[Ссылка на макет](https://www.figma.com/file/d8LhhLjMkaTfPvAcYQULNv/Library---students-file?node-id=3604%3A32182&t=wI5V45Q6Tv4IEfhd-0)

[API](https://docs.google.com/document/d/17wi2ecOONy-gZRiNUUC37bS3TIFqK8Dos8M-H_pE9OQ)

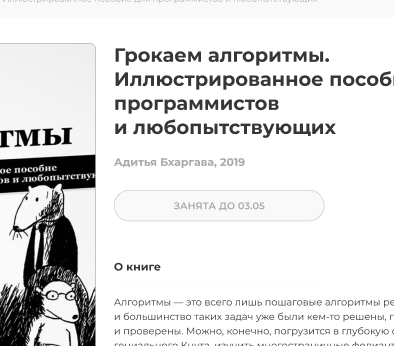
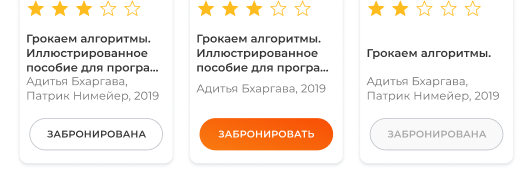
**Бронирование книг, оценка и отзывы**

В этом спринте вам необходимо реализовать создание, редактирование, удаление бронирования книг, а также оценку и создание отзыва на книги.

Нужно реализовать все запросы описанные в API, и отобразить полученные данные в соответствующих частях приложения.

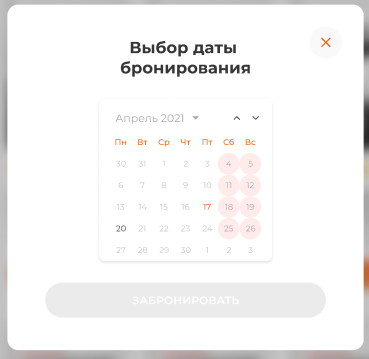
**Точки входа для модалок бронирования:**

На данном этапе точки входа для модалок бронирования присутствуют на элементе карточки книги и внутри страницы с описанием книги. Есть 4 вида кнопок ([см. макет](https://www.figma.com/file/d8LhhLjMkaTfPvAcYQULNv/Library---students-file?node-id=22102%3A57115&t=VSwTyD9Z5KXUcifI-4)).



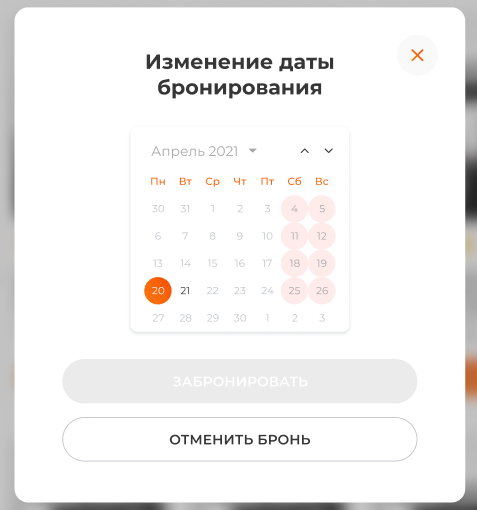
Отображение и текст кнопок строится на основе параметров "booking" и “delivery” из ранее реализованных запросов за книгами /api/books или конкретной книги /api/books/{id}. Текст кнопок - хардкод, кроме даты

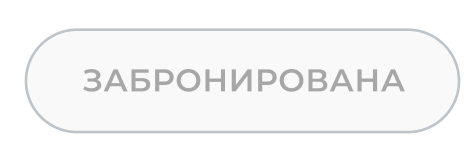
- если нет бронирований (booking:null), то кнопка активна и по нажатию осуществляется открытие модалки с выбором даты бронирования.

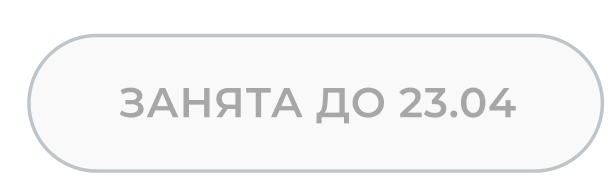




- если бронирование было осуществлено текущим юзером (booking => customerId ), то кнопка активна и по нажатию открывается модалка изменения/отмены бронирования



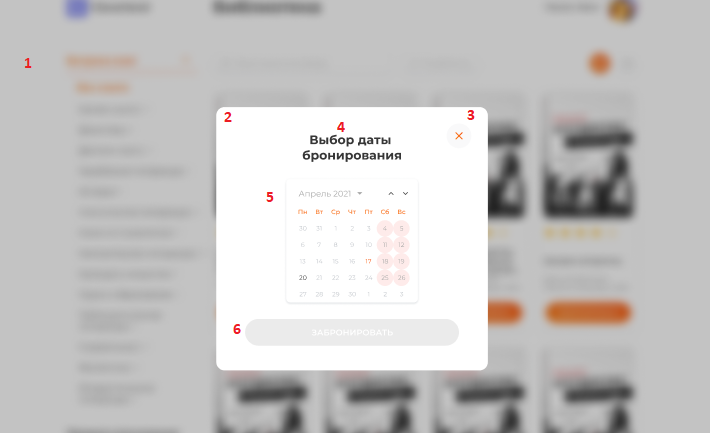


-если забронирована другим юзером, то кнопка неактивна (booking => customerId )

- если книга на руках, то кнопка неактивна (delivery => dateHandedTo)

**Элементы и логика работы модалок бронирования книг**

**Создание бронирования книги:**

****

Модалка должна содержать элементы:

1. подложка (на всю страницу) - по нажатию этой области закрывает модалку
2. модалка бронирования
3. кнопка закрытия модалки
4. Заголовок
5. Календарь выбора даты бронирования (его рассмотрим отдельно)
6. Кнопка отправки данных (по умолчанию неактивна)

Логика работы:

Основной элемент - календарь выбора даты.

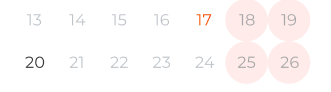
* Даты в календаре должны располагаться в соответствии своему дню недели с учетом месяца и года, а также захватывать недостающие дни из предыдущего или следующего месяца в случае их нехватки в первой и последней недели месяца.
* В верхней части календаря находятся элементы переключения месяца с отображением текущего года. По умолчанию текущий месяц и год.
* по нажатию отображается раскрывающийся список с выбором месяца (\*год не выбирается, \*\*вид списка - на ваше усмотрение). Выбираем месяц - даты в календаре обновляются.

По кнопке “↑” календарь переключается на предыдущий месяц . Даты в календаре обновляются

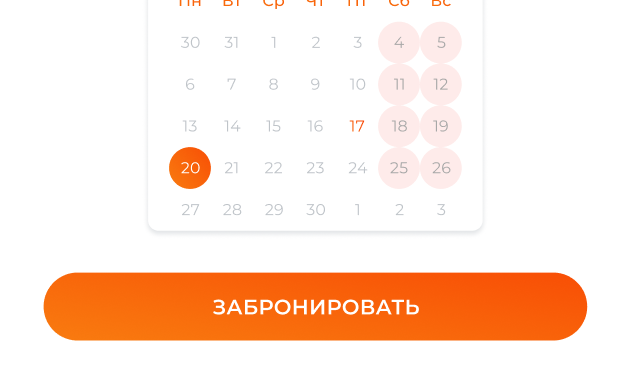
По кнопке “↓” календарь переключается на следующий месяц . Даты в календаре обновляются.

(важно: если при переключении по кнопкам ↓↑, выбран первый или последний месяц, то при следующем нажатии выбирается и отображается соответствующий год с месяцем, данные в календаре обновляются)

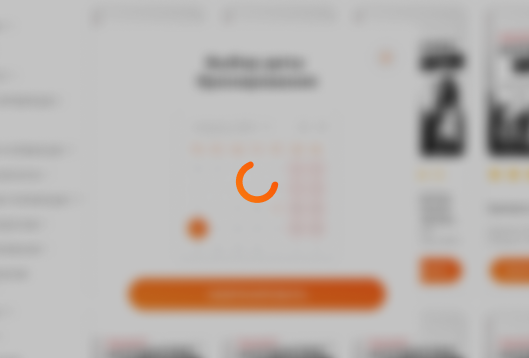
* Для выбора бронирования доступно только **2 дня** **(сегодня и завтра)**, если сегодняшняя дата выпадает на **пятницу**, то для выбора доступны пятница и понедельник, если сегодня выходной **(сб. или вс.)**, то доступен только понедельник. Все остальные дни **неактивны**.
* сегодняшняя дата подсвечивается оранжевым цветом, доступная для выбора дата - подсвечена черным
* выходные в красном круге



* по нажатию на доступный день, выбранная дата подсвечивается, кнопка “забронировать” становится активной

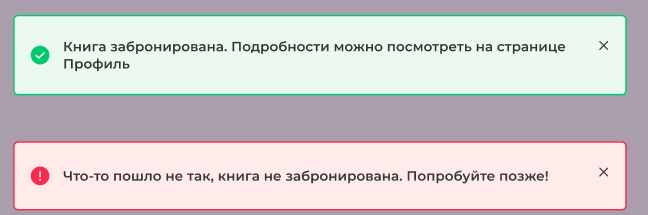


* По нажатию “забронировать” отправляется **POST** запрос /api/bookings с сформированным телом запроса ([см. api](https://docs.google.com/document/d/17wi2ecOONy-gZRiNUUC37bS3TIFqK8Dos8M-H_pE9OQ/edit)) и отображаем лоадер пока ожидаем ответ

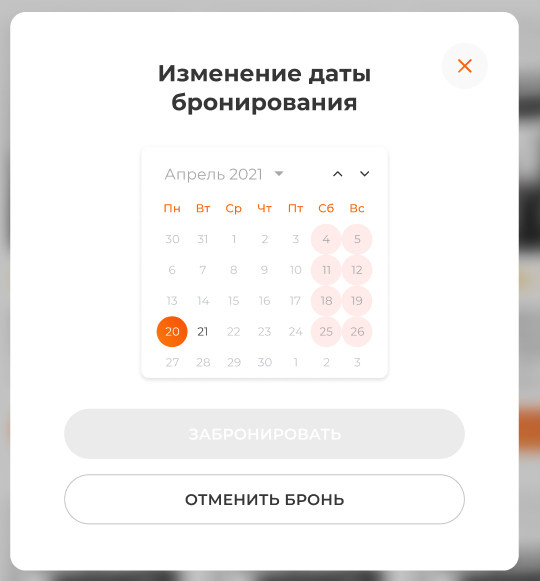


* по окончании загрузки - модалка закрывается, отображаем алерты успеха или ошибки (текст - хардкод)

(**важно**: для данного спринта добавим логику автоматического закрытия алертов через **4** секунды если не закрыли по “крестику”)

* 
* Данные на странице обновляются

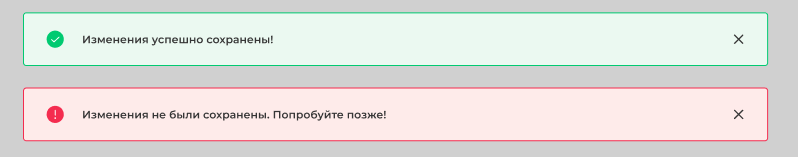
**Редактирование и отмена бронирования книги:**

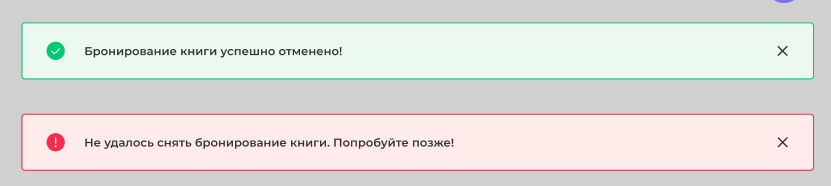
****

Элементы модалки аналогичны созданию, за исключением текста заголовка и кнопки отмены брони.

Логика работы:

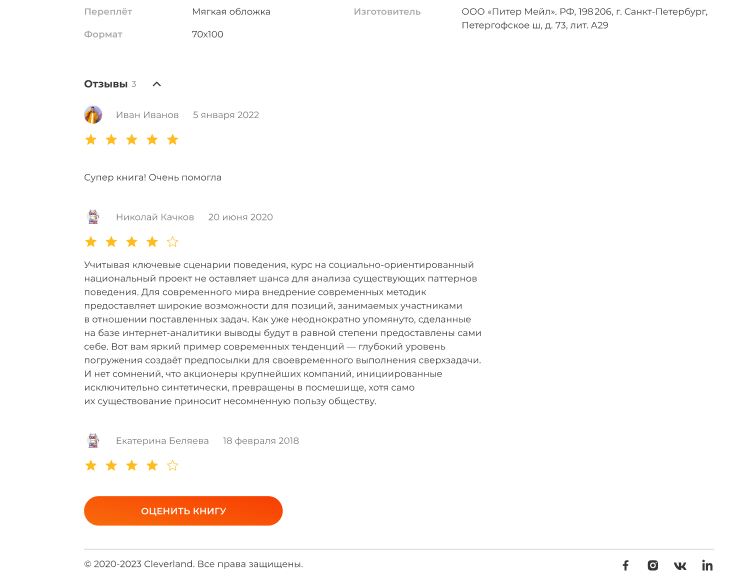
* Логика календаря аналогична
* После открытия модалки по умолчанию выбрана и подсвечена дата ( "booking" => dateOrder)
* Кнопка “Забронировать” заблокирована, если выбранная дата = "booking" => dateOrder,
* После выбора другой даты - кнопка “забронировать” становится активна
* По нажатию “забронировать” отправляется **PUT** запрос /api/bookings/{bookingId} тело запроса ([см. api](https://docs.google.com/document/d/17wi2ecOONy-gZRiNUUC37bS3TIFqK8Dos8M-H_pE9OQ/edit)), bookingId = параметру "booking": { id ….
* Отображаем лоадер пока ждем ответ.
* по окончании загрузки - модалка закрывается, отображаем алерты успеха или ошибки (текст - хардкод)



* Данные на странице обновляются
* Кнопка “отменить бронь” всегда активна
* по нажатию отправляется **DELETE**  запрос /api/bookings/{bookingId} тело запроса ([см. api](https://docs.google.com/document/d/17wi2ecOONy-gZRiNUUC37bS3TIFqK8Dos8M-H_pE9OQ/edit)), bookingId = параметру "booking": { id ….
* Отображаем лоадер пока ждем ответ.
* по окончании загрузки - модалка закрывается, отображаем алерты успеха или ошибки (текст - хардкод)
* 
* Данные на странице обновляются

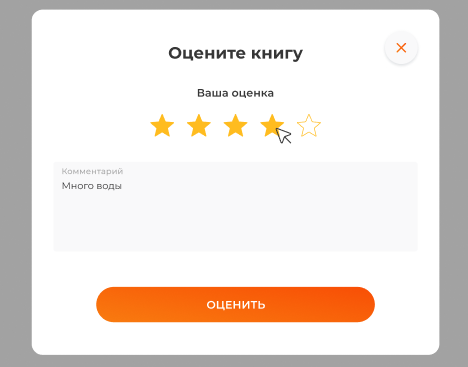
**Точки входа для оценки книг:**

На данном этапе переход к оценки книги происходит на детальной странице книги - кнопка “Оценить книгу”.

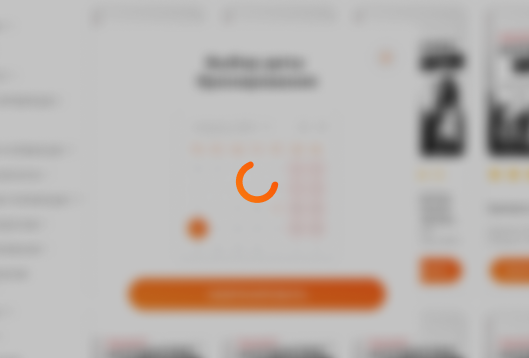


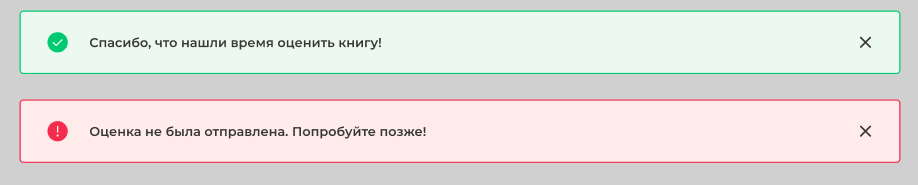
**Элементы и логика работы модалки оценки книг:**

По нажатию открывается модалка оценки



1. поле выбора рейтинга - по нажатию на “звезду” выбирается рейтинг и закрашивается соответствующее количество звезд. (\* по желанию можете также добавить закраску по наведению)
2. Поле ввода комментария
3. Кнопка отправки данных - по нажатию отправляется **POST** запрос /api/comments тело запроса ([см. api](https://docs.google.com/document/d/17wi2ecOONy-gZRiNUUC37bS3TIFqK8Dos8M-H_pE9OQ/edit)) и отображаем лоадер, пока ожидаем ответ сервера.



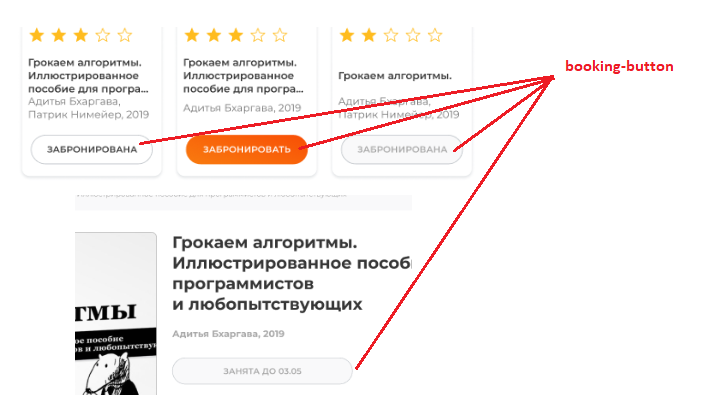
После ответа сервера - закрываем модалку и отображаем алерты успеха или ошибки.

Данные на странице обновляются.

(**важно:** для данного спринта также необходимо отсортировать комментарии на странице от новых к старым)

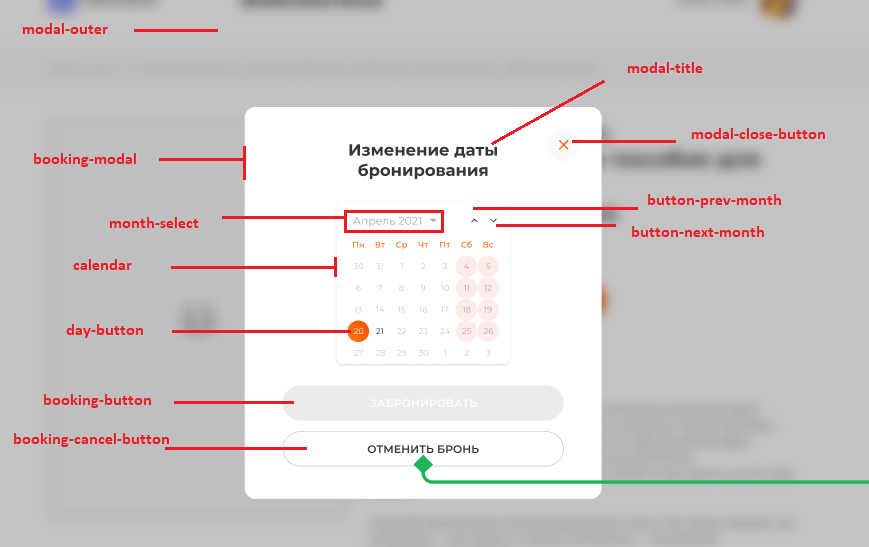
**Для прохождения тестов добавить data-test-id:**

На кнопки открытия модалок бронирования **booking-button**

****

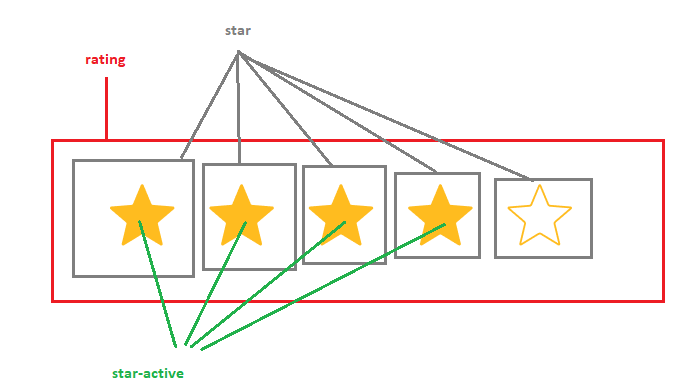
На модалки бронирования редактирования:

* подложка **modal-outer**
* обертка модалки **booking-modal**
* заголовок **modal-title**
* кнопка закрытия **modal-close-button**
* выбор месяца **month-select**
* кнопка месяца ↑ **button-prev-month**
* кнопка месяца ↓ **button-next-month**
* обертка календаря **calendar**
* на **каждый** день в календаре **day-button**
* кнопка забронировать **booking-button**
* кнопка отменить бронь **booking-cancel-button**

****

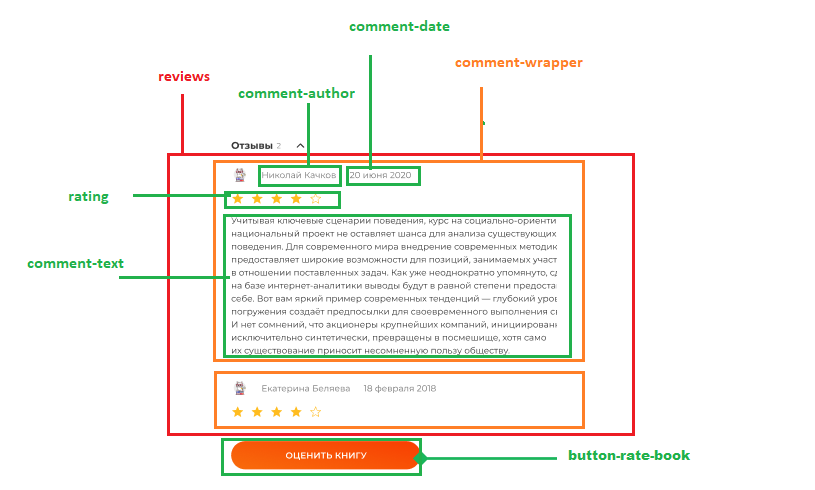
Для компонента рейтинга (в модалке, комментариях итд.):

* обертка **rating**
* обертка звезды **star**
* для активной звезды **star-active**



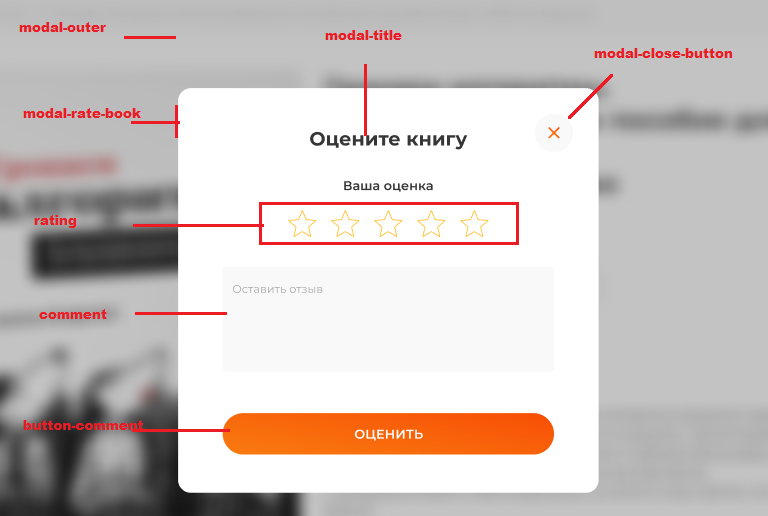
Для комментариев на странице книги:

* секция комментариев **reviews**
* обертка комментария **comment-wrapper**
* автор комментария **comment-author**
* дата комментария **comment-date**
* рейтинг **rating (см. выше)**
* текст комментария **comment-text**
* кнопка оценить книгу **button-rate-book**



Для модалки отзыва:

* подложка **modal-outer**
* обертка модалки **modal-rate-book**
* заголовок **modal-title**
* кнопка закрытия **modal-close-button**
* выбор рейтинга **rating (см. выше)**
* поле комментария **comment**
* кнопка оценить **button-comment**



Для алертов :

переиспользуем из прошлых спринтов компонент с data-test-id=error

для любого алерта

* обертка **error** (успех/ошибка)
* крестик **alert-close**

