INVENTION DISCLOSURE DOCUMENT

ADAPTIVE MULTI-RECIPE COOKING COORDINATION SYSTEM WITH INTELLIGENT TEMPORAL OPTIMIZATION AND CRITICALITY-AWARE TASK MANAGEMENT

Date of Documentation: October 16, 2025

Status: Confidential Invention Disclosure

I. EXECUTIVE SUMMARY

This disclosure documents a novel system and method for coordinating the preparation of multiple recipes with interdependent timing constraints, featuring adaptive scheduling, intelligent task criticality assessment, and dynamic replanning capabilities. The invention addresses the previously unsolved problem of real-time cooking coordination across multiple concurrent recipes with varying temporal flexibility and quality-critical timing windows.

II. TECHNICAL PROBLEM ADDRESSED

Prior Art Limitations

Existing cooking timer and recipe management systems suffer from critical deficiencies:

1. **Static Scheduling**: Conventional systems calculate task timelines once at initialization, failing to accommodate real-world deviations, interruptions, or user-initiated delays.
2. **Undifferentiated Urgency**: Prior systems treat all timing alerts uniformly, providing no mechanism to distinguish between quality-critical operations (e.g., sautéing fish) and flexible preparatory tasks (e.g., bringing water to temperature).
3. **Single-Recipe Focus**: Existing solutions lack the architectural capability to coordinate timing across multiple interdependent recipes with shared resource constraints and synchronized serving requirements.
4. **Binary Temporal States**: Conventional systems recognize only "on-time" and "overdue" states, with no gradation of consequences or predictive impact analysis.
5. **Absence of Consequence Modeling**: No existing system models or communicates the specific culinary or scheduling consequences of temporal deviations.

III. NOVEL SOLUTION ARCHITECTURE

III. NOVEL SOLUTION ARCHITECTURE

A. Core Inventive Concepts

1. Temporal Window Classification System

The invention introduces a hierarchical classification of task timing windows with associated behavioral characteristics:

- **Locked Windows**: Fixed-duration operations with quality-critical timing (e.g., "sauté fish 3-4 minutes"). System enforces strict temporal boundaries.
- **Flexible Windows**: Operations tolerant of temporal variance (e.g., "simmer 20-30 minutes"). System permits dynamic adjustment within bounds.
- **Fixed Points**: Non-negotiable synchronization moments (e.g., "serve at 7:00 PM"). System treats as immutable constraints for backward scheduling.
- **Soft Targets**: Preference-based timing goals with no quality impact. System may sacrifice to accommodate higher-priority constraints.

2. Multi-Dimensional Task Criticality Framework

Novel classification system distinguishing:

- **Quality-Critical Tasks**: Operations where temporal deviation causes irreversible degradation of food quality. Characterized by:
  - Narrow tolerance windows (±1-3 minutes)
  - Chemical/physical transformation processes
  - Point-of-no-return thresholds
  - High-urgency user notification requirements
- **Schedule-Critical Tasks**: Operations forming critical path dependencies. Delay propagates to downstream tasks and final serving time.
- **Flexible Tasks**: Operations with inherent temporal buffers or minimal consequence profiles.
- **Preference Tasks**: Optimal timing suggestions without quality or schedule impact.

3. Consequence Prediction Engine

System maintains structured metadata for each task:

```
consequenceOfDelay: {
```

```
    short: "Concise impact statement"
    detailed: "Comprehensive consequence description"
    maxTolerableDelay: [numeric threshold in minutes]
    cascadeImpact: {
      affectedTasks: [array of downstream task identifiers]
      estimatedDelayPropagation: [minutes added to completion]
    }
  }
```

This enables intelligent communication of specific consequences rather than generic warnings.

4. Dynamic Replanning Algorithm

Upon detection of temporal deviation (user delay, extended duration, etc.), system executes:

1. **Impact Analysis**: Calculate ripple effects through task dependency graph
2. **Flexibility Assessment**: Identify which downstream windows can absorb delays
3. **Critical Path Recalculation**: Determine new optimal scheduling
4. **User Communication**: Present consequences and require acknowledgment for quality-critical overrides

Key innovation: System distinguishes between delays that can be absorbed through flexible windows versus those requiring serve-time adjustment.

