

Dokumentation WBA2

Sportapp

Patrick Czajor 11082001 & Mark Steven Buder 11082179

23.06.2013

Dokumentation wba2

Sportapp - Autoren: Patrick Czajor & Mark Steven Buder

Inhaltsverzeichnis

Idee	2
Vorgehens Weise	2
Konzeptioneller Teil.....	2
Erstellen einer Idee.....	2
Informationsflussdiagramm	2
Fragestellungen	3
Kommunikation	3
Ressourcen-Modell.....	3
Die Applikation - Praktischer Teil	4
GUI.....	4
Vorwort	4
Erstellte User	4
Wie geht man mit der GUI um?	4
Geplante Ideen	5
Konflikte	6
Account.....	6
Server	6
Referenzieren	6
Spielerlisten & Blacklisten Problem.....	6
Anzeigen von allen Orten/Events in der GUI.....	6

Idee

Eine Applikation welche die Planung und Gestaltung von Hobbysportlern vereinfacht. Der Zweck ist es auf einfachem Weg Spieler/Gruppen zu finden und organisieren. Es besteht die Möglichkeit Vereinsplätze zu mieten. Das Mieten eines Platzes/Feldes ist kostenpflichtig. Der Erlös kommt dem Verein zu gute. Gegen einen Aufpreis kann ein Schiedsrichter gebucht werden. diese Option ist besonders geeignet für kleinere Turniere, wie zum Beispiel interne Firmenturniere.

Vorgehens Weiße

Bei dem Vorgehen, wurden zuerst sämtliche konzeptionelle Aufgaben erledigt und dann anschließend wurde die Konzeption programmiert.

Konzeptioneller Teil

Erstellen einer Idee

Das Team erstellte mehrere mögliche Auslegungen einer Idee, bis die Tutoren diese für eine angemessene hielten.

Informationsflussdiagramm

Anschließend wurde ein Informationsflussdiagramm erstellt. Das Diagramm macht deutlich, welche Funktionen die Applikation braucht und welche Daten dabei von Bedeutung sind.

1. Event erstellen - Sämtliche notwendigen Informationen, die gebraucht werden, um ein Event zu erstellen, wird dem Server übermittelt und dieser sendet anschließend eine Bestätigung an den Client zurück.
2. Abonnement erhalten - Subscribed man z.B. Events in dem Ort Gummersbach, werden einem immer die neu erstellten Events in Gummersbach mitgeteilt.
3. Spielerliste eines Events bearbeiten - Der Client fordert vom Server die Spielerliste eines Events an, bearbeitet diese und schickt die neue Liste wieder zum Server zurück. Dies ist nur möglich, sofern man auch Admin, des Events ist.
4. Bearbeiten eines Events - Ähnlich wie "Spielerliste eines Events bearbeiten", nur dass nun alle Daten des angefragten Events übermittelt werden.
5. Verein mischt sich ein - Es ist dem Verein vorbehalten, die Events auf dem eigenen Sportplatz wegen diverser Gründe abzusagen. Dabei schickt der Verein an den Server, welcher Platz wann und warum nicht verfügbar ist. Der Server leitet diese Information an alle eingetragenen Spieler, sowie den Admin der der Veranstaltung weiter.
6. Vereinsbenachrichtigung - Nachdem eine Veranstaltung erstellt wurde bekommt der Verein dieses mitgeteilt. Einige Zeit, bevor die Veranstaltung stattfindet, wird der Verein noch einmal auf diesen Umstand hingewiesen.

Fragestellungen

Bei der Ausarbeitung dieser Idee stand das Team vor diversen Fragen, die bedacht werden müssen, aber für das Projekt nur organisatorischen Wert haben. Es müsste geklärt werden wie das Mieten eines Platzes abläuft bzw. wie und wo die Bezahlung abgewickelt werden soll. Des Weiteren ist die Frage der Finanzierung und Administration der Applikation zu klären. Es ist noch unklar, ob mehrere Events gleichzeitig auf einem Platz stattfinden können, also dass der Platz aufgeteilt wird. Eine Platzaufteilung wäre vorteilhaft wenn sich kleinere Gruppen anmelden, außerdem würde der Inhaber seine Einnahmen steigern.

Die Applikation arbeitet im Moment nur mit IDs für einzelne Events, es wäre aber auch vorstellbar mit Namen zu arbeiten, welchen der administrierende Spieler für das Event vergibt. Für andere Spieler wäre es somit einfacher Events zu finden.

Kommunikation

Die Datenkommunikation der Applikation verläuft sowohl synchron als auch asynchron.

Das Veröffentlichen von Events, das Bereitstellen von Orten, sowie die gezielte Suche nach einem Event verläuft dabei synchron.

Spieler erhalten asynchron Benachrichtigungen auf ihr mobiles Endgerät, wenn ein Event nach ihren persönlichen voreingestellten Filtern erstellt wurde.

Sollte ein gemieteter Platz aus irgendwelchen Gründen doch nicht mehr verfügbar sein, werden die Spieler via Push-Benachrichtigung auf diesen Umstand hingewiesen. Gleiches gilt für das Absagen oder das Bearbeiten von Events, welches nur der Administrator bzw. der Ersteller des Events kann.

Jeder Nutzer ist in der Lage Events und Orte zu veröffentlichen und zu abonnieren.

Ressourcen-Modell

Anfangs ging das Team von einer Ressourcen-Modellierung ohne Subressourcen aus. Es sollte möglich sein, nach allen Daten suchen zu können. Im Laufe der Ausarbeitung entschied man sich dagegen und modellierte die Ressourcen mit Subressourcen. Die Benutzer müssen nicht nach allen Informationen suchen. Es reicht aus, wenn sämtliche Events, Accounts und Orte dargestellt werden können.

Darüber hinaus kann man noch die Sportart als Ressource anlegen, da auch nach dieser gefiltert werden könnte. Da aber vorerst nur Fußball als Sportart modelliert wurde, musste die Ressource noch nicht angelegt werden.

Die Applikation - Praktischer Teil

GUI

Vorwort

Während beim Grizzlyserver alles Problemlos funktioniert, sind bei der GUI diverse Probleme aufgetaucht. Aus zeit technischen Gründen konnten diese von dem Team nicht mehr behoben werden konnten. Das Einloggen auf dem Openfire Server funktioniert, jedoch die einzelnen Funktionsaufrufe nicht. So passiert es zum Beispiel, dass sich die Applikation beim Veröffentlichen von einem Ort bzw. Event aufhängt.

Alle Methoden wurden an die passende Stelle der GUI geschrieben.

Erstellte User

Zum Testen der Applikation wurden 2 Benutzer auf dem Openfire Server erstellt.

Username	Passwort
user1	user
user2	user

Wie geht man mit der GUI um?

Bei der GUI handelt es sich um eine Plattform, welche die Funktionen testen soll. Diese entspricht nicht einem fertigen Design.

Zunächst gibt man Username und Passwort ein. Als Default ist user1 eingetragen. Da dieser am Server registriert ist, muss man nichts weiter eingeben.

Bei erfolgreicher Anmeldung, gelangt man ins Hauptmenü. Dort verfügt der Benutzer über mehrere Auswahlmöglichkeiten: "Publish", "Browse", "Subscribe", "Unsubscribe" und "Verändern". Auf Grund von mangelnder Zeit wurde die "Löschen" Funktion noch nicht eingebaut.

Publish

Bei einem Klick auf Publish Event bzw. Ort, erscheinen alle notwendigen Eingabefelder, um jenes zu veröffentlichen. Nach Eingabe aller erforderlichen Dateien bestätigt man die Veröffentlichung mit einem Klick auf den "Publish!"-Button. Die Veröffentlichung wird in die XML-Datei eingetragen und das dazugehörige LeafNode wird erstellt. Wenn alles erfolgreich war, erscheint in der Konsole der Kommentar "Publish accomplished".

Browse

Im "Browse"-Fenster kann man sich die eigenen Abonnements bzw. Subscriptions oder sämtliche Nodes anzeigen lassen. Bei einer Eingabe einer ID und einem Klick auf einen der beiden Schaltflächen "OrtID" oder "EventID" wird ein Ort/Event mit der angegebenen ID angezeigt, falls vorhanden.

