

Projekt Rapport

*Marios Pizza*

***DAT21b***

**Project Participants:**

*Frederik T.C*

*Johannes S.*

*Naomi M.R.*

*Valdemar K.J.*

*Document Version: 00.02.00*

*Document Template version: 00.03.01*

# 

[**Vision**](#_dn7k51dchwxd) **3**

[**Requirements**](#_2eqi6e6s4em4) **3**

[Functional Requirements](#_2s8eyo1) 3

[Non-Functional requirements](#_h317hw7pte3j) 3

[**Use Cases**](#_wapi245u83lz) **4**

[**Operation Contracts**](#_o0v1odywwncs) **6**

[**Diagrammer**](#_8qo7s4iyq4ey) **6**

[Use Case Diagram](#_qxkv3dbqne) 6

[Class Diagram](#_d0l88xf995a3) 7

[Domain Model](#_xscw8fy79tyo) 7

[SSD - nyBestilling](#_7xgyy88mmyav) 8

[SSD - afslutBestilling](#_u26olm7qvbx2) 9

[SSD - seBestillinger](#_nk3bk163b4b0) 9

[SSD - anullerBestilling](#_1ev17mk3ipdj) 10

[SD - nyBestilling](#_5r8gmmbuhtdo) 10

[SD - printMenukort](#_6at4gfn2sz7j) 11

[SD - seBestillinger](#_iqs2ghfbxtly) 11

[SD - afslutBestilling](#_yzd2nnpgtqn6) 12

[SD - anullerBestilling](#_15u4998fjnf3) 12

[**Glossary**](#_35nkun2) **12**

# 

# 

# 

# 

# 

# 

# Vision

IT-projektet der skal udvikles for Marios Pizza skal kunne håndtere bestillinger. Programmet skal kunne vise restaurantens menukort. Ydermere skal det kunne oprette og opbevare en liste over bestillinger samt deres afhentningstidspunkt. Det skal kunne køre offline, på ældre hardware.

# Requirements

## Functional Requirements

| **Id** | **Requirement** | **Comment** |
| --- | --- | --- |
| REQ004 | Fjerne den fra listen, når den er afhentet og betalt. | Slettefunktion men ordren skal stadig gemmes. |
| REQ010 | Gemme alle ordrene | Efter ordre er færdig skal man kunne se alle de gamle ordre der har været. |
| REQ002 | Hvornår de skal afhentes | dag+tidspunkt |
| REQ001 | Liste med bestillinger | Liste sorteret efter afhentningstidspunkt. |
| REQ005 | Se omsætningen | total omsætning pr.dag + uge + måned måske? |
|  |  |  |

### 

## Non-Functional requirements

| **Id** | **Requirement** | **Comment** |
| --- | --- | --- |
| REQ011 | Blive til en online shop til bestillinger | På sigt skal der gerne være mulighed for at udvide platformen til en online shop hvor der kan indgå bestillinger |
| REQ008 | I java simpelt |  |
| REQ006 | Lave statistik | Statistik over hvilke pizzaer der bliver købt flest af. |
| REQ009 | Sortere efter tidspunkt | Pizzaerne bliver sorteret efter hvornår de skal afhentes. |
| REQ003 | Se menukortet på skærmen | For at Alfonso har et bedre overblik når kunder ringer for at bestille. |
| REQ007 | Skal kunne køre på en gammel computer | Skal kunne køre på en ældre computer uden internetforbindelse |

# 

# 

# Use Cases

**Fully Dressed Use Cases**

**1.**

| **Use Case Section** | **Comment** |
| --- | --- |
| Use Case Name | nyBestilling |
| Scope | Man skal kunne gå ind i systemet og tilføj pizza til bestilling + afhentningstidspunkt. |
| Level | Målet er at Mario skal kunne se hvilken pizza der skal laves først dvs. liste over pizzaer + afhentningstidspunkt. |
| Primary Actor | Alfonso modtager ordre -> skriver ind i liste over bestillinger. |
| Stakeholders and Interests | Marios/Alfonso interesse er at bestilling bliver lagret, det samme for kunde. |
| Preconditions | Systemet skal køre før man kan lave en bestilling. |
| Success Guarantee | Bestillingen er blevet korrekt registreret i systemet. |
| Main Success Scenario | 1. Alfonso vælger ny bestilling 2. Alfonso vælger pizza 3. Alfonso oplyser bestillingsinfo og afslutter bestilling 4. Systemet sætter ordren ind på en liste, alt efter hvornår pizzaen skal være færdig |
| Extensions | 1. Alfonso taster forkert og skriver noget programmet ikke kan forstå.    1. programmet kommer med en error-message. 2. Alfonso taster forkert pizza/tidspunkt ind, eller kunden ombestemmer sig.    1. Alfonso skal kunne ændre pizzaens status til annulleret |
| Special Requirements | Skal kunne se menukortet på skærmen - req003 |
| Technology and Data Variations List | Skal kunne køre på gammel hardware |
| Frequency of Occurrence | Skal køre ved hver ordre. |
| Miscellaneous |  |

