1: githubot belöttem egy local mappára

2: leötltöttem MonoGame-t VisualStudios verziót

2.2: WP8 SDK

3: MSDNA-s visual studio 2012-be vagyok most

templates-ből  
 MonoGame Content Project

“windows is not a supported platform in this version of "xna game studio" or this edition...”

Hibaüzenet: megoldás: passz

feladom visual studiot

xamarin studio:

le kell hozzá külön tölteni a monogame-t, amit utána mint plugin felrakni, ez telerakja monogame-s templatekkel a projekt templateket.

Új projekt, Windows OpenGL monogame-s projekt.

Ezután:

System.DllNotFoundException: Unable to load DLL 'openal32.dll': The specified module could not be found.

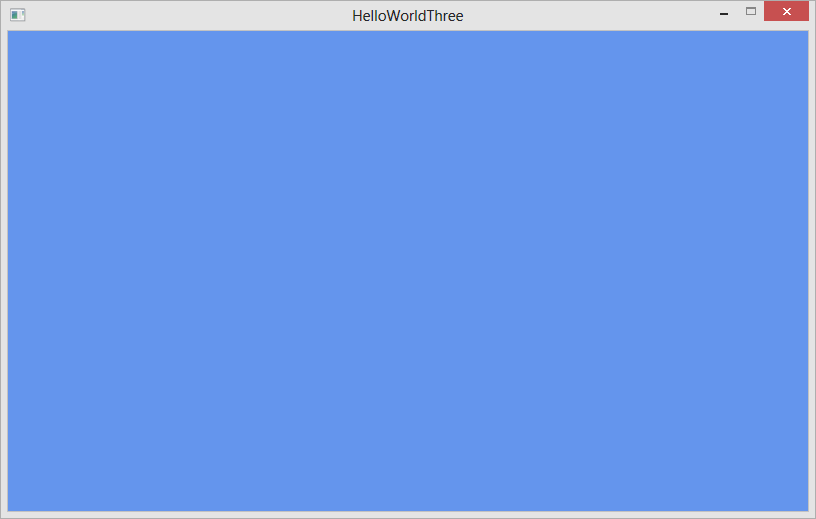
Megoldás:

<http://connect.creativelabs.com/openal/Downloads/oalinst.zip>

Fel kell rakni hozzá openAL-t.

openAL: OpenAL (Open Audio Library) is a cross-platform audio application programming interface (API).

Fordul, kék képernyő:



Többi projekthez kell a Xamarin.Android és a Xamarin.IOs

HelloWorld:

Még nincs kész MonoGame-hez az XNA-s Content Pipeline, azért Visual Studioban létrehoztam egy MonoGame Content projektet, amibe bele lehet pakolni a Contenteket (hang, kép, font, efekt) majd ezeket buildelve, megkapjuk a xnb fájlokat. Ezeket bele kell rakni a készülű projektünk Content mappájába. Csak ide, ezen belül lehet beljebb, de csak itt jó. LoadContent()-ben

sf = Content.Load<SpriteFont>("SpriteFont1");

betöltük (ez a Content/SpriteFont1 fájlra mutat), azátn az Draw-ban:

(legeleján a kódnak Content.RootDirectory = "Content"; be van állítva a directory)

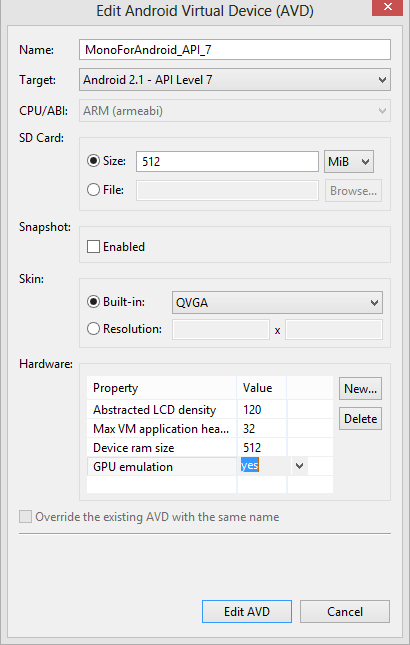
spriteBatch.DrawString(sf, "Hello World!", textpos, Color.White);



Androidos templatét csináltam:

Alapba nem talál két Reference-t gondolom direkt van így: le kell kapni MonoGame-t GitHub-ról, majd buildelni a megfelelő dll-eket. (MonoGame.Framework, Lidgren.Network)

Mepgróbálom elindítani, hiba: nem támogatja emu GPU emulációt. Be kell kapcsolni:



(szenvedés+gugli)

