Initial Draft typed by EB, JW, NM.
Finalized draft edited by NM.

Dog Adoption Network (DAN) Project Plan

Management Plan: The roles of each team member will evolve over time, and often the nature of managing what needs to be done will be improvisational, like jazz. However, below (**Table 1**) is a short list of the core responsibilities taken on by each team member.

Team Member	Experience	Project Role
Elizabeth Bailey (EB)	Python, Tkinter	UI Development + Quiz Module
Jack White (JW)	Python, C, Tkinter	UI Development + Quiz Module
Nathan Malamud (NM)	Technical Writing, Python	Documentation + Filtering and Ranking Modules
Evan Parisier (EP)	MySQL, Python	Database and server development

Table 1. Core responsibilities took up by each team member for DAN development.

Decisions about particular modules were ultimately left up to the persons responsible for that module, as listed in **Table 1.** Decisions that impacted functions of the entire project were discussed with the entire team via texting, in-person meetings after class, and Discord. Important meetings - i.e. ones that entailed major changes to the ARA software architecture - were documented and shared with the entire team via Discord.

Monitoring and Reporting Progress: Short 30-minute check-in meetings were scheduled Tuesdays and Thursdays after class combined with asynchronous communication via Discord. This allowed ample time for discussing the design and implementation of the project.

Assignments and task completion dates were kept track of in a spreadsheet in a stored google drive (attached to this final project plan in **Table 3**). Our code was published on GitHub which enabled us to automatically track new changes to the program modules.

Our Build Plan: We divided tasks across 3 key domains: program views, program logic, and program data. Program views were implemented by two people (EB and JW) using the Flask framework for website development. Program logic was done in native Python (by NM) and program data was stored and managed using MySQL (by EP). The three-domain system allowed for a clear delineation of tasks and modularity.

Tasks were scheduled during meetings. Since there is no hierarchy in terms of leadership, tasks were generally self-scheduled and then approved or disapproved by the team.

Timeline: **Table 2** shows our preliminary timeline (as of May 9th) for our project. It is expected that new milestones will be accounted for as we continue to develop our program design.

Weekday Month.Day	**Milestones**
M 05.09	Initial outline of Project Plan / SDS / SRS
W 05.11	Conduct interviews with local adoption centers
M 05.16	First draft of SRS and SDS Create Github Repository Begin writing task management spreadsheet
W 05.18	Set up a database on U of O IX server Basic windows can be traversed
M 05.23	Windows handle user “quiz” input and store properly (at least locally)
W 05.25	Available dogs can be displayed and filtered from user input via windows
F 05.27	User input from the “quiz” function can toggle appropriate filter, display dog matches
~~M 05.30~~ M 06.06	Final Submission - A functional and complete product, ready for presentation.
~~T 06.07~~	~~Presentation~~ (canceled)

Table 2. Expected timeline for project development as of May 9th, 2022.

Where did we get our dogs from? Rather than reach out to an adoption center to display their dogs on our site, we randomly generated numerical values for the personality traits of different dogs using a python script. A bio for each dog was written to match the randomly-generated values from the python script. Stock photos for the dog were found in Adobe Stock Images. The script that we used to generate our dogs is on our GitHub page.

Database Schema Rationale: We chose MySQL for our project as our group was more familiar with the syntax, and we all had prior experience using it for the ARA application (Project 1).