

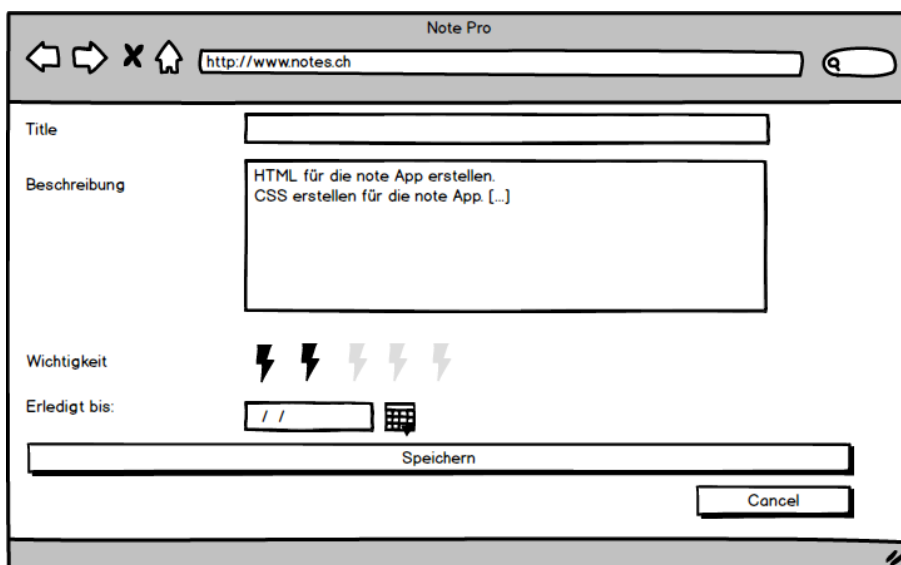
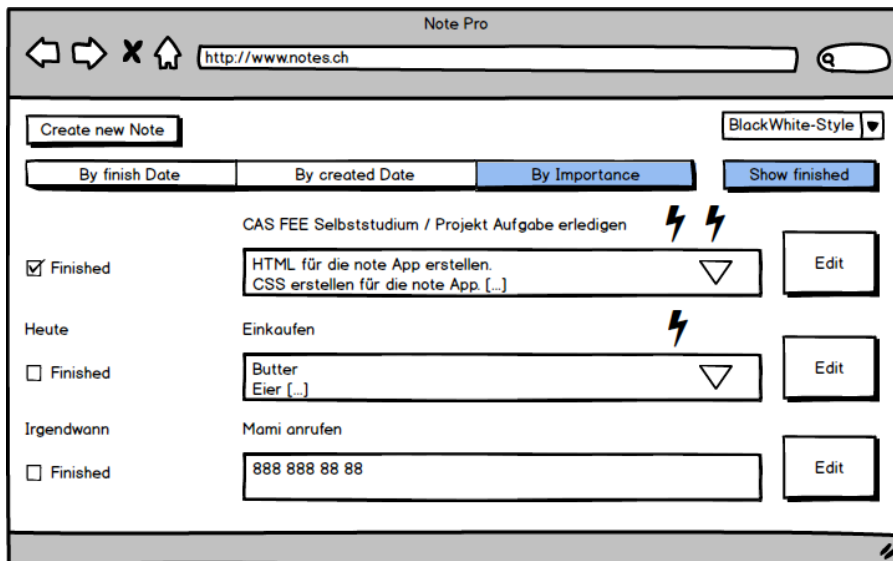
Testat WED 2

Ihre Testat-Aufgabe besteht darin, eine Notizen-Webapplikation zu programmieren. Es soll möglich sein, Notizen zu verwalten.

Die Grundlagen bilden die Wireframes, welche den Funktionalitätsumfang zeigen. Zusätzlich definiert das Video „Testat-WED2.mp4“ die dynamische Ansicht der Webseite (z.B. vom Fluiden Design / Validation / Style Switcher).

Ihre Aufgabe ist den kompletten Funktionsumfang der Wireframes zu implementieren und die im Video visualisierten Feinheiten zu berücksichtigen.

Am Aussehen dürfen Anpassungen vorgenommen werden.



Durchführung

Das Testat ist eine Einzel-Arbeit.

