



Dokumentation Onlineshop: RECORDIUM

Requirements

Die Anforderungen für dieses Projekt besagen, dass der Onlineshop aus mindestens drei Seiten bestehen muss. Gefordert ist eine Produktseite, ein Warenkorb und eine Checkoutseite mit Kontaktformular. Produkte sollen beliebig oft in den Warenkorb hinzugefügt werden können. Im Warenkorb müssen die gewählten Produkte angezeigt werden und es muss die Möglichkeit bestehen ein Produkt wieder aus dem Warenkorb zu entfernen. Das Kontaktformular im Checkout muss Informationen, wie Name, Adresse und Email erfragen. Bei dem Inhalt des Onlineshops handelt es sich um einen Online-Plattenladen mit dem Namen „Recordium“.

Eine Datenbankanbindung ist nicht gefordert.

Paper Prototype/Konzept (1/2)

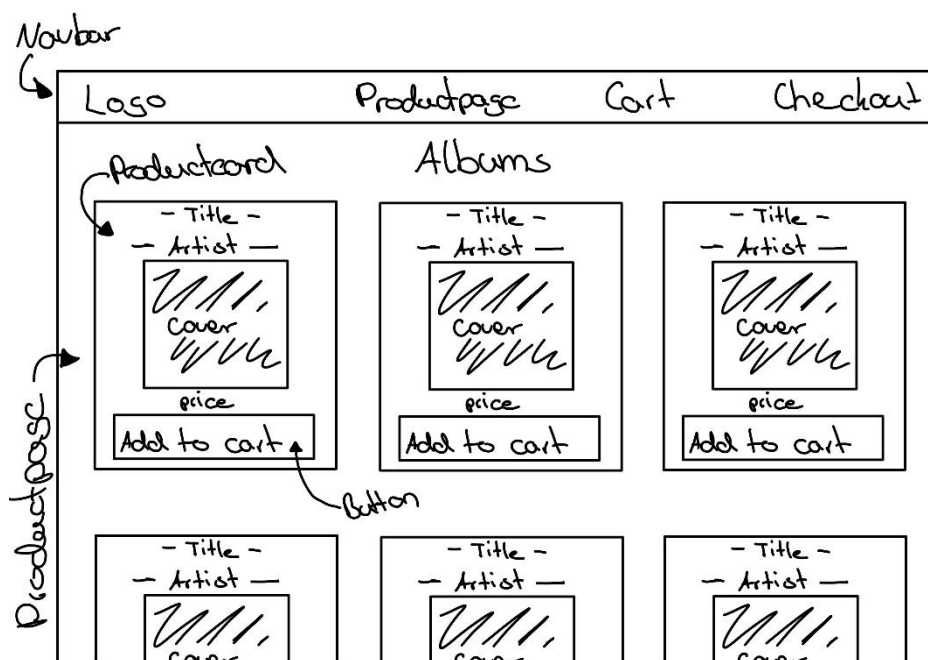
Im Folgenden werden die vorab erstellen Mockups vorgestellt. Diese dienen als Leitlinie für den Verlauf des Projekts und für die Entscheidungen hinsichtlich z.B. React Komponenten, Klassendiagrammen.

Navbar

Navigationsleiste zum Wechseln zwischen den einzelnen Komponenten der Seite.

Productpage

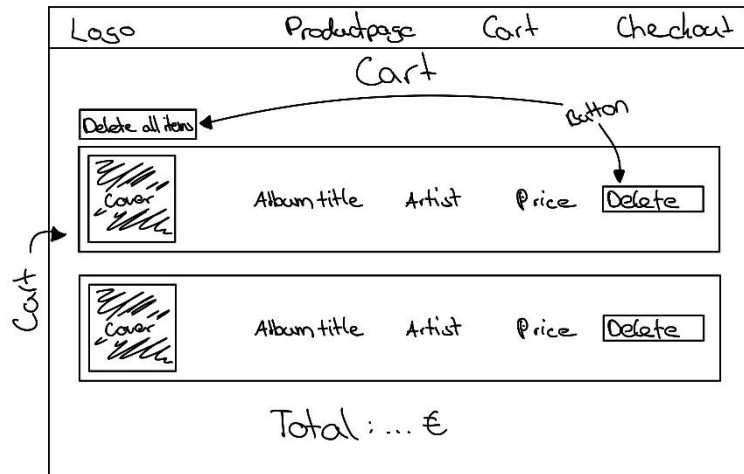
Die Produktseite präsentiert die angebotenen Produkte des Onlineshops. Zur Repräsentation werden Cards verwendet. Diese Cards sind in 5 Unterkomponenten unterteilt: Albumtitel, Künstler, Cover, Preis und einen Button zum Hinzufügen in den Warenkorb.





