

Project Report

*Mario’s Pizza*

***DAT21b***

**Project Participants:**

*Troels Reinholdt Karn*

*Victor Kertsch-Sahlholdt*

*Lasse Rugaard Rasmussen*

*Document Version: 00.03.00*

*Document Template version: 00.04.00*

Contents

[Document Template History](#_heading=h.gjdgxs) **3**

[Document Version](#_heading=h.30j0zll) **3**

[Vision](#_heading=h.1fob9te) **4**

[Business Rules / Domain Rules](#_heading=h.3znysh7) **4**

[Requirements](#_heading=h.2et92p0) **4**

[Functional Requirements](#_heading=h.4d34og8) 4

[Non-Functional requirements](#_heading=h.2s8eyo1) 4

[Use Cases  
Use case diagram:](#_heading=h.17dp8vu) **5**

[System Sequence Diagram](#_heading=h.uozbg51t8hj6) 9

[Operation Contracts](#_heading=h.26in1rg) **11**

[**Operation contract for use case 1:**](#_heading=h.7mux1s5u6rew) **11**

[Design Class Diagram](#_heading=h.lnxbz9) **12**

[Sequence Diagram](#_heading=h.35nkun2) **13**

[“Optional” – Other Information?  
Figur 01 (SSD til UC01):](#_heading=h.1ksv4uv) **15**

[Glossary](#_heading=h.44sinio) **17**

# Document Template History

| **Date** | **Version** | **Author** | **Description** |
| --- | --- | --- | --- |
| 201x-xx-xx | 00.01.00 | HAAM | Initialized |
| 2019-09-25 | 00.02.00 | HAAM | Section added SS & FURPS cells |
| 2019-11-19 | 00.02.01 | HAAM | Added Operation Contracts |
| 2020-03-09 | 00.03.01 | HAAM | Re-ordered and “{}” replaced with “/\* \*/” |
| 2021-10-11 | 00.04.00 | HAAM | Sections SSD, DCD, SD added. |

# Document Version

| **Date** | **Version** | **Author** | **Description** |
| --- | --- | --- | --- |
| 16-10-2021 | 00.01.00 | Victor, Troels, Lasse | Oprettet vores template og startet udfyldningen. Vi har tilføjet Vision, Business Rules, Requirements(Functional), Use cases, Use case diagram, SSD, Operation contracts, Class diagram, Domain model.   * *Dette er ting vi har arbejdet løbende på siden projektstart og er derfor bare blevet tilføjet til vores template.* |
| 17-10-2021 | 00.02.00 | Victor, Troels, Lasse | Vi har opdateret vores SSD´er. Kapitlet med “options” er blevet udfyldt med bilag der indeholder vores ekstra figurer.  Vi har lavet operations contracts til vores SSD´er. Vi har lagt vores virksomhedsanalyser over i et powerpoint.  Lavet ReadMe fil. |
| 18-10-2021 | 00.03.00 | Victor, Troels, Lasse | Vi har lavet sekvensdiagrammer og glossary. Vi har sørget for at opstillingen af rapporten er blevet optimeret og finjusteret resten af opgaven så den var klar til at blive afleveret. |

# Vision

Marios pizzabar ønsker et program der hjælper dem med at få et overblik.

De ønsker et program der opretter bestillinger, viser et menukort (når det skal bruges) gemmer data til statistik, så de ved hvilken pizza der er mest populær og at de kan se og fjerne bestillinger fra en liste over igangværende bestillinger.

Summary of system Features

* Systemet laver bestillinger
* Systemet udregner statistik
* Systemet håndterer flere bestillinger
* Systemet gemmer og håndterer data

# Business Rules / Domain Rules

Vi følte ikke vi kunne finde nok omkring Business Rules / Domain Rules i Mario´s Pizzaria til at lave en tabel over det. Det eneste vi kom i tanke om var de samme eksempler som blev givet i bogen, såsom behandling af skat og behandling af transaktionsoplysninger. Dog ved vi ikke engang om vores system overhovedet skal behandle sådanne oplysninger.

