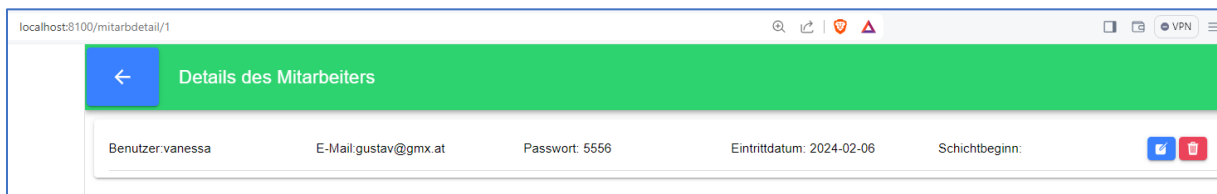
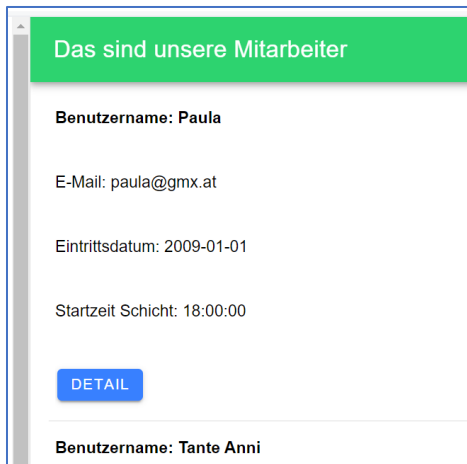


## Ionic 12: Ändern oder Löschen von Mitarbeitern

Testen und ansehen auf: <https://ionic.digbizmistelbach.info/login>

Die Liste der Mitarbeiter soll einen Button pro Mitarbeiter erhalten, der zur Detailsseite führt. Dort kann man zwischen 2 Buttons wählen:

- Editieren (ändern)
- Löschen



### 1) Button in die Liste (ngFor-Schleife einfügen)

Mitarbeiter

Öffne die mitarbeiter.page.html und füge in den „item“ der Liste den Button ein:

```
7 <ion-content [fullscreen]="true">
8   <ion-list>
9     <ion-item lines="insert" *ngFor="let mitarbeiter of mitarbeiters">
10      <ion-label><h5>E-Mail: {{mitarbeiter.mail }}</h5></ion-label><br>
11      <ion-label><h5>Passwort: {{mitarbeiter.passwort }}</h5></ion-label><br>
12      <ion-button color="primary">Detail</ion-button>
13    </ion-item>
14  </ion-list>
```

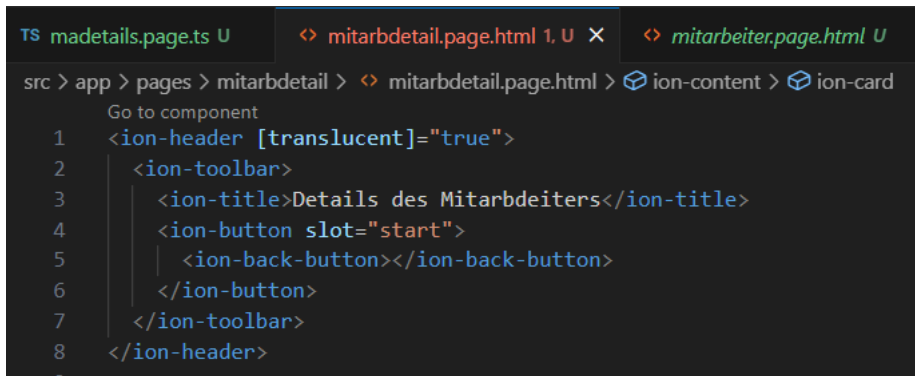
<ion-button color="primary">Detail</ion-button>



#### 4)mitarbdetail.page.html

Öffne die mitarbdetail.html und lass die Details zu der Auswahl des Mitarbeiters ausgeben.

