Requirements Analysis Document

Employee Scheduling System

CSCI 4711: Software Engineering

Fall 2016

Augusta University

Augusta, GA

Date: 19 September 2016

Version 1

Team Members

Chris Gonsalves

Matt Tennis

Connor Williams

Ryan Mahoney

Abstract

This document contains the requirements, analysis and design artifacts for the Employee Scheduling System (ESS) software system. ESS is a personnel scheduling system that facilitates the employee submission and subsequent supervisor approval or denial of time off requests.

The rest of this document is structured as follows: Chapter 1 contains the introduction. This chapter presents a brief description of the system. Chapter 2 outlines the functional requirements of the system.

TABLE OF CONTENTS

1 INTRODUCTION

1.1 SCOPE OF SYSTEM

The Employee Scheduling System (ESS) is a distributed information system used to provide simple and efficient means for an employee to request time off and for appointed supervisors to administrate, approve, or deny those requests.

ESS has an internal database with authorized users and their password hashes. Employees can submit requests for time off, which are stored in the database. Supervisors are then able to see the contents of the time off requests, the employee that initiated it, and the reason for the request. Once the Supervisor responds to a request, it is removed from the Supervisor's queue and the database.

The system includes secure login, logout functionality in addition to the primary scheduling applications.

1.2 OVERVIEW OF DOCUMENT

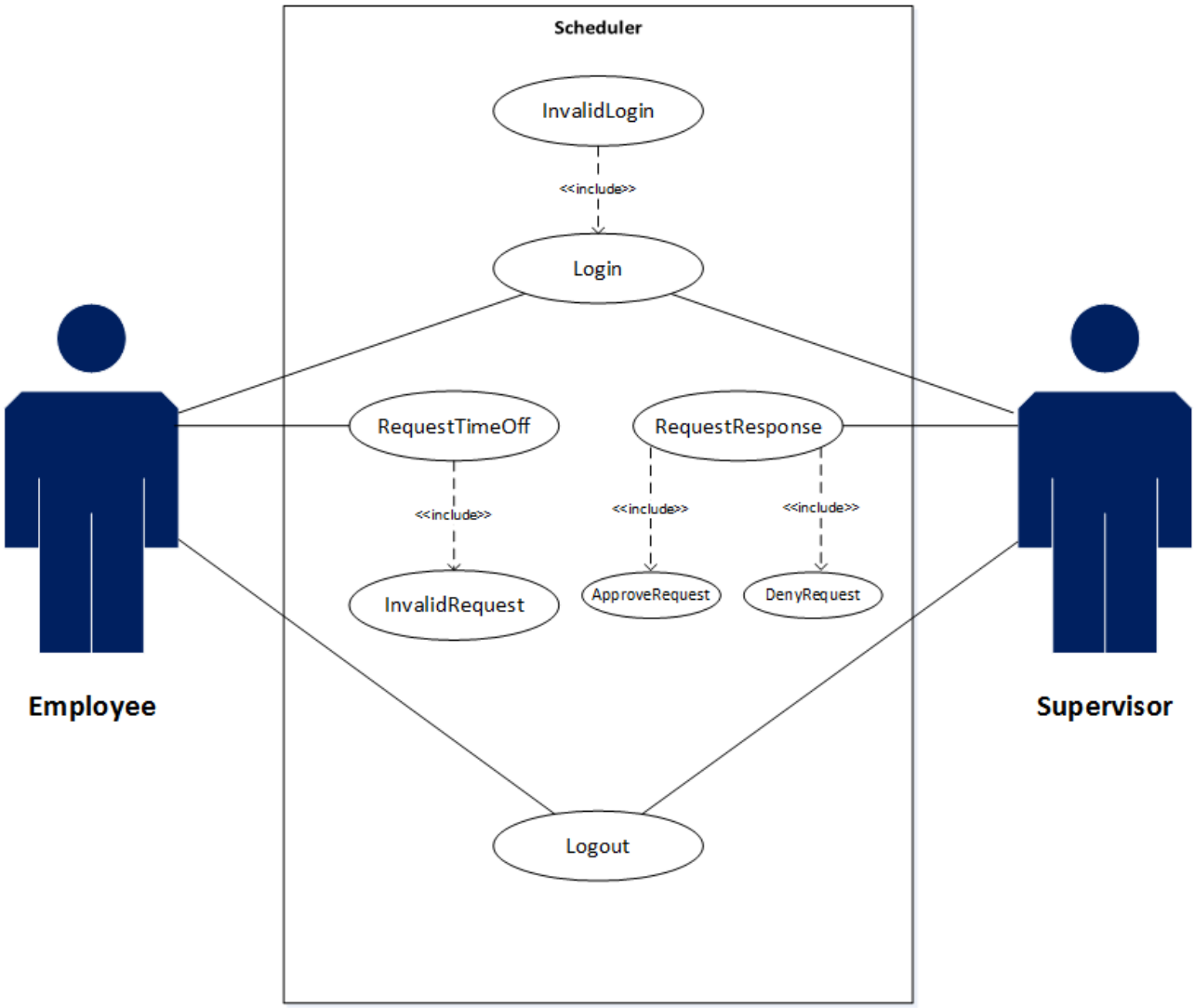
The rest of the document is structured as follows: Chapter 2 outlines the functional requirements of the system, then the use case diagram. Individual detailed use case descriptions are then listed.

2 REQUIREMENTS OF SYSTEM

2.1 FUNCTIONAL REQUIREMENTS

- **Login –** All users, **Employees** and **Supervisors**, must supply valid login credentials (EmployeeID and password) to be authorized to access and use the system. Upon doing so, the user will have created a session with ESS, where a user can modify database contents through normal usage. Valid login will direct the user to his or her appropriate activity based on the user's class.
  - **InvalidLogin** – Handle invalid credentials, out-of-scope characters, and exploitation attempts. Returns control to user after job.
- **Logoff –** All users must have clear and immediate access to a Logoff button in order to gracefully and securely close the connection with ESS. Resources allocated to a user session must be terminated in an orderly fashion as to eliminate potential software bugs. Every form or interface must have a clearly marked Logoff button.
- **RequestTimeOff – Employees** must be able to supply a time off request in the appropriate forms: either the calendar GUI or the text-based message box (which accepts only date information). Radio buttons enable the **Employee** to indicate the reason (and weight) of his or her request. The user can then submit, cancel, or logout from that page, with the first checking if the request is valid, then sending it to the database, and the other two terminating the request and logging out, respectively.
  - **InvalidRequest** – Handle invalid parameters supplied in a request, notifies user, and then returns control.
- **RequestResponse – Supervisors** must be able to view the time off requests that have been submitted in a scroll box queue. The queue will have highlighted regions that correspond to the reason (or weight) supplied by the user's time off request. The **Supervisor** can then approve, deny, or logout from this window. Approvals and denials modify database contents and update the queue with that request's removal, while logout will terminate the session gracefully.
  - **ApproveRequest** – Approves Employee request and updates relevant database data.
  - **DenyRequest** – Denies Employee request and updates relevant database data.

2.2 USE CASE DIAGRAM



2.3 USE CASE DESCRIPTIONS

Use case name	Login
Participating actors	Initiated by Employee
Flow of events	1. Employee enters their EmployeeID and Password. 2. ESS responds by authenticating the entered EmployeeID and password. Upon authentication, ESS displays the appropriate interface.
Entry condition	The Employee enters their login information into ESS
Exit condition	The Employee entered properly authenticated credentials
Security requirements	All login credentials are hashed and stored server-side, allowing for a higher degree of information security.

Use case name	InvalidLogin
Participating actors	Initiated by System after invalid credential input by Employee or Supervisor**.**
Flow of events	1. User supplies invalid credentials to the login interface. 2. **System handles the input, returning a user-specific error in a pop-up message/dialog box. A button is made available once control is passed back to the user.** 3. The user must acknowledge the button in the dialog/box in order to proceed. 4. **System returns the user to the login page, where the user is then able to try to enter valid credentials once more.**
Entry condition	A user enters invalid credentials.
Exit condition	The user acknowledges the invalid entry.
Security requirements	The password must be hashed at all times it is handled by InvalidLogin. The dialogue boxes that handle username and password must be shielded against code execution and SQL injections. Password policy must be used to eliminate malicious input. Windows shortcut-key exploits must be disabled to avoid accessing a shell or forcing an exploit. Only <ENTER> will be recognized for acknowledgement of the message/dialogue box.