# Requirements

### Functional Requirements

| **Id** | **Requirement** | **Comment** |
| --- | --- | --- |
| FR001 | Se hele menukort på skærmen |  |
| FR002 | Se en liste med bestillingerne |  |
| FR003 | Hvornår de skal afhentes |  |
| FR004 | Sortere dem efter tidspunkt |  |
| FR005 | Så han kan fjerne den fra listen |  |
| FR006 | Jeg vil gerne kunne gemme alle ordrene |  |
| FR007 | Og senere lave statistik |  |

### Non-Functional requirements

| **Id** | **Requirement** | **Comment** |
| --- | --- | --- |
| NFR001 | Intern kommunikation mellem Alfonso og Mario |  |
| NFR002 | … |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Use Cases **Use case diagram:**

**Beskrivelse af aktører:**  
Vi har to primære aktører igennem vores 3 use cases. Den første er alfonso, som er den person som interagerer med systemet i use case 1 og 2. Han er vedkommende som primært benytter systemet til daglig brug og videregiver information til mario.  
Den anden primære aktør er mario. Han benytter systemet i use case 3, hvilket er når der skal behandles statistik over ordrene.

**Use case 1:**  
I dette use case beskrives det hvordan Alfonso opretter en bestilling i systemet for en kunde og tilføjer de pizzaer som ønskes. Ligeledes udpensles det at de bliver sendt videre i systemet på sorteret vis.

| Use Case Name | Use case 1 - Modtage Bestillinger |
| --- | --- |
| Scope | Pizza application |
| Level | User goal |
| Primary Actor | Mario og alfonso |
| Stakeholders and Interests | Mario: En hurtig og nem interaktion gennem et effektivt system.  Alfonso: En hurtig og nem interaktion gennem et effektivt system.  Kunde: Vil gerne have pizza bestilt hurtigt og effektivt.  Mario: altid have nem adgang til en fuld liste over de pizzaer de tilbyder fuldendt med ingredienser så han ikke laver pizzaer forkert.  Alfonso: Nem oversigt over menuen så han kan være sikker på at sende den rigtige ordre videre til Mario efter en bestilling.  Kunde: Kunne se et menukort for at vælge hvad for en pizza de kunne tænke sig. |
| Preconditions | Kunde ankommer/ringer til Mario’s pizzeria |
| Success Guarantee (postconditions) | Pizza bliver gemt i systemet. Menu kan ses. |
| Main Success Scenario | 1. Alfonso starter ny bestilling 2. Kunden oplyser ønsket pizza 3. Alfonso tilføjer pizza til ordre   *Gentag step 3-4 indtil kunden er færdig*   1. Alfonso spørger om navn 2. Alfonso spørger om hvornår bestilling skal være klar 3. Kunden oplyser om afhentningstidspunkt 4. Alfonso validere bestilling med kunden 5. kunden godkender bestilling 6. Alfonso sender bestilling videre i et sorteret system |
| Extensions | 3a Kunden ønsker ekstra stuffing  på pizza  1. Alfonso tilføjer ekstra stuffing på  pizza  3b At any time, når alfonso/kunden ønsker at se menukort.   1. Menukortet åbnes 2. return   3c. Kunde ønsker at lave egen pizza.   1. Alfonso indtaster kundens ønskede toppings til en “blank” pizza.   7a kunde oplyser tidspunkt for afhentning der ikke stemmer overens med hvornår pizza kan være klar.  1. - tilbage til step 7. |
| Special Requirements | Alfonso skal nemt kunne tage imod ordre og indtaste dem i et system som gør det både nemt og overskueligt at holde styr på hvilke pizzaer skal laves. |
| Technology and Data Variations list | Ikke relevant for denne Use Case siden der ikke er nogle eksterne enheder som applikationen skal arbejde sammen med |
| Frequency of occurrence | hver gang en kunde laver en bestilling |
| Miscellaneous | - |

**Use case 2:**   
I denne use case bliver der beskrevet hvordan vores bestillinger bliver afsluttet når en pizza er færdig og ekspideret.

