Project Plan

Document Number: SCP-Form02

Date: September 10, 2023

Spectrum Analyzer Project

Masood Afzali

Ashly Altman

Brooke Ebetino

Tyler Haley

Joey Thompson

402 SWEG, Robins AFB

Table of Contents

1. INTRODUCTION

This Project Plan describes the scope, context, stakeholders, activities, resources, schedule, constraints, risks, quality goals, and processes for developing a video-based spectrum analyzer application. It will provide a comprehensive guide for how the development work will be managed throughout the four-month project timeline.

This project aims to develop a cross-platform software application that can perform frequency and amplitude analysis on input video files. The program will analyze a video of a screen of a spectrum analyzer from a local folder and write a descriptive CSV file containing a timestamp, center frequency, minimum amplitude, maximum amplitude, and center amplitude.

The project stakeholders include the 5 Kennesaw State University student developers, their course instructor, Professor Yan Huang, and Robins AFB, which will provide funding and act as the customer. The development team consists of:

- Team Lead: Joey Thompson
- Frontend/Packaging Lead: Ashly Altman
- Backend Lead: Brooke Ebetino
- Testing Lead: Tyler Haley
- Documentation Lead: Masood Afzali

The project will leverage an agile development process with 2-week sprints and daily standup meetings. The codebase will be maintained in a GitHub repository, and the team will utilize project management tools like Jira.

The application will be developed with no frontend interface and Python (NumPy, OpenCV, Matplotlib) on the backend for analysis and data processing. Unit testing and end-to-end testing will be used for quality assurance.

This document will detail the project background, scope, schedule, team roles, dependencies, risks, and all processes followed throughout the 4-month development cycle. The goal is to provide a comprehensive guide for the smooth and successful execution of the project.

2. PROJECT OVERVIEW

2.1 Scope

2.1.1 Identification

The system being developed is the Real-Time Video Spectrum Analyzer Application. It will be identified by the name SpecVidAnalyzer and Project ID SVAP2023.

The software components include:

SpecVidAnalyzer Backend - Python backend for data processing.
SpecVidAnalyzer Testing Suite - Testing code and scripts.
The initial release version of the complete application will be SpecVidAnalyzer v1.0.

2.1.2 System Overview

The SpecVidAnalyzer software application performs real-time frequency and amplitude analysis on video files. The application logs the analysis data to CSV files.

SpecVidAnalyzer is being developed by a team of 5 undergraduate software engineering students at Kennesaw State University as part of their senior project course. The project sponsor and acquirer are Robins Air Force Base. The intended end users are electronics engineers and researchers analyzing signals and systems in video footage.

The project development takes place over one academic semester from August to December 2023. The GitHub repository spectrumAnalyzerProject contains the source code and documentation. Other relevant documents include the requirements, specifications, and design documentation.

2.1.3 Document Overview

This Project Plan summarizes the development efforts for the SpecVidAnalyzer application. It outlines the scope, objectives, team roles, schedule, resources, risks, and processes. The document is intended for use by the development team as a guide throughout the project. No security or privacy considerations apply. Current System or Situation

2.2 Current System or Situation

There is currently no existing system that meets the goals and objectives of the SpecVidAnalyzer application. This project will develop the application from scratch.

2.3 Background, Objectives, and Scope

The background of this project is to develop real-time video spectrum analysis software for electronics engineering applications. The main objectives are to:

- Analyze input video and extract frequency and amplitude data
- Generate an interactive visualization of the spectrum analyzer output
- Log analysis data to CSV files for further examination
- Package the program to an executable file

The project scope includes fully developing the SpecVidAnalyzer application, including the SpecVidAnalyzer application software, testing, documentation, and deployment.

2.4 Operational Policies and Constraints

This project follows the standard operating policies and constraints for senior software engineering projects at Kennesaw State University. The key constraints include the 4-month development timeline and 170 lines of code

limit.

2.5 Description of Current System or Situation

2.5.1 Operational Environment and its Characteristics

The system operates in a controlled computational environment for professionals analyzing spectrum analysis data on standard desktop or laptop systems. Users primarily interact with the tool by moving videos into a folder and then running an executable file.

