

TÜBİTAK

2242- Lisans Öğrencileri Yazılım Projeleri Yarışması

Grup Adı

NHTS

Projenin Açık Adı

Nişan hatalarının bilgisayar yardımı ile tespitini sağlayan yazılımın geliştirilmesi

Proje Ekibi:

Öğrenci

Enver KESMEN

Akademik Danışman

Arş. Gör. Müh. Ütğm. Yavuz Selim ÖZZENGİN

İçindekiler

Özet	4
1.Giriş:	5
1.1 Problemin Tanımı	5
1.2 Amaç	5
1.3 Kapsam	5
2.İlişkili çalışmalar.....	5
3.Tasarım	6
3.1 Sistemde kullanılacak donanımlar.....	6
3.1.1 Silah	6
3.1.2 Silah sabitleme aparatı	6
3.1.3 Bilgisayar.....	8
3.2 Simülasyon adımları	8
3.2.1 Sistem düzeneğinin kurulması.....	8
3.2.2 Simülasyonun uygulanması	9
3.3 Simülasyon yazılımının genel özellikleri	11
3.4 Hocalık sistemi.....	11
3.5 Veri Tasarımı.....	11
3.5.1 E-R Diyagram	11
3.5.2 Veri tabanındaki tablolar	12
3.5.2.1 Kullanıcı Tablosu.....	12
3.5.2.2 Üçgen Tablosu	12
3.5.2.3 Hoca Tablosu	13
3.5.2.4 Yorum Tablosu.....	13
5.Gerçekleştirim	13
5.1 Veri Tabanı.....	13
5.2 Simülasyon yazılımı	13
5.2.1 Sistemde kullanılan önemli sınıflar.....	13
5.2.1.1 public static class Ekran	13
5.2.1.2 public class Nokta	14
5.2.1.3 public class Ucgen.....	14
5.2.1.4 public class Kullanici	16
5.2.1.5 public class VeriTabanıErisim	17
5.2.1.6 public static class SqlSorgusu	18
5.2.1.7 public abstract class BaglantiSatiri	18
5.2.2 Kullanıcı arayüzü.....	19

5.2.2.1 İlk Ekran	19
5.2.2.2 Giriş Ekranı	19
5.2.2.3 Kayıt Ekranı	19
5.2.2.4 Şifre Sıfırlama Ekranı	20
5.2.2.5 Kullanıcı girişi yapıldıktan sonra ulaşılabilen ekranlar(internet bağlantısı gerektir)	20
5.2.2.5.1 Karşılama Ekranı	20
5.2.2.6 Kullanıcı girişi yapılmadan ulaşılacak ekranlar (İnternet bağlantısı gerektirmez) ...	22
6.Proje takvimi.....	22
7.Değerlendirme.....	23
8.Sonuçlar	23
9.Projenin özgünlüğü.....	23
9.1 Yenilikçilik	23
9.2 Etki	23
9.3 Uygulanabilirlik	23
9.4 Kullanışlılık	23
10.Referanslar	23

Özet

Bu proje atıcının sadece silahı ve bilgisayarı yardımıyla herhangi başka bir donanıma veya personele ihtiyaç duymadan atıcıyı her defasında aynı şekilde ve doğru nişan almaya alıştırmayı, nişan almadaki becerilerini geliştirmeyi, varsa hatalarını tespit ederek gidermeyi amaçlamaktadır. Bilgisayar ekranına nişan aldıktan sonra sabitlenmiş bir silah ve geliştirilen yazılım sayesinde atıcı kendi nişan hatalarını görebilecektir. Atıcı ekrandaki hedefi fare yardımıyla nişan aldığı noktaya getirdikten sonra o noktayı işaretlemek için tıklar. Hedef daha sonra rastgele başka bir noktaya gider. Atıcı hedefi tekrar nişan aldığı noktaya getiri ve yine tıklar bu işlem 3 defa gerçekleştirilir. Programda geliştirilen algoritmalar işaretlenen 3 noktayı birbirine göre değerlendirerek nişanın hatalı veya hatasız olduğunu kullanıcıya bildirir.

Simülasyonu kullanmak için sadece bir bilgisayara ihtiyaç olduğu için herhangi bir yer istenilen her anda nişan hatası tespit dershanesine dönüştürülebilmektedir. Her ortamda kullanılabilir olması projenin en güçlü yanıdır. En küçük dağ karakolundan en büyük atış eğitimi dershanelerine kadar her ortamda ekstra herhangi bir harcama yapmadan uygulanabilir. Atıcı veya atıcının hocası simülasyon sonuçlarını internete bağlı herhangi bir bilgisayarda görüntüleyebilir. Öğrencisinin nişanı hakkında yorum yapabilir.

1.Giriş:

1.1 Problemin Tanımı

Atış temel teknikleri içerisinde en önemli faktör, doğru nişan almaktır. Nişan almayı bir atıcıya anlatmak oldukça kolaydır fakat atıcının doğru nişan almayı öğrenmesi zaman alabilir. Doğru nişan almayı öğrenmek ve refleks haline getirmek için çok fazla tekrar gerekir. Bu tekrarlar sırasında da atıcının nişanı doğru alması gerekir. Atıcının doğru nişan alıp almadığını tek başına anlayabileceği bir yöntem bulunmamaktadır. Doğru nişan aldığını anlaması için kendisi hariç en az bir personel yardımıyla farklı yöntemlerle nişanın kontrol edilmesi gerekmektedir. Bu yöntemlerin hata payı ise oldukça fazladır ve belli bir tekrardan sonra konsantrasyonun düşmesi sebebiyle bu hata payı daha da artmaktadır.

1.2 Amaç

Bu proje atıcının sadece silahı ve bilgisayarı yardımıyla herhangi başka bir donanıma veya personele ihtiyaç duymadan atıcıyı her defasında aynı şekilde ve doğru nişan almaya alıştırmayı, nişan almadaki becerilerini geliştirmeyi, varsa hatalarını tespit ederek gidermeyi amaçlamaktadır.

1.3 Kapsam

Bu proje Türk Silahlı Kuvvetleri'nde atış eğitimi yapan bütün birliklerde kullanılabilir. Ayrıca atış poligonları ve bireysel atış yapmak isteyen herkes bu programı bilgisayarına yükleyerek bulunduğu her ortamı küçük bir atış simülasyon dershanesine dönüştürebilir.

2.İlişkili çalışmalar

Piyasada çeşitli atış simülasyonları olsa da nişan hatalarına yönelik çıktı veren simülasyonlar sınırlı ve oldukça pahalı. Bu çalışmaların birçoğu da gizlilik derecesine sahip.

3.Tasarım

3.1 Sistemde kullanılacak donanımlar

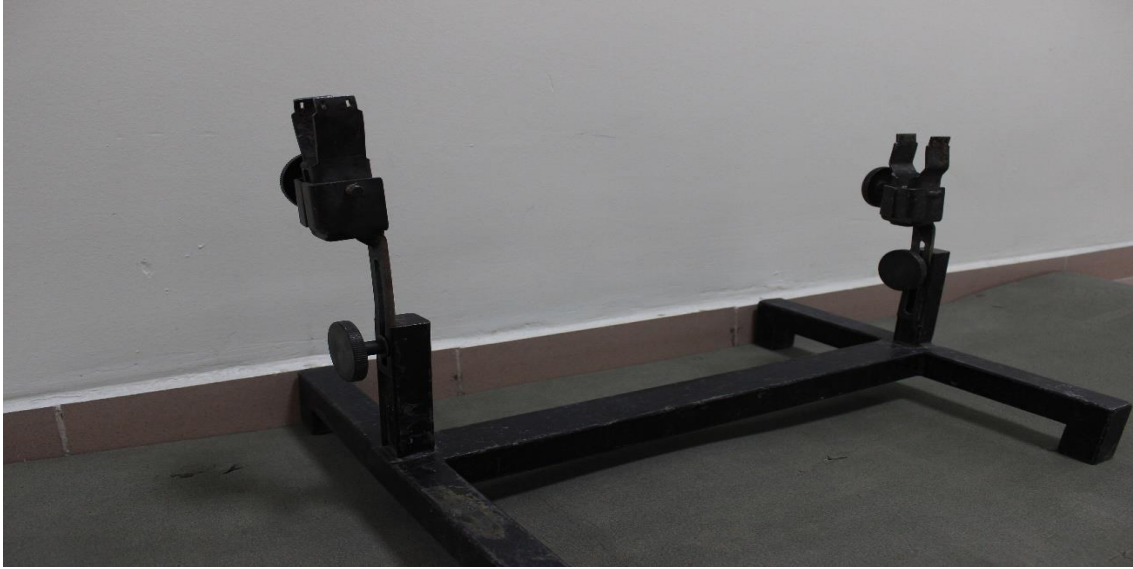
3.1.1 Silah

Proje boyunca gerçek silah kullanılmamıştır. Resimlerdeki silah değil bir Airsoft¹ oyuncaktır.



3.1.2 Silah sabitleme aparatı

Silahı sabitleyip nişan almak için kullandığımız aparat. Her türlü piyade tüfeği bu aparatla kolaylıkla sabitlenebilir ve yüksekliği ayarlanabilir.



Airsoft silahının sabitlenmiş hali.



3.1.3 Bilgisayar

Simülasyonu kullanmak için Windows işletim sistemi yüklü herhangi bir bilgisayar kullanılabilir.

3.2 Simülasyon adımları

3.2.1 Sistem düzeneğinin kurulması

Aparat yardımıyla sabitlenen silah bilgisayar ekranına nişan alacak şekilde 15 metre uzaklığa yerleştirilir. Sistemden doğru sonuçlar alınabilmesi için mesafenin tam 15 metre olması gerekir çünkü ekrandaki hedef 3x4 cm'dir. Hedefin boyutlarının 3x4 cm olması ve silahın 15 metrede olması hedefin 200 metredeki insan görünümünü vermesini sağlar.



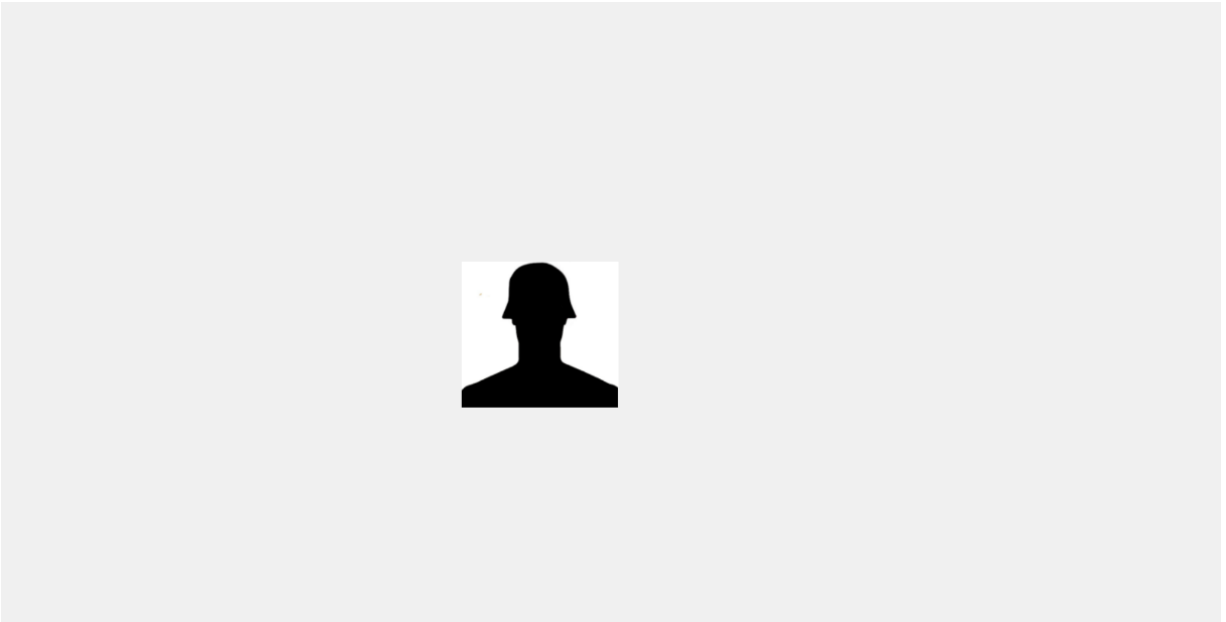
3.2.2 Simülasyonun uygulanması

Sistem düzeneği kurulduktan sonra atıcı silahın arkasına geçer. Ekranı nişan aldıktan sonra ekrandaki hedefi fare ile kontrol ederek nişan hattına getirir. Hedefin alt kenar ortasını nişan hattına getirdikten sonra fare ile tıklar.(Bu işlem raporun devamında işaretleme olarak adlandırılacak.). İşaretlemeden sonra bilgisayar, hedefi ekranın rasgele bir köşesine gönderir atıcı hedefi tekrar nişan hattına getirir ve işaretler. 3 işaretlemeden sonra programdaki algoritmalar atıcıya nişan hatası yapıp yapmadığını bildirir(Düzgün nişan şartı:

60 saniye içinde yapılan 3 işaretlemeden oluşturulan üçgenin en uzun kenarının 7mm'den küçük olması).Simülasyonun uygulanma anı.



