# Use Cases/Scenarier

## Jesper

### Use case 1

Kunde søger efter spil baseret på flere kriterier. Derefter tilføjer de ønskede spil til indkøbskurv, indtaster leveringsønsker udfra muligheder og afslutter bestilling med betaling.

### Scenario

Jesper er glad for DnD og prøver derfor at søge efter dem. Han ved at det er i ”fantasy” genren, og søger derfor i søgefeltet med DnD. Der kommer resultater med ”DnD” ”dungeons & dragons” og ”pathfinder”, sidste er ikke DnD, men er samme spil type som han kunne være interesseret i.

### Use case 2

Kunde opretter ønske om brætspil som bliver gemt i systemet som ny vare der skal indkøbes.

### Scenario

Jakob kunne godt tænke at købe ”Cluedo”, men Genspil har ikke nogen spil på lager. Efter at have søgt forgæves, foreslår hjemmesiden at han opretter en bruger og tilføjer spillet til en ønskeliste.

### Use case 3

Ansat tilføjer flere spil til lager i forskellig tilstand, med forskellig prissætning.

### Scenario

Rafed har modtaget en masse spil i forskellig stand som han skal tilføje til databasen. Han har et excel ark med priser på de forskellige spil og navne som han omdanner til CSV så han hurtigt kan uploade listen til lageret, som selv kan oprette objekter fra filen.

### Use case 4

Ansat udskriver liste med inventar der kan sorteres efter navn og genre.

### Scenario

Peter skal lave lageroptælling og vil derfor gerne printe databasen ud i en bestemt struktur. Efter han trykker export spørger system om hvordan han vil strukturere det. Han vælger at sortere først efter genre, derefter navn.

### Use case 5

Kunde udfylder “ticket” angående en fejl på hjemmesiden.

### Scenario

Jeppe har opdaget at der ingen leveringsmuligheder er til hans område. Efter at have oprettet en profil skriver han en ticket til Genspil der beskriver problemet, med kommentarer, type af problem valgt i dropdown, og for bagefter en email om at hans ticket er modtaget og vil blive gennemgået af en ansat ved snareste mulighed.

## Jeppe

### Jeppe vil registrere sin egen profil I systemet

Jeppe laver et username passsword, og bruger sin email

### Jesper vil registrere et nyt spil I systemet

Jesper logger ind med sit username og password

Han trykker på "tilføj spil" funktionen

Han skriver nu detaljer om spilet (navn, genre, antal af spillere, pris osv.)

Han gemmer nu spillet I systemet og systemet opdaterer inventarlisten

### Jakob vil søge efter et spil I systemet

Han logger ind I systemet

Han tryker på '"søg spil"

Han vil søge efter navn og taster derfor "navn: seq"

Systemet finder nu alle spil der indeholder "seq" I deres navn og udskriver dem

### Rafed vil fjerne et spil fra inventaren

Han logger ind I systement

Han trykker på "Fjern spil" og kan nu se alle spil I inventaren

Nu indtaster han ID'et på det spil han vil fjerne

Systemet beder ham om at bekræfte

Han trykker ja

Systemet opdaterer nu spillets inventor

## Jakob

### Lager: Kombineret søgning

Jonas vil gerne tjekke, hvilke spil de har for 4 personer eller flere som mindst er I nogenlunde stand. Han indtaster de to kriterier I lagersystemet og får en liste tilbage over alle spil, der opfylder kravene.

### Lager: Fjern spil ved køb

Jonas sælger et spil i butikken og går ind i lagerstyringsprogrammet for at notere, at spillet er solgt. Han søger på navn og pris, da søgningen returnerer det ønskede resultat, og fjerner spillet fra lageroversigten.

### Lager: Tilføj nyt spil, der er blevet forespurgt

En potentiel ny kunde, Poul, har været inde i GenSpil--butikken og forespurgt, om de har Backgammon på lager. Jamal søger i lagersystemet på Backgammon, men uden at finde et resultat. Han opretter derfor et nyt spil, Backgammon, i lagersystemet, og noterer, at det er på forespørgsel fra Poul. Han modtager Pouls kontaktoplysninger og fortæller Poul, at han vil vende tilbage, når han ved, om de kan få spillet hjem.

