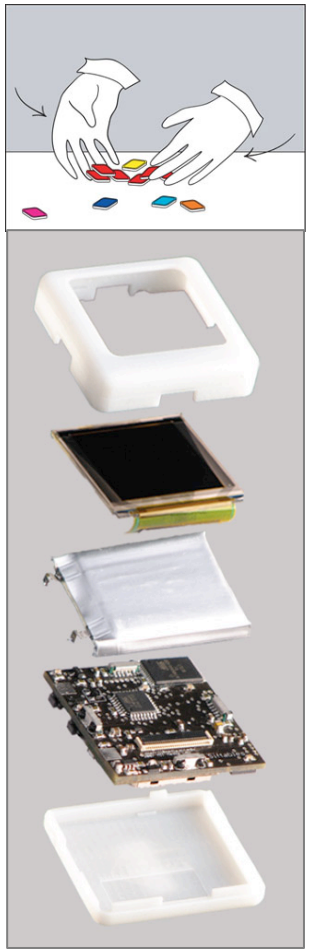
Siftables (2007)

MIT - „*Siftables aims to enable people to interact with information and media in physical, natural ways that approach interactions with physical objects in our everyday lives.*“

- Kombination von Sensor Network Technologies mit GUIs für direkte Interaktion mit digitalen Informationen.



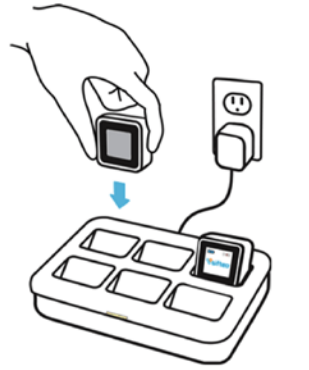
Siftables → Sifteos

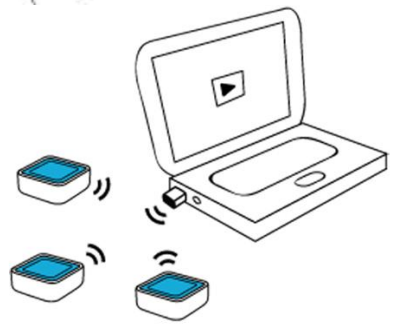
- https://www.sifteo.com/
- Interaktionsmöglichkeiten:

- Programmiersprache: C#
  - 2. Generation: C++
- Einschränkungen: Computer, max. 6 Sifteos
  - 2. Generation: nicht mehr nötig, max. 12 Sifteos

Sifteos in Betrieb nehmen

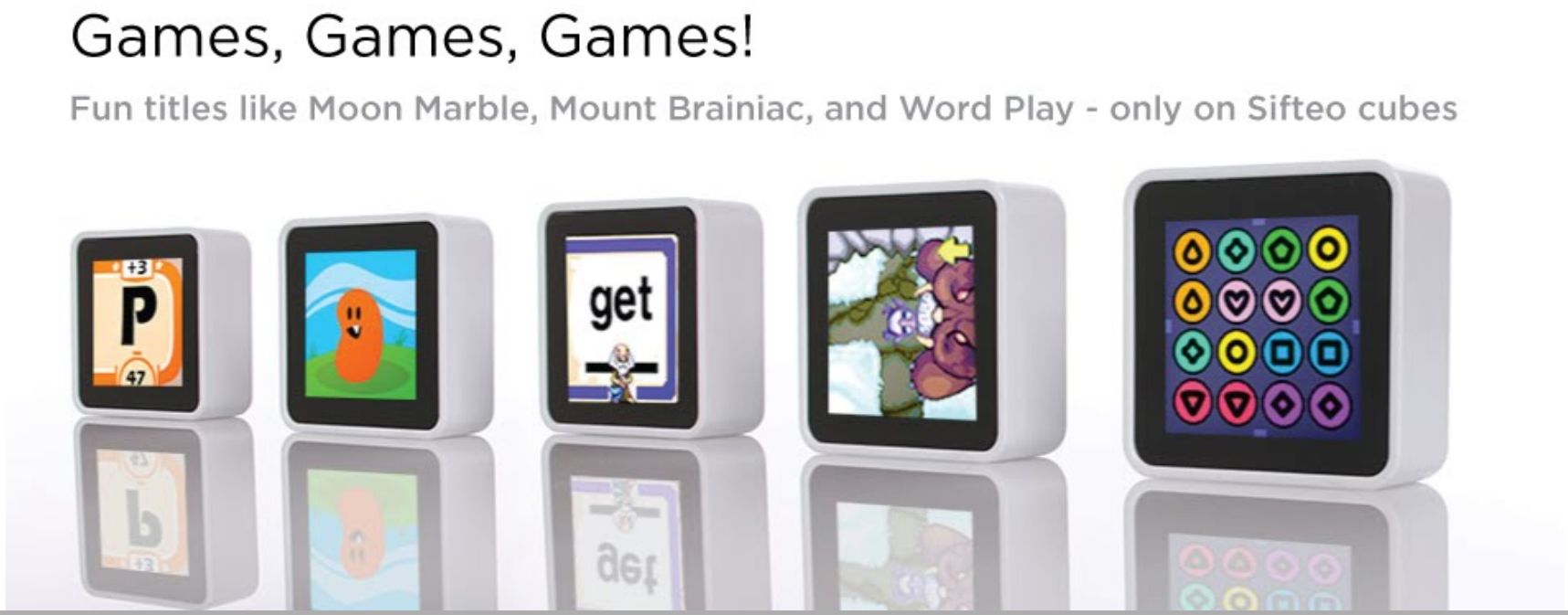
1. Aufladen
2. Account anlegen
   - https://www.sifteo.com/register
3. Download Siftrunner
   - https://www.sifteo.com/original/download
4. Dongle anstecken
5. Siftrunner starten und Sifteos verbinden
6. Spiel auswählen und installieren
7. Spiel starten





