|  |
| --- |
| Menschenwerk |
| Konzept für Projekt „Mobile Social Hub“ |
| Eine Applikation für Nokia S60-Geräte, als Schnittstelle zu wichtigen Social Media Plattformen |

|  |
| --- |
| [www.menschenwerk.net](http://www.menschenwerk.net)  13.09.2008 |

Inhaltsverzeichnis

[Projektvorstellung 3](#_Toc219031244)

[Kurzkonzept 3](#_Toc219031245)

[Problemstellung 3](#_Toc219031246)

[Beispiele 3](#_Toc219031247)

[Schlussfolgerung 3](#_Toc219031248)

[Claims 3](#_Toc219031249)

[Technik 3](#_Toc219031250)

[Client-/Server-Architektur 3](#_Toc219031251)

[Vorteile einer Client-/Server-Architektur: 4](#_Toc219031252)

[Nachteile: 4](#_Toc219031253)

[Sicherheit 4](#_Toc219031254)

[Komponenten und Ablauf 4](#_Toc219031255)

[Grundsätzliche Bedienung 4](#_Toc219031256)

[Dienstbasierte Nutzung 4](#_Toc219031257)

[Aktionsbasierte Nutzung 4](#_Toc219031258)

[Komponenten 4](#_Toc219031259)

[Dienste 5](#_Toc219031260)

[Twitter 5](#_Toc219031261)

[Twitpic 5](#_Toc219031262)

[Flickr 5](#_Toc219031263)

[Mögliche weitere Dienste 5](#_Toc219031264)

[Implementation 5](#_Toc219031265)

[Namensvorschläge 5](#_Toc219031266)

# Projektvorstellung

## Kurzkonzept

Ziel ist eine Anwendung für Nokia S60-Geräte (3rd Edition), welche als Schnittstelle zu wichtigen Social Media-Plattformen dient. Mit dieser Anwendung soll es möglich sein mit dem Handy schnell eigene Beiträge zu Twitter, Twitpic und Flicker zu senden, sowie die aktuellen Feeds dieser Dienste abzuholen und anzuzeigen. Weitere Dienste können (und sollen) in Zukunft folgen.

## Problemstellung

Es gibt keine wirklich guten Social Media-Anwendungen für Nokia S60-Geräte. Die existierenden sind entweder zu beschränkt, laufen nicht nativ auf S60 oder zielen in eine andere Richtung ab.

### Beispiele

**ShoZu** integriert sich gut in das System, ist jedoch sehr verwirrend. Es ist leicht versehentlich Inhalte zu posten, zudem sind die Möglichkeiten mit den einzelnen Diensten sehr beschränkt. Es ist beispielsweise nicht möglich Bilder auf Flickr umzubenennen oder den Beschreibungstext zu ändern. Es werden nur die letzten 10 Einträge der Twitter-Timeline angezeigt, was bei vielen Follows ungenügend ist.

**Twibble** ist grundsätzlich eine funktionierende Lösung, allerdings ist die Anwendung Java-basiert. Damit ist die gesamte GUI entgegen der Styleguides – es bedient sich einfach anders, als die üblichen S60-Anwendungen, die ein Nokia-Benutzer gewohnt ist.

**IYOUIT** bietet viel, versteht sich aber als eigenständige Plattform. Es geht nicht darum andere Dienste mit eigenen Inhalten zu füllen, sondern IYOUIT sendet seine Informationen weitgehend selbstständig an die Plattformen.

### Schlussfolgerung

Da ist noch eine Lücke für eine native S60-Anwendung, die sich darauf konzentriert ein Hub für die wichtigsten Social Media-Anwendungen zu sein. Es geht nicht darum eine eigene Community aufzubauen, sondern nur vorhandene und beliebte Plattformen zu bedienen.

## Claims

* *Applikation* ist ein Social Media Hub für Ihr Nokia S60 Mobiltelefon.
* Mit *Applikation* können sie Ihre Social Media-Welt bequem von Ihrem Nokia S60 Mobiltelefon aus verwalten.

# Technik

Die Anwendung wird auf dem Mobiltelefon mit Hilfe von *Python for S60* entwickelt. Die entsprechenden Versionen werden als SIS-Pakete verteilt.

## Client-/Server-Architektur

Die Anwendung verwendet eine Client-/ Server-Architektur. Ankommende Daten holt sich das Handy selbstständig und verarbeitet diese direkt. Zu versendende Daten werden über einen eigenen Webserver versendet. Dieser nimmt die zu versendenden Daten auf und verteilt sie an die entsprechenden Dienste.

### Vorteile einer Client-/Server-Architektur:

* Ein Inhalt muss nur einmal versendet werden, obwohl er eventuell an mehrere Dienste gesendet wird (Beispiel: Twitter und Twitpic).
* Neue Dienste lassen sich leicht auf Serverseite implementieren, ohne jedes Protokoll umfangreich auf dem Client zu implementieren. Auf diese Weise bleibt der Client schlank.
* Es findet eine Kontrolle statt, wer wie viel sendet und ermöglicht dadurch ein eventuelles Finanzierungssystem (nur bestimmte Anzahl an Updates/Tag kostenfrei).
* Der Benutzer verliert nach Verlust seines Mobiltelefons nicht seine Accounts und Einstellungen.

