

PawJector Final Report Summary

Group 31: Ryan Michalik, Seun Aduyemi, Ivan Torres, Ethan Ky

I Final Project Description:

The “Pawjector” is an interactive projector system designed for cats, combining projection technology with mobile connectivity, cloud services, and pet activity tracking. It values security, reliability, usability, and pet safety while providing a playful and customizable experience for both cats and their owners.

Core Features Include: Interactive Projection and Cat Behavior capture, Mobile App Integration for monitoring and control, and Custom Cat Level Creation for Community.

II Requirements:

Data Management - Local and Cloud Storage: User profiles, maps, and activities saved both locally and in the cloud for offline use and device syncing

Performance - Lag Tolerance and Smooth Animation: Animations at 30–60 fps. Handles network delays up to 1000ms and recovers quickly.

Usability and Accessibility - Simple UI and Accessibility: Quick onboarding; multi-language and elder-friendly design.

Environmental - Indoor Operation and Light Adaptation: Monitor and detect potentially hazardous areas/objects. Adjusts projection based on room brightness.

III Design:

For this section, we will bring up things related to the design of the projector. Firstly, we have several goals for this project, which include engagement from cats, responsiveness, safety, and customizability. To elaborate, the projector should have engaging levels that will keep the cat’s attention and interest in the activities displayed.

Design Goals:

- **Engagement:** Making a system that grabs cats’ attention and keeps them engaged. This is important because customers would expect our product to keep cats engaged and it would be very helpful for those who are often on errands and unable to spend a lot of time with their pets. Otherwise, people would be less likely to buy the product.
- **Performance:** A program should be able to run at a decent fps with minimal lag to satisfy user experience. With more lag, the user experience would be less enjoyable (for the user and the cat).
- **Reliability:** Having a database that stays active when users aren’t using the app is important so personal data can be saved. The data also needs to be protected and encrypted for data security and protection. We want to make sure that users will always have their data saved and backed up even if the servers shut down; otherwise, they lose their data.
- **Customizability:** Pet owners can customize the difficulty and levels for cats to play. Every user should have a unique experience and be able to customize how they use the pawjector to fit their needs and their cats needs.

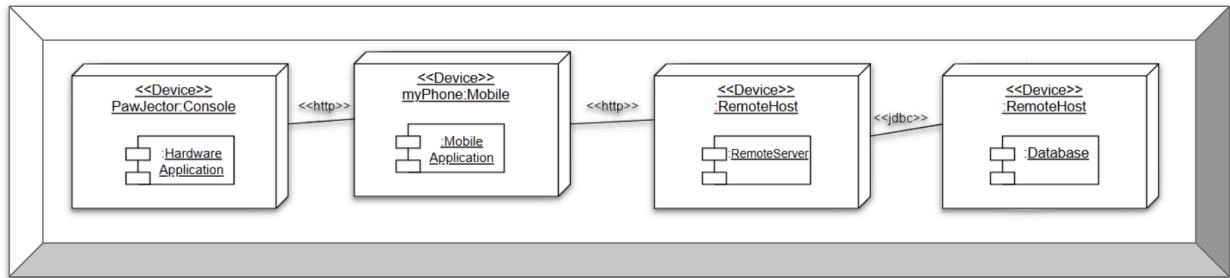
- **Maintainability:** The system is easy to maintain, update, enhance, repair, and modify. The system should be easy to maintain so data about the users' cats can be collected quickly for us to quickly improve our project to grab cats' attention.

Proposed System Architecture:

A 4-tier system splits an application into four distinct layers, each responsible for a different part of the system. These layers interact sequentially, with each communicating only with its adjacent layer. In our case, the subsystems are: the "Pawjector" Console, the "Pawjector" App, the Company Server, and the Database — communicating in the order presented.

- **Pawjector Console:** Displays projections and captures basic cat activity.
- **Pawjector App:** Sends user commands and displays live feedback.
- **Company Server:** Processes requests and manages communication between app and database.
- **Database:** Stores user, cat, and activity data securely.

Deployment Diagram:



Class Diagram:

