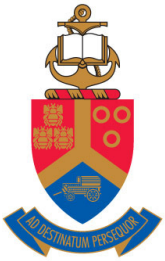

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

COS301 Mini Project Functional Requirements Specification

Group 4B

Kyhle Ohlinger *u11131952*
Andrew Parkes *u12189139*
Sifiso Shabangu *u12081622*
New Member *uxxxxxxxx*
New Member *uxxxxxxxx*
New Member *uxxxxxxxx*
New Member *uxxxxxxxx*

Here's a link to Github.

Version 1
February 19, 2015

Contents

1 Introduction **2**

2 Vision **2**

3 Background and System Description **2**
3.1 Related project 2
3.2 System Environment 3

4 The Stakeholders **3**
4.1 The Client 3
4.2 The customers 3
4.3 Maintenance Users 3

5 Functional Requirements **4**
5.1 Scope and Limitations/Exclusions 4
5.2 Use case Prioritization 4
5.3 Use case/Services Contracts 4
5.4 Required Functionality 4
5.5 Process Specifications 4
5.6 Domain Model 4

6 Temporary Space **5**
6.1 Kyhle 5
6.1.1 Point 1: 5
6.1.2 Point 2: 6
6.1.3 Point 3: 6
6.2 Andrew 8
6.3 Sifiso 9
6.3.1 Point 1: 9
6.4 name 10
6.5 name 11
6.6 name 12
6.7 name 13

1 Introduction

The purpose of this document is to fully specify and outline the functional requirements of "The use of Online Discussions in Teaching (TODT)" research project, received from the Computer Science Education Didactic and Applications Research (CSEDAR) team of the Computer Science Department of the University of Pretoria. The document also serves to give the client and developers a clear description and elaboration of the system to be implemented in its totality.

2 Vision

The project aims to provide an online space which will be integrated into the CS website, where students, teaching assistants, and lecturers can engage in activities related to learning the content of our module.The system will also apply game concepts to motivate students to increase the quality of their participation and consequently experience deeper learning of the course content.

3 Background and System Description

This project is due to the Computer Science department of the University of Pretoria having problems with the currently available tools for discussion forums, the following problems are hampering positive engagement of both teaching staff and students: Unorganised content, user inexperience and low levels of excitement.
The System intends to create an online discussion forum that has automated feedback on common mistakes, game-like presention as well as automated feedback. The system also provides the COS 301 students with the opportunity to learn about the procedures used for creating, designing and developing projects for businesses, while also providing the University with a potentially new system that may, be released as an opensource project, that could possibly be implemented worldwide.

3.1 Related project

The project is a face lift to the existing dicussion forum of the Department of Computer Sciences and aims to improve the existing one by bringing new features that help Students to be more involved in dicussing certian modules.

3.2 System Environment

The system will interract with LDAP ,which will handle credentials avoiding the need for a database

4 The Stakeholders

4.1 The Client

The Client is Ms Vreda Pieterse at the Department of Computer Science.

4.2 The customers

The Customers are Students of the Computer science who are enrolled in the modules and lectures in the department

4.3 Maintenance Users

The system will be assigned admimstartors from the Computer Science department and they will ensure its maintenace

5 Functional Requirements

Temporary words

5.1 Scope and Limitations/Exclusions

Temporary words

5.2 Use case Prioritization

Temporary words

5.3 Use case/Services Contracts

Temporary words

5.4 Required Functionality

Temporary words

5.5 Process Specifications

Temporary words

5.6 Domain Model

Temporary words

6 Temporary Space

6.1 Kyhle

Use diagrams for points 4,5,and 6

6.1.1 Point 1:

1. **Scope:** Users should be able to create, read, update and delete posts. **Limitations/exclusions:** Not all users should be able to use all the functions. Some users may even CRUD other users posts.

2. **Use case Prioritization:** Critical

3. **Use case/Service Contracts:**

   Pre-Conditions:

   - User must be connected to the buzz system.
   - *Create:*
     - Must have necessary permission to create posts.
     - Must be registered on the buzz system.
   - *Read:*
     - Post must exist.
   - *Update:*
     - Post must exist.
     - Must either be owner of the post, or have necessary permissions to update the post.
   - *Delete:*
     - Post must exist.
     - Must either be owner of the post, or have necessary permissions to delete the post

   Post-Conditions:

   - *Create:*
     - Post will have been created.
     - Post may not have been created, due to some error.
   - *Read:*
     - If logged in, post will be marked as read for the specific user.

- *Update:*
  - Post will be updated if user has required permissions.
  - Post may not have been updated if permission requirements aren't met.
- *Delete:*
  - Post will be marked as deleted, and thus removed from the discussion board.
  - Post is not actually removed from the server, it is however hidden from all users.
  - Post may not have been deleted if permission requirements aren't met.

6.1.2 Point 2:

1. **Scope:** The system must keep track of who has read what, and highlight unread messages for each user.
   Limitations/exclusions:
2. **Use case Prioritization:** Critical
3. **Use case/Service Contracts:**
   Pre-Conditions:
   - User must be registered to the buzz system.
   - User must be logged into the system while viewing the post.

   Post-Conditions:
   - Post is marked as read.
   - Post remains unmarked because user isn't registered or logged into the system while viewing the post.

6.1.3 Point 3:

1. **Scope:** This deals with the restriction of posting messages.
   Limitations/exclusions: Message length should be restricted. Content type should also be restricted based on level and status of the user posting the message.
2. **Use case Prioritization:** Critical

3. **Use case/Service Contracts:**
   Pre-Conditions:

   - User must be registered to the buzz system.
   - User must have necessary permissions to create posts of a certain length or content type.
   - Content type and message length must be established by the creator of that specific buzz.

   Post-Conditions:

   - Post is created.
   - Post may not have been created because user doesn't have required permissions.
   - post created with incorrect length or content type due to creator of the buzz not configuring the requirements.

6.2 Andrew

1.
2.
3.
4.
5.
6.

6.3 Sifiso

6.3.1 Point 1:

1. **Scope:** This deals with social tagging,broad folksnomy type is used in this social tagging
   Limitations/exclusions: Not all users will be able to tag a buzz space,Users with higher privilagies and lectures will be able to social tag buzz space

2. **Use case Prioritization:** nice to have

3. **Use case/Service Contracts:**
   Pre-Conditions:

   - User must be registered to the buzz system.
   - User must have necessary perivilages.
   - Buzz space must have a rating from users to be tagged

   Post-Conditions:

   - Buzz space tagged with a keyword.
   - Buzz space avilable at tag box for fast access.

6.3.2 Point 2:

1. **Scope:** This deals with self-organisation based on social tagging and allow the user to view according to the base structure, owns structure or public structure.
   Limitations/exclusions: Users with higher privilagies will be able to organise view to thier own structure.

2. **Use case Prioritization:** nice to have

3. **Use case/Service Contracts:**
   Pre-Conditions:

   - User must be registered to the buzz system.
   - User must have necessary perivilages.
   - Buzz space must have a rating from users.
   - Social tagging must be applied to other buzz space

   Post-Conditions:

- Tagged buzz space with higher rating from users must be organised to be in base structure
- Most accesed tagged buzz space must be included in public structure.
- Users with buzz space can organise thier own structure .

6.4 name

1.
2.
3.
4.
5.
6.

6.5 name

1.
2.
3.
4.
5.
6.

6.6 name

1.
2.
3.
4.
5.
6.

6.7 name

1.
2.
3.
4.
5.
6.