| Use Case Name | Use case 2 - Afslut Bestillinger |
| --- | --- |
| Scope | Pizza application |
| Level | User goal |
| Primary Actor | Mario og Alfonso |
| Stakeholders and Interests | Mario: En hurtig og nem interaktion gennem et effektivt system.  Alfonso: Oplysninger når en bestilling er færdig |
| Preconditions | * Når en ordre er blevet oprettet. * Mario har lavet bestillingen. * Mario har oplyst Alfonso at en bestilling er færdig. |
| Success Guarantee (postconditions) | At en ordre der er færdig, ikke længere er på listen over ventende bestillinger. |
| Main Success Scenario | 1. Alfonso checker listen for afventende bestillinger 2. Alfonso validere bestilling fra listen med bestilling numre. 3. Pizzaen der er ekspederet bliver fjernet fra listen over aktive bestillinger. 4. Den bliver samtidig gemt i en liste over alle solgte pizzaer. |
| Extensions | 1a Mario glemmer en pizza på  bestillingen  1. Alfonso gør Mario  opmærksom på at en  pizza mangler  2. Mario laver pizza  3. Mario sender manglende  manglende pizza til Alfonso  *Tilbage til step 4* |
| Special Requirements | - |
| Technology and Data Variations list | Dell-pc and card reader |
| Frequency of occurrence | Hver gang en bestilling skal afsluttes |
| Miscellaneous | - |

**Use case 3:**  
Denne use case beskriver hvordan Mario kan tilgå et arkiv og lave statistiske udregner på sit salg fra en given periode.

| Use Case Name | Use case 3 - Se Statistik |
| --- | --- |
| Scope | Pizza application |
| Level | Sub Function |
| Primary Actor | Mario |
| Stakeholders and Interests | Mario: Har en interesse i at se alle de pizza som har været ekspederet indenfor et givent tidspunkt og kan hente statistisk ud fra disse. |
| Preconditions | At en ordre er blevet afsluttet. |
| Success Guarantee (postconditions) | At de afsluttede pizzaer bliver arkiveret og trukket ud til statistisk behandling. |
| Main Success Scenario | 1. Mario tilgår arkivet og kigger listen igennem. 2. Mario vælger en ønsket periode han ønsker at se salg fra. 3. Han klikker på knappen “statistik” som viser ham omsætningen, antal salg og hvor mange af hver pizzaer der er solgt. |
| Extensions | - |
| Special Requirements | - |
| Technology and Data Variations list | Database |
| Frequency of occurrence | Når mario ønsker at tilgå information omkring hans salg.  Når en bestilling er færdig |
| Miscellaneous | - |

