

Final Fantasy XIV (Patch 5.1) Crafting Simulator

Qihao Gu (qihaogu2) Moderator: Max Qian

1. Abstract

1.1. Project Purpose

A tool to simulate crafting in Final Fantasy XIV (Patch 5.1).

1.2. Background/Motivation

I'm a player of Final Fantasy XIV (an online game). The game has an interesting crafting system which various projects have been developed to simulate. Most notably are:

<https://ffxivteamcraft.com/simulator/custom>

<https://ffxiv-beta.lokyst.net/#/simulator>

In the upcoming patch (5.1), major changes will be applied to crafting system in game (mainly the add /removal / remake of various crafting actions), so all major crafting simulators need to update accordingly. I want to build a crafting simulator that covers upcoming changes to

1. Learn how the simulator works, so that one day I can contribute to open sourced crafting simulator projects like teamcraft.
2. Learn how to interact with the game data (there's a popular api <https://xivapi.com/> developed by players) so I might build other projects besides the crafting simulator.
3. Do something cool \o

The theory support of my simulator mainly comes from the following post, which explains the backend logic of crafting (it is written in Chinese):

<https://nga.178.com/read.php?tid=18839082>

2. Technical Specifications

- 2.1. **Platform:** Website / Java application
- 2.2. **Programming Languages:** Java
- 2.3. **Stylistic Conventions:**
- 2.4. **SDK:**
- 2.5. **IDE:** IntelliJ
- 2.6. **Tools/Interfaces:** Maven

2.7. **Target Audience:** myself, other players

3. **Functional Specifications**

3.1. **Features**

- User can search for the requirement of all crafting recipes in the game
- User can perform (a sequence of) any crafting actions and see the result in the simulator. The simulated result should reflect in game result.
- User can make macro (i.e. a fixed sequence of crafting actions) in the simulator and test the success rate of the macro by running it many times.

3.2. **Scope of project**

- The mechanism behind some of the new crafting actions might be unknown through out the development of the simulator.

4. **Timeline:**

4.1. **Week 1**

Simulator Logic

- Learn how other simulators work
- Learn to use <https://xivapi.com/>
- Describe crafting actions programmatically
- Describe crafting recipes programmatically
- Describe the crafting process (including random condition) programmatically
- Simulate crafting process with implemented actions and recipes

Database

- Deploy and store necessary data in a database (mariaDB)
- Learn to interact with the database using prepared statements

4.2. **Week 2**

Simulator Logic

- User can search for actions / recipes
- User can redo / undo crafting actions
- User can create macro to record a fixed sequence of crafting actions
- User can test the success rate of a macro
- Macro can be saved in database

Web application (Java Server Page)

- Use Maven, JSP and JDBC to build a web application
- Deploy simulator online, build user interface on the website

- Icons related to actions / items / recipes should be displayed on the website

4.3. Week 3

Crafting material list

- Gather information (quantity, acquiring method, clock, etc.) about all the materials used to craft a recipe
- Save the information in an efficient data structure so it can be reused

Web application

- Improve the look of simulator
- Deploy crafting material list to the website
- Display the related information of an item nicely on the web application, for example <https://garlandtools.org/db/#>

5. Future Enhancements