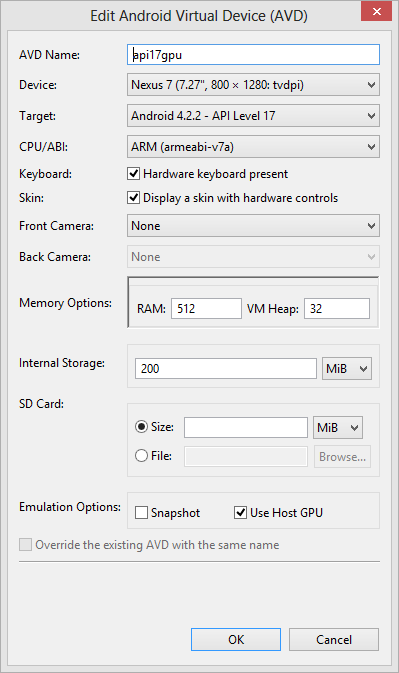
Csak 17 API lvl-es amulátorral sikerült müködésre birni az üres projektet.

"Monogame 3 is GLES 2.0 only. The emulators prior to 4.1 do not support GLES 2.0 . You will need to enable GPU emulation on a 4.1 Image or greater."

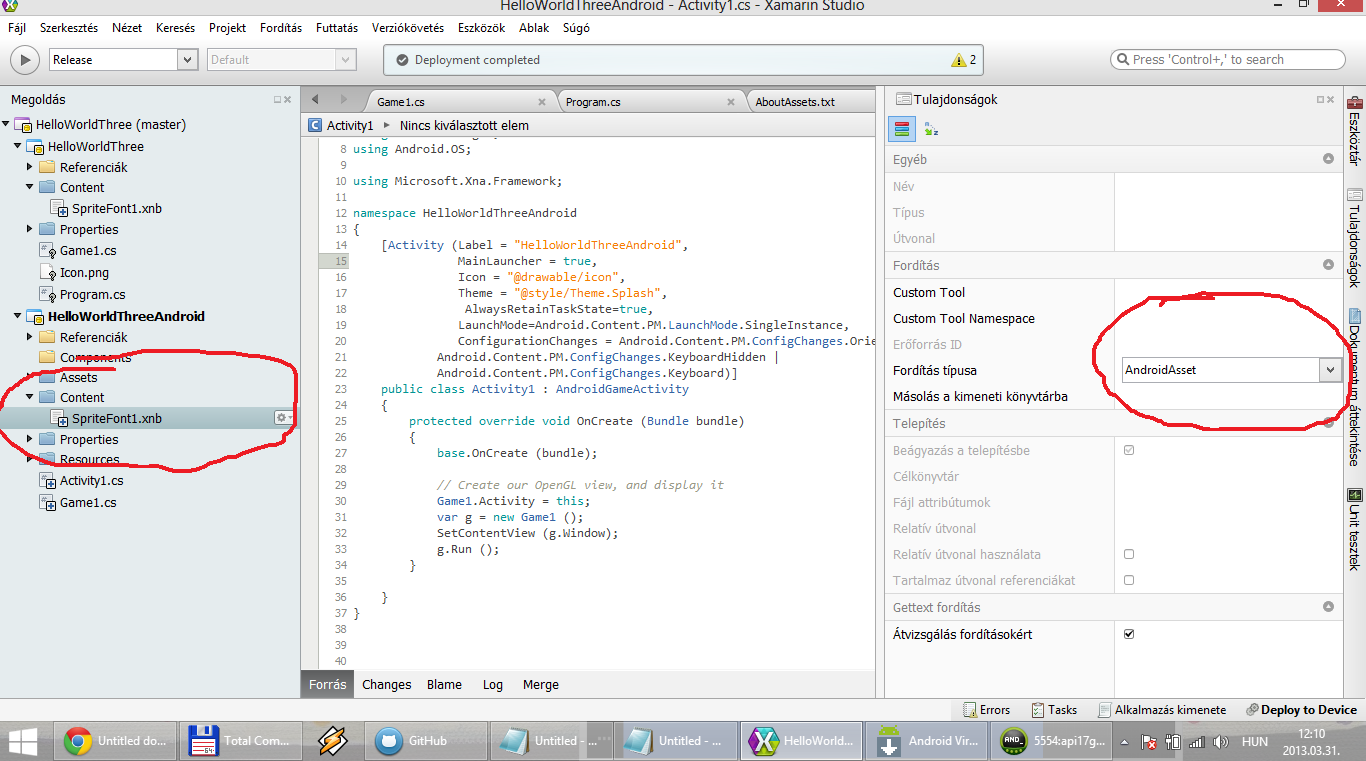
"GPU emulation is only available on Intel x86 Images and will only work on Intel Machines , if you have an AMD machine you will not be able to use the GPU emulation at all. If you have a ARM emulator Image , it will not work."

