçekleştirilmiş sunucudur. İkincisi kullanıcı tarafındaki web istemcisi, üçüncü ise yine kullanıcı tarafındaki mobil istemcidir. Böylece kullanıcının servisten yararlanabilmesi için esneklik sağlanmıştır.

Projenin aşamaları:

- Aktarmasız ulaşım önerisi
  - Araç ve durak bilgilerinin oluşturulması.
  - Trafik ağının oluşturulması.
- Aktarmalı ulaşım önerisi
  - İki nokta arasındaki en kısa ve az sıçramalı yolun bulunması

Araçların duraklara varış saatleri göz önünde bulundurularak yapılacak güzergah önerisi ise planlanmış fakat süre yetersizliği nedeniyle henüz uygulanmamıştır.

### 2.2 Proje Planı

#### 2.2.1 İhtiyaç Analizi

Projenin hangi aşamaları kapsayacağına ve geliştirme sırasında hangi teknolojilerin kullanılacağına Yrd. Doç. Dr. Feza Buzluca'nın danışmanlığında karar verildi. Öncelikle hem sunucu hem de istemci tarafının Java tabanlı uygulama olmasına karar verildi. Uygulamanın sahip olması gereken temel özellikleri web servis teknolojisi kullanılarak uyaktan çağrılabilen sunucu ve sunucunun yordamını çağırarak çalışan web ve mobil istemcilerdir. Java diliyle yazılmış olması nedeniyle platformdan bağımsız olan proje Windows işletim sisteminde geliştirilmiştir. İhtiyaçlar doğrultusunda projenin geliştirilmesinde gerek duyulan tüm yazılım ürünleri açık kaynaklı ve ücretsiz araçlardan oluşmaktadır.

Tasarım ve geliştirme sürecinde ihtiyaç duyulan yazılımlar ve araçlar:

- |                  |  |
|------------------|--|
| • Sun JDK 1.5.02 | <i>Sun Java uygulama geliştirme kiti</i>   |
| • Eclipse 3.2.1  | <i>Java ile uygulama geliştirme ortamı</i> |
| • Tomcat Server  | <i>Uygulama sunucusu</i>                   |
| • MySQL 5.0      | <i>Açık kaynaklı veritabanı</i>            |

- ArgoUML *UML diyagramı tasarım ve çizim aracı*
- Gannt Project *Proje zamanlama, Gannt diyagramı çizim aracı*
- KSOAP 1.2 *J2ME için SOAP API'si*
- Sun WTK for CLDC 2.5 beta *J2ME için uygulama geliştirme platformu*

## 2.2.2 Proje Adımları

Projenin ihtiyaç analizinin yapılmasının ardından izlenecek adımlar belirlendi. Sistemin genel tasarımının yapılmasının ardından, kullanılacak teknolojiler üzerine bir araştırma sürecinin ardından veri modeli, nesne modeli, implementasyon, test ve dökümantasyon aşamalarının gerçekleştirilmesi projede yer alan adımlardır.

### 2.2.2.1 Sistem Tasarımı

Sistem tasarımı adımına benzer projelerin incelenmesi, yazılım sahip olacağı mimarinin ve kullanılacak teknolojilerin belirlenmesi amacıyla iki hafta süre ayrıldı. Bu süre boyunca benzer amaçlar taşıyan masaüstü uygulamaları incelendi. Sistemin genel yapısının tasarlanmasının ardından bu yapıyı gerçeklemek üzere en uygun teknolojilerin araştırılması adımına geçildi.

#### Araştırma Süreci

Çok kullanıcı, yoğun veritabanı erişimi yapacak olan bu tür bir proje için en yüksek performansı sağlayacak ve geliştirme sürecini düşürmede katkıda bulunabilecek yazılım teknolojilerinin araştırılması için iki hafta zaman ayrıldı. Mevcut yazılım teknolojileri hakkında yalnızca dökümantasyonlar üzerinden araştırma yapmak çok da verimli bir yol olmadığı için küçük uygulamalar yazılarak, teknolojilerin performans, geliştirme kolaylığı, problemlere uygunluğu açısından karşılaştırılmasının ardından veri haberleşmesi ve veritabanı erişiminde kullanılacak yöntemler belirlendi.

### 2.2.2.2 Varlık modelinin oluşturulması

Geliştirme sürecinde aşağıdan yukarıya (bottom-up) tasarım süreci izleneceği için ilk önce sistemin temel yapısına ait varlıkların aralarındaki ilişkilerin tanımlanması (entity model). Varlık & ilişki modeline ait diyagramların oluşturulması ve modelin kullanılacak veritabanında yaratılması işlemleri gerçekleştirildi.

### 2.2.2.3 Nesnel ve dinamik modellerin oluşturulması

Yaklaşık bir haftalık bir süre içerisinde uygulama nesneleri, varlık nesnelерinin uygulama içinde ifade edilmesi, bu nesnelerin birbirleriyle ilişkilerinden ve yazılım akışından meydana gelen dinamik modellerin oluşturulması, bu modellerin UML diagramlarına dökülmesi.

### 2.2.2.4 Uygulama ve Test

Oluşturulan modellere dayanarak yazılımın kodlanması ve test edilmesi için sekiz haftalık

süre ayrıldı. Bu adım boyunca her kod parçası bittikten sonra gerekli birim testler, ve entegrasyon testleri yapıldıktan sonra uygulamaya devam edildi. Uygulama sürecinde karşılaşılan zorluklar, modelleme adımlarda fark edilmemiş bazı sorunlar nedeniyle zaman zaman iterasyonlar yapılarak modellerde değişiklikler yapılması söz konusu olmaktadır. Bu durumlarda daha önceki adımlara zaman zaman geri dönüp, gerekli değişikliklerin yapılması gerekmiştir.

### **2.2.2.5 Dökümantasyon**

Projenin son iki haftası dökümantasyon hazırlanmasına ayrıldı. Proje planlarının gözden geçirilmesi, proje sırasında karşılaşılan zorluklar ve çözümlerinin anlatılması ve proje sonucunda elde edilen deneysel çıkarımlar proje raporuna yazılmak üzere bir araya getirildi.