Cart

Im Warenkorb sind die ausgewählten Alben aufgelistet. Die einzelnen Produkte werden ebenfalls durch Cards repräsentiert. Jedoch handelt es sich hier um horizontale Cards, um die Produkte besser auflisten zu können. Sie bestehen aus den 5 Komponenten Cover, Albumtitel, Künstler, Preis und einem Button, der es dem User ermöglicht das Produkt wieder aus dem Warenkorb zu entfernen. Über den Cards befindet sich ein Button mit dem der User seinen gesamten Warenkorb leeren kann. Unter der Auflistung der Produkte befindet sich eine Anzeige mit dem Preis für den gesamten Einkauf. Befindet sich jedoch kein Produkt im Warenkorb wird eine Seite angezeigt auf der „Your cart is empty“ angezeigt wird.

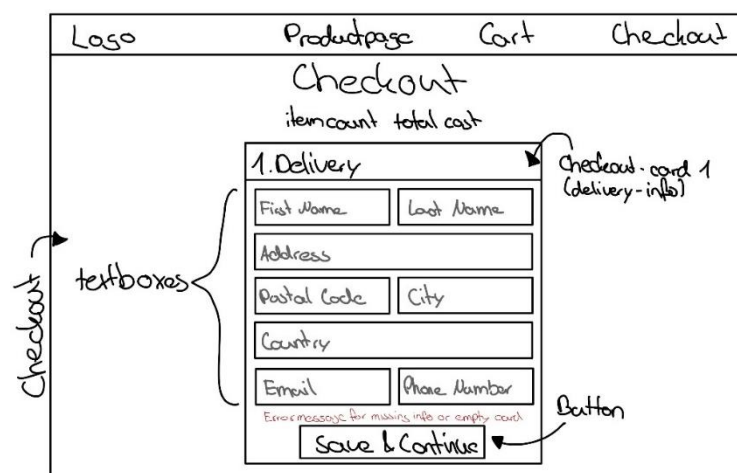


Checkout

Die Checkoutseite ist in 2 Teile unterteilt. Sie besteht aus einer Seite für den Checkout und aus einer Seite mit Dankesnachricht, die nach dem Abschluss der Bestellung angezeigt wird. Für die Dankesnachricht wurde kein Mockup erstellt, da es sich hier um keinen komplexen Bestandteil der Seite handelt. Auf der Checkoutseite befindet sich oben der Titel. Darunter ist die Anzahl an Produkten im Warenkorb und der Gesamtpreis dokumentiert. Die einzelnen Schritte des Checkouts werden mit Hilfe von Cards durchgeführt. Es gibt jeweils eine Card für das Kontaktformular, die Auswahl des Zahlungsmittels und eine Zusammenfassung der Bestellung.

Delivery Card

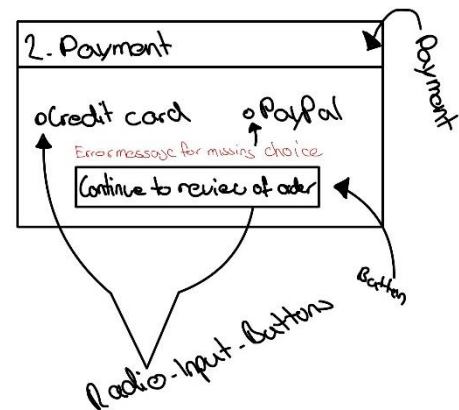
Die Delivery Card besteht aus diversen Textboxen in denen die nötigen Informationen eingetragen werden. Mit ausgegrautem Text ist der gewünschte Inhalt der Textboxen markiert. Darunter kann, falls Informationen fehlen eine Fehlermeldung angezeigt werden. Abschließend findet sich dort ein Button mit dem der User zur nächsten Card gelangen kann.





