Software Requirements Specification

for

Cybersecurity Scenario Generator

**Version 1.0 approved**

**Prepared by**

**Jared Diosdado,**

**Michael Hoover,**

**Stephanie Molina,**

**Zachary Jones**

**10/02/24**

**Table of Contents**

[**1. Project Summary**](#_340pe1hy7etv) **1**

[**2. Points of Contact**](#_ge99yjm2q5hh) **2**

[**3. Introduction**](#_1fob9te) **3**

[3.1 Purpose](#_3znysh7) 3

[3.2 Document Conventions](#) 3

[3.3 Intended Audience and Reading Suggestions](#) 3

[3.4 Product Scope](#) 3

[3.5 References](#) 3

[**4. Overall Description**](#) **4**

[4.1 Product Perspective](#_2s8eyo1) 4

[4.2 Product Functions](#) 4

[4.3 User Classes and Characteristics](#) 4

[4.4 Operating Environment](#) 4

[4.5 Design and Implementation Constraints](#) 4

[4.6 User Documentation](#) 4

[4.7 Assumptions and Dependencies](#) 5

[**5. External Interface Requirements**](#) **5**

[5.1 User Interfaces](#_2jxsxqh) 5

[5.2 Hardware Interfaces](#) 5

[5.3 Software Interfaces](#) 5

[5.4 Communications Interfaces](#) 5

[**6. System Features**](#) **6**

[6.1 System Feature 1](#) 6

[6.2 System Feature 2 (and so on)](#) 6

[**7. Other Nonfunctional Requirements**](#_3whwml4) **6**

[7.1 Performance Requirements](#_2bn6wsx) 6

[7.2 Safety Requirements](#) 7

[7.3 Security Requirements](#) 7

[7.4 Software Quality Attributes](#) 7

[7.5 Business Rules](#) 7

[**8. Other Requirements**](#) **7**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# Project Summary

The Cybersecurity scenario generator project will involve the development of a customizable scenario generator for both high school and middle school students. This will enable the creation of interactive investigative scenarios, where students must engage in solving mysteries by analyzing clues that are embedded in digital artifacts. With inputs from the teacher such as the school mascot, a location, and a rival team, the project would automate the generation of scenario specific learning materials. This would include the generation of a PowerPoint presentation, disk images with digital clues, and a quiz. This would enhance student engagement through hands-on problem-solving activities. The project’s aim is to save teachers time by automating content creation, allowing them to focus on tailoring the materials to their specific teaching goals. A user-friendly interface will be provided to accommodate teachers with varying levels of technical expertise. The success of this project will be measured by its ease of use and the positive feedback from educators who implement it in their classrooms.

# 

# Points of Contact

# Sponsor, David Hochstetler, [jacob.hochstetler@unt.edu](mailto:jacob.hochstetler@unt.edu)

# Project Manager, Michael Hoover, [michaelhoover@my.unt.edu](mailto:michaelhoover@my.unt.edu)

# Project Scribe, Stephanie Molina, [stephaniemolina@my.unt.edu](mailto:stephaniemolina@my.unt.edu)

# Project Developer Jared Diosdado, [jareddiosdado@my.unt.edu](mailto:jareddiosdado@my.unt.edu)

# Project Developer, Zachary Jones, [zacharyjones4@my.unt.edu](mailto:zacharyjones4@my.unt.edu)

# Points of contact are either through email and the project’s discord server where the team has access to real-time instant messaging with the sponsor and the group members.

# 

# 

# Introduction

## Purpose

This Software Requirements Specification defines the requirements for version 1.0 of the Cybersecurity Scenario Generator. It covers the entire system, including the automated generation of learning materials (presentations, disk images, quizzes) based on teacher input. The project aims to save educators time while allowing them to customize materials to their specific curriculum. The scope of this SRS includes functional and non-functional requirements for the software, ensuring that it can be easily used by educators with varying levels of technical expertise.

## Intended Audience and Reading Suggestions

This document is intended for developers, project managers and testers. Developers and testers should pay close attention to the functional and technical requirements sections, while project managers and educational technologists will find value in the overview and use case sections. Educators may wish to read the product scope and user interaction sections to understand how the tool will help them in the classroom. It is recommended to begin with the product overview before proceeding to the detailed requirements and use cases.

## Product Scope

The Cybersecurity Scenario Generator is a tool designed to automate the creation of interactive learning materials for teaching cybersecurity. The product will generate PowerPoint presentations, disk images containing digital clues, and quizzes based on teacher inputs such as school-specific details. The tool's primary benefit is to save educators time in material preparation while providing engaging, hands-on cybersecurity challenges for students. By automating these tasks, the project aligns with the broader goal of integrating cybersecurity education into school curriculum and enhancing student learning through interactive experiences. The success of the product will be measured by educator feedback and the effectiveness of the generated materials in engaging students.

