This document explains how the feedbacks system works.

There are three ways to express user ratings (corresponding to three different objects):

- AsDriverFeedback
- AsPassengerFeedback
- GeneralFeedback

The AsDriverFeedback object contains the average ratings for the user when driver (owner) and precisely contains these fields:

- User global vote as driver (avg)
- User reliability as driver (avg)
- User punctuality as driver (avg)
- User courtesy as driver (avg)
- User feedback as driver count (= number of as driver feedback left for the user)

The AsPassengerFeedback object contains the average ratings for the user when passenger and precisely contains these fields:

- User global vote as passenger (avg)
- User reliability as passenger (avg)
- User punctuality as passenger (avg)
- User feedback as passenger count (= number of as passenger feedback left for the user)

The GeneralFeedback object contains the averages beetween ratings as passenger and as driver:

- User global vote (avg)
- User reliability (avg)
- User punctuality (avg)

More info about these objects can be found in the StatsModuleDocumentation.chm

---

When an user requests to join a trip and is accepted by the owner, are inserted in the database two pending feedbacks: a pending feedback from owner to user and from the user to the owner, so at the end of the trip the can send the feedback (negative or positive) about the other one.

On a trip join is called this method to insert related pending feedbacks in the database (located in StatsModule):

```
public static bool insertPendingFeedbacksOnPSNTripJoin(int tripID, int psnID, bool vehicleIsCar)
```

Pending feedbacks are requests to leave a feedback; when an user should leave a feedback opening his homepage he will find a notification about “You have 1 feedback to leave”.
Pending feedbacks can be left after one day from the trip departure date (it's a parameter, can be changed in dataFile.xml) and expire 60 days after the trip departure date (same thing, it's a parameter). When a pending feedback expires (because the user don't leave it in 60 days), it's removed from the database and cannot be left anymore.

If an user decides to unsubscribe from a trip the pending feedback from the user to the trip owner is removed but not the pending feedback from the owner to him, so the owner can send a negative feedback to the user if he wants. Same thing happens if an owner removes a created trip and an user has joined that trip already. The owner cannot send a feedback to the accepted user, but the user can send a negative feedback to the owner if he wants.

More info about these objects and methods can be found in the StatsModuleDocumentation.chm

---

When an user opens his homepage, he will find the number of feedbacks he has to leave.
To retrieve the number of pending feedback the user has to leave, should be called this method located in StatsModule:

public static int hasPendingFeedbacks (int userID)

This method returns the number of pending feedback the userID has to leave.

To retrieve the trips infos which the pending feedbacks to leave are related, can be called this method located in StatsModule:

public static List<TripProperties> getUserTripsWithPendingFeedbacks(int userID)

which returns a list of TripProperties objects containing infos about each trip such as departure date, departure address, arrival address.....

To retrieve all pending feedbacks related to a specified trip, can be called this method located in StatsModule:

public static List<FeedbackRequest> getPendingFeedbacksForTrip(int userID, int tripID)

which returns a list of pending feedbacks to leave for that trip.

When an user wants to release a feedback, he has to fill out the insert feedback form, then by click the send button, is called this method located in StatsModule:

public static bool insertReleasedFeedback(ReleasedFeedback releasedFeedback)

which has a parameter: a ReleasedFeedback object created with the ratings left by the user. This method returns true if the operation is succeeded, false otherwise.

When a feedback is released, the feedback is stored in the database, the receiver ratings averages are updated and related pending feedback is removed from the PendingFeedback table in the db. If the operation fails, nothing is written in the database and the pending feedback is still to be left.

More info about these objects and methods can be found in the StatsModuleDocumentation.chm