**YAZ / 2018**

**İSTANBUL TEKNİK ÜNİVERSİTESİ**

**Bilgisayar ve Bilişim Fakültesi**

**STAJ**

**Fatih KURU**

**150140040**

**JAVA VE SQL İLE TABLO DÜZENLEME**

**İstanbul Teknik Üniversitesi**

**Bilgisayar ve Bilişim Fakültesi**

**STAJ RAPORU**

Akademik Yıl:2017-2018

Staj yapılan dönem:☐Yaz ☐Bahar ☐Güz

**Öğrenci ile ilgili bilgiler**

|  |  |
| --- | --- |
| Adı ve Soyadı: | Fatih KURU |
| Öğrenci Numarası: | 150140040 |
| Bölüm: | Bilgisayar Mühendisliği |
| Program: | %30 İngilizce |
| E-posta Adresi: | [kurufa@itu.edu.tr](mailto:kurufa@itu.edu.tr) |
| (Cep) Tel No: | 05362640975 |
| ÇAP öğrencisi misiniz? | ☐Evet (ÇAP yaptığınız Fakülte/Bölüm: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)  ☐ Hayır |
| Mezuniyet  durumunda mısınız? | ☐ Evet  ☐ Hayır |
| Yaz okulunda ders alıyor musunuz? | ☐Evet (Ders sayısı: \_\_)  ☐ Hayır |

**Öğrencinin çalıştığı kurum ile ilgili bilgiler**

|  |  |
| --- | --- |
| İsmi: | Akbank T.A.Ş |
| Birimi: | Teknoloji ve Operasyon |
| Web Adresi: | www.akbank.com |
| Kısa Adresi: | Akbank Genel Müdürlüğü Sabancı Center 4.Levent 34330 İstanbul |

**Yetkili kişi ile ilgili bilgiler**

.

|  |  |
| --- | --- |
| Bölümü: | Teknoloji ve Operasyon |
| Unvanı: | Müdür |
| Adı ve Soyadı: | Erkan Saip DUMAN |
| (Kurumsal) E-posta: | erkan.duman@akbank.com |
| (Kurumsal) Tel. No.: | +90 262 686 28 12 |

**Yapılan iş ile ilgili bilgiler**

|  |  |
| --- | --- |
| Staj yeri | ☐Türkiye  ☐Yurtdışı |
| Staj başlangıç tarihi | 18.06.2018 |
| Staj bitiş tarihi | 13.07.2018 |
| Stajda çalışılan net **gün** sayısı | 20 |
| Staj süresince sigortanız var mıydı? | ☐Evet, İTÜ tarafından sigortalandım.  ☐Evet, kurum tarafından sigortalandım.  ☐Hayır, yurtdışı stajı yaptım.  ☐Hayır. |

**1.KURULUŞ HAKKINDA BİLGİLER**

Akbank özel sermayeli bir ticaret bankası statüsünde 30 Ocak 1948 tarihinde Adana’da Sabancı ailesi tarafından kurulmuştur. 1954 yılında ise birçok yerde şube açtığından dolayı Genel Müdürlüğü’nü İstanbul’a taşımıştır. Akbank’ın banka olarak kapsadığı faaliyet alanları, kurumsal ve yatırım bankacılığı, ticari bankacılık, KOBİ bankacılığı, bireysel bankacılık, ödeme sistemleri, Hazine işlemleri ve özel bankacılık ile uluslararası bankacılık hizmetleriyle sınırlıdır. İstanbul’da Genel Müdürlük faaliyetlerini sürdüren Akbank 22 bölge müdürlüğü, 800 şubesi ve 14 binin üzerinde çalışanıyla bankacılık alanında Türkiye’nin önde gelen bankası ve işverenlerinden biri olmasını sürdürmektedir. Aynı zamanda dijital bankacılık alanında sektörün en iyi uygulamalarından biri olan Akbank Direkt ’in geliştirilmesi ve kurum içi uygulamaların modern çağın gerekliliklerinin sağlaması Akbank’ın en önem verdiği konulardandır.

