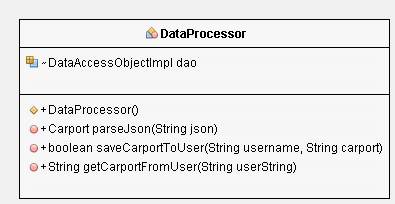
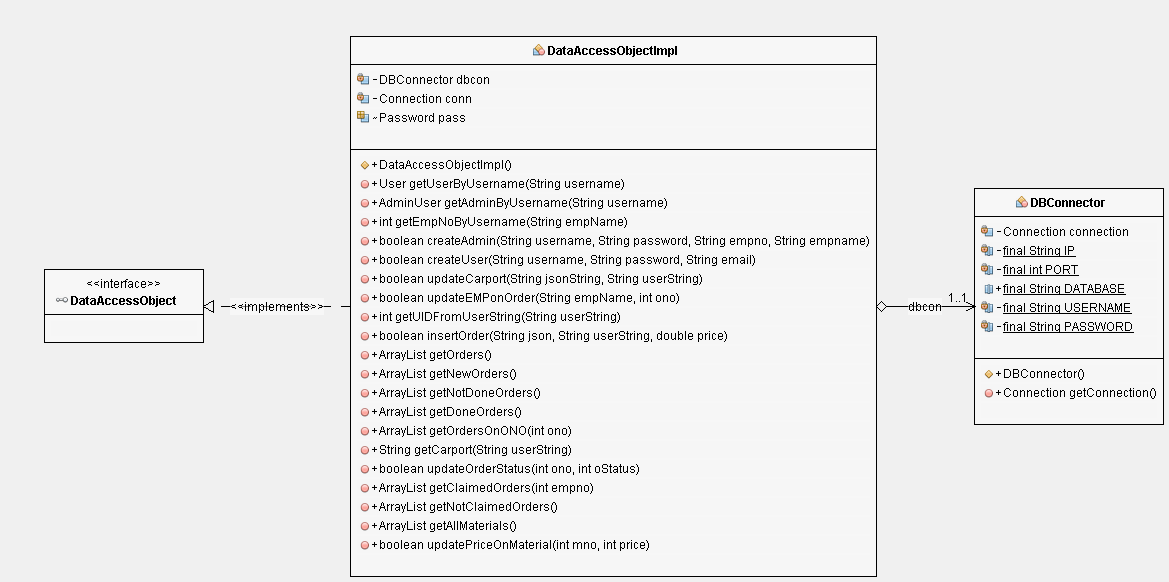
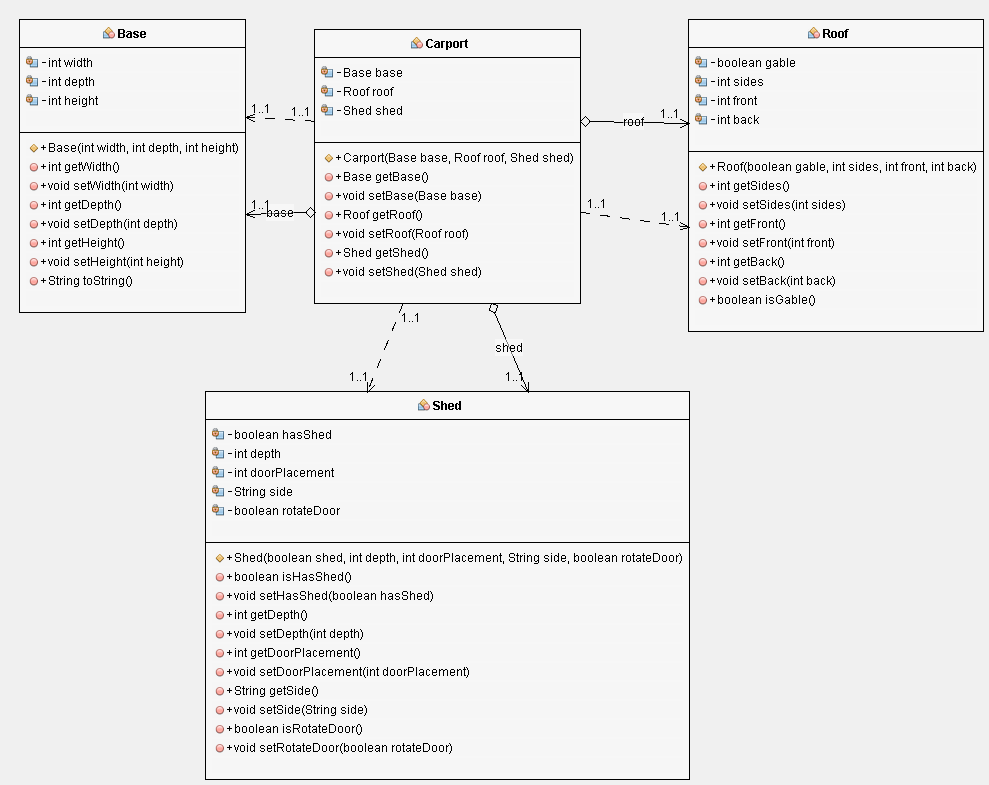
Selvom vores program er trelags arkitektur har vi valgt at sortere de enkelte classes i packages ud fra deres direkte tilhørsforhold. Selvom nogen classes importerer fra andre classes i andre packages, har vi valgt at opdele klassediagrammet i enkelte packages, for på den måde at forbedre overskueligheden.



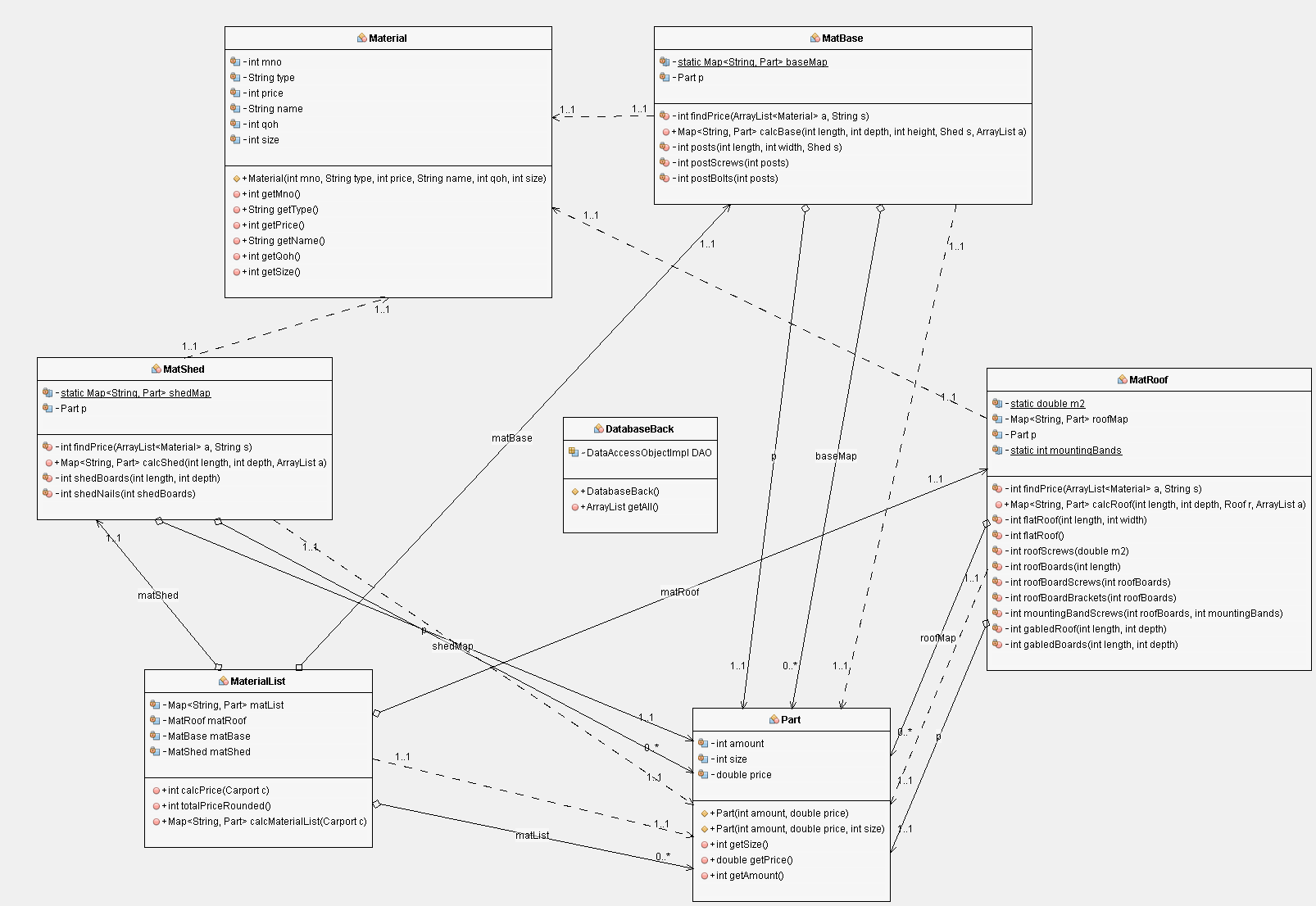
Vores backend package indeholder kun DataProcessor klassen, der bruges til både at lave en JSON string til at gemme en carport i databasen og til at hente en JSON string fra databasen og lave et carport object til at udregne prisen på en carport og generere en MaterialList ud fra denne.



