**Översikt**

Klassdiagrammet i Figur 6 visar komponenterna i systemet och hur de är utformade för att fungera med varandra. Det ger idén om hur systemet är utformat och riktlinjerna för hur det kan utvecklas i framtiden.

**Funktinaliteten:**

Arkitekturen som används har tre olika designmönster som gör dess funktionalitet.

**Abstract factory mönster:** vilket är ett designmönster som används för att underlätta skapandet av objekt utan att visa klienten någon detaljerad information. Vi använder det här mönstret för att skapa produkterna. Detta mönster är också lätt att förlänga i framtiden när du lägger till nya kategorier eller nya produkter i befintliga kategorier.

**Observer mönster:** Med hjälp av detta mönster kan vi uppdatera informationen relaterad till produkterna när det gäller deras existens eller information relaterad till dem som beskrivningen eller priset

**Singelton mönster**: vi använde detta designmönster i objekten som vi behöver ha en instans av dem på varje instans av applikationen som till exempel Cart klassen (Varukorgen).

**Implementation**

Varje kategori implementeras som en abstrakt klass som implementerar Product interafcet och objekten i kategorin arver den klassen. Det gör det lättare att ange varje kategoris specifikationer och att nya kategorier kan läggas till.

Varje kategori har sin egen Factory klass som implementerar factory interface och är ansvarig för att skapa objekten. Klassen FacOfFactories användes för att välja vilket Factory som ska anropas.

Databasehandler klassen är den klass som ansvarar för att uppdatera produkterna i databaserna