



# ROADMAP - Micro Architect v0.4 → v0.5+

**Objetivo:** Un solo documento para ir abordando poco a poco.

**Base:** v0.4-alpha (energía numérica, Polish UI aplicado)

**Última actualización:** 2025-01-31

**Hecho reciente (fuera de bloques):** Hotkeys 1-9 (output\_scene), clic central en edificio/suelo (misma orientación, colocar y mantener en mano), grid guía (pulso 50–100% + desvanecimiento por zoom), starter pack (Constructor 1, God Siphon solo DEV), merger 3x1 footprint (validación/registro multi-celda), RECETAS unificado en HUD (HUD\_CATEGORIAS/HUD\_LABELS), deprecated eliminado (energy\_pulse). Prismas T1/T2 apagado, bolas al quitar sifón, tamaños bolas. **Sistema de selección múltiple por arrastre:** SelectionManager (hold threshold, rectángulo fantasma, zoom fit solo aleja, margen), modo selección activable con botón SELECCIÓN (mismo estilo que GUARDAR/MENÚ), botón ELIMINAR en esquina inferior derecha, márgenes HUD (panel superior e inferior izq.). **Barra de recursos superior centrada** (HUD script: `_centrar_panel_recursos`, full rect, `size_changed` si existe). **Análisis de código y correcciones null-safety:** `beam_emitter` (`get_node_or_null` + check `MeshInstance3D`), `god_siphon` (check `current_scene` antes de `has_signal`), `save_system` (check `raiz` / `current_scene` en guardar/reconstruir/generar\_partida\_test), `world_generator` (`_posicionar_camara`), `inventory_button` (check `current_scene`), hud (conectar `size_changed` solo si la señal existe).

**Pendiente (pulir más adelante):** Layout/visual del HUD y botones de selección/eliminar no han quedado perfectos; se mantiene el estado actual.



## Cómo usar este doc

- Trabaja por bloques en orden (o el que prefieras).
- Marca [x] al completar cada ítem.
- Si algo bloquea, déjalo y sigue con el siguiente.
- Actualiza "Estado actual" al final cuando termines una sección.

## BLOQUE 1: Bugs menores (estabilidad)

### 1.1 Haces visuales cortados en prismas

- ☒ Reproducir: colocar prisma recto/angular, ver haz entrante/saliente.
- ☒ Identificar causa (primer segmento empezaba en origen+0.6, longitud 0.5 → hueco 0.35).
- ☒ Ajustar: HAZ\_OFFSET\_ORIGEN 0.6 → 0.25 para que el haz arranque en el prisma.
- ☐ Probar con varias rotaciones (recomendado en juego).

**Archivos probables:** `scenes/world/beam_segment.tscn`, `scripts/buildings/prism_logic.gd`, `scripts/components/beam_emitter.gd`, escenas `prism_straight.tscn` / `prism_angle.tscn`.

### 1.2 Salidas de mergers (posición/visual)

- ☒ Reproducir: merger con 2+ entradas, ver salida de energía/quarks.
- ☒ Revisar posición: `from_pos` estaba en `center + 1.5*dir` (muy adelante).
- ☒ Corregir: `from_pos = center + 0.5*dir` (cara de salida del mesh 3x1x1).
- ☐ Probar con rotaciones del merger (recomendado en juego).

**Archivos probables:** `scripts/buildings/merger.gd`, escena `merger.tscn`, `scripts/managers/beam_manager.gd`.

---

### 1.3 Verificar que visuales NO afectan lógica

- ☒ Listar: `spawn_pulse_visual` en `siphon_logic`, `god_siphon`, `prism_logic`, `compressor`, `merger`; `dibujar_haz` en los mismos.
- ☒ Confirmar: lógica = `EnergyManager.register_flow + EnergyFlow` → `recibir_energia_numerica`; visuales detrás de `MOSTRAR_VISUAL_PULSO`.
- ☒ Documentar: `ARCHITECTURE.md` (sección "Verificación visuales opcionales").

**Referencia:** `docs/ARCHITECTURE.md` , `docs/ENERGY_SYSTEM.md` .

---

## ● BLOQUE 2: Pulido UX pendiente

### 2.1 Feedback visual al colocar edificios

- ☒ Colocar válido: pequeño "pop" de escala ( $1.2 \rightarrow 1.08 \rightarrow 1.0$  con `TRANS_BACK`).
- ☒ Colocar inválido: sacudida breve del fantasma (posición  $\pm 0.08 \rightarrow \pm 0.04 \rightarrow 0$ ).
- ☒ Implementado en `ConstructionManager` : `confirmar_colocacion` (tween) y `_feedback_colocacion_invalida` (shake).
- ☐ Opcional: sonido corto en colocación válida/inválida.

**Archivos probables:** `scripts/managers/construction_manager.gd` , escena del fantasma, posible nodo de feedback en `main_game_3d.tscn` .

---

### 2.2 Mejorar menú (transiciones, feedback)

- ☒ Menú principal: transición suave al cambiar Main/Opciones (fade con `modulate`).
- ☒ Botones: hover scale 1.05, pressed scale  $0.98 \rightarrow 1$ .
- ☒ Versión "v0.4-alpha" bajo el título en menú principal.

