Software Project Management Plan

<AeroGotchi>

<10/8/2023>

<OoeyGUI>

Computer Science Dept/ CSUN

<Edmund Dantes>

# Revisions

| Version | Primary Author | Description of Version | Date completed |
| --- | --- | --- | --- |
| 1.0 | All Team Members | Original Document | 10/8/2023 |
| 1.1 | Inderjit Singh, Peter Hernandez | Added Elijah Lockett to the group member list and fixed due dates on Assignments and updated Gantt Chart | 12/3/2023 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Table of Contents

[**Revisions**](#_gjdgxs) 2

[**Table of Contents**](#_30j0zll) 3

1. [**Introduction**](#_1fob9te) 4
   1. [Project Overview](#_3znysh7) 4
   2. [Literature Review](#_2et92p0) 4
2. [**Project Organization**](#_tyjcwt) 6
   1. [Roles and Responsibilities](#_3dy6vkm) 6
   2. [Tools and Techniques](#_1t3h5sf) 6
3. [**Project Management Plan**](#_4d34og8) 7
   1. [Tasks](#_2s8eyo1) 7
   2. [Assignments](#_17dp8vu) 9
   3. [Timetable](#_3rdcrjn) 9

[**Additional Material**](#_26in1rg) 11

[Definitions, Acronyms and Abbreviations](#_lnxbz9) 11

[Appendices](#_35nkun2) 11

[References](#_1ksv4uv) 11

# Introduction

## Project Overview

**Purpose**

*AeroGotchi is an innovative concept where a drone pet takes inspiration from the digital pet format seen in games like Tamagotchi. This project involves the development of a mobile application that allows for drone controls and some gaming elements. We will be creating a Tamagotchi-style game, complete with pet-related responsibilities such as feeding, walking, and playing. We also plan to incorporate hand motion controls that will enhance AeroGotchi's personality and its ability to adapt to its surroundings, as well as its interaction with the user.*

**Scope**

*Aerogotchi will be a delightful drone companion available to the public who prefer not dealing with the responsibilities of living pets. The intended audience are people who yearn for a bonding experience with a companion and entertainment and excitement. Aerogotchi will serve as a staple building block of what can be achieved with hardware and AI, driving technological innovation to new heights.*

**Assumptions and constraints**

*Assumptions:*

* *Users who do not want real life pets will have an interest in owning and interacting with a drone pet using the mobile application with hand gesture controls and voice command.*
* *The technology and capability of smartphones with mobile application development is readily available.*
* *Users will feel emotional attachment towards the virtual drone pet similar to real life pets.*
* *Users will have an option of syncing the app towards the drone.*
* *User feedback will be gathered and used to improve the product.*

*Constraints:*

* *Regulatory compliance due to no fly zone, altitude limits and regulations regarding drones.*
* *Weather conditions will impact the drone’s operations.*
* *There is a learning curve towards user experience with drone controls and hand motion gestures.*
* *Safety concerns arise when drone’s do not have a collision avoidance mechanism to prevent accidents. Safety is our first priority.*

## Literature Review

**Ethics and Regulations:**

* We will be following the federal USC 44809 regulations aka “The Exception for Limited Recreational Operations of Unmanned Aircraft.” 1, 2 There are, as of September 9th 2023, 9 rules that we must follow, otherwise we’d have to follow 14 CFR Part 107 regulations, aka the “Small UAS Rule.”
  1. Fly for recreational purposes. Educational purposes do fall under this rule.
  2. We must follow the safety guidelines of any FAA-recognized Community Based Organizations (CBO) or the Advisory Circular (AC) 91-57B. If we were to follow a CBO’s guidelines, we would be following the Flight Test Community Association (FTCA) Safety Guidelines.
  3. We must always maintain our drone within visual line of sight or use a visual observer, who is physically next to the pilot.
  4. We will give way to and not interfere with other aircraft.
  5. We will fly under FAA-authorized altitudes in controlled airspaces only with prior FAA authorization, using Low Altitude Authorization and Notification Capability (LAANC) for getting said authorization.
  6. Within uncontrolled airspace, we will not fly above 400 feet.
  7. Whoever is the pilot of our drone would have taken The Recreational UAS Safety Test (TRUST) and carry the proof of test passage when flying.
  8. Our drone will have current FAA registration on the outside with the registration number, and carry proof of registration with us when flying.
  9. We will not operate our drone in a manner that endangers the safety of the national airspace system.

