

1. Overview

This production plan outlines the structure, workflow, deliverables, milestones, and responsibilities for the development of **Hotdog Hustler**, a micro-scope 2D cooking/time-management game intended as the first shipped project under **Kato.8 Studios**. The plan assumes the project will start **next week**, once the team is assembled.

Development timeline is approximately **2 months**, but **flexible as needed**.

2. Team Roles & Responsibilities

Founder / Producer (You)

- Project management (ClickUp)
 - Weekly standups
 - Scope control
 - Steam page creation
 - Documentation oversight
 - Deliverable review & approvals
 - Marketing (free promotion)
 - Build coordination with developer
-

Programmer / Developer

Responsible for all Unity development, including:

- Full gameplay implementation
 - All systems: cooking, toasting, toppings, customer orders, timers
 - UI wiring + UI logic
 - Unity scene setup
 - Animation integration (artist-provided)
 - Input system implementation
 - Build pipeline creation (PC build)
 - Bug fixing, optimization
 - Preparation of final release builds for Steam/itch.io
 - Weekly deliverables and milestone demos
 - Technical decisions (UI system, architecture, scene structure)
-

2D Artist

Responsible for **all visual content**, including:

- Hotdog cart
- Background/environment
- Hotdog sprite & cooking states
- Bun sprites & toasting states
- Topping sprites (full list for v1)
- UI elements (buttons, order tickets, icons)
- Customer sprites (as many as can be delivered by milestone)

- Smooth/simple animations (6–8 frames)
- Basic visual FX (steam, sizzle, etc.)
- Asset optimization for Unity
- Art delivery in organized folders
- Weekly art progress updates

Style (pixel art or clean animated art) is **decided by artist**, as long as it fits scope and readability.

Sound Designer

Responsible for:

- Music track(s)
 - Sound effects: sizzling, toasting, topping additions, order notifications, UI clicks
 - Loudness normalization
 - File QA
 - Delivery in Unity-friendly formats (WAV/OGG)
 - No Unity implementation required (programmer integrates)
 - Weekly sound progress updates
-

3. Tools & Workflow

Project Management

- **ClickUp** (free)

- Tasks, roadmap, documentation repo, weekly sprint boards

Version Control

- **GitHub** for code repository
- Weekly branch merge schedule recommended

Communication

- Discord server (team channels)
- Weekly standup call
- Async updates via ClickUp comments

Engine & Technical Stack

- Unity
 - New Unity Input System
 - UI system chosen by dev (uGUI or UI Toolkit)
 - Keyboard controls only (controller optional for future)
-

4. Production Structure

Production is broken into:

- **Pre-Production** (1 week)
- **Sprint 1** (Week 1) – Core Systems
- **Sprint 2** (Week 2) – Toppings, UI, Art Integration

- **Sprint 3** (Week 3) – Customers, Orders, Loop Completion
- **Sprint 4** (Week 4) – Content Completion
- **Sprint 5** (Week 5) – Polish + QA
- **Sprint 6** (Week 6) – Release Prep
- **Optional Buffer Weeks (2 weeks)** — if the team needs extra time

Every sprint = **weekly deliverable + milestone playtest**.

5. Production Timeline (Flexible 6–8 Weeks)

Below is the **baseline**, which can shift depending on team readiness.

PRE-PRODUCTION (~1 week)

Goals

- Team onboarding
- ClickUp setup
- GitHub repo creation
- Confirm UI system
- Final reference moodboard
- Finalize toppings list & asset needs
- Architecture outline by programmer

- Art tests (cart prototype, bun/hotdog tests)

Deliverables

- Project board live
 - Documentation repo initialized
 - Visual art test (1–2 sprites)
 - Prototype scene with placeholder UI
 - Final approvals on technical direction
-

SPRINT 1 — CORE GAMEPLAY SYSTEMS (Week 1)

Programmer Deliverables

- Project setup in Unity
- Input system (keyboard)
- Base interactions (toast button, cook button) with placeholder art
- Scene setup and manager systems
- Basic state machine for hotdog cooking
- Stub UI for testing

Artist Deliverables

- Final art style locked
- Hotdog sprite + cooking states

- Bun sprite + toasting states
- Basic cart mockup

Sound Deliverables

- SFX prototype for cooking + UI click

Milestone Playtest

- Can toast bun & cook hotdog using placeholder UI
 - Game is “interactive,” even if ugly
-