Simülasyon ekranı



3.3 Simülasyon yazılımının genel özellikleri

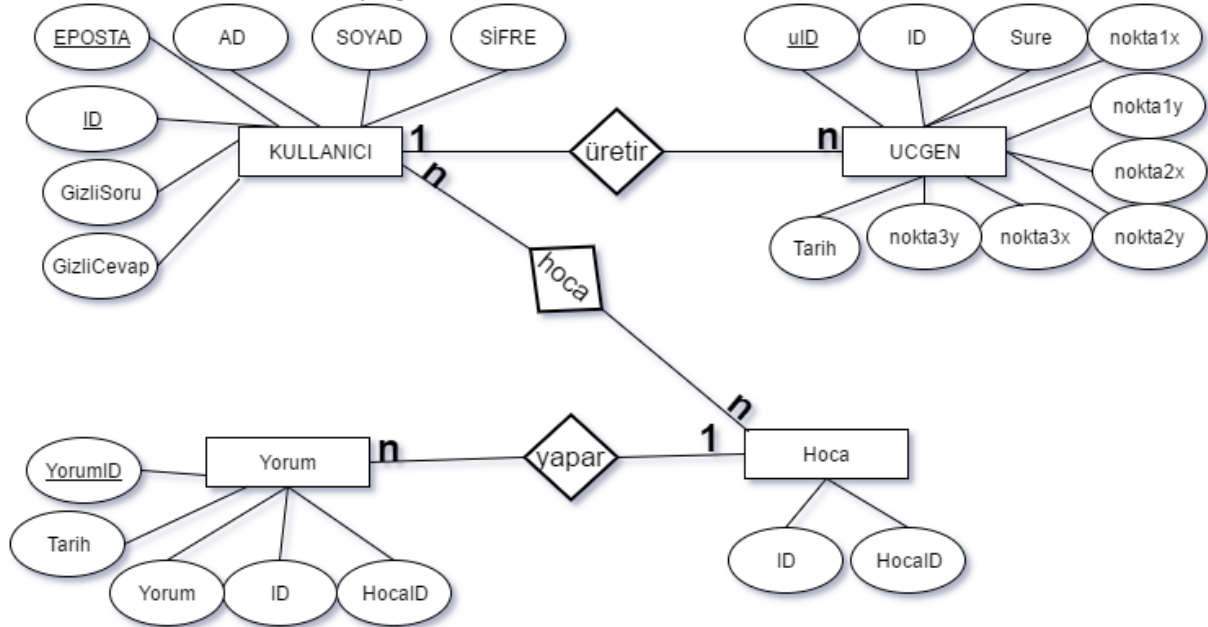
Sistem kurulduktan sonra atıcının yaptığı her üç nokta birbirine referans alınarak belli hesaplamalara tabi tutulacaktır. Her üç işaretlemeden bir üçgen oluşturulacak ve bu üçgendeki noktaların konumuna göre atıcının nişan alma yeteneği kıymetlendirilecektir. Eğer 60 saniye içerisinde yapılan 3 işaretlemeden en uzak 2 noktanın arası 7mm'den küçükse o nişan başarılı olarak kabul edilecektir. Burada ki problem her bilgisayarın ekran çözünürlüğü ve ekranın fiziksel boyutları birbirinden farklı olmasıdır. Bu farklılık bazı problemleri beraberinde getirecektir. Örneğin 1600x1200 ekran çözünürlüğünde yaptığınız bir simülasyon sonuçlarını 800x600 ekran çözünürlüğünde görüntülediğinizde üçgenlerin boyutu 2 katına çıkacaktır. Çok başarılı olan nişanlar bile bu sebeple başarısız olarak değerlendirilecek ya da atış poligonunda laptopta yaptığınız simülasyonun sonuçlarına evde masaüstü bilgisayarınızdan bakarken çözünürlük aynı olsa bile masaüstü bilgisayarınızın ekran boyutları daha fazla olduğu için üçgenler bu defa daha büyük görünecektir. Bu problemi aşmak için piksel üzerinden değil mm üzerinden yürüyen algoritmalar geliştirilmesi planlanmıştır. Sistem yazılımı bu hesaplamaları yapacaktır.

3.4 Hocalık sistemi

Sisteme eklediğimiz bu özellik sayesinde kullanıcılar e-posta girerek kendilerine hoca seçebilecektir. Bir kullanıcı hem atıcı hem de hoca olabilecektir. Hocalar öğrencilerinin nişan simülasyon sonuçlarını görebilecek, onların nişanları hakkında yorumlarını öğrencilerine iletebilecektir ve öğrenci sisteme giriş yaptığı anda hocasının onun hakkındaki görüşlerini görebilecektir. Bu özelliğin atıcının motivasyonunu arttırdığı gibi nişan hatalarını düzeltmesine de yardımcı olacağı değerlendirilmektedir.

