

CMSC - 495

GROUP – Team Screenshot

Project Plan & Technical Specifications Document

Project – Screenshot Capturing System

Instructor: Shanna Kuchenbecker

Group Members

Ronald Diggs
Evan Troutt
Jorge NavedoCabrera
Ravinder Miani

Date: 11/01/2021

Contents

1	Software Project Plan	5
1.1	Introduction	5
1.1.1	Project Overview	5
1.1.2	Project Deliverables	5
1.1.3	Evolution of the SPMP	6
1.1.4	Reference Materials	6
1.2	Project Organization	6
1.2.1	Organizational Structure	6
1.2.2	Project Responsibilities	6
1.3	Managerial Process	7
1.3.1	Assumptions, Dependencies, and Constraints	7
1.3.2	Staffing Plan	7
1.4	Technical Process	7
1.4.1	Methods, Tools, and Techniques	7
1.4.2	Software Documentation Work Breakdown Structure	7
1.4.3	Development Milestone Schedule	10
2	Software Requirement Specifications	12
2.1	Introduction	12
2.1.1	Purpose	12
2.1.2	Document Conventions	12
2.1.3	Intended Audience and Reading Suggestions	12
2.1.4	Product Scope	12
2.1.5	References	12
2.2	Overall Description	13
2.2.1	Product Perspective	13
2.2.2	Product Functions	13
2.2.3	User Classes and Characteristics	13
2.2.4	Operating Environment	13
2.2.5	Design and Implementation Constraints	13

2.2.6	User Documentation	13
2.2.7	Assumptions and Dependencies	14
2.3	External Interface Requirements	14
2.3.1	User Interfaces	14
2.3.2	Hardware Interfaces	14
2.3.3	Software Interfaces	14
2.3.4	Communications Interfaces	15
2.4	Functional Requirements	15
2.4.1	Front end	15
2.4.2	Database	16
2.5	Other Nonfunctional Requirements	17
2.5.1	Performance Requirements	17
2.5.2	Safety Requirements	17
2.5.3	Security Requirements	17
3	Business Requirement Specifications	19
3.1	Introduction	19
3.1.1	Purpose & Scope of the document	19
3.1.2	Intended Audience	19
3.1.3	Definitions and Acronyms	19
3.1.4	Reference	19
3.2	Proposed System	20
3.2.1	Screenshot Capturing System Scope	20
3.2.2	Assumptions	20
3.2.3	Proposed Process	20
3.3	Screenshot Capturing System Business Requirements	21
3.3.1	Business Requirements for the Screenshot Capturing System	21
4	Change Log	23

Software Project Plan

1 Software Project Plan

1.1 Introduction

This plan is for the development of the Screenshot Capturing System project by a group consisting of Ronald Diggs, Ravinder Maini, Evan Troutt and Jorge NavedoCabrera.

The name of the game is accountability. In a virtual world things are constantly changing, this is especially true for websites. A person can list a product on their site at one price one day and it can change another. This can also be said about sites that were once trusted but ended up getting compromised due to a vulnerability. We want to audit to ensure that websites are doing right by their customers. We plan on doing this by implementing a screenshot capturing system that will periodically capture web pages thus helping customers of those sites to legally fight any discrepancies.

The Screenshot Capturing System project involves developing a software product to allow users to view, edit and save Website contents. In addition, it will allow users to use this data for website accountability and audit purposes.

1.1.1 Project Overview

The purpose of the Screenshot Capturing System project is to develop a program that will implement multiple features to see success at the end of the semester. These features include the ability to select which websites we are capturing screenshots for, how often we are capturing screenshots, sending HTML based emails to multiple users, add/edit/delete users, error logging, and maintenance of history and archives. The project will be built as a team using java with GitHub as a repository for source control. Microsoft Teams will be our main source of communication with the idea to be able to integrate Asana and GitHub, as well as being able to share any other documents that we may need.

1.1.2 Project Deliverables

The project deliverables for the Help Desk Ticket Tracker Project include:

- Overview – Week 1
- Project Plan- Week 2
- Requirements Specification- Week 2
- System Specification- Week 2

- User's Guide- Week 3
- Test Plan and Results- Week 3
- Design and Alternate Designs- Week 4
- Phase 1 Development & Report- Week 5
- Phase 2 Development & Report- Week 6
- Phase 3 Development & Report- Week 7
- Review program & code- Week 7
- Final Report- Week 8

These deliverables will each be delivered according to their above due dates. The completed project will be submitted within 8 weeks after the project group's formation.

1.1.3 Evolution of the SPMP

All changes to the project management plan must be agreed to by the group before they are implemented. All changes should be documented in order to keep the project management plan correct and up to date.

