



BIB INTERNATIONAL COLLEGE

LERNAUFGABE 2020

bib now

Charlie Wiegand
Jawoon Kim

betreut von Frau Langeheinecke

30. September 2020

Inhaltsverzeichnis

Abbildungsverzeichnis	2
1 Über bibnow	1
2 Aufgabenstellung	2
3 Konzepte	3
3.1 Inhaltskonzept	3
3.2 Navigationskonzept	4
3.3 Gestaltungskonzept	5
3.4 Zielgruppendefinition	6
4 Durchführung	7
4.1 Allgemeine Beschreibung zur Durchführung	7
4.2 Planung und Vorbereitung	7
4.2.1 Vor dem Projekt	7
4.2.2 Während des Projekts	7
5 Ergebnis	8
5.1 GitHub-Pfad	8
5.2 Loginseite	8
5.2.1 Desktop Ansicht	9
5.2.2 Mobile Ansicht	10
5.3 Dashboard	13
5.3.1 Desktop Ansicht	14
5.3.2 Mobile Ansicht	15
5.4 Datenbank	16
6 Fazit	17
6.1 Mögliche Erweiterung	17
7 Quellen	18

Abbildungsverzeichnis

1	Sitemap	3
2	Navigationskonzept	4
3	Schriftart 100	5
4	Schriftart 300	5
5	Schriftart 700	5
6	Farbenschema	6
7	Authentifizierung über Firebase	8
8	Screenshot LandingPage Desktop	9
9	Screenshot Registrierung Desktop	9
10	Screenshot Anmeldung Desktop	10
11	Screenshot LandingPage Mobil	10
12	Screenshot Registrierung Mobil	11
13	Screenshot Anmeldung Mobil	12
14	ContentViewer	13
15	Screenshot Dashboard	14
16	Screenshot Postdialog	14
17	Screenshot Dashboard	15
18	Screenshot Postdialog	16
19	ERM	17

bib now

Jawoon Kim, Charlie Wiegand

30. September 2020

1 Über bibnow

bib-now, das Forum, das exclusive Forum für alle Studierenden und Mitarbeiter.

Erstellt von Jawoon Kim und Charlie Wiegand(PBT3H19A).

Letzte Revision: September 2020

Note

Diese Version enthält die Dokumentation der Webseite und Erläuterungen. Sie dient der Einarbeitung in bib-now sowie dem Verständnis der Konzepte.

2 Aufgabenstellung

Da zur Zeit häufig Mails an die ganze Schule geschickt werden, wenn beispielsweise jemand etwas verloren hat, wollen wir unseren Mitschülern und uns ein Forum bieten, das diesen Austausch erleichtert.

Für dieses Vorhaben haben wir uns folgende Ziele und Richtlinien gesetzt

- **Minimum** Es soll eine lauffähige Webseite erstellt werden, auf der man sich mindestens einloggen kann und anschließend Beiträge in Textform hochladen kann.
- **Optional** Es soll die Möglichkeit geben seine Beiträge durch Bilder zu ergänzen und die Posts sollen in Kategorien eingeteilt werden
- **Optional** Desweiteren soll es die Möglichkeit den Stundenplan und Klausurplänung aus dem offiziellen Intranet zu übernehmen.

Voraussetzungen Um das Minimum zu erreichen benötigen wir einen PC, einen Texteditor, npm, ein AngularFramework und Firebase. Um uns die Zusammenarbeit zu erleichtern haben wir uns entschieden Versionskontrolle über git laufen zu lassen und die Ergebnisse über GitHub auszutauschen.

