

# CorePOS Project Dashboard

One compact CorePOS planning view with a high-level dashboard and the deeper Living Feature Map in the same static page.

Generated 2026-05-18T12:48:15.307Z Thailand time 18 May 2026, 19:48 ICT Source commit fbbe355459ff

Primary share link [./PROJECT\\_DASHBOARD.html](/PROJECT_DASHBOARD.html)

Clapham Cycle core go-live

**72%**

On track for 2026-08-01.

**22 tracked launch fundamentals**

Full product/commercial

**63%**

Includes wider rollout maturity and add-on scope.

Bolt-on maturity

**50%**

QuickBooks, rental/hire, widgets, suppliers, and other optional packages.

Own-shop trial use

**70%**

Immediate controlled single-shop trial readiness across operationally relevant areas.

Future ideas

**1 item**

Parked outside current Clapham Cycle core scope.

**Open the parked list**

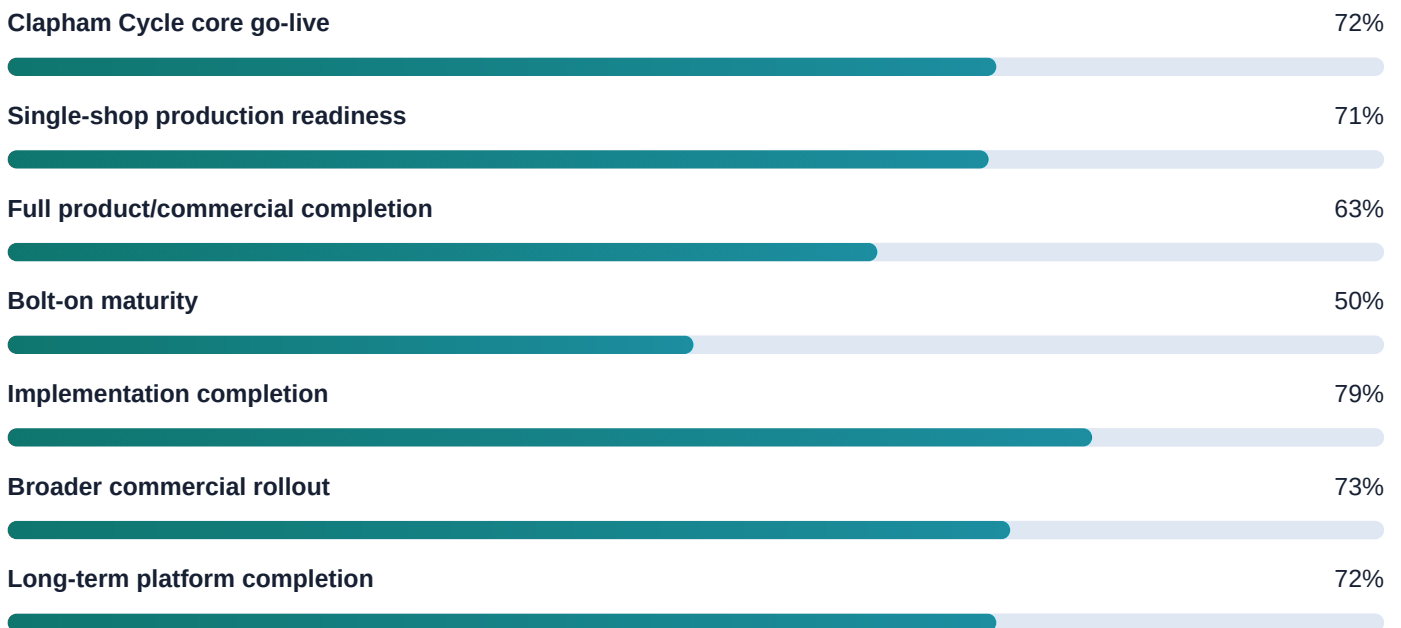
Long-term platform completion

**72%**

Weighted completion across the tracked platform feature map.

## CURRENT SNAPSHOT

### Readiness at a glance



Clapham Cycle core go-live is deliberately narrower than full product/commercial maturity. Bolt-ons and future ideas do not lower the core score unless Thomas explicitly moves them into launch scope.

## SOFT GO-LIVE TARGET

**Yes, at the current recent rate the core scope still fits inside the target window.**

Current status: On track

Projected threshold date  
**2026-06-20**

Schedule position  
**42 days ahead of target**

Target  
**2026-08-01**

Days remaining  
**75**

Core gap  
**13%**

Estimated time to threshold  
**33 days**

Confidence band: 4 weeks to 6 weeks.

### Last 30 days snapshot

Commits  
**802**

PR / merge signals  
**599**

Roadmap events  
**91**

Net changed lines  
**372,112**

**Reason:** The current heuristic completion window fits inside the time remaining to the soft go-live target.

External single-store pilot planning starts only after 3-6 months stable Clapham Cycle operation. From the target date, that means 2026-11-01 to 2027-02-01, subject to real stability evidence.

## What counts toward the current launch score

Switch between the current Clapham core launch scope, the later commercial expansion scope, and the bolt-ons or parked work that should not dilute the core percentage.

