

# Lastenheft

## hg17b - Android App für Weiterbildungsmanagement

M. Herhold, C. Pfeiffer, N. Kortenbruck, J. Dietz, R. Mueller, M. Rudolph  
3. Januar 2018

### Inhaltsverzeichnis

<b>1 Ausgangssituation</b>	<b>2</b>
<b>2 Zielsetzung und Produkteinsatz</b>	<b>2</b>
<b>3 Funktionale Anforderungen</b>	<b>2</b>
3.1 Muss-Ziele . . . . .	2
3.2 Kann-Ziele . . . . .	3
<b>4 Nichtfunktionale Anforderungen</b>	<b>4</b>
4.1 Muss-Ziele . . . . .	4
<b>5 Qualitätsmatrix nach ISO 25010</b>	<b>5</b>
<b>6 Lieferumfang und Abnahmekriterien</b>	<b>5</b>
6.1 Lieferumfang . . . . .	5
6.2 Abnahmekriterien . . . . .	5
<b>7 Vorprojekt</b>	<b>6</b>
<b>8 Glossar</b>	<b>6</b>

## 1 Ausgangssituation

Innerhalb des nächsten Jahres sollen Schüler/innen von Leipziger Schulen für Zukunftsthemen im außerunterrichtlichen Bereich sensibilisiert werden. Vorbild ist dabei das Trierer Zukunftsdiplom. Dabei sollen interessierte Schüler/innen einen Zukunftspass erhalten der eine ID zur eindeutigen Identifizierung beinhaltet. Die Zuordnung zum Namen ist nur der jeweiligen Schule bekannt. Schüler die an einer gewissen Mindestanzahl an Veranstaltungen teilgenommen haben, sollen am Ende eine Auszeichnung bekommen.

Bisher gibt es bereits Ferienpässe, jedoch existiert im Moment noch keine digitale Struktur um eine schnelle Erfassung der Teilnehmer einer Veranstaltung zu Realisieren. Eine solche soll nun Rund um nachhaltiges-leipzig.de (welches Veranstalter und Events erfasst) erstellt werden.

## 2 Zielsetzung und Produkteinsatz

Im Rahmen des Projektes soll eine App entwickelt werden über der sich die Schüler durch Eingabe oder Scannen ihrer IDs anschauen können an wie vielen Events sie teilgenommen haben sowie eine Liste von weiteren Events haben.

Gleichzeitig soll es eine Möglichkeit für Veranstalter geben sich bei dieser App anzumelden um somit Teilnehmer ihrer Events zu erfassen, damit im System eingetragen werden kann das die Person zu diesem Event anwesend war. Dazu sollen die IDs auf den Ausweise der Schüler vom Veranstalter über die App gescannt werden können.

Des Weiteren soll die dazu nötige Infrastruktur entworfen und prototypisch implementiert werden. Die App soll als Android App auf Basis des Android SDKs entwickelt werden.

## 3 Funktionale Anforderungen

### 3.1 Muss-Ziele

#### Startbildschirm und Anmeldefunktion

AF010 Der Benutzer wird beim erstmaligem Start der App oder wenn keine gespeicherten Anmeldedaten vorliegen aufgefordert sich zu identifizieren.

Schüler geben dazu lediglich ihre ID ein oder Scannen diese. Ist die eingegebene ID im System nicht vorhanden, wird zur erneuten Eingabe aufgefordert.

Für Veranstalter gibt es eine Schaltfläche, durch welche die App in den Veranstaltermodus wechselt.

AF020 Ein Veranstalter muss zum Anmelden seine E-Mail-Adresse eingeben, diese wird an den Server übertragen, welcher bestätigen muss, dass die Adresse zu einem Veranstalter gehört. Der Server verfügt dazu über eine Liste von E-Mail-Adressen die zu Veranstaltern gehören.

Ist die eingegebene E-Mail-Adresse in dieser Liste enthalten, erzeugt die App lokal ein Schlüsselpaar aus private und public key und sendet den public key an den Server. Der Server sendet an die E-Mail-Adresse automatisch eine E-Mail die einen Bestätigungslink enthält. Der Veranstalter muss diesen aufrufen. Dadurch wird der public key freigegeben, so dass sich die App des Veranstalters mit diesem authentifizieren kann.

