



```
checkout(products: ProductRequest[]){
  return this.http.post<{url: string}>({ url: "http://localhost:4000/payments/checkout", products})
}
```

```
@Post( path: 'checkout')
@UseGuards(new AuthGuard())
@Role(UserRoleEnum.CUSTOMER)
async createPayment(
  @User( dataOrPipes: 'id') userId: string,
  @Body() cartItems: CartItem[],
) {
  const session = await this.paymentService.processPayment(userId, cartItems);
  return {url: session.url}
}
```

```
next: (res: {url: string}) => {
  window.location.href = res.url
}
```

```
// running 'stripe listen --forward-to localhost:4000/payment/webhook' in stripe CLI
@Post( path: 'webhook')
async handleStripeIncomingEvents(
  @Headers( property: 'stripe-signature') signature: string,
  @Req() req: RawBodyRequest<Request>,
) {
  if (!signature) {
    throw new HttpException(
      response: 'Missing stripe-signature header',
      HttpStatus.BAD_REQUEST,
    );
  }

  const event = await this.paymentService.constructEventFromPayload(
    signature,
    req.rawBody,
  );
  return this.paymentService.handleStripeEvent(event);
}
```