



# CLIPGENERATOR MOBILE



# Inhaltsangabe

## Projektbeschreibung

- Anforderungen
  - Use Cases
- Projektteam
- Projektzeitraum
- Projekttreffen (Briefings)

## Analyse

- Marktanalyse
  - Grobanalyse
  - Detailanalyse
  - Gegenüberstellung CF und J2ME

## Design

- Designentscheidungen
  - Farben
  - Struktur der Benutzeroberfläche
  - Eigene GUI-Komponenten
  - Verwaltung der GUI-Grafiken
- GUI-Elemente
  - Grundlayout eines Forms
- Verwaltung der GUI-Elemente
  - Klassen-Struktur

## Technik (Backend)

- Kommunikation mit dem VideoServer
- Einbindung von Windows Media Player
- Externe Datenhaltung in der Anwendung



# Inhaltsangabe

## Integration

- Einbindung der SOAP-Funktionen in die GUI
- Paketdiagramm

## Testing

- Grobtests
- Usabilitytest

## Zukunftsaussichten für das Programm

- Mögliche Erweiterungen

## Lessons Learned

- Erkenntnisse
- Problemstellen
- Fazit





# Anforderungen



Ziel des Projektes „Der Clipgenerator auf dem Handy“ soll es sein, den Internetdienst [www.clipgenerator.de](http://www.clipgenerator.de) bzw. [www.clipgenerator.eu](http://www.clipgenerator.eu) auf die mobile Plattform zu übertragen.

Beim Clipgenerator handelt es sich um eine Applikation, die aus privaten Bildern, automatisch ein Musicvideo generieren kann. Dazu sucht der Nutzer sich eine Musik aus, lädt die erforderliche Anzahl an Bildern hoch und legt die Bildreihenfolge fest. Der Clipgenerator errechnet dann aus diesen Daten das fertige Video und sendet es dem Nutzer zu.

In diesem Projekt geht es darum, ein funktionsfähiges Userinterface auf einem mobilen Plattform zu gestalten, das folgenden Programmablauf abdeckt.



- User stellt Verbindung zum Clipgenerator her.  
-> Clipgenerator liefert eine Auswahl an zur Verfügung stehenden Clip-Templates zurück.



- User wählt seine Vorlage aus eine Liste aus

■ Projektbeschreibung



# Anforderungen



- User geht sein Bilder durch. Ein definierter Tastendruck wählt das aktuelle Bild aus und lädt es in die Bildfolge



- Durch definierte Tasten lassen sich einzelne Bilder verschieben



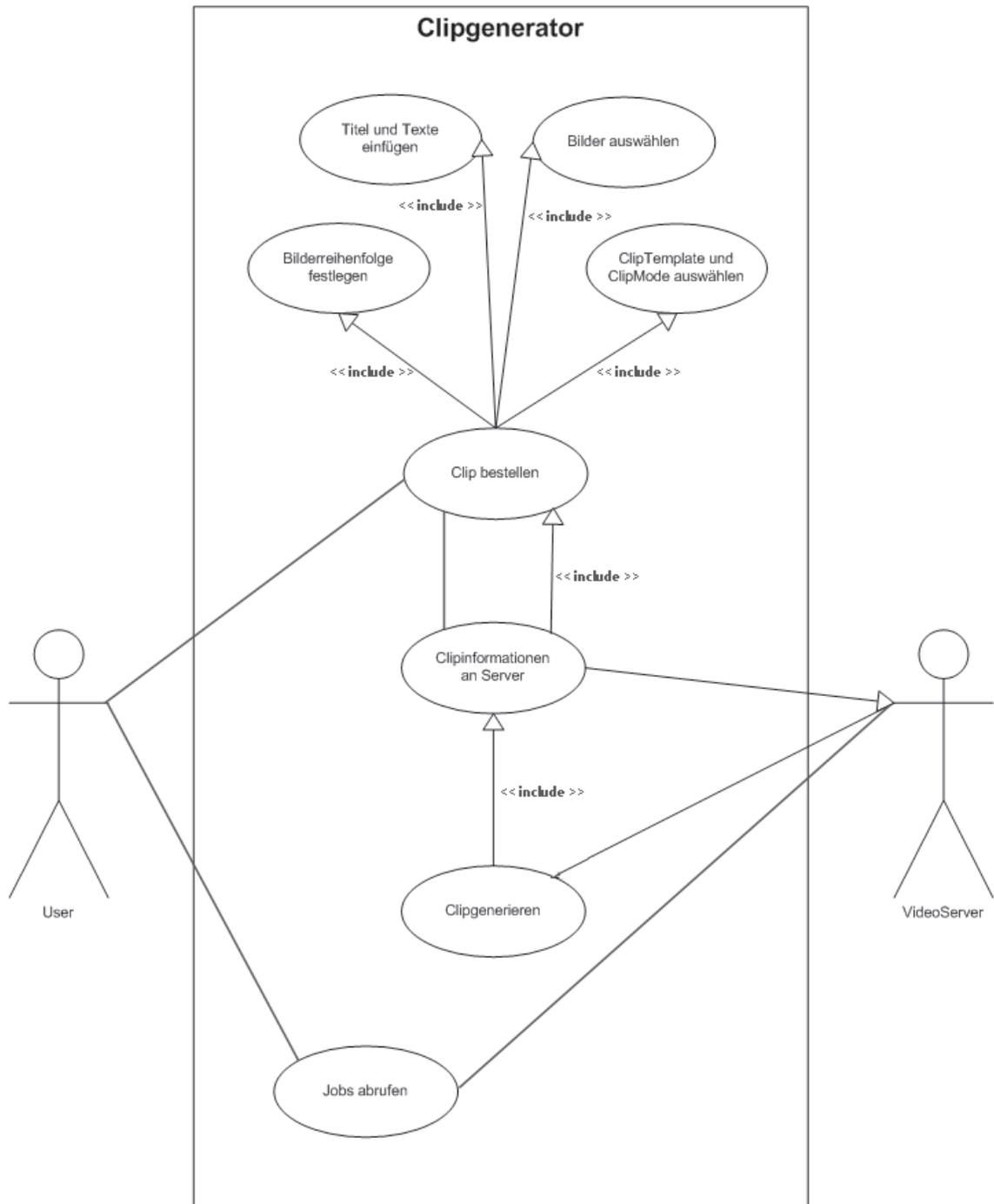
- Es besteht die Möglichkeit, Texte in das Video einzufügen.



