



SEP – Wintersemester 2013/2014

Ticket-System

Lastenheft

Christopher Auer, Christian Bachmaier, Arne Bilzhause,
Martin Bochenek, Matthias Dangl, Camelia Rosenkranz

1 Motivation

Im Rahmen von größeren Events und Partys ist die Verwaltung der Eintrittskarten (Tickets) mit nicht unerheblichem Aufwand verbunden. Dies beginnt bereits im Vorfeld der Veranstaltungen, da Vorkaufsrechte und Reservierungen berücksichtigt werden müssen. Zusammen mit der Ausgabe von Tickets an der Abendkasse und einer zuverlässigen Einlasskontrolle müssen diese Mechanismen am Ende eine kontrollierbare Teilnehmerzahl gewährleisten. Nur so können gesetzliche Vorgaben eingehalten werden und aktiv Kostenkontrolle für die Veranstaltung betrieben werden.

2 Aufgabenstellung

Ziel des SEP im Wintersemester 2013 ist es, ein online Ticket-System zur leichten Verwaltung von Tickets für größere Veranstaltungen zu entwickeln. Veranstalter soll darin die Möglichkeit gegeben werden die Tickets für ein Event zu verwalten. Benutzern soll des weiteren die Möglichkeit gegeben werden Tickets per Sofortkauf oder per Reservierung zu erwerben. Das Ziel ist eine Web-Anwendung, deren Benutzung intuitiv

ist. Sie soll geringe Anwendungshürden aufweisen, zugleich aber den Betreibern etliche Features zur Vereinfachung bieten. Eine stets aktuelle Auswertung (Monitoring) der Ticketbestände zur Kostenkontrolle und ein zuverlässiges Bezahlungssystem sollen die zentralen Bestandteile der Anwendung sein.

Beispiele für bereits existierende Systeme sind

- <http://www.fusionticket.org/>
- <https://www.ztix.de/>
- <http://www.evention.eu/>
- <http://www.proticket.de/>
- <http://de.amiendo.com/ticketshop-ticketing-software.html/>

Die Anwendung muss mit Java 7 und JavaServer Faces (JSF) entwickelt werden. Auf Benutzerseite kann ein moderner Webbrowser vorausgesetzt werden.

3 Produkteinsatz

Die Zielgruppe besteht aus Veranstaltern von größeren Events und Partys aller Art. Die Anwendung muss über das Internet erreichbar sein.

4 Produktfunktionen

Veranstaltung Eine Veranstaltung wird im System in der Regel im Vorfeld einer realen Feier etc. angelegt. Eine Veranstaltung umfasst das gesamte Ticket-Kontingent, Zeitplanung, optische Erscheinung und weitere Eigenschaften die den Zustand der Veranstaltung innerhalb des Systems definieren. Es kann in einem System mehrere parallele Veranstaltungen geben.

Ticket Ein Ticket kann innerhalb der Veranstaltung verschiedene Ticketzustände haben. Das Ticket kann einem Käufer in verschiedenen Formen vorliegen: digital (z.B. auf Android-Gerät) oder ausgedruckt. Jedes Ticket hat eine Ticket-ID und wird u.a. mit Hilfe eines QR-Codes repräsentiert.

Kontingent Jede Veranstaltung hat ein gesamtes Ticket-Kontingent. Ein Kontingent kann in kleine Ticket-Pools für verschiedene Zwecke aufgeteilt werden.

Benutzergruppen

1. **System-Betreiber/System-Admin** Der System-Betreiber ist für die Installation und Konfiguration des Systems zuständig. Er hat volle Rechte im gesamten System (höchster Admin). Er kann das System customisieren/anpassen (z.B. optisch). Außerdem kann er andere Benutzerkonten zum Admin für das System oder einzelne Veranstaltung befördern.

2. **Veranstalter/Veranstaltungs-Admin/Admin** Der Veranstaltungs-Admin ist für Verwaltung einer oder mehrerer Veranstaltungen zuständig. Er kann eine Veranstaltung konfigurieren und ihre Erscheinung optisch anpassen.
3. **Registrierter Benutzer/Käufer** Ein registrierter Benutzer muss sich als solcher ausweisen können (Authentifizierungsverfahren). Er wird im System durch ein Benutzerkonto mit persönlichen Daten repräsentiert. Er hat ggf. die Möglichkeit Tickets zu reservieren. Er kann Tickets online oder auch offline bezahlen.
4. **Unregistrierter Benutzer/Käufer** Ein unregistrierter Benutzer hat keine persönlichen Daten im System. Er hat keine Möglichkeit Tickets zu reservieren. Er kann Tickets ohne Registrierung sofort per Kreditkarte oder offline bezahlen.
5. **Käufergruppen** Käufergruppen haben möglicherweise eigene reservierte Ticket-Pools und somit Vorkaufsrechte. Käufergruppen können im System durch mehrere zusammengefasste registrierte Benutzer oder durch einzelne Gruppen-Verantwortliche (z.B. Sekretariat für einen ganzen Lehrstuhl) dargestellt werden.