3.5 Veri Tasarımı

3.5.1 E-R Diyagram



3.5.2 Veri tabanındaki tablolar

3.5.2.1 Kullanıcı Tablosu

Sütun Adı	Tür	Tanım
Eposta	VARCHAR(100)	E-posta adresi.
ID	INT(PRIMARY KEY)	ID numarası
Ad	VARCHAR(30)	İsim bilgisi
Soyad	VARCHAR(50)	Soyisim bilgisi
Sifre	VARCHAR(255)	Kullanıcının şifresi
GizliSoru	VARCHAR(255)	Seçtiği gizli soru
GizliCevap	VARCHAR(255)	Soruya verdiği cevap

Bu tabloda sistemi kullanan kullanıcıların bilgileri tutulmaktadır. Üye olurken bu bilgilerin tamamı doldurulmak zorundadır. Veri tabanında ID sütunu 1'den başladığı ve her kayıta otomatik olarak 1 arttığı için aynı zamanda kullanıcının sisteme üye olan kaçınıcı kişi olduğunu da göstermektedir. Şifre ve gizli cevap veri tabanına yalın haliyle değil hash ve salt işleminden geçtikten sonra kaydedilmektedir. Salt işlemi için şifrenin ve gizli cevabın sonuna e-posta adresi eklenmiştir. Daha sonra hash işlemi için *SHA-256* algoritması tercih edilmiştir.

3.5.2.2 Üçgen Tablosu

Sütun Adı	Tür	Tanım
uID	INT	Üçgen ID
ID	INT	Üçgeni oluşturan kullanıcının ID'si
Sure	INT	Üçgenin oluşturulma süresi
nokta1x	Float	Üçgenin ilk noktasının ekranın sol kenarına mm cinsinden uzaklığı
nokta1y	Float	Üçgenin ilk noktasının ekranın üst kenarına mm cinsinden uzaklığı
nokta2x	Float	Üçgenin ikinci noktasının ekranın sol kenarına mm cinsinden uzaklığı
nokta2y	Float	Üçgenin ikinci noktasının ekranın üst kenarına mm cinsinden uzaklığı
nokta3x	Float	Üçgenin üçüncü noktasının ekranın sol kenarına mm cinsinden uzaklığı
nokta3y	Float	Üçgenin üçüncü noktasının ekranın üst kenarına mm cinsinden uzaklığı
Tarih	DATETIME	Üçgenin oluşturulduğu tarih.

Bu tabloda simülasyon sırasında oluşturulan üçgenlerin bilgileri tutulmaktadır. Üçgenlerin koordinatları mm cinsinden tutulmaktadır. Üçgenin oluşturulma süresi ise saniye cinsindendir. Üçgen ile tutulabilecek minimum veriler tutulmuştur en uzun kenar atışın hatalı olup olmadığı gibi veriler tekrar programda hesaplanmaktadır.

3.5.2.3 Hoca Tablosu

Sütun Adı	Tür	Tanım
ID	INT	Kullanıcı ID'si
HocaID	INT	Kullanıcının hocası olan kullanıcının ID'si

Bu tabloda atıcı ve atıcıların hocalarının bilgileri tutulmaktadır.

3.5.2.4 Yorum Tablosu

Sütun Adı	Tür	Tanım
YorumID	INT	Yapılan yorumun ID'si
Tarih	DATETIME	Yorumun oluşturulduğu tarih
Yorum	VARCHAR(510)	Yapılan yorum
ID	INT	Yorum yapılan kullanıcının ID'si
HocaID	INT	Yorumu yapan hocanın ID'si

Bu tabloda yorumlar tutulmaktadır.

5. Gerçekleştirim

Sistemin gerçekleştirilmesi için C# dili ve MSSQL veri tabanı tercih edilmiştir.

5.1 Veri Tabanı

Projede Microsoft Azure SQL Veri tabanı kullanılmış olup bütün veriler nhts.database.windows.net isimli serverda nhts adlı veri tabanında tutulmaktadır. Veri tabanının konumu Kuzey Avrupa'dır. Veri tabanındaki bütün tablolar tasarım bölümünde verilen bilgilere göre oluşturulmuştur.

5.2 Simülasyon yazılımı

5.2.1 Sistemde kullanılan önemli sınıflar

5.2.1.1 public static class Ekran

Static bir sınıf olan Ekran sınıfında ekran ölçülerini mm-piksel dönüşümünü yapacak sabitler hesaplandı.

Erişim belirteci	Değişken Türü	Değişken Adı	Açıklama
public (static)	double	YksklkMM	Ekran yüksekliği mm cinsinden
public (static)	double	GnslkMM	Ekran genişliği mm cinsinden
public (static)	double	YksklkPX	Ekran yüksekliği piksel cinsinden
public (static)	double	GnslkPX	Ekran genişliği piksel cinsinden
public (static)	double	PxtoMM_genislik	Yatay ekseninde pikselden mm'ye çevirme sabiti
public (static)	double	PxtoMM_yukseklk	Dikey ekseninde pikselden mm'ye çevirme sabiti
public (static)	double	MmtoPX_genislik	Yatay ekseninde mm'den piksele çevirme sabiti

public (static)	double	MmtoPX_yukseklık	Dikey eksende mm'den piksele çevirme sabiti
-----------------	--------	------------------	---

Aşağıdaki metod giriş ekranında ekran ölçüleri girildikten sonra çalıştırılıyor.

```

public static void sabitleriHesapla(double ysklkCM, double gnslkCM)
{
    YsklkMM = ysklkCM * 10;
    GnslkMM = gnslkCM * 10;
    GnslkPX = SystemInformation.VirtualScreen.Width;
    YsklkPX = SystemInformation.VirtualScreen.Height;
    MmtoPX_genislik = GnslkPX / GnslkMM;
    MmtoPX_yukseklık = YsklkPX / YsklkMM;
    PxtoMM_genislik = 1 / MmtoPX_genislik;
    PxtoMM_yukseklık = 1 / MmtoPX_yukseklık;
}

```

5.2.1.2 public class Nokta

koorX ve koorY olmak üzere 2 değişkeni var. Atıcı her işaretleme yaptığında hedefin koordinatlarını tutmak için oluşturulan sınıftır. Sınıfın yapıcı metodu:

```

public Nokta(int koorX, int koorY)
{
    this.koorX = koorX;
    this.koorY = koorY;
}

```

5.2.1.3 public class Ucgen

Üçgen sınıfı bu projedeki en önemli sınıftır. Nişan hatalarının tespiti bu üçgenler üzerinden yapılmaktadır. Bu sınıftaki önemli metotlardan birisi UcgenCiz metotodudur.