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

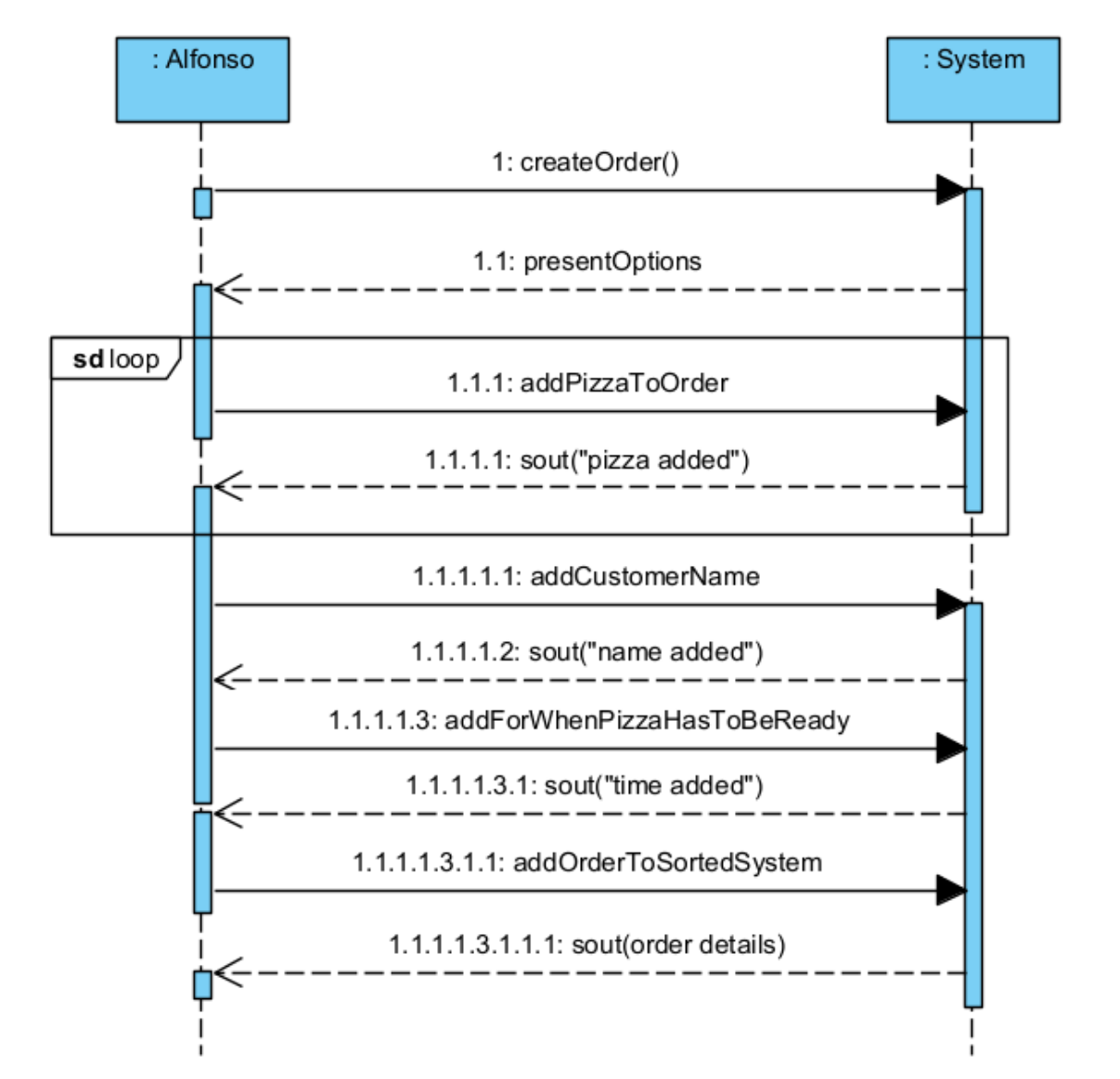
## 

## 

## System Sequence Diagram

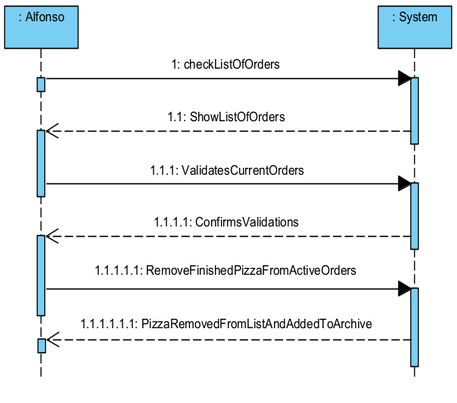
**SSD UC01: Modtage Bestilling.**

**Første iterering:**



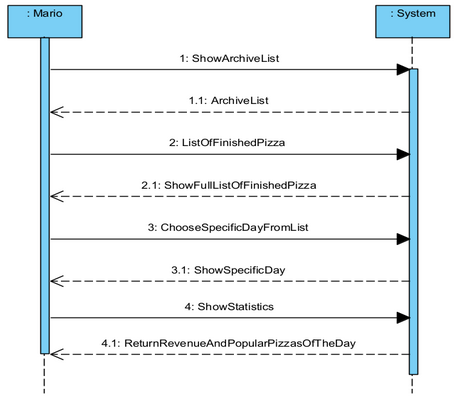
**2. iterering:**  
Se figur 01 under Optional kapitlet.  
*Første iteration var lavet før/under kode processen. Anden iteration er lavet efter kode processen for at give en sammenligning.*

**SSD UC02: Afslut Bestilling.  
Første iterering:**

****

**2. iterering:**Se figur 02 under Optional kapitlet.  
*Første iteration var lavet før/under kode processen. Anden iteration er lavet efter kode processen for at give en sammenligning.*