Vores Database package har et DataAccessObject interface og dets implement klasse. Database indeholder også en DBConnector class til at kontakte selve databasen.



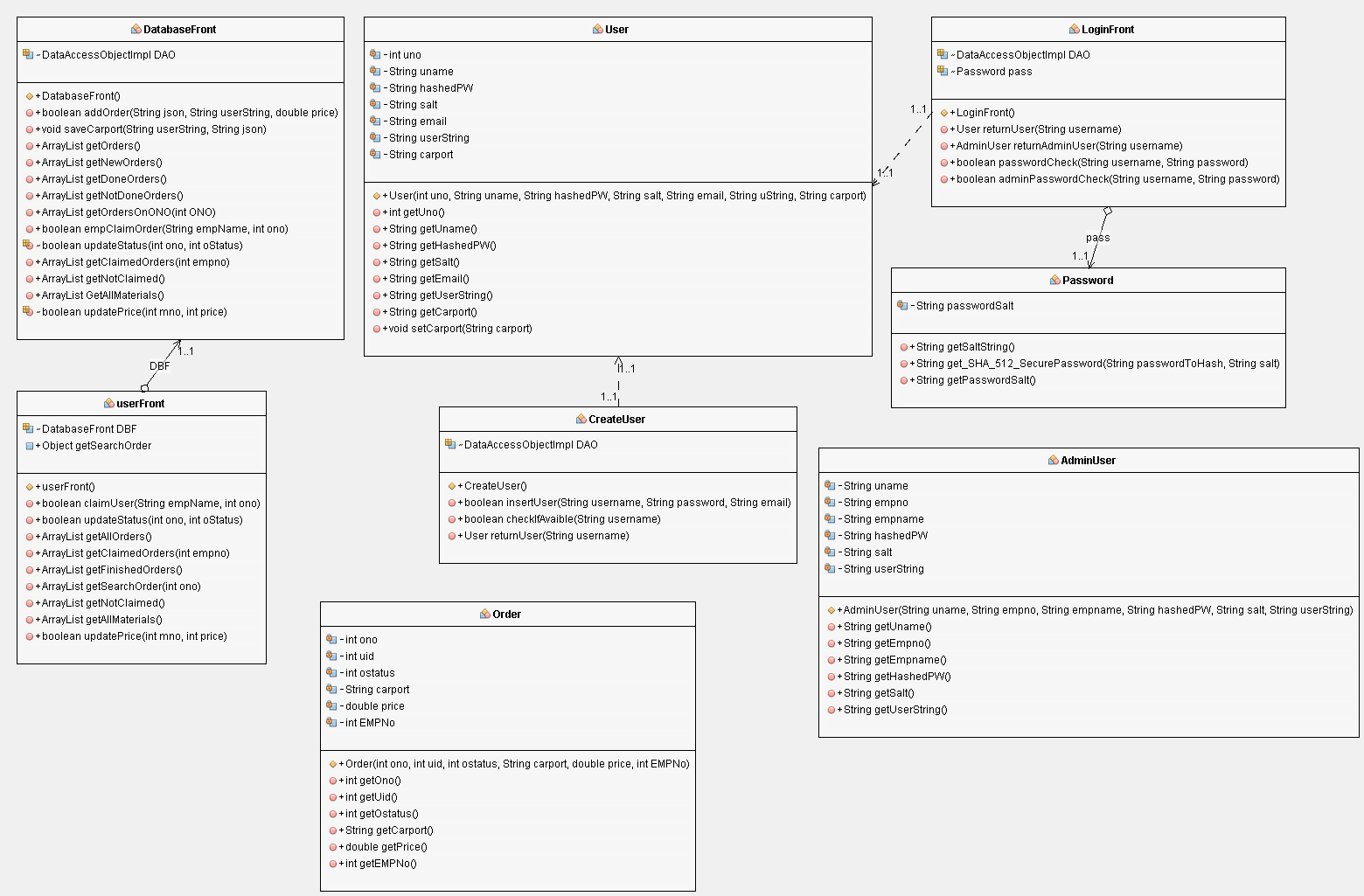
Carport package bruges til at opbevare info fra en JSON string fra vores database. Hver Carport object indeholder et Shed, et Base og et Roof object.