DEĞİŞKENLER:

Erişim belirteci	Değişken Türü	Değişken Adı	Açıklama
public	Nokta	N1	Üçgenin ilk oluşturulan noktası
public	Nokta	N2	Üçgenin ikinci oluşturulan noktası
public	Nokta	N3	Üçgenin üçüncü oluşturulan noktası
public	DateTime	Tarih	Üçgenin oluşturulduğu tarih
public	Double	Kenar1	N1 ile N2 arasındaki piksel cinsinden uzaklık
public	Double	Kenar2	N1 ile N3 arasındaki piksel cinsinden uzaklık
public	Double	Kenar3	N2 ile N3 arasındaki piksel cinsinden uzaklık
public (static)	int	SagSınırPX	Üçgenler çizilirken en son çizilen üçgenin piksel cinsinden sağ noktası
public (static)	İnt	AltSınırPX	Üçgenler çizilirken en son çizilen üçgenin piksel cinsinden alt noktası
Public (static)	İnt	UstSınırPX	Üçgenler çizilirken çizilebilecek piksel cinsinden en üst sınır

Public (static)	int	basariStandarti	Üçgenin başarılı sayılabilmesi için en uzak 2 noktasının alabileceği mm cinsinde maksimum değer
public	int	Sure	Üçgenin saniye cinsinden oluşturulma süresi

METOTLAR:

Erişim belirteci	Metot adı	Parametreler	Döndürdüğü değer	Açıklama
Public static	EnSagNoktaBul	Nokta[] noktalar	Nokta	Metoda gönderilen noktalardan ekranın en sağından olanı tespit eder.
Public static	EnAltNoktaBul	Nokta[] noktalar	Nokta	Metoda gönderilen noktalardan ekranın en altında olanı tespit eder.
Private	SolUsteTasi	Nokta[] noktalar	Nokta[]	Metoda gönderilen 3 noktayı ekrandan taşmayacak ve aralarındaki mesafe değişmeyecek şekilde ekranın sol üstüne taşır ve geriye noktaların yeni koordinatlarını döndürür.
private	EnUzunKenarBul	double k1, double k2, double k3	double	Geriye üçgenin en uzun kenarını döndürür.
public	BasariliMi	double BasariStandartMM	İnt	Yapılan simülasyonda nişan hatası olup olmadığınıın tespiti yapar.
public	UcgenCiz	int lblSagNokta	Nokta[]	Parametre olarak simülasyon sonuç ekranında en son oluşturulan label'ın en sağ noktası gönderilir. Metot bu noktaya göre çizilecek üçgenin köşe koordinatlarını döndürür.

YAPICI:

```
public Ucgen(Nokta[] nokta, int sure)
{
    //Üçgenlerin kenar uzunlukları Pixel cinsinden hesaplandı,(2 nokta arası
    uzaklık formülü)
    Kenar1 = Math.Sqrt(Math.Pow((nokta[0].KoorX - nokta[1].KoorX), 2) +
    Math.Pow((nokta[0].KoorY - nokta[1].KoorY), 2));
    Kenar2 = Math.Sqrt(Math.Pow((nokta[0].KoorX - nokta[2].KoorX), 2) +
    Math.Pow((nokta[0].KoorY - nokta[2].KoorY), 2));
    Kenar3 = Math.Sqrt(Math.Pow((nokta[1].KoorX - nokta[2].KoorX), 2) +
    Math.Pow((nokta[1].KoorY - nokta[2].KoorY), 2));
    //bu üçgenin en uzun kenarı mm cinsinden tespit ediyor.
    EnUzunKenarMM = EnUzunKenarBul(Kenar1, Kenar2, Kenar3);
    //Üçgeni ekranın sol üst köşesine taşıyor.
    nokta = SolUsteTasi(nokta);
    N1 = nokta[0];
    N2 = nokta[1];
    N3 = nokta[2];
    //Üçgenin en sağdaki noktası ve en alttaki noktasının x ve y koordinatları
    tespit ediyor
    //Simülasyon sonuç ekranında üçgenleri çizmek için gerekli.
    EnSagNokta = EnSagNoktaBul(nokta).KoorX;
    EnAltNokta = EnAltNoktaBul(nokta);
    Sure = sure;
    //Nişan hatası olup olmadığı tespit ediyor.
    Basari = BasariliMi(basariStandarti);
}
```

Bütün üçgenleri sol üste taşıyarak oluşturduktan sonra simülasyon sonuç ekranında üçgenleri yan yana çizmek için bir metoda daha ihtiyaç vardır. Bu işlemi UcgenCiz metodu sağlayacaktır. Bu sınıfta static int olarak SagSınırPX, AltSınırPX, UstSınırPX değişkenleri olacaktır ve bu değişkenler simülasyon sonuç ekranında çizilen son üçgene göre şekillenecektir. Örneğin son çizilen üçgenin en sağdaki noktası artık SagSınırPX'dir ve yeni çizilecek üçgenin en solu bu sınırdan olmalıdır ki üçgenler üst üste gelmesin. UcgenCiz metodu bu hesaplamaları yaparak geriye çizilecek 3 Nokta döndürecek ve form ekranında bu noktalardan bir üçgen çizilecektir.

3.2.1.5 public class Kullanici

Kullanıcının bilgilerinin tutulduğu sınıf ayrıca hash ve salt işlemleri de bu sınıfta yapılıyor.

DEĞİŞKENLER:

Erişim belirteci	Değişken Türü	Değişken Adı	Açıklama
public	İnt	ID	Kullanıcı ID'si
public	String	Ad	Kullanıcının ismi
public	String	Soyad	Kullanıcının soyadı
public	String	Sifre	Kullanıcının şifresi
public	String	Mail	Kullanıcının E-posta adresi

Sifre oluştur metodu bu sınıftadır. Statik bir sınıftır parametre olarak kullanıcının girdiği şifreyi ve epostayı alır, şifrenin sonuna epostayı ekler(salt işlemi) ve SHA-256 algoritması ile hash işlemini gerçekleştirir şifre veri tabanında bu şekilde tutulmaktadır. Aynı

işlem gizli cevap içinde yapılmaktadır. Gizli cevabın sonuna eposta eklenir ve hash işlemi gerçekleştirir.

2.5.1.4 public class VeriTabanıErisim

Uygulamadaki bütün veri tabanı erişimleri bu sınıf üzerinden yürümektedir.