**SSD UC03: Se Statistik.**

****

# 

# 

# 

# Operation Contracts

# **Operation contract for use case 1:**

| Operation | addPizzaToOrder |
| --- | --- |
| Cross References | Use Case : Modtage Bestillinger. |
| Preconditions | Kunde vil have en pizza.  Bestillings objekt er blevet oprettet. |
| Postconditions | * Pizza objekt bliver instansieret (instance creation). * Pizza objekt bliver lagt til en liste over pizza objekter i den oprettede bestilling (attribute modification). * Association formed med Bestilling klassen |

‘

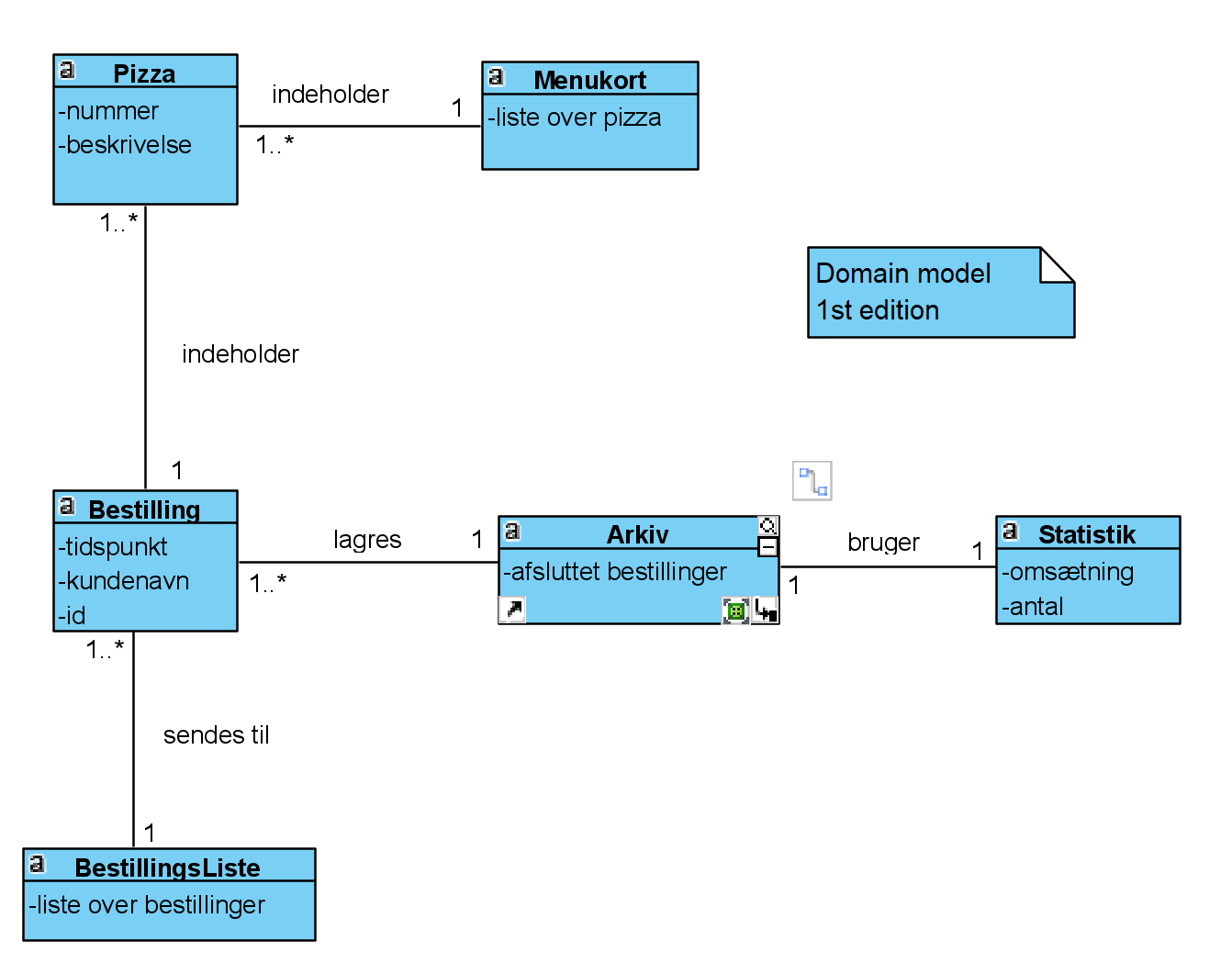
**Operation contract for use case 2:**

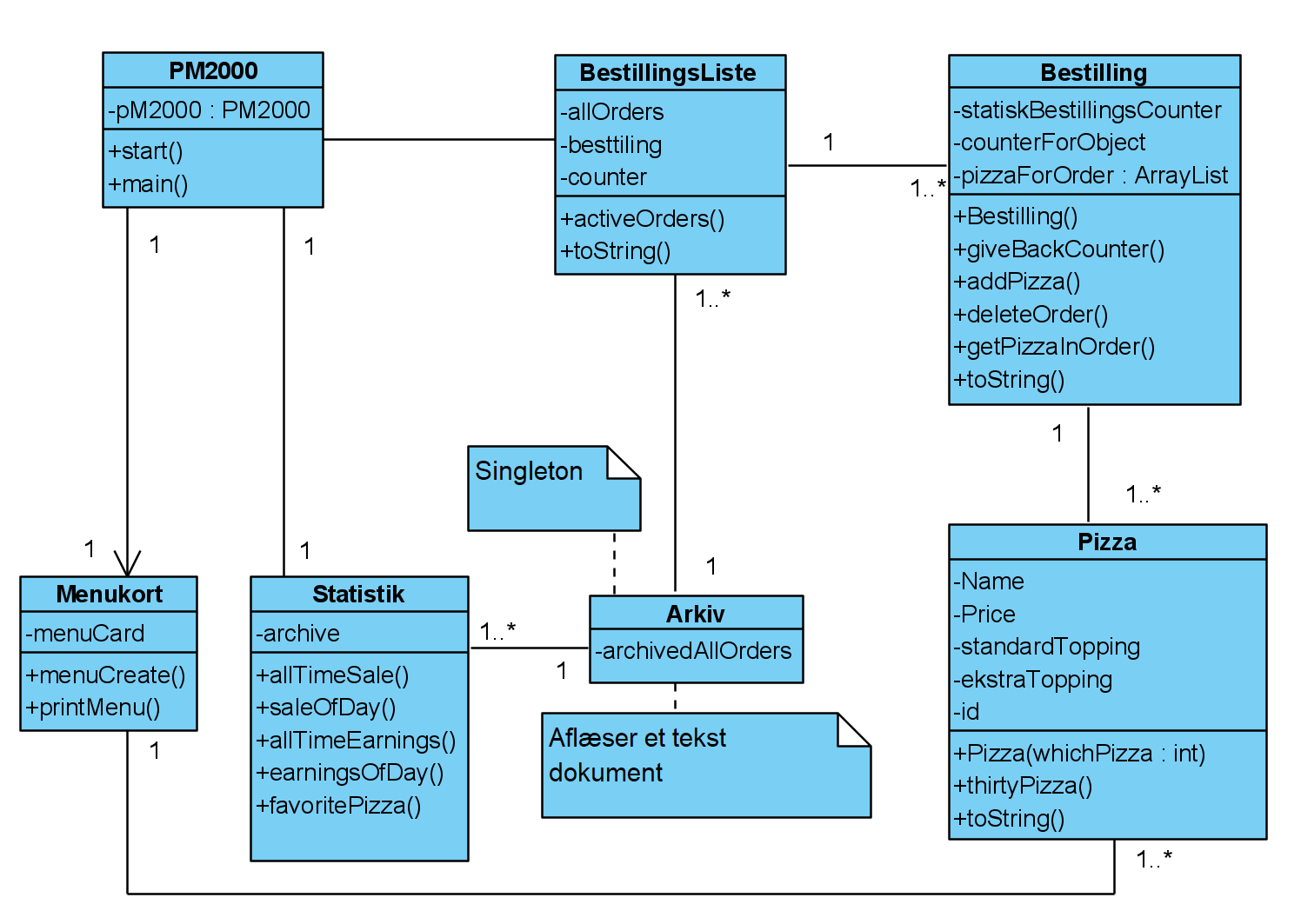
| Operation | removeFinishedPizzaFromActiveOrder |
| --- | --- |
| Cross References | Use Case : Afslut Bestilling. |
| Preconditions | En bestilling eksisterer. |
| Postconditions | * Fjerner forholdet mellem Bestilling Objektet og Bestillingslisten (Association broken). * Bestillings informationer bliver lagt til Statistik (Association formed). |

**Operation contract for use case 3:**

| Operation | showStatistic |
| --- | --- |
| Cross References | Use Case : Se Statistik |
| Preconditions | En bestilling er afsluttet |
| Postconditions | * Fremviser salg af alle bestillinger * Fremviser mest populærer pizza |

# Design Class Diagram

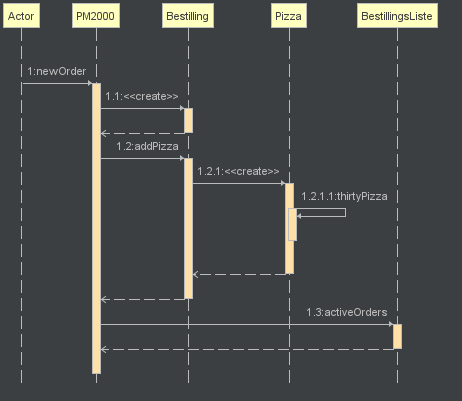
**Domain model:**  


**Class diagram:  
Første iteration(som vi brugte til at give overblik i den kode vi havde skrevet ud fra domain model):**

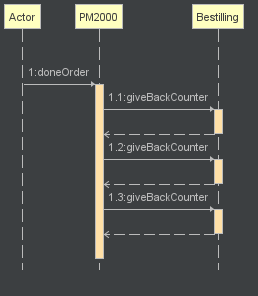
**2. iterering:**Se figur 03 i “Options”.  
*Første iteration var lavet før/under kode processen. Anden iteration er lavet efter kode processen for at give en sammenligning.*

