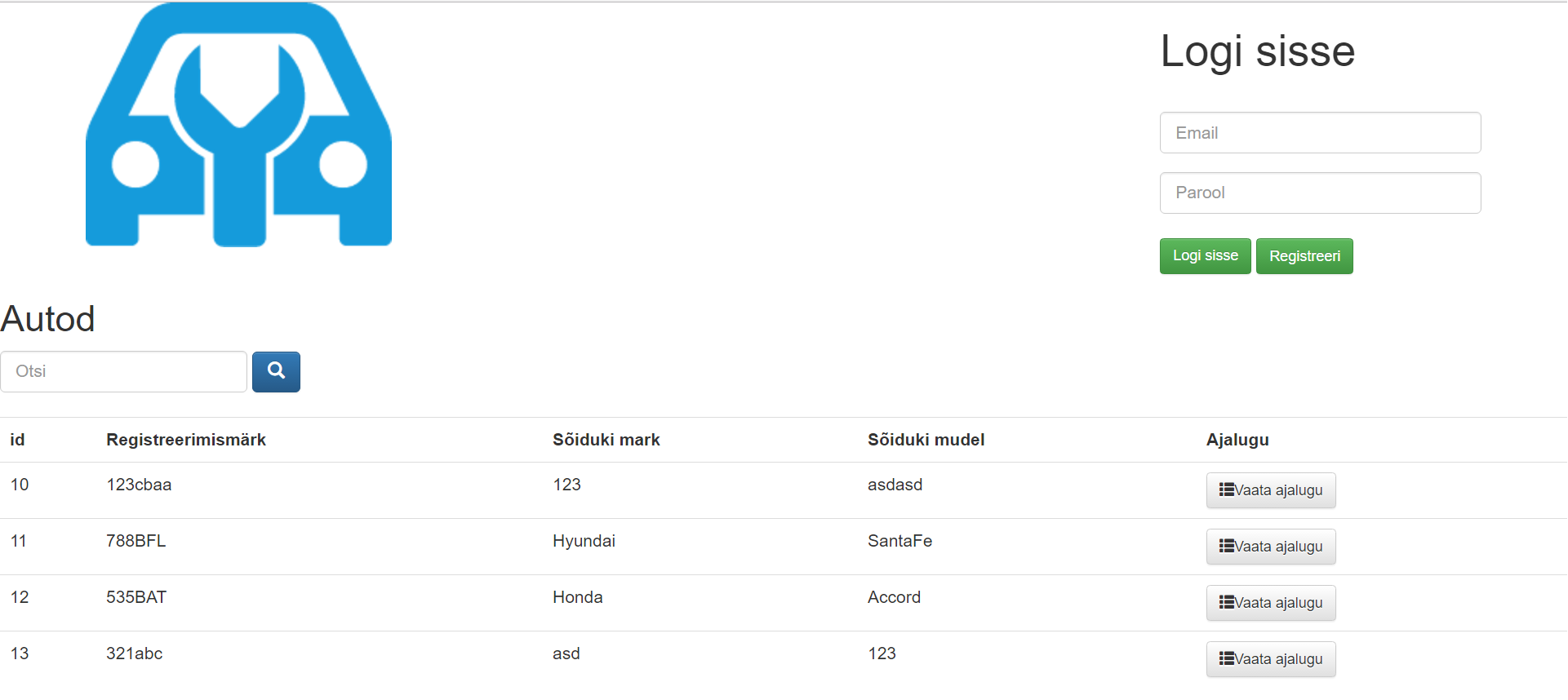
RepairHistory



CREATE TABLE repairUsers(id INT NOT NULL PRIMARY KEY, email VARCHAR(50), password VARCHAR(128), firstname TEXT, lastname TEXT, deleted DATETIME);

CREATE TABLE repairCars(id INT NOT NULL PRIMARY KEY, UserId INT, RegPlate VARCHAR(10), Mark VARCHAR(20), Model VARCHAR(20), deleted DATE, FOREIGN KEY(UserId) REFERENCES repairUsers(id));