- Eine Vorschau wird angezeigt.
- Der Nutzer gibt den Clip in Auftrag.
- Clipgenerator berechnet in kurzer Zeit das Video.
- Clipgenerator benachrichtigt User über Fertigstellung per SMS
- User lädt sich den Clip herunter.



# Use Cases



Hier wird durch ein Use Case-Diagramm die Grundfunktionen, die eine Benutzer zur Verfügung stehen, beschrieben.

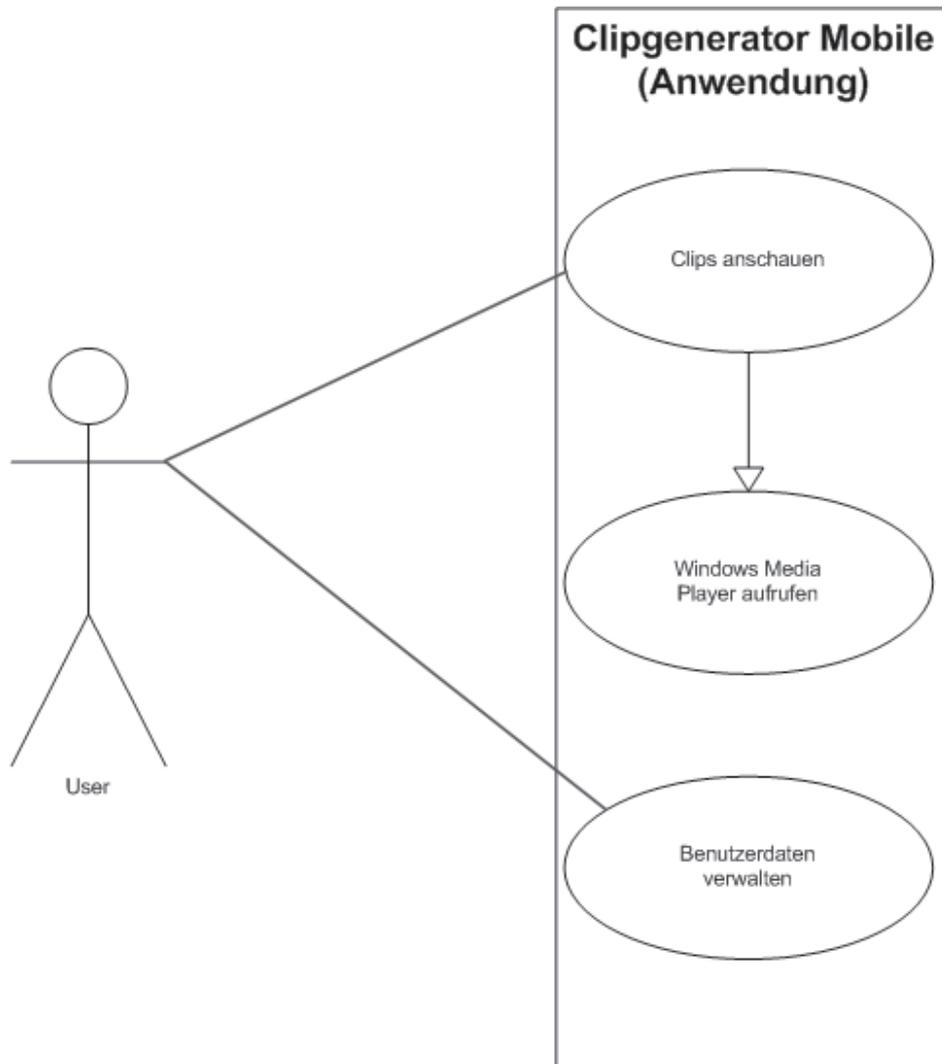
■ Projektbeschreibung



# Use Cases

Die Anwendung ist nicht nur dafür da um Clips zu erstellen, sondern es soll auch die Möglichkeit bieten erstellte Clips aus dem Programm heraus abzuspielen.

Des weiteren muss auch die Verwaltung von Benutzerdaten gewährleistet werden, welches später für eine mögliche Markteinführung wichtig werden würde.