Spielen

Habt Spaß und probiert aus ☺



SDK installieren

- .NET 4.0 Framework, Mono runtime & MonoDevelop installieren
  - http://www.go-mono.com/mono-downloads
  - http://monodevelop.com/Download
- Sifteo SDK installieren
  - Windows: https://s3.amazonaws.com/updates.sifteo.com/Sifteo_SDK_win_1_1_3.exe
  - Mac: https://s3.amazonaws.com/updates.sifteo.com/Sifteo_SDK_mac_1_1_3.zip

C# vs. Java

- Java seit 1995 und C# seit 2002
- „*C# and Java are actually quite similar, from an application developer's perspective*“
- Beide arbeiten mit Klassen und sind objektorientierte Programmiersprachen
- Beide werden in Virtual Machines ausgeführt
- Beide nutzen Heap Based Classes and Garbage Collection

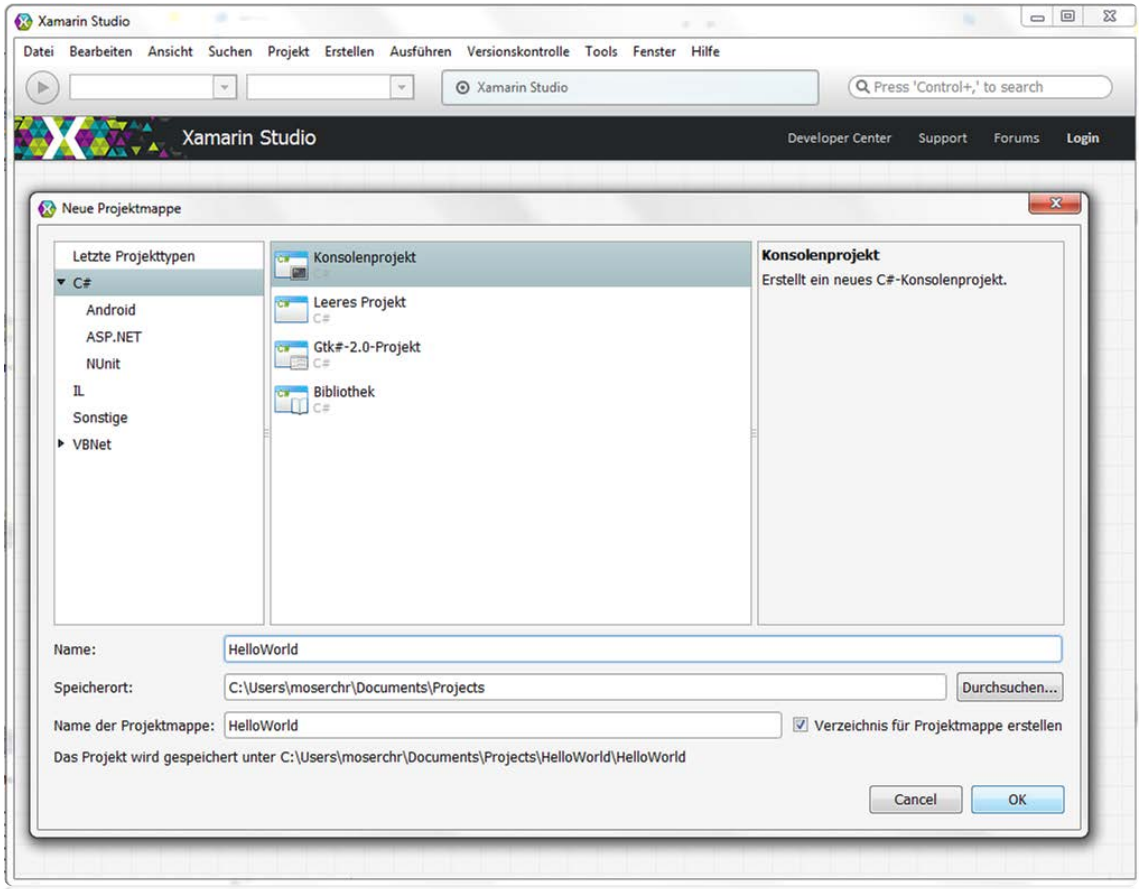
Keywords

C#	Java	C#	Java	C#	Java	C#	Java
abstract	abstract	extern	native	operator	N/A	throw	throw
as	N/A	false	false	out	N/A	true	true
base	super	finally	finally	override	N/A	try	try
bool	boolean	fixed	N/A	params	...	typeof	N/A
break	break	float	float	partial	N/A	uint	N/A
byte	N/A	for	for	private	private	ulong	N/A
case	case	foreach	for	protected	N/A	unchecked	N/A
catch	catch	get	N/A	public	public	unsafe	N/A
char	char	goto	goto	readonly	N/A	ushort	N/A
