

# Large Animal Care Simulator Final Summary

## Group 30 - Lance Guevarra, Alex Legashvili, Zitao Li, William O'Brien

The Large Animal Caring Simulator was created to give users a hands-on, interactive way to learn how to care for large animals in a realistic virtual world. From the very beginning, our team aimed to build a system that feels intuitive to use, engaging to explore, and firmly based on real animal-care practices. Our goal was to make the learning experience both meaningful and enjoyable, while still capturing the complexity of caring for large animals.

The goal of our design is to create a simulator that is both accurate and genuinely fun to use. We want people to learn real animal care concepts, but we also want them to enjoy coming back and spending time with the animals. Just as importantly, the system should feel simple and approachable for everyone, whether it's a 5th grader trying it for the first time or an adult exploring it out of curiosity. Users shouldn't feel overwhelmed or frustrated and the experience should feel smooth and intuitive. We aim to keep the program lightweight so it can run well on everyday devices like school laptops and Chromebooks, making sure as many people as possible can access and enjoy it.

Looking ahead, the simulator aims not only to provide a fun and engaging experience but also to serve as an educational tool that fosters empathy and understanding of large animal care. Future updates could include additional animal species, more complex environmental interactions, and advanced gameplay features based on user feedback and expert consultation. By continuously refining the interface, expanding content, and enhancing realism, the simulator seeks to remain both an enjoyable game and a meaningful learning platform for users of all ages.

Our final system design is built using a Client-Server architectural design to clearly separate the user experience from the core simulation. The client handles everything the player sees and does, while the server processes actions, updates the game session, and then sends the result back to the client to keep the client data up to date, allowing us to maintain and update the system more efficiently. The system is organized into three subsystems, the UI subsystem, which manages all user-facing screens and interactions; the Game Handler subsystem, which runs the main simulation for ecosystems and animals; and the Data Management subsystem, which stores and maintains all game states and handles saving and loading through the SaveLoadManager. Together, these parts create a clean, flexible structure that supports an intuitive and reliable user experience.

The design prioritizes accessibility, ensuring that the simulator runs smoothly on a wide range of devices, from school laptops to personal computers. The modular structure also allows for future expansion, making it easier to add new animals, ecosystems, or features, and to incorporate feedback from users or experts in the field.

The object design of our system is organized around four main areas; the UI subsystem, the Game Handler, Data Management, and the Server. The UI subsystem focuses on everything the player sees and interacts with, including login screens, settings, and simulated animals and

environments, without managing internal data. The Game Handler drives the core simulation, managing animals, environments, and player interactions, with key classes such as Animal, Environment, Player, and SimulationManager. The Data Management subsystem handles the organization, saving, and transferring of game data, allowing progress and environmental state to be managed independently and synced seamlessly. The server also ensures data persistence across devices, supports real-time environmental features like weather, and manages updates, all coordinated through a singleton pattern for reliability. These systems create a flexible, intuitive, and immersive experience that allows users to explore and care for large animals in a realistic virtual world.

In addition, the system was designed to work seamlessly across multiple platforms, including mobile devices, consoles, and PCs, with each client sharing the same core interaction and game logic while adapting to platform specific interfaces. Persistent data management ensures that user profiles, animal information, and environmental states are safely stored and synchronized through a relational database, with security measures in place to protect privacy. The system handles startup and shutdown scenarios carefully to prevent data loss, and the UI is designed to be interactive and engaging, including dynamic animal populations, behavior and intuitive controls, all while remaining lightweight and accessible on everyday devices.

Lastly, issues currently not resolved with the project were outlined. The concern of how consultation with panels of different demographics and experts in the field was brought up, as these consultations could change our approach to the final product. Off-the-shelf solutions were highlighted in an attempt to consider any already existing solutions to problems that the project may encounter. Existing libraries, development toolkits, and similar animal care games were considered as components that could be incorporated into the game in some capacity. Problems pertaining to the environment in which the game will be launched were brought to attention as well. This included how the game might impact current hardware in educational environments (especially concerning performance), potential effects on users, and how the product should evolve over its lifespan. Risks involving the product were also evaluated. This included performance on both the server and client sides of the game, as well as user interface intuitiveness and data storage. The cost of the game was considered, with mention given to the time, labor, and opportunity costs that play a role in estimating all the resources needed to see the product to completion. The issues section concluded with plans for future updates, ideas to consider for potential solutions to some outlined issues, and a retrospective for the project as a whole

Overall, the Large Animal Caring Simulator combines an engaging and interactive experience with meaningful educational value, providing users of all ages a realistic way to learn about large animal care. The modular and flexible design ensures the system can evolve over time, incorporating new features, user feedback, and expert guidance. By balancing the usability, realism, and accessibility, the simulator lays a solid foundation for continued learning, enjoyment, and future development,