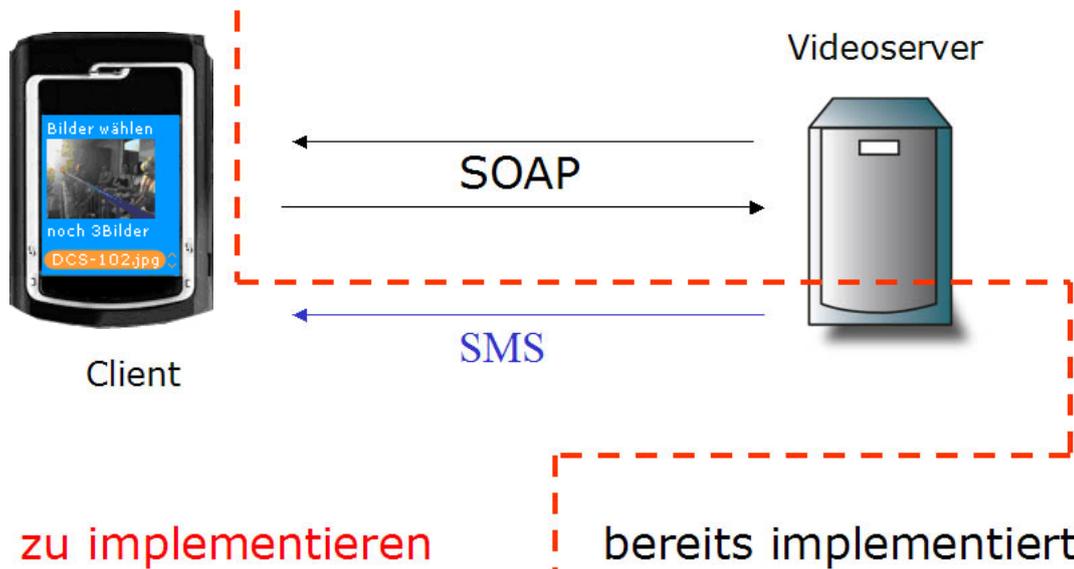
Zu dieser Zeit hatte man eine grobe Ansicht über die grundlegenden Funktionen der zu entwickelnden Software.

Doch wie diese Funktionen realisiert werden war zu diese Zeit nicht festgelegt.



# Aufgabe des Projektteams

Übersicht: Programmieraufgabe/-grenzen



Die Hauptaufgabe des Projektteams war eine mobile Applikation auf einem Smartphone / Pocket PC zu entwickeln. Dabei war nicht gegeben, auf welcher Plattform und mit welcher Programmiersprache dies zu realisieren ist.

■ Projektbeschreibung



# Projektteam

## Projektmitglieder

Ferhat Tatar  
Allmandring 08A 13  
70569 Stuttgart

Mobil: 0176-203 87 6 92  
Email: [ft008@hdm-stuttgart.de](mailto:ft008@hdm-stuttgart.de)  
Skype: ferhat1984

Markus Heimann  
Neckarstr. 35  
71083 Herrenberg

Tel.: (07032)-79 71 792  
Mobil: 0176-610 69 842  
Email: [mw044@hdm-stuttgart.de](mailto:mw044@hdm-stuttgart.de)  
Skype: marheimann

## Projektbetreuer HdM

Dr. Johannes Maucher  
Nobelstrasse 10  
70569 Stuttgart

Tel.: 0711 89 23-21 78  
E-mail: [maucher@hdm-stuttgart.de](mailto:maucher@hdm-stuttgart.de)

Thomas Suchy  
Nobelstrasse 10  
70569 Stuttgart

Tel.: 0711 89 23 - 21 58  
E-mail: [suchy@hdm-stuttgart.de](mailto:suchy@hdm-stuttgart.de)

## Projektbetreuer Trivid GmbH

Simon Meyborg  
Karlstrasse 19  
72135 Tübingen-Dettenhausen

Tel.: (0421) 59 63 193  
Mobil: 0176 - 700 64 536  
Email: [s.meyborg@trivid.de](mailto:s.meyborg@trivid.de)

Angelika Hänsler  
Pfadfinderweg 20  
5020 Salzburg

Tel.: (07157) - 56 43 20  
Mobil: 0172 - 736 02 37  
Email: [a.haenssler@clipgenerator.com](mailto:a.haenssler@clipgenerator.com)



# Projektzeitraum

13.Oktober - 02.November

Marktanalyse  
Welche Plattformen gibt es?  
Welche Programmiersprachen gibt es für die einzelnen Plattformen?

02.November - 22.Dezember

Programmierung der Applikation  
Design

22.Dezember - 15. Januar

Testphase  
Vorbereitung der Präsentation für die MI-Präsentationstag und Medianight

15.Januar - 19.Februar

Projektdokumentation

## Kompetenzen (Aufgabenverteilung)

Ferhat Tatar

- User-Interface (Benutzeroberfläche)
- Projektdokumentation
- Benutzerhandbuch
- Zusammenführung der Programmteile
- Testen

Markus Heinemann

- SOAP Kommunikation
- Windows Media Player Anbindung
- Datenhaltung im Programm
- Zusammenführung der Programmteile
- Testen

 Projektbeschreibung



# Treffen

## Treffen am 13. Oktober 2006, mit Trivid- und HdM Betreuer

Wir trafen uns am 13. Oktober für eine Vorstellungsrunde.  
Dabei haben wir uns unsere Ziele festgelegt und besprochen die Einzelheiten.  
Zusätzlich haben wir einen groben Projektzeitplan aufgestellt.

Die Aufgabe zunächst war eine Marktanalyse über mögliche Plattformen und Programmierungssprachen durchzuführen.

Wir haben uns einen Termin für Anfang November geeinigt, wo wir dann die Marktanalyse präsentieren sollten und uns über die zu einsetzende Technologie entscheiden.

## Teamtreff am 19. Oktober 2006

Analyse der Ergebnisse der Marktanalyse.  
Beantwortung und mögliche Antworten für folgenden Fragen:  
Welche Technologien kommen für und in Frage?  
Was können wir , welche Technologie kann das unterstützen, was wir wollen?

## Treffen am 2. November 2006, mit Trivid- und HdM Betreuer

**Ablauf:** Präsentation über unsere Marktanalyse.  
Wir haben uns darauf geeinigt, die Applikation auf dem Microsoft Mobile 5.0 Plattform mit dem Compact Framework und mit der Programmierungssprache C# zu entwickeln.

