

OOAD Project Update

Project Summary

- **Project Title:** Communal Lodestone
- **Developer:** Robert LaChausse (solo project)
- **Overview:**

Communal Lodestone builds on top of the FFXIV Lodestone's public APIs and community submissions to provide a collaborative layer. Users will be able to create shared build guides, set up item/event alerts, and store shared data to better coordinate Free Company activities.

The goal is to make Lodestone more than a static, read-only database by adding dynamic tools for planning, alerts, and guides.

Work Done: Sadly, I was extremely busy this week with family matters in denver. Due to this, I only really completed the test-light part of the project. I estimate that I'm only 30-40% done with what I need. This is also partly due to the api i'm using (XIVapi) being extremely confusing-it's translated almost directly from japanese/korean into english and other european languages, making even simple requests turn into arduous tasks. This project will be a priority going into next week. Below is what I HAVE done, and what will essentially be represented in the video.

- created a log-in interface.
- Made database schemas and tested user entrance.
- Reworked UML diagram (as seen in repo)
- Finished planning of work of the observer pattern and multiple interfaces alongside.
- Formulated JSON-object modification class, which allows for data in the back end to be sent to the front end and vice-versa.

Changes or Issues Encountered:

The #1 #1 #1 issue was working with the api. I cannot fully describe how unfriendly both the general endpoint formatting AND the documentation is. Unless you go into the user help resources or are familiar with FFXIV's infrastructure from inner development, you will not be able to use it at all. AND EVEN IF you do go ask for help, there is only one primary english-speaking developer available at certain times of the day. While I cannot blame the dev for having limited availability, it does bottleneck my own progress severely.

My #2 and the last issue was on the web development side. Working with front-end frameworks was not as friendly as I believed it to be from past experience and did not allow me the customization I wanted. I had to scrap one entire day's work and utilize a classic css&html design instead. This definitely slowed me down, but it allowed me to work while not having access to my primary station.

Patterns:

I plan on using builder and observer patterns in my design. The observer pattern, while not fully implemented, is completely planned. This is what is going to allow me to operate my alert system. I intend on using the builder pattern to simplify and expedite my "guide blocks" system, which will allow different things in a guide (items, achievements, skills) to all be processed thoroughly in one method.

Plan for next iteration:

I need to work hard on finishing these pattern implementations and allow for the user to see their data (aka, their guide and alerts) dynamically through the webpages. I also need to clean it up and make it look way cleaner, but that comes second to functionality.