Gorgon Oyun Motorunu Visual Studio ile kullanmak için, lütfen aşağıdaki adımları takip ediniz. Bu Doküman Visual Studio 2012 Ultimate sürümüne göre yazılmıştır.

1. Projenin Oluşturulması

Visual Studio'da proje oluşturmak için, File > New > Project'i seçiniz. Ardından, Visual C++ > Win32'i seçiniz ve Create directory for solution seçeneğindeki işareti kaldırınız. Projeye bir isim vererek Ok tuşuna basınız. Next tuşuna bastıkdan sonra karşınıza çıkan ekranda, Application type altında Windows Application seçili olmalı. Application options altındaki Empty project'i seçip, diğer seçili öğelerin işaretini kaldırınız. Finish tuşuna tıklayarak proje oluşturma sürecini sonlandırınız. Son olarak, projeye kaynak dosyası eklemek üzere Solution Explore'dan projenize sağ tıklayıp; Add > New Item ile kaynak dosyası—C++ File ekleyiniz.

2. Gorgon Oyun Motoru Kurulumu

Oluşturulmuş bir Win32 projesinde, Gorgon Oyun Motorunun kullanılabilmesi için gerekli kütüphane (lib) ve başlık (header—.h) dosyalarının eklenmesi ile birlikte bir takım varsayılan ayarların değiştirilmesi gerekmektedir.

2.1 Başlık Dosyalarının ve Kütüphanenin Eklenmesi

Visual Studio, Debug (hata ayıklama/test modu) ve Release (yayınlama modu) altındaki ayarları farklı tutar. Bu noktada, başlık dosyaları her iki modda aynı; kütüphane dosyası, Debug ve Release modları için farklıdır. Başlık dosyalarını eklemek için öncelikle projenize sağ tıklayıp Properties'i seçiniz. Ardından, sol üstte yer alan Configuration listesinden, All Configurations'ı seçiniz. Daha sonra, sol menüden Configuration Properties > VC++ Directories > Include Directories'i değiştirmek üzere açılan listeden Edit'e tıklayınız. Buraya, Gorgon Oyun Motorunun dosyalarının bulunduğu dizini giriniz. Örneğin, masaüstünde, sizin projenizi ve Gorgon Oyun Motorunu temsil eden iki klasör—Proje1 ve GOM olsun. Bu örnekte, başlık dosyalarının doğru dosya yolu aşağıdaki gibi olmalıdır.

```
../GOM/
```

Windows ortamında “..” bir üst klasörü temsil eder. Benzer bir şekilde, “.” o anki klasörü temsil eder. Kütüphane dosyasının eklemek için, öncelikle bir Configuration listesinden Debug'ı seçiniz. Daha sonra, Linker > Input > Additional Dependencies'e gelip Edit'e tıklayınız. Az önce verilen klasör hiyerarşisini baz alırsak, kütüphane dosyası için doğru dosya yolu aşağıdaki gibi olmalıdır.

```
../GOM/GGE_d.lib
```

Aynı işlemi Release modu için uygulaynız. Fakat, kütüphane dosyasının isminden “_d”

kaldırılmalıdır. Örneğin,

../GOM/GGE.lib

2.2 Diğer Ayarlar

Configuration Properties > General altında bulunan birtakım ayaların değiştirilmesi gerekmektedir. Bunun için, öncelikle Configuration'ı All Configurations olarak seçiniz. Daha sonra aşağıdaki ayarları takip ediniz.

Use of MFC : Use MFC ın a Static Library

Character Set : Not Set

3. Örnek Program Kodu

Programı çalıştırmadan önce, UI.wgt dosyasını, proje klasörünüze kopyalayınız.

```
// InitializeApplication fonksiyonunun tanımlandığı başlık dosyası
#include <Widgets/Main.h>
#include <Widgets/Button.h> // Button sınıfının bulunduğu başlık dosyası

#include <vector>
#include <string>


int Application(std::vector<std::string> &arguments) {
  // parametreler sırası ile:
  // sistem ismi, pencere ismi, genişlik, yükseklik
  gge::widgets::InitializeApplication("test-program", "Test Program", 200, 200);


  gge::widgets::Button button;
  button.SetContainer(gge::widgets::TopLevel); // düğmeyi en üst katmana ekler
  button.Text = "Tikla"; // düğmenin üstünde görünecek yazı

  // tıklama olayı gerçekleştiğinde yapılacak işlemi kaydet
  button.ClickEvent.RegisterLambda([] {
    gge::os::DisplayMessage("Mesaj", "Merhaba, Dünya!");
  });

  // pencereyi açık tut
  gge::Main.Run();
  return 0;
}
```