1.1.4 Reference Materials

Reference Documents	Document Name
Screenshot_Capture_ProjectPlan_Specification_V1.docx	Screenshot Capturing System Project Plan & Requirements Specifications

1.2 Project Organization

1.1.5 Organizational Structure

The development team for this project is Ronald Diggs, Ravinder Maini, Evan Troutt and Jorge NavedoCabrera

1.1.6 Project Responsibilities

Responsibilities for the Helpdesk Ticket Tracker Project will be as follows:

- Ronald Diggs will be project manager for the team.
- Evan Troutt will lead the programming efforts for this project and coordinate all programming tasks.
- Jorge NavedoCabrera will lead and coordinate the QA testing for this project.
- Ravinder Maini will lead for this project design and development.

1.3 Managerial Process

1.1.7 Assumptions, Dependencies, and Constraints

- Assumptions
 - Development will be started before all documentation is completed in order to maximize development time.
 - Testing will be started prior to development completion in order to maximize QA Testing and documentation time.
 - The User Guide will be completed prior to development completion as soon as the user interface nears finalization.

1.1.8 Staffing Plan

For the Screenshot Capturing System Project, all members of the project team will be needed for all 8 weeks of the project. Each team members will lead in their respective area but will assist in other project tasks as needed to complete that week's deliverables.

1.4 Technical Process

1.1.9 Methods, Tools, and Techniques

The methods, tool, and techniques utilized by the project team to develop the Screenshot Capturing System project will include:

- Methods
 - Agile Design Methodology
- Tools
 - Operating System- Ubuntu 20.04
 - Development Environment- IntelliJ IDEA
 - Programming Language- Java with Spring, XML, HTML/CSS
 - Code Version Control- GitHub
 - Debugger- IntelliJ IDEA debugger
 - SQL- MongoDB
- Techniques
 - Object Oriented Design
 - Model View Controller

1.1.10 Software Documentation Work Breakdown Structure

Below is the Work Breakdown Structure for the Screenshot Capturing System Project:

1. Overview
 - 1.1. Define project statement and definition
 - 1.2. Identify the team members for the project
 - 1.3. Define scope
 - 1.4. Define team member responsibilities
 - 1.5. Identify team leader/project manager
2. Develop Project Documentation
 - 2.1. Develop Project Plan
 - 2.1.1. Establish project schedule
 - 2.1.2. Define deliverables
 - 2.1.3. Write Project Plan
 - 2.2. Develop Requirements Specification
 - 2.2.1. Define goals of the product
 - 2.2.2. Define complete scope of product
 - 2.2.3. Determine how the scope will be implemented
 - 2.2.4. Perform preliminary design analysis
 - 2.2.5. Write Requirements Specification
 - 2.3. Develop System Specification
 - 2.3.1. Identify methods that will be used to develop the product
 - 2.3.2. Identify techniques that will be used to develop the product
 - 2.3.3. Identify tools that will be used to develop the product
 - 2.3.4. Identify system requirements for the product
 - 2.3.5. Write System Specification
 - 2.4. Develop User's Guide
 - 2.4.1. Review functionality
 - 2.4.2. Review functional process flows
 - 2.4.3. Write User's Guide
 - 2.5. Develop Test Plan
 - 2.5.1. Review Requirements Specification
 - 2.5.2. Review User's Guide
 - 2.5.3. Determine scenarios
 - 2.5.3.1. Determine Inputs/steps
 - 2.5.3.2. Determine Outputs/expected results
 - 2.5.4. Write Test Plan
 - 2.6. Document Test Results
 - 2.6.1. Review Test Plan
 - 2.6.2. Test product according to Test Plan
 - 2.6.3. Compare testing results to Test Plan
 - 2.6.4. Document Test Results
 - 2.7. Develop Design Documentation
 - 2.7.1. Review Requirements Documentation
 - 2.7.2. Review System Specification
 - 2.7.3. Determine design approach
 - 2.7.3.1. Determine classes & methods required

