Project Overview

I'll present you with a description of data to be represented in a database.

- Milestone 1: Develop a logical model
- Milestone 2: Develop a physical model and populate with sample data
- Milestone 3: Construct a Java data model and JDBC data access classes
- Milestone 4: Final Implementation (web front-end)

Project Overview

Problem:

Construct a database for representing data that might be used to represent player data in an online role-playing game.

(This is based on a real online game, but you do not need to be familiar with the specific game in order to complete the project.)

Detailed write-up coming in the next few days.

Project Overview: Players

Player Account:

- email address
- zero or more characters

Project Overview: Character

- Name (top right, obscured)
- Max HP
- Max MP is always 10k; no need to store
- Attributes
  - Includes all categories from the list ("Attributes", "Offensive Properties", etc.) – don't worry about categories
  - Average Item Level is computed
- Equipped Gear

Project Overview: Character

Gear is equipped in various slots:

main hand	off-hand
head	earring
body	neck
hands	wrist
legs	left ring
feet	right ring

We'll simplify and consider a single ring slot.

Project Overview: Character

A character must *always* have something equipped in the main-hand slot. All other slots are optional.

(You may or may not want to represent this in your model.)

Project Overview: Characters and Jobs

Characters can switch between multiple *jobs*

Each job has its own level.

In the example here, this character hasn't played all jobs.

Current job is determined by item equipped in main-hand slot.

Project Overview: Characters and Jobs

Each job has an independent number of experience points, and a threshold to reach the next level.

Project Overview: Character Currency

Characters have a variety of currencies

- Currencies have caps
- One currency has 2 caps: total cap & weekly cap

Project Overview: Inventory

- Inventory consists of a number of slots
- Each slot contains a *stack*: copies of identical items
  - White number indicates number of copies
- Position is significant

Project Overview: Items

Items have various properties:

- Name
- Stack size (how many can be in a single stack; can be as low as 1)
- Sale price (optional)

Various types of items:

- Gear
- Weapons
- Consumables

Project Overview: Gear & Weapons

Equippable gear, including weapons, has some additional properties:

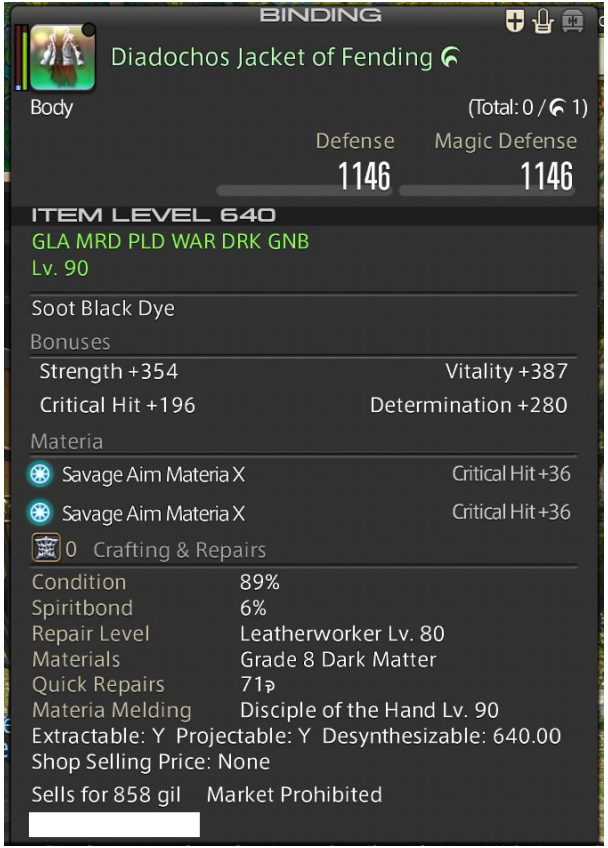
- Slot (here, Body)
- Item Level (640)
- Allowed jobs (GLA MRD PLD …)
- Required level (90)
- Stat bonuses (Strength +354, etc.)

There's a lot of additional stuff in the screenshot that you will not need to include in your model (condition, spirit bond, repair level, etc.)

