National Institute of Technology Karnataka, Surathkal

Department of Computer Science and Engineering

CS303 - Software Engineering Lab

Software Development Life Cycle
and
Requirements Engineering Tasks

for the project

An Automated System to Connect Users and Patients based on Service-Oriented Architecture

Submitted to:

Dr. K. Chandrasekaran
Professor, Dept. of CSE
NITK, Surathkal

Submitted by:

Manas Trivedi
181CO231
manasdtrivedi@gmail.com
8197112777

Omanshu Mahawar
181CO237
omanshumahawar1234@gmail.com
8209104660

Contents

1. Software Development Life Cycle

1.1 Waterfall model

The Software Development Life Cycle (SDLC) model chosen for building the project, ‘An Automated System to Connect Users and Patients based on Service-Oriented Architecture’ (ASCPH) is the Waterfall Model.

1.2 Rationale

The waterfall model was chosen for the development of ASCPH for the following reasons.

a. The size of the team working on the project is small.
b. The requirements, which have been documented in the Software Requirements Specification, are expected to undergo minimal changes, if any.
c. The model uses a clear structure, and the progression is intuitive.
d. There is a clean transfer of information between the phases of the model.
e. The end goal of the project is determined from the beginning, which reduces the potential of deviation from the goal.

1.3 Stages

1.3.1 Requirements engineering

The services, constraints, and goals of the system required by the stakeholders shall be established. Upon obtaining the requirements, an analysis of the requirements shall be performed to ensure that the requirements are clear, complete, consistent, and unambiguous. A documentation of the requirements shall be presented to the stakeholders in the form of a Software Requirements Specification (SRS) document. The requirements shall then be validated by communicating the same with the stakeholders.

1.3.2 Software design

In this stage, a detailed study of the existing state-of-the-art methodologies in developing SOA-based systems shall be carried out. A methodology shall be proposed for the development of ASCPH, which shall be followed by preparing conceptual, logical, and technical designs.

1.3.3 Implementation and unit testing

Based on the design document obtained from the design phase, the code for the software shall be written. Optionally, improvements may be made during this phase which may involve using newer approaches or technologies. Program units shall then be tested to ensure that they satisfy the specifications mentioned in the SRS.

1.3.4 Integration and system testing

In this stage, program units shall be integrated to form the ASCPH. The integrated system shall be tested to ensure that it satisfies the specifications mentioned in the SRS.

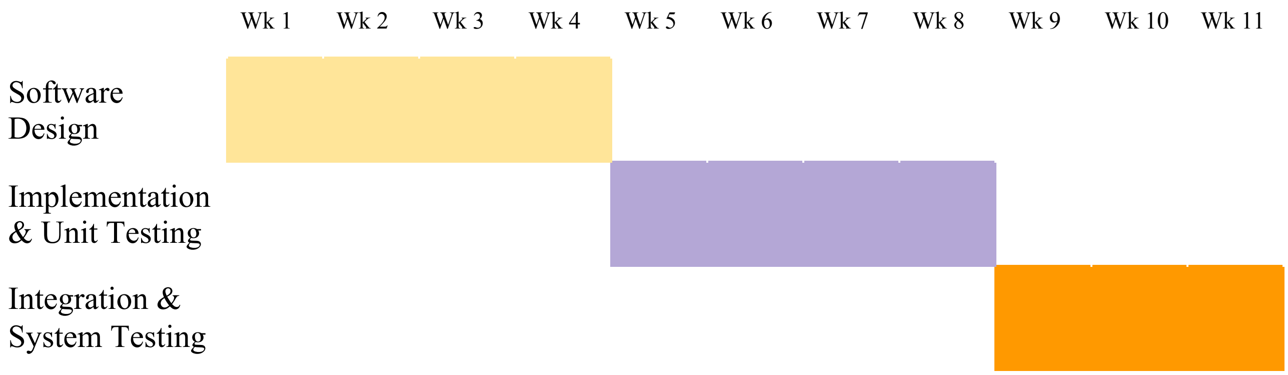

Fig. 1 Waterfall model for ASCPH

2. Requirements engineering tasks

2.1 Requirements elicitation

For ASCPH, the requirements were discovered and understood by communicating with stakeholders. Since ASCPH is targeted towards the general public for finding hospitals and booking appointments, it was agreed upon by the project members that peers, family members, and other individuals could be considered as the stakeholders of the system. Interactions with these stakeholders resulted in an unordered collection of requirements.

In addition to the abovementioned elicitation technique, some requirements were decided by the project members by ethnography. Since ASCPH is to be used by individuals living in a variety of environments, which includes remote locations, some requirements were written by considering these scenarios.

The requirements were then classified into functional and non-functional requirements.

2.2 Software Requirements Specification

The requirements obtained from the elicitation phase were documented in the form of an SRS document. This document was written by following the template provided by the IEEE, namely, IEEE-830-1998. The requirements were initially written in natural language, and were later rewritten in structural natural language.

2.3 Requirements verification and validation

The requirements mentioned in the SRS were reviewed by peers, who stated that structural natural language would be a better alternative, as opposed to natural language, since some requirements in the SRS were ambiguous due to being written in the latter manner. The SRS written in structural natural language was reviewed again by peers, who verified the requirements mentioned in the document. The SRS was then presented to the stakeholders identified during the elicitation phase, who validated the requirements.