Génie Logiciel
Writing documentation

Sylvain Lobry

10/12/2021

Resources: www.sylvainlobry.com/GenieLogiciel

Menu of the day

Documentation

Introduction

- A good software must come with a good documentation
- **Definition**: the set of written documents and material (e.g. figures, diagrams) that accompanies computer software
- Essential part of software engineering

- Main purpose of the documentation: **sustainable communication** with:
  - Stakeholders
  - Development teams
  - Users

Documentation

Types of documents – Contractual

- **Specification (cahier des charges)**: describe the requirements and constraints of the project
- **Contract**
- **Convention**: contractual document following an agreed upon form
- **Reception**: formalization of the end of a project

Documentation
Types of documents – Support documents

- **Presentation of the software**: general presentation of the software, without details
- **User manual**: describes the functionalities of the software and how to use them
- **Installation manual**: describes the procedures to install the manual
- **Exploitation manual**: troubleshooting, maintenance and updates procedures

Documentation

Types of documents – Project management

- **Development plan**: defines the procedures to achieve the project
- **Conception**: describes the outcome of the design phase.
- **Project's reports**: intermediary reports to document the development process
- **Feedback**: towards the end of a project, what did we learn? What can be re-used?

Documentation

Types of documents – Quality

- **Quality assurance plan**: if followed, ensures that the finished product meets all pre-defined quality criteria
- **Internal audit plan**: defines the frequency and procedures for internal audits.
- **Evaluation plan**: at the end of each phase, verify that the initial objectives are met
- **Test plan**: at the end of the project, validate that the initial objectives are met

Menu of the day

Documentation

Writing a document – General structure

- **Identification of the document:**
  - Name of the document
  - Name of the project
  - Name of the authors / institutions
  - Date
  - Version of the document

- **Quality of the document:**
  - Information on verification
  - Information on validation
  - Information on the confidentiality
  - Keywords

Documentation

Writing a document – General structure

- **Outline**
- **List of figures**
- **List of tables**

Documentation

Writing a document – General structure

- **Introduction:**
  - Context
  - Objectives of the document
  - Summary
  - Reminder on the necessary concepts
  - How to read this document: reading guide for each category of reader

- **The specific content of the document**

Documentation

Writing a document – General structure

- **Glossary**
- **References**
- **Index**

Menu of the day

Documentation

Cahier des charges (specification)

- **Definition:** Le cahier des charges est un document rassemblant les obligations et les éléments nécessaires pour définir un besoin et les principales contraintes à respecter pour le satisfaire. (AFITEP)
- **Objective:** allow the client to define its needs and choose the appropriate party to conduct the project

- **European norm:** NF EN 16271 (European norm), 02/2013
- Previously: AFNOR X50-151

Documentation

Cahier des charges – AFNOR X50-151

- Présentation générale du problème:
  - Projet (finalités, volet financier)
  - Contexte:
    - Positionnement par rapport aux autres projets de l'entreprise
    - Etudes préalables
    - Suites prévues
    - Prestations demandées
    - Parties concernées par le déroulement du projet et ses résultats (demandeurs, utilisateurs)
    - Caractère confidentiel du projet
  - Enoncé du besoin (quel finalité pour l'utilisateur final)
  - Environnement du produit:
    - Liste des éléments de l'environnement du produit (personnes, équipement, matières) et des contraintes
    - Caractéristiques pour chaque élément de l'environnement

Documentation

Cahier des charges – AFNOR X50-151

- Expression fonctionnelle du besoin:
  - Fonctions de service et de contrainte
    - Fonctions de service principales
    - Fonctions de service complémentaires
    - Contraintes
  - Critères d'appréciation
  - Niveaux des critères d'apprécpiation

Documentation

Cahier des charges – AFNOR X50-151

- Cadre de réponse
  - Pour chaque fonction:
    - Solution proposée
    - Niveau atteint pour chaque critère d'appréciation de cette fonction et modalités de contrôle
    - Part du prix attribué à chaque fonction
  - Pour l'ensemble du produit:
    - Prix de la réalisation de la version de base
    - Options et variantes proposées non retenues au cahier des charges
    - Mesures prises pour respecter les contraintes et leurs conséquences économiques
    - Outils d'installation, de maintenance à prévoir
    - Décomposition en modules
    - Prévisions de fiabilité
    - Perspectives d'évolution technologique

Menu of the day

Documentation
Development plan

- **Definition:** document that presents the development strategy
- **Objective**: provide a reference on the planification and the organization of the software development

Documentation

Development plan - sections

- Introduction:
  - Objectives of the development
  - Methodology
  - Related documents
- Organization:
  - Description of the tasks
  - Description of the personnel
- Planification:
  - Software development life cycle
  - Timeline
- Methods and technological choices
- Documentation:
  - Which document will support the development
  - Which standard will be followed
  - Which management tools will be used
- Quality: which quality standard will be used, how will you evaluate the quality?

Menu of the day

Documentation

Conception

- **Definition:** document that presents the outcomes of the design phase, with a complete and detailed description of the software's components.
- **Objective**: reference for developpers and the management.