SPRINT 2 — TOPPINGS & BASIC UI (Week 2)

Programmer Deliverables

- Topping system architecture
- Topping interaction logic
- Basic order assembly logic
- UI buttons working for toppings
- Improve hotdog station feel

Artist Deliverables

- All topping sprites (full v1 list)
- Cart final art

- Basic UI icons (toppings, order ticket, buttons)

Sound Deliverables

- Topping SFX
- Toast/cook SFX pass 2

Milestone Playtest

- Full assemble-hotdog loop possible
 - UI visible and functional
-

SPRINT 3 — CUSTOMER SYSTEM + ORDER LOOP (Week 3)

Programmer Deliverables

- Customer spawning system
- Order generation logic
- Timer system
- Order fail/success logic
- Delivery interaction
- First pass on end-of-day summary

Artist Deliverables

- Customer sprites (as many variations as possible)

- Order tickets UI final
- Background art finished

Sound Deliverables

- Customer reaction SFX (success/fail)
- Order bell SFX

Milestone Playtest

- Core gameplay loop complete

- Player can:

1. Receive orders
2. Cook
3. Build hotdog
4. Serve customer
5. Get results

After Sprint 3 → **No more new features are added.**

SPRINT 4 — CONTENT COMPLETION (Week 4)

Programmer Deliverables

- Polish interaction timing
- Add small variations (visual/UI polish)

- Integrate all animations
- End-of-day screen polishing
- Basic data tracking (money earned, orders served)

Artist Deliverables

- Final customer variations
- Final UI elements
- Basic VFX (steam, sizzle)
- Animation polish

Sound Deliverables

- Final music loop(s)
- Final SFX polish

Milestone Playtest

- Content complete build
 - Focus shifts to polish/bug fixes
-

SPRINT 5 — POLISH + QA (Week 5)

Programmer

- Fix bugs
- Smooth timing

- Improve responsiveness
- Remove friction from controls
- Add small animations/feedback cues

Artist

- Polish passes
- Fix readability issues
- Tighten animations

Sound

- Final loudness balancing
- Optional alternate music version

Milestone Playtest

- Game feels smooth, readable, responsive
 - No major bugs
 - Ready for release prep
-

SPRINT 6 — RELEASE PREP (Week 6)

Programmer

- Final build prep
- Steam build setup (if possible; optional in this phase)

- itch.io build packaging
- Create build folder structure

Producer

- Build Steam page (unless help available)
- Capture screenshots + trailer clips
- Optimize store description
- Prepare marketing posts

Artist

- Provide trailer art, banners if needed

Sound

- Provide trailer audio clips (optional)

Milestone

- Release candidate build ready
 - Steam page complete
 - itch.io page complete
-

OPTIONAL: SPRINT 7–8 BUFFER (If Needed)

To accommodate delays, more customer art, animation polish, or feature tweaks.

6. QA & Testing Plan

Testing Occurs:

- End of every sprint (milestone playtest)
- Final full 2–3 rounds in Sprint 5

QA Focus Areas

- Responsiveness of cooking
- UI clarity
- Timing fairness
- Order validation correctness
- No crashes or soft locks
- Steam build stability (if included)

7. Build & Release Pipeline

itch.io Release

- Programmer builds final PC build
- Producer uploads to itch.io
- Simple screenshots + GIFs + description

Steam Release

- Producer handles Steam page setup
 - Programmer delivers builds in correct folder structure
 - Achievements and SteamSDK integration optional secondary plan
-

8. Risk Mitigation

Potential Risks

- Artist unable to complete number of customers → fallback to fewer variations
- Topping list too large → cut non-essential sprites
- Programmer time constraints → reduce animation complexities
- Sound delays → use placeholder SFX until final passes

Solutions

- Scope reduction rules enforced
 - Weekly check-ins to catch blockers early
 - Buffer weeks built into timeline
 - Cut features rather than extend deadlines
-

9. Deliverable Expectations (Weekly)

Every week, each contributor must deliver:

- Clear progress update

- At least one tangible deliverable (art asset, gameplay feature, SFX, etc.)
- Participation in milestone playtest

Producer reviews all deliverables weekly.

10. Final Output

By the end of production, Hotdog Hustler will have:

- Fully working cooking/time-management loop
- Complete v1 topping set
- Background art
- Customer variations
- Responsive UI
- Updated art/animations
- Music & full SFX set
- itch.io + Steam releases ready