Teknoloji ve Operasyon birimi Gebze Şekerpınar’da yer alan Akbank Bankacılık Merkezi adıyla kullanılan işyerinde faaliyetlerini sürdürmektedir. Bilişim teknolojileri kısmında mobil bankacılıktan *ERP*(Enterprise Resource Planning)’ye kadar birçok alanda küçük alt birimler mevcuttur. Bilişim teknolojileri biriminde *Scrum* proje yönetim biçimi kullanılır. Benim de içinde yer aldığım birim BT Mimari ve Temel Bankacılık Uygulama Geliştirme bölümü, temel bankacılık taleplerine cevap verecek 9-10 kişilik *Scrum* takımlarından oluşur. İçinde yer aldığım *Scrum* takımının görevi *Fraud*(Dolandırıcılık) olmasını engelleyen *outsource* kaynaklarını kullanarak dolandırıcılıkları tespit eden uygulamalar geliştirmek ve bu uygulamaları kullanan kurum içi çalışanlarının taleplerine karşılık vermektir. Herhangi bir uygulamada ya da bir temel bankacılık kurallarındaki bir değişiklikte *Fraud* kontrollerinin de değişmesi gerektiğinden, ekip diğer ekiplerle iyi iletişim halinde olmak zorundadır. Takımda 2 analizci, 1 testçi, 6 *developer* ve 1 *product owner* yer almaktadır. Takımda veri tabanı erişimleri ve işlemleri için TOAD ve ORACLE SQL DEVELOPER, Java kodu yazmak için IDE olarak IntelliJ IDEA ya da Eclipse, web servis ve test için ise SOAP kullanılıyor. Bilgisayarlarda windows10 kullanılmaktadır ve takımlar arası iletişim için CISCO Jabber yazılımı ya da mail yoluyla sağlanıyor.

**2. GİRİŞ**

Okul hayatım boyunca öğrendiğim bilgilerin iş hayatındaki karşılığını Akbank’ta 20 günlük bir staj sürecinde aldım. Stajın konusu yeni bir proje olan güvenlik ekranı kimlik kontrol ekranına eski programdaki verileri önişleme yaparak ve eski verileri anlamlandırarak kimlik kontrol tablolarındaki verilerin cinsinden kimlik kontrol tablosuna geçişini sağlamak. Bunu yapmamızdaki amaç eski programdaki verilerin yeni program için anlamlı olmaması ve yeni programın bu eski verilerdeki bilgilere ihtiyacı olmasıdır. Bu geçişin sağlanması için veri tabanı kısmında Akbank’ın kendi sunucularına bağlanmak için Oracle SQL Developer ve kod kısmı için ise Eclipse IDE’sini kullanarak Java ile yazılmıştır.

**3.STAJ PROJESİNİN TANIMI VE ANALİZİ**

Akbank’ta işlerin ve çalışanların daha koordine olması için *Scrum* metodolojisini kullanılıyor. Staj projesinde ben de bu metodolojinin bir parçası oldum. Aşağıda projenin kapsamında yer alan *Scrum* toplantıları ve metodolojisi, projenin yapılış etapları ve kullanılan yazılımlar detaylı bir şekilde anlatılmaktadır.

**3.1 Projenin Kapsamı ve Amacı**

Verilen projede programların her detayını ve verisini detaylıca anlatmam bankanın güvenlik kuralları nedeniyle yasak olduğundan dolayı programları sadece geniş çerçevede izah edebilirim. Bulunduğum ekip *Fraud* ekibi olduğundan dolayı, *Fraud*’u izleyen operasyon ekiplerine program yaparak onların rahatça izlemelerine olanak sağlamak bizim ekibin temel görevidir. Bu programlar dolandırıcılık için bazı kurallar(*Rule*) içeren programlardır. Ekip ben staja başladığımda eski bir *Fraud* izleme programından yeni bir *Fraud* izleme programına geçiş yapmak için iş planı oluşturmuştu. Benim bu plandaki yerim eski programın veri tabanındaki verilerin yeni tablodaki veriler cinsine çevrilerek yeni tabloya geçişini sağlamak.

