UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

DEPARTMENT OF COMPUTER SCIENCE

COS 301 - MINI PROJECT

Assignment 1
Requirements and Application Design

Author:
Daniel King

Azhar Mohungoo

Andreas du Preez

Banele Nxumalo

Frederic Ehlers

Diana Obo

Bilal Muhammad

Student number:
u13307607

u12239799

u12207871

u12201911

u11061112

u13134885

u13080335

February 18, 2016

Requirements Specification

Bravo github repository link

For further references see [gitHub](#). February 18, 2016

1   Introduction

This document aims to specify the functional and non-functional requirements of a document archiving system, as specified by Ms Vreda Pieterse of the Computer Science Department.

It will serve as a means of communication between the client and developers as well as providing an elaboration and a clear discription of it's implementation specifications.

2   Vision

We intend to create a system that will allow authors and their co-authors to work on their research papers in an environment that reassures collaborative work, which in turn diminishes the time spent on papers with multiple authors. The following are what we plan to achieve:

- Keep track of research papers.

- View meta data of research papers.

- Allow multiple authors to collaborate on the same research paper.

- Different levels of authority, i.e. Admin, Author, User.

- View and edit the details of a text based profile of differents researchers.

- Implemented as a Website and Android application.

3   Background

We live in a world where time is valuable. We would like to do as much as we possibly can in the shortest amount of time. And if that's not possible, we work in teams to ensure that we achieve that goal.

Reseacher papers tend to be fairly lengthy, and if completed by only one author, it could be quite a tedious process. Hence we propose a system which would make the storage and collaboration of research articles and papers effortless by producing an archive system.

4   Architecture Requirements

4.1   Access channel requirements

The different access channels for the system will be as follows:

Web Application

The system can be used via a web application that uses RESTful web services and bootstrap technology. The system will be fully supported on the following web browsers:

- Chrome 7.0.517 and up

- Firefox 3.6 and up

- Safari 4 and up

- Edge

Bootstrap will be used so that the web application will also be accessible by mobile web browsers that support HTML5 and JavaScript technology.

Mobile Application

The system will also be accessible via a mobile application and will be operational on the following mobile operating systems:

- Android 4.0 (Ice Cream Sandwich) and up.

4.2 Quality requirements

- Security - The system shall identify all of its client applications before allowing them access to those enities. Possible measurement methods: Success rate in authentication, percentage of successful attacks, encryption level and probability/time/resources needed to attack the system.

- Usability - The degree of ease of use and training needed for end users. Possible measurement methods: Time it took a user to find a report (search functionality), percentage of deadlines met (notification functionality) and time it took a user to perform certain tasks.

- Auditability - The degree to which transactions can be traced. Possible measurement methods: The number and precision of logs generated in a period of time (metadata accessed/deleted/added).

- Portability - Measure ability of the system to run under different computing environments. Possible measurement methods: Number of targeted software environments (ie different browsers and operating systems) and proportion of platform specific functionality.

- Maintainability - Measures ability to make changes quickly and cost effectively. Possible measurement methods: Degree of complexity to make changes to the format of the metadata and mean time to add or change certain functionalities (ie new report types and new types of users).

- Availability - Percentage of time that the system is up and running correctly. Possible measurement methods: Length of time between failures and length of time needed to resume operation after a failure.

- Performance - Possible measurement methods: How well the system perform under high workload and number of events processed/denied in some interval of time.

4.3   Integration requirements

4.4   Architecture constraints

- The platform must exist in 2 mediums, namely web and android application.

- It is an internal network to be used by the Computer Science department, so it need not to source any information from the web.

- Any form of database can be used, but the team has decided that MySQL would be the best option from the current skillset.

- PHP and JavaScript will be used for the web and Android Studio for the android application.

- The system should cater for and including 100 users as a maximum.

- There should be a log that records all activity on the system

5   Functional requirements and application design

5.1   Use case prioritization

This section specifies the ranking or order in terms of the priority of services per use case. The use cases can be classified under 3 catogories: Critical ,important and nice-to have.

- Server connection (Critical)

- Database connection(Critical)

- User gateaway (Critical)

- Research paper management(critical)

- User account management(critical)

- Log services (critical)

- notification services(important)

- search services(important)

- Archival services(important)

6   Open Issues

7 References

PIETERSE, V. 2016. COS301: Mini project. In: Lecture notes issued online. University of Pretoria. Pretoria, South Africa