Zu einem Veranstalten können mehrere Schlüssel (pro Gerät einer) gehören.

AF030 Die Anmeldedaten werden gespeichert. Die App nutzt beim nächsten Neustart die gespeicherten Anmeldedaten.

AF040 Ist der Benutzer angemeldet, hat er über das Menü die Funktion sich abzumelden. Dadurch werden die gespeicherten Anmeldedaten zurückgesetzt und der zur Anmeldung auffordernde Startbildschirm wird angezeigt. So kann die App von mehreren Benutzern auf dem selben Gerät verwendet werden.

Die bei der Anmeldung eines Veranstalters erzeugten Schlüssel werden nicht gelöscht, sondern zusammen mit der E-Mail-Adresse gespeichert, so dass beim erneut anmelden nur diese Adresse erneut angegeben werden muss.

AF050 Für Veranstalter gibt es die Möglichkeit das für die Anmeldung verwendete Schlüsselpaar zu löschen. Dabei wird der Veranstalter abgemeldet.

### **Allgemeine Anzeigefunktionen der App**

GF010 Die App bietet dem Nutzer die Möglichkeit, sich die Termine der nächsten Veranstaltungen anzeigen zu lassen.

### **Funktionen für angemeldete Schüler**

SF010 Ein angemeldeter Schüler kann seine persönliche Statistik (Anzahl besuchter Veranstaltungen) einsehen.

SF020 Ein angemeldeter Schüler wird die eigene ID angezeigt.

Dabei soll diese Anzeige bei einer Veranstaltung anstatt des physischen Passes eingescannt werden können.

SF030 Ein angemeldeter Schüler kann eine Bestenliste, welche die Platzierung (nach Anzahl besuchter Veranstaltungen), ID und die Anzahl besuchter Veranstaltungen der besten 10 und des Schülers selbst anzeigt.

### **Funktionen für Veranstalter**

VF010 Dem angemeldeten Veranstalter werden die eigenen Veranstaltungen angezeigt. Von diesen kann er eine auswählen um den Scavorgang zu starten.

VF020 Der Scavorgang bietet die Möglichkeit die ID von Teilnehmern bei einer Veranstaltung zu scannen. Die Anwesenheit der Teilnehmer für diese Veranstaltung wird dann automatisch in die Datenbank aufgenommen.

Alternativ kann die ID manuell eingegeben werden.

VF030 Dem angemeldetem Veranstalter werden bei seinen eigenen Veranstaltungen, die Anzahl der in der Datenbank gespeicherten Teilnehmer angezeigt.

VF040 Der angemeldete Veranstalter hat die Möglichkeit sein Passwort zu ändern.

## **3.2 Kann-Ziele**

### **Generelle Funktionen**

GF110 Die App kann personalisiert werden. (Z.B. Hintergrundfarbe)

### **Erweiterte Funktionen der Punktevergabe**

- EF110 Für eine Veranstaltung soll es auch einen Punktewert geben.
- EF120 Die erreichten Punkte werden bei der persönlichen Statistik und in der Bestenliste zusätzlich angezeigt.
- EF130 Die Bestenliste für Schüler wird nach erreichten Punkten sortiert.
- EF140 Die Punkte werden nach Ablauf eines Aktionszeitraumes für das Zertifikat zurückgesetzt.

### **Funktionen für angemeldete Schüler**

- SF110 Der angemeldete Schüler hat die Möglichkeit sich bereits im voraus unverbindlich für eine Veranstaltung anzumelden.
- SF120 Der angemeldete Schüler kann sich eine Historie der von ihm besuchten Veranstaltungen anzeigen lassen.
- SF130 Der angemeldete Schüler kann die Bestenliste nach verschiedenen Kriterien (z.B. Schule, Art der Veranstaltung) filtern.<sup>1</sup>
- SF140 Der angemeldete Schüler kann Veranstaltungen an denen er teilgenommen hat bewerten.

### **Funktionen für Veranstalter**

- VF110 Dem angemeldetem Veranstalter werden bei eigenen zukünftigen Veranstaltungen die Anzahl der Schüler, welche sich über die App im Voraus angemeldet haben, angezeigt.
- VF120 Der angemeldete Veranstalter kann sich eine Bestenliste seiner eigenen Veranstaltungen nach Anzahl der Teilnehmer oder Bewertung anzeigen lassen.