Anschließend zeigten wir ein Bsp Programm auf einem Pocket PC.  
Wir sprachen über die Möglichkeit in Messen aufzutreten.  
Entscheidung über den Auswahl des Smartphones. Wir entschieden uns für den HTC TyTN.

### Grobe Verteilung der Kompetenzen im Team:

- Ferhat Tatar ist für Dokumentation und Design zuständig
- Markus Heimann ist für die Funktionale Programmierung zuständig



# Treffen

## **Teamtreff am 13. November 2006**

Wir trafen uns um über den aktuellen Projektstand zu informieren. Dabei haben wir uns über bestimmte Merkmale der Anwendung unterhalten.

## **Teamtreff am 4. Januar 2007**

Einbindung weiterer SOAP-Funktionen in die Anwendung.  
Testen der Anwendung auf Fehler und mögliche Schwachpunkte.

## **Teamtreff am 7. Januar 2007**

Die Hauptaufgabe an diesem Tag war die Einbindung der externen Datenhaltung in Form von XML in die Anwendung. Das war notwendig, da die Anwendung auf die Benutzer abgestimmt werden muss, was die Authentifizierungsdaten angeht.

## **Treffen am 11. Januar 2007, mit Trivid-Betreuer**

Wir trafen uns mit Frau Hänssler und Herrn Meyborg um über den Ablauf der externen Veranstaltung an der MediaNight zu entscheiden.

Zunächst präsentierten wir die aktuelle Anwendung, womit die Trivid-Betreuer sehr zufrieden waren. Wir überlegten uns mögliche Zukunftsansätze der Anwendung.

## **Treffen am 18. Januar 2007, mit Prof. Dr. Maucher**

Wir haben uns mit Herrn Prof. Dr. Maucher zusammengesetzt um über den aktuellen Projektstatus zu reden. Zudem hat man sich über die kommenden Präsentationen unterhalten.

**Prof. Dr. Maucher war mit dem bisherigen Projektverlauf sehr zufrieden.**

 Projektbeschreibung



# Treffen

## **Teamtreff am 16. Januar 2007**

Besprechung über die Präsentationen für den MI-Präsentationstag und MediaNight. Wir haben uns über den möglichen Ablauf der Präsentation unterhalten.

Zudem haben wir uns über mögliche Inhalte für die Poster, die wir an der MediaNight an unserem Stand hängen, festgelegt.

## **Teamtreff am 23. Januar 2007**

Erstellung der Präsentationsfolien für den MI-Präsentationstag und auch für die MediaNight.

## **MI-Präsentationstag am 24. Januar 2007**

Auf der MI-Präsentation haben wir unser Projekt vorgestellt. Aufgrund der beschränkten Zeit haben wir nur Infos über unseren Projektverlauf präsentiert.

## **MediaNight am 25. Januar 2007**

Als erstes haben wir den Raum 148 für die externe Präsentationsveranstaltung von Trivid ausgeräumt. Ab 16.30 haben wir unseren Stand im Foyer aufgebaut.

Um 18.10 haben wir die externe Veranstaltung im Raum 148 gemacht. Die Gäste bekamen zuerst Infos über die Firma Trivid GmbH. Anschließend hat Markus Heilmann unser Projekt vorgestellt. Wir blieben für einige Zeit um die Fragen zu beantworten.

Abwechselnd haben wir im Foyer die Anwendung der Besucher vorgestellt.

Um 22:00 haben wir dann unseren Stand abgebaut.



# Grobanalyse

## Mobile Betriebssysteme und Programmierungssprachen

Welche Plattformen (Betriebssysteme) gibt es?

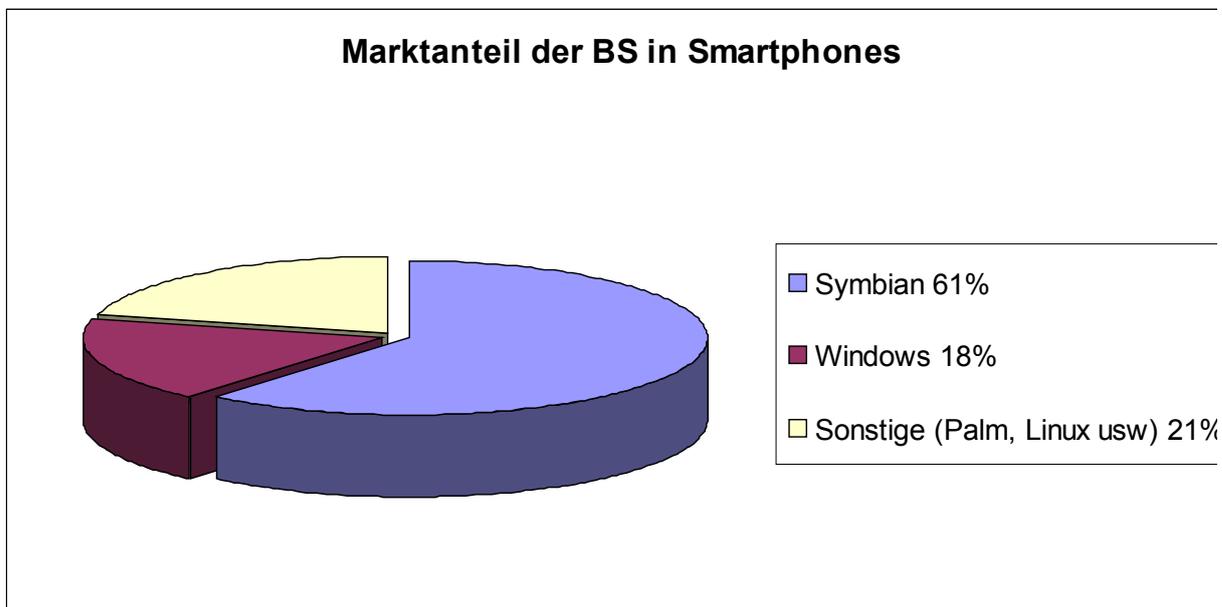
- Symbian
- Microsoft Mobile 5 / CE
- Java

Welche Programmiersprachen sind möglich?