**Drone Programming and Hardware:**

* We will be programming the drone in Python while using machine learning libraries to implement certain image recognition properties.
* The Ardupilot firmware is used to command the drone’s hardware by sending 400 commands a second to the drones motors, which results in a smooth and steady flight.
* The Tello Software Development Kit (SDK) allows us to connect our drone through Wi-Fi to better control it with the use of text commands.
* We will use MediaPipe/OpenCV for machine learning. It has pre-trained models and frameworks to perform various tasks related to computer vision and machine learning.
* To train our custom face recognition model to recognize their owner, we will collect a diverse dataset of the owner's face images, preprocess and label them, select a suitable machine learning model and then refine.
* Pixhawk - flight controller
* ArduPilot - software for the PixHawk
* Gazebo can be considered for implementing a drone-based environmental monitoring system. Utilizing Gazebo's capabilities to create realistic environments and simulate various flight conditions.
* Implementing ArduPilot, an open-source autopilot software, with Gazebo for a drone simulator. This combination allows us to simulate and testdrone's flight control algorithms in a realistic environment.

**Mobile Application Related:**

* For our Mobile Application we will be using Flutter. Flutter (https://www.flutter.dev) is an open source framework developed by Google to build natively compiled applications for mobile, web and desktop from a single codebase.



1 US Department of Transportation “Recreational Flyers and Community Based Organizations” Federal Aviation Administration, August 07 2023,<https://www.faa.gov/uas/recreational_flyers>

2 Pilot Institute “What are the rules to fly your drone in 2023?” YouTube, 26 June 2021,<https://youtu.be/oyE2x9B0CVA?si=aBLqwMK8I37-u09d>

3 Tello SDK 2.0 User Guide <https://dl-cdn.ryzerobotics.com/downloads/Tello/Tello%20SDK%202.0%20User%20Guide.pdf>

4“Build Apps for Any Screen.” *Flutter*, <https://flutter.dev/>

5 OpenCV. "OpenCV." <https://opencv.org/about/>. Accessed 27 September 2023.

6 Chakraborty, A. "Image Processing and Image Pattern Recognition: A Programming Tutorial." Proceedings of the 2018 First International Conference on Artificial Intelligence for Industries (AI4I), Laguna Hills, CA, USA, 2018, pp. 122-123, doi: 10.1109/AI4I.2018.8665702.

7Ismail, Ahmad Puad, et al. "2021 IOP Conference Series: Materials Science and Engineering, vol. 1045, article number 012043." IOP Publishing, 2021, doi: 10.1088/1757-899X/1045/1/012043.

# Project Organization

## Roles & Responsibilities

| Team Member | Roles | Email |
| --- | --- | --- |
| Inderjit Singh | Team Lead, Full-Stack Grandmaster | inderjit.singh.644@my.csun.edu |
| Peter Hernandez | Backend Designer, Autonomous Engineer, Developer Apprentice | peter.hernandez.590@my.csun.edu |
| Jarret McIntire | Frontend Designer, Autonomous Engineer | jarrett.mcintire.279@my.csun.edu |
| Zamir Barbosa | Developer Wizard | zamir.barbosa.112@my.csun.edu |
| Keith Chua | Backend Designer, Image Recognition Developer | keith.chua.349@my.csun.edu |
| Elijah Lockett | Frontend Designer | elijah.lockett.147@my.csun.edu |

## Tools & Techniques

* Github - a platform for version control and collaboration of software development projects. It has tools that help developers and teams to manage code, track changes and work together on projects.
* ArduPilot - open source software used for controlling autonomous vehicles specifically in unmanned aerial vehicles (UAV). Frameworks available to use Flight Control, Autonomous Navigation.
* MediaPipe - open source software for computer vision and machine learning related to Video Analysis. Frameworks available to use Augmented Reality (AR), gesture recognition, image tracking.
* OpenCV - open source computer vision and machine learning software library to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. It can be used to detect and recognize faces, identify objects, and track movement.
* MySQL - database management system that is used to store, retrieve, process data.
* Flutter - mobile development software that has user interfaces (UI) and user experiences (UX). Works well with MediaPipe since both are from Google.
* Waterfall Model - Sequential Software development process that breaks down the development into certain cycles and phases.
* Tello SDK - Documentation of commands for programming the drone.
* Discord - communication software used for chats, voice and video calls.
* IntelliJ IDEA - integrated development environment with Python Software Development Kit extension to write and compile python code.