Core launch

Commercial expansion

Bolt-ons / parked

### POS / till / retail workflow

Operational / Staff Workflow

Active Hardening

80% progress 12% of core

2 modules

Progress

80%

Scope share

12%

Modules

2

POS, till, in-store payments, receipts, quotations, layaways

Refunds, exchanges, gift cards, liabilities, and stored value

### Workshop UX and behaviour audit

Operational / Staff Workflow

Built, Needs Hardening

76% progress 7% of core

1 modules

Progress

76%

Scope share

7%

Modules

1

Workshop jobs, estimates, approvals, parts, collection, print

### Stocktake and inventory accuracy

Operational / Staff Workflow

Built, Needs Hardening

72% progress 6% of core

1 modules

Progress

72%

Scope share

6%

Modules

1

Inventory, stock movements, stocktake, transfers, replenishment

### Procurement and receiving maturity

Operational / Staff Workflow

Active Hardening

72% progress 6% of core

1 modules

Progress

72%

Scope share

6%

Modules

1

### Diagnostics, support packs, and operational readiness

Operational / Staff Workflow

Built, Needs Hardening

72% progress 6% of core

1 modules

Progress

72%

Scope share

6%

Modules

1

### Online workshop booking

Customer Facing

Active Hardening

72% progress 5% of core

1 modules

Progress

72%

Scope share

5%

Modules

1

Procurement, purchase orders, receiving, supplier returns

Deployment, diagnostics, support packs, backups, integrity

Workshop diary, scheduler, capacity, and online workshop booking

### Financial reporting, store analysis, BI, and management dashboards

Active Hardening

Active Hardening

74% progress 5% of core

1 modules

Progress  
74%

Scope share  
5%

Modules  
1

Financial reporting, store analysis, BI, and management dashboards

### Cash-up, daily close, trade close, cash oversight

Active Hardening

Active Hardening

78% progress 5% of core

1 modules

Progress  
78%

Scope share  
5%

Modules  
1

Cash-up, daily close, trade close, cash oversight

### Customer bike lifecycle CRM

Bike-shop-specific

Built, Needs Hardening

73% progress 5% of core

1 modules

Progress  
73%

Scope share  
5%

Modules  
1

CRM, customer management, customer bikes, and service context

### Supplier / Madison integration

Operational / Staff Workflow

Active Hardening

70% progress 5% of core

1 modules

Progress  
70%

Scope share  
5%

Modules  
1

Supplier integration foundation and Madison baseline path

### Permissions and sensitive action hardening

Operational / Staff Workflow

Foundation Exists

66% progress 5% of core

1 modules

Progress  
66%

Scope share  
5%

Modules  
1

Roles, permissions, login, PIN auth, sensitive-action controls

### Clapham Cycle launch rehearsal, live data setup, and acceptance sign-off

Planning

Planning

35% progress 5% of core

1 modules

Progress  
35%

Scope share  
5%

Modules  
1

Clapham Cycle launch rehearsal, live data setup, and acceptance sign-off

### Communication Centre: email, SMS, WhatsApp

Customer Facing

### System configuration, first-run onboarding, dashboard settings

Built, Needs Hardening

### Rota, staff scheduling, closed days, working hours

Built, Needs Hardening

**Built, Needs Hardening**

66% progress 4% of core

1 modules

Progress

66%

Scope share

4%

Modules

1

Customer communication basics and account-visible communication history

**Built, Needs Hardening**

70% progress 4% of core

1 modules

Progress

70%

Scope share

4%

Modules

1

System configuration, first-run onboarding, dashboard settings

**Built, Needs Hardening**

66% progress 4% of core

1 modules

Progress

66%

Scope share

4%

Modules

1

Rota, staff scheduling, closed days, working hours

**Staff records and staff directory**

Foundation Exists

**Foundation Exists**

78% progress 3% of core

1 modules

Progress

78%

Scope share

3%

Modules

1

Staff records and staff directory

**Google Business Profile / Reviews / Maps**

Customer Facing

**Foundation Exists**

64% progress 3% of core

1 modules

Progress

64%

Scope share

3%

Modules

1

Store Info, opening hours, public business presence, basic Google Business Profile foundation