**3.2 Scrum Toplantıları ve Projedeki Yeri**

Akbank proje gerçekleştirme metodu olarak scrum metadolojisini kullanıyor. Scrum’da işler mümkün olduğu kadar küçük parçalara bölünür ve bu işlere *Product Backlog item*(PBI) denir. Buradaki her iş süreci 2 hafta sürecek şekilde planlanır ve bu 2 haftalık iş yapma sürecine *Sprint* denir. Bir *Scrum* takımında bir *Product owner*, bir *Scrum Master* ve bir *Development* takımı yer almaktadır. *Product owner* ürünün iş listesini (*Product Backlog*) düzenleyip *Development* takımının gelecek *Sprint*’lerde neler yapacağını planlar. *Scrum Master*, *Scrum’*ın yönetilmesi ve iyi anlaşılmasından, aynı zamanda bir problem olduğunda herkese yardım etmekle sorumludur. *Development* takımı ise projenin geliştirilmesi, analizi ve test edilmesinden sorumludur. Bir *Sprint*’te *Sprint* planlama, *Daily Scrum*, *Sprint Review* ve *Sprint Retrospective* yer almaktadır. Biz bu 4 evreyi de verimli bir şekilde şirkette uygulayan bir takımız. *Sprint* planlama, *Sprint* ’in ilk gününde hangi PBI’ın alınıp alınmayacağını bütün takım birlikte konuşarak ve her PBI’ın boyutu oylanarak yapıldı. Aynı zamanda *Sprint*’te ulaşılması gerekli olan bir PBI, *Sprint Goal* olarak belirlendi. Herkes her PBI’ı fibonacci serisinden bir sayıyla oyladı ve ortak bir paydada buluşularak PBI’ların boyutları belirlendi. *Daily Scrum*, sabah 9.10’da takımın PBI’ların ne durumda olduğu, kimin neyi bitirdiği panoda 10-15 dk. boyunca üzerinden geçildi. *Sprint Review*, Sprint ‘in son gününde yapılan birim müdürünün de katıldığı bir *Sprint*’i değerlendirme sürecidir. *Sprint Retrospective*, takımın kendi içinde kendini bu Sprint’te neyi iyi neyi kötü yaptığını eleştirip sonraki *Sprint*’te bu kararların uygulamaya konulmasıdır. Ben de bir *Sprint*’te *Sprint Retrospective*’ini yönettim. *Retrospective* temasını buldum ve bunun takımca düzgün işlenmesini sağladım. Tema olarak *2 Truths and a Lie*(2 doğru ve 1 yalan) adlı birtema seçtim. Kısaca bu temada herkesin bir kağıda bu *Sprint*’te hoşlandığı, hoşlanmadığı veya tecrübe ettiği şeyler hakkında 2 doğru ve 1 yalan bir ifade yazmasını istedim. Herkes yazdıktan sonra teker teker herkes yazdığını okudu ve takım okuyan kişinin neyi doğru yazdığını neyi yanlış yazdığını bulmaya çalıştı. Olumsuz olan ve gelecek Sprint’te değiştirilmesini istenilen şeyler not edildi. Bu Retrospective temasındaki temel amaç takımdaki kişilerin neyi sorun edip neyi sorun etmeyeceğini diğer takım arkadaşlarınca değerlendirilmesinin yapılıp yapılamayacağının kontrolünü eğlenceli bir şekilde test etmekti. *Scrum* yönteminin iş hayatındaki ve projedeki yerini öğrenmek açısından bu stajda çok verimli tecrübeler kazandım.

