

Magasinet KBH Acceptance Audit

Repository under test: /Users/qodeboy/Workspace/UpWare/playground/magasinet-kbh

Generated: 2026-04-17 07:54 UTC

Executive Verdict

Recommendation band: Reject/refund candidate. The deliverable does build locally, but it still fails too many high-impact acceptance checks: core mandatory capabilities are only partially implemented, several public-facing surfaces are fake or stubbed, and the security posture is weak enough to block sign-off on its own.

The strongest rejection signals are not cosmetic. They include public exposure of paywall-bypass data, insecure organization and upload APIs, a stubbed subscription upgrade path, a fake newsletter page, and a fabricated archive. That is not normal punch-list territory.

Verdict	Reject/refund candidate
Required capabilities scored	61
Required full	27
Required partial	25
Required skipped	9
High-impact required gaps (impact >= 8)	13
Critical audit findings (9-10)	12
Severe mocked/hardcoded findings (>= 8)	6

Methodology

1. Built a canonical requirement list from RFQ.csv, KBH 2025 specs.pdf, the legacy Drupal custom modules, and the public legacy site structure. 2. Traced the new Next.js codebase statically across app routes, server modules, components, and payment/auth paths. 3. Ran local evidence checks: pnpm build, pnpm test, DB migrate/seed, and public route smoke probes on a local server. 4. Separated required, optional, withdrawn, and out-of-scope rows so the verdict is based on mandatory feature delivery rather than wishful accounting.

Confidence limits: public runtime was available locally, but contractor-hosted environments and admin/payment credentials were not. Authenticated, payment, and organization-management flows were therefore judged mostly from code paths and local unauthenticated smoke checks. Confidence scores are lowered where runtime proof was not available.

Evidence commands used in this pass included: `pnpm build`, `pnpm test`, local HTTP probes against `http://127.0.0.1:3100`, and selective live-site checks against `https://www.magasinetkbh.dk/` for legacy-baseline confirmation.

Feature Parity Matrix

Required rows drive the verdict. Optional, withdrawn, and out-of-scope rows are still included for transparency and repeated again in the appendix.

ID	Area	Capability	Requirement / Refs	Legacy vs New Evidence	Runtime Check	Verdict	Scope	Conf.	Impact	Notes
F001	Backend	API authorization and security policies	Scope: RFQ+Repo Refs: RFQ.csv:2; src/app/api/organization/[id]/route.ts:8-136; src/app/api/upload/delete/route.ts:4-32; src/server/modules/paywall-bypass/paywall-bypass-token.router.ts:28-48	Legacy: Legacy custom modules enforce admin/editor access patterns across account, paywall, and content flows in mkbh_account_page, mkbh_paywall, mkbh_article_page. New: Auth exists, but multiple sensitive paths are under-protected: arbitrary organization read/update by any logged-in user, unauthenticated upload delete, and public paywall-bypass token/IP list/get/count procedures.	Local code review only. No contractor deployment or admin credentials were available for live exploitation testing.	Partial	required	0.95	10	Security posture is materially below acceptance grade even though auth primitives exist.
F002	Backend / Users	User accounts: login, registration, profile, and self-management	Scope: RFQ+Repo Refs: RFQ.csv:3; RFQ.csv:51; src/server/auth/index.ts:33-135; src/app/(main)/(auth)/_components/sign-up-form.tsx; src/app/(main)/profile/page.tsx	Legacy: Legacy account handling lives in __legacy/sites/all/modules/custom/mkbh_account_page and public profile/account surfaces on the Drupal site. New: Better Auth, sign-in/sign-up flows, profile page, subscription list, and organization selection are implemented in the new app.	Public local route probes confirmed auth entry points render; protected profile routes require session.	Full	required	0.83	8	Quality issues remain elsewhere, but core user-account capability is present.
F003	Backend / CMS	Article CMS core: create, edit, publish, and schedule content	Scope: RFQ+Repo Refs: RFQ.csv:4-6; src/components/dashboard/post/post-editor.tsx; src/components/dashboard/post/post-form.tsx; src/server/commands/post-publish-scheduled.command.ts	Legacy: Legacy Drupal editor, content types, and scheduling live in mkbh_article_page and theme overrides. New: Dashboard editor, Tiptap content model, post CRUD, and scheduled publish job exist in the Next.js codebase.	Local build passed and post routes rendered seeded content. No admin session was available to exercise create/edit in browser.	Full	required	0.82	8	Core authoring exists; several editorial deltas below are still incomplete.
F004	Backend / CMS	Support for public article families: KBH+, city life, city spaces, city housing, opinion, visions, projects, reviews, and photo	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:5; src/server/modules/post/types/post-type.enum.ts:1-49; src/app/(main)/kbhplus/page.tsx; src/app/(main)/anmeldelser/page.tsx; src/app/(main)/projekter/page.tsx	Legacy: Legacy public IA exposes /kbhplus, /byens-rum, /byens-liv, /opinion, /visioner, /projekter, /anmeldelser, /foto on the live Drupal site. New: Equivalent public routes and post-type enum values exist in the new app.	Local probes returned HTTP 200 for /kbhplus, /projekter, /anmeldelser, /foto, /opinion, /visioner.	Full	required	0.90	8	The families exist. Requested taxonomy/menu cleanup is a separate gap.

ID	Area	Capability	Requirement / Refs	Legacy vs New Evidence	Runtime Check	Verdict	Scope	Conf.	Impact	Notes
F005	Backend / CMS	Static/basic pages CMS	Scope: RFQ+Specs+Repo Refs: RFQ.csv:6; KBH 2025 specs.pdf:112-139; src/server/modules/page/page.router.ts; src/app/(main)/about/page.tsx; src/app/(main)/kontakt/page.tsx	Legacy: Legacy basic pages, member FAQ, contact, and about pages are Drupal node/page surfaces. New: New app has page module, page CRUD, and multiple public static pages backed by page content fetches.	Local probes returned HTTP 200 for public page routes such as /nyhedsbrev and /payments/business. /annoncering currently fails at runtime and is tracked separately.	Full	required	0.79	6	The page framework exists, but specific page surfaces can still be broken.
F006	Comments	Frontend comments widget with threads and replies	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:7; RFQ.csv:37; src/component s/shared/comments/comment-thread.tsx; src/server/modules/comment/comment.router.ts; _legacy/sites/all/modules/custom/mkbh_comment/mkbh_comment.module	Legacy: Legacy site supports threaded comments via mkbh_comment and ajax_comments. New: New app includes comment list/create/update/delete, threaded rendering, and reply handling.	Comment-capable article templates are wired; public local article probes rendered content pages that include comment-section components in code.	Full	required	0.80	7	No admin moderation runtime proof was available in this pass.
F007	Comments	Comment notifications and unsubscribe flow	Scope: Legacy+Repo Refs: _legacy/sites/all/modules/custom/mkbh_comment/mkbh_comment.module; src/server/modules/comment; src/server/mailer	Legacy: Legacy comment module includes mail-driven comment interactions and unsubscribe-style behavior. New: New code shows likes and CRUD, but no equivalent comment-notification or unsubscribe flow was located.	Code review only. No notification routes or mail templates specific to comment subscriptions were found.	Skipped	required	0.72	5	Community retention feature from legacy appears missing.
F008	Analytics	Article metrics: views, shares, reactions, and GA-enriched counts	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:8; src/server/modules/analytic/analytic.service.ts:38-168; src/components/shared/google-analytics/google-analytics-stat.tsx:5-20	Legacy: Legacy custom analytics module mkbh_node_stats tracked public engagement metrics. New: The new app stores view/share/reaction stats and can sum GA views, but dashboard reporting is limited and live proof is incomplete.	Article detail code triggers 'analytic.view'; local runtime exercised public article pages successfully.	Partial	required	0.77	6	Metrics plumbing exists, but acceptance-grade reporting and verification are incomplete.
F009	Media	Media library, upload workflows, and media CRUD	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:9; src/server/modules/media/media.router.ts; src/components/dashboard/media/media-list.tsx; src/components/dashboard/media/media-crea te-many-form.tsx	Legacy: Legacy site has custom media flows, photo bulk upload, audio/video players, and inline-image support. New: New app ships media CRUD, multi-upload, preview, caption/alt/author fields, and reusable picker flows.	Build and seeded runtime passed. No authenticated browser upload was performed in this pass.	Full	required	0.81	7	A separate security finding exists for unauthenticated asset deletion.
F010	Recommender	Related-content or recommender system	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:10; src/components/shared/post-sections/related-article.tsx; src/components/shared/post-sections/related-vision.tsx	Legacy: Legacy site has related-content blocks and article adjacency in mkbh_blocks and mkbh_article_page. New: New app renders related content UI blocks, but no AI or richer recommendation logic was evidenced.	Public article templates reference related-content components; no dedicated recommender service was found.	Partial	required	0.73	4	Baseline related-content UX exists. RFQ hint toward stronger recommender behavior is unmet.
F011	Map / Geodata	Geocoding, geodatabase, and map-backed content	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:11; src/app/(main)/kort/page.tsx; src/server/modules/post/post.router.ts; src/components/tip-tap/uptap-widgets/map	Legacy: Legacy site has dedicated map and projects modules backed by map markers and geodata. New: The new app exposes a public map page, post geolocation queries, and map widgets, but explicit geocoding workflows were not fully proven.	Local probe of /kort returned HTTP 200. Public rendering works, but the page heading is sparse and deeper map interactions were not manually exercised.	Partial	required	0.73	7	Public map surface exists. Back-office geocoding fidelity needs authenticated verification.
F012	Archive	Magazine archive surface and archive management	Scope: RFQ+Legacy+Live+Repo Refs: RFQ.csv:12; https://www.magasinetkbh.dk/arkiv; _legacy/sites/all/modules/custom/mkbh_archive_page/mkbh_archive_page.module; src/app/(main)/arkiv/page.tsx:19-179	Legacy: Legacy live archive exposes the full historical magazine archive with real issue downloads. New: New app replaces the archive with a hardcoded five-row constant that reuses the same image and PDF path while claiming all 57 issues are available.	Local GET /arkiv -> 200 with heading 'bladarkiv'; page text claims all 57 issues, but code only contains five static entries.	Partial	required	0.96	8	This is one of the clearest acceptance failures and also a mock/hardcode finding.
F013	Newsletter	Newsletter subscription and management	Scope: RFQ+Legacy+Live+Repo Refs: RFQ.csv:13; https://www.magasinetkbh.dk/nyhedsbrev; src/server/modules/newsletter/services/newsletter.service.ts:25-46; src/app/(main)/nyhedsbrev/_components/newsletter-form.tsx:35-39	Legacy: Legacy site exposes newsletter capture and management surface on /nyhedsbrev. New: Generic newsletter subscribe mutation exists, popup/project widgets call it, but the dedicated newsletter page submit handler is empty and no management UI was found.	Local GET /nyhedsbrev -> 200 with heading 'Modtag vores nyhedsbrev'. The page renders, but its primary form does not call the backend.	Partial	required	0.96	8	Public-facing newsletter entry point is materially fake.
F014	Newsletter	RSS feed surface	Scope: Legacy+Live+Repo Refs: https://www.magasinetkbh.dk/nyhedsbrev; _legacy/sites/all/modules/custom/mkbh_rss_feed; src/app/(main)/nyhedsbrev/page.tsx:63-83	Legacy: Legacy system has a custom RSS feed module and public feed discovery. New: New site only links out to an external static XML feed hosted elsewhere; no new in-app RSS generation was found.	Local GET /nyhedsbrev -> 200; RSS section renders a direct external URL.	Partial	required	0.86	4	Users still get a feed link, but the capability was not migrated into the application itself.
F015	Payments	Payment processing via Stripe and MobilePay	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:14-18; src/server/modules/payment/providers/stripe.provider.ts; src/server/modules/payment/providers/mobilepay.provider.ts	Legacy: Legacy site used Stripe and MobilePay across member and donation flows. New: Both providers exist in the new backend, but adjacent flows remain incomplete or uneven, especially upgrade/self-service and business seat changes on MobilePay.	No end-to-end live payment was run. Public payment pages render locally.	Partial	required	0.75	9	Core payment providers exist, but acceptance cannot treat the surrounding flow as complete.
F016	Payments	Recurring billing management and self-service subscription lifecycle	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:16; src/components/payment/blocks/subscription-card.tsx:65-211; src/server/modules/payment/services/subscription.service.ts:170-175	Legacy: Legacy account pages supported paid-member lifecycle management. New: Cancel and restore exist, but upgrade is stubbed and UI still calls it. Portal behavior also degrades to errors or fallbacks.	Code review only for authenticated lifecycle flows. Public runtime confirmed test/demo surfaces still expose upgrade attempts.	Partial	required	0.95	9	The most obvious self-service upgrade path is not implemented.
F017	Payments	Business users, seats, invitations, and sub-accounts	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:17; src/app/api/organization/dashboard/route.ts:35-227; src/app/api/organization/invitations/route.ts:22-168; src/server/modules/payment/providers/mobilepay.provider.ts:74-76	Legacy: Legacy account/business handling supported organization-style memberships and seat concepts. New: The new app does implement organization dashboards, invitations, membership lists, and seat adjustment, but the feature is uneven: seat adjustment is unsupported for MobilePay and a separate by-id organization API is insecure.	Feature judged primarily from code because business flows require authenticated state and provider access.	Partial	required	0.84	9	This is not absent, but it is not cleanly complete enough for an unconditional acceptance.
F018	Payments	Webhook integration for Stripe and MobilePay	Scope: RFQ+Repo Refs: RFQ.csv:18; src/app/api/mobilepay/webhooks/route.ts; src/server/modules/payment/services/mobilepay-webhook-handler.service.ts; src/server/modules/payment/services/stripe-webhook.service.ts	Legacy: Legacy payment flows relied on webhook-style asynchronous state updates. New: Webhook routes and handlers exist for both providers in the new application.	No external provider callbacks were replayed in this audit pass.	Full	required	0.68	6	Implemented in code, but runtime confidence is lower without provider-side execution.

ID	Area	Capability	Requirement / Refs	Legacy vs New Evidence	Runtime Check	Verdict	Scope	Conf.	Impact	Notes
F019	Data	Pre-launch data migration	Scope: RFQ+Legacy Refs: RFQ.csv:19; __legacy/sites/all/modules/custom/mkbh_migration	Legacy: Legacy codebase contains migration-related custom tooling. New: This audit did not evaluate migrated content/data parity. The new repo includes seeders and schema, but not a validated migration result.	Explicitly excluded from this pass by audit scope.	Skipped	out_of_scope	0.40	5	Not scored in acceptance verdict for this report.
F020	Non-functional	SLA / TTLB / deployability expectations	Scope: RFQ+Repo Refs: RFQ.csv:20; pnpm build output; pnpm test output	Legacy: Legacy system at least operates publicly at scale today. New: New app builds locally, but default test command is broken, public /annoncering returns 500 locally, and there is no evidence of SLA enforcement or performance budgets.	Local `pnpm build` compiled successfully; local `pnpm test` fails because legacy tests are included; local `/annoncering` returned 500.	Partial	required	0.94	8	Deployability is better than zero, but not acceptance-clean.
F021	Dashboard	User management dashboard	Scope: RFQ+Repo Refs: RFQ.csv:22-24; src/app/dashboard/auth/users/page.tsx; src/app/dashboard/admin/users/page.tsx:253-258	Legacy: Legacy back-office supported user administration via Drupal admin views. New: The new app has user management surfaces, but there are duplicate dashboard implementations and at least one broken create-user link under /dashboard/admin/users/create.	Code review only. No authenticated dashboard walkthrough was available.	Partial	required	0.88	7	Core capability exists, but dashboard integrity is inconsistent.
F022	Dashboard	Article management dashboard, writing editor, and custom widget placements	Scope: RFQ+Specs+Repo Refs: RFQ.csv:25-27; KBH 2025 specs.pdf:21-74; src/components/tip-tap/components/tiptap-toolbar.tsx; src/components/tip-tap/tiptap-node	Legacy: Legacy editor supported rich embedded widgets and flexible article layouts. New: New dashboard editor is rich and extensible, with inline widget nodes, media selection, poll widgets, map widgets, and metadata forms.	Build passed and seeded content renders widget-aware public pages. Admin-side authoring was assessed from code only.	Full	required	0.82	8	Several specific widget/taxonomy deltas still fail separately below.
F023	Dashboard	Reports for users, subscriptions, and business/content metrics	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:28-30; src/components/dashboard/dashboard/page.tsx:34-119	Legacy: Legacy platform had admin/reporting expectations around users, subscriptions, and business metrics. New: New dashboard shows simple counts, but no real subscription reporting, churn reporting, or business/content metrics suite matching the RFQ wording was found.	Dashboard metrics were assessed from code only.	Partial	required	0.80	6	Basic counters exist. Actual reporting depth does not.
F024	Dashboard	Automated emails	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:31-32; src/server/modules/auth/auth.service.ts:149-188; src/server/mailler/templates	Legacy: Legacy system used automated email behavior in account and community flows. New: New app sends auth and invitation emails, but no broader automated email/reporting capability matching the RFQ phrasing was located.	Code review only.	Partial	required	0.73	5	Email infrastructure exists; the promised feature breadth does not.
F025	Dashboard	CSS injector or admin-controlled dynamic CSS	Scope: RFQ+Specs+Legacy+Repo Refs: RFQ.csv:33; KBH 2025 specs.pdf:5-16; src/server/modules/site-config/enums/site-config-key.enum.ts:2; src/app/page.tsx:142-152	Legacy: Legacy site had dynamic CSS/front-page variant behavior. New: New site only exposes a `homeLayout` switch; no generic CSS injector or equivalent admin CSS control was found.	Code review only.	Skipped	required	0.86	4	The narrow homepage layout toggle does not satisfy a CSS injector feature.
F026	Frontend	Dynamic block templates and grid-based composable front page	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:35-41; src/app/page.tsx:33-175; src/app/(main)/_components	Legacy: Legacy home/section pages used custom block composition. New: New homepage is a composed grid of section blocks and most-read/vision/review/map blocks.	Local GET / -> 200 and rendered homepage sections with seeded content.	Full	required	0.90	7	Composition exists and is publicly reachable.
F027	Frontend	Manual front-page top-layout variant switch	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:5-16; src/app/page.tsx:142-152; src/server/modules/site-config/enums/site-config-key.enum.ts:2	Legacy: Legacy site reportedly toggled two homepage variants via dynamic CSS/time-based behavior. New: New homepage supports `default` and `compact` layouts via site config, but the spec asked for very easy main-menu access and that UX was not verified.	Code review plus local homepage render. No authenticated dashboard/menu walkthrough available.	Partial	required	0.76	5	Likely functionally present, but the operational UX requirement is unproven.
F028	Frontend / Map	Map page and map widget	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:42; src/app/(main)/kort/page.tsx; src/config/enums/tip-tap.ts:76-80	Legacy: Legacy site had a dedicated map page and map-related widgets/markers. New: New app ships a map page and map widget type in the editor.	Local GET /kort -> 200.	Full	required	0.84	7	Deeper interactive map QA was outside this pass.
F029	Frontend / Sections	City spaces, city life, and city housing list/detail pages	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:43; src/app/(main)/byens-rum/page.tsx; src/app/(main)/byens-liv/page.tsx; src/app/(main)/byens-bolig/page.tsx; src/app/(main)/inhold/[slug]/page.tsx	Legacy: Legacy public IA includes city-space and city-life section routes and details. New: New app implements list pages for those sections and shared detail rendering under the article route.	Local probes returned HTTP 200 for seeded /inhold/* content and city-family list routes render via the homepage and section pages.	Full	required	0.82	7	Routing defect is covered separately under root-slug migration.
F030	Frontend / Sections	Opinion list and detail pages	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:44; src/app/(main)/opinion/page.tsx; src/app/(main)/opinion/[slug]/page.tsx	Legacy: Legacy live site exposes opinion list/detail pages. New: New app implements opinion list and detail routes.	Local GET /opinion -> 200.	Full	required	0.87	6	
F031	Frontend / Sections	Visions list and detail pages	Scope: RFQ+Legacy+Live+Repo Refs: RFQ.csv:45; https://www.magasinetkbh.dk/vision/lav-cykelparkering-over-alle-hovedbanegarden-perroner; src/app/(main)/visioner/page.tsx; src/app/(main)/visioner/[slug]/page.tsx	Legacy: Legacy site exposes public vision pages with voting/comment context. New: New app implements visions list/detail routes and renders vote-related stats.	Local GET /visioner -> 200.	Full	required	0.83	6	
F032	Frontend / Sections	Projects list and detail pages	Scope: RFQ+Legacy+Live+Repo Refs: RFQ.csv:46; https://www.magasinetkbh.dk/projekter; https://www.magasinetkbh.dk/projekt/axel-towers; src/app/(main)/projekter/page.tsx; src/app/(main)/projekter/[slug]/page.tsx	Legacy: Legacy site exposes project index/detail pages and project-specific update calls to action. New: New app implements project list/detail routes and project metadata rendering.	Local GET /projekter -> 200.	Full	required	0.86	7	Project alerting is a separate missing sub-capability.
F033	Frontend / Sections	Reviews list and detail pages	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:47; src/app/(main)/anmeldelser/page.tsx; src/app/(main)/anmeldelser/[slug]/page.tsx	Legacy: Legacy site exposes reviews and scene/film review metadata. New: New app implements review list/detail routes and review metadata fields.	Local GET /anmeldelser -> 200.	Full	required	0.87	6	
F034	Frontend / Sections	Photo list and detail pages	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:48; src/app/(main)/foto/page.tsx; src/app/(main)/foto/[slug]/page.tsx	Legacy: Legacy site exposes public photo series/list surfaces. New: New app implements photo list/detail routes and photo media rendering.	Local GET /foto -> 200.	Full	required	0.86	6	

ID	Area	Capability	Requirement / Refs	Legacy vs New Evidence	Runtime Check	Verdict	Scope	Conf.	Impact	Notes
F035	Frontend / Commerce	Personal subscription purchase flow	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:49; src/app/(main)/payments/personal/page.tsx; src/app/(main)/payments/personal/checkout/page.tsx	Legacy: Legacy site offered personal member purchase and support flows. New: New app has personal pricing and checkout pages, but surrounding lifecycle is incomplete and the landing page still carries hardcoded marketing values.	Local GET /payments/personal -> 200. Checkout was not executed without authenticated payment testing.	Partial	required	0.80	9	Core flow exists, but acceptance cannot treat it as fully delivered.
F036	Frontend / Commerce	Business subscription purchase flow	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:49; src/app/(main)/payments/business/page.tsx; src/app/(main)/payments/business/checkout/page.tsx	Legacy: Legacy site supported organization/business memberships. New: New app has business pricing and checkout scaffolding plus organization dashboard flows, but some provider paths and later management actions are incomplete.	Local GET /payments/business -> 200.	Partial	required	0.82	9	Good amount of code exists, but not enough to call the overall business flow fully accepted.
F037	Frontend / Commercial	Advertiser promo surface	Scope: RFQ+Live+Repo Refs: RFQ.csv:50; src/app/(main)/annoncering/page.tsx; https://www.magasinetkbh.dk/annoncering	Legacy: Legacy site has a public advertiser information page. New: New app has an advertising page route, but it crashes locally with a server-side 500.	Local GET /annoncering -> 500; server log captured 'TypeError: a.map is not a function' from the built app.	Partial	required	0.95	7	Public commercial entry point is currently broken at runtime.
F038	Frontend / Commercial	Sponsor promo surface	Scope: RFQ+Live+Repo Refs: RFQ.csv:50; src/app/(main)/payments/sponsorship/page.tsx; https://www.magasinetkbh.dk/sponsor	Legacy: Legacy site exposes sponsor information and sponsor-related assets. New: New app ships a sponsorship page, sponsor plan CRUD, and sponsor listing, but conversion still relies on mailto/manual contact and legacy assets.	Local GET /payments/sponsorship -> 200.	Partial	required	0.90	6	Usable as a brochure page, not as a fully modernized flow.
F039	Frontend / Profile	User profile and subscription self-management	Scope: RFQ+Repo Refs: RFQ.csv:51; src/app/(main)/profile/page.tsx; src/components/profile/blocks/subscription-list.tsx	Legacy: Legacy account page handled profile and membership management. New: New app includes profile area, subscription cards, invoices, organization list, and comment history.	Protected routes were validated by code review. Public probes confirmed profile entry points exist in the route tree.	Full	required	0.77	7	Subscription upgrade remains incomplete, but the broader profile surface exists.
F040	Frontend / Taxonomy	Author, district, and tag index pages	Scope: RFQ+Legacy+Live+Repo Refs: RFQ.csv:52; https://www.magasinetkbh.dk/search/node/metro; src/app/(main)/bydel/[slug]/page.tsx; src/app/(main)/emne/[slug]/page.tsx; src/app/(main)/skribent/[slug]/page.tsx	Legacy: Legacy public site exposes taxonomy and author-style landing pages. New: District and tag routes clearly exist and render. Author route exists in code, but live proof was limited because seeded author data was sparse.	Local GET /bydel/indre-by -> 200; local GET /emne/metro -> 200; local probe of a synthetic author slug returned 404.	Partial	required	0.78	5	Route infrastructure is present; author-page confidence is lower without seeded author data.
F041	Paywall	SEO-friendly paywall and gated KBH+ behavior	Scope: RFQ+Legacy+Repo Refs: RFQ.csv:39; src/components/paywall/kbhplus-paywall.tsx; src/server/modules/auth/auth.service.ts:73-127	Legacy: Legacy site used paywall logic and subscription-aware access controls. New: New app has KBH+ gating and subscription checks, but paywall bypass exposure and guest-link weaknesses materially undermine the implementation.	Seeded KBH+ route rendered publicly; authenticated subscription gating was assessed from code only.	Partial	required	0.80	9	Feature exists, but the control surface is not safely implemented.
F042	Widgets	Inline timeline widget	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:21-39; src/components/tip-tap/types/types.ts:12-27; src/components/tip-tap/tiptap-node/timeline/timeline-node.tsx	Legacy: Legacy editor did not expose this exact new widget; it was a spec delta. New: The new editor stores title, description, timeline items, icon, media, and text, matching the requested shape.	Code review only for editor-side setup. Public renderer exists in src/components/tip-tap/tiptap-widgets/timeline/timeline-widget.tsx.	Full	required	0.88	5	
F043	Widgets	Inline chart widget with editorial data entry	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:40-48; src/config/enums/tip-tap.ts:55-117	Legacy: This was a net-new requirement. New: The new editor exposes poll, vote, suggestion, timeline, map, etc., but no generic chart widget type or chart-data-entry workflow was found.	Code review only.	Skipped	required	0.92	6	Existing chart components are tied to poll result rendering, not a generic inline chart authoring feature.
F044	Widgets	Inline poll widget with multiple choice, checkboxes, slider, and free text	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:51-61; src/server/modules/poll/types/poll-question-type.enum.ts:1-16; src/components/form/control/poll-question.collection.tsx:46-71	Legacy: This was a new requirement over legacy. New: The editor and poll model support radio, checkbox, slider, and textarea question types.	Code review only. Public poll renderer exists in the Tiptap widget stack.	Full	required	0.89	5	
F045	Widgets	Inline CityChange single-suggestion widget with ID field and address override	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:63-74; src/components/tip-tap/tiptap-node/suggestions/suggestions-node.tsx:37-169; src/components/tip-tap/types/types.ts:76-79	Legacy: The spec originally asked for this as new functionality. New: The editor supports a suggestion ID and variant, but no address-override field exists.	Code review only.	Partial	withdrawn	0.89	3	Withdrawn from pass/fail scoring because the spec later deletes this as net-new scope.
F046	Widgets	Before/after slider 1px white split stroke	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:81-85; src/components/tip-tap/tiptap-widgets/before-after-image/before-after-image.tsx:96-103	Legacy: Legacy had a before/after slider; spec asked for a styling tweak. New: The slider renders a white split line ('w-1 bg-white') and draggable handle.	Code review only.	Full	required	0.89	3	
F047	Editorial taxonomy	Global tag replace / merge workflow	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:88-97; src/server/modules/tag; src/components/dashboard/tag	Legacy: Legacy editorial workflow explicitly needed tag cleanup/merge ability. New: New tag CRUD exists, but no tag replace/merge flow across articles was found.	Code review only.	Skipped	required	0.87	5	
F048	Editorial taxonomy	Improved multi-tag add UX	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:100-108; src/components/form/control/tag.collection.tsx:135-167	Legacy: This was a requested editorial UX improvement. New: Selected tags are maintained separately from search results and the picker clears search state on close, matching the intended flow substantially better than legacy.	Code review only.	Full	required	0.84	3	
F049	Editorial model	Reworked article-type taxonomy and creation menu	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:112-139; src/server/modules/post/types/post-type.enum.ts:1-49; src/components/dashboard/post/post-sidebar-form.tsx:106-123	Legacy: Legacy taxonomy and menu structure were explicitly called illogical and needed restructuring. New: New app still exposes a flat topic enum. Several requested distinctions such as Standard-with-review, sponsored as type, and richer hierarchy/menu semantics are not present as specified.	Code review only.	Partial	required	0.90	6	The platform supports the public families, but not the requested editorial creation model.
F050	Editorial model	Primary topic tag behavior with frontend display and quote icon for 'kommentar'	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:141-159; src/components/shared/post-tag.tsx:30-34; src/components/dashboard/post/post-sidebar-form.tsx:162-164	Legacy: This is a spec delta modeled on legacy pretitle behavior. New: Primary tag exists in the editor, renders on frontend, links to tag pages, and adds a quote icon when the tag name is 'kommentar'.	Local homepage/article rendering plus code review support this.	Full	required	0.90	4	

ID	Area	Capability	Requirement / Refs	Legacy vs New Evidence	Runtime Check	Verdict	Scope	Conf.	Impact	Notes
F051	Editorial UX	Special-character shortcut buttons in editor for em-dash and guillemets	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:164-172; src/components/tip-tap/components/tiptap-toolbar.tsx:132-260	Legacy: This was a new editorial convenience requirement. New: Toolbar buttons exist for formatting, alignment, lists, links, etc., but no explicit shortcut buttons for `—`, `»`, or `«` were found.	Code review only.	Skipped	required	0.91	3	
F052	Routing / SEO	Root-slug article URLs plus redirects from old article paths	Scope: Specs+Live+Repo Refs: KBH 2025 specs.pdf:174-179; https://www.magasinetkbh.dk/search/node/metro; src/config/routes.ts:10-18; src/app/(main)/inhold/[slug]/page.tsx:21-83; src/middleware.ts	Legacy: Legacy site used `inhold/...`; spec requested root-slug URLs with redirects. New: New app still routes many article families under `inhold/\${slug}` instead of root slugs, and no redirect implementation for the old scheme was found.	Local seeded article probe /inhold/byens-kulturelle-liv -> 200.	Skipped	required	0.95	8	This is both a spec miss and a migration/SEO miss.
F053	Editorial media	Bulk-fill photographer and ALT fields for inline images	Scope: Specs+Legacy+Repo Refs: KBH 2025 specs.pdf:182-188; __legacy/sites/all/modules/custom/mkbh_photo_series_bulk_upload; src/components/dashboard/media/media-create-many-form.tsx:61-137; src/components/tip-tap/tiptap-node/inline-image/inline-image-modal.tsx:198-229	Legacy: Legacy had dedicated photo bulk-upload tooling and the spec asked for mass-populate behavior. New: New app lets editors edit alt/caption/author per image, but no one-shot bulk-populate action for all selected images was found.	Code review only.	Skipped	required	0.90	5	
F054	Editorial media	Thumbnail-based inline-image placement in the editor	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:189-191; src/components/tip-tap/tiptap-node/inline-image/inline-image-node.tsx:57-93; src/components/tip-tap/tiptap-node/inline-image/inline-image-modal.tsx:204-210	Legacy: Legacy editor used opaque inline-image tokens; spec asked for thumbnail placement. New: New editor renders selected images as visible previews/thumbnails inside the node UI.	Code review only.	Full	required	0.89	4	
F055	Editorial media	Drag-and-drop reordering of inline-image placeholders	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:192-193; src/components/tip-tap/tiptap-node/inline-image/inline-image-modal.tsx	Legacy: Spec marked this as a nice-to-have. New: No drag-and-drop ordering UI for inline-image placeholders was found in the modal or node.	Code review only.	Skipped	optional	0.86	2	Kept in appendix scoring only; does not drive refund on its own.
F056	Editorial media	Edit inline-image captions directly from the thumbnail UI	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:194-197; src/components/tip-tap/tiptap-node/inline-image/inline-image-modal.tsx:219-229	Legacy: Spec marked this as nice-to-have if UX was good. New: Caption fields are editable directly in the inline-image modal alongside the thumbnail preview.	Code review only.	Full	optional	0.86	2	Optional item; implemented.
F057	Editorial UX	Pre-populate writer/byline from the creating user	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:200-205; src/components/dashboard/post/post-editor.tsx:191-206	Legacy: This was an editorial productivity requirement. New: New post defaults `authorId` to the main profile of the creating user while keeping the field editable.	Code review only.	Full	required	0.88	3	
F058	Search	AND semantics for multi-term search	Scope: Specs+Live+Repo Refs: KBH 2025 specs.pdf:208-212; https://www.magasinetkbh.dk/search/node/metro; src/server/modules/post/repositories/post.repository.ts:381-384	Legacy: Legacy search behavior used OR and the spec explicitly asked to change it. New: New search repository still uses `or(like(title), like(description))`, so the requested AND semantics were not implemented.	Local GET /search?s=test -> 200. Semantics assessed from repository code.	Partial	required	0.94	5	Feature exists, but the required semantic change does not.
F059	Design system	Reuse Caecilia and Montserrat fonts from the old site	Scope: Specs+Legacy+Repo Refs: KBH 2025 specs.pdf:213-221; __legacy/sites/all/themes/custom/mkbh/assets/fonts; src/lib/utills/font.ts:7-37; src/components/tip-tap/components/text-style-control.tsx:54-60	Legacy: Legacy theme ships Caecilia and Montserrat assets. New: New app imports local Caecilia files and Montserrat via next/font, and editor font controls expose both.	Code review only.	Full	required	0.88	2	
F060	Design system	Use additional 'Publish' title font pre-launch	Scope: Specs Refs: KBH 2025 specs.pdf:219-221	Legacy: Originally floated as an extra font change in the spec PDF. New: The same spec explicitly marks this as removed.	Not scored.	Skipped	withdrawn	0.90	1	Withdrawn requirement; excluded from pass/fail scoring.
F061	CityChange	Deep links from CityChange suggestions into the app	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:224-250; src/components/tip-tap/tiptap-widgets/suggestion/suggestion-widget.tsx:227-273	Legacy: Legacy/live widget existed already; spec asked only for deep-link enrichment. New: The existing location-based suggestion widget now builds the requested `citychange.page.link` deep links per suggestion ID.	Code review only.	Full	required	0.90	3	
F062	Paywall	Guest/share links that bypass the paywall	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:252-286; src/components/dashboard/post/post-guest-link-dialog.tsx:17-66; src/components/dashboard/paywall-bypass-token/paywall-bypass-token-list.tsx:63-69	Legacy: Spec asked for one-time and time-window guest links. New: New app can mint tokens with optional expiry, but it does not implement one-time invalidation, list/edit UI is broken, and URL generation is wrong for `inhold/*` content.	Code review only. Public local article route accepts `?token=` in access checks, but no full guest-link workflow was safely exercised.	Partial	required	0.94	8	Feature exists in reduced and unsafe form.
F063	Profile	Profile page list of the user's comments with deep link and reply marker	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:287-296; src/app/(main)/profile/_components/blocks/comments-block.tsx:25-131; src/components/shared/comments/comment-thread.tsx:19-116	Legacy: This was added late in the spec. New: New profile comments block lists user comments, marks replies, and links to the specific comment highlight on the article page.	Code review only.	Full	required	0.88	4	
F064	Payments	Admin control to pause all subscriber payments while preserving access	Scope: Specs+Repo Refs: KBH 2025 specs.pdf:287-341; src/server/modules/payment; src/app/dashboard/payment	Legacy: Spec explicitly asked for a pause-all-payments admin control across Stripe and MobilePay handling. New: No pause-all-subscriber control, service, or admin surface matching this requirement was found.	Code review only.	Skipped	required	0.87	8	
F065	Projects	Project-specific update alerts / email notifications	Scope: Legacy+Repo Refs: https://www.magasinetkbh.dk/projekt/axel-towers; src/components/shared/post-newsletter-form.tsx:61-89; src/server/modules/newsletter/services/newsletter.service.ts:25-37	Legacy: Legacy project detail pages advertise update email alerts. New: The new CTA text promises project-specific alerts, but the handler only subscribes the email to the generic newsletter list and sends no project context.	Project CTA is present in article templates; behavior assessed from code.	Skipped	required	0.96	7	This is a deceptive mismatch between copy and implementation.

ID	Area	Capability	Requirement / Refs	Legacy vs New Evidence	Runtime Check	Verdict	Scope	Conf.	Impact	Notes
F066	Dashboard	Paywall-bypass admin UI integrity	Scope: Legacy+Repo Refs: __legacy/sites/all/modules/custom/mkbh_paywall; src/components/dashboard/paywall-bypass-token/paywall-bypass-token-list.tsx:111-142; src/components/dashboard/paywall-bypass-token/paywall-bypass-token-editor.tsx:38-79; src/app/dashboard/paywall-bypass/tokens/page.tsx	Legacy: Legacy paywall bypass/admin utilities existed in custom paywall tooling. New: New app has token and IP list UIs, but token edit routes are missing and the token editor binds `expireAt` instead of `expiresAt`.	Code review only.	Partial	required	0.95	7	Admin tooling exists but is internally broken.

Audit Findings

These findings rank security, availability, performance, and engineering quality issues by business impact. Scores 9-10 are acceptance blockers.

ID	Category	Title	Impact	Reachable	Evidence	Why It Matters	Recommendation
A001	Security	Tracked secrets and third-party credentials are committed in <code>`.env.example`</code>	10	Repo-wide	<code>`.env.example:29-30, 38-39, 62-63, 75-87, 106-107`</code> include real-looking Better Auth, SMTP host, Google, Facebook, Stripe, and MobilePay credentials.	This is direct credential exposure in a deliverable repository. Even if some keys are test-only, it normalizes unsafe handling and exposes integration surface details that do not belong in versioned examples.	Rotate every exposed secret, purge them from history if possible, and replace the example file with placeholders only.
A002	Security	Paywall-bypass tokens and bypass IPs are publicly enumerable through tRPC	10	Reachable to any client that can call public procedures	<code>src/server/modules/paywall-bypass/paywall-bypass-token.router.ts:28-48</code> and <code>paywall-bypass-ip.router.ts:28-48</code> expose <code>get/list/count</code> as <code>`publicProcedure`</code> ; corresponding services at <code>token.service.ts:66-84</code> and <code>ip.service.ts:61-79</code> do not authorize access.	This turns paywall exception data into public metadata. It undermines the access-control model and exposes operational bypass mechanisms that should be tightly restricted.	Change <code>list/get/count</code> to protected admin-only procedures and enforce gate authorization in the service layer.
A003	Security	Any authenticated user can read and update arbitrary organizations by ID	10	Reachable to any logged-in user	<code>src/app/api/organization/[id]/route.ts:8-50</code> performs only a session check before returning org data; PUT at lines 68-122 updates by raw organization ID with no membership or role check.	This is a read/write insecure direct object reference on organization data. It exposes and allows modification of another organization's profile details with nothing more than a valid session.	Require membership/role checks before GET and PUT, and route these operations through a single authorized service instead of raw table access.
A004	Security	Unauthenticated upload-delete endpoint can delete assets by path or ID	9	Public API surface	<code>src/app/api/upload/delete/route.ts:4-32</code> accepts DELETE with a <code>`path`</code> query parameter and calls <code>deleteAsset</code> ; <code>src/lib/utils/upload-utils.ts:24-33</code> passes the identifier straight to <code>`nup.deleteAsset`</code> .	This is an obvious availability and integrity problem. Any caller that can guess or discover an asset path can attempt deletion without authentication.	Require authentication and ownership/permission checks before deletion. Prefer signed, short-lived deletion intents rather than raw public file-path deletion.
A005	Engineering quality	Public <code>`/test-stripe`</code> demo page ships in the deliverable	9	Public route	<code>src/app/test-stripe/page.tsx:8-195</code> renders a public Stripe test dashboard that attempts subscription upgrades, organization creation, and organization listing.	A public demo/test route on a production magazine codebase is not a harmless leftover. It exposes non-customer flows, confuses acceptance testing, and increases attack surface.	Remove the route from production code or lock it behind an internal-only environment gate.
A006	Engineering quality	Core subscription upgrade path is stubbed while the UI still advertises it	9	Authenticated users	<code>src/server/modules/payment/services/subscription.service.ts:170-175</code> leaves <code>`upgrade()`</code> as TODO and returns <code>`{}`</code> ; <code>src/components/payment/blocks/subscription-card.tsx:144-151</code> still calls it.	This is not a minor defect. It is an acceptance-critical commerce flow presented as if it works.	Implement upgrade end-to-end or remove/disable the UI until it exists.
A007	Engineering quality	Default test command is broken by vendored legacy tests	7	Repo-wide	<code>package.json:39</code> defines <code>`test`</code> as <code>`vitest run`</code> ; local <code>`pnpm test`</code> fails on <code>`__legacy`</code> suites such as <code>beerslider</code> , <code>googleanalytics</code> , <code>jquery-match-height</code> , and <code>Drupal</code> views tests.	A deliverable that cannot pass its default test command without manual scoping is not cleanly maintainable or CI-ready.	Exclude <code>`__legacy`</code> from Vitest, or scope tests to the new app explicitly and add that configuration to the repo.
A008	Availability	Public advertiser page crashes at runtime	7	Public route	Local GET <code>`http://127.0.0.1:3100/annoncering`</code> returned 500; the running server logged <code>`TypeError: a.map is not a function`</code> . Route source: <code>src/app/(main)/annoncering/page.tsx</code> .	This is a public-facing commercial page. If the contractor claimed the site was ready, a deterministic runtime 500 on a public route is an acceptance blocker.	Fix the content-shape assumption in the advertising page renderer and add route smoke tests for all public landing pages.
A009	Performance	Homepage and article routes do heavy server work without evidence of caching discipline	5	Public routes	<code>src/app/page.tsx:38-118</code> fires eleven server fetches for the homepage; article pages such as <code>src/app/(main)/inhold/[slug]/page.tsx:47-66</code> fetch content and trigger analytics on render.	The code may work at low volume, but it leaves too much expensive server work on hot public routes without visible acceptance evidence for caching or performance budgets.	Add explicit caching strategy, route-level performance budgets, and smoke benchmarks for the homepage and article detail routes.
A010	Security / Quality	Auth configuration still contains demo/test-only values	4	Repo-wide	<code>src/server/auth/index.ts:101</code> trusts provider <code>`demo-app`</code> ; <code>src/server/auth/index.ts:473-480</code> injects <code>`dd: "test"`</code> into the session payload.	These are not catastrophic on their own, but they are classic signs of unfinished or non-production auth hardening.	Remove demo-only trusted providers and test payload fields before acceptance.
A011	Engineering quality	Next.js middleware convention is deprecated in the current framework version	3	Build-time	Local <code>`pnpm build`</code> emitted <code>`The "middleware" file convention is deprecated. Please use "proxy" instead.`</code> ; repo uses <code>src/middleware.ts:1-52</code> . Official Next.js 16.1.5 docs confirm <code>`middleware`</code> is deprecated and renamed to <code>`proxy`</code> .	This is not a business-critical defect, but it is a clear sign the contractor did not finish framework hygiene on a modern stack they chose themselves.	Run the official Next.js middleware-to-proxy migration and update the file/export convention.
A012	Security / Ops	Public health endpoint reveals uptime, environment, version, and database state	3	Public route	<code>src/app/api/health/route.ts:4-55</code> returns environment, uptime, version, and database connectivity status to any caller.	Health checks are useful, but this version discloses more operational detail than necessary for a public website.	Restrict detailed health data to internal callers or return a minimal boolean/liveness response publicly.
A013	Release engineering	App startup hard-requires Stripe and MobilePay env vars even outside payment flows	9	Startup / deploy-time	<code>src/config/env.ts:62-84</code> makes Stripe and MobilePay variables mandatory during env validation; <code>src/server/auth/index.ts:221-252</code> registers the Better Auth Stripe integration unconditionally and reads plans from the database.	This blocks minimal bootstraps and makes the app non-portable across environments that only need core web/database/storage services. A release candidate should not refuse to start just because optional payment credentials are absent.	Make payment providers truly optional at startup: gate env validation and provider/plugin registration behind explicit feature flags, and only initialize Stripe/MobilePay when those integrations are enabled.
A014	Ops / Data safety	Default <code>`pnpm db:seed`</code> command is destructive demo seeding, not production bootstrap	10	Repo-wide	<code>package.json:24</code> exposes <code>`db:seed`</code> ; <code>src/server/commands/db-seed.command.ts:35-95</code> starts by calling <code>clearAllSeeder</code> ; <code>src/lib/database/seeder/clear-all-seeder.ts:14-26</code> deletes existing comments, tags, memberships, invitations, and organizations before the script creates demo users/content.	This is a release blocker because the repo's primary seed command is unsafe on a real environment and mixes destructive wipes with demo fixture creation. Running the wrong documented bootstrap step can destroy data and pollute production with test content.	Split seeding into clearly named commands: a non-destructive, idempotent production bootstrap for required system records, and a separate demo/dev seed. Hard-fail the demo seed in production environments.
A015	Release engineering	Fresh databases are incomplete after migrations and require manual Stripe sync plus destructive seed paths	9	Startup / bootstrap	<code>package.json:30</code> wires Stripe sync to a separate <code>`stripe:fetch-plans`</code> command; <code>src/server/commands/payment/stripe-fetch-plans.command.ts:1-15</code> manually runs <code>`syncAllProducts()`</code> and <code>`syncAllPrices()`</code> ; <code>src/server/modules/payment/services/stripe.service.ts:33-50</code> fetches products/prices from Stripe, while <code>src/server/auth/index.ts:239-252</code> expects active prices to already exist in the database; <code>src/server/database/seeder/payment-plan.seeder.ts:210-219</code> skips missing plans, and <code>src/server/database/seeder/payment-price.seeder.ts:102-117</code> only updates rows that already exist.	A clean database is not release-ready after migrations alone. Checkout plan resolution depends on Stripe-backed plan/price rows being present, but the repo provides no safe, idempotent production bootstrap for that catalog or for required content/admin scaffolding.	Automate plan and price reconciliation as part of startup or a dedicated idempotent production bootstrap, and keep the mapping driven from live Stripe product metadata instead of contractor-specific preloaded rows. Add a separate non-destructive bootstrap for required pages, blocks, and admin state.

ID	Category	Title	Impact	Reachable	Evidence	Why It Matters	Recommendation
A016	Release engineering	Deployment tooling mixes dev-only mounts, fake production tags, and machine-specific compose files	8	Repo-wide	docker-compose.yml:28-31 and 43-55 still use `dev` image tags and bind mounts; docker-compose.yml:93-99 mounts source and env files directly into the running container; Makefile:189-198 pushes a Swarm-style `docker stack deploy` path; infrastructure/docker/docker-compose.yml:9-10,42-43,69-72 hardcodes another developer's local filesystem.	This is not a cosmetic cleanup issue. It signals that the deployment story is fragmented and partly non-portable, which increases the chance that the accepted deployment path is undocumented, unreproducible, or simply not the one checked into the repo.	Separate development and production orchestration cleanly, remove bind mounts and `dev` tags from release definitions, delete or template machine-specific compose files, and document one reproducible deployment path that actually matches the target platform.
A017	Security / Release engineering	Mailchimp config is split between env and database, and the DB-backed secret path is publicly readable	9	Public tRPC + runtime + scheduled jobs	src/server/modules/site-config/router/index.ts:7-13 exposes `values` as `publicProcedure`; src/server/modules/site-config/services/site-config.service.ts:21-31 returns the full config map; src/server/modules/site-config/validators/site-config.validator.ts:10-12 includes `MAILCHIMP_API_KEY` and `MAILCHIMP_LIST_ID`; src/components/dashboard/site-config/config-editor.tsx:579-590 writes those values through the admin UI; meanwhile src/server/modules/newsletter/services/newsletter.service.ts:17-21 still defaults the newsletter list ID from env, while src/server/clients/mailchimp/mailchimp.service.ts:16-19 and src/lib/mailchimp/client.ts:6-12 read the API key from site-config; src/server/commands/site-config-update-stats.command.ts:20-35 also reads the list ID from site-config and requires an admin user.	This is worse than ordinary config drift. The repo creates two competing Mailchimp configuration paths, and the one stored in the database is exposed through a public query if operators use the dashboard for secret storage.	Remove secrets from public site-config entirely, split secret config from public metadata config, and enforce one authoritative Mailchimp configuration path for app and job runtime.
A018	Data integrity / Analytics	Google Analytics sync injects hardcoded fake page-view data	8	Analytics sync command	src/server/clients/google-analytics.service.ts:66-104 appends `result.set('kort/test', 1000)` after the real GA report loop; src/server/commands/update-ga-page-views.command.ts:11-24 persists every returned path into analytic views.	This poisons the analytics dataset with fabricated rows and proves the sync path is not trustworthy for acceptance, reporting, or business decisions.	Delete the hardcoded test record, quarantine analytics fixtures to test-only code, and add a regression test that rejects unknown synthetic paths in production syncs.
A019	Release engineering / Analytics	GA credentials are wired as a file-path assumption rather than a deploy-safe secret input	8	Build and runtime configuration	src/server/clients/google-analytics.service.ts:15-19 constructs `BetaAnalyticsDataClient()` with default ADC behavior; src/config/env.ts:90-93 exposes only `GOOGLE_APPLICATION_CREDENTIALS`; .env.example:70 points that variable at `ga-service-account.json`; docker-compose.dev.yml:108-110 passes the path through. Repo search of tracked deploy files found no credential JSON mount or copy beyond that env reference.	The deployment depends on an undocumented credential file appearing inside the runtime container. That is brittle, hard to reproduce, and not encoded in the checked-in deployment path.	Support env-injected JSON credentials or workload identity, and fail bootstrap with an explicit validation error if the analytics auth mechanism is incomplete.
A020	Release engineering / Payments	Stripe catalog sync exists, but plan typing and checkout behavior still depend on mutable plan names	9	Checkout, sync, and subscription update flows	src/server/modules/payment/services/stripe.service.ts:54-61 falls back from `metadata.type` to `name.includes('business')`; src/server/auth/index.ts:268-282 derives business and annual behavior from `plan.name`; src/components/subscription/subscription-button.tsx:8-29 upgrades by arbitrary `planName`; src/app/(main)/profile/organization/_components/organization-subscription.tsx:88-97 sends `selectedPlan.plan` into upgrade; meanwhile metadata is already modeled in src/server/modules/payment/types/plan-metadata.type.ts:17-25 and src/server/modules/payment/types/plan-metadata.type.ts:1-6.	This undermines the whole point of syncing the Stripe catalog. A renamed product or mismatched label can flip business vs personal logic or billing period behavior even when Stripe metadata is correct.	Make metadata authoritative everywhere, pass stable plan identifiers from UI to backend, and remove name-based heuristics from sync and checkout logic.
A021	Release engineering / Runtime	Every runtime container and scheduled job runs migrations on boot	8	Container startup and cron jobs	Dockerfile:115-117 and Dockerfile.railway:87-88 set `docker-entrypoint.sh` for all runtime containers; infrastructure/docker/common/docker-entrypoint.sh:66-76 and 104-119 always wait for DB and run `node migrate.js` before the requested command; docker-compose.dev.yml:134-195 defines scheduled job services with the same image and endpoint pattern.	App replicas, one-off jobs, and cron tasks all become implicit migration runners. That creates avoidable startup coupling, migration races, and failure modes where an unrelated job cannot run because schema migration behavior changed.	Move schema migration to an explicit one-shot deploy step and keep app and job containers focused on their actual command.
A022	Security / Ops	Migration script logs the full database connection string	7	Migration logs	migrate.js:27-34 prints debug state and then logs `Connection string value:` followed by the raw connection string.	That leaks database credentials into CI, container, or platform logs. It is careless at best and operationally unsafe in any shared logging environment.	Redact secrets in migration logging and keep only non-sensitive connection diagnostics such as host classification or presence checks.
A023	Release engineering	Checked-in bootstrap scripts are local-dev oriented, partially broken, and seed localhost content	8	Repository bootstrap path	infrastructure/seed_db.sh:135-145 checks for non-existent `src/lib/database/seed.ts` and tells operators to run missing `./infrastructure/setup_docker.sh`; infrastructure/setup_env.sh:64-132 generates a localhost-only `.env.local`; infrastructure/seed_db.sh:179-195 advertises seeded demo accounts and localhost services; src/server/database/seeders/page.seeder.ts:755-763 hardcodes `http://localhost:3000/faq` into seeded page content.	The documented bootstrap path is not production-ready. It is misleading, partly broken, and it injects developer URLs and demo assumptions into the dataset that would be unacceptable in a real environment.	Split dev bootstrap from production bootstrap, fix the broken script references, remove localhost URLs from seeded content, and make base content creation idempotent and environment-aware.
A024	Engineering quality / Admin tooling	The dashboard 'Google' integration UI does not control the actual analytics integration	6	Admin dashboard + runtime config	src/server/modules/site-config/enums/site-config-key.enum.ts:7-9 only defines `GOOGLE_ADS_ID` for Google-related site-config; src/components/dashboard/site-config/config-editor.tsx:571-577 exposes only 'Google ADS ID' in the integrations tab; actual page tagging uses env-backed config in src/app/layout.tsx:31 and src/config/env.ts:240-244, while server-side GA reporting uses env-backed config in src/server/clients/google-analytics.service.ts:17-18.	Operators are shown a Google integration control panel that does not actually govern the live analytics implementation. That is a fake control surface and a reliable way to misconfigure production.	Either wire the admin UI to the real integration state or remove it. Keep Ads, GA tag ID, GA property ID, and service-account auth as explicit, separately documented settings.
A025	Release engineering / Config	Hardcoded domains and localhost defaults still leak into auth, build config, and seeded content	7	Runtime config + seeded CMS content	src/server/auth/index.ts:489-492 falls back to `magasinetkbh.vercel.app` for cross-subdomain cookies; Dockerfile:46-60 bakes localhost and dummy host defaults into build-time auth/app/mobilepay env; src/server/database/seeders/page.seeder.ts:755-763 seeds `http://localhost:3000/faq`; src/app/(main)/payments/personal/page.tsx:28-52 and src/app/(main)/payments/sponsorship/page.tsx:20-44 hardcode live-domain share URLs and legacy asset URLs.	Wrong host defaults do not stay theoretical. They bleed into cookie scoping, generated links, seeded pages, and public marketing surfaces, which makes deployment behavior environment-dependent and error-prone.	Move every host/domain value behind validated configuration, remove hardcoded domains from seed data and page constants, and fail builds when required public URLs are missing instead of falling back to localhost or contractor domains.

ID	Category	Title	Impact	Reachable	Evidence	Why It Matters	Recommendation
A026	Release engineering / Operations	Railway deployment definition omits the scheduled job lifecycle entirely	9	Target-platform deployment	railway.toml:1-8 defines only build and healthcheck for a single web service; scheduled work exists only in swarm compose, with cron infra at docker-compose.dev.yml:27-38 and jobs at docker-compose.dev.yml:134-195 for MobilePay processing, site-config stats, GA sync, and scheduled publishing.	A Railway deployment can come up 'healthy' while silently missing the background work that keeps subscriptions, analytics, and scheduled content alive. That is an incomplete production topology, not a minor ops TODO.	Model the target deployment as multiple first-class services or platform schedulers and include those workers in the checked-in Railway deployment path.

Mocked / Hardcoded / Faked Functionality

This section isolates surfaces where the implementation presents itself as functional while relying on static data, dead handlers, demo scaffolding, or materially reduced behavior.

ID	Surface	Type	Evidence	User Visible Effect	Severity
M001	Archive page	hardcoded data	src/app/(main)/arkiv/page.tsx:19-99 defines a fixed `items` array with five records and repeated image/pdf assets while the page copy claims all 57 issues are available.	Users see something that looks like an archive, but it is a fabricated five-row sample instead of the real legacy archive.	10
M002	Newsletter page	dead submit handler	src/app/(main)/nyhedsbrev/_components/newsletter-form.tsx:35-39 leaves the submit action as a comment placeholder.	The primary newsletter signup page looks complete, accepts input, and then does nothing.	10
M003	Subscription upgrade	stubbed backend	src/server/modules/payment/services/subscription.service.ts:170-175 returns `{}` from `upgrade()` while src/components/payment/blocks/subscription-card.tsx:144-151 still calls it.	Users are offered an upgrade flow that is not implemented.	10
M004	Project alerts	misrepresented behavior	src/components/shared/post-newsletter-form.tsx:61-69 sends only `{ email }` to the generic newsletter mutation, despite CTA copy promising project-specific alerts.	The UI claims project update alerts, but the implementation is just a generic newsletter signup.	9
M005	Public Stripe test page	demo/test surface	src/app/test-stripe/page.tsx:8-195 ships a public demo control panel for test subscription and organization actions.	A non-customer demo flow is presented on the public app surface.	8
M006	Personal payments page	hardcoded marketing data	src/app/(main)/payments/personal/page.tsx:136-175 hardcodes `2808` members and `MobilePay 711911`; lines 28-52 use legacy asset URLs.	The page mixes real dynamic plan data with fake/static proof points that can mislead acceptance review and end users.	7
M007	Sponsorship page	manual fallback posing as productized flow	src/app/(main)/payments/sponsorship/page.tsx:92-97 converts through `mailto:` and lines 20-44 and 111-117 pull legacy-hosted assets.	The surface looks migrated, but conversion is still manual and piggybacks on legacy assets.	6
M008	Guest links	reduced implementation sold as full feature	src/components/dashboard/post/post-guest-link-dialog.tsx:17-66 only supports an optional expiration date; it does not implement the requested one-time mode and builds the link with raw `token` querystring.	Editors are given a 'guest link' tool that looks finished, but it omits key promised behavior.	6
M009	Organization pricing component	dead hardcoded fallback	src/app/(main)/profile/organization/_components/organization-content.tsx:19-110 contains a hardcoded business plan table and `currentPlan` fixed to `undefined`, but the active route uses a different component.	Not currently user-visible on the active route, but it is still unfinished fallback code that imitates real pricing logic.	3
M010	Google Analytics sync	hardcoded fake data	src/server/clients/google-analytics.service.ts:102 injects `/kort/test` with 1000 views, and src/server/commands/update-ga-page-views.command.ts:14-23 persists that fabricated row like real analytics data.	Analytics can appear populated and healthy even when part of the dataset is synthetic contractor filler.	9

Appendices

Optional, Withdrawn, and Out-of-Scope Items

ID	Capability	Scope Modifier	Verdict	Notes
F019	Pre-launch data migration	out_of_scope	Skipped	Not scored in acceptance verdict for this report.
F045	Inline CityChange single-suggestion widget with ID field and address override	withdrawn	Partial	Withdrawn from pass/fail scoring because the spec later deletes this as net-new scope.
F055	Drag-and-drop reordering of inline-image placeholders	optional	Skipped	Kept in appendix scoring only; does not drive refund on its own.
F056	Edit inline-image captions directly from the thumbnail UI	optional	Full	Optional item; implemented.
F060	Use additional 'Publish' title font pre-launch	withdrawn	Skipped	Withdrawn requirement; excluded from pass/fail scoring.

Evidence Notes

Legacy baseline URLs checked in this pass: <https://www.magasinetkbh.dk/>, <https://www.magasinetkbh.dk/projekter>, <https://www.magasinetkbh.dk/arkiv>, <https://www.magasinetkbh.dk/nyhedsbrev>, <https://www.magasinetkbh.dk/search/node/metro>, <https://www.magasinetkbh.dk/projekt/axel-towers>, <https://www.magasinetkbh.dk/vision/lav-cykelparkering-over-alle-hovedbanegarden-perroner>.

Local smoke highlights: `/`, `/arkiv`, `/nyhedsbrev`, `/payments/personal`, `/payments/sponsorship`, `/payments/business`, `/projekter`, `/anmeldelser`, `/foto`, `/opinion`, `/visioner`, `/kbhplus`, `/bydel/indre-by`, `/emne/metro`, and seeded `/inhold/...` routes returned HTTP 200; `/annoncering` returned HTTP 500; `/test-stripe` returned HTTP 200 and exposed a public Stripe test page.

Framework note validated against official docs: Next.js 16.1.5 deprecates the `middleware` file convention in favor of `proxy`, which matches the local build warning emitted by this repository.