checked	N/A	if	if	ref	N/A	using	import
class	class	implicit	N/A	return	return	value	N/A
const	const	in	N/A	sbyte	byte	virtual	N/A
continue	continue	int	int	sealed	final	void	void
decimal	N/A	interface	interface	set	N/A	volatile	volatile
default	default	internal	protected	short	short	where	extends
delegate	N/A	is	instanceof	sizeof	N/A	while	while
do	do	lock	synchronized	stackalloc	N/A	yield	N/A
double	double	long	long	static	static	:	extends
else	else	namespace	package	string	string	:	implements
enum	N/A	new	new	struct	N/A	N/A	strictfp
event	N/A	null	null	switch	switch	N/A	throws
explicit	N/A	object	N/A	this	this	N/A	transient

Mein erstes C# Programm

Xamarin (MonoDevelop)

- Datei > Neu > Projektmappe > Konsolenprojekt
- Datei > Neu > Datei > C# > General > Leere Klasse



Hello World!

Java

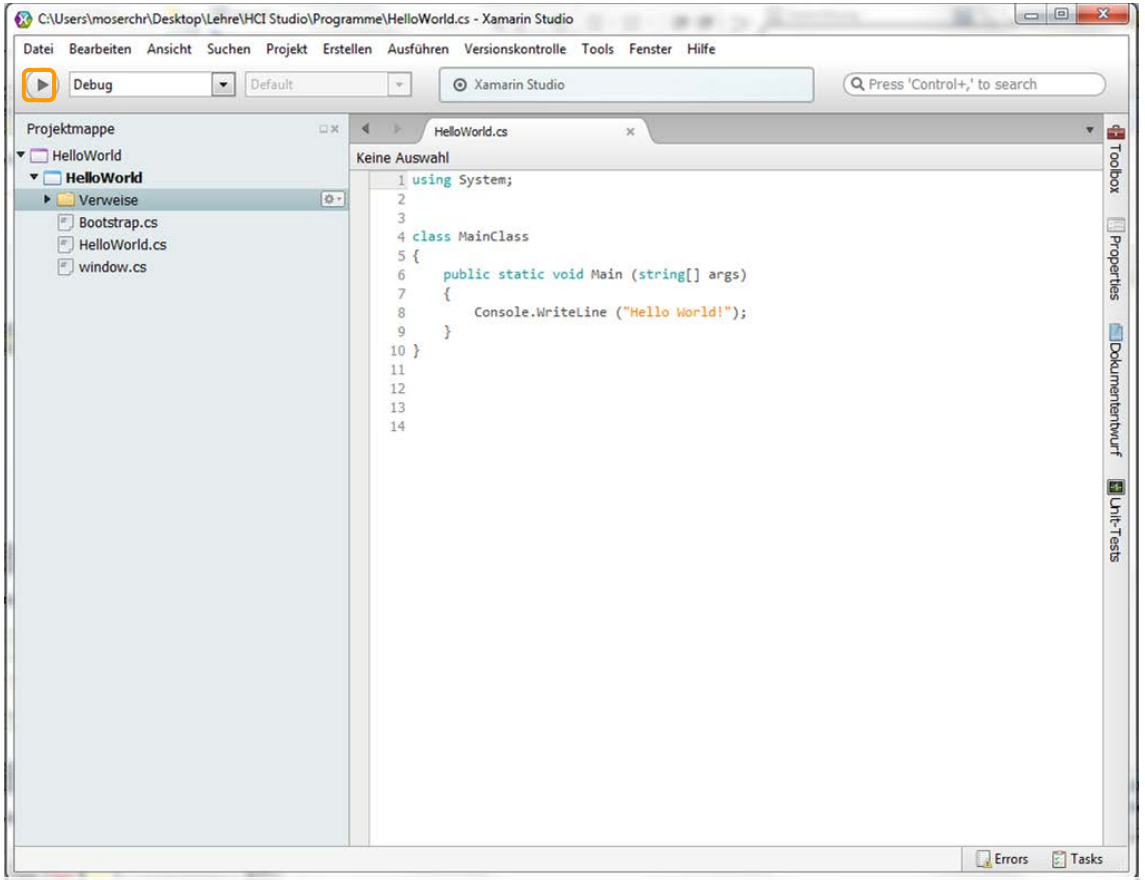
```java
package HelloWorld;

public class HelloWorld {

 public static void main(String[] args){

   System.out.println("Hello World");

 }

}
```

C#

```csharp
using System;

namespace HelloWorld {

 class MainClass {

   public static void Main (string[] args){

    Console.WriteLine ("Hello World!");
  }

 }

}
```

Hello World!

