UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Department of Computer Science

COS 301 - Software Engineering

Software Requirements Specification

Authors:	*Student number:*
Herman Keuris	u13037618
Johan van Rooyen	u11205131
Estian Rosslee	u12223426
Ivan Henning	u13008219
Muller Potgieter	u12003672

February 18, 2016

Contents

1 Introduction

This section deals with the software software requirements specification of the program. This includes:

- The vision.
- The background.
- The access channel requirements.
- The quality requirements.
- The integration requirements.
- The architecture requirements.
- Use cases.
- Required functionality.

2 Vision

The client wishes to create a program that allows researchers to keep track of their own work, as well as collaborate with other authors, so that they can write papers together. Researchers will be able to specify the progress that they have made with their papers and alter them as needed. The program will keep a full record of all changes made to the papers, in order to create a time line of events. The program will be available as a browser and mobile application.

3 Background

1. A specialized program, aimed at research papers does not exist
2. It can be used as a common platform for researchers around the world to easily manage their work and collaborate more easily
3. It will allow researchers easier access to their work. persisting and provides access to domain objects.
4. It will allow researchers to keep track of their progress more easily
5. The program can be expanded to include researchers from other universities or scientific bodies

4 Architecture Requirements

4.1 Access Channel Requirements

1. A desktop application available in the form of a Windows (7/8/10) client, a Linux client and binaries ready to be built on either system with an interactive GUI
2. A web version compatible with all major browsers (eg. Mozilla Firefox, Google Chrome or Opera)
3. A mobile application developed for Android and compatible with all current and upcoming versions thereof

This will be accomplished by making use of RESTful web services (these being based on the REST, or REpresentational State Transfer architecture). The system itself will accept HTTP requests from any of these channels and create responses in the form of JSON strings, a format easily handled in any one of the aforementioned access systems. Additionally, the following additional access channels can be added:

1. A command line (terminal) based version of the desktop application, which could be suitable for the target audience, who are very technologically capable ready to be built on either system with an interactive GUI
2. A mobile application developed for Windows Phone and/or iOS.

4.2 Quality Requirements

The following assurances must be made in terms of quality:

- Performance
  1. The server must always provide the minimum data required to fulfill a request. That is to say, were a user to log in to the system, the server should only send the data pertaining to that user to be displayed, no Papers related to his or her Co-Authors or other related parties.
  2. The system must be created with the most minimal and efficient coding practices possible, given that the result must still be reliable and robust.
  3. No actual files are to be stored in the system, lest it negatively affect the performance components of the system itself.
- Reliability
  1. The system must be thoroughly tested on both the client and server side, to ensure it will not cause faults or problems. It is important that no data is lost, thus the coding used to create the system must be defensive and thorough.

- Scalability

    1. The system must be designed such that:
       a-The client is able to handle and display details of a large, potentially infinite number of Publications. b-The server is able to handle, display details pertaining to large, potentially infinite number of Users, Authors and Publications.
    2. Modular programming should be used in order to ensure that there are no restrictions in terms of the system's ability to be extended and improved upon at later stages.

- Security

    1. It should not be possible for individuals other than the actual Users to access or modify the system. This means that security has to be ensured in terms of password storage, secure login methods and user management (methods such as re-obtaining password via email should be very carefully guarded).
    2. It should not be possible for Users to make changes to other Users' details, as it is with non-User Authors, unless they are one of a select few Super Users or Administrators.
    3. A publication should not be able to be removed from a system, only edited, unless it is removed by an aforementioned Super User.
    4. A User should not be capable of viewing or editing a publication for which they are not on the list of Authors.

- Flexibility

    1. The system should be capable of reacting quickly to different stimuli. This means (as an example) that if multiple users are concurrently using the system and performing vastly differing tasks which make use of completely different parts of the same system, there should not be any noticeable loss of performance.
    2. The system should be able to perform well even under bulk loads, without loss of data on the way.
    3. It should be possible to add new components or fields to existing components in the system without making major changes.

- Maintainability

    1. The system should be developed with current and maintained technologies, so as to avoid loss of support for as long as possible.
    2. The system should be well documented so as to ensure future developers on the system are capable of maintaining the system without worry.
    3. When changes are made in current technologies, the system should be updated as soon as possible to reflect relevant changes.
    4. The modular design of the system must be such that if changes must be made to a part of the system, only that part itself should be changed.

- Monitorability

  1. **All** actions taken that have any affect on the databases stored server side are to be logged.
  2. All logs, current connections and current activity must be viewable by the Super Users in charge of the system.

- Integrability

  1. The system should be designed in such a manner (with modularity and common interfacing methods) that it is capable of having pieces or services plugged in and catered to with minimal effort, such as that of Google Calender, which could be a logical future addition to the system for the sake of deadline maintenance.

- Cost

  1. The tools used to design the system should, as far as possible, be open source, free and not require a license.
  2. In certain cases, paid and licensed software may be suitable for some individual pieces of the system, such as having a Database Management System (DBMS) to handle the storage of data as best possible.
  3. Costs may be created in the form of external hosting for the web service and database storage, should the client desire it to be so.

- Usability

  1. The Users, being staff members, must have easy access from any channel.
  2. The system should be designed in such a manner that the interface is easy to learn and use.
  3. The system should be minimal and avoid having unneeded visuals that could impair a User's ability to use the system comfortably.

