

Projekt: Hotel Buchungsoberfläche

Inhalt

- Kontext und Einordnung
- Technische Anforderungen
- User Stories

Kontext und Einordnung

Liebes Team,

Ich bin begeistert, euch mitteilen zu können, dass unser kleines Softwareunternehmen den Auftrag erhalten hat, eine Buchungsapp für das Boutique-Hotel Technikum zu entwickeln. Diese App wird sowohl auf Smartphones als auch auf Desktops verfügbar sein und bis zu 100 Besucher pro Tag verarbeiten können. Wir haben uns zum Ziel gesetzt, eine moderne und nachhaltige App zu erstellen, die auch in den kommenden Jahren um weitere Features erweitert werden kann.

Die Buchungsapp wird unseren Kunden des Boutique-Hotels Technikum eine benutzerfreundliche Möglichkeit bieten, Hotelzimmer für einen bestimmten Zeitraum zu buchen. Die Gäste können sich auch registrieren, um ihre Buchungen und Personendaten bequem einzusehen. Das Projekt wird von uns in Teams zu dritt umgesetzt, wobei jedes Team sowohl das Backend als auch das Frontend erstellen wird.

Wir wissen, dass dieses Projekt eine Herausforderung darstellt. Aber wir sind zuversichtlich, dass wir mit unserem Engagement und unserer Kreativität diese Herausforderungen meistern werden. Wir sind uns auch bewusst, dass wir nicht nur eine großartige App entwickeln, sondern auch wertvolle Erfahrungen und Fähigkeiten als Entwickler sammeln werden.

Ich freue mich auf die Zusammenarbeit mit euch und darauf, unsere Fähigkeiten und Ideen zu teilen, um das bestmögliche Ergebnis zu erzielen. Wir werden das Projekt durch Feedbackrunden und Code Reviews begleiten und euch bei Bedarf unterstützen. Lasst uns gemeinsam an diesem Projekt arbeiten und eine großartige Buchungsapp für das Boutique-Hotel Technikum erstellen!

Beste Grüße,
euer Projektleiter

Technische Anforderungen

- Die Buchungssapp wird unter Verwendung von Vue.js (Version 3) entwickelt und soll mithilfe des Frameworks Ionic schnell umgesetzt werden.
- Für API-Calls wird die Bibliothek Axios verwendet und das State-Handling wird mithilfe von Pinia realisiert.
- Die Anwendung wird hauptsächlich für mobile Endgeräte entwickelt, soll aber auch auf Desktops bedienbar sein.
- Für das Backend steht es den Teams frei, entweder Spring Boot (Version ≥ 2.4) oder PHP Laravel (Version ≥ 8) zu verwenden.
- Als Datenbank wird eine MySQL-Datenbank eingesetzt.

User Stories

U1: Hotelwebsite

Als **Gast** möchte ich **in Form einer Website** das Hotel präsentiert bekommen um **mich darüber zu informieren**.

Details

Die Single Page Application enthält neben den Funktionen zur Buchung und Verwaltung auch statische Seiten, die den Besucher einerseits einladen das Hotel genauer anzusehen und andererseits solche Informationen bereitstellen, dass der Besucher weiß, was ihn vor, während und nach erfolgreicher Buchung erwartet.

Recommended Approach

- Definieren Sie, welche Informationen für die Präsentation des Hotels bereitgestellt werden müssen
- Aufbereiten der Infos, Recherche und bereitstellen von Bildern
- Erstellen sie für jede Page eine Komponente

Definition of Done

- [] die HTML Templates für die statischen Pages (zumindest Landingpage, Impressum, About) wurden erstellt
- [] Texte und Inhalte wurden aufbereitet und eingepflegt
- [] die einzelnen Seiten können aufgerufen und im Browser korrekt angezeigt werden
- [] die Inhalte sind sowohl auf einem Smartphone, als auch auf einem Desktop Gerät gut angeordnet und lesbar

U2: Hotelzimmer Auswahl

Als Gast möchte ich eine Übersicht über Hotelzimmer und die zugehörigen Details sehen können, um mir nach meinen Vorstellungen ein Zimmer auszusuchen.

