

OOAD Project Documentation

*Delfinen*

**DAT17c**

**Project Participants:**

*Martin Jensen*

*Rasmus*

*Christian*

*Casper*

Document Version: 00.01.00

Contents

[History 3](#_Toc494920576)

[Vision 4](#_Toc494920577)

[“Optional” - Additional informative Artifacts before Requirements? 5](#_Toc494920578)

[Requirements 6](#_Toc494920579)

[Use Cases 6](#_Toc494920580)

[Supplementary Specification 7](#_Toc494920581)

[FURPS+ 7](#_Toc494920582)

[Functional Requirements 8](#_Toc494920583)

[Non-functional requirements 8](#_Toc494920584)

[“Optional” – Other Information? 9](#_Toc494920585)

[Appendix 10](#_Toc494920586)

[Glossary 10](#_Toc494920587)

{ TO BE REMOVED…

**Remember** to update Contents.

}

# History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Author** | **Description** |
| 201x-xx-xx | 00.01.00 |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

{ TO BE REMOVED…

* Who has last updated and for what reason?
* What has changed?

}

# Vision

{ TO BE REMOVED…

**The GOAL!**It defines the stakeholders view of the product/service to be developed, specified in terms of the stakeholders key needs and features. Containing an outline of the envisioned core requirements.

**Summary of System Features**

Definition  
Features are behavioral functions a system can do. They should pass this linguistic test:  
The system does <feature X>.

For example:  
The system does payment authorization.

The major features include:  
• POS services  
• Inventory management  
• Web-based shopping  
• ...

**Other Requirements in the Vision?**In the Vision, system features briefly summarize functional requirements often detailed in the use cases. Likewise, the Vision can summarize other requirements (for example, reliability and usability) that are detailed in the Supplementary Specification. But be careful to avoid going down the path of repeating yourself.

**See Larman for more information:**  
Chapter 7.6. NextGen Example: (Partial) Vision  
Chapter 7.7. Commentary: Vision

}

# “Optional” - Additional informative Artifacts before Requirements?

{ TO BE REMOVED…

Sometimes it is good to have one or more short/small informative texts or illustrations depending on nature of the project and/or the style of the document.

**Possible Artifacts to include:**

* Domain Model
* System Sequence Diagram
* Sequence Diagram
* Activity Diagram
* Class Diagram
* Glossary
* …

}

# Requirements

{ TO BE REMOVED…

Extracted from meeting, interview and/or other gathered information for the new system

* Req1…
* Req2…
* Req3…
* …

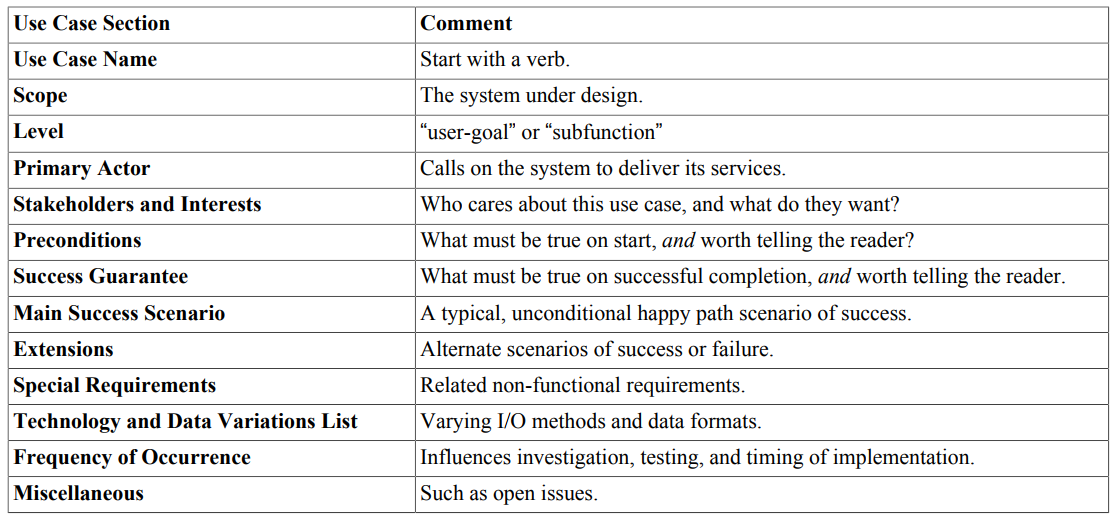
]

## Use Cases

{ TO BE REMOVED…

* Short Use Cases text/description
* Use Case Diagram(s)
* Use Case 1 text – Brief/Casual/Fully Dressed
* Use Case 2 text – Brief/Casual/Fully Dressed
* Use Case 3 text – Brief/Casual/Fully Dressed
* …

**Remember to add**  
“: INCLUDE *UCxx IncludedUseCaseName*.” in main use case step where INCLUDE is used.

Supporting **Use Case sections** template created by Alistair Cockburn  


**See Larman for more information:**  
Chapter 6. Use Cases  
!!! 6.9. What do the Sections Mean?

}

## Supplementary Specification

{ TO BE REMOVED…

**Use Either FURPS+ or Functional/Non-Functional**basically, everything not in the use cases. This artifact is primarily for all non-functional requirements, such as performance or licensing. It is also the place to record functional features not expressed (or expressible) as use cases; for example, a report generation.

**See Larman for more information:**  
Chapter 5.4. What are the Types and Categories of Requirements?  
Chapter 5.5. How are Requirements Organized in UP Artifacts?

}

### FURPS+

Functional

( To be removed — features, capabilities, security. )

Usability

( To be removed — human factors, help, documentation. )

Reliability

( To be removed — frequency of failure, recoverability, predictability. )

Performance

( To be removed — response times, throughput, accuracy, availability, resource usage. )

Supportability

( To be removed — adaptability, maintainability, internationalization, configurability. )

+  
Implementation

( To be removed — resource limitations, languages and tools, hardware, ... )

Interface

( To be removed — constraints imposed by interfacing with external systems. )

Operations

( To be removed — system management in its operational setting. )

Packaging

( To be removed — for example, a physical box. )

Legal

( To be removed — licensing and so forth. )

### Functional Requirements

|  |  |  |
| --- | --- | --- |
| **Id** | **Requirement** | **Comment** |
| FR001 | … |  |
| FR002 | … |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Non-functional requirements

|  |  |  |
| --- | --- | --- |
| **Id** | **Requirement** | **Comment** |
| NFR001 | … |  |
| NFR002 | … |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# “Optional” – Other Information?

{ TO BE REMOVED…

Other informative text/diagram to include?

}

# Appendix

{ TO BE REMOVED…

“Optional” - Additional informative Artifacts after Requirements?

* Domain Model
* System Sequence Diagram
* Sequence Diagram
* Activity Diagram
* Class Diagram
* …

}

Glossary

{ TO BE REMOVED…

**Use either 1 Glossary with 2 section or 2 Glossaries:**  
One for Domain/Business  
One for Technical/Data dictionary

**See Larman for more information:**Chapter 7.8. NextGen Example: A (Partial) Glossary for more information  
Chapter 7.9. Commentary: Glossary (Data Dictionary)

**}**