Ticket ID Die Ticket-ID wird zufällig generiert. Sie identifiziert im System eindeutig ein einzelnes Ticket. Sie dient u.a. zur Zuordnung, welche Tickets bereits bezahlt wurden, welche Tickets Einlass zu einer Veranstaltung erhalten haben, usw.

Verifizierungs-Code Der Verifizierungs-Code ist ein QR-Code, der auf dem gedruckten Ticket oder einer digitalen Darstellung auslesbar ist. Er beinhaltet mindestens die Ticket-ID.

Ticketzustände

1. **Reserviert** Reservierte Tickets sind noch nicht aktiv. Allerdings sind reservierte Tickets auch nicht mehr für andere Benutzer reservierbar. Sie können nur innerhalb eines gewissen Zeitraumes vom berechtigtem Käufer bezahlt werden. Ist eine Reservierung abgelaufen, wandern die vormals reservierten Tickets wieder in den ursprünglichen Ticket-Pool zurück.
2. **Inaktiv** Ein Ticket, das noch nicht bezahlt wurde, ist inaktiv.
3. **Aktiv** Ein bezahltes Ticket ist aktiv. Es kann zur Einlasserlaubnis genutzt werden.

Angemeldeter Benutzer Jeder Benutzer des Systems (Admin, registrierter Benutzer) der sich erfolgreich eingeloggt hat.

Anonymer Benutzer Alle anderen Besucher, die keine Login-Daten besitzen.

Monitoring Das Monitoring zu einer Veranstaltung ermöglicht es den Admins der Veranstaltung zu jeder Zeit Überblick zu Einnahmen, Kontingenten, Reservierungen, pot. noch zu erwartende Einnahmen, usw. zu erhalten.

Eine Veranstaltung umfasst das gesamte Ticket-Kontingent, sowie die Ticketzustände jeder einzelnen Ticket-ID. Eine Veranstaltung bietet mehrere Phasen (z.B. Vorverkauf, eigentliches Event). Ein Ticket kann für festgelegte Zeiträume reserviert werden. Bestimmte Nutzergruppen können gegebenenfalls Vorkaufsrechte erhalten. Wenn ein Ticket bezahlt wurde, wechselt der systeminterne Zustand der Ticket-ID auf aktiv. Es kann online per Kreditkarte oder bar an einer Zahlungsstelle, bzw. der Abendkasse bezahlt werden.

Der Veranstalter soll stets Zugriff auf die aktuellen Ergebnisse des Monitoring haben. Außerdem kann der Veranstalter nicht nur beim erstmaligem Setup, sondern auch während der Vorverkaufsphase stets Ticket-Kontingente, Preise, Zeitfristen usw. anpassen.

Somit ergeben sich mehrere Gruppen von Benutzern des Systems: System-Administratoren, Veranstaltungs-Administratoren, anonyme Nutzer (Käufer) und autorisierte Benutzer (registrierte Käufer). System-Administratoren sind für die Verwaltung und Konfiguration des gesamten Systems verantwortlich. Veranstaltungs-Administratoren sind für die Verwaltung und Konfiguration der einzelnen Veranstaltungen verantwortlich. Sowohl anonyme, als auch autorisierte registrierte Benutzer können freie Tickets aus dem Kontingent ausdrucken, um sie später zu bezahlen. Allerdings erhalten nur registrierte Benutzer die Möglichkeit einer Reservierung. Es kann auch Käufergruppen geben, die ebenfalls Rechte zur Reservierung haben. (Sie kaufen aus ihrem Pool, bis dieser leer ist, danach aus dem restlichen freien Kontingent.) Eine detaillierte Auswertung kann durch Administratoren erfolgen. Das System soll den reibungslosen Ticketverkauf, zuverlässige Ticketkontrolle und Planungssicherheit für die Veranstalter gewährleisten. Der Ticketverkauf kann hierbei auf verschiedene Arten erfolgen: mit oder ohne vorherige Reservierung, bar bezahlen an einer Zahlstelle, bar bezahlen an der Abendkasse, online per Kreditkarte. Die Kreditkarten Abwicklung erfolgt über die Infosun-Bank. Nach erfolgreicher Bezahlung wird ein Ticket aktiv. Die Ticket-Verifizierung erfolgt über das Prüfen des QR-Codes (oder manueller Eingabe, der im QR-Code enthaltenen Daten in das System). Der QR-Code kann auf einem Papier-Ticket oder digital auf einem Android-Gerät bereitgestellt werden.

Der Zugang zu dem System soll über das Internet soll möglich sein.