Details

Es besteht eine API Schnittstelle über die Informationen zu den Zimmern abgerufen werden können. Diese ist in der API Dokumentation zu finden.

Im Ordner `images/rooms` des Repositories finden sich Bilder. Die Dateinamen entsprechen der jeweiligen Zimmer id. Es kann sein, dass die Bilder nicht alle gleich groß sind. Finden Sie auch dafür eine Lösung.

Zu den Extras sollen Aussagekräftige Icons gestellt werden, damit der Benutzer sich gut zurechtfindet. Es sollen beim ersten Laden maximal 5 Hotelzimmer angezeigt werden. Das heißt, es bedarf einer Lösung, wie man eine ungerade Zahl an Zimmern gut darstellt und wie man auf Seite 2 (und später evtl. 3 und 4 - Ein Zubau von Zimmern ist für das kommende Jahr geplant) wechseln kann.

Recommended Approach

- Machen Sie sich mit der Schnittstelle vertraut
- Analyse der Rückgabe Daten
- Paperprototype für die Anzeige und das Paging
- Welche Interaktionen sind notwendig?
- Machen Sie sich mit den [bootstrap icons](#) vertraut
- Wenn sie ein anderes Designsystem einsetzen, machen Sie sich mit den Icons dort vertraut
- Wählen Sie Icons für die Extras aus
- Verwenden Sie für die Pageination Buttons s.g. Buttongroups

Definition of Done

- ☐ Hotelzimmer können mit Bild, Titel, Beschreibung und Extras angezeigt werden
- ☐ Auf Seite eins werden 5 Hotelzimmer angezeigt
- ☐ Es ist möglich weitere Seiten anzuzeigen: Pagination
- ☐ Die Extras werden mit Icons versehen, die das Extra repräsentieren
- ☐ Das Feature wurde in einem eigenen Branch eingecheckt und ins Git-Repository gepusht
- ☐ Das Feature wurde nach einem Code Review in den Master gemerged

U3: Verfügbarkeit prüfen

Als Gast möchte ich überprüfen ob ein bestimmtes Zimmer zum meinem gewünschten Reisezeitraum verfügbar ist, um festzustellen, ob ich das Zimmer werde buchen können.

Details

Es besteht eine API Schnittstelle über die Informationen zu den Zimmern abgerufen werden können. Diese ist in der API Dokumentation zu finden.

Der Benutzer soll in der Oberfläche die Möglichkeit haben, seinen Buchungszeitraum zu definieren. Sie haben hier freie Hand, wie sie das dem Benutzer am besten darstellen. Denken Sie innovativ und out of the box dabei.

Recommended Approach

- Machen Sie sich mit der Schnittstelle vertraut
- Analyse der Rückgabe Daten
- Bilden Sie zwei bis drei Varianten, wie man dem Benutzer den Zeitraum definieren lassen könnte, wie man die Zimmerverfügbarkeit darstellt, etc.
- Denken Sie auch den Userflow, bzw. das Usecase Szenario durch, Welcher Fehlerfälle gibt es
- Überlegen Sie, wie sie neue Komponenten entwerfen müssen, oder alte erweitern / ändern, damit Ihre Idee des GUI auch aus Implementierungssicht gut umsetzbar ist
- Bauen Sie zuerst den Store und die API Anbindung und dann das GUI

Definition of Done

- ☐ Der Benutzer kann über einen Datumsdialog den Zeitraum der Buchung festlegen
- ☐ der Benutzer bekommt klares Feedback über die Verfügbarkeit in diesem Zeitraum
- ☐ Mögliche Fehlerfälle werden abgefangen und der Benutzer entsprechend darauf hingewiesen
- ☐ Das Feature wurde in einem eigenen Branch eingecheckt und ins Git-Repository gepusht
- ☐ Das Feature wurde nach einem Code Review in den Master gemerged

U4: Hotelzimmer reservieren

Als Gast möchte ich zu einem ausgewählten Zeitraum ein ausgewähltes Zimmer buchen um eine Unterkunft in meinem Urlaub zu haben.