METOTLAR:

Erişim belirteci	Metot adı	Parametreler	Döndürdüğü değer	Açıklama
public	SifreDegistir	string eposta, string sifre	int	Metoda gönderilen eposta adresini gönderilen şifre ile değiştirir
public	EpostaUygunMu	string eposta	int	Metoda gönderilen eposta veri tabanında kayıtlı mı diye kontrol eder
public	KullaniciKayit	Kullanici kul, string GizliSoru, string GizliCevap	Boolean	Metoda gönderilen kullanıcıyı sisteme kayıt eder
public	KullaniciGiris	string eposta, string sifre	Kullanici	Metoda gönderilen kullanıcı adı ve şifre ile eşleşen kullanıcı bilgilerini geriye kullanıcı nesnesi olarak döndürür
public	HocaKayit	string eposta	int	Metoda gönderilen epostayı oturum açmış olan kullanıcının hocası olarak sisteme ekler
public	HocaSil	int ID, int HocaID	Boolean	Metoda gönderilen HocaID'yi oturum açmış olan kullanıcının hocaları arasından çıkarır
public	HocaListele	int ID	List<string>	Metoda ID'si gönderilen kullanıcının hocalarının ad,soyad ve ID'lerini liste olarak geri döndürür
public	OgrenciListele	int ID	List<string>	Metoda ID'si gönderilen kullanıcının

				öğrencilerini ad,soyad ve ID'lerini liste olarak geri döndürür
public	UcgenKayıt	Ucgen u	Boolean	Metoda gönderilen üçgenin uzunluklarını mm'ye çevirerek oturum açmış olan kullanıcın ID'si ile birlikte veri tabanına kaydeder.
public	KayitliUcgenleriG etir	int secim, int ID	List<Ucgen>	Metoda ID'si gönderilen kullanıcın veri tabanına kayıtlı olan üçgenlerini liste olarak geri döndürür. Secim 0 için bütün üçgenleri, 1 için başarılı üçgenleri, 2 için başarısız üçgenleri seçer. Veri tabanında mm olarak çekilen üçgenler geriye döndürülmeden önce piksel cinsine çevrilir ve geri döndürülür.
public	YorumYaz	string yorum,int ID,int HocalID	Boolean	Hocanın yaptığı yorumu veri tabanına kaydeder
public	Yorumlar	int ID	List<string>	Metoda gönderilen ID ile ilgili yapılmış bütün yorumları liste olarak geriye döndürür.

2.5.1.4 public static class SqlSorgusu

Programda kullanılacak bütün SQL sorgularının tutulduğu sınıf.

2.5.1.5 public abstract class BaglantiSatiri

Veritabanı bağlantı satırının App.config'den çağrıldığı sınıftır.

5.2.2 Kullanıcı arayüzü

5.2.2.1 İlk Ekran

Programın ilk açılışında ekran bilgilerinin girildiği ekrandır.

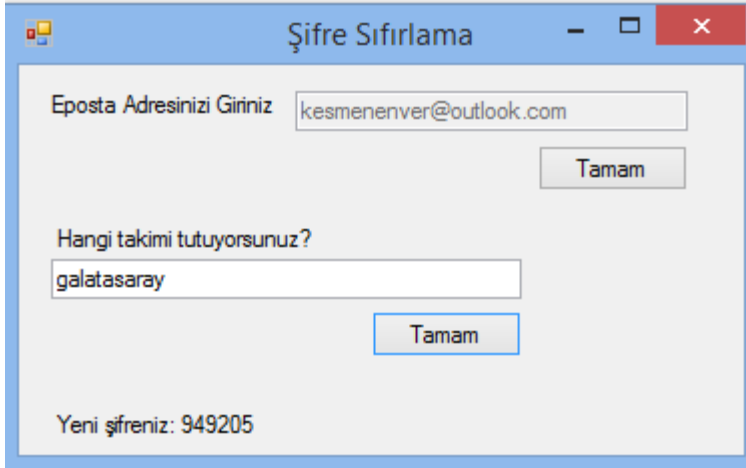
5.2.2.2 Giriş Ekranı

İlk ekrandan sonra açılan ekran. Kullanıcı giriş yapabilir yada simülasyonu offline olarak kullanmayı tercih edebilir.

5.2.2.3 Kayıt Ekranı

Simülasyona gerekli bilgileri doldurarak üye olunan ekran.

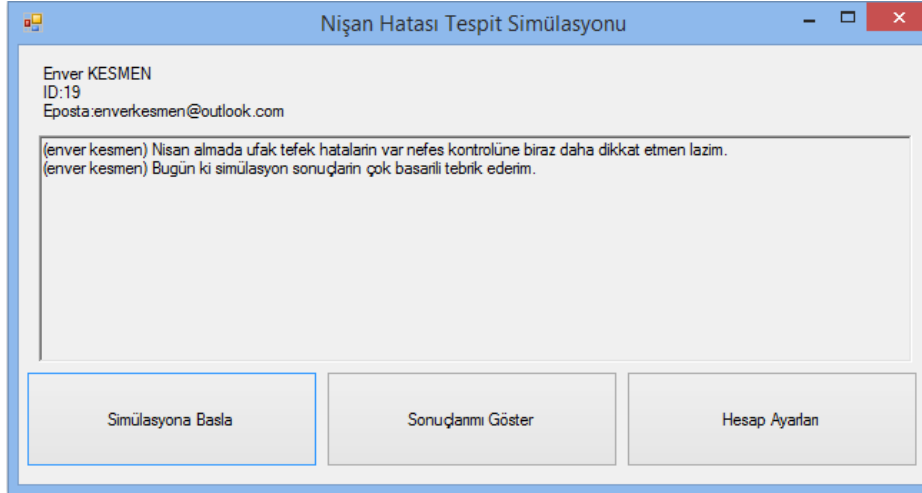
5.2.2.4 Şifre Sıfırlama Ekranı



Kullanıcının şifresini sıfırlaması için tasarlanan ekran. Mail ve gizli cevap girildikten sonra kullanıcıya yeni bir şifre verecek.

