# Sunward Portal Management System

**DOCUMENT INFORMATION**

| | |
|---|---|
| Prepared by | Appcentric Technologies |
| Client | Sunward |
| Project Name | Sunward Portal Management System |
| Prepared by | Arun |
| Module Name | Planning Module |
| Pages | 20 |

**Disclaimer and Confidentiality Statement**

This document is a technical reference prepared by Appcentric Technologies for the Sunward Portal Management System project. It provides detailed information on the system architecture, technical approach, implementation methodology, and related technical specifications.

This document is confidential and is intended solely for the internal use of the authorized technical team of Sunward.

No part of this document may be copied, modified, distributed, or disclosed to any third party without prior written approval from Appcentric Technologies.

# Table of Contents

## 1. Project Overview

The **Planning Module** is designed to support Sunward's monthly production planning, forecasting, and inventory balancing processes. It consolidates stock information from distributors, NAV, and internal consumption data to generate accurate mid-month and end-month simulations. These simulations help planners anticipate shortages, evaluate demand trends, align production schedules, and ensure sufficient holding stock throughout the month.

The module includes multiple tools—Simulation Mid Month, Simulation End Month, Monthly Stock Balance, and Ticket Calculation—to streamline operational planning activities. Users can upload distributor stock files, compare NAV stock balances, calculate required production tickets, and analyze generic-level item performance using dynamic filtering and drilldown views. With real-time visibility and structured workflows, the Planning Module ensures that all departments have consistent, reliable, and actionable data for inventory control and production decisions.

## 2. Objectives

The Planning Module aims to streamline and standardize the organization's forecasting, inventory monitoring, and production planning workflows. Its primary objective is to provide planners with accurate, reliable, and real-time stock visibility based on data sourced from distributors, NAV, and internal operational records. By simulating both mid-month and end-month scenarios, the module enables proactive decision-making to avoid stock shortages, overproduction, or supply chain disruptions.

Additional objectives include ensuring accurate monthly stock balancing, supporting Excel-based imports for distributor inventory, and providing automated ticket calculations for production batches. The system offers intuitive UI components, multi-level grouping, and item drilldown capability, allowing planners to analyze stock variations and consumption trends. Overall, the module enhances coordination between supply chain, production, and distribution teams by offering a centralized, data-driven planning environment..

## 3. System Modules & Key Functions

The Planning Module is composed of several sub-modules, each designed to support different stages of the forecasting, inventory balancing, and production planning process. Together, these modules help planners evaluate stock availability, analyze consumption trends, and determine production requirements for upcoming months. Below is a breakdown of the main components and their key functionalities

### 3.1. Simulation Mid Month

- Performs mid-month forecasting by analyzing Actual Consumption (AC) data from distributors and internal records.
- Provides multi-level grouping by **Generic Code → Item Number**, enabling planners to explore stock levels in granular detail.
- Offers powerful filtering options (Company, Category, Version, Steroid/Non-Steroid, Method Code) to refine simulation output.
- Generates summarized AC units, AC × 3 projections, batch size comparisons, and other metrics essential for mid-cycle planning.

### 3.2. Simulation End Month

- Calculates projected **month-end stock** based on updated consumption data and remaining quantities.
- Supports multiple simulation versions, allowing planners to compare different forecasting scenarios.
- Displays variances between mid-month and end-month simulations to identify shortages and surplus patterns.
- Enables exporting results for reporting and management review.

### 3.3. Monthly Stock Balance

This module contains two stock views:

#### 3.3.1. Distributor Monthly Stock Balance (DIST MSB)

- Allows uploading monthly stock via Excel for distributor-based inventory.
- Auto-maps Distributor Items to NAV Items and validates data before processing.
- Displays stock balance, pack size, category, internal reference, and remaining quantity.
- Includes searching, grouping, and filtering tools to manage large inventory datasets.

#### 3.3.2. NAV Monthly Stock Balance

- Shows monthly stock received directly from NAV with fields like Item Description, UOM, Pack Size, and Quantity.
- Provides consolidated view of headquarters inventory for comparison with distributor stock.
- Supports month-level filtering to track stock trends historically and for future planning.
- Helps planners reconcile NAV stock and distributor stock differences.

### 3.4 Ticket Calculation

- Calculates the number of production tickets required based on consumption, batch size, and holding stock parameters.
- Displays month-by-month ticket values (Nov–May) with automatic formulas applied to compute final values.
- Enables drilldown into monthly ticket calculation details via popup flow including holding stock situation and forecast.
- Allows planners to update and save ticket adjustments, ensuring accurate production scheduling.

### 3.5 Dynamic Form Master – Production Timing & Machine Info (Syrup)

This is the **master configuration** for how long each process takes and how many manpower/machines are needed.

**Step details**

#### 3.5.1 Define Form Type & Section

- Form Name: *Production Timing and Machine Info – Syrup*
- Section grouping:
    - General
    - Syrup Simplex
    - Syrup Preparation
    - Primary Packing / Machine Filling
    - Secondary Packing
    - Other Process
- Each section becomes a "block" of timing parameters used when generating the planning schedule.

#### 3.5.2 Link to Planning Master

- Each Dynamic Form Master record is linked to a **Planning Master** (e.g., product, strength, pack size, method, line).
- When planning is created (Syrup Planning / Production schedule), system reads the **relevant**

**Dynamic Form** based on:

  - Section (e.g., Syrup)
  - Process Type (Simplex / Filling / Prep)
  - Method / Version
- This allows different products or lines to have different timings and manpower rules without changing code.

### 3.5.3 Master Data Reusability

- Once timings & machine info are configured, the same form can be used across multiple planning runs (different days / batches).
- Changes to the master affect **future schedules**, but won't usually modify already completed runs (depending on design).

### 3.5.4 Data by Section / Process (1–4)

You mentioned the following structure:

Data by Section Process

1. General
2. Syrup Simplex
3. Syrup Filling Primary to Secondary
4. Other Process

**Plus detailed breakdown:**

- Syrup Simplex
- Syrup Preparation
- Primary Packing / Machine Filling
- Secondary Packing
- Other Process

**Let's detail each.**

## 4. General Section

**Purpose**: Holds general control parameters used by planning & scheduling.

**Typical fields / logic:**

1. **Batch Identifiers**

    - Batch Size, Product Name, Pack Size, Line/Room.
    - These values are used to determine how many hours / tickets are needed.

2. **Planning Constraints**

    - Max number of batches per day.
    - Whether this product/line is allowed to run on Night Shift or only Day Shift.

