

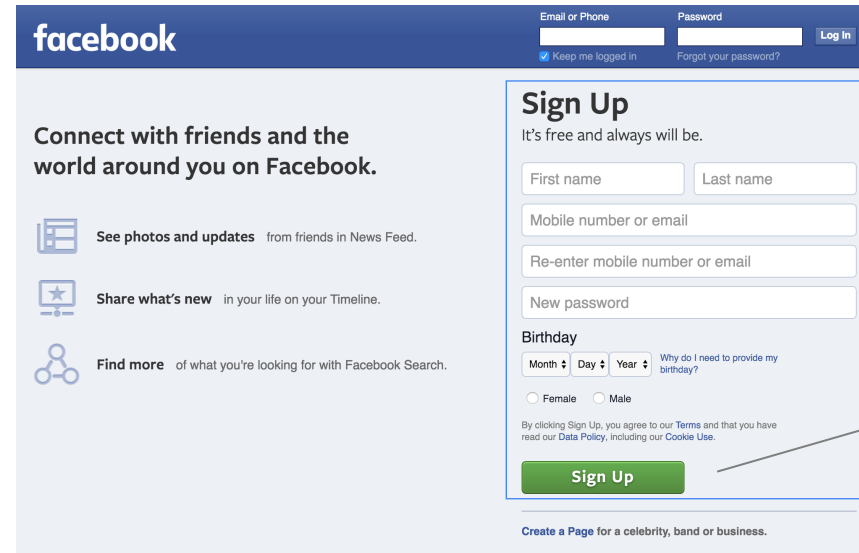
# WEBTECHNOLOGIEN

## 05 – FORMULARE

PROF. DR. MARKUS HECKNER

# HTML FORMULARE ERMÖGLICHEN DIE ERFASSUNG UND WEITERGABE VON NUTZEREINGABEN

- Interaktionsmöglichkeit für Nutzer mit Webseiten
- Daten werden an Webserver zur weiteren Verarbeitung übermittelt (z.B. als Werte für Abfragen an Datenbanken, Überprüfung von Login-Daten, Bestellen von Artikeln bei Amazon, usw.)
- Formulare bestehen aus HTML Elementen zur Eingabe (Text, Radiobuttons, Checkboxes, ...) und zum Absenden des Formulars (Button)

A screenshot of the Facebook 'Sign Up' page. The page has a blue header with the Facebook logo and login options. The main content area is light blue and contains the 'Sign Up' heading, a subtext 'It's free and always will be.', and several input fields: 'First name', 'Last name', 'Mobile number or email', 'Re-enter mobile number or email', and 'New password'. Below these are 'Birthday' selection fields (Month, Day, Year) and gender options (Female, Male). A green 'Sign Up' button is at the bottom of the form. To the left of the form, there are links for 'See photos and updates', 'Share what's new', and 'Find more'. At the very bottom, there is a link to 'Create a Page for a celebrity, band or business.'

Formular mit  
Feldern

Übermitteln  
der Daten an  
den Server

# FORMULAR IM BROWSER ZUM HINZUFÜGEN EINES NEUEN SONGS

# HTML CODE DES FORMULARS (OHNE BOOTSTRAP)

Was passiert, beim Absenden des Formulars?

Hier aufrufen einer parametrisierten Route (mehr später), um einen Neuen Song hinzuzufügen

http-Methode POST (wird im Router wichtig)

```
<form action="/playlist/1/addsong" method="POST">
```

```
  <label>Title</label> <input name="title" type="text" placeholder="Title" />
```

```
  <label>Artist</label> <input name="artist" type="text" placeholder="Artist" />
```

```
  <label>Duration</label> <input name="duration" type="number" placeholder="00" />
```

```
  <button type="submit">Add Song</button>
```

```
</form>
```

Eingabefelder

Schickt das Formular ab, d.h. die in action definierte URL wird mit der in method definierten http- Methode aufgerufen

name ermöglicht es dem Server die Eingaben aus dem Formular auszulesen

Welche Daten werden eingegeben?

# HTTP REQUEST HEADER – KURZ UND KNAPP

Wie werden die Daten  
übermittelt? Hier: POST als Teil  
des headers...

## ▼ Request Headers

**:authority:** playlist-2.mheckner.repl.co  
**:method:** POST  
**:path:** /playlist/2/addsong  
**:scheme:** https  
**accept:** text/html,application/xhtml+xml,application/;  
\*;q=0.8,application/signed-exchange;v=b3;q=0.9  
**accept-encoding:** gzip, deflate, br  
**accept-language:** en-US,en;q=0.9,de;q=0.8,la;q=0.7

Welchen Pfad wollen wir?

## ▼ Form Data    view parsed

title=Song+2&artist=Blur&duration=122

Hier stehen die Daten aus dem  
Formular als Key-Value Paare...  
Key entspricht dem name-Attribut  
des Formulars!

# AUSLESEN UND DARSTELLEN DER DATEN AUF DEM SERVER (1/2)

```
async addSong(request, response) {  
  const playlistId = request.params.id;  
  const newSong = {  
    title: request.body.title,  
    artist: request.body.artist,  
    duration: Number(request.body.duration)  
  };  
  logger.debug("New Song", newSong);  
  await songStore.addSong(playlistId, newSong);  
  response.redirect("/playlist/" + playlistId);  
},
```

body ist der Inhalt des Requests und enthält die geparsen name-Attribute als Keys – Inhalte sind die Eingaben in den Formularfeldern

# BODY-PARSER WIRD BENÖTIGT, UM DIE FORMULARDATEN AM SERVER AUSZULESEN

requiren des Moduls in app.js, und ...

app.js:

```
const bodyParser = require("body-parser");
```

```
app.use(bodyParser.urlencoded({ extended: false }));
```

... konfigurieren des Moduls