5.2.2.5 Kullanıcı girişi yapıldıktan sonra ulaşılabilen ekranlar(internet bağlantısı gerektir)

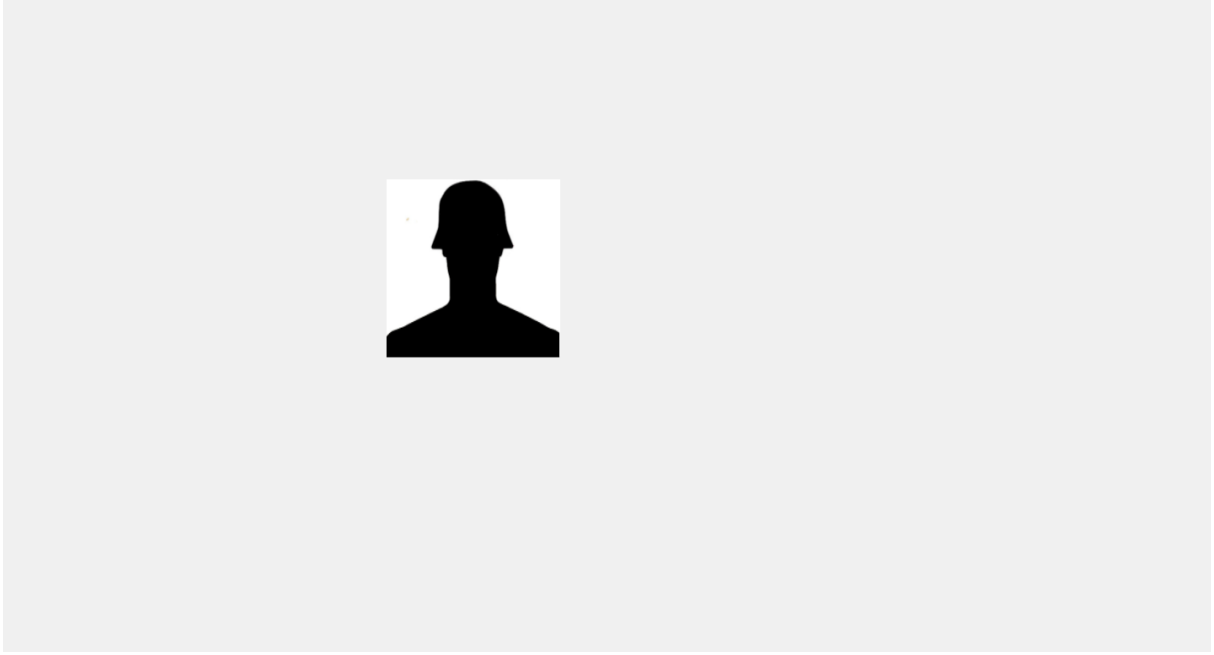
5.2.2.5.1 Karşılama Ekranı



Giriş yapıldıktan sonra kullanıcıyı karşılayan ekran bu ekranda kullanıcı kendisi hakkında hocalarının yaptığı yorumları görebilir, simülasyon ekranına gidebilir, hesap ayarlarına girebilir, Kendisinin veya öğrencilerinin

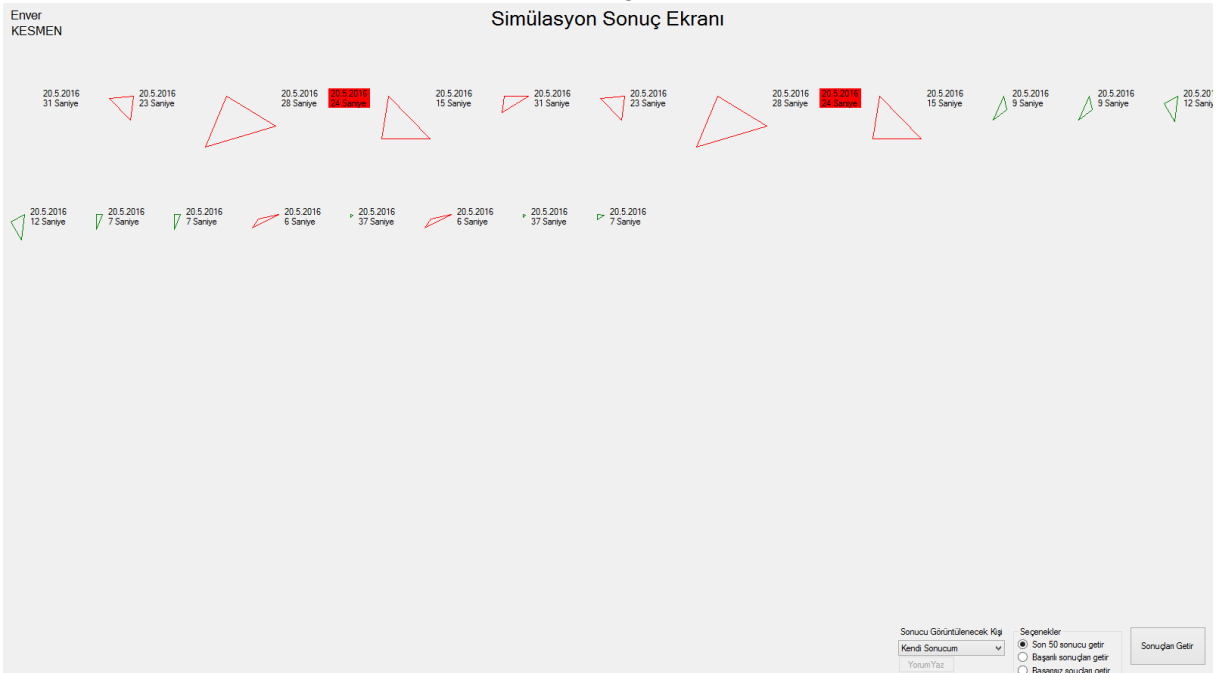
sonuçlarını görmek için sonuç sayfasına gidebilir.

5.2.2.5.2 Simülasyon Ekranı



Simülasyonun yapıldığı ekrandır. Ekrandaki hedef Mouse ile hareket ettirilerek nişan noktasına getirilir ve işaretleme yapılır. İşaretlemeden sonra hedef ekran üzerinde rasgele başka bir noktaya hareket eder.