3 Konzepte

3.1 Inhaltskonzept

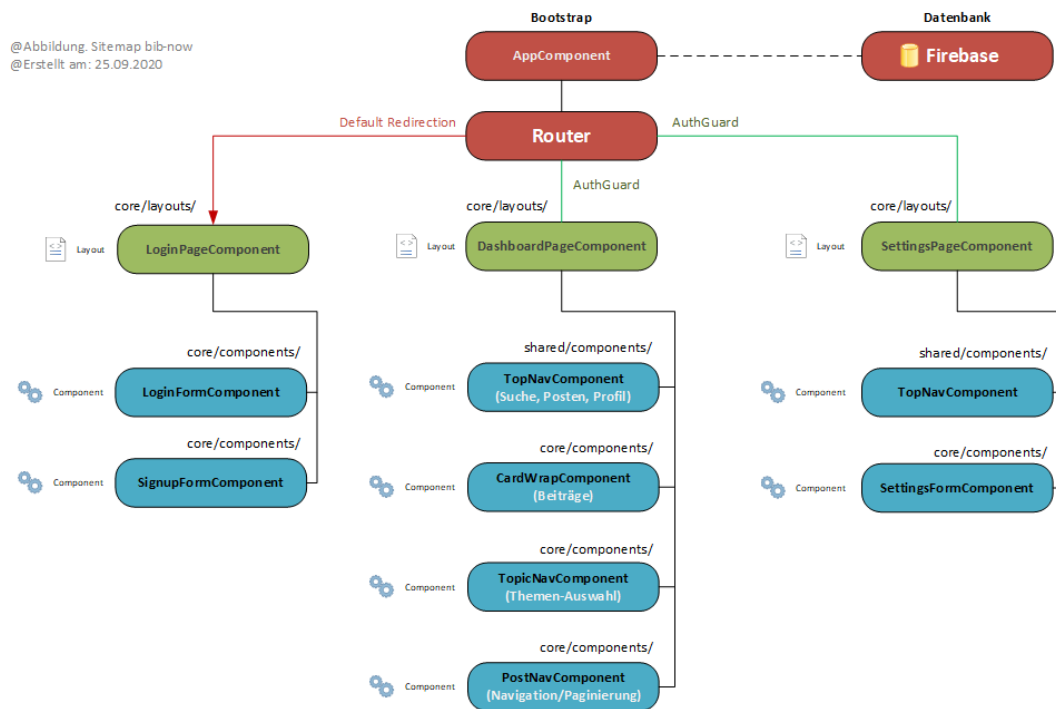


Abbildung 1: Sitemap

Bei bib-now handelt es sich um ein Schülerforum für das bib international College Paderborn. Die technische Umsetzung basiert auf einer Single-Page-Webanwendung mit einer RealTime-Datenbank deren einzelne Komponenten über einen Router verlinkt sind: Die Navigation besteht aus drei Komponenten: **LoginPageComponent**, **DashboardPageComponent** und der **SettingsPageComponent**. Jede dieser Komponenten besitzt ihre eigenen Backend-Komponenten. Die **LoginPage** greift auf die **LoginFormComponent** und **SignUpFormComponent** zurück. Die **DashboardPageComponent** greift auf **SideNavComponent**, **ForumComponent**, **PostComponent** zurück. Außerdem können hier später weitere Elemente ergänzt werden. **SettingsPageComponent** beinhaltet die **SideNavComponent** und **SettingsFormComponent**.