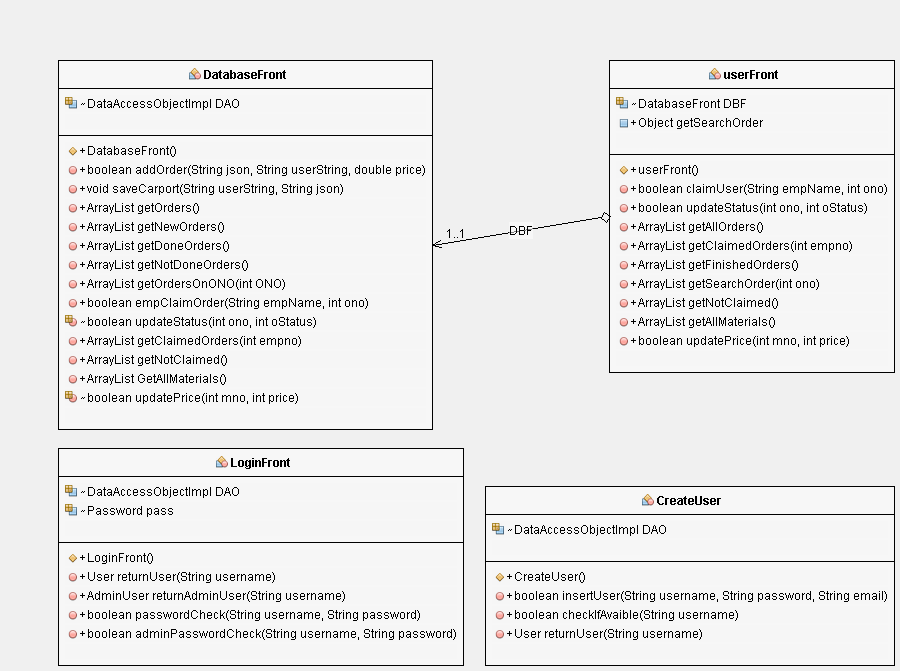
Vores MaterialList package indeholder alle de klasser der bruges til at oprette en MaterialList(stykliste) og ud fra den udregne en pris for materialer. Packagen indeholder også to constructor classes for objectsne Material og Part. Material bruges til midlertidigt at opbevare databasens material table. Part bruges da MaterialList opbevares som Hashmap med String som key og Part som value.



I vores servlet package ligger hele vores presentation layer. Alt hvad en bruger ser bliver ført gennem denne package.



User packagen indeholder constructor classes til user info og til oprettelse af nye users.



User.Logic indeholder de classes der kommunikerer user info mellem database og frontend.