Use case name	InvalidRequest
Participating actors	Initiated by Employee, handled by System
Flow of events	1. Employee is logged in and authorized. Employee enters one or more invalid parameters to the RequestTimeOff GUI interface or time/date text box (such as inappropriate or fabricated dates). **2. System performs preliminary comparisons to avoid an exception. It then provides user-appropriate error message in the form of a dialogue box.** 3. **Employee** must acknowledge the error by responding to the dialogue box. **4. System returns the user control once restoring the RequestTimeOff GUI page**.
Entry condition	**Employee** provides one or more invalid parameters to the calendar GUI or time/date text box.
Exit condition	**Employee** acknowledges the error from system.
Security requirements	The dialogue boxes that handle time and date must be shielded against code execution and SQL injections. Windows shortcut-key exploits must be disabled to avoid accessing a shell or forcing an exploit. Only <ENTER> will be recognized for acknowledgement of the message/dialogue box.

Use case name	Logout
Participating actors	Initiated by Employee**,** handled by **System**
Flow of events	1. Employee presses the logout button on the interface. 2. **System** closes connections between **Employee** and the ESS and returns **Employee** to the login interface.
Entry condition	**Employee** is logged in to the ESS system.
Exit condition	**Employee** is logged out and returned to the login interface.
Security requirements	Resources allocated to the session must be terminated properly to ensure there are no bugs in the software.

Use case name	TimeOffResponse
Participating actors	Initiated by Supervisor
Flow of events	1. ESS displays a queued notification alerting the Supervisor of the pending time off request. 2. Supervisor selects the appropriate request from their ESS interface and clicks either Approve or Deny. 3. ESS sends the resulting response to the originating Employee.
Entry condition	The Supervisor logs into their ESS account
Exit condition	The Supervisor submits a TimeOffResponse approval, OR the Supervisor submits a TimeOffResponse denial.
Security requirements	All responses are tracked by EmployeeID ensuring that no unauthorized individuals are able to surreptitiously gain access to a request.

Use case name	RequestResponse
Participating actors	Initiated by Supervisor
Flow of events	1. Supervisor interfaces with the Supervisor Dashboard. User must navigate scroll window that implements Employee request queue. User selects an element of the list, and chooses to Approve or Deny. 2. ESS launches ApproveRequest or DenyRequest based on user choice.
Entry condition	Supervisor is logged in and at the appropriate default screen.
Exit condition	Supervisor either
Security requirements	

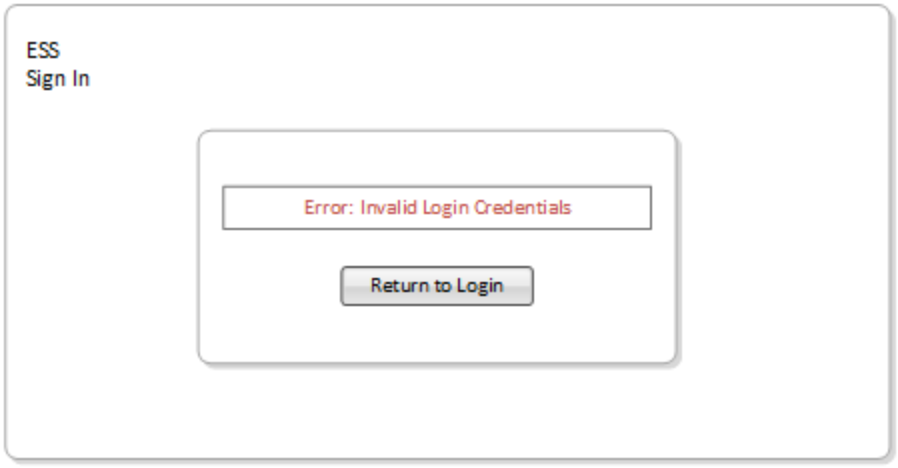
Use case name	ApproveRequest
Participating actors	Initiated by Supervisor only through RequestResponse
Flow of events	1. Supervisor clicks Approve button on Supervisor Dashboard. **2. ESS handles execution of ApproveRequest, updating relevant database entries before returning control with notification.** 3. Supervisor acknowledges system response.
Entry condition	Supervisor clicked Approve on a highlighted Employee request.
Exit condition	ESS executes database update and returns confirmation.
Security requirements	

