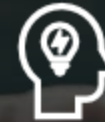




Coding



Plugins



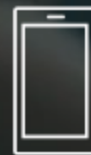
Probleme &
Lösungen



Funktionen

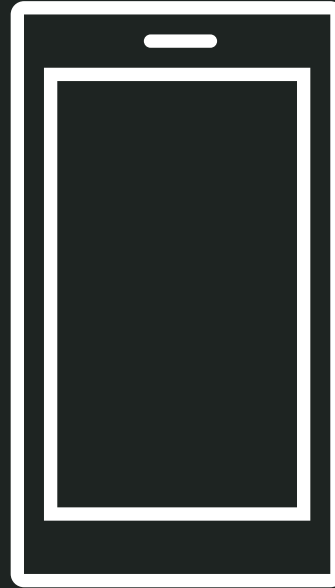


Was gab es zu
beachten?

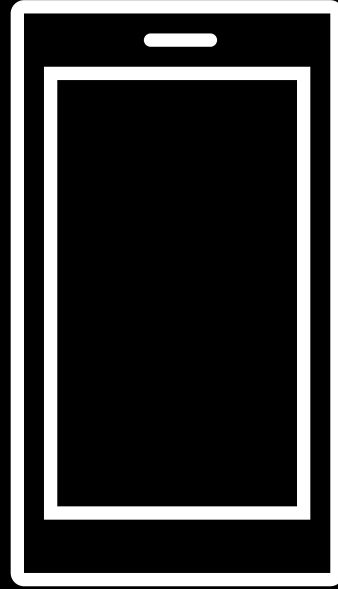


Live Demo

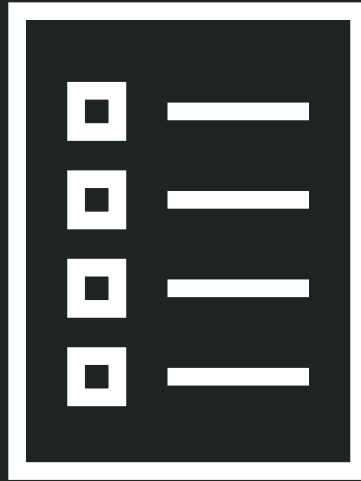
Kontakte App



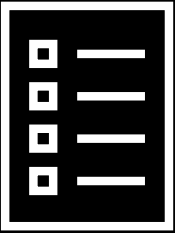
Live Demo



Live Demo

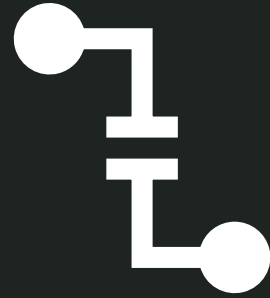


Funktionen



Funktionen

- Berechtigungen werden angefordert
- Es existiert eine nach Vornamen aufsteigend sortierte Listenansicht, die alle bereits auf dem Gerät gespeicherten Kontakte auflistet.
- Es existiert eine Detailansicht, die Vorname, Nachname, Telefonnummer, E-Mail-Adresse und Geburtstag eines Kontakts anzeigt und einen Button besitzt, um zur Listenansicht zurückzukehren.
- Jede Telefonnummer kann in der Standard-Telefon-App geöffnet und in die Zwischenablage kopiert werden, ebenso kann jede E-Mail-Adresse in der Standard-E-Mail-App geöffnet und in die Zwischenablage kopiert werden.
- Kontakte können über die Detailansicht gelöscht werden, wobei das Löschen durch den Nutzer über einen Dialog bestätigt werden muss.
- Es existiert eine Einfügeansicht zum Hinzufügen neuer Kontakte, welche die Eingabefelder für Vor-, Nachname, Telefonnummer, E-Mail-Adresse und Geburtstag enthält , sowie zwei Buttons zum Abbrechen und Speichern.



Plugins

Plugins Contact

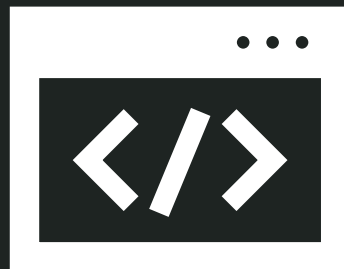
- Zweck:
 - Zugriff auf die Kontaktliste des Geräts
 - Ermöglicht Integration der Kontakte in die App
- Funktionen:
 - Kontakte abrufen, erstellen und löschen
 - Namen, Telefonnummern, E-Mail-Adressen auslesen
 - Kontakte auswählen, anzeigen, weiterverarbeiten
- Verwendung in der App:
 - Kontakte durchsuchen und auswählen
 - Kontaktinformationen weiterverwenden