2.5.2 Major System Components and Interconnections

- OpenCV for image processing
- NumPy for numerical operations
- SciPy for curve fitting algorithms
- CSV for data export

These components interact sequentially, with data flowing primarily from OpenCV to SciPy to CSV.

2.5.3 Interfaces to External Systems or Procedures

The tool interfaces with external video files for input and outputs a CSV file containing detected signal characteristics.

2.5.4 Capabilities/Functions

- Video processing for identifying spectral areas and waves
- Color filtering to isolate areas of interest
- Wave characteristic extraction like center frequency and amplitude

2.5.5 Performance Characteristics

- Faster than real-time analysis depending on the processing capabilities of the host machine
- Single-threaded
- Capable of processing standard video formats

2.5.6 Quality Attributes

- Reliability: Highly dependent on the quality and consistency of the input video
- Maintainability: Modular code structure for ease of updates
- Usability: Requires a professional user familiar with spectrum analysis
- Efficiency: Capable of analyzing faster than real-time on most standard modern laptops.

2.5.7 Provisions for Safety, Security, Privacy, and Continuity of Operations:

Currently, the tool has no specific safety, security, or privacy provisions. However, as it runs locally, the risk is considered low. There is no mechanism for data backup or recovery.

2.6 Users or Involved Personnel

The users and personnel involved in this project are classified into different roles, each with distinct responsibilities and skill sets:

2.6.1 End Users

- Electronics engineers and researchers at Robins Air Force Base will use the system to analyze signals and systems in video footage.
    - Responsibilities: Provide user requirements, participate in UAT (User Acceptance Testing), and provide feedback for system improvements.
    - Interactions: Coordinate with Project Team and Stakeholders for requirements and testing.

2.6.2 Project Team

The Project Team comprises five undergraduate software engineering students from Kennesaw State University.

- Team Lead: Joey Thompson
    - Responsibilities: Overall project management, team coordination, and client liaison.
- Frontend and Packaging Lead: Ashly Altman
    - Responsibilities: Development and maintenance of the frontend application and packaging the application into an executable.
- Backend Lead: Brooke Ebetino
    - Responsibilities: Development and maintenance of the backend Python code.
- Testing Lead: Tyler Haley
    - Responsibilities: Creating and executing test plans and ensuring software quality.
- Documentation Lead: Masood Afzali
    - Responsibilities: Maintaining project documentation and user manuals.

2.6.3 Project Sponsor: Robins Air Force Base

Responsibilities: Project Customer.
Interactions: Regular project status updates, UAT, and final approval.

2.6.4 Course Instructor

- Professor Yan Huang
    - Responsibilities: Academic guidance, project assessment, and quality assurance.
    - Interactions: Periodic project reviews, providing feedback, and academic assessment.

2.6.5 Additional Stakeholders:

- IT and Maintenance Team at Robins Air Force Base
    - Responsibilities: Deployment, maintenance, and providing system infrastructure.

2.6.6 Training/Skills:

- The project team will require Python, NumPy, and OpenCV training.
- End-users may require training sessions for using the system efficiently. These sessoions will be provided by the project team.

3. DEVELOPMENT BACKGROUND/APPROACH

The software will be developed for stand-alone use on a Windows 10 computer. It will be developed in JetBrains PyCharm and Visual Studio for the backend and front end of the project, respectively. The project's wave analysis will be done using OpenCV, NumPy, and SciPy. The development team consists of students from the Software Engineering program at Kennesaw State University with experience developing software via Agile and SCRUM engineering processes. Other related skills that may be useful include knowledge of statistics, computer architecture, business development, and others.

