# Connection to the database

The connection between Java and database is established by using standard Java API - JDBC (Java Database Connectivity). The latest version of JDBC Driver (PostgreSQL JDBC 4.2 Driver, 42.2.2) is imported in order to access to the database data.

blabla

The database is placed on eNTe’s private server with IP address - 207.154.237.196

blabla

Adapter design pattern

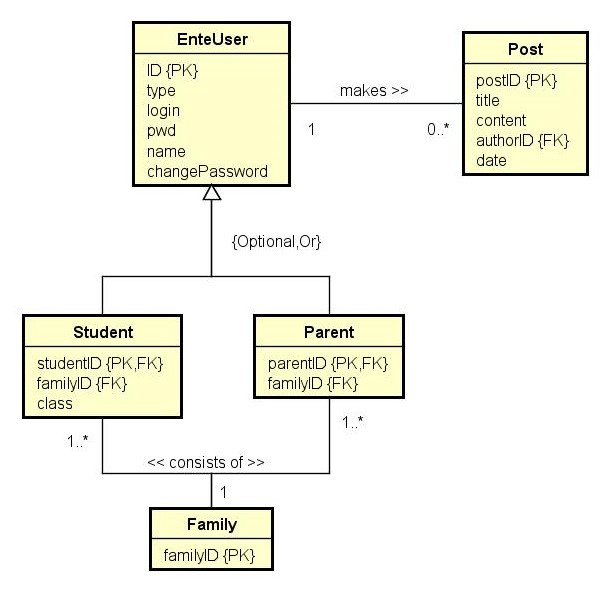
Adapter design pattern is chosen for this part of the system. The main reason for using this design pattern is that it makes the system more clear/clearer, divides responsibility and provides the possibility to change database without any change needed in another part of the system. In eNTe system DBAdapter class is the adapter class that is responsible for transforming data from Java language into SQL and the other way, too.

# Documentation DB design for managing users

There was a need to create database for storing data about users of this system. Exact tasks for a database were following:

1. For every of 4 types of users (Administrator, Teacher, Student and Parent) eNTe wants to keep track of user’s name, login and password.
2. For every student and parent is needed to keep track of family that he/she belongs to. Student or parent can belong only to one family and a family can consist of multiple students (children) and parents.
3. Only for every student eNTe needs to hold student’s current class. Therefore, a student can be part only of one class and a class can consist of multiple students.

The EER diagram below presents possible solution for tasks mentioned above.



At the moment, there is no need to make Teacher or Administrator as an entity as they don’t hold any specific data in comparison with enteuser. Therefore only Student and Parent are made as separate entities that hold specific data about family and also about class in the case of student.

Following logical database design shows the primary and foreign keys for given entities.

**EnteUser** (ID,type,login,pwd,name,changePassword)

PK: ID

**Post** (postID,title,content,authorID,date)

PK: postID

FK: authorID REFERENCES EnteUser (ID)

**Student** (studentID,familyID,class)

PK: studentID

FK: studentID REFERENCES EnteUser (ID)

FK: familyID REFERENCES Family (familyID)

**Parent** (parentID,familyID)

PK: parentID

FK: familyID REFERENCES Family (familyID)

Physical database was based on previous analysis (EER diagram, logical database design). It can be seen in the database visualization below.