5. "Now Line" Temporal Coordination Interface

Novel UI paradigm presenting:

- Horizontal timeline with current moment marker ("Now Line")
- Tasks approaching the Now Line with urgency-graduated visual indicators
- Expired tasks held at Now Line pending user dismissal
- Color-coded criticality visualization (red/orange/yellow/blue)
- Consequence information displayed on-demand

6. Multi-Recipe Coordination Graph

System constructs unified dependency graph spanning multiple recipes:

- Shared resource identification (oven, stovetop burners, cook attention)
- Cross-recipe timing constraints ("both dishes ready simultaneously")
- Intelligent task interleaving to optimize resource utilization

- Bottleneck detection and mitigation strategies

IV. TECHNICAL IMPLEMENTATION DETAILS

A. Data Structures

Task Object Schema

```typescript
interface Task {
  id: string;
  recipeId: string;
  description: string;
  durationType: 'locked' | 'flexible' | 'instantaneous';
  duration: number | [number, number];  // single value or range
  startTime: Date | null;
  endTime: Date | null;
  windowType: 'locked' | 'flexible' | 'fixed' | 'soft';
  dependencies: string[];  // task IDs that must complete first
  criticalityLevel: 'quality_critical' | 'schedule_critical'
                  | 'flexible' | 'preference';
  consequenceOfDelay: ConsequenceMetadata;
  status: 'pending' | 'active' | 'completed' | 'overdue';
  resources?: string[];  // 'oven', 'stovetop_1', 'attention', etc.
}
```

Timeline State Management

System maintains:

- Current time pointer
- Task queue ordered by scheduled start time
- Active task registry with real-time duration tracking
- Completed task history
- Override/delay event log with timestamps and user actions

B. Algorithmic Processes

Backward Scheduling from Fixed Points

For recipes with absolute serving deadlines:

1. Start with fixed end point (serve time)
2. Work backward through dependency chain
3. Place locked windows at calculated positions
4. Distribute flexible windows to fill gaps
5. Calculate initial prep start time
6. Validate resource availability throughout timeline

Dynamic Rescheduling on Deviation

When user delays task or extends duration:

1. **Freeze State**: Capture current timeline configuration
2. **Propagate Forward**: Calculate impact on dependent tasks
3. **Identify Absorption**: Find flexible windows that can compress
4. **Critical Path Analysis**: Determine if serve time must shift
5. **Generate Options**: Present user with:
    - Accept new serve time
    - Compress flexible windows to maintain schedule
    - Acknowledge quality risk if quality-critical task overdue
6. **Update Timeline**: Implement selected resolution
7. **Notify Dependencies**: Alert user to changed task starts

Resource Conflict Resolution

When multiple recipes compete for resources:

1. **Priority Assessment**: Weight by criticality and time urgency
2. **Interleaving Optimization**: Rearrange non-critical tasks
3. **Serial Scheduling**: Queue tasks for shared resources
4. **User Alert**: Notify of resource constraints and suggest timing adjustments

---

V. NOVEL FEATURES AND ADVANTAGES

Comparative Analysis vs. Prior Art

Feature	Prior Art	This Invention
Timeline Adaptation	Static, set-once	Dynamic replanning on deviation
Task Urgency	Uniform alerts	4-tier criticality with consequence data
Multi-Recipe Support	Single recipe focus	Unified coordination graph
Consequence Awareness	Generic warnings	Specific impact predictions
User Override Handling	Binary dismiss	Context-aware acknowledgment requirements
Resource Management	Implicit/manual	Explicit conflict detection and resolution
Temporal Flexibility	Fixed durations only	Range-based flexible windows

Key Innovations

1. **Adaptive Intelligence**: System learns from deviations and adjusts future scheduling accordingly
2. **Consequence-Driven Interaction**: User receives actionable information about impacts rather than generic alerts
3. **Graduated Response System**: Alert urgency and UI behavior scaled to actual risk level
4. **Holistic Multi-Recipe Optimization**: Cross-recipe coordination with shared constraint satisfaction
5. **Quality-Preservation Priority**: System architecture prioritizes culinary outcomes over arbitrary schedule adherence

---

VI. USE CASES AND APPLICATIONS

Primary Use Case: Multi-Course Dinner Party

User preparing:

- Appetizer (bruschetta - flexible timing)
- Main course (pan-seared fish with risotto - quality-critical coordination)
- Dessert (soufflé - locked timing window)

System coordinates all three recipes to achieve simultaneous readiness at 7:00 PM serving time, dynamically adjusting for user interruptions and optimizing resource utilization.

