Requirements Engineering Report

Date: 10/29/2018

Version:

[You can add an abstract or other key statement here. An abstract is typically a short summary of the document content.]

\*\*\*Team Name

UNCW Schedule Generator

Team Members

Chase Robison

Mark Thompson

Dakota Perryman

Contents

[1. Introduction 1](#_Toc410643945)

[1.1. Project Objective 1](#_Toc410643946)

[1.2. Procject ScoPe 1](#_Toc410643947)

[1.3. Success Criteria 1](#_Toc410643948)

[1.4. Collaboration with Stakeholders 1](#_Toc410643949)

[2. Project Plan 2](#_Toc410643950)

[2.1. Work Breakdown Structure (WBS) 2](#_Toc410643951)

[2.2. Project Resources 2](#_Toc410643952)

[2.3. Responsibility Matrix 2](#_Toc410643953)

[2.4. Gantt Chart 2](#_Toc410643954)

[2.5. Pert Chart 2](#_Toc410643955)

[2.6. Cost Estimation 2](#_Toc410643956)

[2.6.1. Function Point Estimation 2](#_Toc410643957)

[2.6.2. Lines of Code Estimation 2](#_Toc410643958)

[2.6.3. Cost Estimates 2](#_Toc410643959)

[2.7. Risk Plan 2](#_Toc410643960)

[2.8. Project Monitoring and Control Mechanisms 2](#_Toc410643961)

[3. Requirements/Analysis Models 3](#_Toc410643962)

[3.1. Major Software Functions 3](#_Toc410643963)

[3.2. Use Case Diagrams 3](#_Toc410643964)

[3.3. Use Case Descriptions (Fully Dressed) 3](#_Toc410643965)

[3.4. Activity Diagrams 3](#_Toc410643966)

[3.5. Sequence Diagrams 3](#_Toc410643967)

[3.6. Requirements Class Models 3](#_Toc410643968)

[3.7. Prototype Description 3](#_Toc410643969)

[3.8. Data Dictionary 3](#_Toc410643970)

[3.9. Limitations and Constraints 3](#_Toc410643971)

[3.10. Non-functional Requirements 3](#_Toc410643972)

[4. Problems Encountered 4](#_Toc410643973)

[5. Bibliography 5](#_Toc410643974)

# Introduction

## Project Objective

To develop a website that provides a user-friendly atmosphere to UNCW students for aiding in the process of making efficient class schedules. This project will take in a number of course subjects with associating course numbers and then generate a schedule based on those constraints and others specified criteria by the user. The result will be a course schedule displayed to the user with no time conflicts, if possible.

## Project ScoPe

Making a schedule every semester is something you can not avoid when being a college student. The process of generating a schedule that has no conflicts and fits all of the students’ needs is a very time-consuming process. The current process consists of writing classes down on paper and manual checking conflicts by comparing them against other classes on SeaNet. With this tool, we hope to reduce time spent finding the perfect schedule by taking this burden off the students and implementing this problem into our own unique algorithm. Once our algorithm finds an optimal schedule, if one exists, the user will be presented with all their desired courses in a structured weekly format.

## Success Criteria

* Use standalone application to gather information from SeaNet
* Update database with SeaNet information
* Have a functional website
* Take user input for desired courses
* Return a schedule with no time conflicts based on input
* Allow user to make multiple searches

# Project Plan

## Work Breakdown Structure (WBS)

Insert text of WBS.

Insert figure of WBS.

## Project Resources

* Visual Studio
* Selenium
* HTMLAgilityPack
* MySQL
* SeaNet
* GitHub

## Responsibility Matrix

|  |  |  |  |
| --- | --- | --- | --- |
|  | Chase | Mark | Dakota |
| Database | I | I | R |
| Time Conflict Algorithm | I | R | C |
| WebPage | R | I | C |
| StandAlone App | R | C | C |

R- Responsible A-Accountable C-Consulted I-Informed

## Gantt Chart

## Pert Chart

## Cost Estimation

### Function Point Estimation

### Lines of Code Estimation

* 5000 lines of code

### Cost Estimates

## Risk Plan

* None

## Project Monitoring and Control Mechanisms

* None

# Requirements/Analysis Models

## Major Software Functions

## Use Case Diagrams

## Use Case Descriptions (Fully Dressed)

## Activity Diagrams

## Sequence Diagrams

## Requirements Class Models

## Prototype Description

## Data Dictionary

## Limitations and Constraints

## Non-functional Requirements

# Problems Encountered

# Bibliography