5.2.2.5.3 Simülasyon Sonuç Ekranı

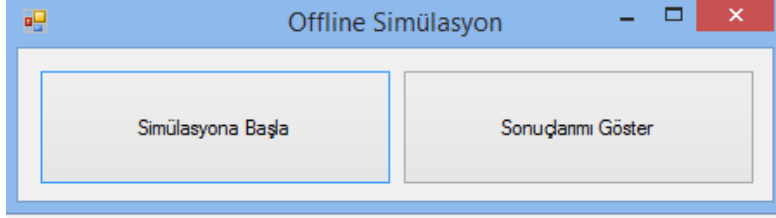


Simülasyon sonuçlarının gösterildiği ekrandır. Kullanıcı kendi sonuçlarını veya öğrencilerinden istediğinin sonuçlarını görebilir. Yeşil olan üçgenler o işaretleme de doğru nişan alındığını gösterir kırmızılar ise nişan hatası yapıldığını gösterir. Arka planı kırmızı olan yazılar ise o işaretlemede oluşturulan üçgenin 4 cm'den büyük olduğunu gösterir. Bu üçgenler ekranda çok fazla yer kaplayacağı için ekrana çizilmemiştir.

5.2.2.6 Kullanıcı girişi yapılmadan ulaşılabilecek ekranlar (İnternet bağlantısı gerektirmez)

Program internet olmadan da nişan hatalarının tespitini yapabilmektedir. Ancak burada yapılan simülasyon sonuçları geçici bellekte tutulmaktadır ve program kapatıldıktan sonra kaybolmaktadır.

5.2.2.6.1 Offline Simülasyon Karşılama Ekranı



Ekranda 2 buton vardır. İlk buton simülasyonu başlatır. İkinci buton ise yapılan simülasyonun sonuçlarını gösterir.

5.2.2.6.2 Offline Simülasyon Ekranı

Yukarıdaki simülasyon ekranı ile birebir aynıdır. Yapılan üçgenler veri tabanı yerine `public static List<Ucgen> offlineUcgenListesi` listesinde tutulmaktadır.

5.2.2.6.3 Offline Simülasyon Sonuç Ekranı

Yukarıdaki simülasyon sonuç ekranı ile birebir aynıdır. `public static List<Ucgen> offlineUcgenListesi` listesinde tutulan üçgenler analiz edilerek ekrana çizilir.

6.Proje takvimi

İş paketleri	Aralık 2015	Ocak 2016	Şubat 2016	Mart 2016	Nisan 2016	Mayıs 2016
İş Paketi 1						
İş Paketi 2						
İş Paketi 3						
İş Paketi 4						
İş Paketi 5						
İş Paketi 6						
İş Paketi 7						

İş Paketleri	İş Paketinde Yapılacaklar
İş Paketi 1	Proje konusuyla ilgili daha önceden yapılmış olan çalışmaların araştırılıp incelenmesi.
İş Paketi 2	KHO atış hocalarıyla birlikte fikrin değerlendirilmesi.
İş Paketi 3	Simülasyon yazılımının yazılması
İş Paketi 4	Simülasyonun test edilmesi
İş Paketi 5	Test sonuçlarının KHO atış hocalarıyla değerlendirilmesi
İş Paketi 6	Yazılımda gerekli düzenlemelerin yapılması ve yeniden test edilmesi
İş Paketi 7	Proje sonuç raporunun hazırlanması

7.Değerlendirme

Proje sonucunda ortaya çalışan ve doğru tespitler yapan bir simülasyon yazılımı çıktı. Nişan hatalarını tespitteki başarısı tahminlerimizin çok üzerinde oldu. Bu yazılımın bütün askeri birlikler ve atış simülasyonlarınca kullanılabileceğini değerlendiriyoruz. Projeyi gerçekleştirirken atıcılık uzmanlık alanımız olduğu için herhangi bir zorluk çekmedik. 10'dan fazla subay ve atış hocasından fikir aldık. Ancak raporları hazırlarken kaynakların tamamına yakınının gizlilik derecesine sahip olmasından dolayı ciddi zorluklar çektik.

8.Sonuçlar

Nişan Hatası Tespit Simülasyonu'nu 200'den fazla askeri öğrenci ile birlikte test ettik ve gördük ki klasik yöntemlere göre çok daha başarılı. Klasik yöntemlerden farklı olarak personel sayısını yarıya düşürüyor, rapor hazırlıyor ve kesin sonuçlar veriyor.

9.Projenin özgünlüğü

9.1 Yenilikçilik

Proje yıllardır klasik yöntemlerle yapılmaya çalışılan, hata payı çok fazla olan ve herhangi bir veri tutulamayan yöntemleri birleştirerek bilgisayar ortamına taşımıştır. Nişan hatalarını tespitini bu kadar az maliyetle ve bu kadar kesin sonuçlarla tespit eden herhangi bir çalışma bulunmamaktadır. Diğer simülasyonların aksine bu projede pahalı sensörler veya kapalı bir ortama gerek yoktur.

9.2 Etki

Temel atış eğitiminin en önemli kısmı doğru nişan almaktır. Atıcı doğru nişan alıp almadığını kendisi fark edemez. Nişan hatası başka bir personel yardımıyla tespit edilmeye çalışılsa da hata payı oldukça yüksektir. Bu problemlere getirilecek bir çözüme atış eğitimi veren kurumların ilgi göstereceği anlamına gelmektedir.

9.3 Uygulanabilirlik

Bu projenin en güçlü yanı uygulanabilirlik kısmıdır. Projeyi hayata geçirmek ve bulunduğunuz ortamı bir atış simülasyon dershanesine çevirmek için silahınıza ek olarak sadece bir bilgisayara ihtiyacınız vardır. İster evde olun ister kışla da isterseniz de mevziide fark etmez sadece bir bilgisayarla doğru nişan alma eğitimi yapabilir ve nişan hatalarınızı düzeltebilirsiniz. Atış hocasıysanız öğrencilerinizin simülasyon sonuçlarını internet bağlantısı olan herhangi bir bilgisayardan görebilir ve öğrencilerinize tavsiyelerde bulunabilirsiniz.

9.4 Kullanışlılık

Nişan hatalarının tespitine çok ucuz ve etkili bir çözüm sağlayacağı için Türk Silahlı Kuvvetleri'ne hizmette büyük bir rol alacağına inanıyorum. Proje her şarta ve ortama uygun şekilde tasarlanmıştır. Silahı olan herkes bilgisayarına 1mb'lık bu yazılımı yükleyerek bulunduğunuz ortamı bir atış simülasyonuna çevirebilir.

10.Referanslar

¹ Billings, Zach. (2011, 20 Ekim) "What is Airsoft" Erişim Tarihi: 20 Mayıs 2016,<http://zachbillings.com/what-is-airsoft/>