### Lager: Udskriv lageroversigt

Emilie vil gerne gøre lagerstatus, og beder lagersystemet om at printe en komplet oversigt over alle spil, de har. Listen hun får er dog rodet, så hun vælger at sortere spillene efter deres stand og printer listen igen.

## Rafed

UC1: Tilføjelse af et nyt spil til lageret:

En medarbejder logger ind i systemet.

Vedkommende navigerer til "Tilføj nyt spil" sektionen.

Indtaster spillets navn, genre, antal spillere, stand og pris.

Hvis spillet ikke er på lager, markeres det som en forespørgsel.

Systemet gemmer oplysningerne.

### UC2: Søgning efter spil

En medarbejder logger ind i systemet.

Vedkommende navigerer til "Søg efter spil" sektionen.

Vælger de ønskede søgekriterier såsom genre, antal spillere, stand, pris osv.

Systemet viser en liste over relevante spil baseret på de valgte kriterier.

### UC3: Opdatering af spillets status

En medarbejder logger ind i systemet.

Vedkommende finder det relevante spil på lageret.

Opdaterer spillets status (f.eks. tilgængelig, reserveret, solgt).

Hvis nødvendigt, opdaterer standen og prisen på spillet.

Systemet gemmer ændringerne og opdaterer lagerets status.

### UC4: Behandling af forespørgsler

En medarbejder logger ind i systemet.

Navigerer til "Forespørgsler" sektionen.

Ser en liste over alle indkomne forespørgsler.

Hvis et ønsket spil er kommet på lager, markeres forespørgslen som behandlet og kontaktoplysningerne noteres.

Systemet opdaterer forespørgslens status.

### UC5: Udskrivning af lagerliste til lageroptælling

En medarbejder logger ind i systemet.

Navigerer til "Lagerliste" sektionen.

Vælger muligheden for at udskrive lagerlisten.

Kan vælge at sortere listen efter navn eller genre.

Systemet genererer og viser en lagerliste, som kan udskrives.