# Project Management Plan

## Tasks

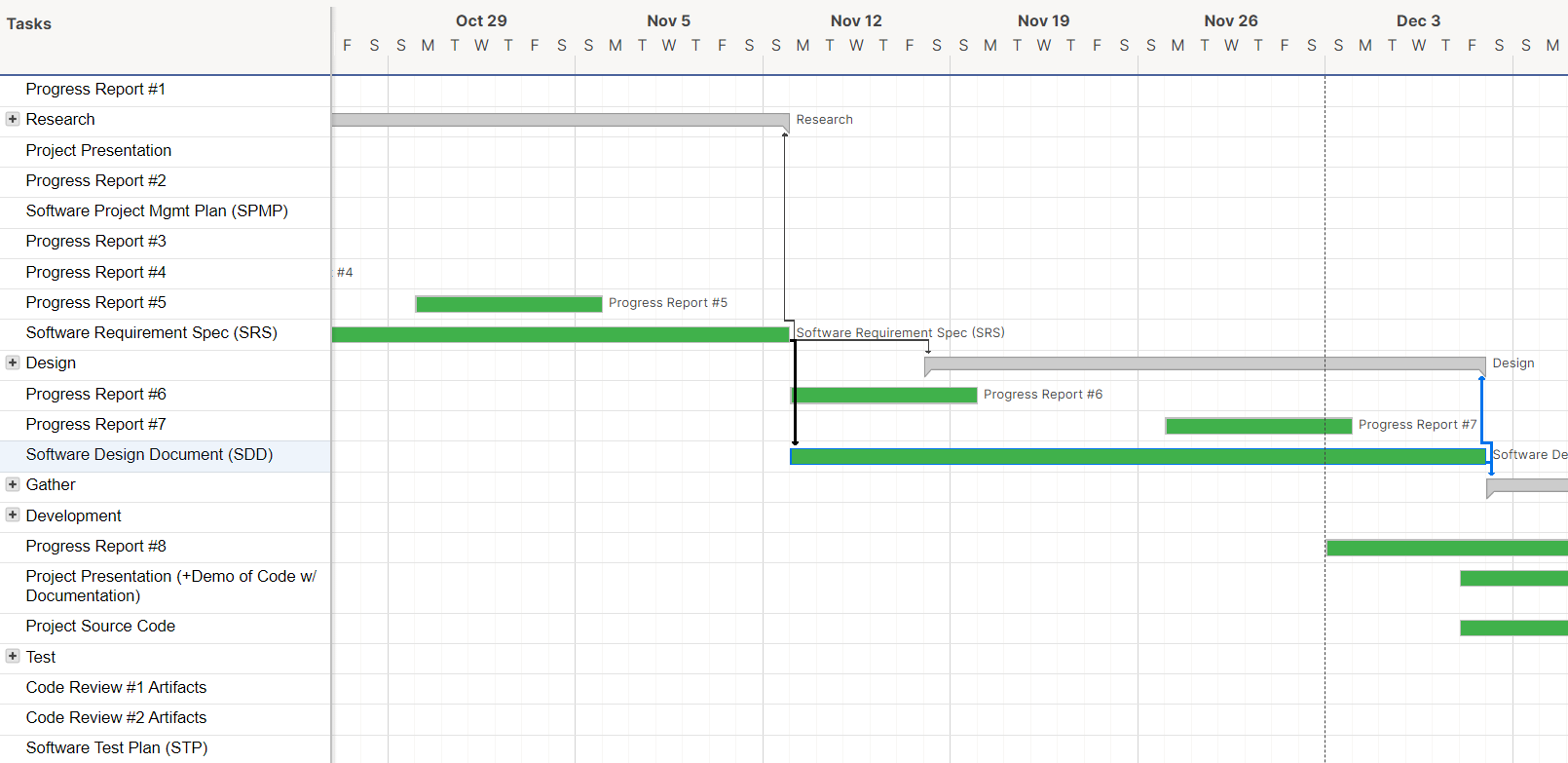
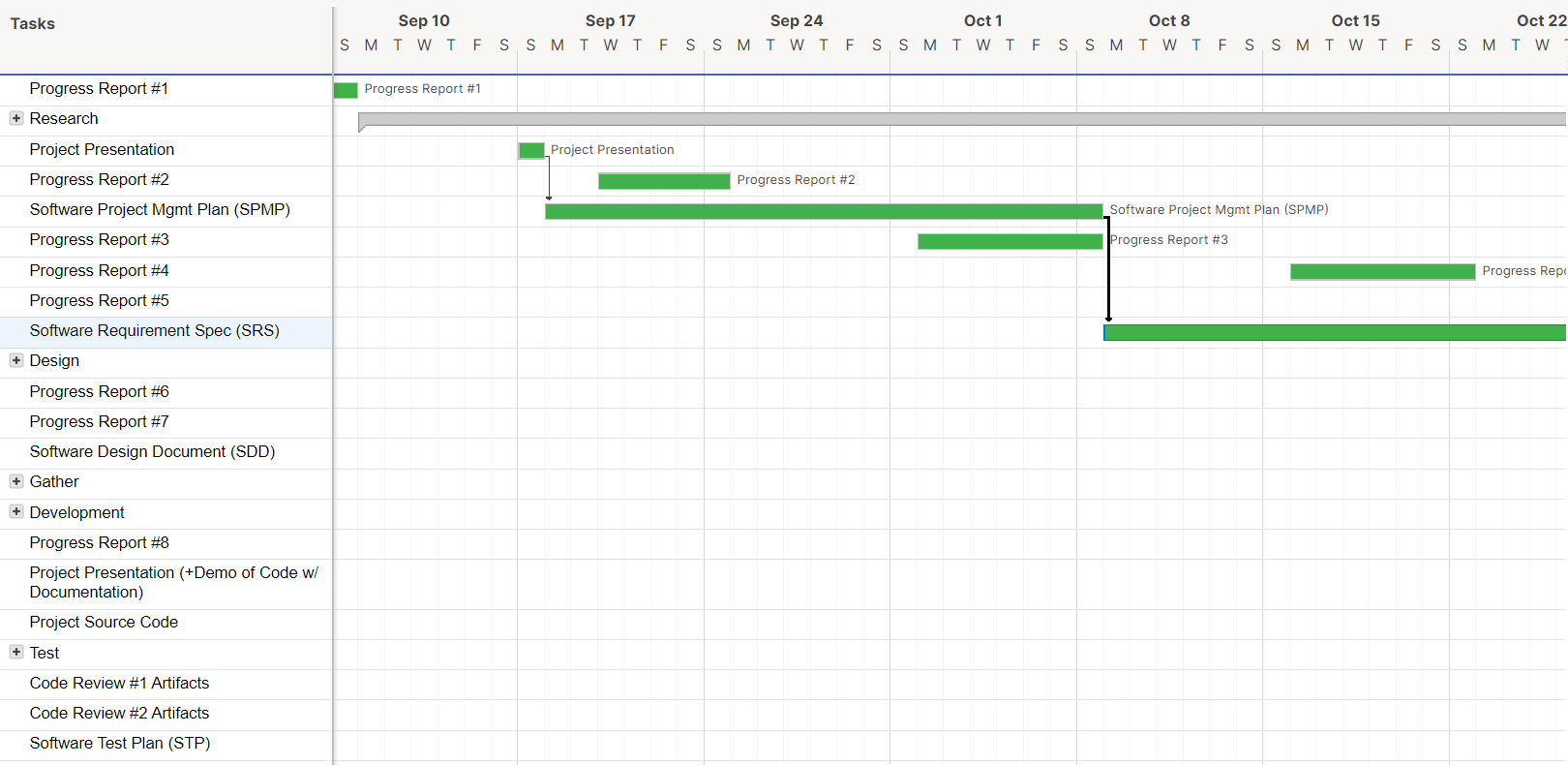
| Task#/Phase | Task Description | Start Date | End Date |
| --- | --- | --- | --- |
| Research 1 | Mobile Application | 09/11/23 | 11/12/23 |
| Research 1.1 | Language Types | 09/11/23 | 11/12/23 |
| Research 1.2 | Game Design | 09/11/23 | 11/12/23 |
| Research 1.3 | Drone Controls | 09/11/23 | 11/12/23 |
| Research 2 | Autonomous Flight | 09/11/23 | 11/12/23 |
| Research 2.1 | Drone Hardwares | 09/11/23 | 11/12/23 |
| Research 2.2 | Drone Softwares | 09/11/23 | 11/12/23 |
| Research 2.3 | Drone Simulation | 09/11/23 | 11/12/23 |
| Research 3 | Image Recognition | 09/11/23 | 11/12/23 |
| Research 3.1 | Computer Vision | 09/11/23 | 11/12/23 |
| Research 3.2 | Drone integration | 09/11/23 | 11/12/23 |
| Design 1 | Mobile Application | 11/13/23 | 12/03/23 |
| Design 1.1 | Tamagotchi Architecture | 11/13/23 | 12/03/23 |
| Design 2 | Autonomous Flight | 11/13/23 | 12/03/23 |
| Design 2.1 | Enclosed Environment Architecture | 11/13/23 | 12/03/23 |
| Design 3 | Image Recognition | 11/13/23 | 12/03/23 |
| Design 3.1 | Computer Vision Architecture | 11/13/23 | 12/03/23 |
| Gather 1 | Mobile Application | 12/04/23 | 12/24/23 |
| Gather 1.1 | IDE Configuration | 12/04/23 | 12/24/23 |
| Gather 2 | Autonomous Flight | 12/04/23 | 12/24/23 |