- C++
- Compact Framework (C# oder VB)
- Java Micro Edition

## Marktanteile von mobilen Betriebssystemen

- Artikel von [www.teltarif.de](http://www.teltarif.de) vom 27.04.2005 über Smartphones



Marktanalyse



# Detailanalyse

- Symbian Homepage (Marktanteile weltweit)

Worldwide total smartphone device market - Market shares 2006 Q2 2005 / Q2 2006		
	Q2 2005	Q2 2006
Symbian	75.19%	69,7%
Microsoft	3.5%	4,92%

- Palm setzt auf Microsoft Mobile OS

- 7 der 8 neusten Smartphones haben Microsoft Mobile 5.0 als OS (c´ t von 30.10.06)

=> Wir konnten leider keine aktuellen Studien finden, da diese z.B. von Gartner sehr teuer zu bekommen sind.

Plattformen und Programmierungssprachen, welches wird für das Projekt berücksichtigen konnten:

- Java mit Programmierungssprache J2ME
- Microsoft Mobile 5.0 mit dem Compact Framework und C# als Programmierungssprache

# Gegenüberstellung CF mit C# und J2ME

## Plattformabhängigkeit

- CF läuft nur auf Windows Plattformen
- J2ME/MIDP funktioniert auf allen Plattformen, auf den der Java Virtual Machine läuft. In diesem Fall die KVM (KiloByte Virtual Machine)

## Portierbarkeit zwischen Geräten

C# -Anwendungen sind nur auf Windows Geräten lauffähig, dort aber i.d.R. uneingeschränkt, da das CF auf allen gleich ist.

J2ME wird oft durch Pakete einzelner Anbieter erweitert und wird so plattformabhängig, selbst zwischen den Geräten eines Anbieters kann dies schon zu Problemen führen. Selbst die Standardplattform MIDP kann von Anbieter zu Anbieter unterschiedlich implementiert sein.

->Anpassungen notwendig wenn portiert wird. Ohne diese Anbietererweiterungen ist die Entwicklung jedoch mit erheblichem Mehraufwand verbunden

## Anforderungen an das Gerät

Das Kompakt Framework ist relativ umfangreich und benötigt daher recht viel Speicherplatz, dafür sind aber auch grundsätzlich keine Zusatzbibliotheken notwendig.

J2ME/MIDP ist kompakter und daher Speicherfreundlicher, benötigt aber i.d.R. Zusatzbibliotheken um damit besser Entwickeln zu können. Diese Zusatzpacks sind aber von Hersteller zu Hersteller oftmals nicht kompatibel.

## Unterstützung für Entwickler

Das Entwickeln von Anwendungen für das CF ist schnell erlernbar vor allem wenn man das .NET Framework bereits kennt, denn das CF ist eine Teilmenge davon. Es sind keine Unterschiede zwischen den Windows Geräten zu beachten.

Bei der Entwicklung von Anwendungen mit J2ME muss man auf die Unterschiede zwischen den Geräten achten und diese berücksichtigen. Generell macht dies die Entwicklung Aufwendiger.

 Marktanalyse



# Fazit

## Zusammenfassung

Beide Plattformen, CF und J2ME haben ihre Stärken und Schwächen. J2ME sind auf fast allen Plattformen lauffähig, CF Anwendungen nur auf Windows Geräten. J2ME wird von vielen Anbietern unterstützt, die auch Tools und SDK zur Entwicklung zur Verfügung stellen. Aber die Entwicklung von Anwendungen für das CF ist einfacher und Schneller als die Entwicklung von Anwendungen für J2ME. Die Unterstützung durch VS2005 ist besser und es sind keine Unterschieden zwischen den Geräten zu beachten.

## Fazit

Geht es um Anwendungsentwicklung für kleine low-end Geräte, so ist J2ME die Wahl, denn in diesem Bereich ist Windows nicht vertreten. Auf Smartphones und PDAs, welche über eine größere Leistungsfähigkeit verfügen ist jedoch das CF besser, da es mehr bietet, ohne das es Zusatzpacks anderer Hersteller benötigt.

## Entscheidung über Plattform und Programmiersprache

Aufgrund der Marktanalyse waren wir der Meinung, dass wir uns aus zeitlichen und technischen Gründen von J2ME fernhalten sollten.

Mit dem Compact Framework und C# als Programmiersprache konnten wir nahezu garantieren, dass wir die Anforderungen realisieren können.

Auf dem Treffen von **02.11.2006** mit den Trivid- und HdM Betreuern haben wir uns auf die Entwicklung der Software auf der Windows Mobile Plattform geeinigt.

Als Pocket PC haben wir uns auf das von HTC TyTN entschieden, welches Wir nach kurzer Zeit per Post zugeschick bekommen haben.



# Designentscheidungen

## Farben

Da unsere Zielgruppe aus den jüngeren Nutzern besteht, hab ich mich für die Farben „Orange“ und „Grau“ entscheiden. Für das Hintergrund entschied ich mich für die Farbe „Weiß“.

## Struktur der Benutzeroberfläche

Eine wichtige Entscheidung für uns war, wie wir die Benutzeroberfläche realisieren. Es gab 2 verschiedene Möglichkeiten, entweder verwenden wir ein Objekt der Klasse Form, wo wir mehrere Panels haben und je nach Aktion die Komponenten sichtbar oder unsichtbar setzen. Oder wir entwickeln mehrere Forms und je nach Aktivität wird ein bestimmter Form erzeugt und anschließend angezeigt.

