Anforderungen an nativen Android Dienst

Die nachfolgende Tabelle stellt eine Auflistung von Anforderungen an die Lösung mittels nativem Android Dienst dar.

| **Kriterium** | **Beschreibung** | **Bewertung** |
| --- | --- | --- |
| **1. Allgemeine technische Anforderungen** | **Der native Dienst…** | **erfüllt?** |
| 1.1 Plattform: Android | …soll für die Android Plattform entwickelt werden. Als Mindestversion soll Android 5.0 (API Level 21) vorausgesetzt werden. Eine Unterstützung anderer Betriebssysteme ist nicht vorgesehen. |  |
| 1.2 Hintergrunddienst | …soll als Hintergrunddienst ohne GUI implementiert werden. |  |
| 1.3 Automatischer Start | ...soll automatisch nach Systemstart gestartet werden. Wird der Dienst aus irgendeinem Grund beendet, soll er automatisch neustarten. |  |
| 1.4 Webserver | ...soll einen Webserver bereitstellen, welcher aus Sicht der Client-Webanwendung das „Caching“ der statischen Ressourcen und die Bereitstellung des Datamodels übernimmt. |  |
| 1.5 Push-Notification | …soll Benachrichtigungen über die native Notification-API auf dem mobilen Endgerät anzeigen. |  |
| **2. Anforderungen Webserver-Komponente** |  |  |
| 2.1 Bereitstellung statischer Ressourcen | …soll einen HTTP Server bereitstellen, welcher die statischen Ressourcen für die Webanwendung auf dem Client lokal bereitstellt. Die Ressourcen bezieht die Dienstanwendung von einem entfernten Webserver und hält diese für die Webanwendung vor (Caching). Dadurch soll eine Offlinefähigkeit der Webanwendung sichergestellt werden. Für die Kommunikation mit der Webanwendung soll die HTTP-Methode GET angewendet werden. |  |
| 2.2 selbstständige Aktualisierung der statischen Ressourcen | …soll mit einem entfernten Webserver kommunizieren, um die vorgehaltenen statischen Ressourcen aktuell zu halten. Dazu wird ein Updatemechanismus implementiert, der alle 4 Stunden prüft, ob Ressourcen aktualisiert werden müssen. |  |
| 2.3 Verwaltung der GUI Logik | ...soll die Anzeigelogik innerhalb der GUI verwalten. |  |
| 2.4 Weiterleitung von Anfragen | …soll Anfragen (Geschäftslogik) aus der Webanwendung an einen entfernten Webserver weiterleiten. Dazu nutzt die Webserverkomponente die RESTful-Schnittstelle des entfernten Servers. Das Ergebnis soll durch die Dienst zwischengespeichert und die Webanwendung bereitgestellt werden. |  |
| 2.5 Speichern von Änderungen im Offlinebetrieb | …soll Änderungen des Models durch die Webanwendung intern speichern, wenn keine Internetverbindung vorhanden ist. Sobald der Dienst wieder mit dem Internet verbunden ist, soll er die Anfragen an den entfernten Anwendungsserver übertragen. |  |
| **3. Anforderungen Push- und Notification-Komponente** |  |  |
| 3.1 Anzeige von Benachrichtigung | …soll Push-Benachrichtigungen bei Freundschaftsanfragen, Einladung zu einer Aufgabe und Änderungen an einer geteilten Aufgabe anzeigen. |  |
| 3.2 Benachrichtigungen verarbeiten | …soll eingehende GCM Push-Benachrichtigungen annehmen und entsprechend verarbeiten. |  |
| 3.3 Push-Aktualisierung des Models | …soll eingehende Push Nachrichten verarbeiten und daraufhin das vorgehaltene Datenmodell aktualisieren. Ggf. kann eine Benachrichtigung angezeigt werden (abhängig vom Push-Typ). |  |