**Staff performance reporting**

Built, Needs Hardening

**Built, Needs Hardening**

88% progress 3% of core

1 modules

Progress

88%

Scope share

3%

Modules

1

Staff performance reporting

**Verification / multi-agent tooling**

Platform / Commercial

**Trial Ready**

88% progress 3% of core

1 modules

Progress

88%

**Web builder / templates / embeddable widgets**

Customer Facing

**Foundation Exists**

78% progress 2% of core

1 modules

Progress

78%

**Stripe/payment trust**

Platform / Commercial

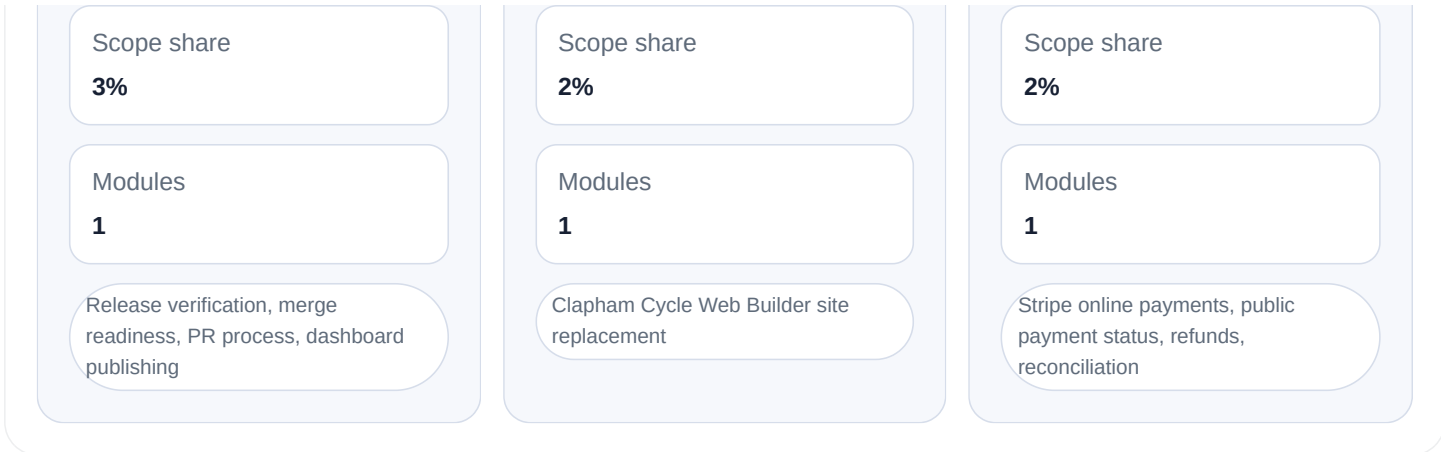
**Active Hardening**

75% progress 2% of core

1 modules

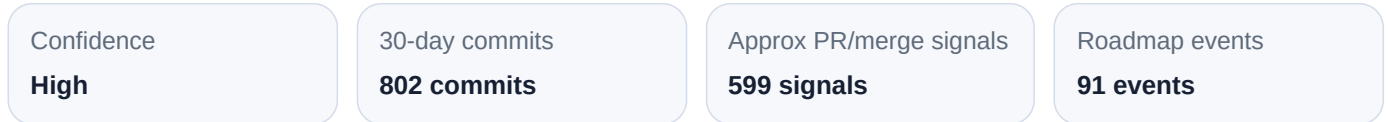
Progress

75%



**WORK-RATE**

**Recent delivery signals**

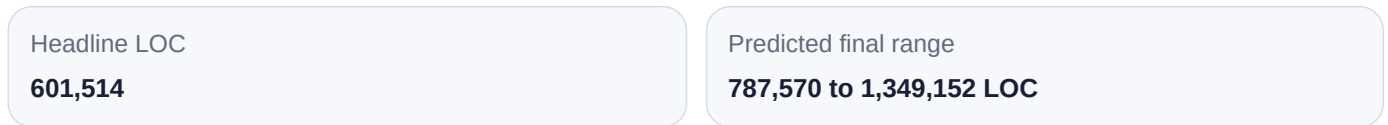


**History note:** Approximation based on full local origin/main history, including merge commits and squash-merge-style PR subjects where detectable.

- **7-day commits:** 315 commits
- **14-day commits:** 646 commits
- **30-day roadmap events:** 91 events
- **Changed lines (30d):** +397,739 / -25,627 (net 372,112)

**LOC SUMMARY**

**Current size and likely range**



**LOC source:** `cloc . --exclude-dir=node_modules,dist,build,.git`  
 LOC remains a size and complexity signal, not a quality score.

## Current blockers

- **Workshop UX and behaviour audit:** A single workshop readiness audit is not yet codified, so UX debt is spread across several surfaces. **high**
- **Procurement and receiving maturity:** Supplier API ordering, supplier returns/RMA, supplier documents, and procurement accountability still need more operational polish. **high**
- **Supplier / Madison integration:** Live sync, supplier API ordering, and normalization maturity are still intentionally incomplete. **high**
- **Permissions and sensitive action hardening:** Sensitive action review is still distributed across many modules rather than a single hardening ledger. **high**
- **Diagnostics, support packs, and operational readiness:** Support-safe troubleshooting flows still need tighter packaging for real trial use. **high**
- **Track My Bike / Workshop Progress View:** Portal polish, customer wording, and broader regression coverage still need hardening before it feels fully trial-safe. **high**
- **Communication Centre: email, SMS, WhatsApp:** Channel reliability, support tooling, and operator trust signals still need more maturity than the raw transport foundation. **high**
- **Customer account / portal:** Confirmed rental booking readback and storefront order-history readback are account-only and view-only; rental self-service, public confirmation, secure readback tokens, customer-visible receipt/invoice controls, and broader longitudinal account history remain future work. **high**

## RECOMMENDED NEXT TASKS

### Best next planning moves

- **Workshop UX and behaviour audit:** Define a workshop hardening checklist covering queue, job detail, check-in, bookings, collection, and portal handoff. **high**
- **Workshop UX and behaviour audit:** Add a compact workshop regression bundle for the highest-risk operator flows. **high**
- **Procurement and receiving maturity:** Turn the procurement gap audit into a living maturity checklist with explicit completion criteria. **high**
- **Procurement and receiving maturity:** Add targeted regression coverage for discrepancy and exception-resolution paths. **high**
- **Supplier / Madison integration:** Run supervised Madison/procurement pilot sessions and use readiness decisions, the operating dashboard, and bounded evidence reports to decide whether narrow manager-led limited use remains justified. **high**
- **Permissions and sensitive action hardening:** Map the highest-risk manager/admin actions into an explicit permissions hardening checklist. **high**
- **Diagnostics, support packs, and operational readiness:** Keep the dashboard's trial-readiness section concise as diagnostics, integrity, and support-pack maturity continue to grow. **high**
- **Track My Bike / Workshop Progress View:** Add a focused portal hardening checklist covering progress wording, approvals, attachments, and reply flows. **high**
- **Communication Centre: email, SMS, WhatsApp:** Translate post-service follow-up consent, suppression, review-request wording, Google separation, and approval policy into explicit product-data requirements before any outbound workflow is considered. **high**
- **Communication Centre: email, SMS, WhatsApp:** Clarify the maturity ladder for workshop notifications versus broader omnichannel communication. **high**