**3.3 Kullanılan Tabloların Tanıtımı**

Staj projesinde yapacağım iş eski güvenlik ekranı bilgilerini yeni güvenlik kimlik kontrol doğrulama ekranına göre uyarlamak olduğu için Oracle SQL Developer ile birlikte eski ve yeni ekranın tablolarını anlamam gerekiyordu. Tablolara erişimim için gerekli *TNS* isimleri *Scrum Master* tarafından sağlatıldı. Oracle SQ L Developer, tabloları oluşturmak ve gerekli sorguların yapılması için çok gerekli ve kolay bir ara yüzü olan bir yazılım oldu benim için. Tabloları tanıtmak gerekirse, eski programda iki tablo olarak tutulan veriler yeni tabloda bir tablo olarak yer almaktadır. Eski 2 tablo SUBRULEDETAY ve SUBRULEMASTER tabloları olarak isimlendirilmiştir. SUBRULEMASTER eski programdaki kuralların verilerinin tutulduğu tablodur. SUBRULEDETAY ise eski programdaki kuralların detaylarının tutulduğu tablodur. Bu iki tablodaki veriler KK\_RULE denen yeni kimlik kontrol kurallarının bulunduğu tabloya aktarılacak şekilde düzenlenerek atılacağı için eski tablolardaki verilerin üzerinden geçmem ve anlamam gerekiyordu. Bu yüzden bana rehberlik yapan geliştirici ile tablolardaki verilerin özelliklerinin üzerinden geçtik. Teker teker verilerin üzerinden geçmek gerekirse SUBRULEMASTER’da yeni tablo için de anlam ifade eden veriler: RuleID, priority, explanation, aktif, timeinterval, ruletype, actdate, actdate1, actdate2, acttime1, acttime2, custtype, acıklama. RuleID her kuralın tekrar etmeyen bir değeri vardır ve bu tablo için *primary key* olarak belirlenmiş. Priority, yazılan her kuralın önceliğini belirtir. Explanation, kural adının açıklaması için kullanılmış. Aktif, kuralın aktif olup olmadığının tutulması için kullanılmış. Timeinterval, kuralın ne kadarlık bir aralıkta çalışacağının belirlenmesinde önemli bir yer tutar. Ruletype, kuralın tekrar edip etmeyeceğini belirlenmesinde tekrarlı ise 1 değilse 0 şeklinde yazılarak tutulur. Actdate, periyot ve tarih bilgisi için kullanılmıştır. Actdate1 ve acttime1, kuralın başlangıç tarihine karşılık gelirken, actdate2 ve acttime2 kuralın bitiş tarihine denk gelir. Custtype, iyi müşteri bilgisinin tutulduğu veridir. Açıklama, yazılan kuralla ilgili açıklamanın girildiği kısımdır. Burada KK\_RULE yani yeni tablo için gerekli olan verileri açıkladım. Diğer veriler yeni tabloda anlamsız olduğu için diğer sütun verilerini açıklamadım. SUBRULEDETAY tablosunda ise gerekli olan şu sütun verileri vardır: RuleID, oper\_type, amount, amntop, totamount, totamntop, count, cntop. RuleID, her kuralın bir değeri var fakat burada tekrar etmeyen bir şekilde değil. Oper\_type, operasyonların kimlik numaralarının tutulduğu veridir. Amount alt limit tutar bilgisini tutarken, amntop bu alt limitin operasyon bilgisinin tutulduğu sütunlardır. Totamount üst limit tutar bilgisini tutarken, totamntop bu üst limitin operasyon bilgisinin tutulduğu sütunlardır. Count, tekrarlı kural adedini tutarken, cntop bu işlemin operasyon bilgisini tutan sütunlardır.

KK\_RULE tablosunda id, operation\_id, description, active, priority, repetitive, repetitive\_oper\_control\_time, period, sublimit\_amount, sublimit\_amount\_op, total\_sublimit\_amount, total\_sublimit\_amount\_op, transaction\_count, transaction\_count\_op, good\_cust\_flag, suspicious\_branch\_flag, third\_person\_flag, note, created\_user, created\_date, rule\_begin\_date, rule\_end\_date, rule\_number sütunları yer almaktadır. Bu verilerin teker teker ne anlama geldiğini açıklamam bankanın güvenliği için sorun olacağından dolayı bu sütunların ne anlama geldiğini açıklamayacağım.

**3.4 JDBC ile veritabanına erişim ve Sınıfların oluşturulması**

Java dili kullanılarak Eclipse’e ojdbc6.jar eklentisi eklenerek tablolara erişim yapabileceğim veritabanına bağlantımı sağladım. JDBC bana veritabanındaki verilere sorgu yapabilme şansı tanıması açından çok yararlı bir eklenti oldu. Tüm proje boyunca 2 tane sınıf oluşturdum. Bu sınıflardan ilki olan JoinClass sınıfı, 2 eski tablodan birleştirilerek oluşturulan verilerin yeni tablodaki cinsten değişkenleri olan bir sınıftır. Diğer sınıf ise veritabanı bağlantısı ve gerekli işlem fonksiyonlarının bulunduğu ConOracle sınıfıdır. İki sınıf arasındaki bağ ise ConOracle’da veritabanından çekilen veriler JoinClass objelerinden oluşan bir vektöre atanır. Bu vektörü kullanarak gerekli birleştirme ve veri temizliği işlemleri yapılır. Objeleri vektöre atıp kullanma okulda öğrendiğim ve işte tecrübe ettiğim bir deneyim oldu.