Subscribe

In der Textarea werden alle vorhandenen Nodes angezeigt. Um einem Node zu abonnieren, muss der Nodename eingegeben und mit einem Klick auf "Subscribe!" Schaltfläche bestätigt werden.

Unsubscribe

In der Textarea werden alle eigenen Subscriptions angezeigt. Um ein Abonnement zu kündigen, muss der Nodename eingegeben und mit einem Klick auf "Unsubscribe!" Schaltfläche bestätigt werden.

Verändern

Das Verändern eines Events oder eines Ortes verläuft parallel zum Veröffentlichen. Einziger Unterschied ist die erforderliche Eingabe einer ID um das gewünschte Objekt eindeutig zu identifizieren.

Löschen - geplant

Eine weitere Funktion ist das Löschen von Orten bzw. das Absagen von Events. Diese Funktion wurde aus zeittechnischen Gründen nicht mehr implementiert.

Ähnlich wie bei der Verändern Option muss das Objekt mit Angabe einer ID eindeutig identifiziert werden.

Zurückknopf

Auf jeder Seite, außer der Login Seite und im Hauptmenü gibt es einen Zurückknopf. Mit diesem, gelangt man wieder ins Hauptmenü ohne die Applikation neu zu starten.

Geplante Ideen

Filtermöglichkeiten

Es sollte möglich sein, sämtliche erstellte Events nach Belieben zu filtern z.B. nach Ort, so dass alle Events in dem gewählten Ort angezeigt werden. Passend zu den Filtermöglichkeiten könnte man sich seine eigenen Abonnements zusammenstellen, um über jedes neue Event, nach seinen Vorlieben, informiert zu werden.

Geplante Filtermöglichkeiten:

- Sportarten
 - Fußball, Basketball, Tennis..
- Örtlichkeit
 - Ort, spezieller Platz
 - öffentlicher Platz - kostenlos, kostenpflichtiger Preis, Preis pro Kopf
- Datum & Uhrzeit
 - spezifischer Wochentag/Uhrzeit bzw. Zeitraum
- Spieler
 - Anzahl, Spielernamen

Konflikte

Account

Beim Implementieren des XMPP Clients kam die Fragestellung auf, ob es User vom Restservice und XMPP geben muss oder die User nur vom XMPP Server verwaltet werden können. Denn bei der Ressourcenmodellierung erstellte man eine Ressource für die Accounts, welche Anzeigenamen und Telefonnummern speichert. Deshalb musste eine Version mit beiden Systemen realisiert werden.

Server

Um XMPP und Rest zusammen zu führen, werden im XMPP Client auch die Methoden der Restressourcen aufgerufen. Es gibt auch die Möglichkeit den Restressourcen einen Trigger zu schreiben, der aktiv wird, sobald diese sich verändern.

Referenzieren

Als das Projekt schon weit vorangeschritten war, tauchte eine neue Perspektive auf. Anstelle bei Events alle Daten über Ort und Spieler zu übergeben, könnte man auch auf die Ressourcen über Links verweisen. Somit würde die zu übertragene Datenmenge gesenkt werden.

Da das Projekt sich aber schon in der Finalen Phase befand, wurde sich dagegen entschieden.

Spielerlisten & Blacklisten Problem

Die von JAXB generierte Klasse Events beinhaltet die Spielerliste bzw. eine Blacklist in denen Spieler abgespeichert werden, die nicht zu diesem Event zugelassen werden. Dabei tauchte folgendes Problem aus. Die Listen wurden keine Javalisten. Somit konnte nicht auf eine unbekannte Menge von Spielern zugegriffen werden. Auch die Tutoren konnten bei diesem Problem nicht helfen.

Anzeigen von allen Orten/Events in der GUI

Auf das Anzeigen von sämtlichen eingetragenen Orten und Events wurde in der GUI verzichtet. Man kann alle erstellten Nodes anzeigen lassen oder auch einzelne Events oder Orte über die jeweilige ID anzeigen.