## **4 Nichtfunktionale Anforderungen**

### **4.1 Muss-Ziele**

#### **Nutzung**

- NF010 Die Oberfläche und Bedienung der App sollte einfach und intuitiv sein, damit auch nicht Technik affine Veranstalter sowie Schüler sie bedienen können.
- NF020 Der ID-Scanner soll zuverlässig und schnell arbeiten, so dass es Sinn macht, die ID nicht per Hand einzugeben.
- NF030 Ein Benutzer kann sich auf mehreren Geräten gleichzeitig anmelden.

---

<sup>1</sup>Diese Erweiterung setzt voraus, dass genügt Daten, nach denen gefiltert werden kann, vorhanden sind.

## Datenübertragung

- DF010 Die Datenübertragung der Veranstalter Daten soll verschlüsselt sein. Die App kennt dabei bereits den public key des Servers. Für die Erstellung des Schlüsselpaares der App siehe AF020.
- DF020 Es soll sichergestellt werden, dass gescannte Schüler auch bei nicht bestehender Internetverbindung als an der Veranstaltung teilgenommen eingetragen werden. In diesem Fall soll die Teilnehmerliste temporär lokal gespeichert werden und bei bestehender Internetverbindung übertragen werden.

## 5 Qualitätsmatrix nach ISO 25010

Kriterien	Bedeutung
Funktionale Software	hoch
Effiziente Performance	niedrig
Höchste Sicherheit	mittel
Hohe Kompatibilität	niedrig
Verlässliche Software	mittel
Perfekte Nutzbarkeit	hoch
Einfache Wartung	niedrig
Leichte Portierbarkeit	niedrig

## 6 Lieferumfang und Abnahmekriterien

### 6.1 Lieferumfang

**Front-End** Wir liefern eine Andriod-App, welche gleichermaßen von Schülern und Veranstaltern genutzt werden kann.

**Back-End** Wir erstellen eine Datenbank im RDF-Format zum Speichern der Teilnehmer in Form von anonymen IDs, deren besuchte Veranstaltungen und eventuell ihrer Punktestände, auf die die App zugreifen kann. Für die Anbindung soll die das Apache Jena Framework verwendet werden. Zunächst wird diese Datenbank auf dem Praktikumsserver implementiert. Später soll jedoch die Möglichkeit bestehen die Datenbank auf einen anderen Server zu verschieben.

**Dokumentation** Das Projekt wird dokumentiert, sowohl um den Endnutzern die Bedienung zu erleichtern, als auch um die Weiterentwicklung durch andere Entwickler zu vereinfachen. Die Dokumentation soll in Form von Kommentaren im Quellcode bzw. Javadoc sowie als PDF-Dateien erzeugt werden.

### 6.2 Abnahmekriterien

**Front-End** Die Andriod-App muss auf jedem aktuellen Android Gerät installiert werden können und stabil laufen. Alle oben genannten Muss-Ziele sind zu erfüllen.

**Back-End** Die Datenbank muss in der Lage sein, Daten sowohl über die App zu beziehen, als auch mit den Informationen der bereits bestehenden Primärplattformen zu arbeiten.

**Dokumentation** Die Dokumentation beinhaltet:

- Darstellung und Bedienung der Features aus Sicht des Endnutzers
- Aspekte der Installation und Konfiguration für Systemadministratoren
- Darstellung von Informationen und Komponenten der Software für Entwickler mit verschiedenen Detailierungsstufen
- Anforderungen und Entscheidungen während der Entwicklung für nachhaltige Kommunikation über den Kontext der Entwicklung

## 7 Vorprojekt

Als Vorprojekt soll eine Vorab Version der App konstruiert werden, die bereits einige der Muss-Ziele umgesetzt hat und zeigt dass diese funktionieren.

Als Erstes soll die Benutzeroberfläche bis dato größtenteils fertig sein und zeigen wie das Team die App designen möchte.

