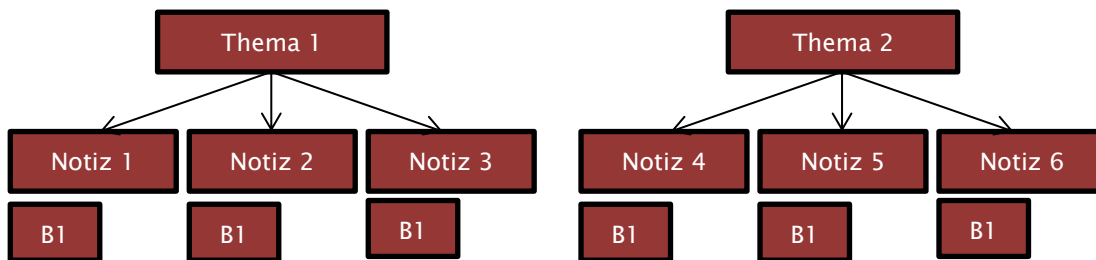




Es sollen Notizen zu bestimmten Themen chronologisch von einem Benutzer erstellt, lokal am Browser abgelegt und eingesehen werden können (**ANMERKUNG:** In einer weiteren Ausbaustufe sollen diese Notizen von mehreren Benutzern erstellt, verwaltet und online zur Verfügung gestellt werden. Dies wird aber erst in einem der nächsten Kapitel behandelt).

Jede Notiz hat eine *Überschrift* und einen *Text*. Zur Notiz werden das *Erstellungs-* und *Änderungsdatum* abgelegt. Jede Notiz gehört einem Benutzer – dem *Ersteller*. Notizen werden *Themen* zugeordnet. Der Benutzer muss im Zuge der Erstellung auch das *Thema*, zu dem die Notiz gehört auswählen oder neu erstellen können. Die *Themenbeschreibung* ist eindeutig und muss festgelegt werden. Für ein Thema muss dessen Beschreibung festgelegt werden. Ein Thema kann nur gelöscht werden, wenn es keiner Notiz zugeordnet wurde. Eine Notiz muss über einen Titel, ein Thema und einen Benutzer verfügen.



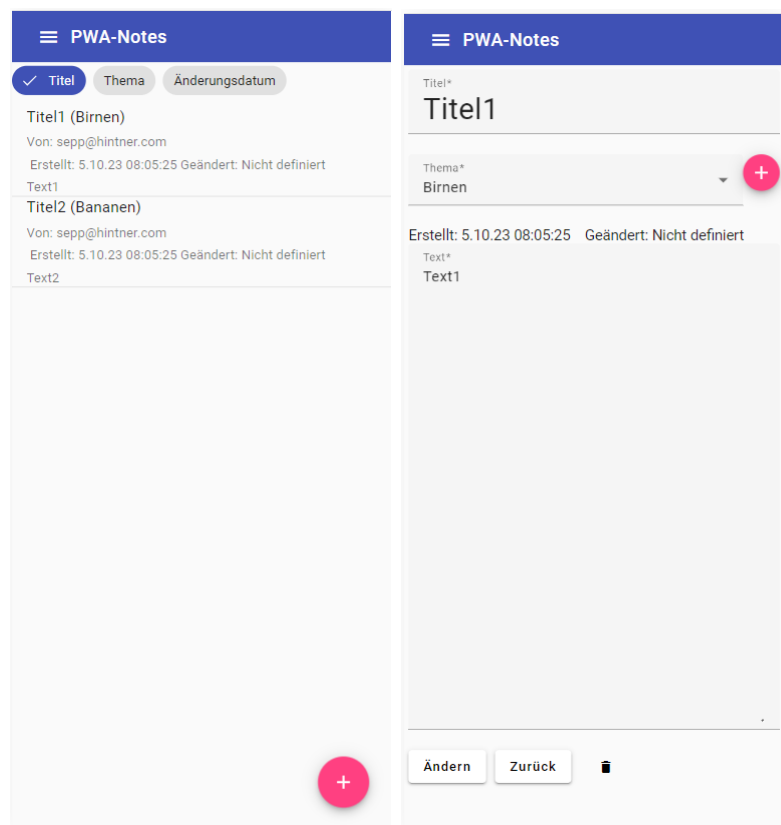
Der *Benutzer* ist bereits fix vorhanden, wird nicht geändert und enthält als *Benutzernamen* seine E-Mail-Adresse und sein *Passwort*.

Dem Benutzer müssen neben der Verwaltung seine Notizen und Themen auch die Möglichkeit gegeben werden, die Notizen nach sinnvollen Kriterien *sortiert* einzusehen. Folgende Sortierkriterien müssen ausgewählt werden können:

- Aufsteigend nach *Titel* der Notiz
- Aufsteigend nach *Thema* und bei gleichem Thema aufsteigend nach Titel der Notiz
- Absteigend nach *Änderungsdatum* und bei leerem Änderungsdatum absteigend nach Erstellungsdatum

Erstellen Sie eine *Progressive Web App*, die *responsiv* ist, registrieren Sie den *Service-Worker*, machen Sie die *App installierbar*, und verwalten Sie die Notizen und Themen über *IndexedDB*. Beachten Sie dabei Folgendes:

- Die Klassen *Theme*, *Note* und *User* werden als *Rohdateien* zur Verfügung gestellt. der *Service DbService* muss von Ihnen programmiert werden.
- Über die *Navigationskomponente* sollen die Menüpunkte *Notizen* und *Themen* angesprungen werden können. Beim Öffnen der App werden automatisch die Notizen angezeigt.
- Das Hinzufügen von neuen Notizen und Themen soll über einen *Floating Action Button* möglich sein. Änderungen erfolgen über einen Klick auf das entsprechende Objekt.
- Die Auswahl der Sortierreihenfolge



soll über die Angular Material *Chips-Komponente* erfolgen. Sollte der Benutzer kein Chip auswählen, werden keine Notizen angezeigt. Über eine Angular Material *Snackbar-Komponente* soll ein Hinweis darauf angezeigt werden.

- Bei der Notizeneingabe soll es möglich sein, der Notiz ein bestehendes oder neues Thema hinzuzufügen. Wird ein neues Thema durch den Plus-Knopf erstellt, so soll es automatisch der Notiz hinzugefügt werden.
- Beim Ändern einer bestehenden Notiz muss es die Möglichkeit geben, diese auch zu löschen.
- Die Eingabe eines neuen Themas bzw. das Ändern eines bestehenden muss über eine `MatDialog`-Komponente erfolgen.
- Das Abtesten ob die Themenbeschreibung bereits existiert, muss mit einem *asynchronen Validierer* erfolgen (**ANMERKUNG:** Um von einem *Promise* ein *Observable* zu generieren wird die *rxjs-Methode from()* verwendet).
- Ein bereits bestehendes Thema kann über das entsprechende Symbol im Dialog gelöscht werden. Dabei darf ein Thema nur dann gelöscht werden, wenn es nicht in einer Notiz verwendet wird.

