

Familien-Organizer

Beschreibung: Eine App für Familien, die die Organisation des Alltags vereinfacht und es ermöglicht, gemeinsam Aufgaben zu planen, Einkaufslisten zu erstellen, wichtige Termine zu verwalten und Autos zu reservieren. Die App besteht aus fünf Entitäten, die jeweils eine spezifische Funktion erfüllen und so eine umfassende Organisation für die ganze Familie ermöglichen.

Aufgabenliste (ToDo-Liste): Eine zentrale Tabelle, in der alle anfallenden Aufgaben für die Familie festgehalten werden. Hier können Aufgaben hinzugefügt, bearbeitet und als erledigt markiert werden. Auch Fälligkeitsdaten und Prioritäten können hinterlegt werden.

Einkaufsliste: Eine Tabelle für gemeinsame Einkäufe, in der alle benötigten Artikel aufgeführt sind. Artikel können hinzugefügt, bearbeitet und abgehakt werden, sobald sie gekauft wurden.

Auto-Reservierung: Eine Tabelle, in der die Reservierungen für das Familienauto verwaltet werden. Mitglieder können Zeitfenster buchen und sehen, wann das Auto bereits reserviert ist.

Familienmitglieder: Eine Tabelle zur Verwaltung der Familienmitglieder und deren Zugriffsrechte auf die verschiedenen Funktionen der App. So können bestimmte Aufgaben und Funktionen nur für ausgewählte Mitglieder sichtbar oder bearbeitbar sein.

1. Aufgabenliste (ToDo-Liste)

- **GET /tasks:** Liste aller Aufgaben abrufen.
- **GET /tasks/{id}:** Details einer spezifischen Aufgabe abrufen.
- **POST /tasks:** Eine neue Aufgabe erstellen.
- **PUT /tasks/{id}:** Eine vorhandene Aufgabe aktualisieren (z. B. Fälligkeitsdatum oder Priorität ändern).
- **PATCH /tasks/{id}:** Teilspezifische Aktualisierung einer Aufgabe, z. B. nur das Erledigt-Statusfeld.
- **DELETE /tasks/{id}:** Eine Aufgabe löschen.

2. Einkaufsliste

- **GET /shopping-list:** Gesamte Einkaufsliste abrufen.
- **GET /shopping-list/{id}:** Details eines bestimmten Einkaufslisten-Artikels abrufen.
- **POST /shopping-list:** Neuen Artikel zur Einkaufsliste hinzufügen.
- **PUT /shopping-list/{id}:** Eintrag in der Einkaufsliste aktualisieren, z. B. Menge oder Notiz.
- **PATCH /shopping-list/{id}:** Teilaktualisierung eines Einkaufslisten-Eintrags, z. B. als gekauft markieren.
- **DELETE /shopping-list/{id}:** Einen Artikel von der Einkaufsliste löschen.

3. Kalender (Wichtige Erinnerungen)

- **GET /calendar:** Alle Kalendereinträge anzeigen.
- **GET /calendar/{id}:** Details eines spezifischen Kalendereintrags abrufen.
- **POST /calendar:** Neuen Kalendereintrag hinzufügen.
- **PUT /calendar/{id}:** Kalendereintrag aktualisieren, z. B. Datum oder Beschreibung ändern.
- **DELETE /calendar/{id}:** Einen Kalendereintrag löschen.

4. Auto-Reservierung

- **GET /car-reservations:** Alle Auto-Reservierungen anzeigen.
- **GET /car-reservations/{id}:** Details einer spezifischen Auto-Reservierung abrufen.
- **POST /car-reservations:** Eine neue Auto-Reservierung hinzufügen.
- **PUT /car-reservations/{id}:** Details einer Reservierung aktualisieren (z. B. Zeiten oder Nutzer ändern).
- **DELETE /car-reservations/{id}:** Eine Auto-Reservierung löschen.

5. Familienmitglieder

- **GET /family-members:** Liste aller Familienmitglieder anzeigen.
 - **GET /family-members/{id}:** Details eines spezifischen Familienmitglieds abrufen.
 - **POST /family-members:** Ein neues Familienmitglied hinzufügen.
 - **PUT /family-members/{id}:** Familienmitglied-Daten aktualisieren, z. B. Zugriffsrechte oder Kontaktinformationen.
 - **DELETE /family-members/{id}:** Ein Familienmitglied aus dem System entfernen.
-

Zusätzliche Funktionen (optional)

Falls nötig, könnten einige weitere API-Methoden hinzugefügt werden:

- **PUT /tasks/{id}/assign/{member_id}:** Weist eine Aufgabe einem Familienmitglied zu.
 - **PUT /shopping-list/{id}/mark-as-bought:** Markiert einen Artikel als gekauft.
 - **GET /car-reservations/available:** Gibt eine Liste verfügbarer Zeitfenster für die Auto-Reservierung zurück.
 - **GET /calendar/upcoming:** Zeigt nur die bevorstehenden wichtigen Termine.
-

Beispiel für die Struktur der API-Ressourcen

Die API-Routen könnten folgendermaßen strukturiert sein:

/api/

└─ tasks/

└─ shopping-list/

└─ car-reservations/

└─ calendar/

└─ family-members/

Mit diesen Methoden ist sichergestellt, dass die wichtigsten CRUD-Operationen (Create, Read, Update, Delete) für jede Entität verfügbar sind und die Familienmitglieder reibungslos auf alle Funktionen der App zugreifen können.