3.2 Navigationskonzept

@Abbildung. Navigationskonzept
@Erstellt am: 25.09.2020

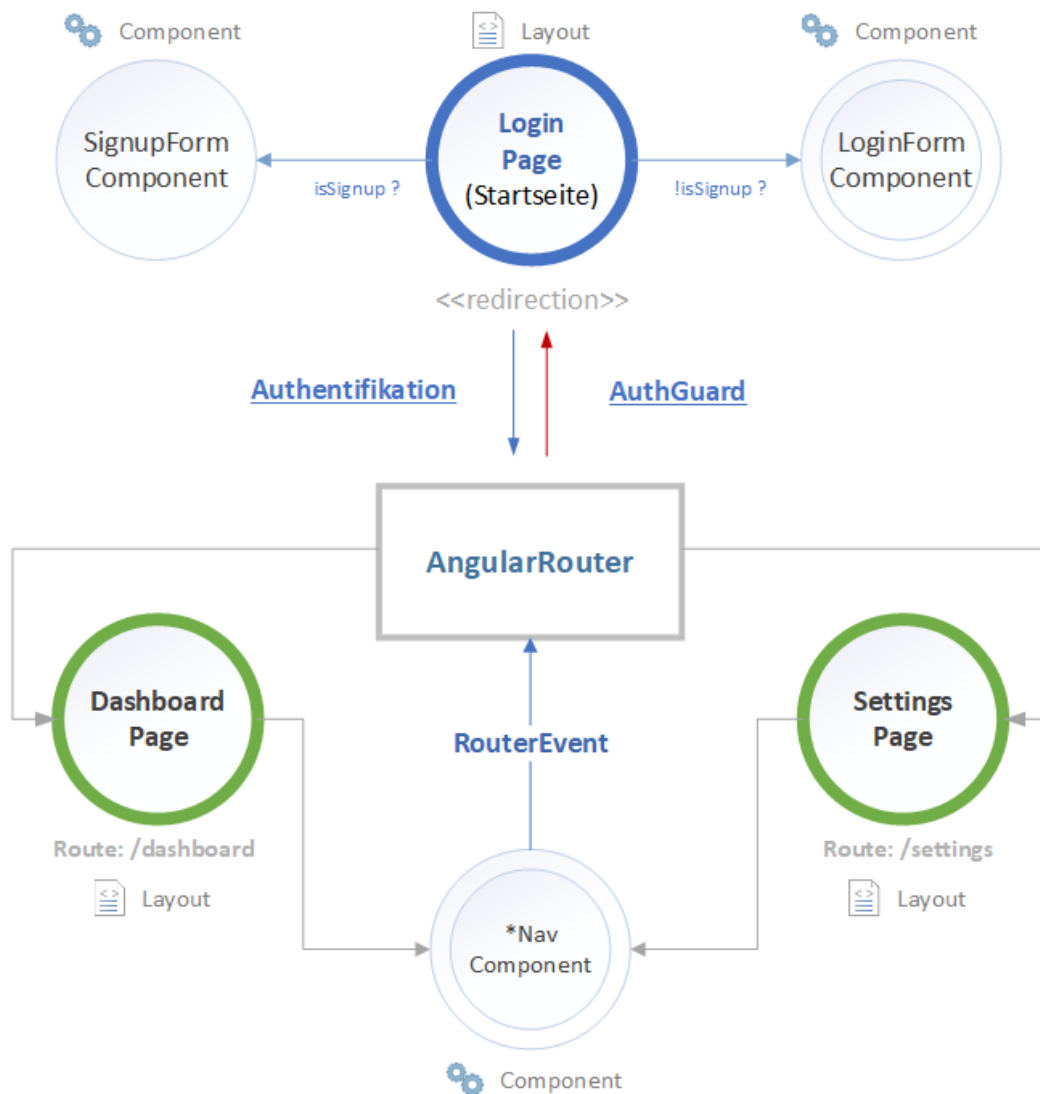


Abbildung 2: Navigationskonzept

Beim Betreten der Seite wird der User zunächst aufgefordert sich einzuloggen. Sollte er sich noch nicht registriert haben hat er die Möglichkeit in das Registrierungsformular zu switchen und sich dort zu registrieren. War die Registrierung oder das einloggen erfolgreich, wird der User auf die DashboardPage weitergeleitet. Dort werden ihm Posts angezeigt und er hat die

Möglichkeit über ein Formular einen eigenen Post zu verfassen. Außerdem kann er über das Navigationsmenu zu der SettingsPage zu wechseln. Seine aktuelle Position wird durch eine Farbveränderung am jeweiligen punkt in der Navigatonsleiste erkenntlich.

3.3 Gestaltungskonzept

Font: Montserrat (<https://fonts.google.com/specimen/Montserrat>) Font Weights: 100, 300, 700

- Für Subtitel, gedämpfte Texte, etc

Thin 100

Die Schriftart für bib-now

Abbildung 3: Schriftart 100

- Für normale Texte, Standard-schriftart

Light 300

Die Schriftart für bib-now

Abbildung 4: Schriftart 300

- Für Titel und Akzent(Betonung)

Bold 700

Die Schriftart für bib-now

Abbildung 5: Schriftart 700

Farbenschema

Schemakonzept: Neon + Material

Primärfarbe:#D43D61

Sekundär:#039BE5

Hintergrund:#271D33

Hintergrund Beiträge:#F8F6F6

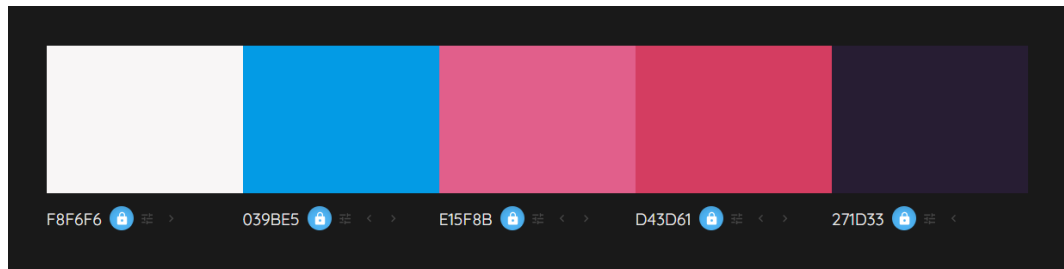


Abbildung 6: Farbenschema

3.4 Zielgruppendefinition

Aus den MUSS - und KANN-Zielen ergibt sich unsere Zielgruppe: alle Schüler und Mitarbeiter des bib international College Paderborn.

4 Durchführung

4.1 Allgemeine Beschreibung zur Durchführung

Beim Entwickeln der Webseite haben wir uns täglich Abgesprochen welche Features wir als nächstes brauchen und diese in Form von Backend und Frontend unter uns aufgeteilt. Da wir über GitHub gearbeitet haben konnten wir beide unsere branches pushen und mit dem Master-Branch auf konfliktfreie Kompatibilität prüfen und anschließend testen.

4.2 Planung und Vorbereitung

4.2.1 Vor dem Projekt

Corona bedingt wurden wir erst recht spät über die LEA informiert. Wir haben die Zeit die wir dennoch noch zur Vorbereitung hatten, genutzt um uns über Nutzung des Frameworks Angular sowie die dortige implementierung von Firebase Funktionen zu informieren. Dies geschah in Form von diversen Youtube-Videos sowie dem Durchstöbern der jeweiligen Dokumentationen.

Wir sind davon ausgegangen, dass wir eine lauffähige Webseite mit den minimal Anforderungen in den ersten Tagen realisieren können und uns anschließend mit den optionalen Zielen und der Dokumentation beschäftigen können.

4.2.2 Während des Projekts

Zu Beginn haben wir die Dateistruktur sowie den Ablauf abgesprochen. Wir haben uns darauf geeinigt, dass wir die Anwendung der Seite zurnächst in Core und Shared unterteilen. Der Core-Ordner soll außerdem in Components und Layouts unterteilt werden, um eine physische Trennung von Logik und Frontend zu erzielen. Der Shared-Ordner soll alle restlichen Komponenten enthalten unterteilt in ihre jeweiligen Oberkategorien: Models, Guards, Services, Module und Components die in das Core Modell integriert werden.

Außerdem haben wir am Anfang der LEA einen Plan erstellt, wann welches Modul erledigt sein soll. Da das jedoch ein sehr steifes Modell war und überhaupt nicht zu den flexiblen Problemen in der LEA gepasst hat, haben wir das Modell schussendlich nur als grobe Richtlinie gesehen und unsere Planung für den Tag an die gerade benötigten Aufgaben angepasst.

5 Ergebnis

5.1 GitHub-Pfad

Der Source-Code zum Ergebnis lässt sich unter folgendem Pfad finden:
<https://github.com/bib-lea/bib-now>

5.2 Loginseite

Auf der Loginseite hat der Nutzer drei mögliche Ansichten zwischen denen er hin und her wechseln kann. Eine Auswahl zwischen Anmeldung und Registrierung in Form von Buttons, ein Registrierformular und ein Anmeldeformular.

Beim Abschicken des Registrierungsformular wird ein Nutzer in der Datenbank erzeugt. Beim Abschicken des Anmeldeformulars wird geprüft ob der Nutzer mit diesen Daten in der Datenbank hinterlegt ist. Ist das der Fall ändert sich der Authentifizierungsstatus und der Nutzer wird zur Dashboardseite weitergeleitet. Ist das nicht der Fall bleibt der Nutzer auf der Loginseite.

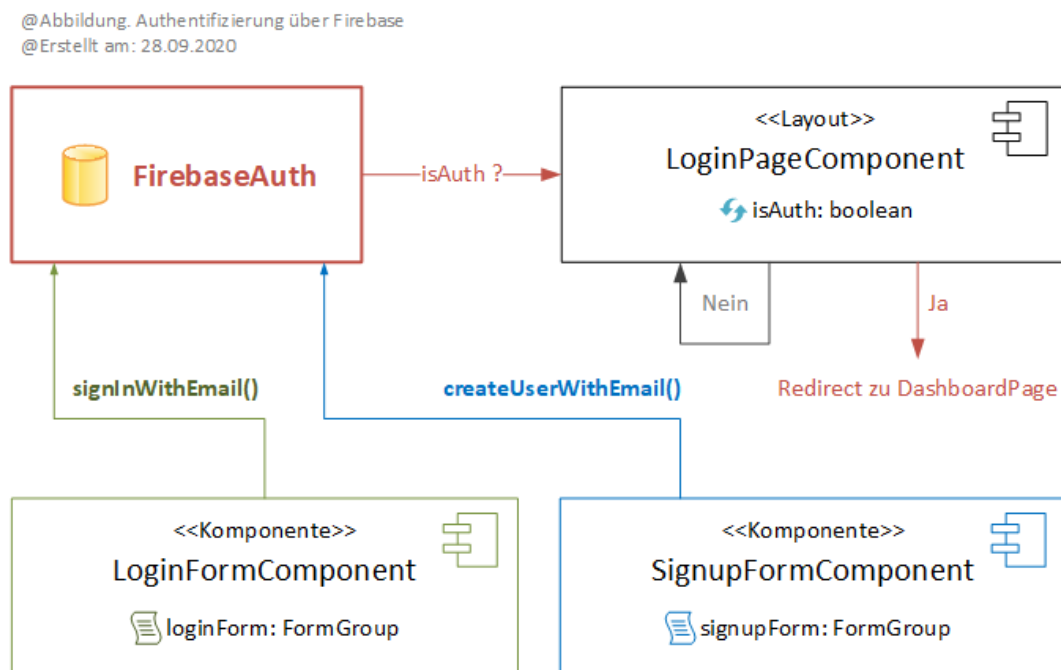


Abbildung 7: Authentifizierung über Firebase

5.2.1 Desktop Ansicht

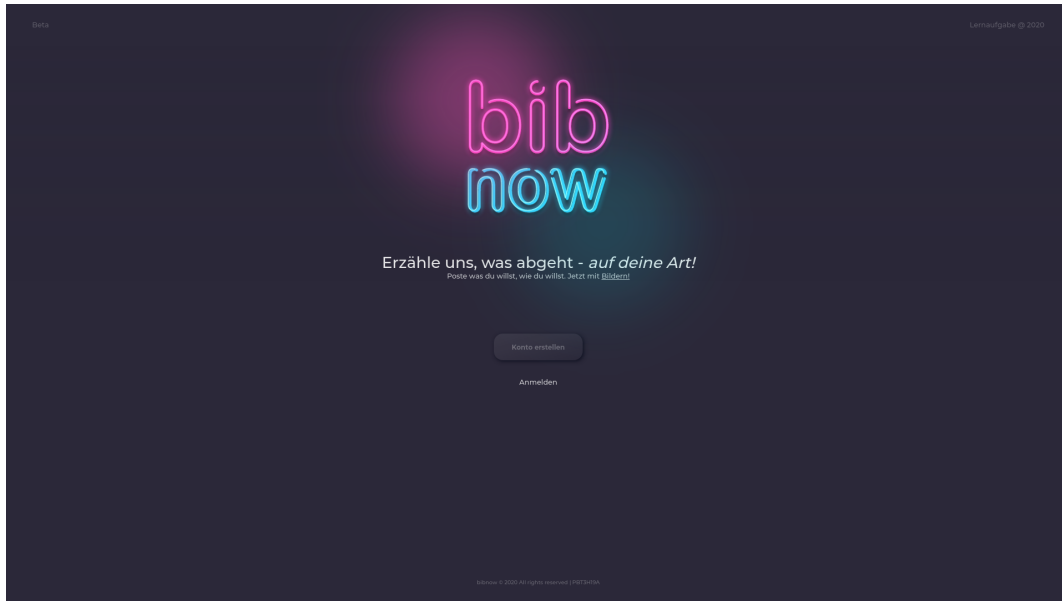


Abbildung 8: Screenshot LandingPage Desktop

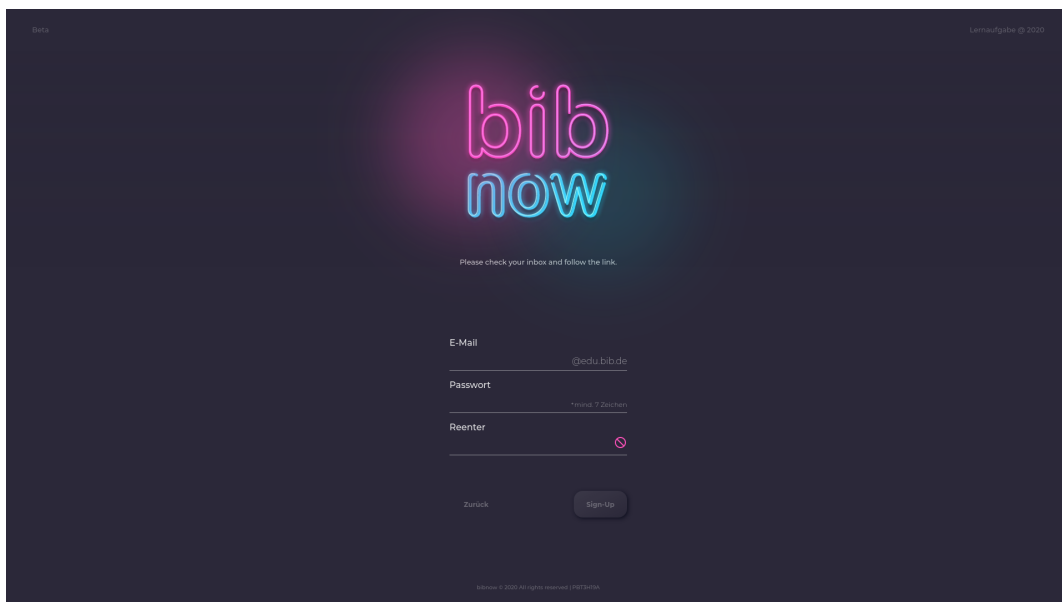


Abbildung 9: Screenshot Registrierung Desktop

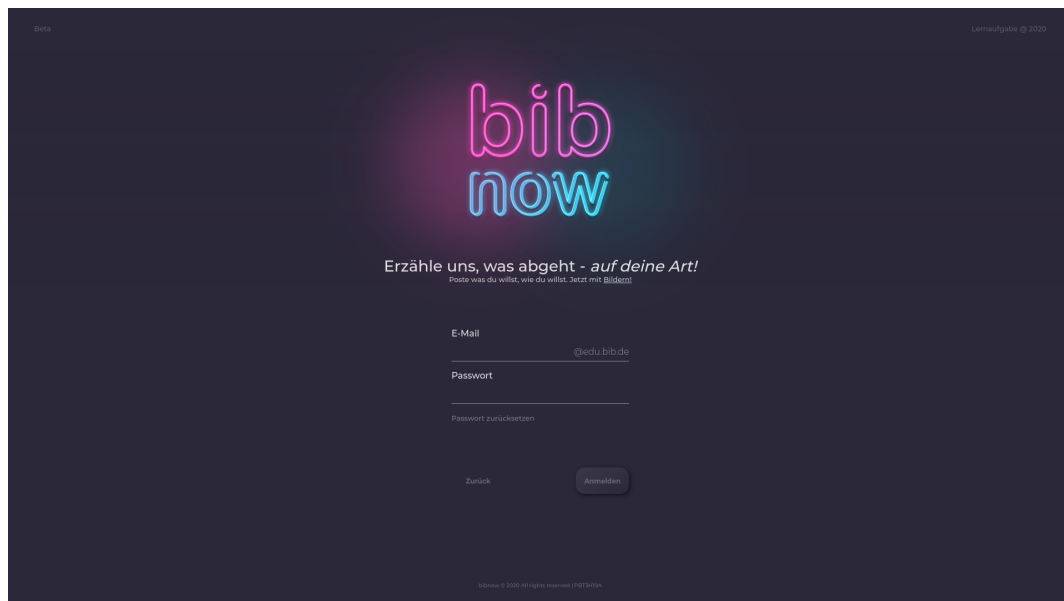


Abbildung 10: Screenshot Anmeldung Desktop

5.2.2 Mobile Ansicht

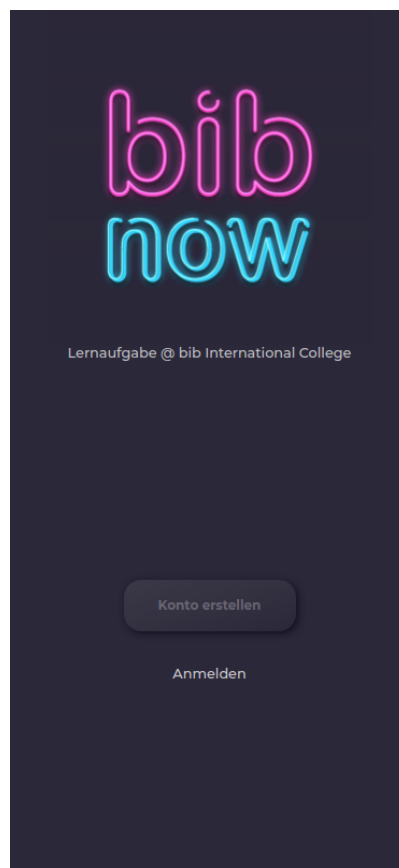


