

FlowMatic-SOLO R2 Admin Panel - Appendix A: Enhanced Features

Date: June 28, 2025

Version: 2.0 - Appendix A

Status: Additional Requirements

Purpose

This appendix captures additional features and requirements identified during architecture review. These enhancements will be integrated into the Admin Panel development without requiring updates to the main architecture document.

Enhanced Features

1. Global Language Settings

1.1 Language Selection Behavior

- **Admin Control:** Language selection available **only in Admin Panel Settings tab**
- **Global Application:** Selected language applies **system-wide** across all interfaces
- **System-Wide Coverage:**
 - Customer Kiosk interface
 - Agent Terminal interface
 - Monitor Display (TV)
 - Customer Counter Display
 - Admin Panel interface
- **Individual Page Language:** Available **only during development** for testing purposes
- **Production Behavior:** All interfaces follow admin-selected language

1.2 Implementation Details

javascript

// Settings API Enhancement

PUT /api/admin/settings/language

```
{
  "language": "en|th|hi",
  "applyGlobally": true
}
```

// Global Language Setting

```
{
  key: "system.language",
  value: "en", // en, th, hi
  type: "select",
  category: "system",
  description: "Global system language",
  validValues: ["en", "th", "hi"],
  applyScope: "global"
}
```

1.3 User Interface

⚙ Settings Tab - System Configuration Section:

SYSTEM LANGUAGE	
Current Language: English	
Language: [English ▼]	
└ English	
└ ไทย (Thai)	
└ हिन्दी (Hindi)	
⚠ This will change language across all system interfaces	
[Apply Language Change]	

1.4 Technical Implementation

- **Database:** Store global language in settings table
- **Real-time Update:** Broadcast language change via Socket.IO to all connected clients
- **Interface Reload:** All interfaces automatically reload with new language

- **Persistence:** Language setting persists across system restarts

2. Automated System Reset

2.1 Reset Configuration

- **Schedule Options:**
 - **Daily Reset:** Set specific time (e.g., 12:00 AM)
 - **No Reset:** Disable automatic reset
- **Reset Scope:**
 - Clear all active queues
 - Reset service counters to starting numbers
 - Clear ticket history for the day
 - Reset agent session states
 - Maintain configuration and settings

2.2 Reset Settings Interface

⚙ Settings Tab - System Configuration Section:

DAILY SYSTEM RESET

☐ Enable Daily Reset

☐ Disable Reset

Reset Time: [00:00] (24-hour format)

RESET SCOPE:

☒ Clear all queues

☒ Reset service counters

☒ Clear daily ticket history

☒ Reset agent sessions

☒ Keep settings & configuration

Last Reset: 2025-06-28 00:00:00

Next Reset: 2025-06-29 00:00:00

[Save Reset Schedule]

[Reset Now]

2.3 Implementation Details

javascript

```
// Reset Configuration Settings
{
  key: "system.daily_reset_enabled",
  value: "true",
  type: "boolean",
  category: "system"
}

{
  key: "system.daily_reset_time",
  value: "00:00",
  type: "time",
  category: "system"
}

// Manual Reset API
POST /api/admin/system/reset
{
  "resetType": "full", // full, queues_only, counters_only
  "confirm": true
}

// Scheduled Reset Function
function performDailyReset() {
  // 1. Clear all tickets in 'waiting', 'called', 'serving' states
  // 2. Reset service current_number to range_start
  // 3. Clear today's events (keep audit trail)
  // 4. Reset agent sessions
  // 5. Log reset event
  // 6. Broadcast reset notification to all clients
}
```

2.4 Reset Notifications

- **Pre-Reset Warning:** 5-minute notification to all connected clients
 - **Reset Execution:** System pause during reset (30-60 seconds)
 - **Post-Reset Confirmation:** Success notification with new state
 - **Error Handling:** Rollback capability if reset fails
-

3. Service Preset & Queue Pre-Population

3.1 Service Preset Configuration

- **Per-Service Settings:** Each service can have its own preset configuration

- **Preset Parameters:**
 - **Start Number:** Starting ticket number for preset
 - **Queue Count:** Number of tickets to pre-create
 - **Preset Mode:** Testing, Recovery, or Custom
- **Use Cases:**
 - **Testing:** Pre-populate queues for system testing
 - **Recovery:** Restore queues after system reset
 - **Demo:** Create realistic demo data

3.2 Service Preset Interface

🔑 Services Tab - Enhanced Service Management:

SERVICE PRESETS

Service: General Service (A)

PRESET CONFIGURATION

Start Number: [101]

Queue Count: [50]

End Number: [150] (auto-calculated)

Preset Mode: [Testing ▼]

Testing

Recovery

Custom

⚠️ This will create 50 tickets (A101-A150)

[Apply Preset] [Clear Preset] [Save Config]

PRESET HISTORY

• 14:30 - Applied Testing preset: A101-A150 (50)

• 12:00 - Applied Recovery preset: A201-A225 (25)