### Nachteile:

* Es muss zusätzlich eine Server-Applikation entwickelt werden.
* Alle Benutzer nutzen diese Server-Lösung (=Flaschenhals im Gegensatz zu den verteilten Clients).
* Wenn der Webserver down ist funktionieren alle Clients nicht mehr bzw. nur noch eingeschränkt.
* Falls wirklich Inhalte direkt und von der Webseite hochgeladen werden können, muss der Benutzer eventuell die Anwendung mehrfach autorisieren.

## Sicherheit

Der Benutzer erstellt sich einen Account mit Nutzername und Passwort, an den noch zusätzlich die ID des Mobiltelefons gebunden wird.

# Komponenten und Ablauf

## Grundsätzliche Bedienung

Die Anwendung kann auf zwei Arten benutzt werden: dienstbasiert oder aktionsbasiert.

### Dienstbasierte Nutzung

Der Benutzer kann direkt in den Bereich für einen speziellen Dienst navigieren (z.B. Flickr). Dort kann er dienstspezifische Aufgaben wahrnehmen (Bilder hochladen, kommentieren etc.). Seine Aktionen beschränken sich auf diesen Dienst (Bilder werden nur nach Flickr gesendet).

### Aktionsbasierte Nutzung

Der Nutzer wählt im Aktionsbereich eine Aktion aus (z.B. „Bild hochladen“) und wählt anschließend die Dienste aus, auf die sich die Aufgabe bezieht (Flickr, Twitpic).

## Komponenten

Core App:

Aktionsmenü

Dienstbereich/Dienstbereich/Dienstbereich

Webserver:

Dispatcher

Accountverwaltunig

Dienste:

## Dienste

Dienste sind die Social Media-Plattformen, mit denen die Applikation kommunizieren kann. Grundsätzlich sollte die Lösung so offen gestaltet sein, dass jederzeit neue Plattformen hinzukommen können, allerdings sollte sie sich zunächst auf die wirklich wichtigen (sprich: Plattformen mit hohen Nutzungsraten und Beliebtheit) konzentrieren.

Die Applikation sollte mit folgenden Diensten starten:

### Twitter

* Tweets schreiben
* Zeitleiste abrufen
* Nutzerstreams anzeigen
* Nutzern folgen/nicht mehr folgen

### Twitpic

* Bilder hochladen, mit Titel
* Bilder löschen
* Bilder kommentieren

### Flickr

* Bilder hochladen, mit Titel und Beschreibung
* Eigene Bilder/Fotostream ansehen
* Hochgeladene Bilder nachträglich benennen, beschreiben, taggen und löschen
* Bilder von Freunden ansehen
* Bilder kommentieren

## Mögliche weitere Dienste

* Facebook (Nachrichten anzeigen, Status ändern, Bilder hochladen)
* Brightkite (Check in, Freundesliste, Nachrichten, Placemarks)
* Goodreads (Bücher hinzufügen, taggen, bewerten, einordnen)
* Qype (Orte suchen, eintragen, bewerten und taggen)

# Implementation

Da Twitter von den 3 Startdiensten die einfachste API besitzt sollte damit begonnen werden.

Twitpic hat in dem Sinne noch nicht einmal eine API, sondern wird über Mails angesprochen.

Anschließend kommen die komplexeren Social Media Plattformen dran.

# Namensvorschläge

Das Baby braucht einen Namen. Wie wäre es mit:

* Pocketkun
* Social60 (so60)
* Mobilekun
* Communikun
* Gokun
* Mobling
* Pocketster
* Mobiliter
* Walkker
* Walkkun
* Dispattch
* Stutter
* Jitter
* Sendkun
* Hubster
* Hubkun
* Kuncast
* Castkun
* Twitcast

# Finanzierungsvorschläge

Hier gesammelt Vorschläge zur Finanzierung des Ganzen.

Nachteile einer kommerziellen Lösung sind die API Keys mancher Plattformen. Beispielsweise Flickr vergibt kommerzielle API Keys nur nach Prüfung. Als „kommerziell“ wird in diesem Zusammenhang schon eine Werbefläche auf der Webseite bezeichnet oder „Spenden“-Buttons von PayPal.

## Werbefinanziert

Die Anwendung könnte über Werbung auf der Webseite finanziert werden. Da es sich jedoch um eine hauptsächlich mobile Anwendung handelt, wird die Webseite nur zum Anmelden bzw. Herunterladen des Clients genutzt. Die Reichweite wäre also eher gering.

Eine weitere Möglichkeit wäre Werbung direkt beim Starten der Anwendung auf dem Handy. Dadurch würden dem Benutzer jedoch zusätzliche Kosten entstehen – er würde sogar noch für das Herunterladen der Werbung bezahlen, um sie sich anzeigen zu lassen. Angesichts der aktuell noch vorherrschenden Preise für mobilen Traffic ist das nicht sehr kundenfreundlich und fällt raus.

## Spenden

Kann eigentlich eine Anwendung von Spenden leben? Also, außer vielleicht Wikipedia und selbst die nur mit Müh und Not. Solche Spendenkonzepte sind vielleicht nutzerfreundlich, aber wirklich finanzieren tun sie eine Entwicklung nicht.

## Premium-Accounts

Ohne Premium-Accounts sind manche Funktionalitäten nur eingeschränkt verfügbar. Es ist zum Beispiel denkbar die aktionsbasierte Benutzung nur für Premium-Kunden zugänglich zu machen. Nutzer können über die kostenfreie Version Nachrichten und Bilder einzeln an die Dienste hochladen oder eben den Webservice als bequemen Verteiler nutzen.

Als Teaser eventuell die ersten 2 Wochen nach Anmeldung den Webservice kostenlos anbieten.