**Archivos probables:** `scenes/ui/main_menu.tscn` , `scripts/ui/main_menu.gd` , `temas/estilos`.

---

## ● BLOQUE 3: Técnico / cleanup

### 3.1 Unificar fuentes de escenas (RECETAS vs `menu_data`)

- ☒ Listar dónde se usa `GameConstants.RECETAS` y dónde `menu_data` en HUD.
- ☒ Elegir fuente única (recomendado: `RECETAS`).
- ☒ Eliminar o derivar `menu_data` desde `RECETAS` en `hud_manager.gd` .
- ☐ Probar que todas las categorías y edificios aparecen bien (recomendado en juego).

**Archivos probables:** `scripts/autoload/game_constants.gd` , `scripts/managers/hud_manager.gd` .

---

### 3.2 Limpiar deprecated

- ☒ Confirmar que nada referencia `scenes/deprecated/` ni `scripts/deprecated/` .
- ☒ Si hay referencias, migrar o eliminar (no había referencias en código).
- ☒ Borrar carpetas/archivos deprecated (`energy_pulse.tscn`, `energy_pulse.gd` eliminados).
- ☒ Actualizar `PROJECT_STATE.md` y este doc.

**Referencia:** `docs/PROJECT_STATE.md` .

---

## BLOQUE 4: v0.5 – Electrones

### 4.1 Recurso y constantes

- ☒ Añadir recurso `Electron` (o nombre elegido) en `GameConstants` .
- ☒ Definir color, icono, cadena de producción (quarks → electrón).
- ☒ Añadir a `GlobalInventory` y HUD (categoría adecuada).

**Archivos:** `scripts/autoload/game_constants.gd` , `scripts/autoload/global_inventory.gd` , `scripts/ui/hud.gd` .

---

### 4.2 Edificio "Electrón" / Fabricador Hadrón

- ☒ Fabricador Hadrón (nucleones): consume quarks ( $2U+1D \rightarrow$  Protón,  $1U+2D \rightarrow$  Neutrón), produce Proton/Neutron al inventario.
- ☒ Crear escena `hadron_factory.tscn` y script.
- ☒ Registrar en RECETAS ( $40U+40D$ ), restricciones `TILE_VACIO`, `placement_logic`.
- ☒ Integrar con `recibir_energia_numerica` (quarks), añadir producto a `GlobalInventory`.
- ☐ Edificio Electrón (quarks→Electron): pendiente si se desea extender la cadena.

**Archivos:** `scenes/buildings/hadron_factory.tscn` , `scripts/buildings/hadron_factory.gd` , `game_constants.gd` , `placement_logic.gd` .

---

### 4.3 Integrar en cadena

- ☒ Fusionador puede alimentar al Fabricador Hadrón (pulsos de quarks).
- ☒ Flujo: Quarks → Fabricador Hadrón → Proton/Neutron (inventario).
- ☒ Actualizar F1/F2 (ayuda y recetario) con Fabricador Hadrón, Proton, Neutron.

**Archivos:** `scripts/ui/help_panel.gd` , `scripts/ui/recipe_book.gd` , `tech_tree.gd` .

---

### 4.4 Visuales electrón

- ☐ Definir aspecto (ej. esfera pequeña, color distinto a quarks).
  - ☐ Añadir visual de flujo si hay pulsos (opcional, coherente con `PulseVisual`).
- 

## Estado actual del ROADMAP

Bloque	Estado	Notas
1. Bugs menores	<input checked="" type="checkbox"/> Completado	1.1, 1.2, 1.3
2. Pulido UX	<input checked="" type="checkbox"/> Completado	2.1 y 2.2 (colocar + menús)
3. Técnico	<input checked="" type="checkbox"/> Completado	3.1 RECETAS/menu_data, 3.2 deprecated eliminado
4. Nucleones (Hadrón)	<input checked="" type="checkbox"/> Completado	4.2 Fabricador Hadrón <input checked="" type="checkbox"/> ; 4.3 F1/F2 <input checked="" type="checkbox"/> ; 4.4 visuales opcionales

---

## Referencias rápidas

Doc	Para qué
docs/README.md	Índice de toda la documentación
PROJECT_STATE.md	Estado del juego, bugs, versión
FUTURE_PLAN.md	Visión largo plazo (protones, átomos)
POLISH_PLAN.md	Detalle de lo ya hecho en UI/UX
ARCHITECTURE.md	Reglas de arquitectura (simulación vs visual)
ENERGY_SYSTEM.md	Flujos numéricos, EnergyManager
NOTAS_DESARROLLO.md	Commit + push al decir "hasta mañana"
API_MANAGERS.md	API de GridManager, EnergyManager, BuildingManager

---

Cuando termines un bloque, actualiza la tabla "Estado actual" y el **Última actualización** arriba.

---

## Siguiendo en la lista

**Orden sugerido:** Bloque 1 (bugs) → Bloque 2 (UX) → Bloque 3 (técnico) → Bloque 4 (Electrones).

Siguiente	Tarea
1.1	Haces visuales cortados en prismas
1.2	Salidas de mergers (posición/visual)
1.3	Verificar que visuales NO afectan lógica
2.1	Feedback visual al colocar edificios
2.2	Mejorar menús (transiciones, versión)
3.1	Unificar RECETAS vs menu_data en HUD
3.2	Limpiar deprecated 
4.1	Recurso Electron (GameConstants, GlobalInventory, HUD) 
4.2	Edificio Electrón (escena, script, RECETAS, EnergyManager)
4.x	4.3 Integrar en cadena, 4.4 Visuales electrón