• 09:00 - Cleared all presets for daily reset

3.3 Implementation Details

javascript

```
// Service Preset Configuration
{
  serviceId: 1,
  presetEnabled: true,
  startNumber: 101,
  queueCount: 50,
  presetMode: "testing", // testing, recovery, custom
  lastApplied: "2025-06-28T14:30:00Z"
}

// Apply Preset API
POST /api/admin/services/:id/apply-preset
{
  "startNumber": 101,
  "queueCount": 50,
  "presetMode": "testing",
  "confirm": true
}

// Preset Application Function
function applyServicePreset(serviceId, config) {
  // 1. Validate number range doesn't conflict
  // 2. Create tickets in 'waiting' state
  // 3. Update service current_number
  // 4. Log preset application event
  // 5. Broadcast queue updates to all clients
  // 6. Return confirmation with created ticket range
}
```

3.4 Preset Management Features

- **Conflict Detection:** Prevent overlapping number ranges
- **Bulk Preset:** Apply presets to multiple services simultaneously
- **Preset Templates:** Save common preset configurations
- **Undo Capability:** Remove preset tickets if applied incorrectly
- **Audit Trail:** Track all preset applications

API Enhancements

Additional Admin APIs Required

javascript

// Global Language Management

GET /api/admin/language // Get current global language

PUT /api/admin/language // Set global language

// System Reset Management

GET /api/admin/system/reset-config // Get reset configuration

PUT /api/admin/system/reset-config // Update reset schedule

POST /api/admin/system/reset // Manual system reset

GET /api/admin/system/reset-status // Reset status and next scheduled

// Service Preset Management

GET /api/admin/services/:id/preset // Get service preset config

PUT /api/admin/services/:id/preset // Update preset config

POST /api/admin/services/:id/apply-preset // Apply preset to service

DELETE /api/admin/services/:id/preset // Clear service preset

POST /api/admin/services/bulk-preset // Apply presets to multiple services



Database Schema Enhancements

Additional Settings

sql

-- Global Language Setting

INSERT INTO settings (key, value, description, category) VALUES
(*'system.language'*, *'en'*, *'Global system language (en, th, hi)'*, *'system'*);

-- Daily Reset Settings

INSERT INTO settings (key, value, description, category) VALUES
(*'system.daily_reset_enabled'*, *'false'*, *'Enable automatic daily reset'*, *'system'*),
(*'system.daily_reset_time'*, *'00:00'*, *'Daily reset time (HH:MM format)'*, *'system'*),
(*'system.last_reset'*, *''*, *'Last system reset timestamp'*, *'system'*);

-- Service Preset Settings

INSERT INTO settings (key, value, description, category) VALUES
(*'system.presets_enabled'*, *'true'*, *'Enable service preset functionality'*, *'features'*);

Service Preset Table

sql

-- Service Presets Configuration

```
CREATE TABLE service_presets (  
  id INTEGER PRIMARY KEY AUTOINCREMENT,  
  service_id INTEGER NOT NULL,  
  preset_enabled BOOLEAN DEFAULT false,  
  start_number INTEGER,  
  queue_count INTEGER,  
  preset_mode TEXT DEFAULT 'testing', -- testing, recovery, custom  
  created_at DATETIME DEFAULT CURRENT_TIMESTAMP,  
  updated_at DATETIME DEFAULT CURRENT_TIMESTAMP,  
  last_applied DATETIME,  
  FOREIGN KEY (service_id) REFERENCES services(id),  
  UNIQUE(service_id)  
);
```

Task Integration

Task 1 Enhancement: Authentication & Layout

- Add language selector to admin interface
- Include global language state management

Task 6 Enhancement: Settings & Feature Flags

- Add Global Language section
- Add Daily Reset configuration section
- Include service preset enable/disable toggle

Task 3 Enhancement: Services Management

- Add Service Preset configuration to each service
- Include preset application interface
- Add preset history tracking

Task 8 Enhancement: System Maintenance

- Add manual reset capability
 - Include reset schedule monitoring
 - Add reset history and status
-

Development Impact

Additional Development Time

- **Global Language:** +2 hours (Task 1 & 6)
- **System Reset:** +3 hours (Task 6 & 8)
- **Service Presets:** +3 hours (Task 3)
- **Total Additional:** +8 hours

Updated Timeline

- **Original Timeline:** 4 days (24-28 hours)
 - **Enhanced Timeline:** 4.5 days (32-36 hours)
 - **Recommendation:** Extend Sprint 3 by half day
-

✓ Success Criteria Additions

Global Language

- ✓ Admin can change system language globally
- ✓ All interfaces immediately reflect language change
- ✓ Language setting persists across restarts
- ✓ Development language selectors are disabled in production

System Reset

- ✓ Admin can schedule daily automatic reset
- ✓ Manual reset functionality works correctly
- ✓ Reset scope is configurable and accurate
- ✓ System provides pre-reset warnings

Service Presets

- ✓ Admin can configure presets per service
 - ✓ Preset application creates tickets correctly
 - ✓ Conflict detection prevents number overlaps
 - ✓ Preset history provides audit trail
-

This appendix can be integrated into the main architecture without modification, providing clear enhancement specifications for the Admin Panel development. 🚀