- Run Application



Sifteo App Development

Dokumentation – Sifteo Ordner api-docs
https://developers.sifteo.com/docs/gen1/

Project Generator

- Programme > Mono for Windows > Mono-2.10.8 Command Promt
- Navigiere in der Command Line in das folgende Verzeichnis
  cd ..\..\Sifteo\Sifteo-SDK-1_1_3\tools\project_gen (magische Tabulatortaste)
- Führe folgenden Befehl aus mono project_gen.exe HelloWorld

- Verschiebe den Ordner HelloWorld und erstelle einen Unterordner assets mit 3 Unterordnern images + sounds + instructions

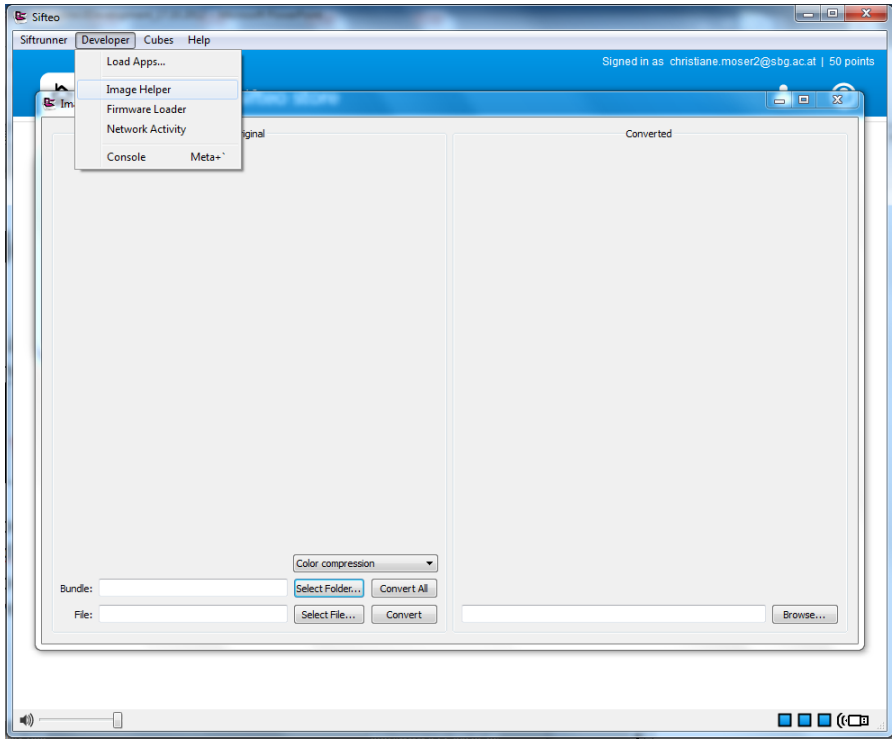
ODER

- Kopiere den template Ordner mit folgender Struktur
- template — Top level containing folder for your app
  - template.sln — Solution file for your entire app - open this in MonoDevelop
  - manifest.json — Manifest for your app bundle - specifies resources & configuration info
  - Template — Subdirectory for your main app logic
    - Template.cs — Main file for your app (add source files at this level as needed)
    - Template.csproj — Project file for your app

Hello World - Images

- 3 Bilder (.png) à 128x128 Pixel mit dem Text „Hello World!“ in unterschiedlichen Farben mit weißem Hintergrund erstellen

- Ersetze cover.png durch eines der neuen Bilder
- Mit Image Helper eine Bundle (images.sftbndl) erstellen
- Mit Image Helper die Bilder in .siftimg konvertieren