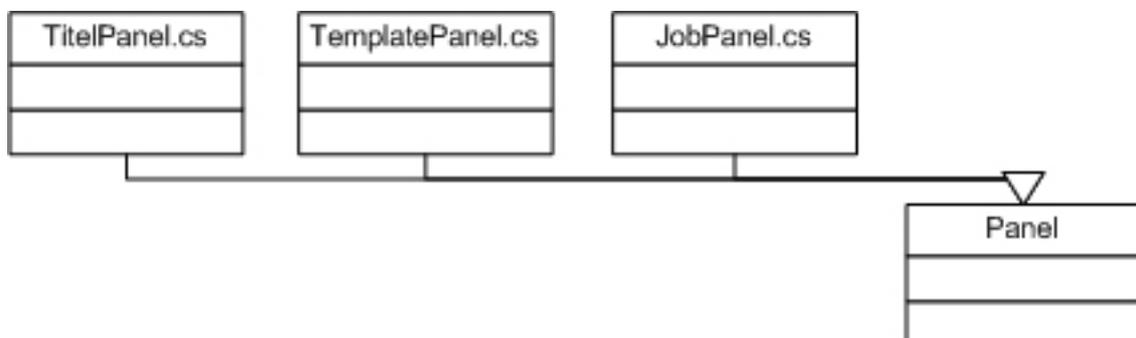
Wir entschieden uns für die ausgewogene Performance und realisierten die Benutzeroberfläche mit mehreren Forms, die zur Laufzeit erzeugt und dargestellt werden.

## Eigene GUI-Komponenten

In Smartphones und PocketPCs hat man nicht ein besonders großes Display, wo man viele Informationen darstellen kann.

Deshalb habe ich eigene Komponenten definiert, wo ich bestimmte Informationen kapseln kann. Diese Komponenten können vom Benutzer je nach Wunsch angeklickt werden, um die detaillierten Informationen zusehen.

Diese GUI-Komponenten sind von der Klasse Panel abgeleitet.



Design



# GUI-Elemente

## Verwaltung der GUI-Grafiken

Da wir die Grafiken nahezu in allen Forms verwenden, haben wir die GUI-Grafiken wie Logo usw. in einer Ressource-Datei abgelegt. Dadurch konnten wir sehr einfach mit den Grafiken im Programmcode umgehen.

Diese Ressource-Datei enthält alle JPEG-Grafiken.

## Grundlayout eines Forms



Alle GUI-Forms haben die gleiche Struktur, was für eine professionelle Anwendung sehr entscheidend ist. Dadurch haben die Benutzer eine bessere Übersicht über die Anwendung und können sich nach kurzer Zeit zurechtfinden.

# Verwaltung der GUI-Elemente

Alle GUI-Klassen außer die Windows Media Player-Komponente, in „Controls“ abgelegt, haben wir in der Klassenbibliothek „MobileClipgenerator“ abgelegt.

Die GUIHandler Klasse enthält je eine Instanz der Form-Klassen. Sie dient als die zentrale Klasse die nach Benutzerinteraktion eine bestimmte Form auf dem Display darstellt.

Die Klasse Programm beinhaltet die Main-Methode und dabei erzeugt es eine Instanz der Klasse GUI-Handler.



Um längere Ladezeiten bei Start der Anwendung zu vermeiden, werden die einzelnen Forms zur Laufzeit erzeugt und dargestellt.

Design



# Kommunikation mit dem VideoServer

## SOAP

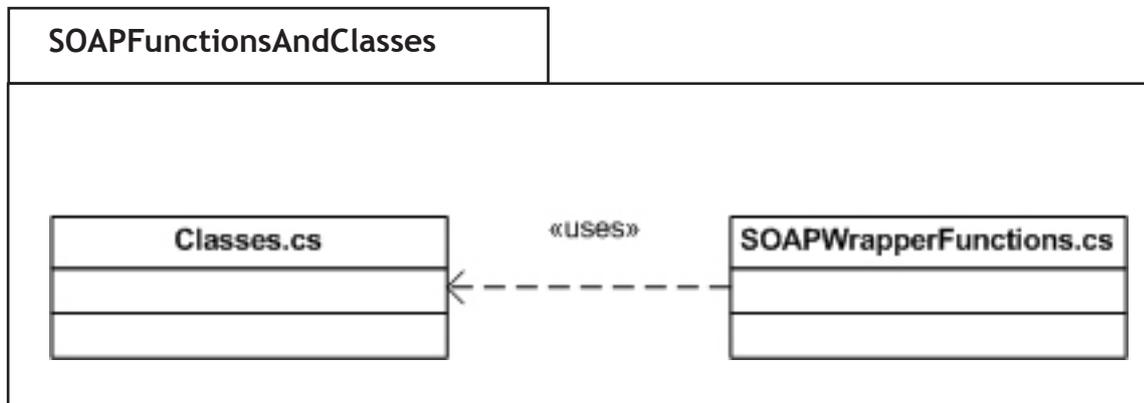
Die Kommunikation zum VideoServer erfolgt durch die SOAP Schnittstelle. Deshalb hat Markus SOAP-Wrapper-Funktionen implementiert, die wir je nach Aktivität , z.B. Bilderhochladen, aufrufen können und bestimmte Informationen zu bekommen oder zum VideoServer zu schicken.

Zusätzlich mussten wir Klassen definieren, damit wir Objekte der Informationen vom und zum Server erzeugen und übersichtlicher im Programm verwalten können.