2.

| **Use Case Section** | **Comment** |
| --- | --- |
| Use Case Name | afslutBestilling |
| Scope | Bestilling skal kunne ændre status, efter udlevering af pizza. |
| Level | Alfonso ændre status på bestillingen til færdig og afhentet. |
| Primary Actor | Alfonso |
| Stakeholders and Interests | Mario og Alfonso - har interesse i at pizzaen ikke længere er på står som en nyoprettet ordre, når pizzaen er blevet udleveret. |
| Preconditions | Der er lavet en bestilling, dvs. Use-Case nr.1 er opfyldt. |
| Success Guarantee | Bestillingen har korrekt ændret status til færdig eller afhentet. |
| Main Success Scenario | 1. Kunden afhenter pizza 2. Alfonso ændre pizzaens status til afhentet |
| Extensions | 1. Kunden annullerer ordre 2. Alfonso ændre pizzaens status til annulleret |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | Skal kunne køre efter der er lavet en bestilling. |
| Miscellaneous |  |

## 

3.

| **Use Case Section** | **Comment** |
| --- | --- |
| Use Case Name | seBestillinger |
| Scope | Man skal kunne se bestillingsliste. |
| Level | Alfonso og Mario skal kunne fremkalde en liste over bestillinger på skærmen. |
| Primary Actor | Alfonso |
| Stakeholders and Interests | Mario |
| Preconditions | Der er blevet bestilt en pizza med afhentningstidspunkt og den er kommet på listen. |
| Success Guarantee | En liste over bestillinger fremkaldes når programmet bliver bedt om det. |
| Main Success Scenario | 1. Bestilling kan ses på bestillingslisten |
| Extensions |  |
| Special Requirements | Skal kunne sortere efter tidspunkt - req009 |
| Technology and Data Variations List | Skal kunne køre på gammel hardware |
| Frequency of Occurrence | Skal kunne køre hver gang den bliver kaldt |
| Miscellaneous |  |

4.

| **Use Case Section** | **Comment** |
| --- | --- |
| Use Case Name | annullerBestilling |
| Scope | Bestilling skal kunne ændre status på listen efter ønske. |
| Level | Alfonso skal kunne ændre ordrens status på listen til annulleret |
| Primary Actor | Alfonso |
| Stakeholders and Interests | Mario og Alfonso - har interesse i at pizzaen ændre status på listen, f.eks. fordi kunder har annulleret bestillingen, eller Alfonso har skrevet en forkert bestilling ind. |
| Preconditions | Der er lavet en bestilling, dvs. Use-Case nr.1 er opfyldt. |
| Success Guarantee | Bestillingens status er korrekt opdateret på listen |
| Main Success Scenario | 1. Alfonso ændre pizzaens satus |
| Extensions |  |
| Special Requirements |  |
| Technology and Data Variations List | Skal kunne køre på gammel hardware |
| Frequency of Occurrence | Skal kunne køre hver gang en pizza skal anulleres |
| Miscellaneous |  |

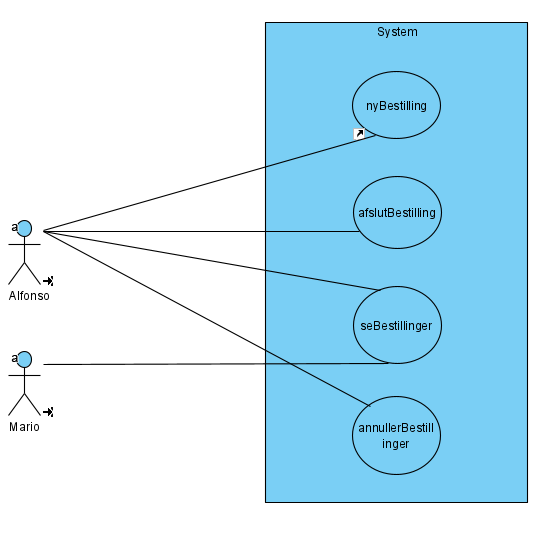
# Operation Contracts

| Operation | nyBestilling(ArrayList<> bestillingsListe, ) |
| --- | --- |
| Cross References | lav Bestilling |
| Preconditions | En bestilling bliver oprettet. |
| Postconditions | * Et bestillings objekt bliver oprettet. * Der bliver skabet en association mellem bestilling og pizza. * Der bliver oprettet et objekt af pizza. * Et objekt af pizza tilføjes til bestilling. * Der bliver skabt en association mellem bestilling og storage. * Bestillings objekt bliver tilføjet til bestillingslisten i storage. |