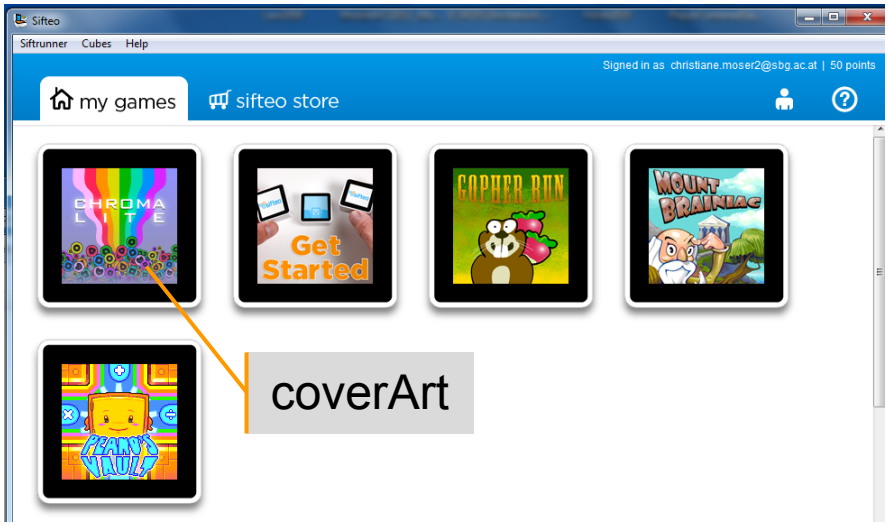


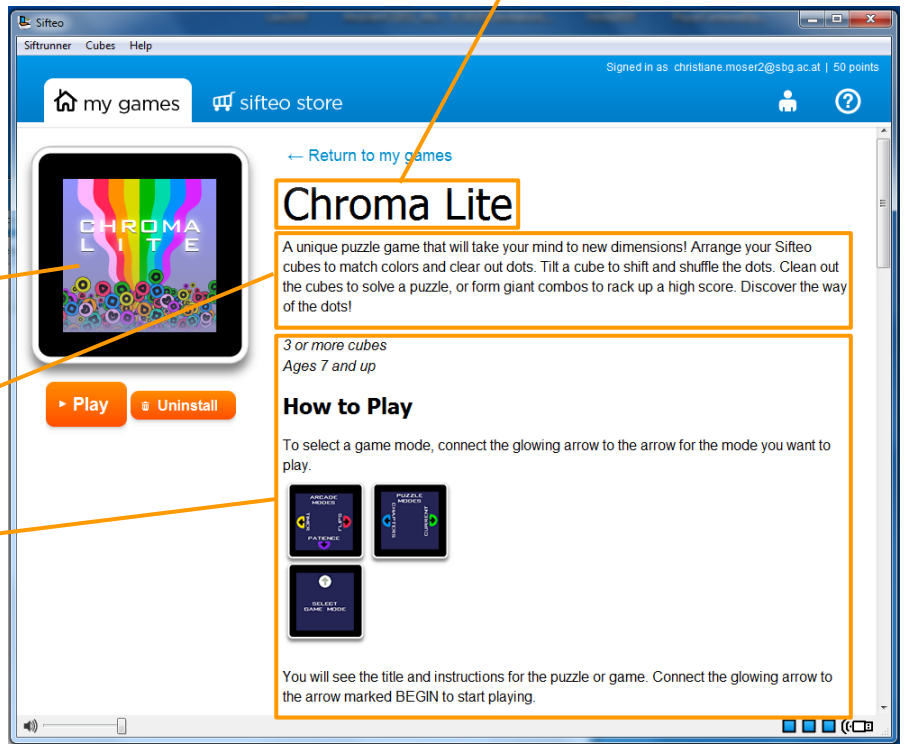
MonoDevelop starten

öffnen von HelloWorld.sln

öffnen von manifest.json

App Manifest





App Manifest - manifest.json

- title specifies the title that will be rendered in Siftrunner within the app library
- description is a short string that will be shown in Siftrunner once a user has selected your app
- instructions specifies the path, relative to manifest.json, of a directory containing html content to be rendered within the page shown once a user selects your app
- coverArt specifies the path, relative to manifest.json, to the image that will be rendered within Siftrunner within the 'My Games' library to represent your app. This should be sized at 128x128 pixels.
- internalArt specifies the path, relative to manifest.json, to the image that will be rendered within Siftrunner within your app's page, to represent your app. This should be sized at 170x170 pixels.
- requiredSiftCount specifies the minimum number of Sifteo cubes that must be connected in order to run this app
- devAppID provides a temporary app ID to be used when installing the app onto cubes. Generally, values in the range of 2000001-205000 are acceptable. Official application IDs will be allocated as part of the application submission process.
- imagesPath specifies the path, relative to manifest.json, that contains the image assets for your app
- soundsPath specifies the path, relative to manifest.json, that contains the audio assets for your app

manifest.json

Original

```
{
  "app" : {
    "title" : "Game",
    "version" : "0.0.1",
    "runtime" : "executable",
    "server" : "127.0.0.1:7000",
    "description" : "A freshly
                     generated game.",
    "appModule" : "UNUSED",
    "coverArt" : "cover.png",
    "requiredSiftCount" : 3,
    "soundsPath" : "assets/sounds",
    "imagesPath" : "assets/images"
  }
}
```