Plugins Clipboard

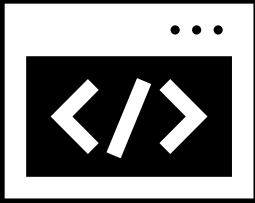
- Zweck:
 - Nützlich für temporäres Speichern und Übertragen von Daten zwischen Apps
 - Text in die Zwischenablage kopieren und einfügen
- Funktionen:
 - Text speichern
 - Text aus der Zwischenablage abrufen
- Verwendung in der App:
 - E-Mail-Adresse und Telefonnummer kopieren
 - Inhalt in der Zwischenablage einfügen

Plugins LToast

- Zweck:
 - Kurze, sofortige Rückmeldungen anzeigen
 - Interaktion mit dem User
- Funktionen:
 - Nachrichten am Bildschirmrand anzeigen
 - Informiert den Nutzer über den Erfolg oder Misserfolg einer Aktion
- Verwendung in der App:
 - Meldung nach einer Aktion
 - Erfolgsbestätigung
 - Fehlermeldung

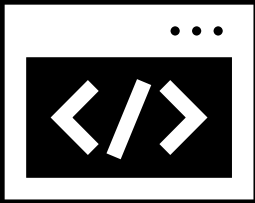


Coding



Coding Implementierung - Berechtigungen

```
<uses-permission android:name="android.permission.READ_CONTACTS" />  
<uses-permission android:name="android.permission.WRITE_CONTACTS" />
```



Coding Implementierung - Listansicht

```
async retrieveListOfContacts () {  
  try {  
    const result = await Contacts.getContacts({  
      projection: {  
        // Specify which fields should be retrieved.  
        name: true,  
        phones: true,  
        emails: true,  
        birthday: true,  
      },  
    });  
    this.contacts = result.contacts;  
    this.sortContactsByFirstName();  
  } catch (error) {  
    console.error("Error loading contacts", error);  
  }  
},
```

2

```
async created() {  
  await this.retrieveListOfContacts();  
},
```

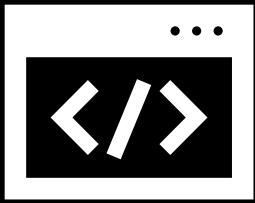
1

```
sortContactsByFirstName() {  
  // Function to sort contacts by given name  
  this.contacts.sort((a, b) => {  
    console.log(a);  
    if (a.name!['given']?.toLowerCase()! < b.name!['given']?.toLowerCase()!) {  
      return -1;  
    } else if (a.name!['given']?.toLowerCase()! > b.name!['given']?.toLowerCase()!) {  
      return 1;  
    } else {  
      return 0;  
    }  
  });  
},
```

3

```
<ion-list>  
  <ion-item v-for="contact in contacts" :key="contact.contactId" @click="pushContactDetail(contact)">  
    <ion-label>  
      <h2>{{ contact.name!["display"] }}</h2>  
    </ion-label>  
  </ion-item>  
</ion-list>
```

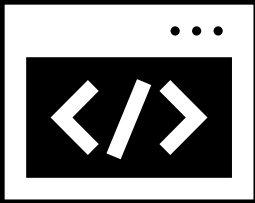
4



Coding Implementierung - Router

```
const routes: Array<RouteRecordRaw> = [  
  {  
    path: '/',  
    redirect: '/home'  
  },  
  {  
    path: '/home',  
    name: 'Home',  
    component: HomePage  
  },  
  {  
    path: '/contact/:contactId',  
    name: 'ContactDetail',  
    component: ContactDetail,  
    props: true  
  },  
]  
  
const router = createRouter({  
  history: createWebHistory(import.meta.env.BASE_URL),  
  routes  
})
```

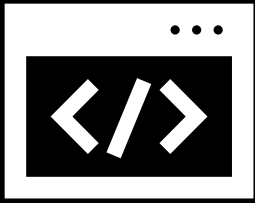
```
pushContactDetail(contact: ContactPayload){  
  this.router.push({name: 'ContactDetail', params: {contactId: contact.contactId} });  
},
```



Coding Implementierung - Detailansicht

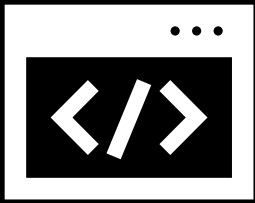
```
<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-button @click="goBackToOverview">
        <ion-icon :icon="arrowBackOutline" slot="start" color="light"></ion-icon>
      </ion-button>
    </ion-buttons>
    <ion-title>Kontakt Details</ion-title>
    <ion-buttons slot="end">
      <ion-button fill="clear" @click="showDeleteAlert(contact.contactId)">
        <ion-icon :icon="trashBinOutline" size="large" color="light"></ion-icon>
      </ion-button>
    </ion-buttons>
  </ion-toolbar>
</ion-header>
```

```
goBackToOverview() {
  this.router.push({ name: 'Home' });
},
```



