

Design Project Final Summary

Paleo Rebirth

Group 8

Areli Celestin, Abrar Makki, Noor Muhsen, Swetha Soppadandi

Paleo Rebirth is an expansive game that combines RPG elements with educational puzzles and demonstrations. One of its main focuses is on providing a fun learning environment for children who wish to learn paleontologic concepts. Another thing that the game focuses on is entertaining gameplay for all age groups. The game's main concept revolves around the player discovering and digging up fossils at dig sites that they can then take to the lab and use to revive ancient creatures. The game also has a compendium feature that includes all the different concepts and animal facts the player has learned through gameplay. This compendium includes an AI chat feature which allows the user to ask the AI any specific questions they may have on the concepts or facts. The game also includes a story and an antagonist that the player will need to ultimately defeat to win the game. This naturally introduces the battle's gameplay feature. The story itself uses NPC's in the story's battles, but the player can also connect to the server and play against other players in a weekly style tournament setting. The game also includes a virtual home that the player can decorate and customize to their liking. There are in-game purchases, but there are also microtransactions (this feature is turned off when in the use of classrooms) that can be made by the player which help Paleo Rebirth make money, and allows the game to continue receiving updates and expanding.

There are some important testing requirements that the implementation of the project will require. To give a brief but important overview of these requirements three important requirements will be listed; A functional requirement, a performance requirement, and a look and feel requirement. The first requirement, the updating system requirement, focuses on notifying the user when there are updates to the game and updating the users local copy of the game when there are. This requirement is important because continuous updates to the game are important for its longevity and expansion. These updates are also important because they may include new performance, security, and system features that the player needs to play the game at its fullest capacity. The second requirement we'll cover in this summary, the speed and latency requirement, covers the game's different loading, speed, and latency requirements. These

requirements include scene change times, data fetching times, player vs player lag, and in-game purchases (micro transactions using real money) and their processing time (bank server / third party payment processing servers). Since this software is a game, speed isn't as important as in other software (such as life saving software), but it is vital to user experience. Latency should not exceed for long enough that the player's attention is pulled elsewhere. It's also important that some things like payments are processed in a timely fashion so that there isn't any confusion about where money is and when it will be transferred from player to company. Lastly, the look and feel requirement (style) should be fun and inviting. Since this game is made for children, the game should appear friendly and inviting, not off putting. It should also appear fun so that children can enjoy learning different things. This game should also be fun not only so that students can enjoy learning, but also so that casual players feel compelled to make purchases.

This game will be making use of the server client architecture. This architecture will be vital for the implementation of player vs player battles and for the games artificial intelligence. The server will handle the coordination of all clients attempting to interact with one another. The server will also store tournament data and distribute it to different player clients when requested. The artificial intelligence used for controlling some NPC battles and for the use of the compendium chat will be stored on the server. This way, there will be one AI "brain" stored for the entire userbase. There are many subsystems that the game will use but we'll give a brief overview of the games vital subsystems. The map subsystem will be used to keep track and connect different locations the player can explore in the game as well as the class that is in charge of displaying the game on screen. The server subsystem will be in charge of connecting vital classes that interact with the server and client side of the overall class system. The player subsystem will be the subsystem in charge of keeping track of important player classes such as skill tree, rank and credentials class. Finally, the battle subsystem connects the server, player and important battle classes together. Paleo Rebirth will be storing long-lived player data on the backend database server, with a small amount of cached data on each player device. When the system starts back up the game server will retrieve the data and load it onto the games client.

Lastly, the costs of the game will be briefly overviewed. The game design is expected to take 10-12 months with a cost of \$500,000 or more. Testing and Localization is expected to take 3-4 months and will be expected to cost no more than \$250,000 dollars.