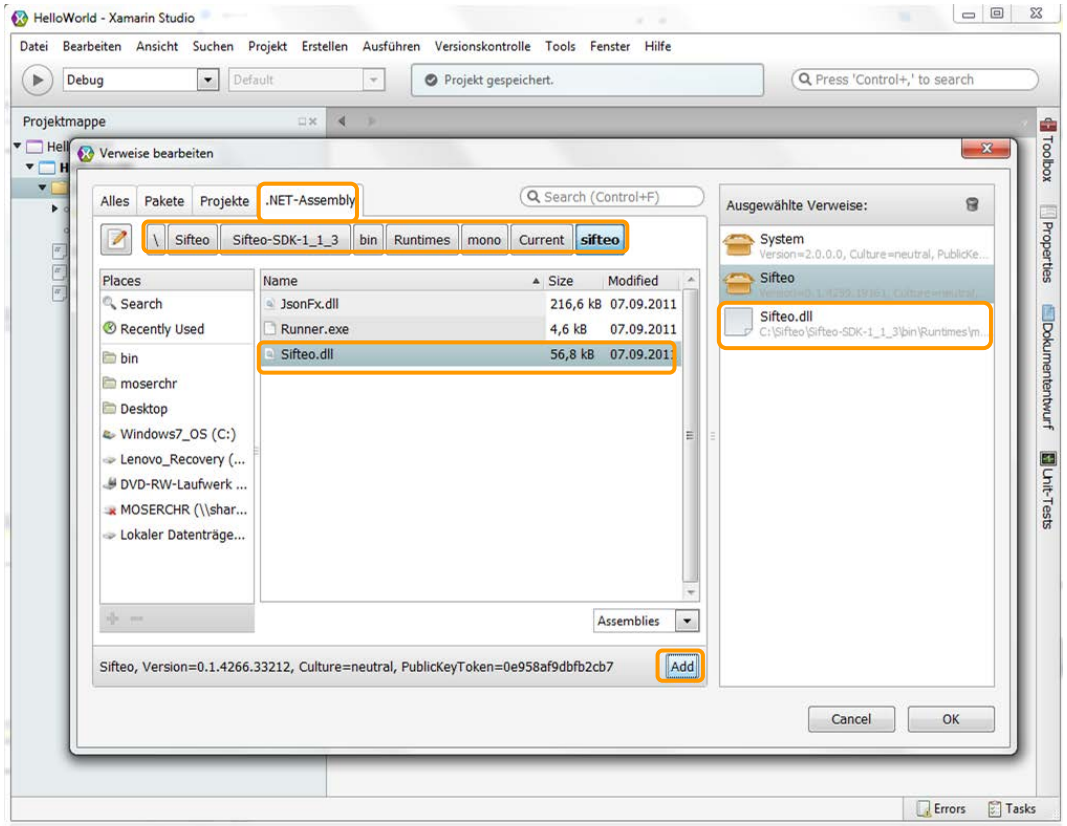
Neu

```
{
  "app" : {
    "title" : „Hello World!",
    "version" : "0.0.1",
    "runtime" : "executable",
    "server" : "127.0.0.1:7000",
    "description" : "Meine erste App.",
    "instructions" : "assets/instructions",
    "requiredSiftCount" : 3,
    "devAppID" : 2000001,
    "imagesPath" : "assets/images",
    "soundsPath" : "assets/sounds",
    "coverArt" : "cover.png",
    "internalArt" : "cover.png"
  }
}
```

Mein erstes Programm

Verweise setzen

- In der Projektmappe rechte Maustaste bei Verweise > Verweise bearbeiten
- Wechsele zu Reiter .NET-Assembly
- Add Sifteo.dll



C:\Sifteo\Sifteo-SDK-1_1_3\bin\Runtimes\mono\Current\sifteo\Sifteo.dll

Basic App Skeleton

```
using Sifteo;
using System;

namespace HelloWorld{

  public class HelloWorld : BaseApp {

    override public int FrameRate {
      get { return 20; }
    }
    override public void Setup() {
      Log.Debug("Setup()");
    }
    override public void Tick() {
      Log.Debug("Tick()");
    }
    static void Main(string[] args) { new HelloWorld().Run(); }

  }

}
```

Bei Programmstart erfolgt der Aufruf zur Initialisierung

Jede 1/20 Sekunde wird diese Methode aufgerufen um die Code auszuführen

HelloWorld.cs

```
...
    override public void Setup() {
      Log.Debug("Setup()");

      int i = 1;

      foreach (Cube cube in this.CubeSet) {
          string name = "HelloWorld" + i;

          cube.FillScreen(Color.Black);
          cube.Image(name, 0, 0, 0, 0, 128, 128, 1, 0);
          cube.Paint();

          i++;

       }
    }
...
```