3. **Shift & Calendar Rules**

    - Working hours per day (e.g., 8am–6pm or 8am–8:30pm with OT).
    - Exclusions: Friday extended lunch, public holidays, maintenance windows.

4. **Method / Version**

    - Link to "Method Code" or "Version" used in Planning.
    - Different methods may produce different timing for the same product (e.g., new process vs old one).

## 4.1 Syrup Simplex

Syrup Simplex

- Syrup Simplex/Manpower 1 hour to 2 hour
- Preparation /Hour
- Level 2 Cleaning/Hours

**Step details:**

**Syrup Simplex / Manpower (1–2 hours)**

- Defines how many hours per batch are required for Simplex activity with given manpower.
- Example: Simplex time = 2 hours if 3 manpower, 1.5 hours if 4 manpower (system may use fixed or derived value).

**Preparation / Hour**

- Time needed to prepare materials before Simplex start (weighing, staging, line check).
- This time is added at the start of the schedule for that batch.

**Level 2 Cleaning / Hours**

- Deep cleaning after certain operations (e.g., product change).
- Typically long duration, e.g. 3 hours, and may block the machine from being used for other batches during that time.

- When planner changes Level 2 Cleaning hours here, all future schedules will allocate more or less downtime between batches.

## 4.2 Syrup Preparation

**Syrup Preparation**

- Preparation - Top up to Volume/Hour
- Preparation - First Volume/Manpower
- Level 1 Cleaning/Manpower
- Level 2 Cleaning/Hours

**Preparation – Top Up to Volume / Hour**

- Defines how long it takes to top up from partial volume to full batch volume.
- Used when a batch is prepared in multiple steps (e.g., pre-charge, then top-up).
- The planner uses this to compute overlapping of preparation vs filling.

**Preparation – First Volume / Manpower**

- Time required to prepare the **first volume** (initial charge).
- Often more complex, so this time may be higher than subsequent top-ups.
- Manpower count influences capacity (more people can reduce the time).

**Level 1 Cleaning / Manpower**

- Short cleaning tasks between batches or minor changes (e.g., same product, same line).
- Typically 0.5H or small value.
- The planner inserts these short cleaning tasks between consecutive runs on the same machine.

**Level 2 Cleaning / Hours (again here)**

- Used when process demands more “full cleaning” between certain product types.
- This can override default cleaning rules for specific Syrup Preparation line.

## 4.3 Primary Packing / Machine Filling

**Primary Packing**

- Next Process Name
- Type of planning

**Machine Filling**

- Filling/Hours
- Level 1/Hours
- Level 1/Manpower
- Level 2/Hours

**Next Process Name / Type of Planning**

- "Next Process Name" indicates the process that comes right after Primary Packing (e.g., Secondary Packing).
- "Type of Planning" – whether planning is **by volume**, **by tickets**, or **by batch count**.
- This helps the planning engine know how to push the timeline to the next stage.

**Machine Filling – Filling / Hours**

- Rate at which filling happens, usually linked to bpm (bottles per minute).
- Example from your note:
  - MH 60 ml: 80–85 bpm
  - MH 120 ml: 75–80 bpm
- The system converts bpm + total units → total filling hours.

**Level 1 / Hours & Level 1 / Manpower**

- Short cleaning time between lots on the same filling line.
- Manpower influences how quickly change-over can be completed.

**Level 2 / Hours**

- Major cleaning or full sanitation after batch / product change.
- Planner must insert this into schedule to ensure line compliance.

## 4.4 Secondary Packing

- Process Name
- Next Process Name

**Process Name**

- Secondary packing step (cartoning, bundling, overwrapping, etc.).
- Tied to specific machine / line.

**Next Process Name**

- Defines further downstream activity (e.g., QC Hold, Dispatch).
- Used in scheduling engine to know where to end the timeline or which module to trigger next (e.g., Ticket consumption / shipment).

**Timing Parameters**

- Can have fields similar to filling: Speed, Units / Hour, Manpower.
- These values control how long secondary packing will take for each batch.

## 4.5 Other Process

- Other Jobs Information
- Location of process
- No of Manhours / Hours

**Other Jobs Information**

- Used for non-standard or "ad-hoc" tasks (e.g., rework, repacking, sampling, small transfers).
- Text description for planner & production team.

**Location of Process**

- Which room / line / building this job takes place in.

**No of Manhours / Hours**

- Duration and manpower for that process.
- Used to block capacity in the schedule even though it's not a standard Syrup / Filling step.

# 5. Syrup Planning Logic

## 5.1 Generate Schedule (Gantt / Timeline)

**Input**

- Planning Master entry (product, batch, date).
- Dynamic Form Master parameters (all section timings).
- Calendar & team availability.

**Process**

- Calculate start/end time for each step:
  - Prep → Simplex → Filling → Cleaning → Secondary.
- Ensure no overlap on same machine or room beyond allowed capacity.
- Insert breaks and cleaning automatically.

**Output**

- Visual **timeline (Gantt-like)** showing:
  - Start / End per step.
  - Resource (line, room) allocation.
  - Color-coded for Normal / Cleaning / Break.

**Adjustment**

- Planner can manually adjust start time or assign to another day/line.
- Regenerate schedule to see updated impact.

## 5.2 Production Sync

**Purpose**

- Push final planned schedule to **Production Execution** system.
- Ensures operators see the same times & sequences defined in planning.

**Sync Data**

- Batch ID, Product, Line/Room.
- Start / End times per process.
- Required manpower & cleaning indicators.

**Local Database Storage Logic (Manual Insert)**

- After the schedule is generated, the system **stores a copy of the schedule in the local database** for tracking, reporting, and manual adjustments.
- In **manual insert mode**, certain key fields (e.g., actual manpower, adjusted start time, remarks) can be filled in or overridden by the planner or production supervisor.
- This local storage acts as the **single source of truth** for what was actually communicated to production, even if external systems are not available.

**Week-of-Month Handling (Start with Monday)**

- Each synced record is also tagged with a **WeekOfMonth** value to support weekly reports and KPIs.
- Business rule:
    - **Weeks are calculated starting on Monday.**
    - Example:
        - 1st Monday → Week 1
        - 2nd Monday → Week 2
        - … up to Week 4 / Week 5 depending on the calendar.
- The **WeekOfMonth** is computed from the schedule date using this logic and stored together with the batch record in the local database.
- This allows dashboards and reports to group production by **Week 1, Week 2, Week 3, etc.**, aligned with operational reality (Monday–Sunday weeks), instead of just calendar date.