| Gather 2.1 | Drone | 12/04/23 | 12/24/23 |
| Gather 2.2 | Flight Controller | 12/04/23 | 12/24/23 |
| Gather 3 | Image Recognition | 12/04/23 | 12/24/23 |
| Gather 3.1 | Camera Configuration | 12/04/23 | 12/24/23 |
| Development 1 | Mobile Application | 12/25/23 | 03/07/24 |
| Development 1.1 | Integrate Manual Drone Control Switch | 12/25/23 | 03/07/24 |
| Development 1.2 | Integrate Game Mechanics | 12/25/23 | 03/07/24 |
| Development 2 | Autonomous Flight | 12/25/23 | 03/07/24 |
| Development 2.1 | Connect Autonomous Hardware | 12/25/23 | 03/07/24 |
| Development 2.2 | Create Live Checkpoints | 12/25/23 | 03/07/24 |
| Development 3 | Image Recognition | 12/25/23 | 03/07/24 |
| Development 3.1 | Integrate Computer Vision Code | 12/25/23 | 03/07/24 |
| Development 3.2 | Train Facial and Hand Gestures | 12/25/23 | 03/07/24 |
| Test 1 | Mobile Application | 03/08/23 | 04/20/24 |
| Test 1.1 | Demonstrate Manual Drone Controls | 03/08/23 | 04/20/24 |
| Test 1.2 | Demonstrate AeroGotchi | 03/08/23 | 04/20/24 |
| Test 2 | Autonomous Flight | 03/08/23 | 04/20/24 |
| Test 2.1 | Waypoint Flying | 03/08/23 | 04/20/24 |
| Test 2.2 | Game Checkpoint Flying | 03/08/23 | 04/20/24 |
| Test 3 | Image Recognition | 03/08/23 | 04/20/24 |
| Test 3.1 | User Facial Gestures | 03/08/23 | 04/20/24 |
| Test 3.2 | User Hand Gestures | 03/08/23 | 04/20/24 |

