

# Software Requirements Specification

Version 1.0

April 16, 2015

Career Services  
Event Registration Software

Kyle Johnson (Team Lead)  
Dylan Holliday  
Kevin Reynolds  
Zhenyu Xia

Eastern Washington University - CSCD 488  
Instructor: Chris Peters

## Table of Contents

|      |                              |  |
|------|------------------------------|--|
| 1.   | Introduction                 |  |
| 1.1. | Purpose                      |  |
| 1.2. | Project Scope                |  |
| 2.   | General Description          |  |
| 2.1. | Functions                    |  |
| 2.2. | Description                  |  |
| 3.   | Specific Requirements        |  |
| 3.1. | User Interface               |  |
| 3.2. | Student ID Card Reader       |  |
| 3.3. | Online Pre-Registration Form |  |
| 3.4. | Barcode Receipt Scanner      |  |
| 3.5. | Name Tag Printer             |  |
| 3.6. | Administrative Tool          |  |
| 4.   | Use Cases                    |  |
| 4.1. | Registrant Use Case          |  |
| 4.2. | Administrator Use Case       |  |
| 5.   | Deliverables                 |  |
| 6.   | Timeline                     |  |

# 1. Introduction

## 1.1 Purpose

The purpose of this project is to streamline the Career Services event registration process by creating software that will allow for quicker registration and easier collection of event data. The current event registration process requires kiosk attendants to manually enter data for the people registering and could be made more efficient by giving the registrants the ability to quickly register themselves.

## 1.2 Project Scope

The intended scope of this project is to create software for use by Eastern Washington University's Career Services department for on-site event registration. The software will allow registrants to quickly register for an event by swiping a student ID card through a card reader, scanning a barcode receipt received from pre-registering online, or by using the user interface to enter their information on a kiosk laptop. After registering, the registrant will receive an identifying name tag with their information on it. Statistical data will be collected from the registrants for the event that can then be exported to an Excel spreadsheet for data analysis.

# 2. General Description

## 2.1 Functions

This software:

1. can be installed easily.
2. will have a form for registering.
3. will be able to accept a card swipe.
4. will have a database.
5. will export data to excel.
6. can accept a bar code.

## 2.2 Description

This is a program that will have a window which will be the main front end. If a student pre-registers it will be able to read a bar code. It will allow users to register themselves. If the user chooses to, they can use a card scanner. There will be a back end database filled with individual school's student information. Once a student has registered their information is then looked up in

the database provided. The program will output the data into a database. The database can be exported to Excel.

### 3. Specific Requirements

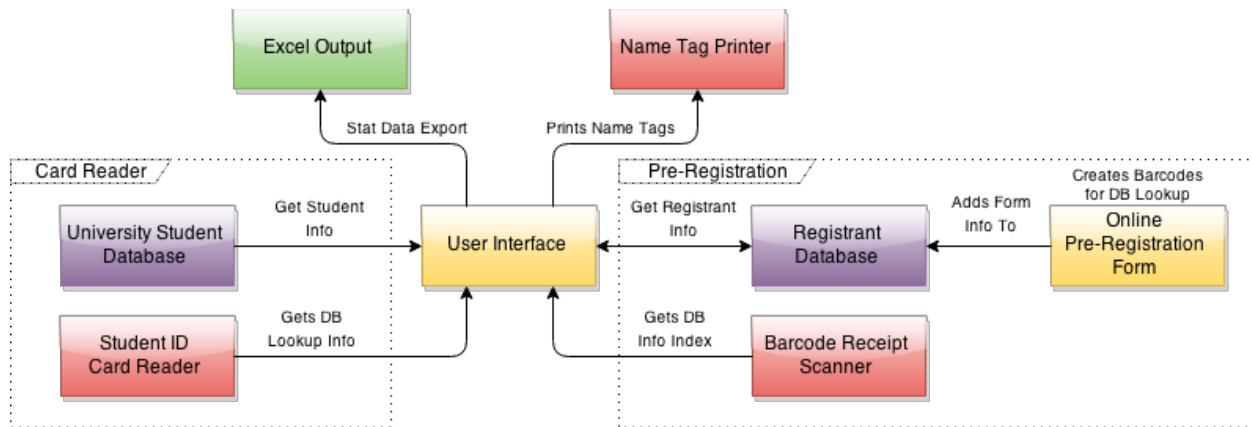


Figure xx - Project Aspects Overview

#### 3.1 User Interface

The user interface will be used for manual input of registrant information. It must be able to allow the user to distinguish what kind of registrant they are--either student, alumnus, business owner, or other attendee--and then input relevant information. After all of the information fields are filled out, the user should then be able to click a button to indicate that they are finished. Upon clicking, the user should receive a name tag printed with their information on it.