Project Overview: Gear

Equippable gear properties *not* shared by weapons:

- Defense
- Magic Defense



Project Overview: Weapons

Weapons have a few specific properties as well:

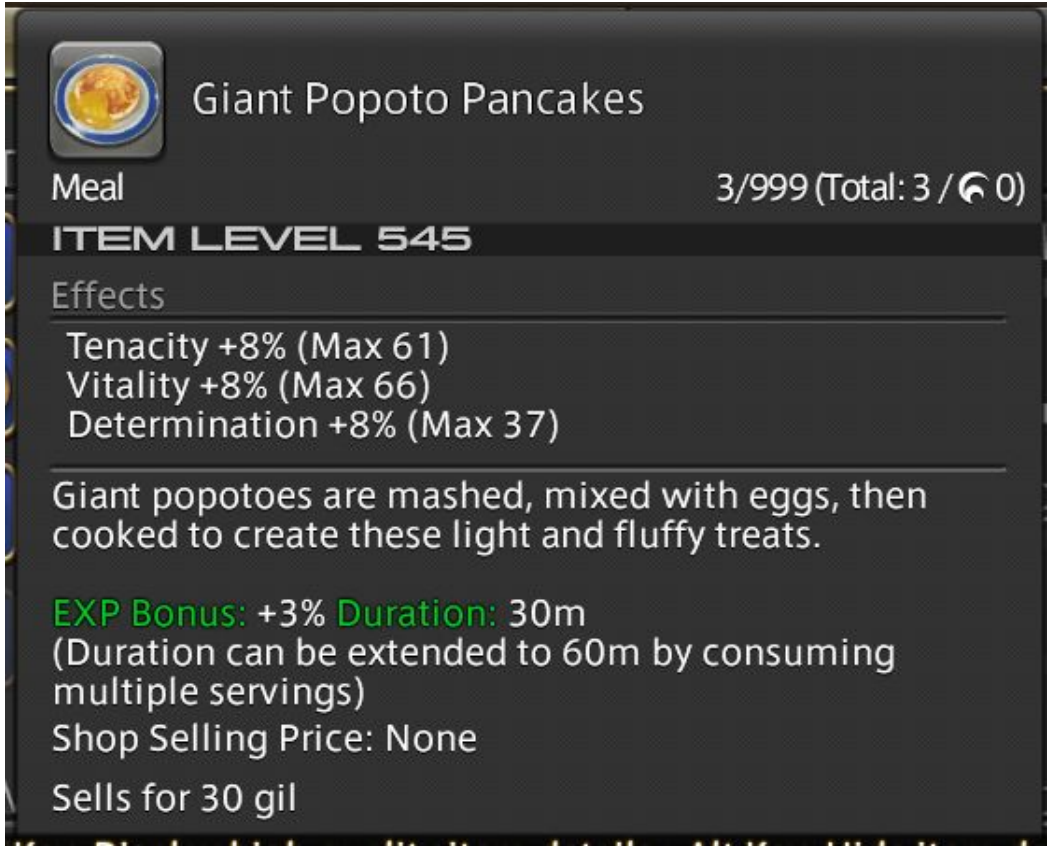
- Physical Damage
- Auto-attack
- Delay

Finally, note that weapons are the only items that can be equipped in the main-hand slot, and only weapons can be equipped in this slot.

Project Overview: Other Items

Consumable items have additional properties:

- Item Level
- Description
- Stat bonuses
  - Note max bonus!
  - EXP bonus and duration are always 3% and 30 minutes, respectively



Looking Ahead

Next Time: Physical Modeling

We're going to start applying these theoretical ideas to an actual database.

Topics:

- Installing and configuring MySQL Server, MySQL Workbench.
- Designing tables based on logical model
- Creating tables in SQL
- Defining constraints

For Next Week

To prepare for lecture next week:

- Download MySQL 8
  - 8.4.x; let's avoid 9.0 "Innovation" releases for the moment
- Download MySQL Workbench.
- Download but please do not install yet.
- Bring your laptop to lecture!