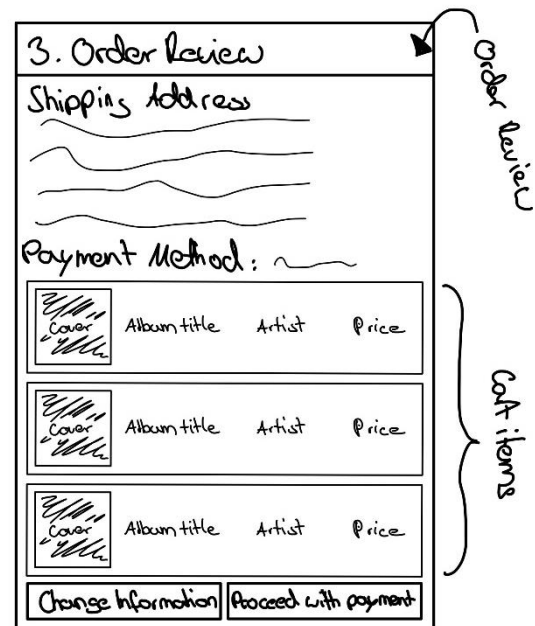
Payment Card

In der Payment Card kann mit Hilfe von Radiobuttons das gewünschte Zahlungsmittel (Paypal oder Kreditkarte) ausgewählt werden. Es besteht ebenfalls die Möglichkeit eine Fehlermeldung anzuzeigen falls keine Zahlungsmethode ausgewählt wurde. Abschließend befindet sich unten wieder ein Button zum Fortfahren.



Order Review

Die letzte Card ist die Order Review. Hier werden abschließend alle Informationen aufgelistet. Dies beinhaltet die Kontaktdaten, die ausgewählte Zahlungsmethode und eine Liste der ausgewählten Produkte. Die Liste besteht aus horizontalen Cards, die denen des Warenkorbs ähneln (kleiner und ohne entfernen-Button). Darunter befinden sich zwei Buttons mit denen man entweder zu Schritt 1 des Checkouts zurückkehren kann, um die eingegebenen Informationen abzuändern oder den Kauf abschließen kann. Mit Click auf den „Proceed with Payment“-Button wird die Dankesnachricht angezeigt. Diese würde normalerweise erscheinen nachdem der Kunde zur Zahlungsseite weitergeleitet worden wäre.





Klassendiagramm/Konzept (2/2)

Der Aufbau der React.js App ist auf Seite 7 im Klassendiagramm zu sehen. Die normalen Linien zeigt die Navigationsverbindung zwischen den einzelnen Komponenten. Die Pfeile zeigen den Informationsaustausch an.

Router.jsx

Die App.js rendert den Router, das Herzstück der Anwendung. Im Router wird als erstes der React Router definiert. Dieser ermöglicht es durch den Switch zwischen den verschiedenen Seiten:

- `Produktpage.jsx`
- `Cart.jsx`
- `Checkout.jsx`

zu navigieren. Im Router wird ebenfalls die Navigationsleiste (`Nav.jsx`) gerendert. Diese wird über jeder Seite mitgerendert.

Hier wird auch eine zentrale State Variable gespeichert. Diese beinhaltet den Inhalt des Warenkorbs. Um zu gewährleisten, dass die Warenkorbinformationen bei einem Neuladen der Seite gespeichert werden, werden die Information bei jeder Änderung über den `localStorage` in den Cache geladen und von dort aus auch wieder abgerufen. Innerhalb des Routers werden 4 Methoden definiert:

- `componentDidMount`
- 3 callback functions
 - `updateCart` → Hinzufügen von Produkten in Warenkorb
 - `deleteCart` → Entfernen eines Produktes aus dem Warenkorb
 - `deleteCartAll` → Entfernen aller Produkte aus dem Warenkorb

Die state Variable und die Methoden werden über props an die einzelnen Routs weitergegeben.

