

**Package** com.mapmyjourney.backend.service

## Class TripService

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

@Service  
public class **TripService**  
extends Object<sup>↗</sup>

### Constructor Summary

Constructors
Constructor
Description
<a href="#">TripService()</a>

### Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
void	<a href="#">addMemberToTrip(Long<sup>↗</sup> tripId, Long<sup>↗</sup> userId)</a>	Agrega un usuario a un viaje con rol VIEWER.
<b>TripDT0</b>	<a href="#">createTrip(TripCreateRequestDT0 request, Long<sup>↗</sup> creatorUserId)</a>	Crea un nuevo viaje.
void	<a href="#">deleteTrip(Long<sup>↗</sup> tripId, Long<sup>↗</sup> userId)</a>	Elimina un viaje.
<b>TripDT0</b>	<a href="#">getTripByCode(String<sup>↗</sup> tripCode)</a>	Obtiene un viaje por su código (para invitaciones).
<b>TripDT0</b>	<a href="#">getTripById(Long<sup>↗</sup> tripId)</a>	Obtiene un viaje por ID.
<b>List<sup>↗</sup>&lt;TripDT0&gt;</b>	<a href="#">getUserTrips(Long<sup>↗</sup> userId)</a>	Obtiene todos los viajes del usuario.
<b>TripDT0</b>	<a href="#">updateTrip(Long<sup>↗</sup> tripId, TripCreateRequestDT0 request, Long<sup>↗</sup> userId)</a>	Actualiza un viaje existente.
void	<a href="#">verifyUserInTrip(Long<sup>↗</sup> tripId, Long<sup>↗</sup> userId)</a>	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.