

1 Use cases

Below the 8 use cases are listed

1.1 Buy one way ticket

Participating actors:

Initiated by Traveler

Flow of events:

- The Traveler chooses to buy one way ticket
 - The Traveler pays
 - The ticket is printed
-

Entry conditions:

The ticket distributor has change and paper

Exit conditions:

The Traveler has received the ticket, **OR** the Traveler has received an explanation indicating why the transaction can't be completed.

Quality requirements:

—

1.2 Buy weekly card

Participating actors:

Initiated by Traveler

Flow of events:

- The Traveler chooses to buy weekly card
 - The Traveler pays
 - The card is printed
-

Entry conditions:

The ticket distributor has change and paper

Exit conditions:

The Traveler has received the card, **OR** the Traveler has received an explanation indicating why the transaction can't be completed.

Quality requirements:

—

1.3 Buy monthly card

Participating actors:

Initiated by Traveler

Flow of events:

- The Traveler chooses to buy monthly card
 - The Traveler pays
 - The card is printed
-

Entry conditions:

The ticket distributor has change and paper

Exit conditions:

The Traveler has received the card, **OR** the Traveler has received an explanation indicating why the transaction can't be completed.

Quality requirements:

—

1.4 Update tariff

Participating actors:

Initiated by CentralComputer

Flow of events:

- The CentralComputer updates the tariff on the distributor.
 - Distributor confirms the tariff has been updated
-

Entry conditions:

There has been changes to the tariff

Exit conditions:

The Distributors tariff has been updated

Quality requirements:

—

1.5 Distributor Exception

Participating actors:

Communicates with Traveler

Flow of events:

- This use case extends BuyOneWayTicket, BuyWeeklyCard and BuyMonthlyCard use cases. It is initiated whenever a problem occurs, as the Traveler is trying to buy a ticket.
-

Entry conditions:

This use case extends BuyOneWayTicket, BuyWeeklyCard and BuyMonthlyCard use cases. It is initiated whenever a problem occurs, as the Traveler is trying to buy a ticket.

1.6 Timeout

Participating actors:

Inherited from DistributorException use case

Flow of events:

- Cancel transaction
 - Notify Traveler that too much time has been spent at one stage of the transaction
-

Entry conditions:

Inherited from DistributorException, **AND** XX time has been spent on one stage of the transaction.

Exit conditions:

User has be notified

1.7 Transaction aborted

Participating actors:

Inherited from DistributorException use case

Flow of events:

- Cancel transaction
 - Notify the traveler that the transaction has been aborted
-

Entry conditions:

Some event has aborted the transaction

Exit conditions:

Traveler has been notified

Quality requirements:

—

1.8 Distributor out of paper

Participating actors:

Inherited from DistributorException use case

Flow of events:

- Notify the traveler that the distributor is out of paper
-

Entry conditions:

The distributor is out of paper

Exit conditions:

The distributor has been refilled with paper

Quality requirements:

—

1.9 Distributor out of change

Participating actors:

Inherited from DistributorException use case

Flow of events:

- Notify the traveler that the distributor is out of change
-

Entry conditions:

The distributor is out of change

Exit conditions:

The distributor has been refilled with change

Quality requirements:

—