## 6. Functional Components

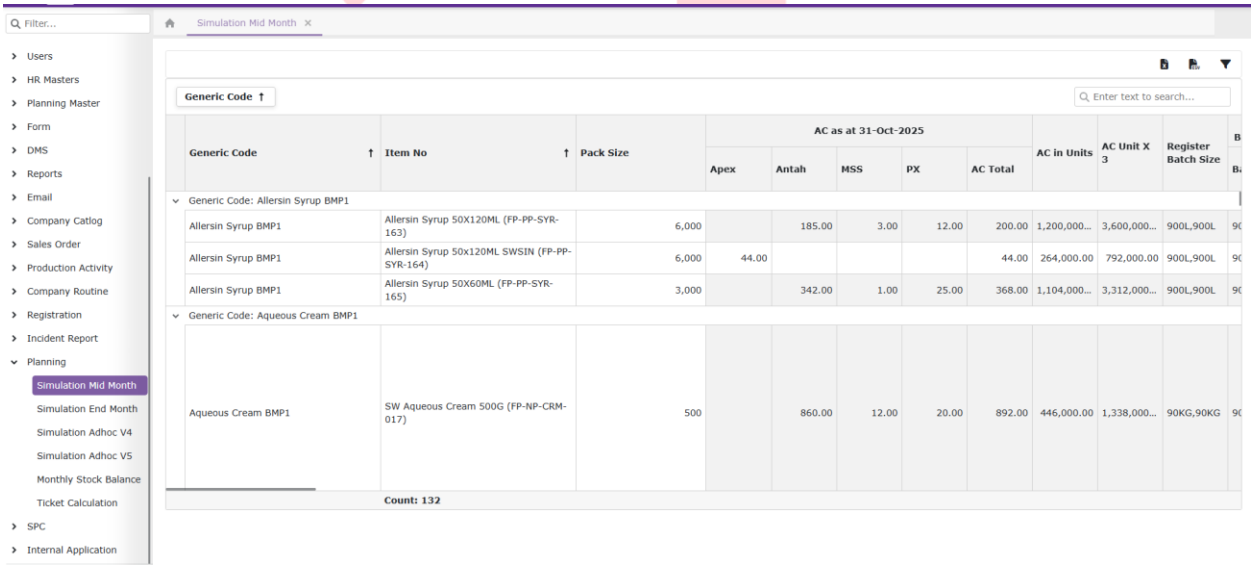
The Planning Module consists of multiple interconnected components that work together to deliver forecasting, stock balancing, and production planning capabilities. Each component handles a specific part of the planning process, ensuring that planners have accurate and structured information for decision-making. The following sections describe the core functional components of the module

### 6.1 Simulation Mid Month Component

This component is designed to generate **mid-month inventory simulations** by analyzing partial-month consumption and available stock.

**Key Functionalities**

- Allows planners to apply advanced filters (Company, Category, Version, Method Code, Steroid/Non-Steroid) to refine the simulation scope.
- Aggregates data using a hierarchical structure: Generic → Generic Code → Item No → Pack Size.
- Displays AC totals, AC in units, AC × 3 projection, and batch size metrics used for forecasting.
- Provides expandable rows that help users drill down into item-level consumption details for accurate mid-cycle evaluation.

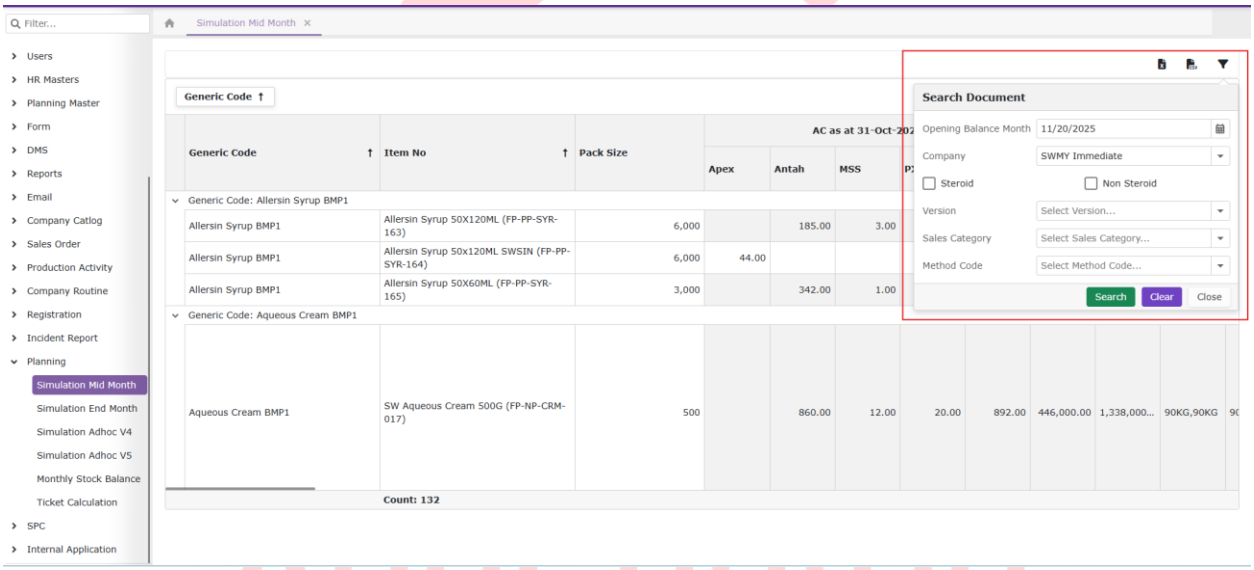


## 6.2 Simulation End Month Component

This component performs **complete end-of-month forecasting** using updated consumption and stock data.

**Key Functionalities**

- Projects the estimated closing balance by combining mid-month actuals with forecast consumption for remaining days.
- Supports multiple simulation versions so planners can compare different forecasting methods.
- Highlights critical variations between expected and actual consumption to identify over/underperforming items.
- Allows exporting simulation results for reporting to management or supply-chain teams.




## 6.3 Monthly Stock Balance Component

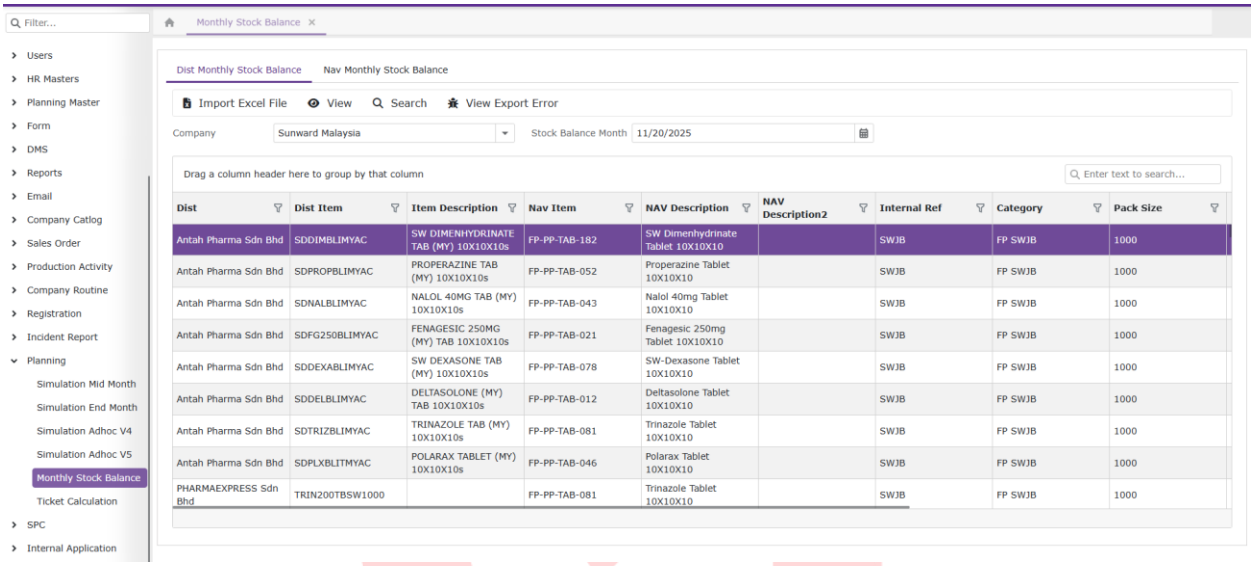
This functional component consolidates stock-related data from distributors and NAV for a unified stock overview. It consists of two sub-components.

### 6.3.1 DIST Monthly Stock Balance

**Functional behaviors:**

- Allows planners to upload monthly distributor stock files (Excel format) and validates them upon import.
- Maps distributor items to NAV items and flags inconsistencies for manual correction.

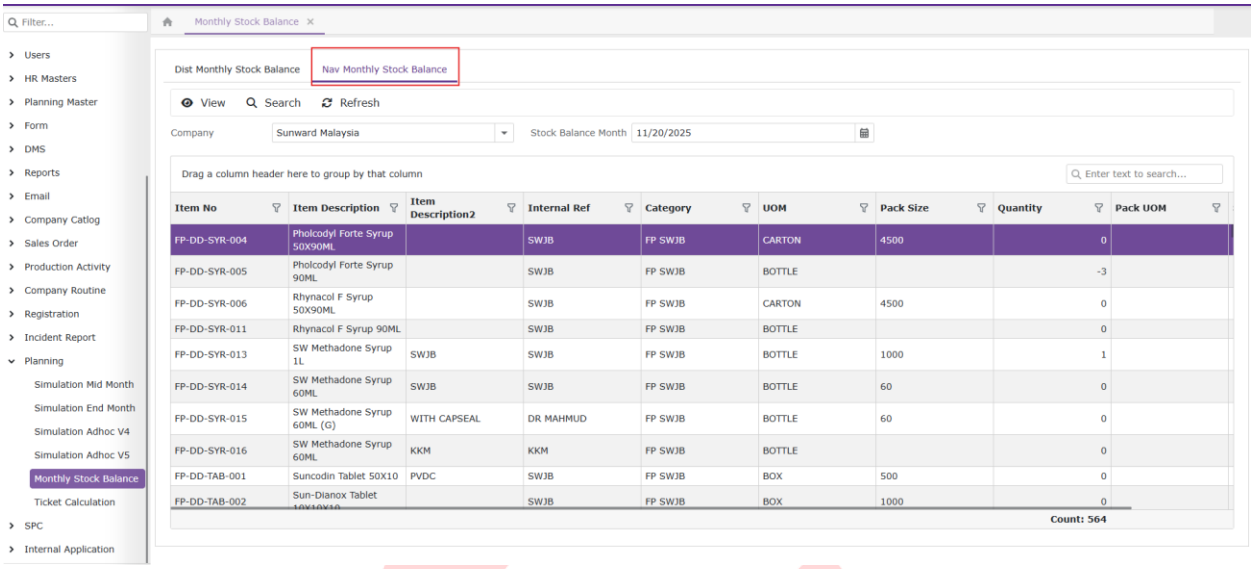
- Displays stock balance details including Distributor Item, Item Description, Category, Pack Size, and Remaining Quantity.
- Supports search, grouping, and filters to help planners navigate large distributor inventories.




### 6.3.2 NAV Monthly Stock Balance

**Functional behaviors:**

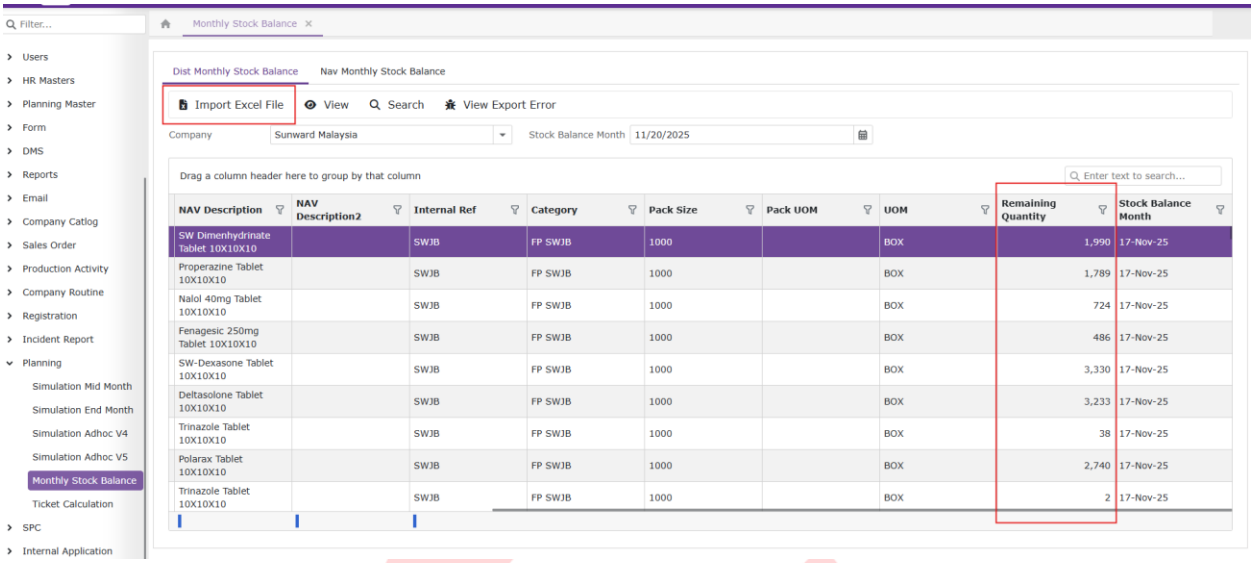
- Synchronizes NAV item stock for a selected month and presents it alongside distributor data.
- Provides accurate internal stock details such as Item No, NAV Description, Category, UOM, Pack Size, and Quantity.
- Facilitates comparison between NAV and distributor stock quantities for consistency checks.
- Offers monthly tracking to help planners understand stock movement and trends.