Peer-Review

Teil der Bestehens-Anforderungen (siehe unten) ist die Durchführung von drei Peer-Reviews. Dazu müssen Sie pünktlich bis zum Abgabedatum Ihre Projektinformationen im Moodle in dem entsprechenden Bereich ("Testat-Abgabe und Peer-Review") hinterlegt haben. Ihnen werden dann drei andere Projekte zugeteilt, welche Sie anhand einer gegebenen Kriterien-Liste (siehe unten) evaluieren müssen. Während Sie diese drei Reviews erstellen, werden für Sie drei Reviews von Ihren Mitstudierenden angefertigt. Die Reviews sind nicht verdeckt. Das heisst Sie sehen für welche andere Person sie den Review machen und in der von uns aktivierten nächsten Phase sehen Sie wer für Sie die Reviews gemacht hat.

Die Abänderung der Projektunterlagen während der Review-Phase ist nicht möglich.

Reviews sollten ehrlich, nach bestem Wissen und Gewissen gemacht werden. Wir werden dies Stichprobenweise überprüfen. Für die Durchführung der Peer-Review steht eine Übungslektion zu Verfügung. Nutzen Sie diese Möglichkeit, um die andere Lösung nachzuvollziehen und inhaltlich korrekt zu evaluieren.

Weitere Erklärung zum Peer-Review-Ablauf folgen noch.

Bestehens-Anforderungen

Für das Bestehen des Testats müssen folgende Bedingungen erfüllt sein

- (1) Sie haben Ihre Projektinformationen fristgerecht im Bereich "Testat-Abgabe und Peer-Review" hinterlegt
- (2) Sie haben Ihre drei Reviews fristgerecht abgeschossen
- (3) Die drei Reviews sind inhaltlich korrekt (wird von uns in Stichproben überprüft)
- (4) Ihre Abgabe genügt den Bestehens-Kriterien (siehe unten)

Nachkorrektur

Sie haben die Möglichkeit für eine Nachkorrektur. Für diese steht allerdings nur kurze Zeit zur Verfügung. Die Nachkorrektur wird von uns abschliessend bewertet (pass/fail). Eine zweite Nachkorrektur ist nicht möglich.

Checkliste

Folgende Checkliste bildet die Grundlage des Peer-Reviews.

Kriterium	Evaluation
Kompletter Funktionsumfang ist vorhanden: Nutzer können Notizen zu erfassen, anzeigen lassen, editieren, filtern und sortieren. Sowie die Art der Anzeige wechseln (Style)	Ja/Nein
Alle Felder der Notiz werden dargestellt.	Ja/Nein
Die Notizen können auf- UND absteigend nach Datum und Wichtigkeit sortiert werden.	Ja/Nein
Abgeschlossene Notizen können von der Anzeige ausgeschlossen werden (Filter).	Ja/Nein
Es gibt zwei unterschiedliche Styles (z.B. Day/Night) und diese können gewechselt werden.	Ja/Nein
Sortierung, Filter und gewählter Style bleibt nach dem Seiten-Refresh erhalten.	Ja/Nein
Responsives Design/Layout funktioniert und nutzt den vorhandenen Platz bei den folgenden Bildschirmbreiten angemessen: 480px und 1080px	Ja/Nein
Wenn keine Notiz vorhanden ist, wird ein Platzhalter (Mitteilung) angezeigt (Empty List Pattern).	Ja/Nein
Server hat eine MVC-Architektur: - Controller (separates File) hat keinen direkten Zugriff auf die Datenbank - Views und Services sind in separaten Files definiert	Ja/Nein
Daten werden persistent in Datenbank gespeichert (nedb)	Ja/Nein
Clean-Code Der JS/CSS Code beinhaltet keine grösseren Probleme: - Keine globalen Variablen - Kein Copy and Paste - Weitere «Clean Code» Eigenschaften in WED1 gelernt	Ja / Nein (Kommentare notwendig)
Tools: Typescript wird sinnvoll eingesetzt; ES-Lint mit sinnvollem Ruleset genutzt. Prettier sinnvoll genutzt.	Ja/Nein
Es wird kein JavaScript auf dem Client ausgeführt: Kein <script> im HTML vorhanden.	Ja/Nein