Abbildung 11: Screenshot LandingPage Mobil

**bib
now**

Lernaufgabe @ bib International College

E-Mail

Passwort

Reenter

[Zurück](#) [Sign-Up](#)

Abbildung 12: Screenshot Registrierung Mobil

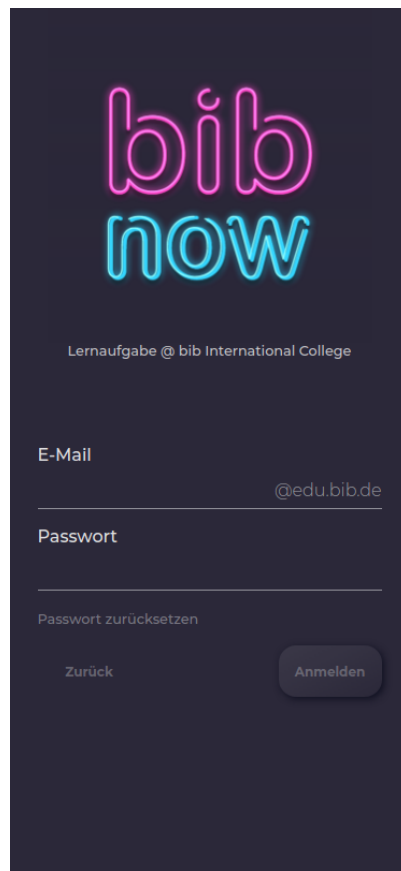


Abbildung 13: Screenshot Anmeldung Mobil

5.3 Dashboard

Auf dem Dashboard ist eine Navigationsleiste, die ContentViewer Komponente und der Footer zu sehen. In der Navigationsleiste ist das Logo, eine Suchfunktion, die nach Schlüsselworten in den Titeln und Inhalten der Beiträge sucht, der Button um den Postdialog zu öffnen, ein Auslogg-Button, der den Nutzer wieder zur Einloggseite führt. Die ContentViewer Komponente zeigt oben eine Auswahl zwischen Fundbüro und Tutorium, um Posts der jeweiligen Kategorie anzuzeigen und unten eine Pagination um sich weitere Posts anzusehen. Die CardWrap Komponente im ContentViewer zeigt immer fünf Posts, wobei der aktive immer durch weiße Hervorhebung markiert ist. Zu dem hat der Nutzer die Möglichkeit den Post zu löschen, sofern er der Autor ist. Dies ist durch ein Mülleimer-Icon markiert.

@Abbildung. Struktur des Content-Viewers(für das Anzeigen der Beiträge zuständig)
@Erstellt am: 28.09.2020

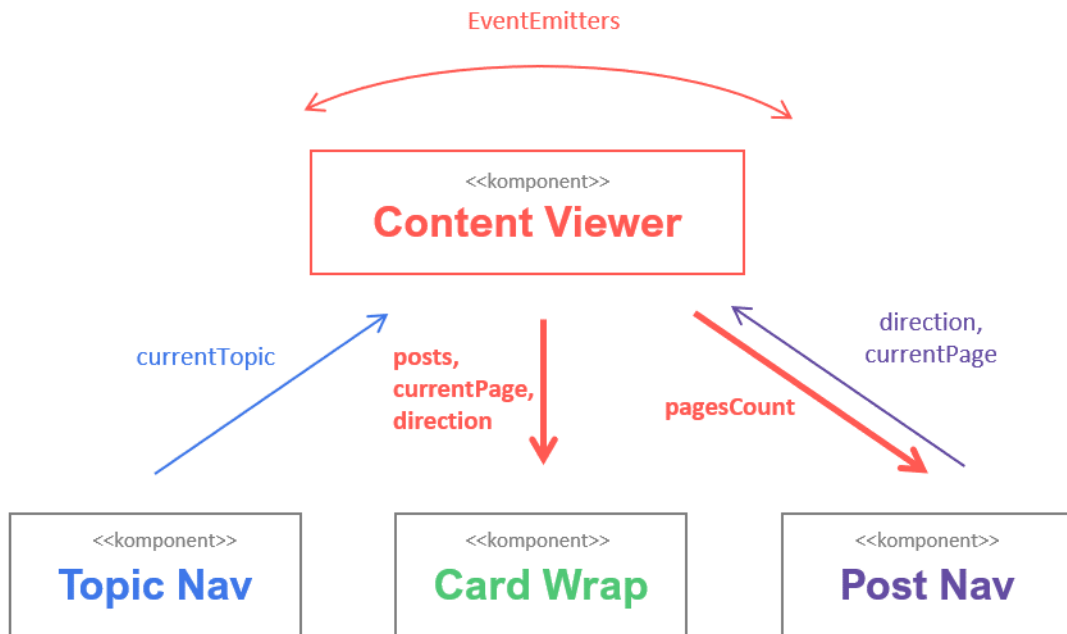


Abbildung 14: ContentViewerQuelle: eigene Darstellung

Betätigt der Nutzer den Posten-Button öffnet sich ein Formular in dem der Nutzer sich für ein Oberthema sowie Untertypen entscheiden kann. Zudem müssen Titel und Inhalt ausgefüllt sein, damit die Daten in die Datenbank geladen werden. Ist das nicht der Fall, wird der Nutzer dazu aufgefordert dies zu tun. Die Zugabe eines Bildes ist optional. Gibt der Nutzer kein Bild an, wird ein Demobild verwendet.