# Objektmodel

A diagram of a flowchart

Description automatically generated

# Domænemodel

A diagram of a computer

Description automatically generated

# Oprindeligt Design Class Diagram

A screenshot of a computer program

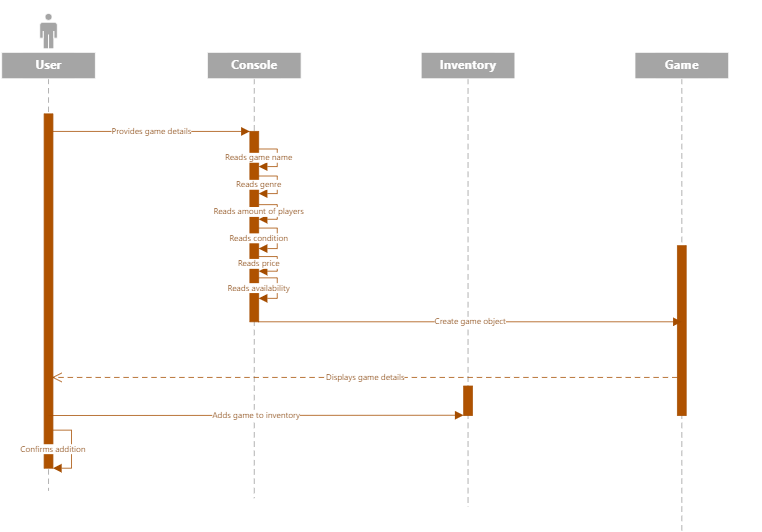
Description automatically generated

A screenshot of a computer program

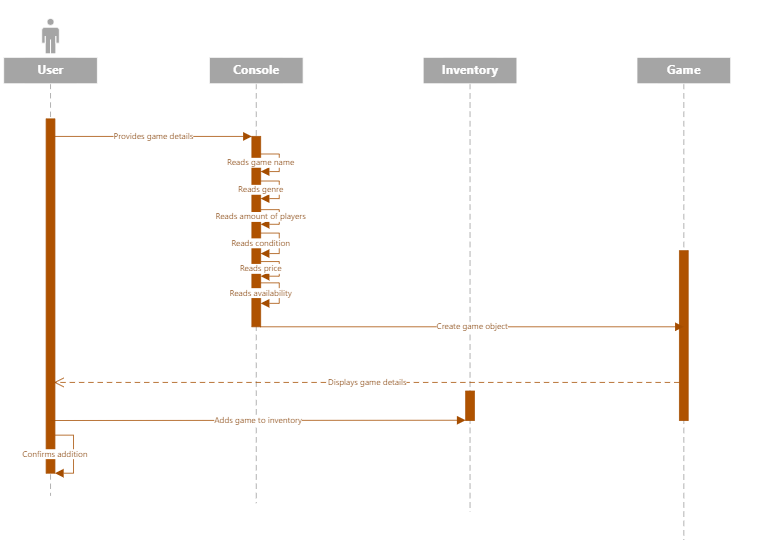
Description automatically generated

# A diagram of a computer Description automatically generatedEndeligt Design Class Diagram

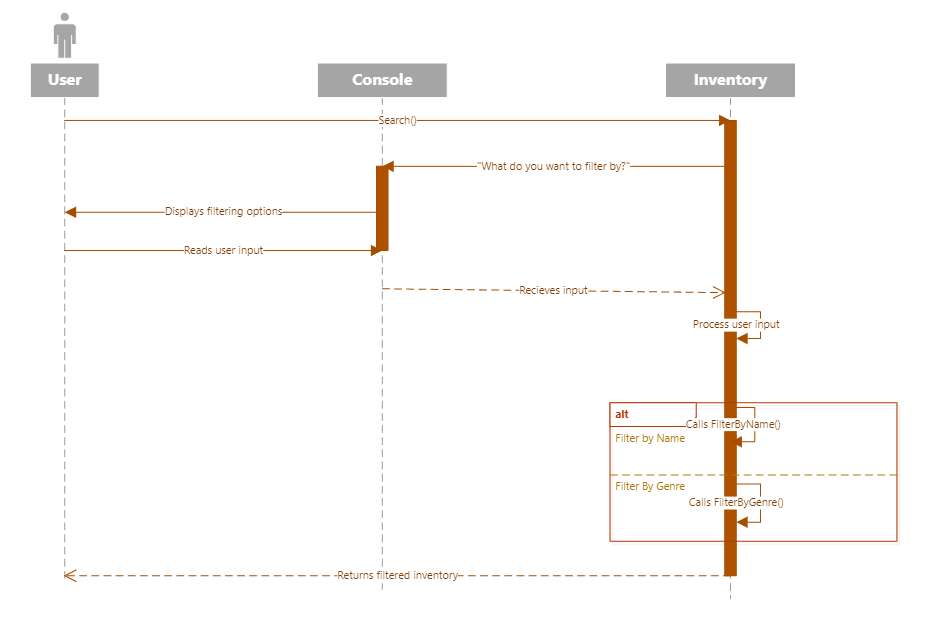
# Sequence Diagram for Filter method



# Sequence Diagram for AddGame method



# Sequence Diagram for InventorySearch method:



# Definition of Done

## Code

* Persistence is implemented and functional
* Arrays and collections are used
* Code follows naming conventions
* Code follows coding conventions
* No build failures
* Unit testing completed and passed
* Integration testing completed and passed
* Internally peer reviewed

## Documentation

* Documentation follows standard conventions
* All documentation completed

# Definition of Test

* Tested using unit testing
* Encapsulation tested and passed
* Methods tested and passed
* Positive test passed
* Negative test passed
* Limit cases test passed