## Verwaltung der SOAP-Wrapperfunktionen und Klassen

Alle SOAP-Wrapperfunktionen und Klassen für die Objekte zur Kommunikation mit dem Server sind in der Projektmappe „SOAPFunctionsAndClasses“ enthalten.

Die Klasse „Classes.cs“ beinhaltet alle Klassen für SOAP und die Klasse „SOAPWrapperFuntions.cs“ beinhaltet wie der Name schon sagt alle SOAP Wrapperfunktionen.



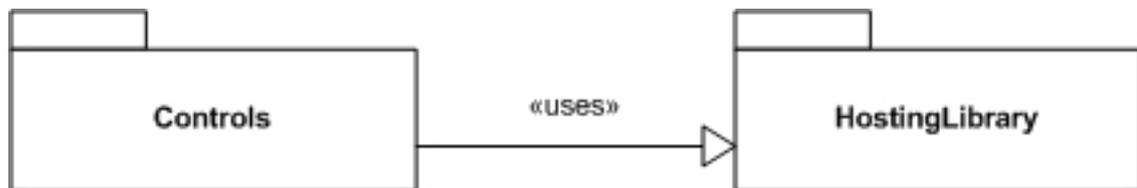
# Windows Media Player-Komponente

## Einbindung von Windows Media Player

Markus hat durch COM-Objekte (ActiveX) versucht die Windows Media Player des Smartphones aufzurufen. Da es leider sehr aufwendig ist und wir nicht viel Zeit zur Verfügung hatten, hat er sich für eine öffentliche Klassenbibliothek entschieden, mit deren Hilfe man die Windows Media Player in die Anwendung integrieren kann.

Diesen frei lizenzierten Klassenbibliothek findet man unter [OpenNETCF.org](http://OpenNETCF.org). Man kann es natürlich für die Anwendungen verwenden aber man hat keine Patentrechte auf diese Komponente.

Die Komponente besteht aus 2 Klassenbibliotheken.



Die Klassenbibliothek Controls greift auf die Klassen der HostingLibrary zu. Diese enthält die Funktionen zum Aufruf der Windows Media Player. Controls enthält die dazugehörige GUI-Komponente „AxWMPLib.cs“, die wir in unser GUI eingefügt haben um den Windows Media Player in der Anwendung zu verwenden.

## Externe Datenhaltung in der Anwendung

Die Benutzerdaten müssen extern abgelegt werden, da diese den Benutzer für die Kommunikation mit dem Server identifizieren. Wir haben uns dafür entschieden, diese Daten in Form einer XML-Datei in Anwendungsordner auf der Endgerät abzulegen.

Diese Funktion haben wir in der Klassenbibliothek mit der Klasse „ConfigXML.cs“ implementiert. Sie legt die angegebenen Benutzerdaten (zuerst statisch abgelegt), in der config.xml Datei in der Anwendungsordner „MobileClipgenerator“ ab.

Es ist auch möglich Daten, die möglicherweise später durch neue Features dazukommen in dieser XML-Datei abzulegen.

Technik



# Integration

## Einbindung der SOAP-Funktionen in die GUI

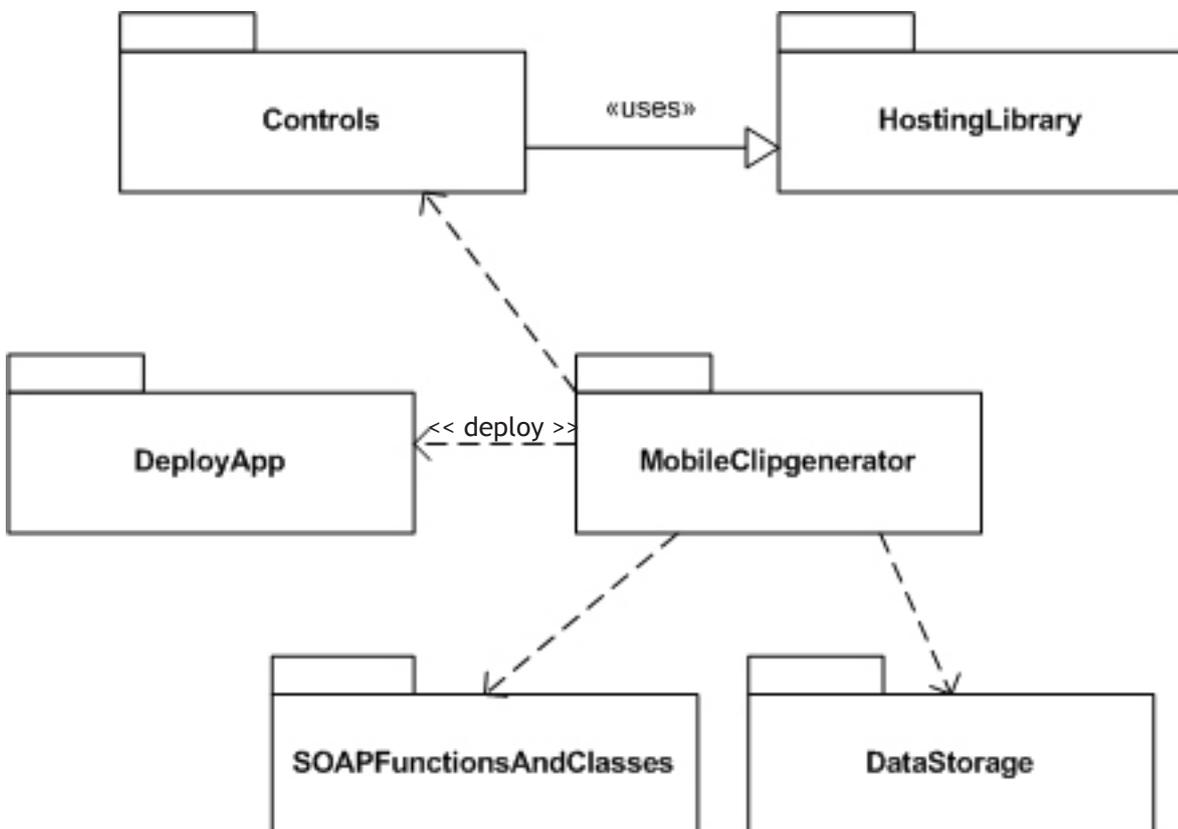
Die Einbindung der SOAP-Funktionen in die GUI war nicht schwer. Dank der Struktur der GUI-Klassen hat man die Funktionen in die vordefinierten Eventhandler eingebunden.