5.3.1 Desktop Ansicht

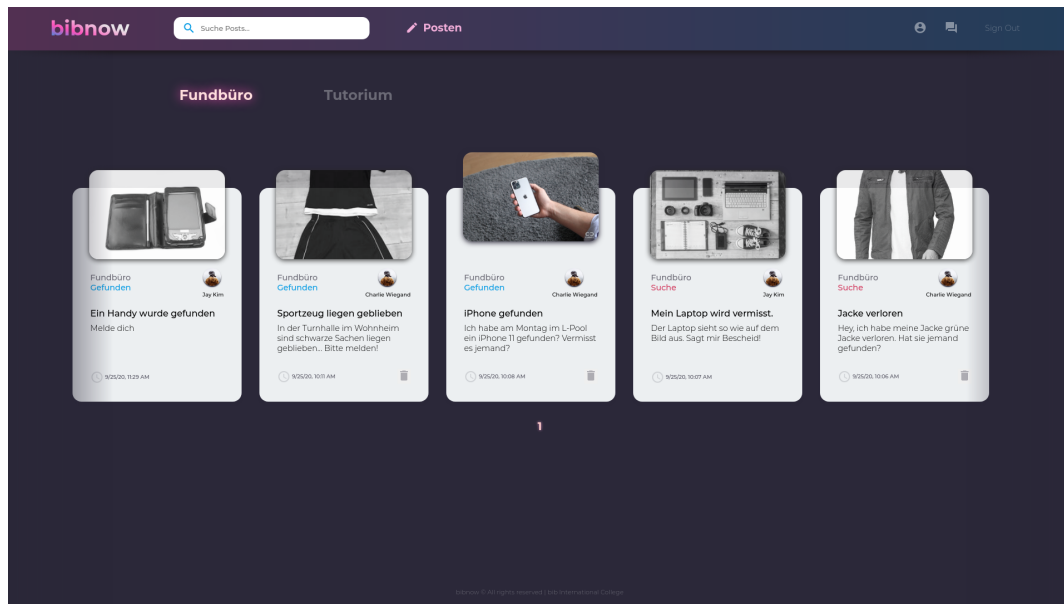


Abbildung 15: Screenshot DashboardQuelle: eigene Darstellung

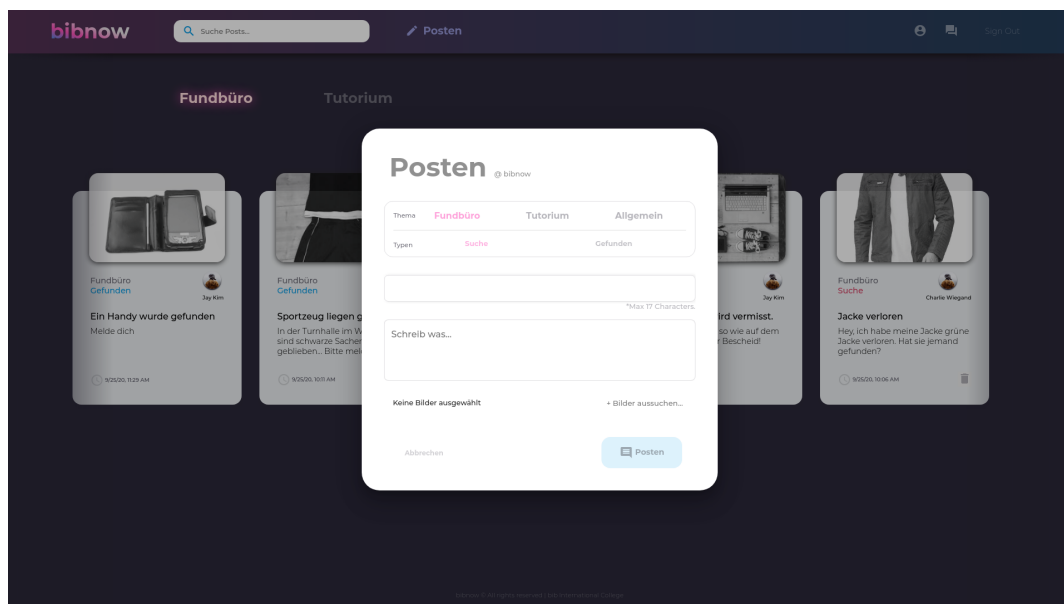


Abbildung 16: Screenshot PostdialogQuelle: eigene Darstellung

5.3.2 Mobile Ansicht



Abbildung 17: Screenshot DashboardQuelle: eigene Darstellung

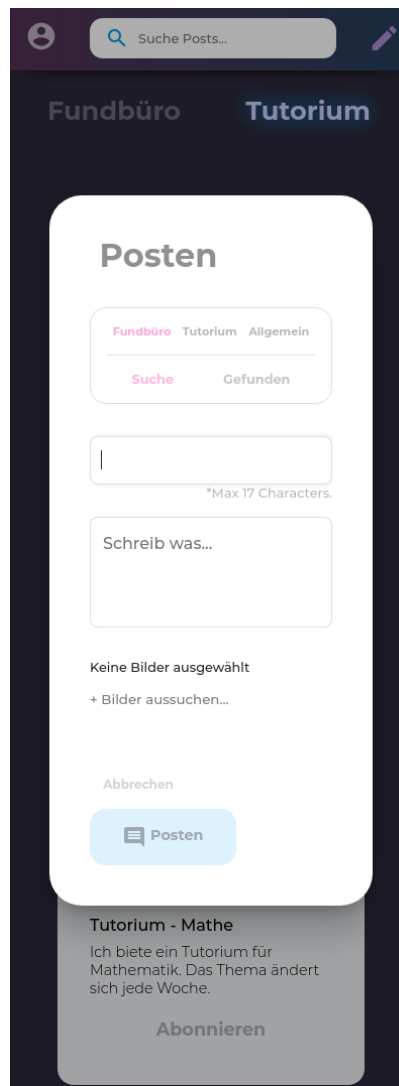


Abbildung 18: Screenshot PostdialogQuelle: eigene Darstellung

5.4 Datenbank

Die Datenbank implementierung fand in Firebase statt. Dort haben wir zwei Sammlungen erstellt: eine für Posts und eine für die Userdaten. Nachdem sich die Bildspeicherung in einer solchen RealTime Cloud Sammlung als problematisch herausstellte, haben wir sie in einen separaten StorageBucket in Firebase geladen und anschließend den Link zu dem jeweiligen Bild als string unter dem jeweiligen Post gespeichert.

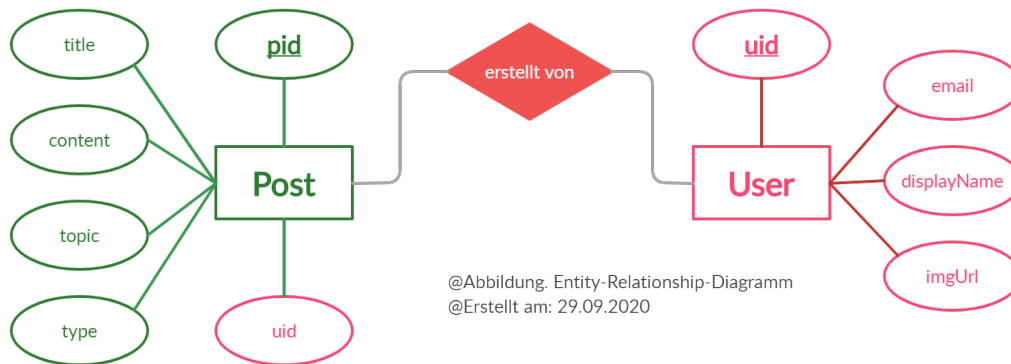


Abbildung 19: Entity Realationship ModellQuelle: eigene Darstellung

6 Fazit

Insgesamt hat uns die LEA und das Arbeiten im Team sehr gut gefallen, allerdings empfanden wir das Prozedere als unorganisiert und nicht einheitlich von Seiten der Dozenten.

Die Verwendung einer neuen Programmiersprache und die Auseinandersetzung mit Webentwicklung war auf jeden Fall lehrreich und hat uns auf fachlicher Ebene weitergebracht. Dennoch halten wir es für unwahrscheinlich, dass wir uns in nächster Zeit noch einmal freiwillig damit beschäftigen.

Für die nächste LEA haben wir uns vorgenommen unser Ziele geringer zu fassen und lieber wenige Kernfeatures voll auszubauen.

6.1 Mögliche Erweiterung

Insgesamt lässt sich das durch weitere Module oder Seiten ersetzen, wie andere Foren-Oberthemen, eine Stundenplanintegration über die ical- API oder ein Messengersystem für die User.

7 Quellen

Wir haben folgende Seiten zur Hilfe genutzt:

- Angular Dokumentation
<https://angular.io/docs>
- Firebase Dokumentation
<https://firebase.google.com/docs>
- LazyLoading.gif
<https://www.npmjs.com/package/angular-responsive-carousel>
- Schema-generator
<http://colormind.io/bootstrap/>
- Schriftart
<https://fonts.google.com/specimen/Montserrat>
- Stackoverflow
<https://stackoverflow.com/>
- Youtube
<https://www.youtube.com>