### 6.3.3 Import Excel File & Update Remaining Quantity Column

This feature allows planners to upload monthly distributor stock using a predefined Excel template. When an Excel file is imported, the system automatically reads item details, maps distributor item codes with NAV item records, and populates the corresponding stock balance fields. Any mismatched or unmapped items are clearly highlighted for manual correction, ensuring data integrity.

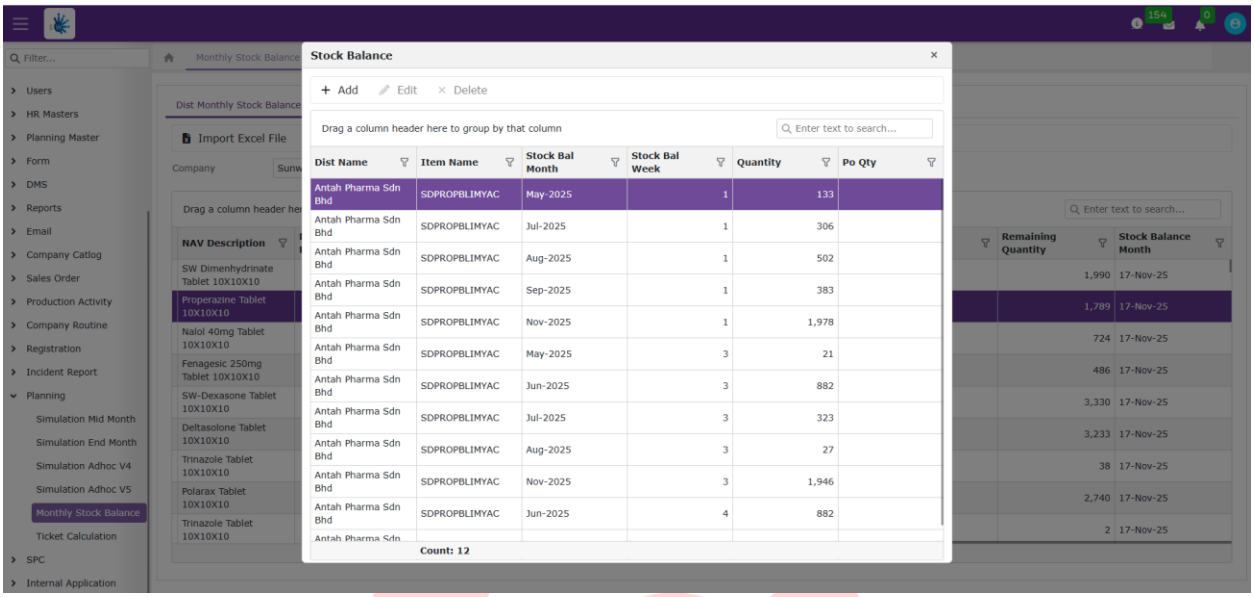
The module includes logic to calculate and **update the Remaining Quantity column** based on imported values, NAV stock, and existing system records. The system displays results in a grid for final review, allowing planners to validate stock quantities before finalizing the monthly stock balance. This import workflow ensures accuracy, eliminates manual data entry errors, and speeds up the monthly planning cycle.




### 6.3.4 Monthly Stock Balance – View (Drilldown Details Popup)

The **View** popup provides a detailed breakdown of the selected stock record, allowing planners to inspect month-specific stock behavior. It includes information such as **Stock Balance (Month), Stock Balance (Week), PO Quantity, Remaining Quantity**, and other item-level details. This detailed view helps planners analyze discrepancies between distributor data and internal system values.

Users can navigate through grouped rows, compare historical stock entries, and validate individual batch-level or week-level information. This drilldown functionality supports decision-making by giving deep insights into the exact stock situation behind summary values shown in the main grid. It helps ensure accuracy before proceeding with simulations or ticket calculations.




## 7. Business Logic

The Planning Module performs several interconnected calculations and validations to ensure accurate forecasting, stock balancing, and ticket computation. Each sub-module follows defined business rules that govern how data is processed, displayed, and validated. This section explains the core logic behind simulations, stock imports, NAV comparisons, and ticket calculation workflows.

### 7.1 Simulation Mid Month – Business Logic

**AC (Actual Consumption) Derivation**

- AC is aggregated from distributor sales for the first half of the month.
- AC Units = Σ(Distributor Sales Quantity × Pack Size).
- "AC × 3" is a projection formula used to estimate total month consumption:
  **Projected Consumption = (AC Units ÷ Days Passed) × 30**

**Stock Projection Logic**

- Opening stock balances + mid-month stock updates – AC Units = Remaining Stock
- If Remaining Stock < threshold (batch size), system flags the item for review.

**Grouping & Hierarchy**

- Items grouped as **Generic → Generic Code → Item No → Pack Size**, enabling multilevel forecasting.

**Filter-based Query Logic**

- Query structure dynamically changes based on (Company, Category, Version, Steroid, Method Code).

## 7.2 Simulation End Month – Business Logic

**End-Month Projection Formula**

- Uses updated consumption data for the second half of the month.
- Formula:
  **Expected Closing = Opening Balance + Inbound Qty – Total Monthly Consumption**

**Versioning Logic**

- Each simulation version maintains separate calculated data sets without overwriting previous runs.