3.1 High Level Estimates

Project Time Estimation:

Project Phase	Estimated Person Hours
Planning	48
Design	110
Development	230
Testing	175
User Manual / Support Documentation	120
Total:	**683**

Estimated lines of code – 666
Estimated pages of requirements – 3
Estimated pages of documentation – 15

3.2 Key Contacts and Stakeholders

- Internal Stakeholders
  - Joey Thompson
    - Team Lead: Overall project management, team coordination, and client liaison.
    - Email: jthom282@students.kennesaw.edu
    - Phone number: 404-822-2694
  - Ashly Altman
    - Front End Lead: Development and maintenance of the frontend application and packaging the application into an executable.
    - Email: aaltman6@students.kennesaw.edu
    - Phone number: 404-955-2699
  - Brooke Ebetino
    - Backend Lead: Development and maintenance of the backend Python code.
    - Email: bebetino@students.kennesaw.edu
    - Phone number: 678-371-5355
  - Tyler Haley
    - Testing Lead: Creating and executing test plans, ensuring software quality.
    - Email: thaley10@students.kennesaw.edu
    - Phone number: 770-634-6298
  - Masood Afzali
    - Documentation Lead: Maintaining project documentation and user manuals.
    - Email: safzali1@students.kennesaw.edu
    - Phone number: 706-667-1773
  - Nasiya Sharif
    - Kennesaw State University Capstone Project Manager: Organizer of the capstone project teams for Kennesaw State University. Acts as the intermediary between the project development team and the Robins Air Force Base representative.
    - Email: nrahman1@students.kennesaw.edu
    - Phone number: 470-578-5572
- External Stakeholders
  - Elizabeth Dayton
    - Direct end user representative from Robins Air Force base. Sponsor contact for the development team for any questions, information, or needs on the project. Working to ensure that the product meets end user need at Robins.
    - Contact via Nasiya Sharif at nrahman1@students.kennesaw.edu

4. FEATURES, PRIMARY DELIVERABLES, AND EXTERNAL COMMITMENTS

4.1 Feature List

- Video Input Handling: Read and process video files.
- Color-Based Filtering: Isolate screen and wave portions using color boundaries.
- Contour Detection: Detect the largest contours in filtered binary masks.
- Wave Analysis: Analyze the wave contour to extract characteristics like frequency and amplitude.
- Data Logging: Print detected wave characteristics in the console.
- CSV Data Export: Save extracted wave characteristics to a CSV file for external use.
- Real-time Visualization: Provide a real-time display of the processed video frames.
- User Configurable Parameters: Allow the user to configure color filters and kernel size.
- Error Handling: Graceful shutdown and error messages for unsupported formats or broken files.

4.2 Customer Deliverables

4.2.1 Software Application:

- SpecVidAnalyzer Backend (Python code for data processing)
- SpecVidAnalyzer Testing Suite
- Final Version: SpecVidAnalyzer v1.0

4.2.2 Documentation:

- User Guide
- Technical Documentation
- Test Plans and Reports
- Project Plan and Status Reports

4.2.3 Additional:

- CSV export functionality for data logging

4.2.4 Delivery Dates:

- Alpha Version for internal testing: October 31, 2023
- Beta Version for UAT: November 15, 2023
- Final Version: December 4, 2023
- Final Deliverables accepted by the client: December 8, 2023

5. PROJECT SCHEDULE

5.1 Major Project Milestones

Date (YYYY-MM-DD)	Milestone/ task	Deliverable	Remarks
2023-09-10	Project Plan	Project Plan	
2023-09-24	Design Document	Design Document	
2023-10-08	Finished Backend Framework	Weekly Report 8	
2023-10-15	Begin Testing	Weekly Report 9	
2023-11-05	Finish Testing	Weekly Report 12	
2023-12-04	User Manuel	User Manuel	
2023-12-04	Submit Code	Finished Code	
2023-12-08	Final Product	Final Product	

5.2 Project Status Tracking & Working Meeting Minutes

Meeting Location: Online
Meeting Time: Fridays from 6pm-8pm

Attendees:

- Brooke Ebetino
- Joey Thompson
- Ashly Altman
- Masood Afzali
- Tyler Haley

Agenda:
The meetings begin with an opening discussion and reviewing the current agenda. Followed by an update on the current status of the project and our task to be implemented by the next deliverable. Next, we discuss any risk that could have arisen during the week and strategies to address them. Additionally, we discuss the status of our deliverable and current trends and their impact on the project. Finally, we summarize the meeting and confirm our next meeting for the following week.