## NEWEST IDEAS

### Worth keeping in view

- **Communication Centre: email, SMS, WhatsApp:** Track communication maturity by channel trust, orchestration depth, and customer self-service coverage.
- **Customer account / portal:** Split portal maturity into identity, self-service, and ongoing customer relationship stages.
- **Customer bike lifecycle CRM:** Create a bike-lifecycle maturity band that combines reminders, history, portal access, and workshop continuity.
- **Customer-facing storefront:** Track storefront maturity by browse/catalogue, checkout, fulfilment, and post-purchase support.
- **Deployment / Windows / Caddy / production serving:** Treat deployment readiness as its own recurring review lane rather than burying it inside docs.
- **Diagnostics, support packs, and operational readiness:** Track support-bundle completeness as a first-class maturity signal beside implementation.
- **Google Business Profile / Reviews / Maps:** Add a dedicated Google presence maturity lane only after controlled posting has operational evidence or public local-marketing value becomes real.
- **HR personnel foundation add-on:** Keep Problem Cycle-specific bonus calculation separate from the reusable commercial HR module unless a configurable framework is deliberately commissioned.

## Core fundamentals behind the 72%

These are the tracked modules currently counted into Clapham Cycle core go-live. The weighted average of their progress makes up the headline core percentage.

Tracked modules **22**

Current average **72%**

Highest progress **88%**

Lowest progress **35%**

Order by

Highest progress

### Staff performance reporting

No linked feature card

Built, Needs Hardening

Progress

**88%**

Launch share

**3%**

Progress toward launch-ready

88%

Share of the core basket

3%

### Release verification, merge readiness, PR process, dashboard publishing

Verification / multi-agent tooling

Trial Ready

Progress

**88%**

Launch share

**3%**

Progress toward launch-ready

88%

Share of the core basket

3%

### POS, till, in-store payments, receipts, quotations, layaways

POS / till / retail workflow

Active Hardening

Progress

**82%**

Launch share

**7%**

Progress toward launch-ready

82%

Share of the core basket

7%

### Clapham Cycle Web Builder site replacement

Web builder / templates / embeddable widgets

Built, Needs Hardening

Progress

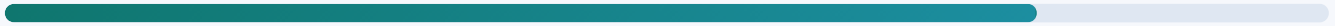
78%

Launch share

2%

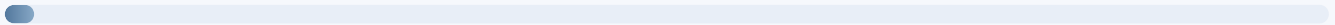
Progress toward launch-ready

78%



Share of the core basket

2%



Cash-up, daily close, trade close, cash oversight

No linked feature card

Active Hardening

Progress

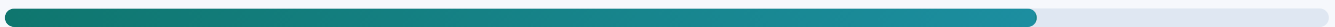
78%

Launch share

5%

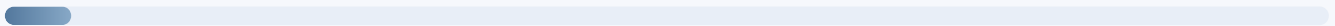
Progress toward launch-ready

78%



Share of the core basket

5%



Staff records and staff directory

No linked feature card

Foundation Exists

Progress

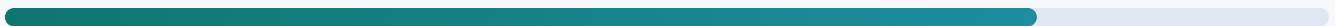
78%

Launch share

4%

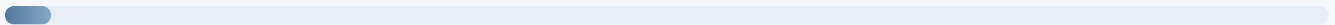
Progress toward launch-ready

78%



Share of the core basket

4%



Refunds, exchanges, gift cards, liabilities, and stored value

POS / till / retail workflow

Built, Needs Hardening

Progress

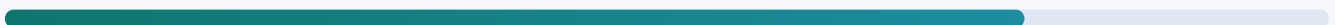
77%

Launch share

5%

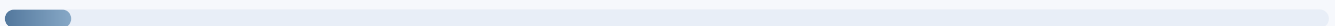
Progress toward launch-ready

77%



Share of the core basket

5%



Workshop jobs, estimates, approvals, parts, collection, print

Workshop UX and behaviour audit

Built, Needs Hardening

Progress

76%

Launch share

7%

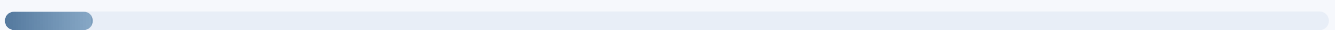
Progress toward launch-ready

76%



Share of the core basket

7%



### Stripe online payments, public payment status, refunds, reconciliation

Stripe/payment trust

Active Hardening

Progress

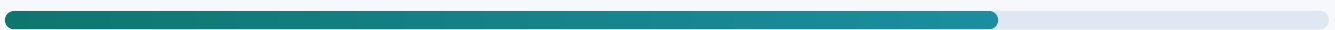
75%

Launch share

2%

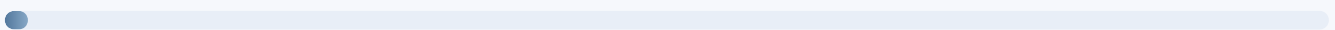
Progress toward launch-ready

75%



Share of the core basket

2%



### Financial reporting, store analysis, BI, and management dashboards

No linked feature card

Active Hardening

Progress

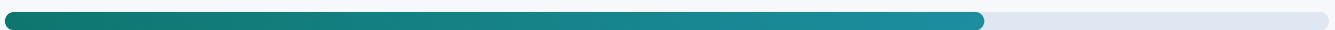
74%

Launch share

5%

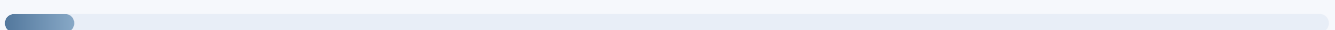
Progress toward launch-ready

74%



Share of the core basket

5%



### CRM, customer management, customer bikes, and service context

Customer bike lifecycle CRM

Built, Needs Hardening

Progress

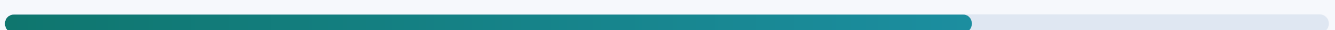
73%

Launch share

5%

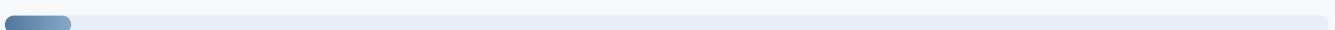
Progress toward launch-ready

73%



Share of the core basket

5%



### Deployment, diagnostics, support packs, backups, integrity

Diagnostics, support packs, and operational readiness

Built, Needs Hardening

Progress

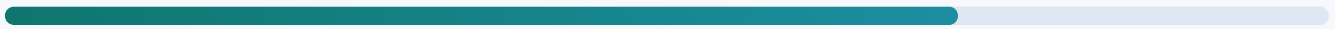
72%

Launch share

6%

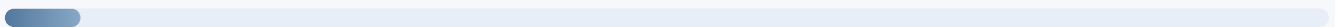
Progress toward launch-ready

72%



Share of the core basket

6%



### Inventory, stock movements, stocktake, transfers, replenishment

Stocktake and inventory accuracy

Built, Needs Hardening

Progress

72%

Launch share

6%

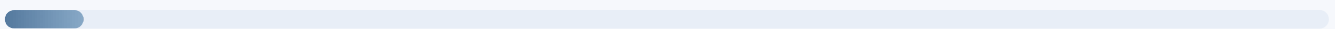
Progress toward launch-ready

72%



Share of the core basket

6%



### Procurement, purchase orders, receiving, supplier returns

Procurement and receiving maturity

Active Hardening

Progress

72%

Launch share

6%

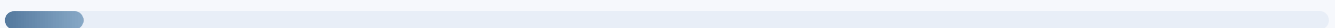
Progress toward launch-ready

72%



Share of the core basket

6%



### Workshop diary, scheduler, capacity, and online workshop booking

Online workshop booking

Active Hardening

Progress

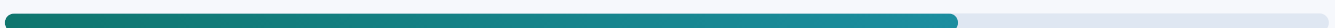
72%

Launch share

6%

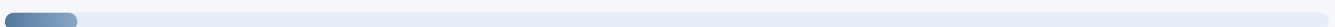
Progress toward launch-ready

72%



Share of the core basket

6%



### Supplier integration foundation and Madison baseline path

Supplier / Madison integration

Active Hardening

Progress

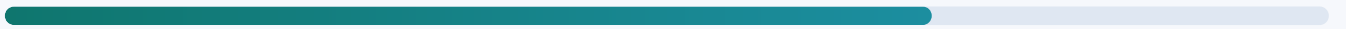
70%

Launch share

5%

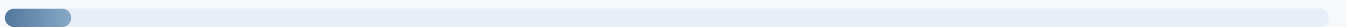
Progress toward launch-ready

70%



Share of the core basket

5%



### System configuration, first-run onboarding, dashboard settings

No linked feature card

Built, Needs Hardening

Progress

70%

Launch share

4%

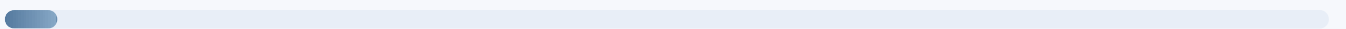
Progress toward launch-ready

70%



Share of the core basket

4%



### Customer communication basics and account-visible communication history

Communication Centre: email, SMS, WhatsApp

Built, Needs Hardening

Progress

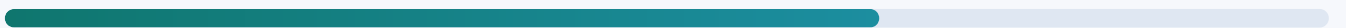
66%

Launch share

5%

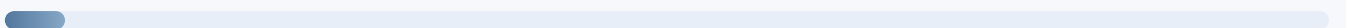
Progress toward launch-ready

66%



Share of the core basket

5%



### Roles, permissions, login, PIN auth, sensitive-action controls

Permissions and sensitive action hardening

Foundation Exists

Progress

66%

Launch share

5%

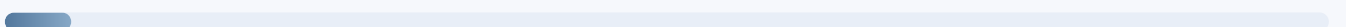
Progress toward launch-ready

66%



Share of the core basket

5%



### Rota, staff scheduling, closed days, working hours

No linked feature card

Built, Needs Hardening

Progress

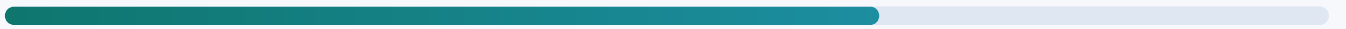
66%

Launch share

4%

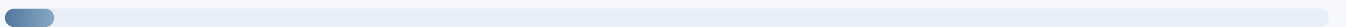
Progress toward launch-ready

66%



Share of the core basket

4%



### Store Info, opening hours, public business presence, basic Google Business Profile foundation

Google Business Profile / Reviews / Maps

Foundation Exists

Progress

64%

Launch share

3%

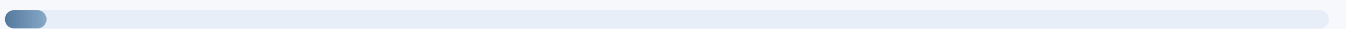
Progress toward launch-ready

64%



Share of the core basket

3%



### Clapham Cycle launch rehearsal, live data setup, and acceptance sign-off

No linked feature card

Planning

Progress

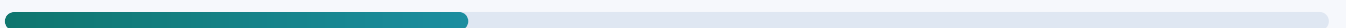
35%

Launch share

5%

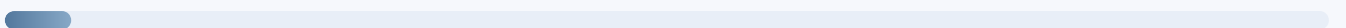
Progress toward launch-ready

35%



Share of the core basket

5%



## Parked future ideas (1 item)

These items are intentionally parked outside the current Clapham Cycle launch scope, so they do not reduce the core go-live score unless their classification changes.

These are rough planning heuristics, not committed estimates. They use the current repo size, recent delivery pace, and how early each idea still is.

### Promotions, vouchers, loyalty, advanced discounting

No linked feature card yet

Planning

Outside Clapham Cycle core go-live unless Thomas explicitly adds it.

Estimated size

**Large**

Heuristic LOC

**14,369-29,843**

Heuristic time

**4.1 months to 6.2 months**

Share of total product

**5%**

**Current gap:** No full promotion, loyalty, voucher, or marketing-discount engine should be assumed from support docs alone.

**Next decision:** Decide whether promotions/loyalty enter a later commercial phase or remain parked.

## Scope classification inventory

Clapham Cycle core completion only moves when scope is explicitly classed as Clapham Cycle core. Bolt-ons and future ideas must not silently lower the core go-live score.