Details

Es besteht eine API Schnittstelle um eine Buchung zu einem Zimmer durchführen zu können. Diese ist in der API Dokumentation zu finden.

Nachdem der Benutzer ein verfügbares Zimmer gefunden hat, soll er es Buchen können. Er muss dazu folgende Daten angeben:

- Vorname
- Nachname
- gültige E-Mail-Adresse
- E-Mail-Adresse bestätigen
- Frühstück Ja / Nein

Er soll vor dem endgültigen Absenden, nochmal die Möglichkeit haben, seine Buchungsdaten (inkl. Buchungszeitraum und Zimmerwahl) zu überprüfen und ggf. zu bearbeiten.

Nach erfolgter Buchung muss dem Benutzer eine Bestätigung angezeigt werden.

Recommended Approach

- Machen Sie sich mit der Schnittstelle vertraut
- Analyse der zu übertragenden Daten
- Paperprototype inkl. Userflow und Errorhandling
- Bauen Sie zuerst den Store und die API Anbindung und dann das GUI

Definition of Done

- ☐ Der Benutzer kann zu einem gewünschten Zeitraum ein gewünschtes Zimmer unter Angabe seiner vollständigen und korrekten Daten Buchen, wenn das Zimmer verfügbar ist
- ☐ Mögliche Fehlerfälle werden abgefangen und der Benutzer entsprechend darauf hingewiesen
- ☐ Der Benutzer kann seine Daten vor Absenden nochmal überprüfen
- ☐ Der Benutzer erhält eine Bestätigung nach erfolgreicher Buchung
- ☐ Das Feature wurde in einem eigenen Branch eingecheckt und ins Git-Repository gepusht
- ☐ Das Feature wurde nach einem Code Review in den Master gemerged

U5: Reservierungsbestätigung verbessern

Als Gast möchte ich nach der Buchung erfahren, ob diese erfolgreich oder fehlerhaft war um zu wissen ob ich meinen Prozess beenden kann, oder weitere Schritte nötig sind.

Details

Die in U4 angesprochene Buchungsbestätigung soll erweitert werden um den Benutzer ausführlich über alle Details zu informieren und ihm die Anreise und das Einchecken zu erleichtern.

Die Details der Buchung sollen angezeigt werden:

- Der Zeitraum
- Das Hotelzimmer inkl. Bild, Titel, Extras, und Beschreibung
- Seine Daten, die er angegeben hat
- Anfahrtsbeschreibung zum Hotel, und eventuelle weitere Infos (Zugverbindungen)
- Kontaktmöglichkeiten

Recommended Approach

- Überlegen Sie welche Infos sonst noch notwendig sind (außer hier beschriebene)
- Paperprototype der Ausgabe
- Können Komponenten wieder verwendet werden?
- Kann man Google Maps oder Ähnliches einbinden für die Anfahrt? Gibt es da Packages, was benötigt man dazu?

Important Comment

Diese Userstory markiert (wenn 1-4 auch erledigt sind) den ersten Mileston und damit ersten Release. Wir erreichen damit V1.0.0

Definition of Done

- ☐ Die Bestätigung wird mit den in der Sektion Details beschriebenen Informationen erweitert
- ☐ Die Bestätigung soll möglichst auf A4 Ausdruckbar sein (drucken aus Browser reicht, kein PDF Umwandlung nötig)
- ☐ Es gibt ein klares Ergebnis zum Thema Anfahrtsplan: Umsetzung, je nach Recherche Ergebnis
- ☐ Das Feature wurde in einem eigenen Branch eingecheckt und ins Git-Repository gepusht
- ☐ Das Feature wurde nach einem Code Review in den Master gemerged
- ☐ V1.0.0 wurde released (tagging)