# Valget for repository patternet

Et repository pattern er et abstraktionslag til databasen for at simplificere koden. Da patternet giver et ekstra abstraktionslag, kan det benyttes på alle databaser, selv om de bruger Entity Framework, ADO.NET eller lignende, da Buisness logic layer (BLL) kalder metoder gennem interfaces. Derudover giver repository pattern mulighed for at unit teste BLL i stedet for at integrationsteste det, netop på grund af det abstraktionslag som kan mockes ud.

Der blev af disse grunde valgt at implementere et repository pattern i Pristjek220, så det netop er muligt at unit teste programmets BLL. Repository patternets abstraktionslag giver dermed også Pristjek220 mulighed for at kunne benytte sig af databaser, som ikke bruger Entity Framework. Så hvis der i fremtiden skulle blive udarbejdet noget nyere og bedre, eller der bare bliver besluttet, at det ikke skal køre på Enity Frameworket længere, så kan programmet nemt skiftes over.



Figur 1: Implementering af Repository pattern i Pristjek220.

På Figur 1 kan der ses hvordan Repository patterenet er blevet implementeret i projektet. Hvor de forskellige repositories indeholder CRUD[[1]](#footnote-1) funktionerne for de tabeler de hører til altså ProductRepository indeholder funktionerne til Product tabellen i databasen. I Repository klassen som de specifikke repositories nedarver fra er de generelle funktioner som Add og Remove for at undgå duplikeret kode. Unit of Work er lavet som et access point til repositoriesne fra buisness logic layeret.

1. Create, Read, Update og Delete [↑](#footnote-ref-1)