

Sitzungsprotokoll WIPRO - HSLU Mobile Apps

Datum/Zeit	30.10.2025, 10:00- 11:00
Ort	Online (Microsoft Teams)
Teilnehmer	Auftraggeber: Jürg Nietlisbach Studenten: Raphael Eiholzer, Samuel Kurmann
Protokoll	Raphael Eiholzer

Traktanden / Diskutierte Punkte

Neuerungen bei iOS APP

- Bestehendes Problem bei Tracking-Einstellungen: Aktuell teilweise überlagernde Einstellungen (zu Consent Tracking); Einmal in den App-Einstellungen, in der WebView wird aber trotzdem noch einmal danach gefragt
- Idee: Neu werden News, Blog, Mensa per Rest-API (JSON) geholt
 - o hub.hslu.ch (<https://hub.hslu.ch/>) läuft auf WordPress-> Daher können per WordPress-API die Beiträge (und weiteres) gesammelt werden
 - o Beispiele:
 - <https://hub.hslu.ch/informatik/wp-json/wp/v2/posts>
 - https://hub.hslu.ch/informatik/wp-json/wp/v2/posts?per_page=20
 - o Neuer Code dazu ist/wird in Swift-App aktuell angepasst (Siehe [hier](#))
 - o Neu erscheint im View eine Liste mit Blogs-> Darüber gelangt man zur Detailansicht (bzw. Preview eines einzelnen Artikels)-> Von hier aus kann zum "Weiterlesen" zusätzlich Safari geöffnet werden
- Links:
 - o Hochschulübergreifende News sind hier zu finden: <https://news.hslu.ch/> (nicht mandatenspezifisch)
 - o Events: SiteCore, nicht WordPress:
[https://www.hslu.ch/en/api/eventlist/load/?page=1&per_page=25&filters\[\]=%E2%80%A621&datasource%E2%80%A6](https://www.hslu.ch/en/api/eventlist/load/?page=1&per_page=25&filters[]=%E2%80%A621&datasource%E2%80%A6)
- Bezug zum Android JetpackCompose Projekt:
 - o Es wäre gut, wenn neben WebView diese «JSON»-View auch zusätzlich noch integriert wird
 - o Nach Ressourcentyp (Web oder JSON) wie im Backend angegeben wird, wird in *BootstrappingProgressView* entschieden, welche View lokal genommen wird
- Dazu die konkrete Frage bezüglich Tracking-Einstellungen bei Android (Weswegen die View-Anpassung zum Thema wurde):
Antwort: Tracking-Einstellungen bei Android kaum ein Thema. Aktuell in XML-App auch nicht vorhanden (kann auch bei JetpackCompose ignoriert werden)

Besprechung zum Projektstand

- Lokalisierung:
 - o Bei iOS wird URL (für API) nicht statisch hinterlegt -> Dafür gibt es ein DTO (ApplInitModuleDTO) [Muss im Android-Code noch nachgebessert werden]
 - o Schlussendlich: Modul kennt nur "eine Sprache", beim Eintrittspunkt entscheidet App anhand Gerätesprache. Ziel: Es wird nur eine URL ans Feature weitergeleitet!

Neuerungen allgemein

- Mensa: Gibt eventuell in Zukunft einen REST-API Endpunkt, um Menü zu laden.
 - o <https://app.food2050.ch/de/v2/zfv/hslu,standort-rotkreuz/hslu-iandw>
 - o https://app.food2050.ch/_next/data/vwJquWC0RICODjqE7xhAM/de/foodstoffi/foodstoffi/menu/foodstoffi/weekly.json
 - o Dies aber noch nicht fix, in Abklärung mit Mensa-Betreiber
 - o Aktuell reicht es aber, einfach das iFrame des Menüs einzubetten (Aktuell ist ganze Website der Mensa im WebView dargestellt):
<https://app.food2050.ch/de/foodstoffi/foodstoffi/menu/foodstoffi/weekly>
- Raumsuche: Eventuell gibt es in Zukunft eine Umstellung auf PDF (Auf iOS eleganter lösbar mit PDFKit)
 - o Deshalb noch nicht direkt mit Raumsuche-Feature bei Android starten
 - o Es stellt sich noch die Frage, wie das bei Android umzusetzen wäre (Aufgabe: ca. 1 Stunde evaluieren, ob es wie bei iOS ein PDFKit gibt?)

Diverses

- Zu Risiken:
 - o Zeitliche Komponente bei Verwendung von AI-Tools: Wenn Zeit verloren geht beim AI-First-Ansatz (z.B. durch langes Refactoring), nur einen «Time-Boxed» AI First Ansatz verwenden (z.B. nach 1 Stunde abbrechen und manuell implementieren). Aktuell ist aber die Auffassung der Studierenden, dass durch AI-First Zeit eingespart wird
- Zur AI-Evaluation:
 - o Aktuell sind vor allem subjektive Eindrücke bei der Verwendung entstanden. Evtl. Messgrößen definieren und damit testen (z.B. Zeit stoppen? Oder wieviele Refactorings/Reviews es dadurch mehr braucht)
- Priorisierung der nächsten Features:
 - o Am wichtigsten: WebView, danach Raumsuche, Stundenplan (Aktuell: Trais nicht)