5 Produktfunktionen

- System-Admin

Veranstaltung erstellen Ein System-Administrator kann eine Veranstaltung erstellen, woraufhin das System eine ID zur Identifikation der Veranstaltung bereit stellt. Es müssen mehrere Veranstaltungen parallel möglich sein.

Veranstaltungen verwalten Der System-Admin kann den Veranstaltungen Verwaltungs-Admins zuweisen.

System customizen Das System kann optisch angepasst werden.

- Veranstaltungs-Admin

Veranstaltung konfigurieren Eine Veranstaltung hat einen festen Zeitpunkt. Zusätzlich hat eine Veranstaltung im Vorfeld verschiedene Phasen, die sich auf den Ticketverkauf auswirken (Phase für Vorkaufrecht, Reservierungen und Phase, in der alle restlichen Ticket frei verfügbar sind). Zu einer Veranstaltung gehört ein bestimmtes Ticket-Kontingent, das wiederum aufgeteilt (umgeschichtet) werden kann in verschiedene kleinere Ticket-Pools für Vorkaufrecht usw. Ein Veranstaltungs-Admin kann die Zahlungsart pro Veranstaltung oder pro Ticket-Kontingent konfigurieren.

Benachrichtigung Der Administrator einer Veranstaltung kann an die registrierten Benutzer einer Veranstaltung Nachrichten senden. Nachrichten können allgemeine Informationen über Ticket-Pools, aber auch benutzerspezifische Informationen zu Reservierungen sein. Außerdem soll das System bei bestimmten Aktionen der Nutzer automatische Nachrichten versenden, z.B. Zahlungsbestätigungen.

Adressbuchverwaltung Veranstaltungs-Admins können ein Adressbuch verwalten. Das Adressbuch dient der Kontaktierung verschiedener Benutzer. Es können Einzelkontakte und Gruppen angelegt werden. Das Adressbuch kann mindestens genutzt werden um benutzerspezifische oder gruppenspezifische Nachrichten (z.B. Reservierungsstatus) zu versenden.

Ticket-Verwaltung Die optische Erscheinung von ausdrucksbaren Tickets ist veränderbar. Tickets können auch innerhalb einer Veranstaltung verschiedene Preise haben (z.B. je nach Gruppe oder Zeitpunkt).

Ticket-Verifizierung Die Ticket-Verifizierung erfolgt über das Scannen des Ticket-QR-Codes oder durch manuelle Eingabe der Ticketdaten in eine spezielle Eingabemaske. Die Verifizierung soll nicht nur den Ticketzustand auf aktiv/inaktiv prüfen, sondern auch in Echtzeit mitverfolgen, ob ein Ticket bereits zuvor benutzt wurde (doppelte Nutzung verhindern). Als Folge lassen sich bei permanenter Ticket-Verifizierung, an den Ein-/Ausgängen einer Feier, stets die aktuellen Besucherzahlen ableiten.

Ticket-Export Man kann eine bestimmte Menge von aktiven oder inaktiven Tickets exportieren. Die exportierten Tickets können beispielsweise an eine Druckerei zum Ausdruck weitergegeben werden.

Monitoring Das Monitoring zu einer Veranstaltung ermöglicht es den Admins der Veranstaltung zu jeder Zeit Überblick zu Einnahmen, Ausgaben, Kontingenten, Reservierungen usw. zu erhalten.

- Registrierter Benutzer (angemeldet)

Ticket-Bezahlung Ein Ticket kann online per Kreditkarte oder bar bezahlt werden.

Kontaktverwaltung Ein angemeldeter Benutzer kann seine eigenen Kontaktdaten verwalten. Anhand der Kontaktdaten muss ein Benutzer identifizierbar sein. Außerdem muss sichergestellt sein, dass jeder Benutzer über die angegebenen Kontaktdaten zu erreichen ist.

Ein Administrator kann auf alle Kontakte einer Veranstaltung zugreifen.
Ein normaler Benutzer nur auf seinen eigenen.

Abmeldung Ein angemeldeter Benutzer kann sich abmelden.

- Anonymer Benutzer

Registrierung Ein anonymer Benutzer kann sich ein Benutzerkonto anlegen.

Anmeldung Ein Benutzer kann sich in ein Benutzerkonto einloggen und ist damit angemeldet. Hierfür ist eine Authentifizierung notwendig.

Online-Hilfe und FAQ Jeder Benutzer muss zu jedem Zeitpunkt schnellen Zugriff auf die für die aktuelle Seite relevante Online-Hilfe und die evtl. vorhandene FAQ haben.

Ticketkauf Der anonyme Benutzer kann sich durch Eingabe der notwendigen Daten (aber ohne zwingende Registrierung) ein Ticket kaufen.

6 Nicht-Funktionale Anforderungen

Folgend die Produktleistungen und Qualitätsanforderungen.