- 2.7.3.1.1. Identify fields required
 - 2.7.3.1.2. Identify data structures to be used
 - 2.7.3.1.3. Identify interfaces required
 - 2.7.3.2. Determine Database Tables/Structure required
 - 2.7.3.3. Determine File Structure required
 - 2.7.3.4. Create Database Diagram
 - 2.7.3.5. Create UML Diagrams
- 2.7.4. Develop design documentation
- 2.8. Develop Phase 1 Report
 - 2.8.1. Review team member activities
 - 2.8.2. Determine project completion status
 - 2.8.2.1. Review project milestones
 - 2.8.2.2. Determine current project completion
 - 2.8.2.3. Compare current progress to project milestones
 - 2.8.3. Review decisions made for the project
 - 2.8.4. Review issues from the week
 - 2.8.5. Review changes made to project documents
 - 2.8.6. Develop Phase 1 Development Report
- 2.9. Develop Phase 2 Report
 - 2.9.1. Review team member activities
 - 2.9.2. Determine project completion status
 - 2.9.2.1. Review project milestones
 - 2.9.2.2. Determine current project completion
 - 2.9.2.3. Compare current progress to project milestones
 - 2.9.3. Review decisions made for the project
 - 2.9.4. Review issues from the week
 - 2.9.5. Review changes made to project documents
 - 2.9.6. Develop Phase 1 Development Report
- 2.10. Develop Phase 3 Report
 - 2.10.1. Review team member activities
 - 2.10.2. Determine project completion status
 - 2.10.2.1. Review project milestones
 - 2.10.2.2. Determine current project completion
 - 2.10.2.3. Compare current progress to project milestones
 - 2.10.3. Review decisions made for the project
 - 2.10.4. Review issues from the week
 - 2.10.5. Review changes made to project documents
 - 2.10.6. Develop Phase 1 Development Report
- 2.11. Develop Final Report
 - 2.11.1. Review all other project documents
 - 2.11.2. Document development history
 - 2.11.3. Document lessons learned
 - 2.11.4. Review product design
 - 2.11.4.1. Determine product design strengths
 - 2.11.4.2. Determine product design limitations

- 2.11.5. Determine ways the project could be improved
 - 2.11.5.1. Review lessons learned
 - 2.11.5.2. Review project documents
 - 2.11.5.3. Review product decisions
 - 2.11.5.4. Review product design strengths
 - 2.11.5.5. Review product design limitations
- 2.11.6. Develop Final Report

1.1.11 Development Milestone Schedule

Below table documenting the planned development milestone schedule for the Screenshot Capturing System.

Milestone	Primary Developers	Task Begin Date	Completed Date Goal
Configure Multiple Websites for whom capture screenshots	Evan Troutt	November 16th	November 22 nd
Implement Send HTML based email to multiple users Functionality	Evan Troutt	November 18 th	November 23 rd
Implement Screen Captured Email View	Evan Troutt	November 19 th	November 23 rd
Implement Periodically capture the screenshots Functionality	Evan Troutt	November 20 th	November 24 th
Implement Add/Edit users View	Evan Troutt	November 21 th	November 25 th
Implement Add/Edit Multiple websites View	Evan Troutt	November 22 th	November 26 th
Implement Log errors and exceptions View	Evan Troutt	November 24 th	November 30 th
Implement Screenshot History Functionality	Evan Troutt	November 25 th	November 30 th
Implement History View	Evan Troutt	November 30 th	December 6 th

Software Requirement Specification

2.1 Introduction

1.1.12 Purpose

The purpose of the Screenshot Capturing System project is to develop a program that will implement multiple features to see success at the end of the semester. These features include the ability to select which websites we are capturing screenshots for, how often we are capturing screenshots, sending HTML based emails to multiple users, add/edit/delete users, error logging, and maintenance of history and archives.

1.1.13 Document Conventions

The requirements specifications follow the IEEE standard 830-1998 format for defining software requirements. Sections of the document are intended to provide the full scope of the Screenshot Capturing System. No escalated prioritization was given to any of the requirement.

1.1.14 Intended Audience and Reading Suggestions

The intended audience for the Software Requirements Specification (SRS) document includes the project manager, customer, development team, and end-users. The SRS document details the internal and external interfaces, outlines design considerations, defines user classes, provides entity relationship modeling, and hardware and software requirements. The document sections are organized to capture the workflow of the overall system, and then define more formalized requirements.

1.1.15 Product Scope

The Screenshot Capturing System software will be used by any customer to assist in capturing the screenshots of the given websites and maintain the history of the screenshot capture. This way it help the customer to view and compare the website current and previous content for the accountability and audit purposes.

1.1.16 References

[1] IEEE Software Engineering Standards Committee, "*IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications*", October 20, 1998.

2.2 Overall Description

1.1.17 Product Perspective

The Screenshot Capturing System is a new, self-contained program that will get deployed to user workstations and will interface with a websites to capture the screenshot periodically and store information on user workstations. Product users will be able to manage screenshots and websites configuration by other team members.