**Variance Logic**

- Variance = Projected End-Month – Actual End-Month
- Highlights MTM (Mid-to-Month) discrepancies.

**Data Refresh Cycle**

- Automatically refreshes NAV and distributor inputs before running final simulation.

## 7.3 Monthly Stock Balance – Business Logic

### *6.3.1 DIST Monthly Stock Balance Logic*

**Excel Import Validation**

- Validates template structure.
- Ensures Item Code, Description, Pack Size, and Qty fields are correctly formatted.
- Rejects rows with missing or invalid NAV mapping.

### Remaining Quantity Calculation

Formula:
**Remaining Quantity = Opening Stock – Sales + Adjustments**

### Distributor-to-NAV Mapping Logic

- Mapping table determines equivalent NAV Item.
- Rows flagged when Distributor Item does not exist in mapping table.

### *6.3.2 NAV Monthly Stock Balance Logic*

**NAV Data Pull**

- Retrieves month-ending stock from NAV tables or OData service.
- Ensures each item has a single, validated NAV reference.

**Stock Comparison Logic**

- Compares NAV quantity with imported distributor quantity:
  **Stock Difference = NAV Qty – DIST Qty**

**Category-Level Aggregation**

- Creates summary totals grouped by Category, Pack Size, or Item Class.

**Historical Tracking Logic**

- Saves month-level stock snapshots for audit and review.

## 7.4 Ticket Calculation – Business Logic

**Ticket Computation Formula**

- Base Formula:
  **Number of Tickets = (AC Units – Holding Stock) ÷ Batch Size**
- System rounds values based on setting:
- Round Down
- Round Up
- Nearest Whole Ticket

**Top-Up Logic**

- If Total Tickets < Minimum Threshold:
  **TopUpTickets = MinimumThreshold – TotalTickets**

**Final Split Ticket**

- Applies split rules when batch sizes differ between months.
- Ensures each production run meets required minimum order quantity (MOQ).

**Update & Save Logic**

- On save, the system:
- Validates batch size
- Updates monthly ticket table
- Recalculates holding stock forecast
- Logs update history for audit

## 7.5 Production Sync

**Local Database Storage Logic**

- All synced schedule records are stored in a local database table (e.g., ProductionSchedule),production system exists.
- In manual insert mode, records can be added or edited directly in this local table for:
- Additional tasks
- Revised times
- Extra cleaning jobs, reworks, etc.
- The local schedule acts as the source of truth for what was communicated to production.

**Week-of-Month Calculation (Starting Monday)**

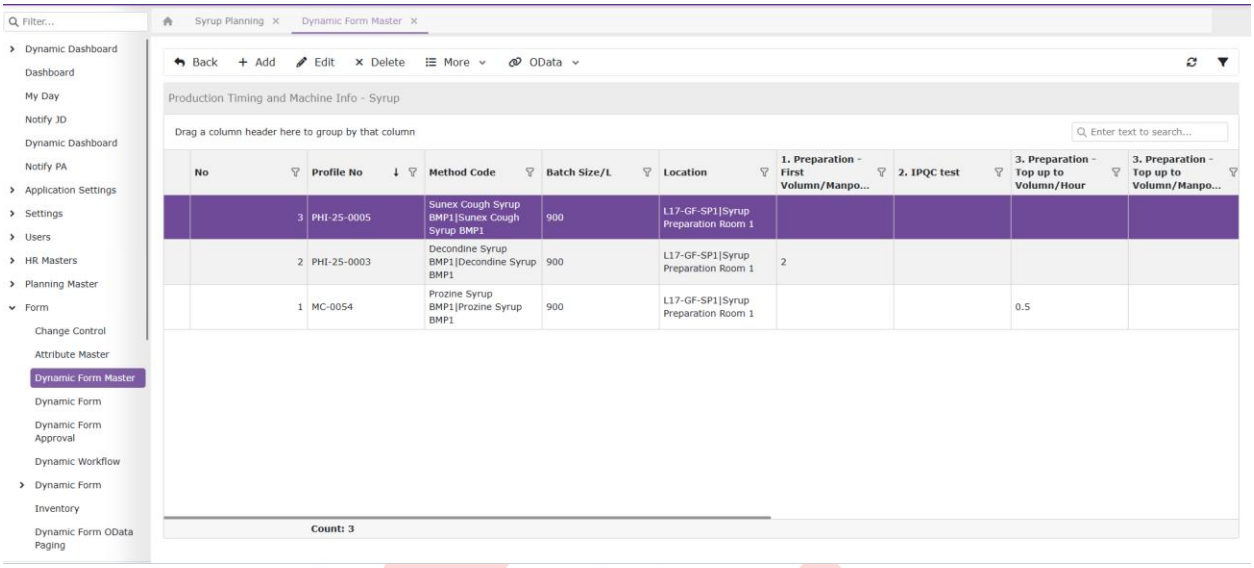
- Each synced record is tagged with a WeekOfMonth value based on the planned date
- This value is used in Generate Schedule,

