Software Requirements Specification

for

TA Processing System (TAPS)

Version 1.0 approved

Prepared by Guanpin Zhong

University of Minnesota--CSci 5801

03/08/2019

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

This document is used to describe the details of the software TA Processing System (TAPS). It will focus on its purpose and how it works, as well as its interface and traits. This document is intended for general user, but also for developers who are interested in its design.

1.2 Document Conventions

This document is based on the IEEE System Requirement Specification Template.

1.3 Intended Audience and Reading Suggestions

There are three main types of users in the system:

- Teaching Assistants, including appointed TAs who want to accept or reject offers, and prospective TAs who want to enter personal information for the applications.
- Faculty, who want to submit TA recommendations and requests.
- Administrators, such as CSE staff who want to make overall arrangements, or payroll staff who want to control the details of appointments.

1.4 Product Scope

The computer science department is often busy dealing with assigning graduate TAs at the beginning of the semester, so the TA Processing System (TAPS) is designed to help solve this problem. Specifically, the faculty and TAs can use it to satisfy their requests, and administrators can use it to deal with the data more conveniently. In general this software is able to increase the efficiency when dealing with the TA issues.

1.5 References

IEEE System Requirement Specification Template:
https://canvas.umn.edu/courses/98114/files/5562794/download?download_frd=1

2. Overall Description

2.1 Product Perspective

TAPS was developed for people who are involved in the process of assigning graduate TAs for all CSCI courses. Everyone will be able to do their own part of work on this software, such as entering requests or analyzing related data. The overall efficiency will be improved obviously, and the assigning results can be more proper.

2.2 Product Functions

For Prospective TAs:

- enters the type of appointment to be requested
- provides previous experience of being TAs
- enters course preference and qualifications
- enters different types of scores, including language and technical aspects
- enters personal and academic information in detail

For Faculty:

- submits TA recommendations and requests forms
- requests students who would not get a TA appointment solely on the basis of priority
- be notified of the TA appointments

For Staff:

- gets data for the list of courses being offered and estimated number of TAs in need
- fetches data for summary and statistics from previous years
- appoints TAs to courses according to a priority score depending on a number of factors
- rejects requests from prospective TAs
- updates the status of the appointments
- sends out notifications to TAs, faculty and payroll
- notifies TAs with some special offers with extra requirements, such as the language

For Payroll staff:

- be notified of the offers
- views the list of appointments and completion proportion of each type
- sends out appointment details to TAs
- checks and updates appointments
- notifies other users the appointment status updates

For Appointed TAs:

- be notified of the offers
- respond to the offers: accept or reject

2.3 User Classes and Characteristics

There are three main classes of users in the system:

- Teaching Assistants, including appointed TAs who want to accept or reject offers, and prospective TAs who want to enter personal information for the applications.
- Faculty, who want to submit TA recommendations and requests.
- Administrators, such as CSE staff who want to make overall arrangements, or payroll staff who want to control the details of appointments.

2.4 Operating Environment

TBD

2.5 Design and Implementation Constraints

TBD
(TAPS uses XXX interface, XXX database abstractions. They can always be revised. It also utilizes a third-party software to request data for various components.)

2.6 User Documentation

TBD

2.7 Assumptions and Dependencies

TBD

3. External Interface Requirements

3.1 User Interfaces

TBD

3.2 Hardware Interfaces

TBD

3.3 Software Interfaces

TBD

3.4 Communications Interfaces

TBD

4. System Features

TBD

4.1 System Feature 1

4.1.1 Description and Priority

TBD

4.1.2 Stimulus/Response Sequences

TBD

4.1.3 Functional Requirements

TBD

TBD

REQ-1:
REQ-2:

4.2 System Feature 2 (and so on)

5. Other Nonfunctional Requirements

5.1 Performance Requirements

TBD

5.2 Safety Requirements

TBD

5.3 Security Requirements

TBD

5.4 Software Quality Attributes

TBD

5.5 Business Rules

TBD

6. Other Requirements

TBD

Appendix A: Glossary

TBD

Appendix B: Analysis Models

TBD

Appendix C: To Be Determined List

TBD