1.1.18 Product Functions

All of the Screenshot Capturing System major functions will be performed by user personnel. These functions include:

Configure Multiple Websites for whom capture screenshots

- Allow users to Periodically capture the screenshots Functionality
- Allow users to Send HTML based email to multiple users Functionality
- Allow users to Add/Edit users View
- Allow users to Add/Edit websites View
- Allow users to Log errors and exceptions View
- Allow users to view History

1.1.19 User Classes and Characteristics

The Screenshot Capturing System will support users at inception, but future releases will support multiple levels of users. Users will be able to enter new website, make updates to existing website, filter open website lists, reassign website, and complete screenshot capture workflows.

1.1.20 Operating Environment

The environment in which the Screenshot Capturing System Software will operate will be Windows 10, Mac, or Linux operating systems.

1.1.21 Design and Implementation Constraints

Access to the Screenshot Capturing System will only be available on computer systems on which the application has been installed. All instances of the application will have the ability to communicate with the website server through internet connectivity. The screenshot and email shall be regularly backed up and make history. The application shall be compatible with all versions of Windows and iOS, and shall be created utilizing Java or .Net technologies.

1.1.22 User Documentation

The Screenshot Capturing System development team shall create a user guide for the application. The user guide will be made available at the time the application is ready for user acceptance testing. Other documentation that shall be completed by the development team include: system requirements specification, system specification, project plan, test plan, design alternatives, and incremental development reports.

1.1.23 Assumptions and Dependencies

Complete functionality is dependent on internet connectivity to facilitate the communication between the Screenshot Capturing System application and the website. The workflow processing is dependent on proper functionality of users, hardware, and software. Proper user training is required for all personnel to ensure data and systems integrity. All hardware must meet the minimum requirements that ensure its compatibility with the software. The integrity and availability of the hardware must be ensured through the development lifecycle of the Screenshot Capturing System application. The hardware systems fail, replacements must be available to ensure operations of the corporation. The software should be written in such a way as to ensure operability in future hardware systems

External Interface Requirements

2.3 External Interface Requirements

1.1.24 User Interfaces

Desktop application will have GUI based user interface with main pages to configure to Add/Edit users View, to Add/Edit Multiple websites View and to Log errors and exceptions View, to view History options.

1.1.25 Hardware Interfaces

There is no hardware interface requirement for the Screenshot Capturing System Software.

1.1.26 Software Interfaces

Software Requirements

Client Operating Systems:

MAC
Windows
Linux

Client Application:

Java
Any web browser
Chrome (needs to be installed on computer for automated browser functionality)

Server Application:

- Java with JRE 11 or above
- MongoDB
- Maven
- Spring Boot

1.1.27 Communications Interfaces

Network software and protocols in order for systems to communicate:

TCP/IP
HTTP
SMTP

2.4 Functional Requirements

Requirements of Screenshot Capturing System are divided into sub functions: Front end, Backup end and NoSQL Configuration.

1.1.28 Front end

2.4.1.1 Front end of Screenshot Capturing System shall allow user to Add/Edit/View Website

- GUI shall include Website ID, Description, URL, Notes, CreatedOn, UpdatedOn
- Allow user to view Website
- Allow user to add new Website
- Allow user to edit existing Website

2.4.1.2 Front end of Screenshot Capturing System shall allow user to view Log errors and exceptions

- GUI shall include Log ID, Log Message, CreatedOn

- Allow user to view Log errors and exceptions

2.4.1.3 Front end of Screenshot Capturing System shall allow user to view History

- GUI shall include ID, Screenshot Message, CreatedOn
- Allow user to view History

1.1.29 Database

MongoDB will be used to handle the data storage.

2.4.1.4 Data integrity

Database will help in performing Commit operation that are completed and/or rollback unfinished or unresolved configuration.

2.4.1.5 Data validation

Data error from the user's end and from the back-end database-processing end must be gracefully handled. There will be data validation and error-handling routines as part of the handling system.

2.4.1.6 Performance

Must resolve locking issues and handle concurrent use of the system on a 24x7 basis. Send, receive and display user messages to assist the over-all user experience. In order to increase the performance, user needs to do software testing, very often.

2.4.1.7 Data repository

The backend of the data will maintain with directory structure as the main repository backend of the data. System Requirement Specifications

2.5 Other Nonfunctional Requirements

1.1.30 Performance Requirements

Users need to have proper access to the application. User will successfully able to perform all the functionalities including create and save in the database.

1.1.31 Safety Requirements

There are no safety requirements for the Screenshot Capturing System Software.

1.1.32 Security Requirements

User will need proper authentication and authorization to access the application.