6. PROJECT WORK AND PRODUCT ESTIMATES

6.1 Estimate Summary

6.1.1 Function Point Analysis

Figure 6.1.1.1.1 Complexity Matrix

	Complexity		
Function type	Simple (S)	Average (A)	Complex (C)
Internal Logical File	7	10	15
External Interface File	5	7	10
External Input	3	4	6
External Output	4	5	7
External Inquiry	3	4	6

Figure 6.1.1.1.2 Component Complexity based on Complexity Matrix Values

Components List	Inputs (EI)	Outputs (EO)
Spectrum Analyzer	1 C 6 = 6	1 A 5, 2 S 4 = 13

Figure 6.1.1.1.3 Function Points

Unadjusted Function Points (UFP)	**18**
Value Adjustment Factor (VAF)	(0.64+ (5 (*ease of use)* + *5* (*ease of installation*)) * 0.01
Adjusted Function Points (AFP)	13.32

6.1.2 COCOMO Estimation

$$\boldsymbol{KLines\ of\ code}\ (\boldsymbol{KLOC}) = \frac{AFP\ *\ QSM\ Index(programming\ language)}{1000}$$
$$= \frac{13.32\ *\ 50}{1000}$$
$$KLOC = \mathbf{0.666}$$

$$\boldsymbol{Effort\ Applied}\ (\boldsymbol{E}) = 3.2\ *\ (0.666)\ \ 1.05$$
$$= \mathbf{2.088}\ Person\ Months$$

$$\boldsymbol{Development\ Time}\ (\boldsymbol{T}) = 2.5\ *\ (2.088)\ \ 0.38$$
$$= \mathbf{3.307}\ Months$$

$$\boldsymbol{People\ Required}\ (\boldsymbol{P}) = \frac{E}{T}$$
$$= \frac{2.088}{3.307}$$
$$= \mathbf{0.6313}\ Persons$$

$$\boldsymbol{Development\ Productivity} = LOC/Effort\ Applied$$
$$= \mathbf{318}\ LOC/Person\ Month$$

The tables below summarize the product size and effort estimates:

Project	Estimate Attributes		
	Size		
WBS areas	**Unit of Size**	**Size**	**Effort**
Total Requirements Effort (includes feature-related **and** "other" (non-feature) Requirements work)	Person Months	1	
Feature Related Requirements Size and Effort Totals	PAGES		
Total Development Effort (includes feature-related **and** "other" (non-feature) Development work)	Person Months	2.088	
Feature Related Development Coding Size and Effort Totals	LOC		
Feature Related Development Documentation Size and Effort	PAGES		
Total Testing Effort (includes feature-related **and** "other" (non-feature) Testing work)	Person Months	2	
Feature Related Testing Size and Effort Totals	TEST CASES		
Feature Level Effort Total (from Feature Estimate Worksheet)			
Development Effort Total (Includes Feature Level and project level overhead for Requirements, Development, and Testing)			
Project Level Effort Total (from Project Level Effort Estimates worksheet, excluding requirements, development, and testing)			
Project Total Effort (Project Totals + Feature Totals)			

7. PROJECT RESOURCE REQUIREMENTS

7.1 Staffing/ Skill Requirements

Team Member	Role	Critical Skills	Skill Gaps
Joey Thompson	Team Leader	• Strong Leadership • Communication Skills • Technical experience in software development • Conflict Resolution • Team Collaboration	• Frontend design
Ashley Altman	Development Manager	• Software Development Experience • Team Management Skills • Project Planning Skills • Resource Management Skills • Cost Management Skills	• People Management Training • Business Awareness
Brooke Ebetino	Planning Manager	• Goal Setting Skills • Strategic Planning • Resource Allocation Skills • Risk Assessment Skills • Data Analysis Skills • Mitigation Planning Skills	• Expertise in Risk Management
Tyler Haley	Quality & Process Manager	• Knowledge of Quality Assurance • Process Optimization Skills • Analytical Skills • Statistical Skills	• Expertise in Process Optimization
Masood Afzali	Support Manager	• Troubleshooting Skills • Problem Solving Skills • Knowledge of Support Tools • Communication Skills	• Advanced Troubleshooting Skills

7.2 Plan to Fill Skill Gaps

In order to make sure the skill gaps listed above are addressed the team will make sure to stay up to date with the knowledge needed for their role. Each member will also reach out and learn about any skills they are missing in order to address the skill gaps.

8. DEPENDENCIES AND CONSTRAINTS

8.1 Constraints

8.1.1 Time

The expected time that the Scrum team will have for product development will be 12-16 hours per week, which corresponds with the expected workload of a university student participating in a four-credit hour program.

The deadlines for the development of this product are:

- Capstone Kickoff (meeting with client), August 25
- Project Plan - September 10th, 3 weeks
- Design Document - September 24th, 2 weeks
- Submit Code and User Manual (prelim.) - December 4, 12.5 weeks (about 3 months)
- Final Version of all Deliverables (accepted by client) - December 8, ~0.5 weeks

Total time for project: 18 weeks (Approx. 4.5 months), 216—288 hours of work will be required for the completion of this project

8.1.2 Cost

Because this is an educational project there will be no budget provided. All resources and tools used for this project will be free to use for educational purposes. According to COCOMO effort calculations, this project will require Approximately 5.088 Person Months to complete.

8.2 Dependencies

The spectrum Analyzer will only be a single functional component, so there will be no project development dependencies.

9. RISK MANAGEMENT

9.1 Risk Management Strategy

The greatest risk to the project development is meeting the time constraint for the program as there is no way that it can be pushed back. In order to ensure that the project is completed by the required date, the target date for the first version of the program is currently set well ahead of time to October 8th, 2023 to allow ample time for testing, bug fixes, and any unforeseen issues that may arise and delay the software development. Weekly meetings are held with the capstone project via Kennesaw State University as well as weekly meetings for the development group (typically on Friday afternoons at 6 PM). Frequent and early communication is a priority as well as having anticipating an early completion date to allow ample time in the unlikely event that that date will have to be pushed back.

9.2 Initial Risk List

This is the initial risk list; the risks are listed in priority order from top to bottom. Descriptions are provided below the table. Risks with a pre-mitigation magnitude of 2.0 or below are not listed.

Risk number	Risk Priority (H, M, L)	Likelihood of Occurrence	Risk name: brief description	Mitigation Strategy
1	H	4	Program difficulty identifying/interpreting different screens displayed on the video inputs	MITIGATED
2	M	2	End user needs change before project completion	MITIGATED
3	M	3	Bugs from product testing will require extensive rework	MITIGATED
4	L	2	End user workstations will change after project creation	ACCEPTED

Risk 1 – Due to the variety of screens that the program will possibly need to read, the finished product needs to be able to interpret a variety of images correctly. To ensure that the program will work as planned with variations of input, a large number of samples will be requested from the client to ensure a large number of sample inputs for product testing. Testing will also begin well in advance of the product due date to ensure enough time to test and make necessary adjustments to make the product as accurate and precise as possible.

Risk 2 – End user needs from the output of the product being developed may change slightly before product completion. These changes should be minimal, if they occur at all, due to the specific goal of the product and should not require many changes to achieve end user approval.

Risk 3 – Bugs will inevitably be uncovered during the testing phase and will require rewriting of certain segments of the product source code to ensure reliable execution of the program. The odds of bugs requiring an extensive amount of man hours or large rework of code is low due to the low amount of complexity coming from the products creation as an independent stand-alone unit.

Risk 4 – End user computer hardware and operating systems may change at some point after product creation. When this change occurs, processing time for the program could potentially decrease if the new computer contains less capable hardware or creates an unforeseen error due to software changes on the new system. These events are difficult to anticipate due to future uncertainty. Issues with future hardware and software are unlikely to happen, but still possible.

10. PROJECT CONFIGURATION AND DATA MANAGEMENT

10.1 Configuration Management

Version control will be administered using Git and GitHub repositories. There will be separate development branches for code that is still in development which will go through a testing and approval process before being

merged with the main repository. The software tools used for version control will be GitHub Desktop, Windows Git Bash, and VS Code GitHub extension.