4.3 Integration Requirements

4.4 Architecture Constrains

5 Functional Requirements and application design

5.1 Use case prioritisation

The Use Case Prioritisation will be elaborated upon in the next section, section 5.2: Use Case/Services Contracts.

5.2 Use case/Services contracts

1. User Login : Use Case Prioritization – Critical

   (a) Pre-Conditions

      i. A user must be registered as a user by admin before he/she is able to login to the Research Paper App.

      ii. In order to login a user must enter in his/her correct authentication details.

   (b) Post-Conditions

      i. The user has access to his/her profile and publications.

      ii. The user may alter his/her publications.

      iii. The user may access only his/her profile and publications and no others.

      iv. A user can be an author.

1. Author Login : Use Case Prioritization – Important

   (a) Pre-Conditions

      i. An author must be registered by admin as an author before he/she may login to the Research Paper App.

      ii. In order for an author to login he/she must enter in his/her correct authentication details

   (b) Post-Conditions

      i. The author has access to any profile that he/she co-authored.

      ii. The author may not alter any publications that he/she was involved in.

      iii. A user can be an author, but an author cannot be a user.

1. Super-user/admin Login: Use Case Prioritization – Critical

   (a) Pre-Conditions

      i. A single user must be able to logon as admin or a super user.

      ii. In order to login a user must enter in his/her correct authentication details.

   (b) Post-Conditions

      i. The super-user has access to any and all user and author profiles.

      ii. The super-user is the only user capable of adding more users and authors.

      iii. The super-user can alter any profile.

1. User Registration: Use Case Prioritization – Critical

   (a) Pre-Conditions

        i. The admin or super-user is in charge of registering the users.

   (b) Post-Conditions

        i. The user receives his/her login details.

        ii. The user has his or her privileges set.

        iii. The user is registered in the user and author database table.

        iv. The user is able to logon to the Research Paper App with the login details supplied by the super-user/admin.

1. Author Registration: Use Case Prioritization – Important

   (a) Pre-Conditions

        i. The admin or super-user is in charge of registering the authors.

   (b) Post-Conditions

        i. The author receives his/her login details.

        ii. The author has his or her privileges set.

            A. Can only view links of papers he/she has co-authored.

        iii. The author is registered in the author database table only.

        iv. The author is able to logon to the Research Paper App, as an author, with the login details supplied by admin or super-user.

1. Super-user/Admin Registration: Use Case Prioritization – Critical

   (a) Pre-Conditions

        i. Upon system-initialization, a single user sets him/herself to the super-user/admin.

        ii. This user uses his/her authentication details to log in as the super-user/admin.

   (b) Post-Conditions

        i. The super-user/admin can add and remove users

        ii. The super-user/admin can view all user and author profiles, as well as lists of publications associated with each.

1. Creating a User: Use Case Prioritization – Critical

   (a) Pre-Conditions

        i. Prior to a user logging into his/her profile, the super-user/admin must have created the user profile, which the user logs onto.

        ii. Upon logging in, a user must have full access to his/her profile page.

        iii. Profile page must include

            A. Full name of user.
            B. Contact details
            C. Cell phone number.
            D. Telephone number.
            E. Email Address.
            F. Conference for whom the user is researching.
            G. List of links to publications.
            H. A list of co-authors per publication (if any).

   (b) Post-Conditions

      i. The user must be able to edit his/her publication list as he/she sees fit.
            A. Adding Publications.
            B. Removing Publications.
      ii. The user may only view/edit his/her own profile and publications.

1. Creating an Author: Use Case Prioritization – Important

   (a) Pre-Conditions

      i. Prior to the author logging into his/her profile the super-user/admin must assign him/her an author profile.
      ii. Upon logging in, the author must have full access to his/her profile.

   (b) Post-Conditions

      i. Authors may not alter publications.
      ii. Displayed on their profile will be:
            A. Full name of the author.
            B. Contact details
            C. Cell phone number.
            D. Telephone number.
            E. Email Address.
      iii. Authors will be able to see each of the publications that they co-authored.
            A. No altering will be allowed.

1. Creating a new publication:

   (a) Pre-Conditions

      i. The user or super-user must provide a publication title.
      ii. The supervisor of the paper must be included.
      iii. All the authors who worked under that supervisor to co-author the paper must be listed.
      iv. A deadline must be set.
      v. Progress of the paper must be specified.

        A. Ongoing.
        B. Terminated.
        C. Completed.

  (b) Post-Conditions
    i. The new publication will be viewable by the user, the super-user and all authors involved.
    ii. The publication may only be edited by the user whom created the publication, or the super-user.

1. Setting the status of a publication: Use Case Prioritization – Nice to Have

  (a) Pre-Conditions
    i. If a user has successfully logged on, then he or she may view and alter the publications.

  (b) Post-Conditions
    i. Depending upon the status of the paper the user may alter it.
    ii. Only a user, whose privileges allow it, may edit the publication.

1. Viewing Publications As an Author: Use Case Prioritization – Nice to Have

  (a) Pre-Conditions
    i. Upon logging in, the author must have full access to his/her profile.
    ii. Profile page must include
        A. Full name of user.
        B. Contact details
        C. Cell phone number.
        D. Telephone number.
        E. Email Address.
        F. Supervisor for whom the author is researching under.
        G. List of links to publications for which he/she has co-authored.

  (b) Post-Conditions
    i. All publications that which the author has co-authored must be available to view.
    ii. Unless the author is also a user, he/she will be prohibited from altering any data

5.3 Required functionality

5.4 Process Specifications

6 Open Issues