

FPC Final Requirements Sheet

Preliminary Info

Module Types

- View
- Component
- Store
- Module

Module List

Views

- ☒ ~~FPC View~~

Components

- ☒ ~~IRPTU Addition Menu~~
- ☒ ~~IRPTU Removal Menu~~
- ☐ Save Slots Menu

Stores

- ☒ ~~Loaded Factory Store~~
- ☐ User Data Store

Modules

- ☐ Calculators
- ☐ Validators
- ☐ Output
- ☐ API

Relationships

- Each view contains its own controller
- No controllers know about, or talk to, each other
- The views bind to store data exposed by the controllers
- The controllers directly talk to the stores
- The controllers refresh the views

- The stores talk to the modules
-

Per-Module Requirements

FPC View

View

- ☐ Production chain visualization via nodes
 - ☐ IRPTU data
 - ☐ The node's demand
 - ☐ Visual link to the other "child" nodes that require the given "parent" node
 - ☐ The portion of the parent node's demand required by each child node
 - ☐ The portion of the parent node's demand required by the user
 - ☐ Crafter data
 - ☐ The # of crafters required to meet the node's demand
- ☐ The various buttons used to control the loaded factory
 - ☐ Clear Factory button
 - ☐ Save Changes button (only displayed when user is logged in)
- ☐ Uses the various menus used to control the loaded factory
 - ☐ IRPTU Addition menu
 - ☐ IRPTU Removal menu

Controller

- ☐ Visual output generation/refresh methods
 - ☐ "Factory Loaded" method; used to call the more specific methods for generating the visual output required to display the loaded factory
 - ☐ "Factory Updated" method; maybe not necessary, used to distinguish between when a factory is being loaded vs. updated in terms of what visually needs to be refreshed/generated
 - ☐ Node generation method

IRPTU Addition Menu

View

☐ Controls

- ☐ Control for selecting the ID of the item to add
 - ☐ Will only be able to select from a list of valid item IDs
 - ☐ List will be searchable
 - ☐ ** List will contain images for each ID
- ☐ Control for selecting the amount of each item to add
 - ☐ Will be a number selector
 - ☐ Will only be able to select uints
 - ☐ The time unit of the item being added will be shown for more clarity
- ☐ ** Control for selecting the time unit of the item to be added
 - ☐ Will only be able to select from a list of valid time units

Controller

☐ Bindings

- ☐ Binding to the list of valid IDs exposed by the loaded factory store

☐ Functionality

- ☐ Form that takes the input from the controls
- ☐ Call to the IRPTU addition method of the loaded factory store

IRPTU Removal Menu

View

☐ Controls

- ☐ Control for selecting the ID of the input to remove
 - ☐ Will only be able to select from a list of valid inputs
 - ☐ List will be searchable
 - ☐ ** List will contain images for each ID
- ☐ Control for selecting the amount of the input to remove
 - ☐ Will be a number selector
 - ☐ Will only be able to select uints
 - ☐ Will be bound to an inclusive range between 1, and the total amount of the item's user demand
 - ☐ The time unit of the item being removed will be shown for more clarity

Control for selecting the time unit of the item to be removed

- ☐ Will only be able to select from a list of valid time units

Controller

☐ Bindings

- ☐ Binding to the list of user demand inputs exposed by the loaded factory store

☐ Functionality

- ☐ Form that takes the input from the controls
- ☐ Call to the IRPTU removal method of the loaded factory store

Loaded Factory Store

☐ Data

- ☒ ~~The object representing the currently-loaded factory~~
- ☒ ~~A list of the data pertaining to items required by user demand~~
 - ☒ ~~The list is refreshed whenever the factory is updated~~
- ☒ ~~The time unit type of the loaded factory~~
- ☒ ~~A list of valid item IDs~~
- ☐ ** A list of images corresponding to each item

☐ Methods

- ☒ ~~Clear factory~~
- ☐ Load factory
- ☒ ~~Handle factory updated~~
- ☒ ~~Time unit conversion method~~
- ☒ ~~Exposing the IRPTU Addition method from the calculators module~~
- ☒ ~~Exposing the IRPTU Removal method from the calculators module~~