Dafür haben wir uns im Team zusammengesetzt und die Integration der SOAP-Funktionen zusammen erstellt.

## Paketdiagramm

Hier sehen wir nun das Paketdiagramm. Bisher haben wir die detaillierten Informationen über die von uns entwickelten Klassenbibliotheken beschrieben.

Das besondere hier ist der Package „DeployApp“. Der beinhaltet die Konfigurationseigenschaften. Dadurch konnten wir unser Produkt in Form einer CAB-Datei bereitstellen.



# Testing

## Testen der implementierten Programmcode

Während der Entwicklung der Anwendung waren wir für die entwickelten Komponenten selbst verantwortlich. Damit die Anwendung ohne Probleme läuft, war es wichtig, dass alle Komponenten einwandfrei funktionieren.

Um verdeckte Fehler aufzudecken, haben wir während der Entwicklung unsere Komponenten selbst getestet.

Dadurch war es möglich, dass wir eine einwandfrei funktionierende Anwendung zu entwickeln.

## Testen der SOAP-Funktionen

Markus hat die SOAP-Funktionen über seinen PC schon vorab getestet. Dabei hat er festgestellt, dass der Server manchmal auch bei Anfragen mit falschen Parametern mit „OK“ bestätigt.

Durch die Absprache mit Herrn Meyborg erst haben wir rausbekommen, dass ein Paar Funktionen Serverseitig noch nicht endgültig unterstützt werden.

## Usability Test

Die Usability der Anwendung haben wir von möglichen Nutzern testen lassen, die sowohl Kenntnisse mit Smartphones und Usability als auch keine Kenntnisse in diesen Gebieten besitzen.

Die meisten Leute haben sich beim ersten Versuch nicht zu sehr verwirren lassen. Durch einige extra Klicks hat man die Funktionsweise der Komponenten sehr schnell verstanden.

Das verwirrendste war die Form „Vorgehensweise“, wo die Reihenfolge der zu ablaufenden Aufgaben beschrieben sind. Alle dachten zuerst an Buttons. Doch nach ein Paar Klicks hat man die Funktion der Form bemerkt.

 Testing



# Zukunftsaussichten

## Entwicklung für Symbian & Java - Plattform

Um den Service Clipgenerator auf die mobilen Plattformen zu übertragen ist der erste Schritt durch den erfolgreichen Abschluss unseres Projektes getan. Als nächstes wird man die Software in andere Plattformen wie Symbian oder Java zu übertragen wollen. Besonders wäre eine Implementierung auf den Java-Plattform ein sehr großer Schritt.

## Mögliche Einbindung einer Vorschau

Da wir aus Kostengründen auf die Vorschau-Funktionalität verzichtet haben, wäre es als nächstes zu überlegen, ob man dem Benutzer eine Vorschau (möglicherweise in Flash) zur Verfügung zu stellen.

## Videsequenzen anstelle von Bildern

Für die Webapplikation besteht die Möglichkeit Videos anstelle von Bildern in Musikvideos zu integrieren. Diese Funktion könnte man auch in die Software mit berücksichtigen.

## Integrierter Email-Versand der Clips

Da man die erstellten Musikvideos gerne auch mit Freunden teilen möchte, könnte man auch eine Email-Funktion implementieren, damit man die Musikvideos direkt an Freunde weiterschicken kann.

## Integrierte Bluetooth-Übertragungsfunktionen

Die Trivid-Verantwortlichen würden in der Software eine Funktion zur Verfügung haben, mit der man aus dem Programm heraus die Musikvideos per Bluetooth übertragen kann.

## Bezahl- und Abrechnungssystem

Um die Software später zu vermarkten muss man ein Abrechnungssystem für die Software implementiert werden. Da die Software bisher als eine Art „Prototyp“, war, konnte man die Musikvideos kostenlos erstellen.



# Lessons Learned

## Erkenntnisse

- Gute Dokumentation von Quellcode ist sehr wichtig
- Gute Kombination von Designer und Programmierer zum Erfolg.
- Trennung von Grund- und „Nice-To-Have“-Funktionen

## Problemstellen

- Einbindung von ActiveX
- Datenhaltung im Programm
- Erwartungshaltung des Auftraggebers
- Zeit

## Fazit

- Vorsichtig mit zusätzlichen Anforderungen, die nach Projektbeginn geäußert werden
- Absprachen sehr wichtig
- Würden wir das Projekt nochmal machen?
  - Ja!







# Projekt Clipgenerator Mobile

Ein Projekt im Rahmen der **Hochschule der Medien**  
Studiengang Medieninformatik WS06/07

Hiermit wollen wir uns bei Prof. Dr. Johannes Maucher und Thomas Suchy für die Unterstützung bedanken.

Zudem bedanken wir uns bei Frau Hänsler und Herrn Meybory für die tolle Zusammenarbeit.

Projektteam:    Ferhat Tatar                      Markus Heimann



**CLIPGENERATOR  
MOBILE**