Business Requirement Specification

3.1 Introduction

The Screenshot project team will be implementing the Screenshot Capturing System project to include features the ability to select which websites we are capturing screenshots for, how often we are capturing screenshots, sending HTML based emails to multiple users, add/edit/delete users, error logging, and maintenance of history and archives.

1.1.33 Purpose & Scope of the document

The purpose of the document is to detail the requirements and functionality of the new Screenshot Capturing System.

1.1.34 Intended Audience

This document is intended to provide the business and functional requirements for users, software developers, QA testers, product trainers and any other involved parties.

1.1.35 Definitions and Acronyms

Acronym / Term	Definition

1.1.36 Reference

Reference Documents	Document Name
Screenshot_Capture_ProjectPlan_Specification_V1.doc x	Screenshot Capturing System Project Plan & Requirements Specifications

3.2 Proposed System

1.1.37 Screenshot Capturing System Scope

The proposed functionality for the Screenshot Capturing System includes:

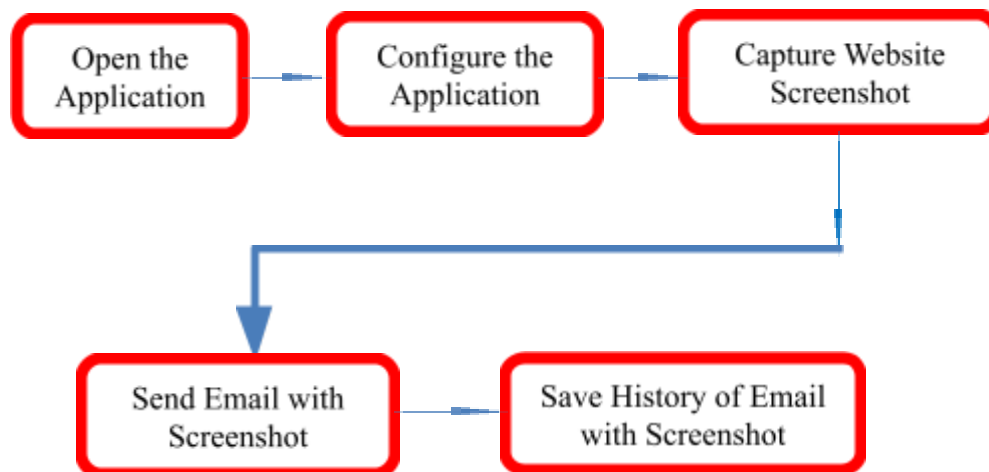
- System will Periodically capture the screenshots
- System will Send HTML based email to multiple email users
- Users will Add/Edit/View Email users
- Users will Add/Edit/View websites
- System will Log errors and exceptions
- Users will view History

1.1.38 Assumptions

- Users will only have access to the Screenshot Capturing System program if it is installed on their machine.
- Users will have the ability to save data locally.

1.1.39 Proposed Process

The following diagram depicts the proposed process for a user utilizing the Screenshot Capturing System while connected to the website:



3.3 Screenshot Capturing System Business Requirements

1.1.40 Business Requirements for the Screenshot Capturing System

The following requirements apply to the design of the Screenshot Capturing System program and process.

ID #	Business Requirements	Business Rules / Requirements Definition	Comments
BRS-	The system will allow Users to Add/Edit/View websites	<ul style="list-style-type: none">A. System will open the Website Configuration GUIB. Add New Website<ul style="list-style-type: none">- Insert Website ID, Description, URL, Notes, CreatedOn, UpdatedOnC. Existing Website<ul style="list-style-type: none">- Update Description, URL, Notes, CreatedOn, UpdatedOnD. View Website<ul style="list-style-type: none">- View Website ID, Description, URL, Notes, CreatedOn, UpdatedOnE. Log Error and Exception if any occurs	
BRS-	System allow user to view Log errors and exceptions	<ul style="list-style-type: none">A. System will open the View Log errors and Exception SettingB. View Log Errors<ul style="list-style-type: none">- View Log ID, Log Message, CreatedOn settingsC. Log Error and Exception if any occurs	
BRS-	System allow user to view History	<ul style="list-style-type: none">A. System will open the View HistoryB. View History<ul style="list-style-type: none">- View ID, Screenshot Message, CreatedOnC. Log Error and Exception if any occurs	

4 Change Log

Date	Version	Description of Change	Item(s) Changed	Requested By
11/01/2021	V0	Created the baseline and Initial draft	Baseline and Initial Draft	Group
11/01/2021	V1	Modified the format and added the cover page for uniformity	Format Change and Added Cover page	Group
12/14/21	V2	Changes to specifications and technical requirements in line with how the project changed	Software Project Plan, Software Technical Specification, Business Requirement Specification	Evan Troutt