Use case name	DenyRequest
Participating actors	Initiated by Supervisor only through RequestResponse
Flow of events	1. Supervisor clicks Deny button on Supervisor Dashboard. **2. ESS handles execution of DenyRequest, updating relevant database entries before returning control with notification.** 3. Supervisor acknowledges system response.
Entry condition	Supervisor clicked Deny on a highlighted Employee request.
Exit condition	ESS executes database update and returns confirmation.
Security requirements	

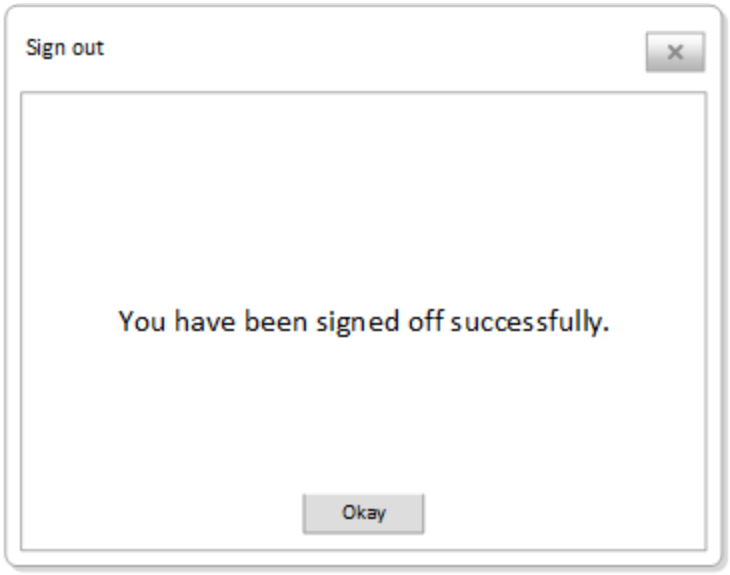
3 User Interface Mockups

3.1 Login