Productpage.jsx

In der Komponente `Productpage.jsx` wird als erster Schritt ein state definiert in dem ein Array erstellt wird, in dem die Produkte gespeichert werden. Dies ermöglicht das einfache Hinzufügen weiterer Produkte. Darunter werden die Produkte mit Hilfe der `map`-Funktion in die Cards gerendert. Mit einem Click auf den „Add to Cart“ Button wird über die props die `updateCart` Methode aufgerufen und das jeweilige Produkt wird dem zentralen State hinzugefügt.

Cart

Mit Hilfe von `useEffect` und einem state für den Cart wird der Warenkorbinhalt bei einem `Pagerefresh` im cache gespeichert. Dann wird mit einer `for`-Schleife der Gesamtpreis berechnet und aktuell gehalten. Mit einem `if`-Statement wird festgestellt ob Produkte im Warenkorb sind. Ausgehend davon wird entweder die „Cart is empty“-Seite oder der Warenkorb mit dessen Inhalt angezeigt. Auf der Warenkorbseite werden die einzelnen Cards ebenfalls mit der `map`-Funktion gerendert.



Checkout

Im Checkout werden vorerst erneut die Anzahl an Produkten und der Gesamtpreis berechnet. Mit Hilfe der verschiedenen Funktionen, die auf den jeweiligen Buttons hinterlegt sind, wird zwischen den einzelnen Seiten und Cards navigiert:

- **deliveryFunction** (Button auf Delivery-Card)
 - Zum Fortfahren nach Ausfüllung des Kontaktformulars
 - Überprüft die Zahl an Produkten im Warenkorb:
 - Keine Produkte → Errormessage „Cart is empty“
 - Überprüft ob alle Felder des Formulars ausgefüllt sind + Zahl an Produkten im Warenkorb
 - Keine Produkte → Errormessage „Cart is empty“
 - Fehlendes Feld → Errormessage „Please fill in missing information!“
 - Kein Error: Inhalt der Textboxen wird in Order-Review übertragen, Delivery-Card wird versteckt und Payment-Card wird gezeigt
- **paymentFunction** (Button auf Payment-Card)
 - Zum Fortfahren nach Auswahl der Zahlungsmethode
 - Überprüft ob Zahlungsmethode ausgewählt wurde
 - Keine Zahlungsmethode ausgewählt → Errormessage „Please select payment method!“
 - Überprüft welche Zahlungsmethode ausgewählt wurde
 - Zahlungsmethode wird in Order-Review übertragen
 - Kein Error: Payment-Card wird versteckt und Order-Review-Card wird gezeigt
- **reviewFunction** (Button auf Order-Review-Card)
 - Zum Abschließen des Checkoutprozesses
 - Order-Review-Card, Checkout-Überschrift, Produktzahl und Gesamtpreis wird versteckt und Dankesnachricht-Seite wird gezeigt
 - Warenkorb wird geleert (mit Hilfe von callback function über props)
- **changeInfo** (Button auf Order-Review-Card)
 - Zum Abändern der eingegebenen Informationen und der Zahlungsart
 - Order-Review-Card wird versteckt und Delivery-Seite wird wieder gezeigt
 - Errormessages werden versteckt damit sie nicht wieder auftauchen beim Informationen ändern

Der Aufbau der einzelnen Cards wurde schon im Teil Konzept (1/2) beschrieben. Wichtig ist nur noch zu erwähnen, dass die horizontalen Produktcards in der Order-Review ebenfalls mit Hilfe der map-Funktion gerendert worden sind.



Resümee

Während der Entwicklung der Website bin ich häufig auf Probleme gestoßen. Das Hauptproblem war die Informationsverknüpfung der einzelnen Komponenten, da dort diverse Fehler aufgetreten sind. Am längsten hat die Lösung des Falles, dass der Warenkorb nur nach einem Pagerefresh aktualisiert war, gedauert. Mit der Funktion `forceUpdate` hat sich dieses Problem gelöst. Allgemein ist mir jedoch aufgefallen, dass der gesamte Prozess am Anfang der Entwicklung sehr langwierig war. Je länger ich aber an der Anwendung gearbeitet habe, desto flüssiger wurde mein Workflow, was auch bewirkt hat, dass mir die Entwicklung immer mehr Spaß gemacht hat. Dies hat unter anderem zu den Animationen und den zusätzlichen Designkomponenten geführt.