# Overall Description

## Product Perspective

The Cybersecurity Scenario Generator is a new, self-contained product designed to automate the creation of cybersecurity learning materials. It is not part of an existing product family but is designed to integrate easily into classrooms, helping educators generate content for interactive investigative scenarios. It operates independently, generating PowerPoint presentations, disk images, and quizzes based on teacher inputs.

## Product Functions

* Collecting teacher input (school mascot, location, rival team, etc.) through a user-friendly interface.
* Automatically generating PowerPoint presentations tailored to the input.
* Using disk images containing digital clues for students to investigate.
* Developing quizzes to assess students' understanding of the scenarios.
* Allowing educators to customize scenarios for specific teaching goals.

## User Classes and Characteristics

One main user class, that is teachers/users. They will solely have access to the actual web application that will prompt them for questions. They will have no authentication or authorization to do any other action but solely respond to the form.

## Operating Environment

The environment will operate solely in a web browser that will handle sending HTTP requests to and from the python web server that will be hosted separately on a domain that is to be determined.

## User Documentation

## There will be no user documentation with the product though any concerns will be correctly signposted on the website

## Design and Implementation Constraints

The product will ensure input validation. This means that the user will not be able to input any profanities. The product will flag the user to input something different. Make sure to consider that any confidential information is not stored or held on the site and is discouraged from being inputted.

# External Interface Requirements

## User Interfaces

The UI should be sleek and minimalist, will feature very little on the page outside of a single form input and prompt, along with a next button that will prompt the user to respond to the prompt.

## Hardware Interfaces

Available hardware interfaces for the websites include any sort of laptop, desktop, or mobile device that is able to access a web browser capable of executing javascript, downloading files, and opening powerpoints.

## Software Interfaces

There will be a variety of software interfaces used. The web server will be coded in python and will use the Flask library for hosting the server and handling HTTP requests. It will generate powerpoints using the Python PPTX library. Users of the application will have to access the actual web page using a web browser of their choice that can execute javascript. Finally, users will need to be able to open and view powerpoint files in order to use the software.

## Communications Interfaces

The assignment will use HTTP for sending data to and from the python web server used to generate the powerpoint

# System Features

## Python Web Server for Generating Powerpoint

4.1.1 Description and Priority

This is a python web server using the Flask framework to send and receive HTTP requests. When it receives the correct request from the actual scenario generator web application it will generate a powerpoint using the Python PPTX library for powerpoint generation. **10/10**

4.1.2 Stimulus/Response Sequences

User access the web application and fills out a form response. That response sends an HTTP POST request to the web server that responds accordingly with generating slides of or the powerpoint

## Scenario Generator Web Application

4.1.1 Description and Priority

This is the front end web application coded using HTML, CSS, and Javascript. This website will have a sleek, minimalist UI with a very simple form input front and center to the user that prompts the user with a series of mad-libs type questions that will be used in the actual story of the powerpoint. **7/10**

4.1.2 Stimulus/Response Sequences

User access the web application and is prompted with the first form question, they will simply just enter a response to the form and click next, where they will be asked a currently to be determined amount of questions until the series of questions is done

* 1. **Quiz Generator**

4.1.1 Description and Priority

Following the creation of the powerpoint by the web server, the python web server will also send an interactive quiz either within the powerpoint or as a separate download (To be determined) **3/10**

4.1.2 Stimulus/Response Sequences

User accesses the web application and fills out the series of form questions as clarified above. Following the generation of the powerpoint the python application will also

# Other Nonfunctional Requirements

## Performance Requirements

Must have a working internet connection. If there is no internet, the product will not load and work.

## Safety Requirements

The product will make sure to input validation. This means that the user will not be able to input any profanities. The product will flag the user to input something different.

## Security Requirements

The product will not have any user authentication that will store information. The product will state to the user to not type any important information about the user for security reasons.

## Software Quality Attributes

The project should always be available, portable, and reusable. There are no constraints on the amount of times a powerpoint can be generated, and as a web application should be accessible from any device capable of running a web browser.

Should have a focus on ease of use, with clearly labeled form inputs and clear security constraints labeled on the website

## Business Rules

The product will allow anyone to use it, but the main purpose of the product is for Teachers to perform the functions of the product to the students.

# Other Requirements

There are no other requirements for this document at this time.