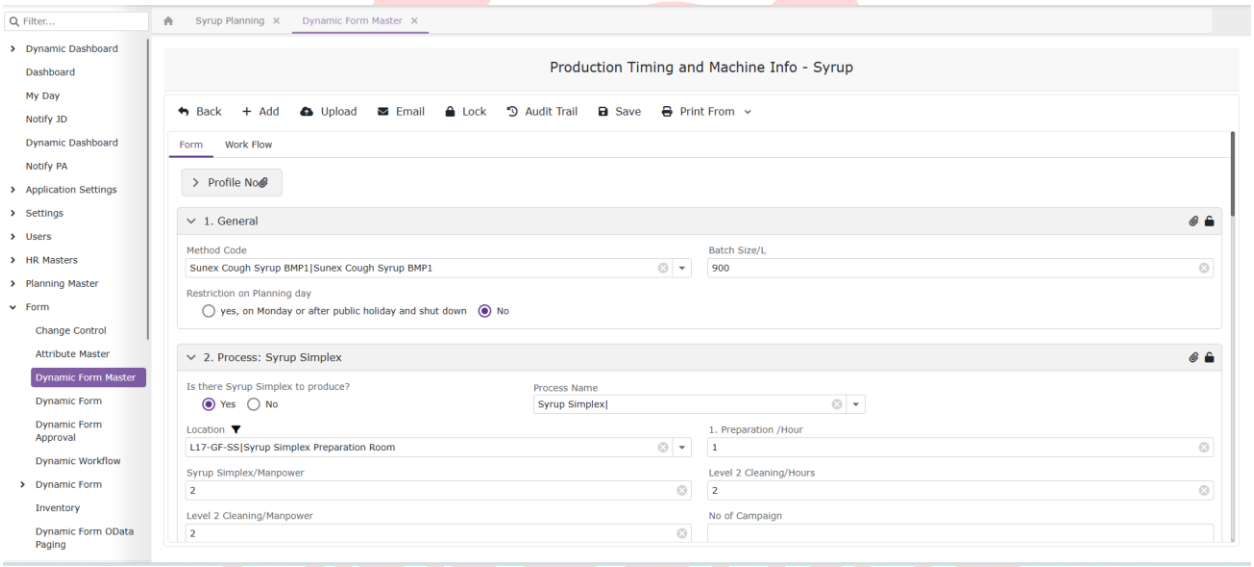
**Validation & Sync Rules**

- Start Time < End Time
- Product are not empty

**Dynamic Form Master – List Screen**

- Method Code determines:
- Sequence of processes
- Syrup type (Simplex, Preparation, Filling, Other)




## Hierarchical Process Logic (Dynamic Form Master – Syrup)

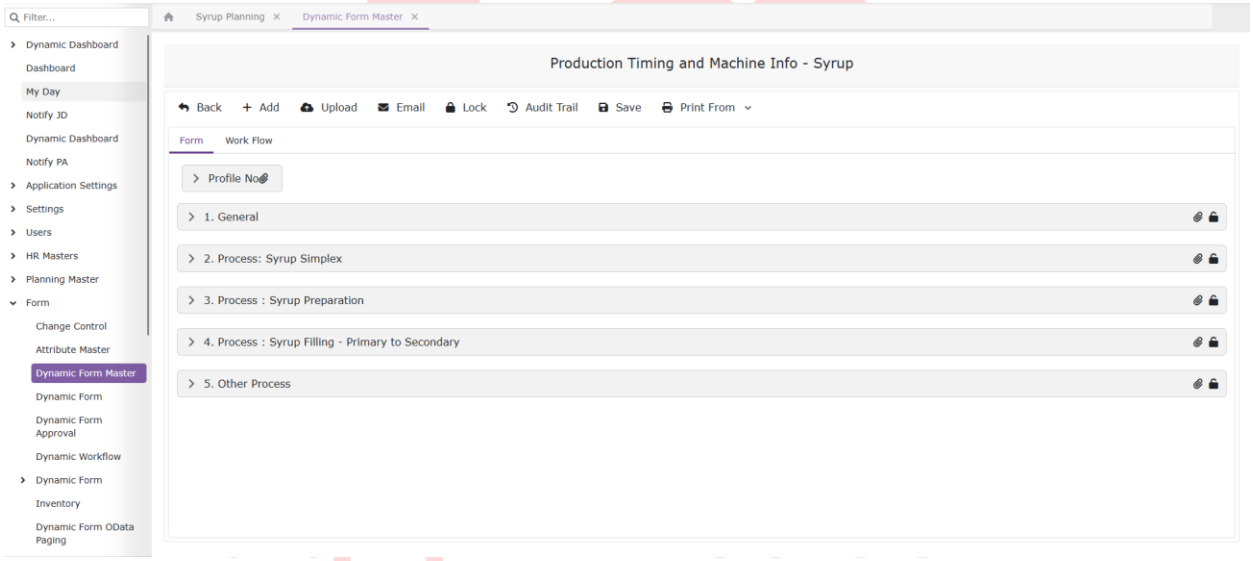
### How the System Decides the Order of Processes

The Dynamic Form Master contains **five major sections**, but NOT every method code uses all sections:

1. **General**
2. **Process: Syrup Simplex**
3. **Process: Syrup Preparation**
4. **Process: Syrup Filling – Primary to Secondary**
5. **Other Process**

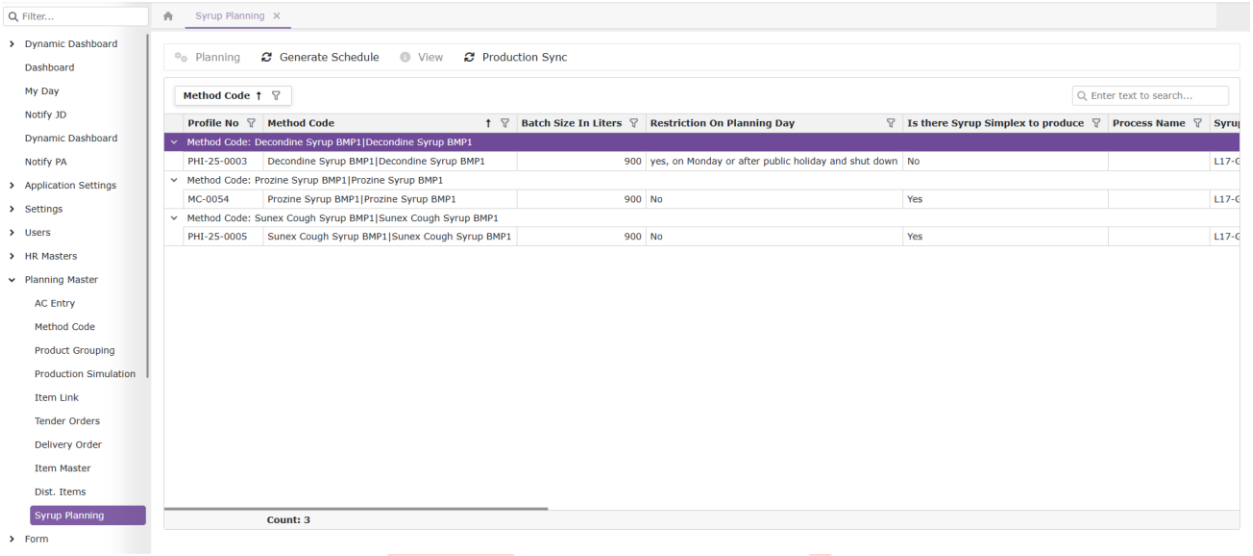
Your rule:

If “Other Process” exists for the selected Method Code, it must be executed FIRST before the normal process flow.
If no “Other Process”, follow the normal sequence.