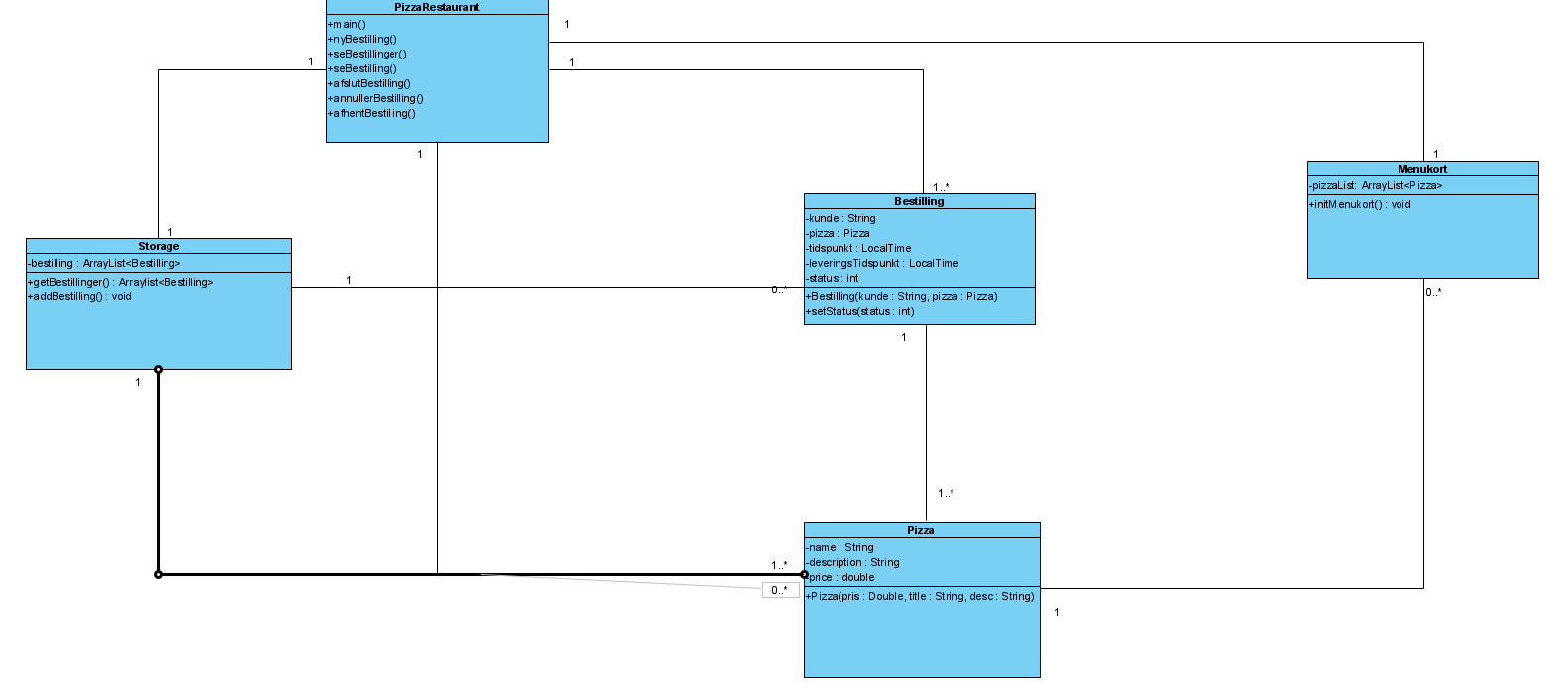
# 

# Diagrammer

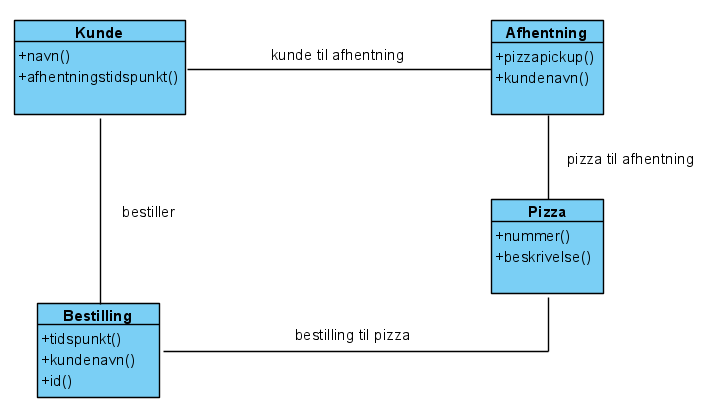
## Use Case Diagram

****

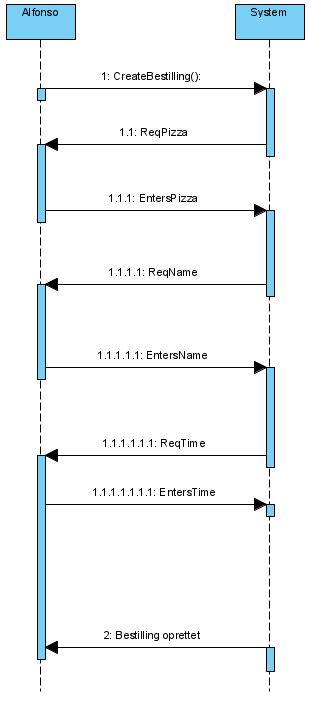
## Class Diagram

****

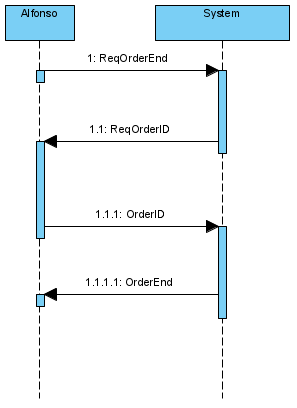
## Domain Model

****

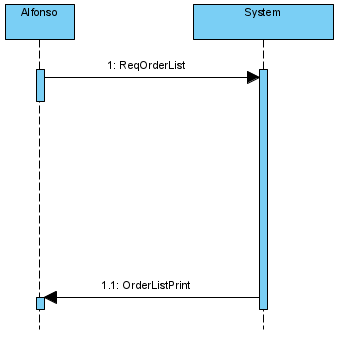
## SSD - nyBestilling

****

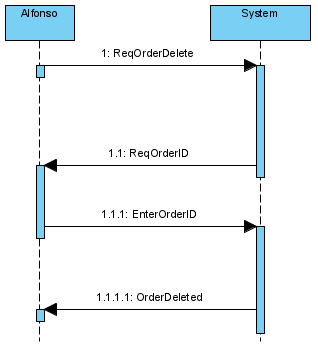
## SSD - afslutBestilling

****

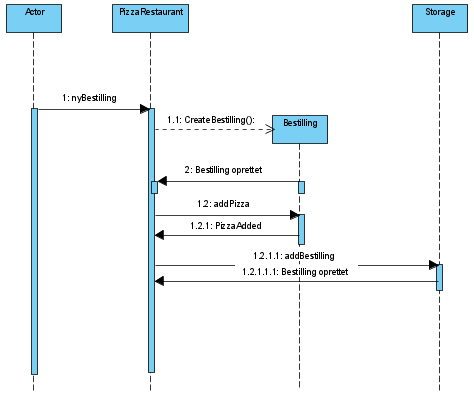
## SSD - seBestillinger

****

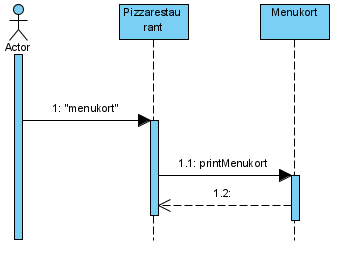
## SSD - anullerBestilling

****

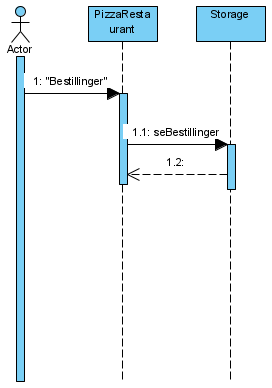
## SD - nyBestilling

****

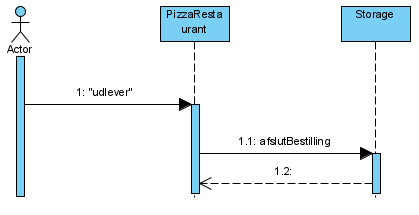
## SD - printMenukort

****

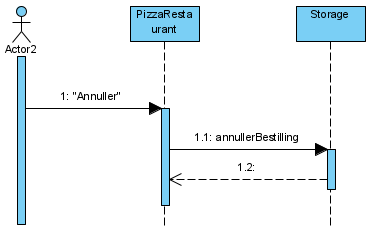
## SD - seBestillinger

****

## SD - afslutBestilling

****

## SD - anullerBestilling

****

# Glossary

Bestillingsinfo - Hvilke pizzaer der skal laves, kundens navn, og hvornår pizzaen skal afhentes.

**Kode-glossary:**

Storage (klasse) - Hvor ArrayLists med Bestillinger opbevares.

pizzaID - pizzaens indexnummer i Arraylisten af bestillinger.