**3.5 Eski tabloların birleştirilmesi ve veri temizlenmesi**

JDBC’yi kullanarak çağırılan sorgu ile SUBRULEMASTER ve SUBRULEDETAY yukarıda tabloların tanıtımında da belirttiğim yeni tablo için gerekli verilerin çekilmesi ve RuleID’leri eşit olma koşulu ile inner join işlemi yapıldı. Bu işlem sadece SUBRULEMASTER’ın aktif sütunun “A” olduğu yani aktif olduğu durumlarda yapıldı. Bu yeni oluşan tabloda oluşan veriler JoinClass objelerinden oluşan vektöre atandı. Atama yapılmadan önce bazı sütun verileri veri ön işlemesine maruz kalmak zorudaydı. Ruletype eski tablolarda 1 ve 2 olarak tekrarlı tekrarlı değil işlemini yaparken, yeni tabloda karşılığı olan repetitive in tekrarlı ve tekrarsız için karşılığı 1 ve 0 olarak değerlendirilmiş. Bu yüzden ruletype’ın değerlerinden bir çıkarılarak vektöre eklenildi. Created\_date için ise gerekli kütüphane kullanılarak o anki zaman neye karşılık geliyor ise o vektöre yazıldı.

**3.6 Veri temizlemesi yapılan verilerin yeni tabloya geçişi**

JDBC’nin sorgu özelliği kullanılarak vektörde bulunan JoinClass objelerinin verilerini teker teker KK\_RULE tablosuna ekledim. Bazı sütunlara ne yazılacağı belli olmadığı için rastgele bir değer ataması yaptım. Suspicious\_ branch\_flag, third\_person\_flag ve rule\_number farazi bir değer olan 1 değeri atadım. Böylelikle eski tablolardaki veriler temizlenerek yeni tablo olan KK\_RULE tablosuna geçiş yapmış oldu. Bu işlemler ekibin güncel olarak yaptığı bir işe katkı yapmamı sağladı.

**4. SONUÇ**

20 günlük olan bu stajda gerek kurumsal çalışma düzeni açısından gerekse de geliştiricilik tecrübesi kazanmak açısından pek çok deneyim kazanmama sebep oldu. Kurumsal iş hayatında hangi yazılım dillerinin ve IDE’lerin kullanıldığını gördüm. Örneğin ben Oracle SQL Developer’ı veritabanında tablo işlemleri yapmak, tablo görüntülemek ve SQL sorgularını kullanmak amacıyla kullandım. Takımın kalanında da Oracle SQL Developer’ın yanında TOAD tercih edildiğini de gözlemledim. Tablolardaki işlemleri yapmak için Java kullandım. Okul hayatımda pek Java kodu yazmamıştım. Bu Java sentaksına alışmamı sağladı. Java kodu yazmak için Eclipse IDE’sini kullandım fakat takımın kalanı Eclipse yerine IntelliJ IDE’sini tercih ediyor. IntelliJ IDE’sinin lisans sorunu nedeniyle yüklenilmediği için Eclipse kulllanmak zorunda kaldım. Kurumsal hayat açısından değerlendirirsem Akbank’ta Scrum metadolojisi kullanılması Software Engineering dersinde teorisini gördüğüm kavramların pratikte nasıl işlendiğini görmem açısından muazzam bir tecrübe oldu. Aynı zamanda bankada çalışmanın ve bazı şeyler geliştirmenin dezavantajları olduğunu da gözlemledim. Güvenlik öncelikli olduğu için yazılımların yüklenilmesi ve bazı erişimleri almam çok zamanımı aldı. Bu dezavantajlarına rağmen Akbank’ta çokca iş deneyimi kazandım.

**5.REFERANSLAR**

<https://www.infoq.com/news/2010/12/agile-scrum-retrospectives>

<https://www.scrumguides.org/scrum-guide.html>

[**http://www.tutorialspoint.com/java/index.htm**](http://www.tutorialspoint.com/java/index.htm)

[**http://www.tutorialspoint.com/jdbc/index.htm**](http://www.tutorialspoint.com/jdbc/index.htm)

[**https://download.oracle.com/oll/tutorials/SQLDeveloper/index.htm**](https://download.oracle.com/oll/tutorials/SQLDeveloper/index.htm)