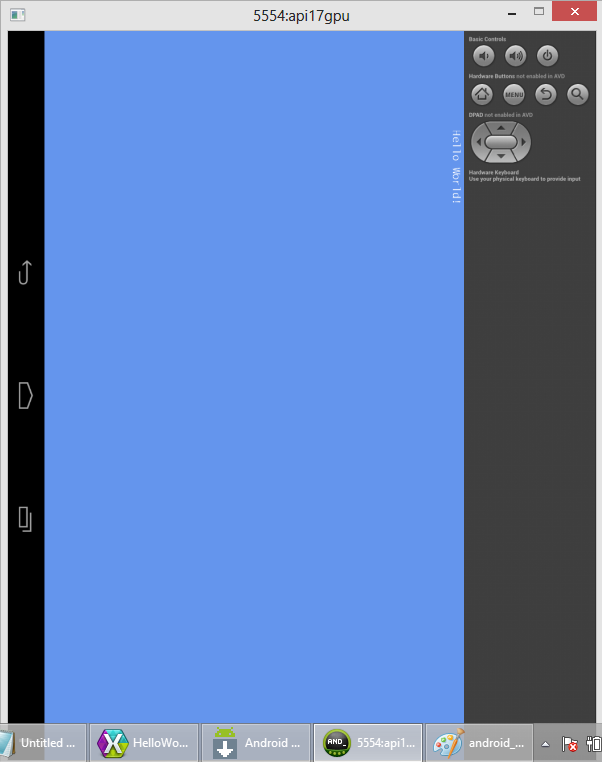
Ezt írták Monoék, ezért igy csináltam,de a fótumba mindkettőre rácáfoltak.

Éppen müködő AVD beállítás:



Ebben a pillanatban két egymástól teljesen független Game.cs fájl írja ki a helloWorld-öt, de a kódjuk egy az egyben megegyezik. Egyedüli macera a Content külön felvétele volt a androidos projektbe. (Content mappába berakni projektbe, majd beállítani hogy ez egy AndroidAsset)





Következő a telómon: Motorola Defy Froyo,

szó nélkül fordul és települ és megy (bár ott van egy fekete átmenet)



WP-re:

Új XNA-s WP projekt, ez magától csinál egy Content Referece-t is, amit az elején összehoztam font-nak, lehet jó ötletl lehet innen kreálni az egészet, lehet külön is fordítgatni a Content projektet, átfordulnak xnb-be a Contentek és lehet használni mindenhol. Az úgy nem müködött, hogy a másik projektben létrehozott xnb-t töltsem be, csak úgy volt hajlandó ha a saját Content projektjébe rakom bele. Game.cs fájl gyakorlatilag itt is ugyan az, betüről betüre, kivéve:

konsturktorban ez volt ahogy létrejön a template:

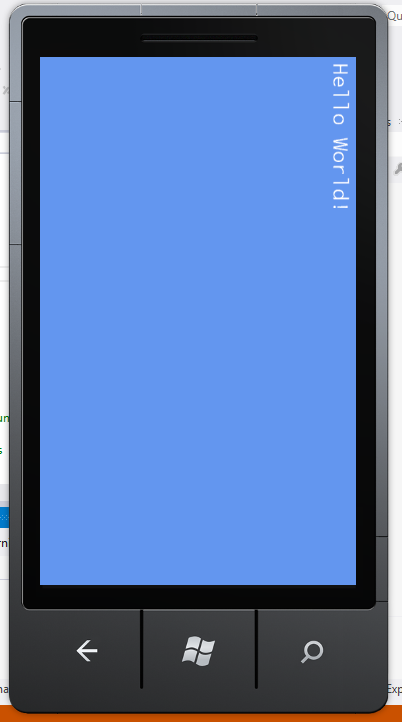
// Frame rate is 30 fps by default for Windows Phone.

TargetElapsedTime = TimeSpan.FromTicks(333333);

// Extend battery life under lock.

InactiveSleepTime = TimeSpan.FromSeconds(1);

Egyelőre nem piszkáltam.



MonoGame-éknek külön van WP8 támogatásul, mer Mictrosoft megszüntette az XNA támogatását WP8-ra, fut rajta is WP7 kompatibilitás módban -> nem lehet WP8as újdonságokat kihasználni. Ezért megnézem, hogy WP8-as projektetk XamarinStudioba mennyire szenvedés HelloWord-re bírni. Nem mégsem nézem meg egyenlőre, gépen nem viszi WP8 emut.