## Assignments

| Deliverables/Progress | Due Date |
| --- | --- |
| Progress Report #1 | 9/10/23 |
| Project Presentation | 9/17/23 |
| Progress Report #2 | 9/24/23 |
| Software Project Mgmt Plan (SPMP) | 10/08/23 |
| Progress Report #3 | 10/08/23 |
| Progress Report #4 | 10/22/23 |
| Progress Report #5 | 11/05/23 |
| Software Requirement Spec (SRS) | 11/12/23 |
| Progress Report #6 | 11/19/23 |
| Progress Report #7 | 12/03/23 |
| Software Design Document (SDD) | 12/08/23 |
| Progress Report #8 | 12/13/23 |
| Project Presentation (+ Demo and Documentation) | 12/13/23 |
| Project Source Code | 12/13/23 |
| Code Review #1 Artifacts | TBD |
| Code Review #2 Artifacts | TBD |
| Software Test Plan (STP) | TBD |

## Timetable

* Green - Tasks
* Blue - Assignments



# 

# 

# 

# 

# Additional Material

Definitions, Acronyms and Abbreviations

* UAV - Unmanned Aerial Vehicle
* AR - Augmented Reality
* UI - User Interface
* UX - User experience
* CBO - Community Based Organizations
* AC - Advisory Circular
* FTCA - Flight Test Community Association
* FAA - Federal Aviation Administration
* LAANC - Low Altitude Authorization and Notification Capability
* TRUST - The Recreational UAS Safety Test
* SDK - Software Development Kit

## Appendices