# Sequence Diagram

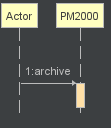
**Sekvensdiagram 1:**

****

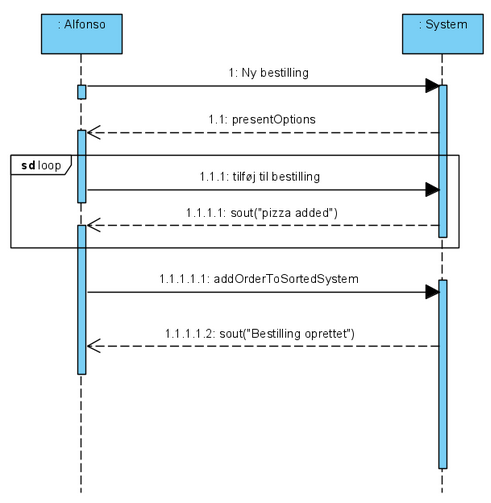
**Sekvensdiagram 2:**

****

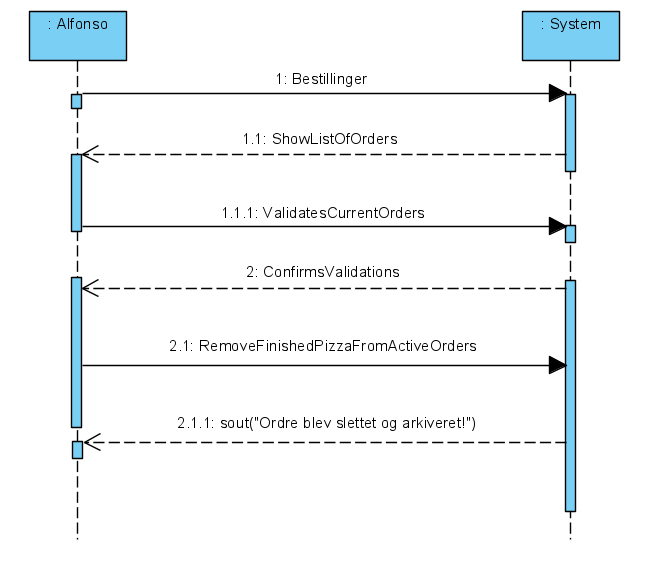
**Sekvensdiagram 3:**

****

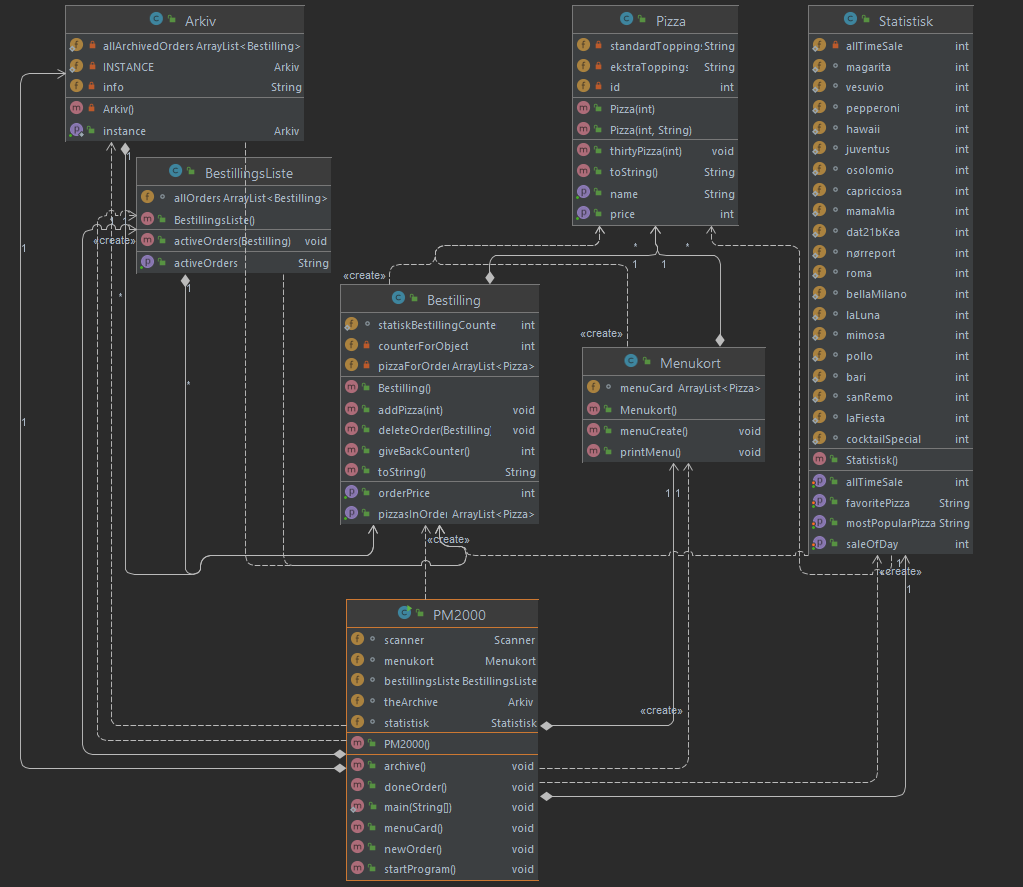
# “Optional” – Other Information? **Figur 01 (SSD til UC01):**



**Figur 02 (SSD til UC02):**

****

**Figur 03 Class diagram**

****

# Glossary

| **Glossary:** | **Explanation:** |
| --- | --- |
| Mario | Primary Actor |
| Alfonso | Primary Actor |
| Customer | Offline |
| PM2000() | ***(Pizza-Machine 2000)*** App runtime environment |
| Arkiv() | gemmer data i memory over hvad der er solgt |
| statistik() | printer den populære pizza |
| Pizza() | Liste over alle pizza |
| Menukort() | Printer liste over alle pizza |
| Bestilling() | Opretter bestillinger |