

Lastenheft

Teil der Software Engineering II Studienarbeit WS 2011/2012, Inf 3

Christopher Pahl,
Christoph Piechula,
Eduard Schneider,
und Marc Tigges

30. November 2011

Inhaltsverzeichnis

1	Produkteinsatz	3
2	Produktfunktionen	4
2.1	Benutzerfunktionen	4
2.1.1	Starten und Beenden	4
2.1.2	Persönliche Daten	4
2.1.3	Persönliches Profil	4
2.1.4	Persönliche Datenbank	4
2.1.5	Kommunikation (Chat)	5
2.1.6	Suchen	5
2.2	Administrator-Funktionen	5
3	Produktdaten	6
4	Produktleistungen	7
5	Qualitätsanforderungen	8
6	Ergänzungen	9
6.1	Realisierung	9
6.2	Die nächste Version	9

1 Produkteinsatz

Für welche Anwendungsbereiche und Zielgruppe ist die Software vorgesehen?

Die Software soll überall da eingesetzt werden, wo Musik abgespielt werden soll. Dabei ist man nicht auf einen Rechner beschränkt, auch Fernseher und Musik-Spieler mit Internetzugang und entsprechender Softwareunterstützung können theoretisch ein solches Programm verwenden.

Hauptsächlich soll sich diese Software allerdings an Nutzer eines Rechners mit einem Unix-artigen System richten. Des weiteren soll die Zielgruppe vorerst auf Benutzer, die der englischen Sprache mächtig sind beschränkt werden.

2 Produktfunktionen

Welche sind die Hauptfunktionen aus Sicht des Auftraggebers?

2.1 Benutzerfunktionen

Beim ersten Start des Systems soll eine Standard-Konfiguration geladen werden und die Verbindungseinstellungen zu einem MPD-Server müssen vorgenommen werden. Bei jedem weiteren Start soll die Konfiguration geladen werden, die vom Benutzer erstellt wurde, falls diese denn lokal gefunden werden kann. Der Benutzer soll sämtliche Einstellungen selbstverständlich zu jeder Zeit ändern können.

2.1.1 Starten und Beenden

- F_0010 Der Benutzer kann das System zu jedem Zeitpunkt Starten.
- F_0020 Der Benutzer kann das System zu jedem Zeitpunkt beenden.

2.1.2 Persönliche Daten

Ein Benutzer verfügt über ein persönliches Passwort und sowie einer persönlichen Verbindungseinstellung zum gewünschten MPD-Server. Diese Daten können von dem Benutzer zu jeder Zeit angepasst werden, selbstverständlich nachdem das Passwort eingegeben wurde.

- F_0110 Der Benutzer kann sich zu jeder Zeit seine Verbindungsdaten anzeigen lassen.
- F_0120 Der Benutzer kann zu jeder Zeit seine persönlichen Daten anpassen.

2.1.3 Persönliches Profil

Da die Software auf Unix-artige Systeme beschränkt werden soll, geht ein angenehmer Vorteil mit einher, nämlich dass eine Profil-Verwaltung seitens des MPD-Clients nicht implementiert werden muss. Die verschiedenen Profile werden durch die verschiedenen Profile des gesamten Betriebssystems definiert und differenziert.

2.1.4 Persönliche Datenbank

Eine persönliche Datenbank soll lokal nicht vorhanden sein. Die Datenbank des Benutzers befindet sich auf dem MPD-Server. Einzig und alleine modulare Erweiterungen des MPD-Clients können lokale Datenbank-Implementierungen erfordern.

2.1.5 Kommunikation (Chat)

Kommunikation von MPD-Client zu MPD-Client kann theoretisch implementiert werden, eine solche Schnittstelle ist vorhanden. Allerdings soll hierauf verzichtet werden, da im Vordergrund das Abspielen und Verwalten von Musik steht und es deutlich einfachere und bessere Systeme gibt, mit Hilfe derer man kommunizieren kann.

2.1.6 Suchen

Eine einfache Textsuche zum finden von Titeln, Alben oder Interpreten innerhalb der Abspiel-listen soll implementiert werden.

- F_0210 Der Benutzer kann nach Titeln suchen.
- F_0220 Der Benutzer kann nach Alben suchen.
- F_0230 Der Benutzer kann nach Interpreten suchen.

2.2 Administrator-Funktionen

Durch das Unix-artige System soll auch der Administrator-Zugriff geregelt werden. Sobald sich der Benutzer im Unix System als Administrator befindet, kann er auch den MPD-Client administrieren. Ein zusätzlicher Administrator-Modus muss also nicht implementiert werden.

3 Produktdaten

Welche Daten sollen persistent gespeichert werden?

Die vom Benutzer vorgenommenen Verbindungseinstellungen und Client spezifischen Einstellungen, sollen auf dem Rechner lokal und persistent gespeichert werden. Nur so kann ermöglicht werden, dass nach jedem Start des Systems diese Einstellungen geladen und übernommen werden können.

Außerdem soll eine Log-Datei auf den einzelnen Rechnern angelegt werden, die dieses System verwenden. In dieser Log-Datei werden Nachrichten des Systems gespeichert, um eventuelle Fehler leicht finden und beheben zu können. Es soll zusätzlich der Zustand des Systems abgespeichert werden, wenn das System beendet wird um das System beim nächsten Start in diesen Zustand versetzen zu können.

- D_0010 Persönlichen Verbindungseinstellungen.
 - Platzhalter
 - Platzhalter
- D_0020 Persönliches Passwort.
- D_0030 Client spezifische Einstellungen.
 - Platzhalter
 - Platzhalter
- D_0040 Eine Log-Datei.
 - Platzhalter
 - Platzhalter
- D_0050 Der Zustand.
 - Platzhalter
 - Platzhalter

4 Produktleistungen

Werden für bestimmte Funktionen besondere Ansprüche in Bezug auf Zeit, Datenumfang oder Genauigkeit gestellt?

Wenn das System beendet wird, soll der aktuelle Zustand des Systems gespeichert werden.

WAAAAHHHHH ?????????

Möglichst wenig ressourcen (Speicher, CPU und Netzwerk) Geschwindigkeit ist abhängig von server-lokation. Liedstand alle 500 ms

5 Qualitätsanforderungen

Welche qualitativen Anforderungen sind von besonderer Bedeutung?

Es soll auf folgende Priorität unter den Qualitätsanforderungen geachtet werden, dabei ist das erste Element das wichtigste und das letzte das unwichtigste.

Priorität 1: Robustheit

Priorität 2: Zuverlässigkeit

Priorität 3: Effizienz

Priorität 4: Intuitive Benutzung

Priorität 5: Design

6 Ergänzungen

6.1 Realisierung

Das System muss mit den Programmiersprachen C und/oder C++ realisiert werden. Dabei ist auf Objektorientierung zu achten, um Modularität und Wartbarkeit gewährleisten zu können.

6.2 Die nächste Version

Aufgrund des modularen Aufbaus kann das System beliebig oft und in verschiedene Richtungen weiterentwickelt werden.