Coding Implementierung - Detailansicht

```
async created() {  
  const route = useRoute();  
  const contactId = route.params.contactId as string;  
  await this.getSingleContact(contactId);  
  return {  
    contactId,  
  };  
},
```



Coding Implementierung - Detailansicht

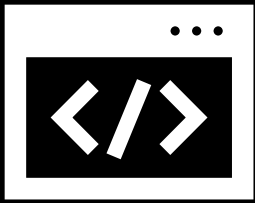
```
<ion-card>
  <ion-card-header>
    <ion-card-title>{{ contact.name?.given }} {{ contact.name?.family }}</ion-card-title>
  </ion-card-header>

  <ion-card-content>
    <ion-item lines="none">
      <ion-icon :icon="phonePortraitOutline" slot="start"></ion-icon>
      <ion-label>{{ contact.phones?.[0]?.number || 'Keine Telefonnummer' }}</ion-label>

      <!-- Icon zum Anrufen der Telefonnummer -->
      <ion-icon
        :icon="callOutline"
        slot="end"
        @click="callNumber(contact.phones?.[0]?.number ?? undefined)"
        style="margin-right: 10px;">
      </ion-icon>

      <!-- Icon zum Kopieren der Telefonnummer in die Zwischenablage -->
      <ion-icon
        :icon="copyOutline"
        slot="end"
        @click="copyToClipboard(contact.phones?.[0]?.number ?? undefined, 'Telefonnummer')">
      </ion-icon>
    </ion-item>
  </ion-card-content>
</ion-card>
```

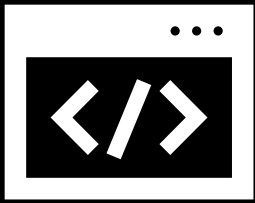
```
async getSingleContact(contactId: string) {
  try {
    const result = await Contacts.getContact({
      contactId: contactId,
      projection: {
        name: true,
        phones: true,
        emails: true,
        birthday: true,
      },
    });
    this.contact = result.contact;
  } catch (error) {
    console.error("Error loading contact", error);
    this.showErrorToast = true;
  }
},
```

Coding

Implementierung – Anrufen/E-Mail

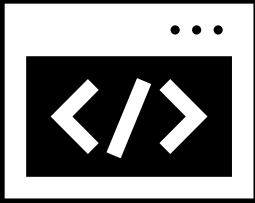
```
callNumber(number: string | undefined) {  
  if (number) {  
    window.open(`tel:${number}`, '_system');  
  } else {  
    this.showToast('Keine Telefonnummer verfügbar');  
  }  
},  
sendEmail(email: string | undefined) {  
  if (email) {  
    window.open(`mailto:${email}`, '_system');  
  } else {  
    this.showToast('Keine E-Mail-Adresse verfügbar');  
  }  
},
```



Coding

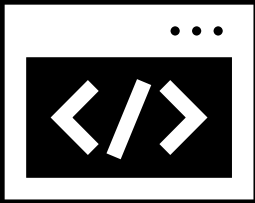
Implementierung –Zwischenablage

```
async copyToClipboard(text: string | undefined, type: 'Telefonnummer' | 'E-Mail-Adresse') {  
  if (text) {  
    try {  
      await Clipboard.write({  
        string: text  
      });  
      this.showToast(`${type} in die Zwischenablage kopiert`);  
    } catch (error) {  
      this.showToast(`Fehler beim Kopieren der ${type}`);  
    }  
  } else {  
    this.showToast(`Keine ${type} verfügbar`);  
  }  
},
```



Coding Implementierung – Toast

```
async showToast(message: string) {  
    await Toast.show({  
        text: message,  
        duration: 'short',  
        position: 'bottom'  
    });  
}
```



Coding Implementierung – Löschen

```
<ion-buttons slot="end">
  <ion-button fill="clear" @click="showDeleteAlert(contact.contactId)">
    <ion-icon :icon="trashBinOutline" size="large" color="light"></ion-icon>
  </ion-button>
</ion-buttons>
```

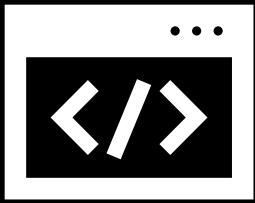
1

```
<ion-alert
  v-if="alertOptions"
  :is-open="isAlertOpen"
  :header="alertOptions.header"
  :message="alertOptions.message"
  :buttons="alertOptions.buttons"
  @didDismiss="handleAlertDismiss"
></ion-alert>
```