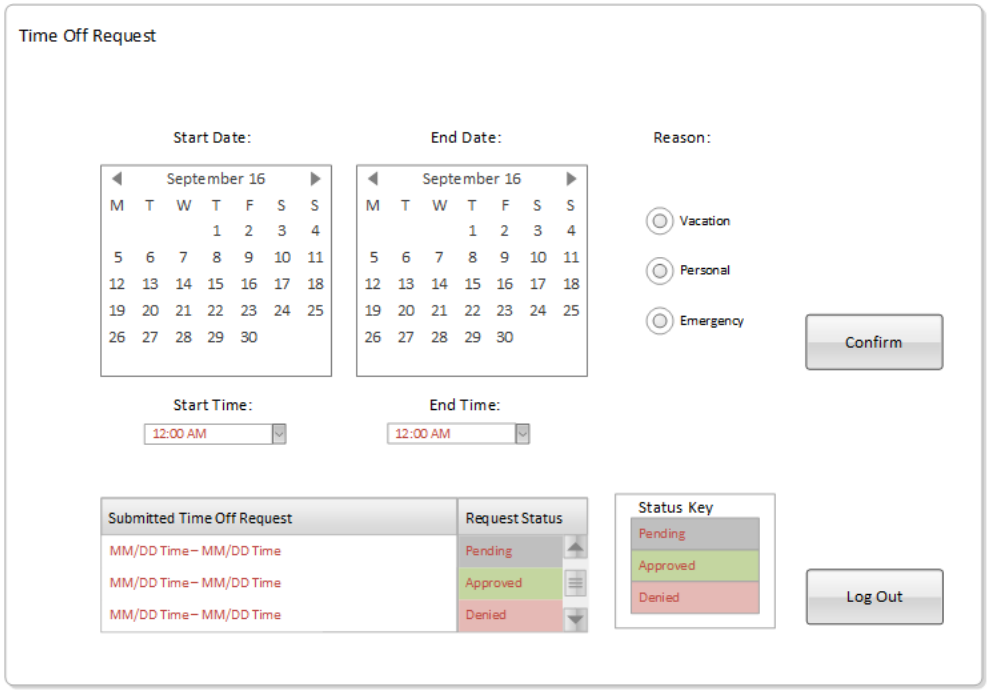
3.2 InvalidLogin



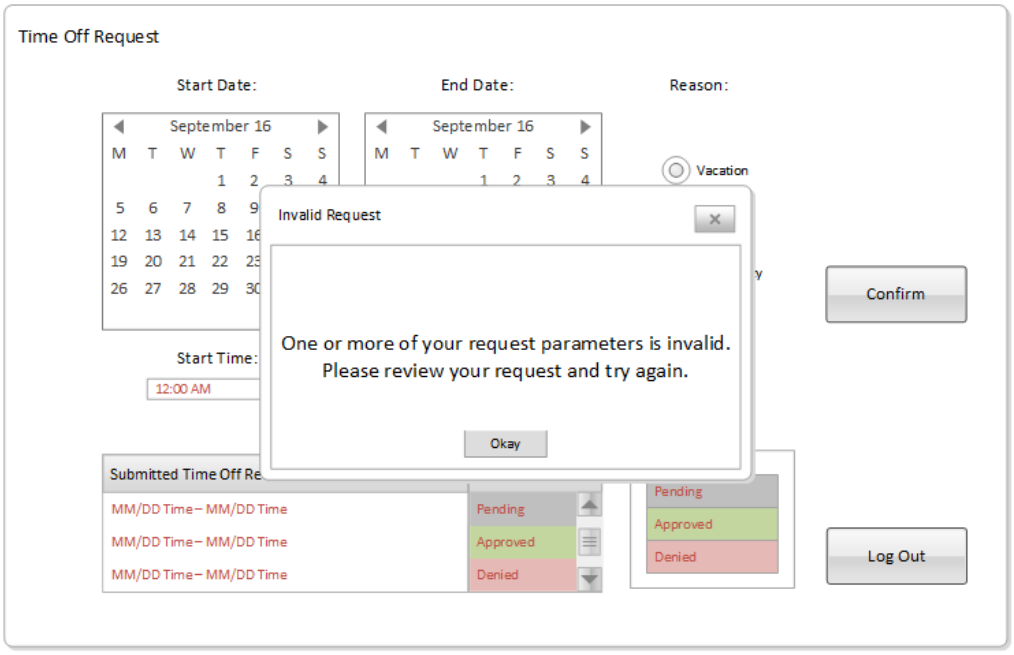
3.3 LOGOUT



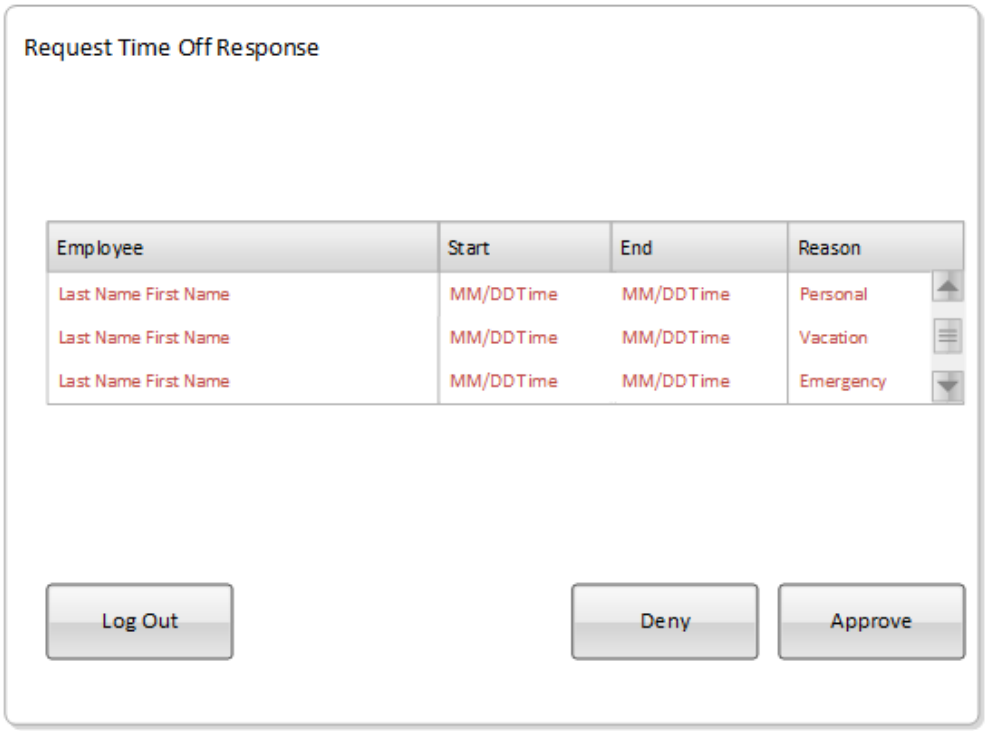
3.4 REQUESTTIMEOFF



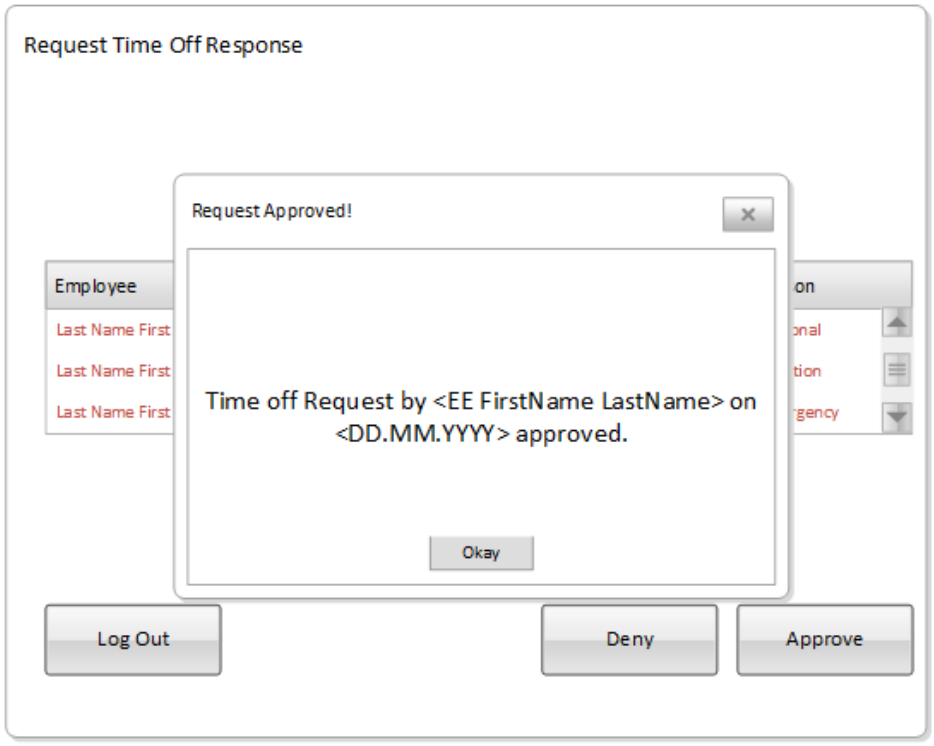
3.5 INVALID REQUEST



3.7 REQUESTRESPONSE



3.6 APPROVEREQUEST



3.8 DENYREQUEST