

Package com.mapmyjourney.backend.service

## Class TripService

java.lang.Object<sup>2</sup>  
com.mapmyjourney.backend.service.TripService

@Service  
public class TripService  
extends Object<sup>2</sup>

### Constructor Summary

#### Constructors

##### Constructor

##### Description

TripService()

### Method Summary

#### All Methods

#### Instance Methods

#### Concrete Methods

##### Modifier and Type

##### Method

##### Description

void

**addMemberToTrip(Long<sup>2</sup> tripId, Long<sup>2</sup> userId)**

Agrega un usuario a un viaje con rol VIEWER.

##### TripDTO

**createTrip(TripCreateRequestDTO request, Long<sup>2</sup> creatorUserId)**

Crea un nuevo viaje.

void

**deleteTrip(Long<sup>2</sup> tripId, Long<sup>2</sup> userId)**

Elimina un viaje.

##### TripDTO

**getTripByCode(String<sup>2</sup> tripCode)**

Obtiene un viaje por su código (para invitaciones).

##### TripDTO

**getTripById(Long<sup>2</sup> tripId)**

Obtiene un viaje por ID.

##### List<sup>2</sup><TripDTO>

**getUserTrips(Long<sup>2</sup> userId)**

Obtiene todos los viajes del usuario.

##### TripDTO

**updateTrip(Long<sup>2</sup> tripId, TripCreateRequestDTO request, Long<sup>2</sup> userId)**

Actualiza un viaje existente.

void

**verifyUserInTrip(Long<sup>2</sup> tripId, Long<sup>2</sup> userId)**

Verifica que un usuario pertenece a un viaje.

## Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructor Details

### TripService

```
public TripService()
```

## Method Details

### createTrip

```
@Transactional  
public TripDTO createTrip(TripCreateRequestDTO request,  
                           Long creatorUserId)
```

Crea un nuevo viaje.

### getTripById

```
@Transactional(readOnly=true)  
public TripDTO getTripById(Long tripId)
```

Obtiene un viaje por ID.

### getUserTrips

```
@Transactional(readOnly=true)  
public List<TripDTO> getUserTrips(Long userId)
```

Obtiene todos los viajes del usuario.

### addMemberToTrip

```
@Transactional  
public void addMemberToTrip(Long tripId,  
                           Long userId)
```

Agrega un usuario a un viaje con rol VIEWER.

### getTripByCode

```
@Transactional(readOnly=true)  
public TripDTO getTripByCode(String tripCode)
```

Obtiene un viaje por su código (para invitaciones).

### verifyUserInTrip

```
@Transactional(readOnly=true)  
public void verifyUserInTrip(Long tripId,  
                           Long userId)
```

Verifica que un usuario pertenece a un viaje.

### updateTrip

```
@Transactional  
public TripDTO updateTrip(Long tripId,  
                           TripCreateRequestDTO request,  
                           Long userId)
```

Actualiza un viaje existente. Solo el OWNER puede actualizar.

## **deleteTrip**

```
@Transactional  
public void deleteTrip(Long✉ tripId,  
                        Long✉ userId)
```

Elimina un viaje. Solo el OWNER puede eliminar.