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Clapham Cycle launch rehearsal, live data setup, and acceptance sign-off	LAUNCH_BLOCKING_GAP	Planning	Launch-blocking governance gap for the 2026-08-01 soft go-live target.	Reusable launch process for external pilots later.	Not a bolt-on
Clapham Cycle Web Builder site replacement	LAUNCH_BLOCKING_GAP	Built, Needs Hardening	Explicitly in Clapham Cycle go-live scope: the CorePOS-generated website is intended to replace the existing Clapham Cycle website after controlled readiness and migration work.	Provides the first real proof case for the later commercial Builder add-on.	Not a generic bolt-on for Clapham Cycle go-live; broader Web Builder packaging for future stores remains a bolt-on/commercial add-on.
Cash-up, daily close, trade close, cash oversight	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for end-of-day operation.	Required for every store.	Not a bolt-on
CRM, customer management, customer bikes, and service context	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required for Clapham Cycle retail and workshop continuity.	Required for bike-shop differentiation.	Not a bolt-on
Customer communication basics and account-visible communication history	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required at basic level for workshop and customer follow-up.	Required for customer-service maturity.	Not a bolt-on
Deployment, diagnostics, support packs, backups, integrity	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required before a real shop trial.	Required before any external pilot.	Not a bolt-on
Financial reporting, store analysis, BI, and management dashboards	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for daily shop management and confidence.	Required for every store; advanced analytics can become add-on scope later.	Not a bolt-on
Inventory, stock movements,	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs	Required for Clapham Cycle	Required for every store.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
stocktake, transfers, replenishment		Hardening	stock confidence.		
POS, till, in-store payments, receipts, quotations, layaways	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for Clapham Cycle soft go-live.	Required for every store.	Not a bolt-on
Procurement, purchase orders, receiving, supplier returns	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for core stock replenishment and supplier workflow.	Required for every bike shop.	Not a bolt-on
Refunds, exchanges, gift cards, liabilities, and stored value	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required for controlled shop operation, especially liability review.	Required for wider rollout trust.	Not a bolt-on
Roles, permissions, login, PIN auth, sensitive-action controls	CORE_FUNDAMENTAL_NEEDS_HARDENING	Foundation Exists	Required for safe staff operation.	Required for rollout and support.	Not a bolt-on
Rota, staff scheduling, closed days, working hours	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required where booking capacity depends on working patterns.	Required for rollout; advanced HR/payroll remains out of scope.	Not a bolt-on
Staff performance reporting	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Useful for launch visibility and manager coaching; attribution review, drilldowns, product classification, and cost-snapshot foundations now exist, but formal pay/incentive decisions still need policy sign-off.	Commercial management feature with bike-shop-specific retail, workshop, line attribution, review workflow, scorecards, and add-on reporting now maturing.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Store Info, opening hours, public business presence, basic Google Business Profile foundation	CORE_FUNDAMENTAL_NEEDS_HARDENING	Foundation Exists	Core public-presence foundation; advanced review/marketing automation is not launch scope.	Commercially important local-presence trust layer.	Not a bolt-on
Supplier integration foundation and Madison baseline path	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Core as a capability, but live automation is not required for day-one use.	Commercial differentiator and paid-connector foundation.	Additional supplier connectors can become paid add-ons.
System configuration, first-run onboarding, dashboard settings	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required to make the trial reproducible.	Required for external onboarding later.	Not a bolt-on
Workshop diary, scheduler, capacity, and online workshop booking	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required as a core workshop scheduling capability; embed packaging is not launch-blocking.	Required for commercial trust and customer-facing maturity.	Not a bolt-on
Workshop jobs, estimates, approvals, parts, collection, print	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required for Clapham Cycle.	Required for every bike-shop rollout.	Not a bolt-on
Release verification, merge readiness, PR process, dashboard publishing	CORE_FUNDAMENTAL	Trial Ready	Required for release safety, though not a shop-floor feature.	Required for maintainable product operations.	Not a bolt-on
Staff records and staff directory	CORE_FUNDAMENTAL	Foundation Exists	Required for staff access and operational records.	Required for every store.	Not a bolt-on
HQ, multi-location, location governance	COMMERCIAL_HARDENING	Foundation Exists	Not required for single-store Clapham Cycle go-live beyond safe defaults.	Important for later multi-store rollout.	Not a bolt-on
Persistent customer account, profile/preferences, workshop portal unification, rental readback	COMMERCIAL_HARDENING	Built, Needs Hardening	Helpful for customer service, but not a blocker for core in-shop operation.	Important for wider customer-facing product maturity.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Stripe online payments, public payment status, refunds, reconciliation	COMMERCIAL_HARDENING	Active	Supporting for Hardening	Required for commercial aonline/payment trust.	Not a bolt-on
Customer-facing storefront, website checkout, online orders	BOLT_ON	Foundation Exists	Not required for Clapham Cycle core go-live.	Important later commercial/customer-facing product area.	Bolt-on / commerce package candidate.
HR personnel foundation and future payroll	BOLT_ON	Foundation Exists	Outside Clapham Cycle core go-live.	Optional HR bolt-on foundation now includes personnel files, review queue, candidate offers, lifecycle event snapshots, regenerated-link controls, withdrawal evidence, new-starter capture, and internal contract snapshot/signing evidence; payroll remains future specialist work.	Paid add-on candidate.
Quote Approval public widget	BOLT_ON	Foundation Exists	Widget packaging is not counted toward Clapham Cycle core until Thomas explicitly adds it.	Merged customer-facing widget maturity for wider rollout.	Bolt-on / paid add-on candidate.
Rental / hire internal platform for bikes and bike boxes	BOLT_ON	Built, Needs Hardening	Explicitly not required for Clapham Cycle core go-live unless Thomas adds it to launch scope.	Bike-shop differentiator and bolt-on candidate.	Bolt-on / paid add-on candidate.
Rental / hire public request, status, widget flow, account and secure-link readback	BOLT_ON	Foundation Exists	Not required for Clapham Cycle core go-live.	Useful customer-facing bolt-on.	Bolt-on / paid add-on candidate.
Rental / hire staff booking	BOLT_ON	Foundation Exists	Not counted for Clapham Cycle	Merged bolt-on hardening for rental	Bolt-on / paid add-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
amendments			core readiness because rental is bolt-on scope.	operations.	candidate.
Website builder, richer public widgets, template publishing	BOLT_ON	Foundation Exists	Commercial Builder packaging remains outside generic Clapham core scoring; the Clapham-specific generated website replacement is tracked separately as go-live scope.	WebCommercial add-on / packaging area.	Bolt-on / paid add-on candidate.
Additional supplier connectors beyond Madison baseline	PAID_ADD_ON	Planning	Outside Clapham Cycle core go-live.	Paid supplier connector expansion after Madison/general sync foundations are proven.	Paid add-on candidate.
Advanced analytics beyond core financial/store analysis	PAID_ADD_ON	Foundation Exists	Outside Clapham Cycle core go-live beyond basic reports.	Potential premium analytics add-on.	Paid add-on candidate.
Advanced Google Reviews, review requests, marketing automation, Maps/Places depth	PAID_ADD_ON	Planning	Outside Clapham Cycle core go-live.	Potential paid local-marketing add-on after public presence foundation is stable.	Paid add-on candidate.
Phone system / telephony integration	PAID_ADD_ON	Foundation Exists	Outside Clapham Cycle core go-live.	Optional paid add-on candidate.	Paid add-on candidate.
QuickBooks integration, mapping, posting, reconciliation controls	PAID_ADD_ON	Active Hardening	Not required for Clapham Cycle core go-live unless Thomas explicitly adds it to day-one scope.	Important paid add-on / accounting trust feature.	Paid add-on candidate.
External single-store pilot after Clapham stability	NOT_STARTED	Planning	Must not precede 3-6 months stable Clapham Cycle operation.	Next rollout stage after Clapham Cycle proves stable.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Promotions, vouchers, loyalty, advanced discounting	FUTURE_IDEA_OUT_OF_CURRENT_SCOPE	Planning	Outside Clapham Cycle core go-live unless Thomas explicitly adds it.	Future retail growth idea.	Future add-on possibility.

#### Detailed LOC breakdown

AREA	LOC	FILES
Backend	204,902	478
Frontend	166,855	298
Scripts	125,558	335
Print Agent	2,893	24
E2E	8,146	24
Prisma	13,612	183
<b>Tracked areas total</b>	<b>521,966</b>	<b>1,342</b>

Headline project LOC is 601,514. The named rows above cover the main implementation areas and may not include every other tracked file in the repo.

#### Full velocity details

- **Reference:** origin/main
- **7-day commits:** 315 commits
- **14-day commits:** 646 commits
- **30-day commits:** 802 commits
- **Approx PR-like subjects:** 309 signals
- **Merge commits detected:** 290 merge commits
- **Roadmap events detected:** 91 events
- **Changed lines:** +397,739 / -25,627 (net 372,112)

## Estimate caveats and readiness windows

### **Clapham Cycle soft go-live**

On track for 2026-08-01. Estimated core gap closure: 4 weeks to 6 weeks.