Secondary Applications

1. **Professional Kitchen Management**: Coordinating multiple orders with varying fire times

2. **Meal Prep Optimization**: Batch cooking multiple recipes with shared ingredient prep
3. **Cooking Education**: Teaching proper timing relationships and consequence awareness
4. **Accessibility Support**: Assisting users with executive function challenges through intelligent task management

---

VII. TECHNICAL IMPLEMENTATION CONSIDERATIONS

System Requirements

- Real-time clock with millisecond precision
- Task scheduling engine with graph dependency resolution
- UI framework supporting dynamic timeline visualization
- State management system for timeline modifications
- Notification system with graduated urgency levels

Scalability Factors

- System accommodates N recipes with M tasks each
- Dependency graph complexity: $O(NM)$ *vertices with potential* $O((NM)^2)$ edges
- Rescheduling computation: $O(NM\log(N*M))$ for critical path recalculation
- UI update frequency: 1Hz for countdown displays, event-driven for state changes

---

VIII. POTENTIAL PATENT CLAIMS

Independent Claims (Method)

Claim 1: A method for coordinating temporal execution of multiple interdependent cooking tasks across concurrent recipes, comprising:

- Parsing recipe instructions to extract tasks with temporal characteristics
- Classifying each task according to temporal flexibility and criticality level
- Constructing a dependency graph spanning multiple recipes
- Computing initial timeline through backward scheduling from fixed serving time
- Detecting deviations from scheduled timeline during execution

- Dynamically replanning task schedule based on deviation impact analysis
- Presenting consequence-specific information to user based on criticality classification

Claim 2: The method of Claim 1, wherein criticality classification comprises at least:

- Quality-critical designation for tasks with irreversible quality degradation consequences
- Schedule-critical designation for tasks affecting downstream dependency chain
- Flexible designation for tasks tolerant of temporal variance
- And wherein user notification urgency is scaled according to criticality level

Claim 3: The method of Claim 1, wherein dynamic replanning comprises:

- Calculating ripple effects through dependency graph
- Identifying flexible temporal windows capable of absorbing delays
- Determining whether deviation requires serving time adjustment
- Generating alternative scheduling options
- Requiring user acknowledgment for quality-critical overrides

Independent Claims (System)

Claim 4: A system for adaptive cooking task coordination, comprising:

- A recipe parser configured to extract temporal task specifications
- A task classifier configured to assign criticality levels based on consequence analysis
- A scheduling engine configured to compute optimal task timelines
- A deviation detector configured to identify temporal variances during execution
- A replanning engine configured to generate updated schedules responsive to deviations
- A notification interface configured to present graduated alerts based on task criticality

Dependent Claims

Claim 5: The method of Claim 1, wherein temporal characteristics include at least: locked duration windows, flexible duration ranges, fixed time points, and soft target times.

Claim 6: The method of Claim 1, further comprising maintaining consequence metadata for each task, including: short-form impact description, detailed consequence explanation, maximum tolerable delay threshold, and cascade impact assessment.

Claim 7: The system of Claim 4, wherein the notification interface comprises a timeline visualization with a current-time marker and task indicators positioned according to scheduled execution time and coded according to criticality level.

Claim 8: The method of Claim 1, further comprising resource conflict detection across multiple recipes and intelligent task interleaving to optimize shared resource utilization.