Cube ist eine Instanz von einem Sifteo Cube

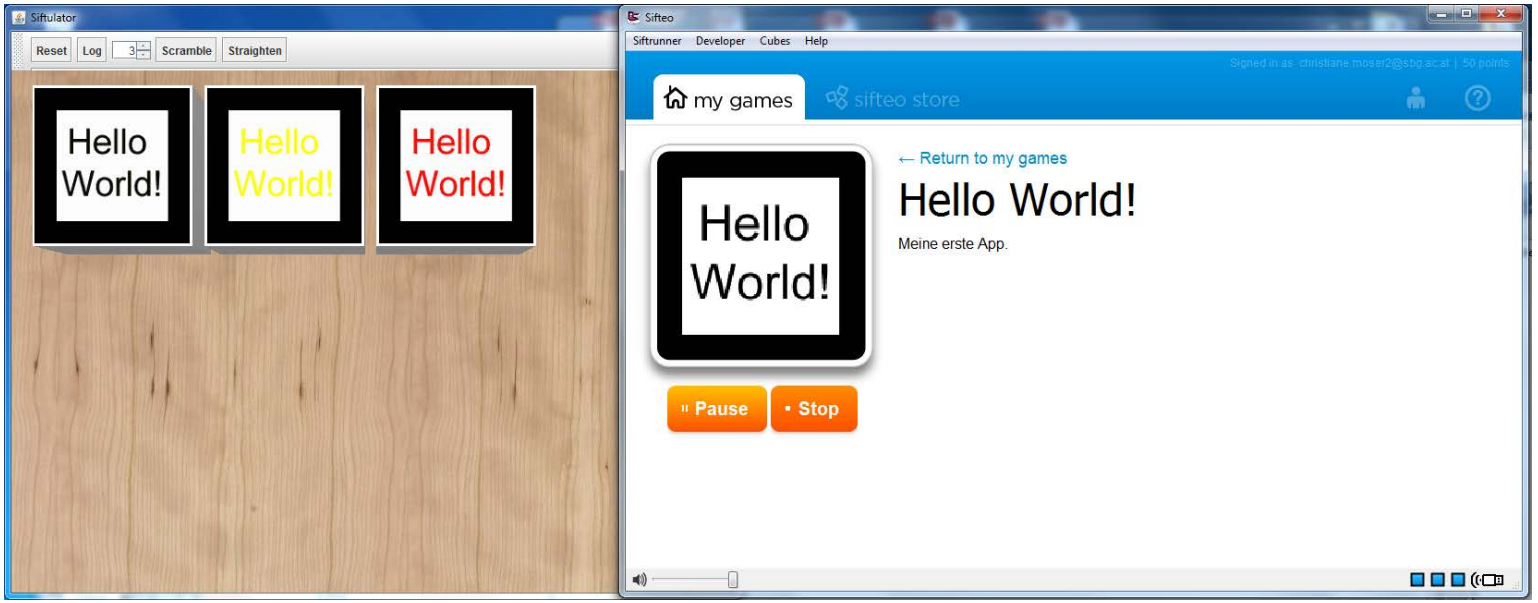
Beinhaltet alle gerade verbundenen Sifteo Cubes

Weist ein Bild einem Sifteo Cube UI zu
String name
Position x am UI
Position y am UI
Rendering von Position X
Rendering von Position Y
Höhe
Breite
Scale
Rotation

Zeigt einen Inhalt am Sifteo Cube UI an

HelloWorld.cs

- Run HelloWorld.cs
- Öffne Simulator &
  Setze den Assests Ordner
- Öffne Siftdev
- Lade App ... und Starte App



Mein zweites Programm

Project Generator

- Programme > Mono for Windows > Mono-2.10.8 Command Promt
- Navigiere in der Command Line in das folgende Verzeichnis
  cd ..\..\Sifteo\Sifteo-SDK-1_1_3\tools\project_gen
- Führe folgenden Befehl aus mono project_gen.exe FarbenTicker

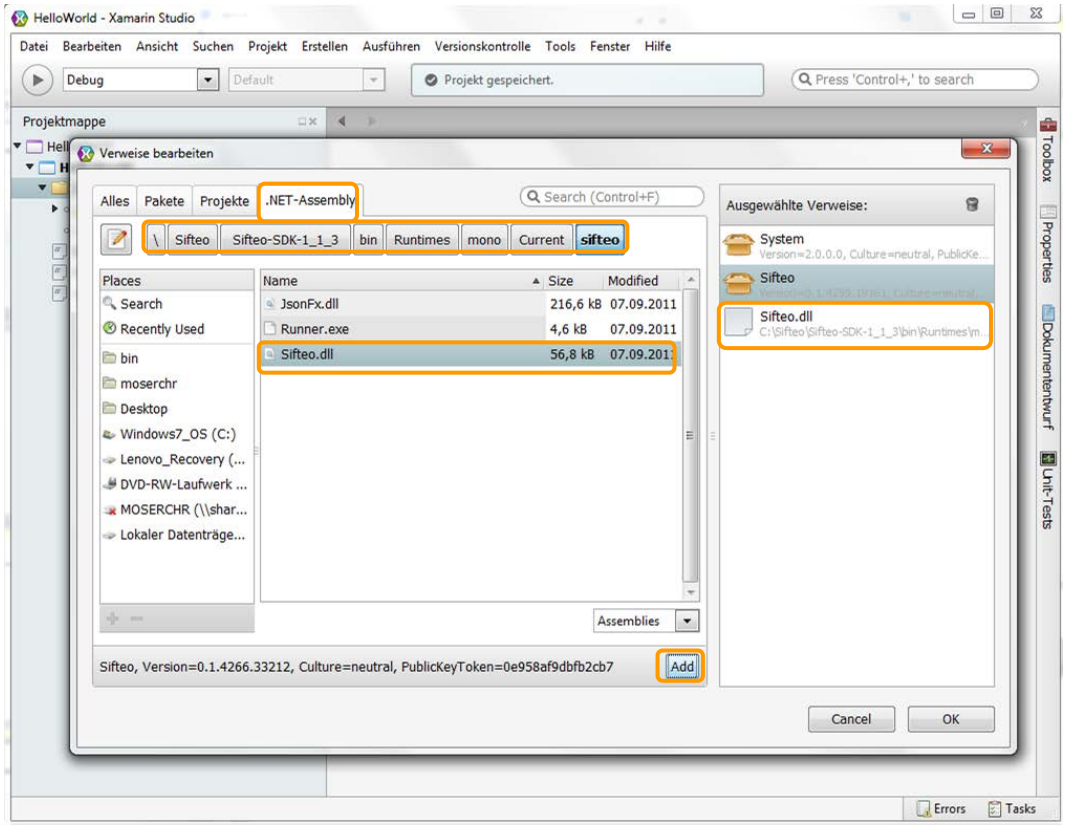
- Verschiebe den Ordner FarbenTicker und erstelle einen Unterordner assets mit 3 Unterordnern images + sounds + instructions