Des Weiteren soll die „Schüler-Seite“ mit Ihren wichtigsten Funktionen entwickelt worden sein. Die App soll eine ID einscannen können (alternativ muss diese eingetippt werden) und anschließend feststellen ob diese bereits in der Datenbank vorhanden ist. Falls die ID unbekannt ist, so soll ein Fehler angezeigt werden. Andernfalls sollen der ID die jeweiligen Werte(z.B. Highscore) zugeordnet werden können. Dazu wird eine Test-Datenbank entworfen welche einige manuell erstellte Datensätze enthält. Diese sollen testweise angezeigt werden, wenn die entsprechende ID erkannt wurde(bzw. hinterlegt ist). Und man sollte neben dem Punktestand auch seinen Platz in einer Rangliste sehen können, verglichen mit anderen IDs.

Auf den „Veranstalter-Teil“ der App soll im Rahmen des Vorprojekts nicht weiter eingegangen werden.

## 8 Glossar

**Android SDK** Das Android Software Development Kit ist eine Entwicklungsumgebung und Sammlung von Tools für die Softwareentwicklung für das Android-Betriebssystem.

**Apache Jena Framework** Apache Jena ist ein kostenfreies Open-Source-Framework für Java zum erstellen von Semantic Web Applikationen. Der Nutzer kann dabei über eine Programmierschnittstelle auf Resource-Description-Framework-Graphen (RDF-Graphen) zugreifen, zum Speichern und Laden von Daten. Die Modelle können Daten aus verschiedenen Quellen beziehen wie zum Beispiel aus Dateien, Datenbänken, URLs oder einer Kombination der genannten.

**Backend** Backend beschreibt das auf dem Server laufende Programm.

**Bestätigungslink** Personalisierte Webadresse bei deren Aufruf dem Server bestimmte Daten übersendet werden. Er wird oft zur Authentifizierung verwendet, indem er in E-Mails verschickt wird von denen man ausgeht, dass sie nur die betreffende Person lesen kann.

**Datenbank** Eine Datenbank ist ein System zur effizienten, widerspruchsfreien und dauerhaften elektronischen Speicherung von Daten.

**Frontend** Frontend beschreibt das auf dem Endgerät des Nutzers laufende Programm.

**ID** (Identifikationsnummer) Im Zusammenhang dieses Projektes ist die Identifikationsnummer auf den Pässen der Schüler gemeint.

**Plattformbetreiber** Der Betreiber einer Plattform ist eine Organisation, die für eine Verbreitung ihrer Angebote erforderliche Netzinfrastruktur entweder selber betreibt, durch Dritte betreiben lässt oder das offene Internet nutzt. Im Zusammenhang dieses Projektes ist der Plattformbetreiber die Stadt Leipzig, beziehungsweise die Verantwortlichen des Ferienpasses.

**RDF** RDF steht für Resource Description Framework. Es wurde von der W3C als Standard zur Beschreibung von Metadaten konzipiert. Es gilt als ein grundlegender Baustein des Semantischen Webs. RDF ähnelt den klassischen Methoden zur Modellierung von Konzepten wie UML-Klassendiagramme und ER-Modell.

**Secure Shell** (kurz SSH) bezeichnet ein Netzwerkprotokoll, mit deren Hilfe man auf eine sichere Art und Weise eine verschlüsselte Netzwerkverbindung mit einem entfernten Gerät herstellen kann.

**Semantic Web** Beim semantischen Web handelt es sich um eine Web-Technologie bei dem die Suchmaschinen Informationen zueinander in Beziehung setzen, sie eigenständig auswerten und aus ihnen eine Bedeutung entnehmen können.

**Trierer Zukunftsdiplom** Ziel des Diploms ist die Sensibilisierung von Kindern im Alter von 6 bis 12 Jahren für Nachhaltigkeit. Von Mai bis September werden verschiedene, von unterschiedlichsten Anbietern ausgerichtete, Aktivitäten angeboten. Jedes Kind, das wenigstens 6 Veranstaltungen besucht kann im November sein Zukunfts-Diplom entgegennehmen.

**private key** Privater Schlüssel – Teil eines Schlüsselpaares das nur dem Besitzer bekannt ist und mit dem durch den dazugehörigen public key verschlüsselte Daten entschlüsselt werden können.

**public key** Öffentlicher Schlüssel – Teil eines Schlüsselpaares mit dem Nachrichten an den Besitzer dieses Schlüssels verschlüsselt werden können.