IX. PRIOR ART SEARCH NOTES

Known Systems (for differentiation)

1. **Kitchen Timer Applications** (e.g., multiple-timer apps)
   - Limitation: No task dependency awareness, no adaptive scheduling

2. **Recipe Management Software** (e.g., Paprika, Mealime)
   - Limitation: Static instruction display, no temporal coordination

3. **Smart Kitchen Assistants** (e.g., Alexa cooking skills)
   - Limitation: Single-recipe focus, no consequence modeling

4. **Project Management Software** (e.g., Gantt chart applications)
   - Limitation: Not optimized for cooking-specific constraints, no quality-criticality concept

Differentiating Factors

This invention uniquely combines:

- Real-time adaptive scheduling
- Culinary consequence modeling
- Multi-recipe coordination
- Criticality-aware task management
- Quality-preservation prioritization

No prior art identified combining these elements in cooking domain.

X. COMMERCIAL APPLICATIONS

Target Markets

1. **Consumer Cooking Applications**: Mobile/web apps for home cooks

1. **Consumer Cooking Applications**: Mobile/web apps for home cooks
2. **Smart Kitchen Appliances**: Integration with connected ovens, cooktops
3. **Professional Kitchen Systems**: Restaurant order coordination
4. **Cooking Education Platforms**: Teaching timing and coordination skills
5. **Meal Kit Services**: Enhanced instruction systems for subscription boxes

Revenue Models

- Consumer subscription (freemium with advanced features)
- License to appliance manufacturers
- B2B licensing for professional kitchens
- White-label platform for meal kit companies

XI. FUTURE ENHANCEMENTS

Potential Extensions

1. **Machine Learning Integration**: Learn user timing patterns and adjust tolerances
2. **Computer Vision Integration**: Detect actual food states (browning, bubbling) to override timers
3. **Voice Interface**: Hands-free interaction during cooking
4. **Collaborative Cooking**: Multi-user coordination with task assignment
5. **Nutritional Optimization**: Adjust timing to optimize nutrient retention
6. **Energy Optimization**: Schedule tasks to minimize energy consumption

---

XII. INVENTOR NOTES

Development Timeline

- **Concept Origin**: October 16, 2025
- **Initial Architecture**: October 16, 2025
- **Criticality Framework**: October 16, 2025

Key Design Decisions

1. **User Agency Preserved**: System advises but never forces user compliance

1. **User Agency Preserved**: System advises but never forces user compliance

2. **Consequence Transparency**: Always communicate "why" not just "when"

3. **Quality First**: Prioritize culinary outcomes over schedule adherence

4. **Graceful Degradation**: System remains functional even with significant deviations

Open Questions for Implementation

1. Recipe parsing accuracy for temporal extraction

2. Criticality assignment methodology (manual vs. automated)

3. User tolerance learning algorithms

4. Cross-cultural timing convention variations

XIII. CONFIDENTIALITY NOTICE

This document contains proprietary and confidential information constituting trade secrets. Distribution is restricted to authorized parties only. Any unauthorized review, use, disclosure, or distribution is prohibited.

Document Control
Version: 1.0
Date: October 16, 2025
Classification: Confidential - Invention Disclosure

XIV. ATTACHMENTS

A. Terminology Glossary

- **Locked Window**: Fixed-duration task with quality-critical timing

- **Flexible Window**: Variable-duration task with tolerable timing range

- **Fixed Point**: Immutable time constraint (e.g., serving deadline)

- **Now Line**: Current time marker in timeline visualization

- **Critical Path**: Sequence of tasks determining minimum completion time

- **Cascade Impact**: Downstream effect of task delay on dependent operations

B. Example Task Dependency Graph

[Would include visual diagram of multi-recipe coordination]

C. UI Mockups

[Would include wireframes of Now Line interface and notification systems]

END OF DISCLOSURE

INVENTOR DECLARATION

The undersigned declares that they are the original inventor(s) of the subject matter disclosed herein and that this disclosure is submitted for the purpose of establishing intellectual property rights.

Inventor Name: [To be completed]
Date: October 16, 2025
Signature: _________________

ATTORNEY NOTES SECTION

[Reserved for patent attorney review and strategy notes]

Review Date: __________
Reviewing Attorney: __________
Patentability Assessment: __________
Recommended Filing Strategy: __________
Prior Art Search Status: __________

This document prepared with assistance of AI system for organizational and documentation purposes. All inventive concepts originated with human inventor.