manifest.json

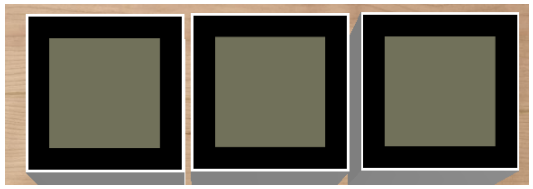
```
{
  "app" : {
    "title" : „FarbenTicker",
    "version" : "0.0.1",
    "runtime" : "executable",
    "server" : "127.0.0.1:7000",
    "description" : "Meine zweite App.",
    "instructions" : "assets/instructions",
    "requiredSiftCount" : 3,
    "devAppID" : 2000001,
    "imagesPath" : "assets/images",
    "soundsPath" : "assets/sounds",
    "coverArt" : "cover.png",
    "internalArt" : "cover.png"
  }
}
```

Verweise setzen

- In der Projektmappe rechte Maustaste bei Verweise > Verweise bearbeiten
- Wechsele zu Reiter .NET-Assembly
- Add Sifteo.dll
- Öffnen von FarbenTicker.cs



FarbenTicker.cs



```
...
    override public void Tick() {
      Log.Debug("Tick()");



      foreach (Cube cube in this.CubeSet) {

          Color color = new Color (100, 100, 100);

          cube.FillScreen (color);
          cube.Paint ();
      }
    }

...
```

Farbe die von Sifteo Cubes unterstützt wird
RR ... rot
GG ... grün
BB ... blau

RGB Werte aus Graphik-programmen (0 bis 255) *

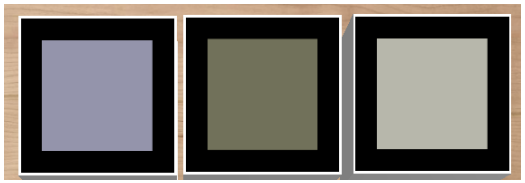
Füllt das Sifteo Cube UI in der angegebenen Farbe

* http://www.rapidtables.com/web/color/RGB_Color.htm

FarbenTicker.cs

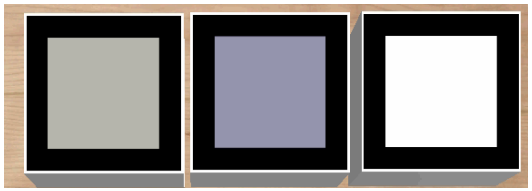
- Run FarbenTicker.cs
- Öffne Simulator
- Öffne Siftdev
- Lade Apps ... und Starte App

FarbenTicker.cs



```
...
    override public void Tick() {
      Log.Debug("Tick()");

      int r = 100; int g = 100; int b = 100;

      foreach (Cube cube in this.CubeSet) {

          Color color = new Color (100, 100, 100);
          r = r + 50; g = g + 50; b = b + 50;
          cube.FillScreen (color);
          cube.Paint ();
      }
    }

...
```

FarbenTicker.cs



```
public int r = 0; public int g = 0; public int b = 0;
...
    override public void Tick() {
      Log.Debug("Tick()");

      int r = 100; int g = 100; int b = 100;

      foreach (Cube cube in this.CubeSet) {

          Color color = new Color (r, g, b);
          r = r + 5; g = g + 5; b = b + 5;
          cube.FillScreen (color);
          cube.Paint ();
      }
      }
      else {
          r = 0; g = 0; b = 0;
      }
    }
```