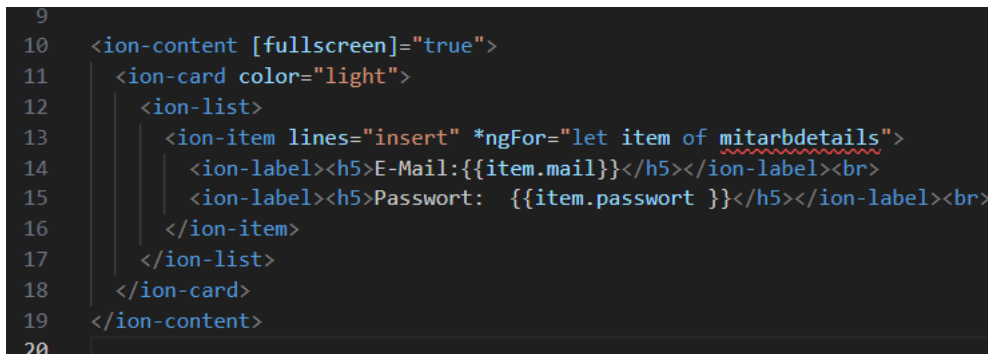
```
<ion-button slot="start">
  <ion-back-button></ion-back-button>
</ion-button>
```



```
TS madetails.page.ts U  <> mitarbdetail.page.html 1, U X  <> mitarbeiter.page.html U
src > app > pages > mitarbdetail > <> mitarbdetail.page.html > ion-content > ion-card
Go to component
1  <ion-header [translucent]="true">
2    <ion-toolbar>
3      <ion-title>Details des Mitarbeiters</ion-title>
4      <ion-button slot="start">
5        <ion-back-button></ion-back-button>
6      </ion-button>
7    </ion-toolbar>
8  </ion-header>
```

Drunter im „content“ die Liste:



```
<ion-card color="light">
  <ion-list>
    <ion-item lines="insert" *ngFor="let item of mitarbdetails">
      <ion-label><h5>E-Mail:{{item.mail}}</h5></ion-label><br>
      <ion-label><h5>Passwort: {{item.passwort }}</h5></ion-label><br>
    </ion-item>
  </ion-list>
</ion-card>
```



```
9
10 <ion-content [fullscreen]="true">
11   <ion-card color="light">
12     <ion-list>
13       <ion-item lines="insert" *ngFor="let item of mitarbdetails">
14         <ion-label><h5>E-Mail:{{item.mail}}</h5></ion-label><br>
15         <ion-label><h5>Passwort: {{item.passwort }}</h5></ion-label><br>
16       </ion-item>
17     </ion-list>
18   </ion-card>
19 </ion-content>
20
```

ICONS findet man zum Kopieren auf: <https://ionic.io/ionicons>



Darunter, vor dem Ende des „item“ soll nun ein Button eingefügt werden mit einem ICON. Wähle zuerst für das Editieren

- `<ion-icon name="create"></ion-icon>`  

und dann für

- das Löschen:

```
<ion-icon name="trash"></ion-icon>
```

```

<ion-button color="primary" routerLink="/mitarbdetail/{{item.iduser}}">
  <ion-icon name="create"></ion-icon>
</ion-button>
<ion-button color="danger" routerLink="/mitarbdetail/{{item.iduser}}">
  <ion-icon name="trash"></ion-icon>
</ion-button>

```

```

15     <ion-label><h5>Passwort: {{item.passwort }}</h5></ion-label><br>
16     <ion-button color="primary" routerLink="/mitarbdetail/{{item.iduser}}">
17       <ion-icon name="create"></ion-icon>
18     </ion-button>
19     <ion-button color="danger" routerLink="/mitarbdetail/{{item.iduser}}">
20       <ion-icon name="trash"></ion-icon>
21     </ion-button>
22   </ion-item>
23 </ion-list>

```

### 5) mitarbdetails.page.ts

Die unterstrichenen „mitarbdetails“ der ngFor-Schleife muss nun noch aufgelöst werden in der „export class“ der „.ts“:

mitarbdetails: any;

```

8   export class MitarbdetailPage implements OnInit {
9     mitarbdetails: any;
10
11   constructor() { }

```

Man kann noch keine Ergebnisse sehen, weil die API noch fehlt.