Output Module

☐ Methods

- ☒ ~~IRPTU Addition method~~
- ☒ ~~IRPTU Subtraction method~~
- ☒ ~~Validation method for addition method~~
- ☒ ~~Validation method for subtraction method~~
- ☒ ~~User Demand Parse method~~
- ☒ ~~Time Unit Recalculation method~~

- ☐ Stuff for crafter data

Calculators Module

☐ Methods

- ☒ ~~IRPTU calculation method (handles removal and addition)~~
- ☒ ~~User Demand Parse method~~
- ☒ ~~Time Unit Recalculation method~~
- ☐ Stuff for crafter data

Validators Module

☐ Methods

- ☒ ~~ID validation~~
- ☒ ~~Recipes validation~~
- ☐ Stuff for crafter data

Recipes Module

☒ Data

- ☒ ~~Currently-loaded recipe set~~
- ☒ ~~List of valid IDs~~

Module Test Requirements

Output Module

☐ Valid input tests

- ☐ IRPTU Addition tests
 - ☒ ~~Addition on empty chain~~
 - ☒ ~~Full output test~~
 - ☒ ~~User demand output test~~
 - ☒ ~~Interm. demand output test~~
 - ☒ ~~Addition on populated chain~~
 - ☒ ~~Full output test~~
 - ☒ ~~User demand output test~~
 - ☒ ~~Interm. demand output test~~
- ☒ ~~IRPTU Subtraction tests~~
- ☐ IRPTU Add + Sub tests

☐ Add and remove tests

Old Requirements Sheet

General Requirements

- ** Production Calculator
 - ** URPS I/O
 - Configurability
 - Crafter Data
- ** User System
 - CRUD capabilities for user data generated by calculators
 - Requires decent performance
- Configuration Menu
 - ** Ability to configure crafter data
 - ** Ability to configure time units
 - Shouldn't be a view
- Train Throughput Calculator
 - ** Throughput I/O
 - Configurability

Specific Requirements

/**

Next Moves:

1. Answer the last 3 questions
2. Draft a definitive requirements sheet for the logged-out controllers/modes/modules logic
3. Draft a general requirements sheet for the views
4. Do use case tracing for the logged-in user requirements

5. Do use case realizations
6. Answer all the questions that result from those
7. Draft a definitive requirements sheet for the views/controllers

*/

Production Calculator

Controls

- Save Slots Menu (When logged-in)
 - Load button
 - Save button
 - Visually-enforce max save count
- IRPTU Input Menu
 - Addition Mode
 - Removal Mode
 - ☒ ~~How will removal be handled by default~~
 - ☒ ~~How will removal be handled in the case where a node is an input, but is also a requirement for a different node~~
- Main View
 - IRPTU Output
 - ☐ ~~Style of output?~~
 - ☒ ~~What specific info will be output?~~
 - Crafters Output
 - ☐ ~~Style of output?~~
 - ☒ ~~What specific info will be output?~~
 - Save Changes Button (When logged-in)
 - Clear Factory Button

Controllers

- FPC controller
 - ☒ ~~How will time units be handled~~

Model

- Loaded factory store
 - Can be cleared to an empty, but not null, state
 - Can load in an existing factory

- All edits are made to the loaded factory, so this store will expose those methods
- Thinking inline with my current view for the system, the user inputs should be exposed by the loaded factory store; this will make it possible for them to be directly displayed and edited
- Save slot data within user data store
 - Can overwrite a save
 - Can load a save
 - Max save count

Modules

- Calculators module
 - Will contain methods for performing the actual calculations powering the “production calculator”
 - IRPTU addition
 - IRPTU removal
 - **Input objects parse**
 - Time unit recalculation
- “Output” module
 - Will contain intermediary methods for validation and connection to the Calculators module
 - IRPTU addition
 - IRPTU removal
 - **Input objects parse**
 - Time unit recalculation
- Validation module
 - Will expose the validation methods used by the output module

Non-Essential

- Automatic Saving
 - Could use a web socket
- Ability to remove any node, not just inputs