

# Mini-Blog

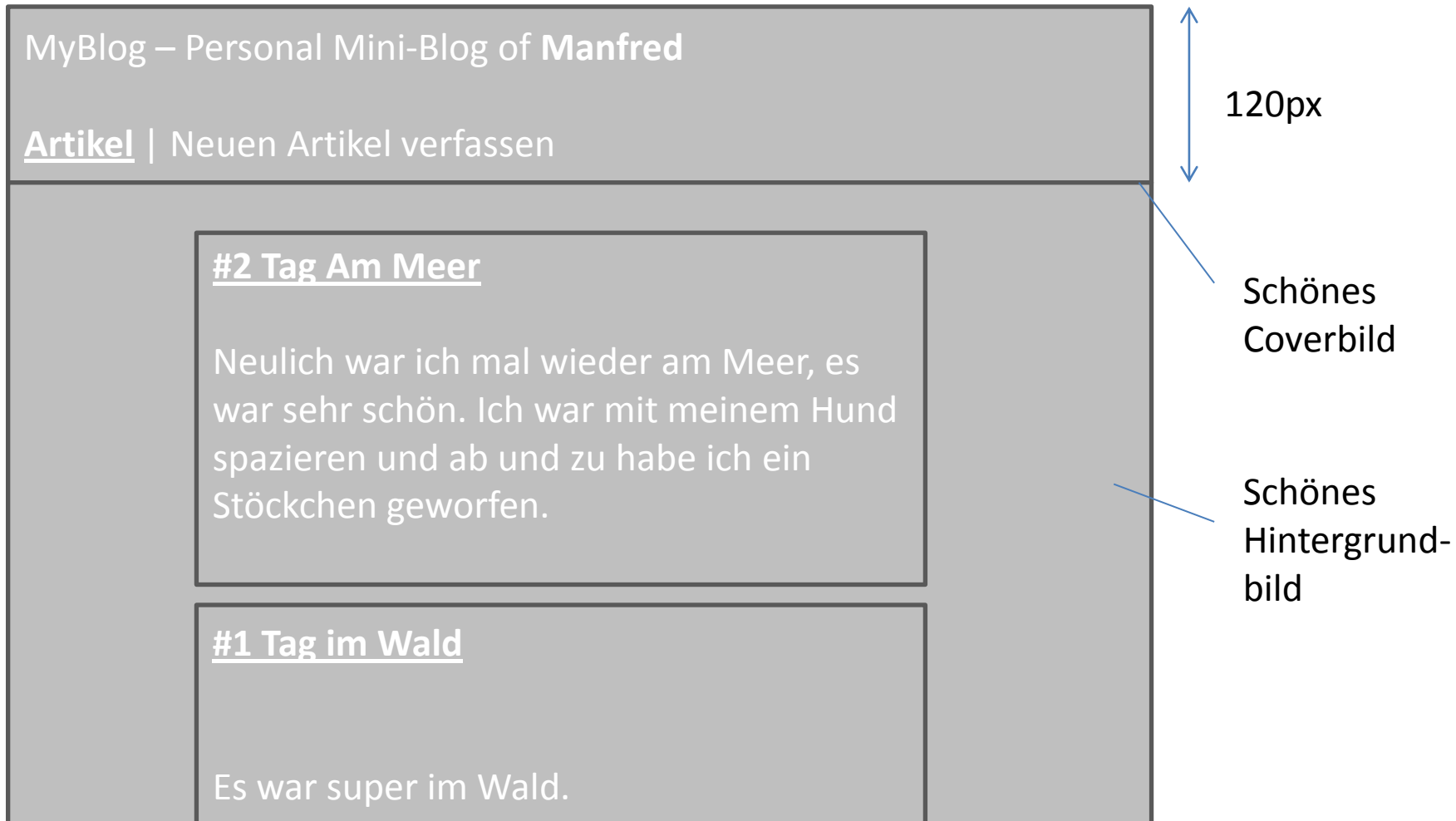
[jan.schulz@digitalcareerinstitute.org](mailto:jan.schulz@digitalcareerinstitute.org)

# Task 1: NodeJS-Fullstack Server

1. Erstelle einen NodeJS-Express-Server der statische Ressourcen im /public-Verzeichnis hostet.
2. Installiere das MySQL-NodeJS Modul.
3. Erstelle eine Basic-Website mit index.html, main.js und style.css.
4. Gib „Hallo World“ aus.

# Task 2: Layout

Implementiere folgendes Layout!



# Task 2: Layout

Implementiere folgendes Layout!

MyBlog – Personal Mini-Blog of Manfred

Artikel | Neuen Artikel verfassen

Titel hier eingeben

Hier einen Text eingeben

Artikel erstellen

120px

Schönes Coverbild

Schönes Hintergrundbild

# Task 3: Datenbank

1. Erstelle eine MySQL-Datenbank “miniblog” mit einer Tabelle “blogpost”. Sie soll 3 Felder haben:
  1. id (autoincrement)
  2. created (timestamp, Zeitpunkt der Erstellung)
  3. title (varchar(50))
  4. content (varchar(1024))
2. Erstelle 3 Beispiel-Blogposts in Deiner Datenbank.

# Task 4: Backend

1. In Deinem NodeJS-Server, erstelle eine Resource GET /blogposts, die alle Blogposts zurückgibt.
2. Erstelle eine Resource POST /blogposts, über die der Nutzer neue Blogposts erstellen kann.
3. Teste GET /blogposts mit Chrome und POST /blogpost mit Postman.

# 5. Frontend

1. Implementiere die Logik zum Darstellen und Erstellen von Blogposts selbstständig auf Deinem Frontend.

# 6. Authentifizierung

1. Erweitere das Frontend so, dass ein neuer Navigationspunkt „Login“ zu sehen ist. Beim Login soll der Nutzer per Textbox seine Email und ein Password eingeben können.
2. Erweitere Deine MySQL-Datenbank mit der Tabelle „Users“ und erstelle 3 Testnutzer mit einr Email und einem Password.
3. Erweitere Dein Backend um zwei POST-Ressourcen „/login“ und „/logout“, bei denen sich die Nutzer per Authentifizierung an- und abmelden können.
4. Die POST-Resource /blogpost soll nun nur noch für authentifizierte Nutzer ansprechbar sein.
5. Falls sich der Nutzer erfolgreich angemeldet hat, soll der Menüpunkt „Artikel verfassen“ angezeigt werden. Falls er nicht eingeloggt ist, soll „Artikel verfassen“ nicht angezeigt werden.
6. Was passiert, wenn der Nutzer die Seite neu lädt? Versuche, den Status des Eingeloggt-Seins im LocalStorage zu speichern und das Frontend immer korrekt darzustellen!