HTL LEONDING

System Specification

Name of the Project

Project Name	Template
Project Leader	P. Bauer
Document state	In process
Version	V. 1.0

Revisions

Date	Author	Change
November 03, 2019	P. Bauer	First version

Inhaltsverzeichnis

1 Initial Situation and Goal

This section describes the initial situation and the motivation to start this project. You describe the problem of the existing system or approach and make clear what led to the decision to carry out this project and point out the benefits of the new system. Furthermore, list all relevant project stake holders and draw a picture how your system will fit into the framework of existing systems and environments. Finally a first draft of general requirements like technical constraints or security constraints has to be given.

1.1 Initial Situation

In this subsection you describe the scope of application of your system. Describe the current state of how the current system works. Give current work flows and introduce and explain domain specific terms. Revisit the section "Initial Situation" from your project proposal and see what you would refine under your now improved understanding of the project

1.1.1 Application Domain

Describe the application domain of your system as well as the environment in which it is embedded. If there exists an environmental business process you should describe it here. Focus on the interrelationship of terms, business processes etc. (... not sure whether I understand what they really mean here...)
Your goal in this section is to introduce the domain-specific terminology and the context in which your system lives. Introduce in a way such that an interested layman can follow your text, establish a clear structure of your text, use illustrations and don't assume a detailed domain specific knowledge.
Take care to clearly separate assumptions from given hard facts. This enables you to trace down your requirements in case of (customer) complaints in a later project stage.

1.2 Goal

Describe in detail what your project wants to achieve. Give clear answers how the shortcomings mentioned in the Initial Situation will be solved by the project.

2 Functional Requirements

Functional requirements describe the features of a system which are expected by a user in order to solve a specific problem. The requirements are derived from the business processes and work flows which are supported by the system.
The description of functional requirements is accomplished by means of use cases. A use case describes a concrete and self-contained process. The sum of all use cases describes the system behaviour. Describe use cases in plain text and support it by provide clear and illustrative use case diagrams. In case of a very data-oriented application provide a first version of a data model (business domain model). This model is the basis for the data base design in a later project stage. The data model is derived from the entities of the domain model.

2.1 Overview

In this subsection you give an overview of all use cases. Starting with a UML use case diagram you list all use cases covered in your system and a short description of each of the use cases.

2.2 Use Case 1: <Name>

This subsection is copied for each use case you listed in the previous section.

2.2.1 General Description

Give a general description of where a tabular form might help:

ID:	Identifier of the use case
Goal:	The goal of the use case
Precondition:	Under which condition is the user case triggered?
Postcondition:	What conditions are true after the use case was successfully executed?
Involved Users:	Role name: Description of users interacting with the system. “Users” can be other systems, too.

2.2.2 UI to call the use case

Give a sketch of the UI and describe the UX controls.

2.2.3 The Standard Use

Describe the happy path.

- *UI*
- *Description*
- *Activity, Sequence, or State Diagram to visualize the workflow*

2.2.4 The Non-Standard Use

Describe the corner cases, possible errors and how the system reacts on them

- *UI*
- *Description*
- *Activity, Sequence, or State Diagram to visualize the workflow*

3 Non-Functional Requirements

Non-functional requirements describe all aspects of a system that cannot be mapped to a specific feature. Nevertheless these requirements are essential for the system itself. Non-functional requirements are, e.g., quality requirements, security requirements, or performance requirements.
Non-functional requirements define basic features of a system which also have an impact on the architecture. They also influence the development costs and, therefore should be formulated in a measurable way. WRONG: The system shall be fast responding. CORRECT: The response time shall be within 500 ms.
The following section has to be copied for each non-functional requirement.

3.1 NFR 1: <Name>

ID:	Identifier of the NFR
Name:	The name of the NFR
Type :	Type as described below
Descritpion:	

The type of the NFR shall be taken from one of the following: The table below shall finally be deleted.

MAINT	Maintenance and portability requirement	Which maintenance or porting effort is expected in the future? Internationalization expected? Porting to different hardware platform?...
SEC	Security requirement	Security requirements comprise confidentiality, data integrity, and availability. How much do we have to consider that data is not accessible to unauthorized persons? Is the correctness and/or consistency of data to be guaranteed? How severe are total system faults?
LEGAL	Legal requirement	Are there any standards or legal constraints to be considered?
USE	Usability Requirement	Usability covers all aspects to make the targeted user like to work with the software.
EFFIC	Efficiency Requirement	Runtime and/or memory efficiency of the program.

4 Quantity Structure

Describe the number of expected records in master data as well as business cases. This assessment is basis to make proper decisions concerning the form of data persistence (e.g., XML or data base) and data base product. Furthermore the quantity structure gives you a better idea about special requirements (e.g. the GUI) for your system because of hight quantity data.

5 System Architecture and Interfaces

To illustrate how your system is embedded in it's environment list all interfaces to surrounding systems. Interfaces to users, supporting systems, logistics, peer-systems are to be listed and described.

6 Acceptance Criteria

The acceptance criteria define which criteria the system has to fulfill in order to be accepted. Describe, what has to be checked such that the system can be accepted. Give at least one acceptance test for each functional requirement described in this document. For each acceptance criterion one subsection has to be written.