

WorkspaceOps — HLR ↔ 30-Day Plan

Mapping

Rule used:

- **Design-day HLRs** = understood + documented
 - **Build-day HLRs** = actually implemented in code
 - Some HLRs span multiple days (that's normal)
-

TENANT & WORKSPACE

HLR ID	Description	Day(s)
HLR00 01	User signup & tenant creation	Day 7–8
HLR00 02	Default workspace creation	Day 8
HLR00 03	Workspace as logical boundary	Day 2 (design) + enforced Day 7 onward

 Note: HLR0003 is an **architectural invariant**, not a single feature.

USER & RBAC

HLR ID	Description	Day(s)
--------	-------------	--------

HLR00 04	Invite users	Day 9
HLR00 05	One role per user per workspace	Day 9
HLR00 06	Support OWNER / ADMIN / MEMBER / VIEWER	Day 9
HLR00 07	Restrict actions by role	Day 9– 10

👉 RBAC is **implemented once**, then **used everywhere after**.

ENTITY MANAGEMENT

HLR ID	Description	Day(s)
HLR00 08	Create entities	Day 10
HLR00 09	Assign entity role	Day 10
HLR00 10	Entities as subjects for docs & work	Day 10 (design) + Day 11 onward

DOCUMENT CONFIGURATION

HLR ID	Description	Day
		(s)

HLR00 11	Define document types	Day 1 1
-------------	-----------------------	----------------------

HLR00 12	Optional metadata support	Day 1 1
-------------	---------------------------	----------------------

HLR00 13	Optional expiry support	Day 1 1
-------------	-------------------------	----------------------

👉 This is a **core abstraction day**.

📁 DOCUMENT MANAGEMENT

HLR ID	Description	Day
		(s)

HLR00 14	Upload documents	Day 1 2
-------------	------------------	----------------------

HLR00 15	Associate document type	Day 1 2
-------------	-------------------------	----------------------

HLR00 16	Enforce required metadata	Day 1 2
-------------	---------------------------	----------------------

HLR00 17	Optional entity linking	Day 1 2
-------------	-------------------------	----------------------

HLR00 18	External file storage + metadata	Day 1 2
-------------	----------------------------------	----------------------

👉 Day 12 is **file + metadata heavy**. Expect friction.

⌚ DOCUMENT EXPIRY

HLR ID	Description	Day(s)
--------	-------------	--------

HLR00 19	Calculate expiry	Day 12
-------------	------------------	---------------

HLR00 20	Flag valid / expiring / expired	Day 12– 13
-------------	---------------------------------	-------------------------

👉 No cron jobs yet — computed on read.

🧠 WORK ITEM (GENERALIZED COMPLIANCE)

HLR ID	Description	Day(s)
--------	-------------	--------

HLR00 21	Define work item types	Day 13
HLR00 22	Create work items	Day 14
HLR00 23	Associate entity, type, owner	Day 14
HLR00 24	Work item lifecycle	Day 14– 15
HLR00 25	Link documents to work items	Day 15

👉 This replaces your old **service + compliance** logic.

AUDIT LOGGING

HLR ID	Description	Day
		(s)
HLR00 26	Record audit logs	Day 1 6
HLR00 27	Actor, action, target, time	Day 1 6

👉 Implement **only once**, then reuse middleware/hooks.



OVERVIEW (NOT DASHBOARD)

HLR ID	Description	Day (s)
--------	-------------	--------------------

HLR0028	Workspace overview	Day 1 9
---------	--------------------	---------------

HLR0029	Aggregate counts only	Day 1 9
---------	-----------------------	---------------

👉 Simple `count()` queries. No charts.

🚫 EXPLICIT NON-REQUIREMENTS

HLR ID	Description	Day(s)
HLR0030–HLR0034	Not in MVP	Never (intentionally)

👉 These are **scope guards**, not tasks.

High-level Day ↔ Focus View (sanity check)

- Days 1–3 → Scope, architecture, ER

- Days 4–6 → Data model + HLR + FRD
- Days 7–10 → Auth, RBAC, Entities
- Days 11–12 → Documents + metadata + expiry
- Days 13–15 → Work items
- Day 16 → Audit log
- Days 17–18 → Fix + refactor
- Day 19 → Overview counts
- Days 20–25 → Polish + README
- Days 26–30 → Freeze + interview prep

This **perfectly aligns** with:

- your earlier roadmap
 - the “2nd full project” intent
 - what interviewers expect
-

Brutally honest check (important)

- ✗ You will **not** finish all HLRs if you skip days
- ✗ You will **not** finish if you mix days
- ✓ You **will** finish if you treat each day as “HLR delivery”

This mapping gives you a **tracking spine**.

-