11. PROJECT PROCESS

11.1 Software Life Cycle Model

This team will be using the Scrum framework of the Agile project management model. Scrum is an iterative software lifecycle model that performs development in periods called "sprints." Sprint planning sessions will be conducted before each sprint, with pending work broken down into "stories" that describe specific use cases for the product, detailing each action the system performs during that use case, including the required inputs and outputs (entry/exit condition). Each team member will be assigned a story for that sprint, and each story will have a designated number of work hours required for completion and points. Points correspond to the effort required for developing a product feature, which considers complexity, risk, and the familiarity developers have with creating similar features.

Short team meetings, called "huddles," will be held periodically each week so the team leader, the Scrum master, can get status reports for the Scrum team, and stories can be moved to the subsequent development lifecycle phase. The stages of Scrum are:

- Planning - The Scrum team meets with the client to prioritize features to be included in the next sprint, and stories are created.
- Implementation - When scrum stories are ready for work, developers begin implementing the features required for each story
- Review and Testing - At the end of each sprint, product stakeholders test and review the stories implemented. The product is checked for errors, bugs, and completeness.
- Retrospective - The Scrum team and master review the sprint process to determine what went well and what can be improved for the next sprint.

The team will use Monday.com for software project management. This project management tool provides Scrum lifecycle management templates and is free for educational and small-scale projects.

12. REFERENCED DOCUMENTS

Cherednichenko, S. (2022, May 31). What are Constraints in a Software Development Project and How to Deal with Them Without Sacrificing the Quality. Mobindustry. Retrieved September 4, 2023, from https://www.mobindustry.net/blog/what-are-constraints-in-a-software-development-project/

Schwaber, K. (n.d.). What is Scrum? Scrum.org. Retrieved September 4, 2023, from https://www.Scrum.org/resources/what-Scrum-module

What Are Scrum Story Points? (n.d.). Wrike. Retrieved September 4, 2023, from https://www.wrike.com/Scrum-guide/faq/what-are-Scrum-story-points/

13. GLOSSARY

Agile: Methodology prioritizing collaboration, incremental development, and adaptability.
Backend: Server-side application component handling business logic and data storage.
Frontend: Client-side application component focusing on user interface and experience.
GitHub: Web-based platform for code management and version control.
Jira: Tool for project management, issue tracking, and bug tracking.
NumPy: Python library for large, multi-dimensional arrays and matrices.
OpenCV: Library for real-time computer vision applications.
Project ID: Unique identifier for project tracking.
Real-time: Immediate data processing and feedback.
Sprint: A set period for task completion in Agile development.
Stakeholder: Anyone who can impact or be impacted by the project.
SVAP2023: Project ID for the SpecVidAnalyzer Project 2023.
UAT: User Acceptance Testing; the final testing phase to ensure software meets user requirements.
Team Lead: Person responsible for overall project management and delivery.
Testing Lead: Person responsible for quality assurance and software testing.
Documentation Lead: Person responsible for maintaining project documentation.
SpecVidAnalyzer: Software developed for real-time video spectrum analysis.
Robins AFB: Robins Air Force Base, the project sponsor.

14. CHANGE RECORD

Date Changed	New Issue #	Description	Reason	Team Member
25 Aug 2023	2.0	Creates file; adds names, title	New file	Joey Thompson
30 Aug 2023	3.0	1, 2.1, 2.2, 2.3, 2.4	New content	Masood Afzali
03 Sep 2023	5.0	8.1, 8.2, 10.1, 11.1	New content	Ashly Altman
05 Sep 2023	6.0	2.5, 2.6, 4.1, 4.2	New content	Joey Thompson
05 Sep 2023	7.0	Changes all descriptions of frontend	Not using angular frontend	Joey Thompson
05 Sep 2023	11.0	Merge and make this master document. Formatting, spelling and Grammar Edits	Three working documents	Joey Thompson
05 Sep 2023	12.0	Ads to Glossary	Empty glossary	Joey Thompson
06 Sep 2023	13.0	3.0, 3.1, 3.2	New content	Tyler Haley
07 Sep 2023	14.0	5.1, 5.2, 7.1, 7.2	New content	Brooke Ebetino
08 Sep 2023	15.0	9.0, 9.1, 9.2	New content	Tyler Haley
09 Sep 2023	16.0	6	New Content	Ashly Altman
10 Sep 2023	17.0	Formatting	Submission	Joey Thompson