Daher muss man hier nun die Funktion für die Übergabe an die apiService erstellen. Das ist eine einfache GET Abfrage in der API.

- Zuerst im constructor die beiden Methoden anlegen, die wir gleich benötigen werden; mit passendem IMPORT dazu.

```

private route: ActivatedRoute,
private apiService: ApiService

```

```

13   constructor(
14     private route: ActivatedRoute,
15     private apiService: ApiService
16   ) {}

```

```

1   import { Component, OnInit } from '@angular/core';
2   import { ActivatedRoute } from '@angular/router';
3   import { ApiService } from 'src/app/api.service';
4

```

- In die „ngOnInit“ die get(id) und die Verknüpfung zur „apiService“.  
Man kann sich an der bereits vorhandenen Quelle bei den Details bedienen und dort kopieren und ändern.

```
const id = this.route.snapshot.paramMap.get('id');
// console.log(id);
this.apiService.getMitarbDetail(id).subscribe((response: any) => {
  this.mitarbdetails = response;
});
```

```
17
18   ngOnInit() {
19     const id = this.route.snapshot.paramMap.get('id');
20     // console.log(id);
21     this.apiService.getMitarbDetail(id).subscribe((response: any) => {
22       this.mitarbdetails = response;
23     });
24   }
25 }
```

Da das apiService noch nicht dazu bereit ist, gibt es diese Fehlermeldung. Die wird sich aber bald auflösen. 😊

## 6)apiService.ts

Öffne die apiService.ts und kopiere die GET von „getMaDetail“ und ändere sie nur beim Namen und dem Endpunkt:

```
54
55   getMitarbDetail(id: any) {
56     return this.http.get('https://ionic.digbizmistelbach.info/api_einkaufsliste/public/index.php/mitarbdetail/'+id);
57   }
58 }
```

```
getMitarbDetail(id: any) {
  return
this.http.get('https://ionic.digbizmistelbach.info/api_einkaufsliste/public/index.php/mitarbdetail/'+id
);
}
```

## 7)API anpassen

Öffne die „index.php“ in der API und kopiere den Code von „manager{id}“ und passe ihn an:

```
$app->get('/mitarbdetail/{id}', function( Request $request, Response $response){
  $id = $request->getAttribute('id');
  $sql = "SELECT * FROM user WHERE iduser = '$id' ";

  try {
    $db = new db();
    $db = $db->connect();
    $stmt = $db->query( $sql );
    $user = $stmt->fetchAll( PDO::FETCH_OBJ );
    $db = null; // clear db object
```

```

// print out the result as json format
echo json_encode( $user );

} catch( PDOException $e ) {
// show error message as Json format
echo '{"error": {"msg": ' . $e->getMessage() . '}}';
}
});

```

```

226 $app->get('/mitarbdetail/{id}', function( Request $request, Response $response){
227     $id = $request->getAttribute('id');
228     $sql = "SELECT * FROM user WHERE iduser = '$id' ";
229
230     try {
231         $db = new db();
232         $db = $db->connect();
233         $stmt = $db->query( $sql );
234         $user = $stmt->fetchAll( PDO::FETCH_OBJ );
235         $db = null; // clear db object
236
237         // print out the result as json format
238         echo json_encode( $user );
239
240     } catch( PDOException $e ) {
241         // show error message as Json format
242         echo '{"error": {"msg": ' . $e->getMessage() . '}}';
243     }
244 });

```

Ergebnis:

