

JS Doc Generate Documentation:

Home

Home

Modules

Controllers/PaymentController
Controllers/Wishlist
Models/Payment
Models/Wishlist
Routes/PaymentRoutes
__tests__/authController
routes/authRoutes

Namespaces

ReviewRoutes

Classes

ReviewController
UserController
UserModel

Global

addToCart

Activate Windows
Go to Settings to activate Windows.

Module: Controllers/PaymentController

Controller for managing payment functionality.

Source: [controllers/PaymentController.js, line 1](#)

Methods

(inner) processPayment(order) → {Promise.<Object>}

Process payment for an order.

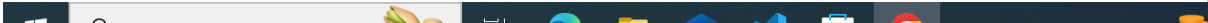
Parameters:

| Name | Type | Description |
|-------|--------|--|
| order | Object | The order object containing payment details. |

Source: [controllers/PaymentController.js, line 8](#)

Returns:

The payment confirmation.



Methods

`(inner) processPayment(order) → {Promise.<Object>}`

Process payment for an order.

Parameters:

| Name | Type | Description |
|-------|--------|--|
| order | Object | The order object containing payment details. |

Source: [controllers/PaymentController.js, line 8](#)

Returns:

The payment confirmation.

Type

Promise.<Object>

Namespace: ReviewRoutes

ReviewRoutes

Router for managing review operations.

Source: [routes/reviewRoutes.js, line 18](#)

Methods

`(static) POST /api/books/:bookId/reviews(req, res) → {Promise.<void>}`

Parameters:

| Name | Type | Description |
|------|---------------------------------|-------------------------|
| req | ExpressRequest | Express request object |
| res | ExpressResponse | Express response object |

Source: [routes/reviewRoutes.js, line 23](#)

Class: ReviewController

ReviewController()

Controller for managing review operations.

Constructor

`new ReviewController()`

Source: [controllers/ReviewController.js, line 8](#)

Methods

`(async) createReview(req, res) → {Promise.<void>}`

Create a new review for a book.

Methods

`(async) createReview(req, res) → {Promise.<void>}`

Create a new review for a book.

Parameters:

| Name | Type | Description |
|------|--------|-------------------------|
| req | object | Express request object |
| res | object | Express response object |

Source: [controllers/ReviewController.js, line 15](#)

Returns:

- Promise representing the operation completion

Type

Promise.<void>

Global

Members

(constant) **Book**

Mongoose model for a book.

Source: [models/Book.js, line 38](#)

(constant) **Review**

Mongoose model for a review.

Source: [models/ReviewModel.js, line 48](#)

author :string

The author of the book.

author :string

The author of the book.

Type:

- string

Source: [models/Book.js, line 25](#)

bookId :string

The ID of the associated book.

Type:

- string

Source: [models/ReviewModel.js, line 22](#)

(constant) **bookSchema**

Mongoose schema for a book.

Source: [models/Book.js, line 15](#)

`rating :number`

The rating given in the review.

Type:

- number

Source: [models/ReviewModel.js, line 36](#)

`(constant) reviewSchema`

Mongoose schema for a review.

Source: [models/ReviewModel.js, line 17](#)

`title :string`

The title of the book.

Type:

- string

Source: [models/Book.js, line 20](#)