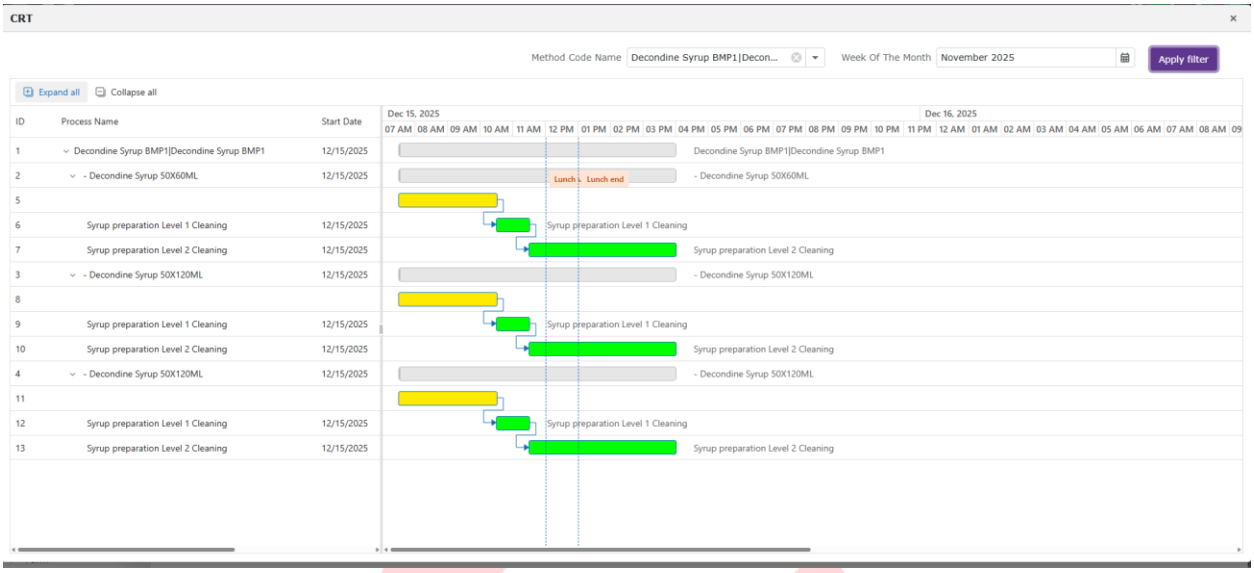

**Syrup Simplex**

- Syrup Simplex is always first step
- Duration = (Preparation Hour
- Cleaning must be inserted between each batch automatically.



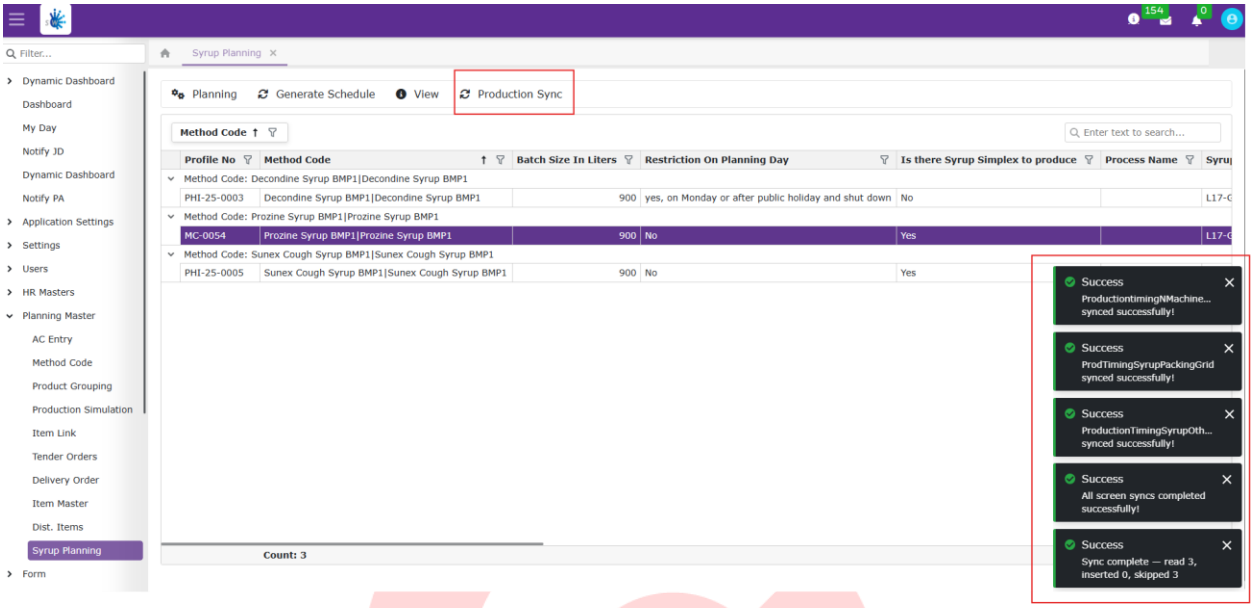

**Generate Schedule (Gantt / Timeline)**

1. **Input**
   - Planning Master entry (Method Code, date).
   - Calendar & team availability.
2. **Process**
   - Calculate start/end time for each step:
     - Prep → Simplex → Filling → Cleaning → Secondary.
   - Ensure no overlap on same machine or room beyond allowed capacity.
   - Insert breaks and cleaning automatically.
3. **Output**
   - Visual **timeline (Gantt-like)** showing:
     - Start / End per step.
     - Resource (room) allocation.
     - Color-coded for Normal / Cleaning / Break.




## Production Sync

1. **Purpose**
    - Push final planned schedule to **Production Execution** system.
    - Ensures operators see the same times & sequences defined in planning.
2. **Sync Data**
    - Method code ID
    - Start / End times per process.
    - Required manpower & cleaning indicators.




## 8. Summary

- The Planning Module relies heavily on monthly input from **Distributors**, **NAV System**, and **Internal Calculations**.
- All stock flows link through **Item**, **Generic Code**, and **NAV Mapping** tables.
- Each simulation version is stored separately to support multi-scenario forecasting.
- Ticket calculation tables use simulation outputs to determine production requirements.

The Planning Module stores all forecasting, stock balance, and ticket calculations in a set of structured tables that link distributor data, NAV stock, and internal planning logic. Distributor stock imports, NAV stock synchronization, simulation outputs, and ticket calculations all connect through consistent item and generic code mappings. Each simulation version and monthly calculation is stored independently to ensure historical tracking and scenario-based planning. Overall, the database structure ensures accurate forecasting, reliable reconciliation, and seamless production planning across all Planning Module components.