Business Requirements Document

NFS Infinity

By David Rosenberg
U00063482
Contact e-mail: dcrsnbrg@memphis.edu

Prepared
October 30, 2017

Version 1.0

TABLE OF CONTENTS

1. Executive Summary

On a solo venture, David Rosenberg has been assigned to a new project: a network file server (NFS) app, which will allow a user to manipulate a remote file system. The project is being considered by ReTrans for incorporation into their business processes. David will report to the ReTrans contact and project overseer, Dr. Michael Bartz.

2. Business Objectives

This project will be an experiential study of agile software development techniques and processes including:

- Project planning
- System design and architecture
- Relationship-building
- Requirements gathering
- Documentation
- Coding (Python and Django)
- Testing

3. Non-functional Requirements

The software will be:

1. Environmentally sound, in that it won’t be using paper
2. Available on a local server and accessible by webpage
3. Simple to administer and use
4. Secured by a user login system

4. Functional Requirements

The software will:

1. Utilize a database to obtain file system information based on user, document type, or other qualifiers (keys).
2. Provide the following file manipulation instructions:
    - Directory list
    - Get all documents
    - Get one document
    - Delete document
    - Put document

5. Personnel Requirements

The engineer-in-training (David Rosenberg) is expected to build on his understanding of:

- Techniques, skills, and modern tools involved in agile development processes
- Meeting the specifications of the customer in terms of software functionality
- Communication with the customer
- Time management
- Respect for agile software development

6. Reporting and Quality Assurance

Development progress will be documented such that work will be traceable for future maintenance and testing purposes. Milestones in this process will also be submitted to Dr. Michael Bartz for approval and critique.

7. Delivery Schedule

Problem Statement: October 20, 2017
BRD: October 31, 2017
User Stories: November 7, 2017
Architecture Document: November 9, 2017
Design Documents: November 15, 2017
Final delivery of the software package: December 1, 2017.

8. Risk Management

Time management and quality assurance will be maintained by accountability to the delivery schedule outlined above. Regular status reports will be delivered to the contact, Dr. Michael Bartz.

9. Constraints

The primary development constraints involve time management and coding. Delivery of functional software is expected by December 1, 2017, and the software will be coded in a Python/Django environment. The interface between the system and user is to be attractive, web-based, and simple to navigate.

10. Glossary of Terms

This section provides clarity to terms used in this document that may be unfamiliar to an outside audience.

Functional/non-functional requirements
Simply put, non-functional requirements describe how the system works. Functional requirements describe what a system should do.

Get/put
Get and put are terms referring to exchange of data with a remote file system. Get means to download and put means to upload.

Network File System (NFS)
NFS is a client/server application that lets a computer user view and optionally store and update files on a remote computer as though they were on the user's own computer (Source: http://searchenterprisedesktop.techtarget.com).