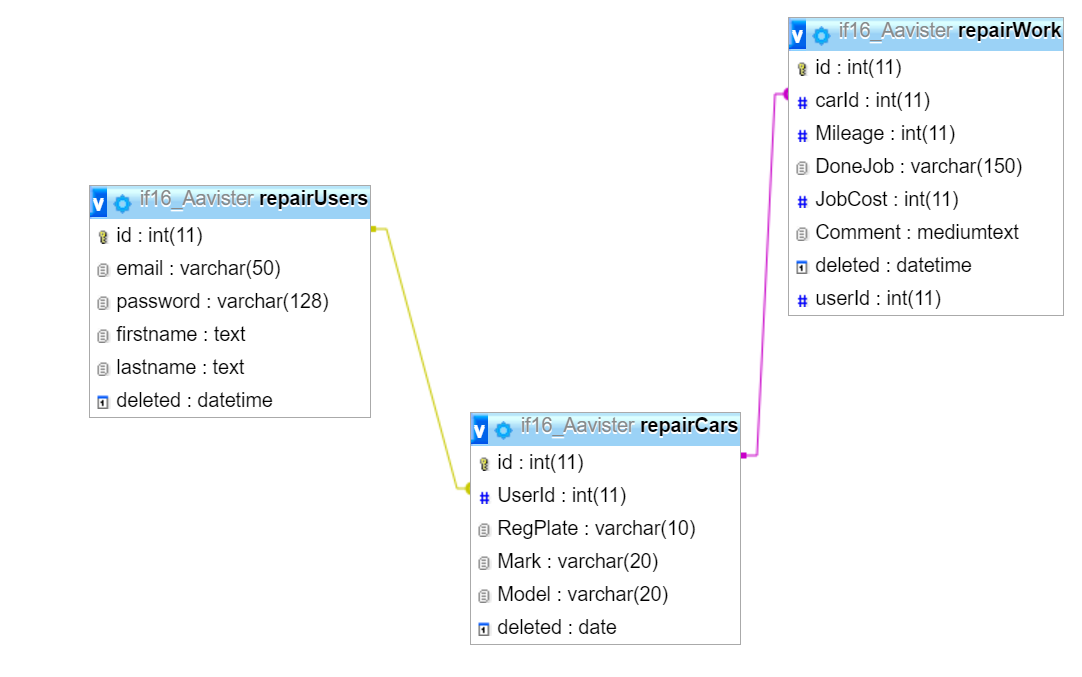
CREATE TABLE repairWork(id INT NOT NULL PRIMARY KEY, carId INT, Mileage INT, DoneJob VARCHAR(150), JobCost INT, Comment MEDIUMTEXT, deleted DATETIME, userId INT, FOREIGN KEY(carId) REFERENCES repairCars(id), FOREIGN KEY(userId) REFERENCES repairUsers(id));

Eesmärgiks seadsime endale tuua inimestele sõidukite ajaloo silme ette. Näiteks kui kellelgi on soov minna ostma autot, siis saab ta meie lehele sisestada sõiduki numbrimärgi ning seejärel vaadata mida on masinal remonditud ning mis väärtuses on see tehtud. Kasutajad saavad lihtsasti lisada tehtud töid ning hoida meeles mida on tehtud.

Sihtrühmaks on peamiselt sõidukiomanikud ning samuti ka inimesed, kes hakkavad omale sõidukit soetama. Samalaadseid rakedusi meie ei leidnud internetist.

Funktsionaalsuse loetelu:

* v0.1 Saab luua kasutaja ning sisselogida
* v0.2 Saab otsida/lisada sõidukeid
* v0.3 Saab lisada tehtud remonditöid, läbisõitu jne



Andmebaasi skeem

# Kokkuvõte

Ats – Õppisin juurde esmalt, kuidas tööd jaotada ning arvestada sellega, kui tuleb tööd teha kellegagi koos. Alguses oli kõige raskem tehtud tööd teisele edastada kasutades putty-t, kuid lõpuks sai seegi selgeks.

Rasmus – Kõige arendavam ja samuti ka keerulisem tegevus oli kodus üksinda koodi kirjutamine ja probleemidele lahenduste otsimine. Kindlasti oskan vastavatele *error*itele nüüd palju paremini lahendust leida ja kellegagi paaris tööd paremini jaotada.