Documentation

Conception - sections

- Introduction:
  - Objectives of the development
  - Methodology
  - Related documents
- Interfaces between the software's tasks
- Detailed description:
  - For each module:
    - objectives
    - relation with other modules
    - types definition
    - description of the functions and variables that are visible to users of the module
  - For each class:
    - Complete description of the class
    - Relations to other classes
    - Attributes and methods

Menu of the day

Documentation

Test plan

- **Definition:** describes the complete testing procedure allowing to verify the software in its entirety and each component individually
- **Objective**: to describe the **unit tests** (to test each component) and **integration tests** (to test the complete software) **before** the implementation phase

Documentation

Test plan - sections

- Introduction:
  - Objectives of the development
  - Methodology
  - Related documents
- Description of the unit tests: for each class/module, for each function/method:
  - Description of the test: how to run, what is covered
  - for each test: which data, what result is expected, how to validate
- Description of the integration tests: for each test:
  - Description of the test: from the exeternal point of view
  - Description of the objective: what does it cover?
  - Testing procedure: which data, what result is expected, how to validate
- Scenarios: propose a set of tests to validate a set of functions, and how do they relate to each others

Menu of the day

Documentation

Acceptance plan (cahier de recette)

- **Definition:** describes the different aspects of software delivery, including acceptance tests
- **Definition (acceptance tests):** Formal testing with respect to user needs, requirements, and business processes conducted to determine whether a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether to accept the system.
- **Objective**: contractually define how to validate that the software meets pre-defined needs and requirements.

Documentation

Acceptance plan - sections

- Introduction**:**
  - Objectives of the development
  - Methodology
  - Related documents
- Deliverables description
- Description of the necessary means and tools to validate the software
- Conformity to general specifications
- Conformity to functional specifications: description of scenarios. For each scenarios, provide the description and the testing procedure
- Conformity to interfaces specifications
- Conformity of the documentation

Menu of the day

1   Introduction and types of documents

2   General structure of a document

3   Documents:

                Specification
                Development plan
                Conception
                Test plan
                Acceptance plan
                Installation manual
                User manual

4   General advices and conclusion

Documentation
Installation manual

- **Definition:** the installation manual is a document gathering all the necessary procedures to install the software in its production environment
- **Objective**: allow the system administrator to **install** and **configure** the software on the targeted information system

Documentation

Installation manual - sections

- **Hardware installation:** which hardware needs to be installed, what procedure are necessary for production
- **System configuration:** which settings need to be applied to correctly configure the system
- **Software installation**: what is the procedure to install the software on the system
- **Software configuration**: which settings need to be applied to correctly configure the software
- **Data**: which procedures are necessary to configure the data of the software
- **Other information**: possible confliclts with other parts of the system, maintenance mode

Menu of the day

1   Introduction and types of documents

2   General structure of a document

3   Documents:

                Specification
                Development plan
                Conception
                Test plan
                Acceptance plan
                Installation manual
                User manual

4   General advices and conclusion

Documentation

User manual

- **Definition:** the user manual is a document describing the functionalities of the software and the information on how to use them.
- **Objective**: to allow the end user of the sotware to use all of the functionalities of the sotware

Documentation

User manual - sections

- **Basic operations:** how to use the software to use the different functions provided
- **Troubleshooting**: description of execeptions/error codes, list of possible issues and how to fix them

Menu of the day

Documentation

General advices on how to write documents

- Be precise
- Write short paragraphs
- Use short sentences
- Choose the right format
- Use illustrations when relevant
- Use language checking tools
- Provide references when they are useful
- Use collaborative features of your editor
- Consider LaTeX (with a versioning system)

Documentation

Conclusion

- Do not neglect documentation!
- A good documentation helps:
  - the planification
  - the design
  - the development
  - the delivery
  - the usage
- Pay attention to the format

Génie logiciel

Conclusion on this class and what's next

- We have seen a set of tools to organize the development of a software
- You should apply them in the next semester
- A good developer knows about software engineering

- Nowadays, software engineering methods are sometimes **Agile**

Génie logiciel

Agile software development

- In 2001, 17 engineers published the **Manifesto for Agile Software Development.**
- Core values:
  - Individuals and interactions over processes and tools
  - Working software over comprehensive documentation
  - Customer collaboration over contract negotiation
  - Responding to change over following a plan
- Aims at lightweight methodologies

Génie logiciel

Agile software development

- The manifesto introduces 12 principles:
  - Our highest priority is to satisfy the customer through **early and continuous delivery** of valuable software.
  - **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.
  - **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
  - Business people and developers must work together daily throughout the project.
  - Build projects around motivated individuals. Give them the environment and support they need, and **trust them to get the job done.**
  - The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**.
  - **Working software is the primary measure of progress.**
  - Agile processes promote sustainable development. The sponsors, developers, and users should be able to **maintain a constant pace indefinitely**.
  - **Continuous attention** to technical excellence and good design enhances agility.
  - **Simplicity**--the art of maximizing the amount of work not done--is essential.
  - The best architectures, requirements, and designs emerge from **self-organizing teams**.
  - At regular intervals, **the team reflects** on how to become more effective, then tunes and adjusts its behavior accordingly.

Génie logiciel

Questions?

- On today's lecture?
- On the class?
- On the exam?

Enjoy the break and good luck for the exam!