2

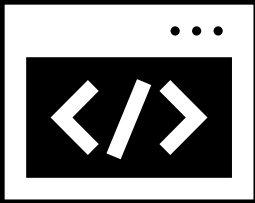
```
showDeleteAlert(contactId: string) {
  this.alertOptions = {
    header: "Kontakt löschen",
    message: "Möchten Sie diesen Kontakt wirklich löschen?",
    buttons: [
      {
        text: "Abbrechen",
        role: "cancel",
      },
      {
        text: "Löschen",
        handler: () => {
          this.deleteContact(contactId);
        },
      },
    ],
  };
  this.isAlertOpen = true;
},
```

3



Coding Implementierung – Löschen

```
async deleteContact(contactId: string) {  
  try {  
    await Contacts.deleteContact({  
      contactId: contactId,  
    });  
    this.router.replace({ name: 'Home' });  
  } catch (error) {  
    console.error("Error deleting contact", error);  
    this.showErrorToast = true;  
  }  
},
```

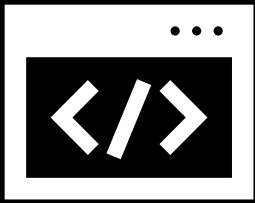


Coding

Implementierung – Einfügen

```
<ion-buttons slot="end">  
  <ion-button @click="showAddContactModal">  
    <ion-icon :icon="add" size="large"></ion-icon>  
  </ion-button>  
</ion-buttons>
```

```
showAddContactModal() {  
  this.isAddContactModalOpen = true;  
},
```

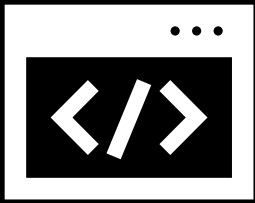


Coding Implementierung – Einfügen

```
<ion-modal :is-open="isAddContactModalOpen" @will-dismiss="closeAddContactModal">
  <ion-header>
    <ion-toolbar>
      <ion-buttons slot="start">
        <ion-button @click="closeAddContactModal">Abbrechen</ion-button>
      </ion-buttons>
      <ion-title>Neuer Kontakt</ion-title>
      <ion-buttons slot="end">
        <ion-button @click="createNewContact">Speichern</ion-button>
      </ion-buttons>
    </ion-toolbar>
  </ion-header>
  <ion-content>
    <ion-item>
      <ion-label class="floating_label">Vorname</ion-label>
    </ion-item>
    <ion-item>
      <ion-input v-model="newContact.given"></ion-input>
    </ion-item>
```

```
<ion-item>
  <ion-label class="floating_label">Telefonnummer</ion-label>
</ion-item>
<ion-item>
  <ion-input type="tel" v-model="newContact.phoneNumber"></ion-input>
</ion-item>
```

```
closeAddContactModal() {
  this.isAddContactModalOpen = false;
  this.newContact.phoneNumber = '';
  this.newContact.birthday = '';
  this.newContact.emailAddress = '';
  this.newContact.given = '';
  this.newContact.family = '';
},
```



Coding Implementierung – Einfügen

```
async createNewContact () {  
  const res = await Contacts.createContact({  
    contact: {  
      name: {  
        given: this.newContact.given,  
        family: this.newContact.family,  
      },  
      birthday: {  
        year: parseInt(this.newContact.birthday.slice(0,4)),  
        month: parseInt(this.newContact.birthday.slice(5,7)),  
        day: parseInt(this.newContact.birthday.slice(8,10)),  
      },  
      phones: [  
        {  
          type: PhoneType.Mobile,  
          label: 'mobile',  
          number: this.newContact.phoneNumber,  
        },  
      ],  
      emails: [  
        {  
          type: EmailType.Home,  
          label: 'private',  
          address: this.newContact.emailAddress,  
        },  
      ],  
    },  
  });  
}
```

```
await this.retrieveListOfContacts ();  
this.isAddContactModalOpen = false;  
this.newContact.phoneNumber = '';  
this.newContact.birthday = '';  
this.newContact.emailAddress = '';  
this.newContact.given = '';  
this.newContact.family = '';
```




Probleme & Lösungen



Probleme & Lösungen

- Zu Beginn kein Live Reload verwendet → Erhöhter Aufwand
→ Im Skript Lösung dafür gefunden
- Debugging konnte nicht aktiviert werden
→ Auf Debugging wurde verzichtet
- Entwicklung mit Browser war nicht möglich, da Erstellen von Kontakten in der Browseroberfläche nicht möglich (Contacts-Plugin nur für iOS und Android)
→ Entwicklung mit Emulator
- Routing: Funktionsweise und Übergabe von Parametern hat anfangs zu Problemen geführt
→ Konnte im späteren Verlauf behoben werden



**Was gab es zu
beachten?**



Was gab es zu beachten?

- Systemsprache beeinflusst die Darstellung des Geburtsdatums
- Für das Contacts-Plugin muss Version **v6.0.5-alpha.0** installiert werden (Vorgabe Genz)