6.1 Usability

1. Einfache und intuitive Bedienbarkeit des Systems.
2. Die Benutzeroberfläche soll sich an von Benutzer gewohnten Bedienkonzepten und der damit verbundenen Funktionen orientieren; alle Tabellen sollen z.B. nach den dargestellten Spalten sortierbar sein; Tabellen die eine gewisse Größe überschreiten sollen um kurze Seitenladezeiten zu ermöglichen auf mehrere Seiten aufgeteilt werden.
3. Häufig wiederkehrende Aufgaben sind durch das System möglichst benutzerfreundlich zu unterstützen; häufig genutzte Funktionen sind möglichst einfach zugänglich zu machen.
4. Daten sollen nicht nur leicht auffindbar und gut lesbar sein, sondern auch leicht eingebbar sein.
5. Bei Fehleingaben in ein HTML-Formular und der darauffolgenden Korrektur sollen die zuvor eingetragenen Felder nicht erneut einzugeben sein, sondern schon vorbesetzt sein. Außerdem sollte die Überprüfung der Eingaben nicht nach der ersten fehlerhaften Eingabe abbrechen, sondern alle Eingaben überprüfen und eine akkumulierte Fehlermeldung an den Benutzer zurückgeben.

6.2 Datensicherheit

1. Alle im System erfassten Daten sind persistent in einer Datenbank abzulegen; die Konsistenz der Daten ist sicherzustellen (Mehrbenutzerbetrieb!). Speziell wenn Änderungen über mehrere Tabellen hinweg vorgenommen werden sind Transaktionen zu nutzen.
2. Für die persistente Speicherung der Daten soll eine Datenbank (PostgreSQL) verwendet werden, die auf den Infosun-Rechnern zur Verfügung gestellt wird. Als Referenzplattform dienen die Rechner im CIP-Pool.
3. Beim Löschen von Daten sind die Abhängigkeiten zwischen Daten einzelner Tabellen zu berücksichtigen; hat das Löschen eines Satzes das Löschen anderer Sätze zur Folge muss der Benutzer vorher deutlich darauf hingewiesen werden!

6.3 Datenschutz

1. Es muss sichergestellt werden, dass durch das System zu keinem Zeitpunkt sensible Daten für unberechtigte Dritte zugänglich gemacht werden.
2. Es sollen möglichst wenige technische Informationen über das System nach außen gegeben werden.
3. Alle personenbezogenen Daten wie z.B. Logindaten sind sensibel und dürfen daher nur per SSL-Verbindung übertragen werden.
4. Passwörter dürfen nicht im Klartext gespeichert werden.
5. Die Nutzerdaten sind so zu speichern, dass kein unauthorisierter Zugriff durch Dritte oder durch andere Nutzer im System stattfinden kann.
6. Änderungen durch Manipulationen mit bekannten Angriffsmethoden wie SQL-Injection oder Cross-Site-Scripting müssen ausgeschlossen werden. Es müssen außerdem Maßnahmen ergriffen werden, um die Sitzungen der einzelnen Nutzer zu schützen (Schutz gegen Session-Hijacking).

6.4 Internationalisierbarkeit

1. Für die Texte auf der Website ist eine Zeichenkodierung UTF-8 zu wählen.
2. Die Sprache des Systems kann deutsch oder englisch (bevorzugt) sein; eine mehrsprachige Implementierung ist optional.

6.5 Entwickelbarkeit

1. Das zu erstellende System soll flexibel für zukünftige Erweiterungen sein. Eine einfache und kostengünstige Weiterentwickelbarkeit des Systems ist sicherzustellen.

6.6 Installation

Es soll eine komfortable Installation für Systembetreiber bereitgestellt werden. Es sollte eine einfache und schnelle Installation bieten und automatisch das entsprechende Datenbanksetup vornehmen.

7 Ergänzungen

Die Benutzung des Systems sollte mit allen gängigen Web-Browsern möglich sein. Wir raten allerdings davon ab, für jeden Browsertyp unterschiedlichen HTML-Code zu generieren. Der HTML-Code soll logisches Markup darstellen und nicht dazu missbraucht werden, eine bestimmte graphische Darstellung zu erzwingen. Deswegen dürfen Features wie Frames nicht und explizites Javascript nur nach Absprache eingesetzt werden. Der HTML-Code muss HTML-konform sein und z.B. durch <http://validator.w3.org/> validierbar sein. Die Verwendung von Cascading Stylesheets (CSS) wird dringend angeraten.

Die Sessionverwaltung darf die Verwendung von Cookies nicht erzwingen.

Die Servlets müssen ein Log über alle Fehler führen, um das Debugging der Anwendung zu vereinfachen. Achten Sie darauf, dass die Fehlerbeschreibungen detailliert genug sind, um auf einen Fehler bzw. dessen Ursache schließen zu können.