#### 3.2 Student ID Card Reader

Students who attend the event should be able to swipe their student ID cards through a card reader to populate the forms on the user interface. The student should then be able to verify their information on the screen, make changes to it if necessary, and accept the information before clicking a button to print a name tag. If the card is unreadable, the student should be notified of the error and be directed to manually enter their information at the kiosk.

#### 3.3 Online Pre-Registration Form

Anyone wanting to attend an event may register ahead of time online to speed up their check-in process. They should be able to go online to a pre-registration form similar to what would be available at an on-site registration kiosk. The user should be able to enter their information in the same way, submit it, and receive a barcode receipt along with a notification that their registration information was received. This barcode should then be used on-site for a quicker check-in.

### 3.4 Barcode Receipt Scanner

Registrants who registered ahead of time online should be able to scan a barcode receipt, either on a mobile device or printed copy, and have their information display on the user interface. The registrant should then be able to verify their information, modify it if desired, then click a button to print a name tag. If the barcode is unreadable, the registrant should be notified of the error and be directed to manually enter their information at the kiosk.

### 3.5 Name Tag Printer

The name tag printer should be able to print a label with the registrant's name and relevant information on it. Student name tags should include their university, major, and class standing (freshman, sophomore, junior, senior, graduate student). Alumnus name tags should include their university, major, and year of graduation. Business owner name tags should include their job title and place of occupation. Other registrant name tags should only display their name and a relevant title.

### 3.6 Administrative Tool

We didn't discuss this one. I think we should have a tool for the administrator to use to export data to an Excel sheet. This would probably be part of the main user interface, but accessed via an administrator button press and a log-in screen.

## 4. Use Cases

### 4.1 Registrant Use Case

...

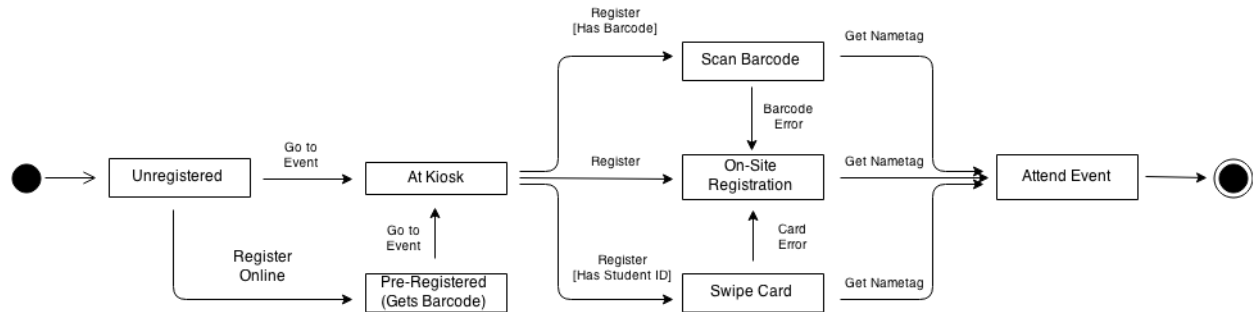


Figure 1 - Registrant State Diagram

### 4.2 Administrator Use Case

...

## 5. Deliverables

1. Software Requirements Specification

## 6. Timeline

Not sure how we should format this or if we need anything definite just yet.

Week 1 (April 20 - 26)

- Planning Process

Week 2 (April 27 - May 3)

Week 3 (May 4 - 10)

Week 4 (May 11 - 17)

Week 5 (May 18 - 24)

Week 6 (May 25 - 31)

Week 7 (June 1 - 7)