

Audit 3. : Entwicklungsprojekt

Richard Reh, Anton Berg, Vassilij Misenko

Inhaltsverzeichnis

- Projektrisiken
- Proof of Concepts überarbeitet
- Alleinstellungsmerkmal
- Zielhierarchie
- Architektur
- Projektplan für den 4. Audit
- Proof of Concept Vorstellung

Projektrisiken 2 Audit

Risiko	Priorisierung
Offene Sitzungen funktioniert nicht	1
Das Erstellen einer Sitzung (AD Hoc) funktioniert nicht	2
Das Beitreten einer Sitzung (AD Hoc) funktioniert nicht	3
Das Teilen von Information (Bild / JPEG / -> falls möglich Dateien)	4
Bestückung von Information (virtueller Stift) funktioniert nicht	5

Proof of Concept – Open-Space Sitzung

Beschreibung:

Eine Sitzung in der sich eine Anzahl an Clients wiederfinden. Darin können Informationen zwischen Clients ausgetauscht werden.

Exit-Kriterien:

- Open-Space Sitzung findet statt, die Clients sind miteinander verbunden.

Fail-Kriterien:

- Es gibt keine synchrone Open-Space Sitzung.
- Beigetretene Nutzer sehen nicht direkt den aktuellen Stand.

Fallbacks:

- Kein Fallback möglich, da es eine essentielle Funktion ist

Proof of Concept – Erstellung einer Open-Space Sitzung

Beschreibung:

Anwender können eine neue Sitzung erstellen.

Exit-Kriterien:

- Eine Open Space Sitzung wurde erstellt.

Fail-Kriterien:

- Einer Open Space Sitzung lässt sich nicht erstellt.
- Beitritts - Code kann nicht erstellt werden

Fallbacks:

- Erstellung einer alternativen Beitrittsmöglichkeit z.B wenn QR-Code Generierung nicht klappt, dann soll stattdessen eine Raum-ID erstellt werden, durch die andere Nutzer beitreten könnten.

Proof of Concept – Beitretten einer Open-Space Sitzung

Beschreibung:

Anwender können einem bereits erstellten Raum beitreten.

Exit-Kriterien:

- Anwender treten anhand eines "Codes" / "Links" einem Raum erfolgreich bei.
- Anwender sieht den synchronisierten Stand (die Information) des Raumes nach dem beitreten.

Fail-Kriterien:

- Einer vorhandenen Sitzung kann der Anwender nicht beitreten.

Fallbacks:

- Da es eine essentielle Funktion ist, kann hierfür kein Fallback erstellt werden.

Proof of Concept - Teilen von Information (Bild / Streaming eigener Dateien)

Beschreibung:

Der Anwender kann eigene Informationen teilen (genauer Prozess wird noch erarbeitet)

Exit-Kriterien:

- Alle Anwender können die geteilte Information synchron sehen
- Jeder Anwender hat die Möglichkeit seine Information zu teilen, solange gerade niemand anderes seine Information teilt

Fail-Kriterien:

- Die geteilte Information wird nicht angezeigt oder nicht synchron angezeigt.
- Die Anwender können nichts teilen

Fallbacks:

- Andere Funktionsart für das Teilen von Informationen verwenden
- Eine bestimmte Dateityp (oder falls möglich viele) für das transiente Hochladen verwenden, damit diese synchron geteilt werden kann

Proof of Concept - Bestückung von Information anhand von virtuellem Stift (Nice To Have)

Beschreibung:

Ermöglichen des Beschriften/Zeichnen auf der geteilten Fläche wie mit einem Stift, wobei das gezeichnete an alle Nutzer in der Sitzung synchron mitgeteilt wird.

Exit-Kriterien:

Das Zeichnen/Beschriften funktioniert und die daraus entstehende Grafik wird für alle Nutzer synchron dargestellt.

Fail-Kriterien:

- Die Zeichnungen/Beschriftungen werden nicht übermittelt
- Die Zeichnungen/Beschriftungen werden zwar übermittelt, werden aber nicht für alle korrekt synchron dargestellt.
- Das Zeichnen/Beschriften an sich funktioniert nicht.

Fallbacks:

- Stattdessen noch eine andere Art zum hinzufügen von Informationen hinzufügen, z.B Text an einer bestimmten Stelle darstellen.
- Keine Fallback, da es theoretisch nicht essentiell ist

Alleinstellungsmerkmal

Ein Aspekt wäre die unkomplizierte und schnelle Einrichtung des Arbeitsraumes ohne Registrierung/Anmeldung, sodass alle Teammitglieder nahezu umgehend starten können. Während der Sitzung werden auch keine versendeten Daten gesammelt und persistent gespeichert, sodass auch Privatsphäre für die Nutzer des Systems gegeben wird.

Andere Anbieter für ähnliche Lösungen bestehen eher darauf, dass ihr Service erst nach einer Registrierung in ihrem System verwendbar ist.

Darüber hinaus wird dadurch ein Austausch von Informationen ermöglicht, ohne dafür extra Hardware benutzen zu müssen wie externe Datenkabel, Speichermedien etc. Und die geteilte Information soll dabei nicht „als eigene Datei“ heruntergeladen werden, sondern nur im Cache des Browsers verfügbar sein.

Zielhierarchie

Strategische Ziele:

1. Kollaboratives Arbeiten für Menschen ermöglichen.
2. Privatsphäre für Menschen innerhalb einer kollaborativen Arbeit schaffen.
3. Erleichterter Austausch von Informationen.

Taktische Ziele:

1. Erstellung eines Open-spaces für eine gemeinsame Sitzung.
2. Teilen von Elementen nur innerhalb der gemeinsamen Sitzung.
3. Anbieten eines Werkzeuges für das Anmerken von Informationen.

Operative Ziele:

1. Webserver (mit HTTP-Protokoll) ermöglicht die Kommunikation in einer Sitzung.
 - 1.1 Synchrone Kommunikation zwischen Clients und Webserver.
2. Hochladen und Anzeigen von Elementen unter den Clients. --> Dadurch Austausch von Informationen.
3. Mittels eines Werkzeuges sollten die geteilten Elemente bearbeitet / ergänzt werden und die ergänzten Informationen sollen für alle Clients live angezeigt werden.
4. Kurzspeicherung der Informationen auf dem Servercache.

Architektur

- Websockets:
 - Geeignetes TCP-Protokoll für den Austausch von Information zwischen Clients und Server
 - Gut geeignet für die Übermittlung Daten in Echtzeit
- Daten:
 - Client-Id und der Nutzernamen des Benutzers (keine Anmeldung)
 - Sitzungs-Id für die Session
 - Weitere Informationen teilen (Beispielweise ein Bild) (geplant)
 - HTML-Canvas Information zu den "gezeichneten Notizen"
- Datenbank:
 - Keine Persistenz
 - Keine Anmeldedaten (Ad Hoc)
 - Alle Informationen nur einsehbar, solange die Sitzung existiert
 - Nur Nutzer in der Sitzung können auf die Daten zugreifen / diese Daten sehen
- Alle wichtigen Informationen sind in dem GitHub Wiki unter [Architektur des Systems](#) nochmal aufzufinden

Projektplan für den 4. Audit

- 1: restliche PoCs verwirklichen und mit anderen verknüpfen
- 2: ein passendes und funktionales Frontend entwickeln
- 3: alle PoCs zu dem Main Prototype verbinden
- 4: das Poster für dieses Projekt erstellen
- 5: kritische Reflektion über das Projekt aus der Sicht der Beteiligten