### **External single-store pilot**

Only after stable Clapham Cycle operation: 2026-11-01 to 2027-02-01 from target, if operation is stable.

### **Long-term platform completion**

4 weeks to 6 weeks

### **Broader commercial rollout**

5 weeks to 1.8 months

Single-shop production readiness is shown above as a current maturity score. The current heuristic time model is still more defensible for own-shop trial use, broader rollout, and longer-term platform completion than for a separate single-shop production delivery date.

- Estimates are heuristic and based on current maturity gaps plus recent local git cadence, not guaranteed delivery dates.
- PR/integration counts are approximated from local git subjects and may still miss some private or unusually titled workflows.
- Open PRs are reported as in-progress and are not counted as merged readiness until they land on main.
- Clapham Cycle core completion excludes bolt-ons, paid add-ons, future ideas, and open PRs unless Thomas explicitly moves them into launch scope.
- LOC is a size/complexity signal, not a quality metric.
- This dashboard is strongest where repo evidence is explicit and weaker where maturity still depends on operator practice or rollout trust.
- The first version does not yet use historical feature-map snapshots, so time estimates rely on current-state heuristics rather than measured maturity deltas.
- Headline LOC follows the cloc baseline only when cloc is available on the local machine; otherwise it falls back to a clearly labelled approximation.
- Scope classification is a governance layer: new ideas do not change Clapham Cycle core completion unless their classification and weights explicitly put them into core launch scope.

## Full feature maturity table

FEATURE	STATUS	CATEGORY	MATURITY
Verification / multi-agent tooling	Trial Ready	Platform / Commercial	84%
Customer account / portal	Built, Needs Hardening	Customer Facing	80%
Supplier / Madison integration	Active Hardening	Operational / Staff Workflow	80%
Procurement and receiving maturity	Active Hardening	Operational / Staff Workflow	79%
Track My Bike / Workshop Progress View	Built, Needs Hardening	Customer Facing	77%
POS / till / retail workflow	Active Hardening	Operational / Staff Workflow	76%
Stripe/payment trust	Active Hardening	Platform / Commercial	76%
QuickBooks/accounting trust	Active Hardening	Platform / Commercial	74%
Communication Centre: email, SMS, WhatsApp	Built, Needs Hardening	Customer Facing	73%
Deployment / Windows / Caddy / production serving	Active Hardening	Operational / Staff Workflow	72%
Diagnostics, support packs, and operational readiness	Built, Needs Hardening	Operational / Staff Workflow	72%
Customer bike lifecycle CRM	Built, Needs Hardening	Bike-shop-specific	72%
Stocktake and inventory accuracy	Built, Needs Hardening	Operational / Staff Workflow	70%
Workshop UX and behaviour audit	Built, Needs Hardening	Operational / Staff Workflow	66%
Permissions and sensitive action hardening	Foundation Exists	Operational / Staff Workflow	61%
Web builder / templates / embeddable widgets	Foundation Exists	Customer Facing	93%
Rental / bike box hire internal platform	Built, Needs Hardening	Bike-shop-specific	85%
Online workshop booking	Active Hardening	Customer Facing	76%
Service reminders	Built, Needs Hardening	Bike-shop-specific	75%
Customer-facing storefront	Foundation Exists	Customer Facing	71%
Workshop service history and customer portal	Built, Needs Hardening	Bike-shop-specific	71%
Managed printing / labels / print agent	Built, Needs Hardening	Platform / Commercial	66%
Google Business Profile / Reviews / Maps	Foundation Exists	Customer Facing	63%
Rental / hire customer-facing flow	Foundation Exists	Customer Facing	62%
Multi-location / HQ	Foundation Exists	Platform / Commercial	54%
HR personnel foundation add-on	Foundation Exists	Platform / Commercial	53%
Support agent / staff help system	Foundation Exists	Platform / Commercial	46%

Use the Feature Map tab for the category-by-category evidence, gaps, and next-task detail.

#### Raw limitations and notes

- This dashboard is strongest where repo evidence is explicit and weaker where maturity still depends on operator practice or rollout trust.
- The first version does not yet use historical feature-map snapshots, so time estimates rely on current-state heuristics rather than measured maturity deltas.
- Headline LOC follows the cloc baseline only when cloc is available on the local machine; otherwise it falls back to a clearly labelled approximation.
- Scope classification is a governance layer: new ideas do not change Clapham Cycle core completion unless their classification and weights explicitly put them into core launch scope.

COREPOS LIVING FEATURE MAP

**Feature-by-feature maturity in one place**

Use this tab for the deeper evidence-backed view: scope classification, category roll-ups, top gaps, ideas, next tasks, and the collapsible feature detail underneath.

Clapham Cycle core

**72%**

Narrow soft go-live scope only.

Full product/commercial

**63%**

Includes wider rollout and add-on maturity.

Bolt-on maturity

**50%**

Optional/premium packages.

Future ideas

**1 item**

Outside current scope.

## Scope classification inventory

Clapham Cycle core completion only moves when scope is explicitly classed as Clapham Cycle core. Bolt-ons and future ideas must not silently lower the core go-live score.

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Clapham Cycle launch rehearsal, live data setup, and acceptance sign-off	LAUNCH_BLOCKING_GAP	Planning	Launch-blocking governance gap for the 2026-08-01 soft go-live target.	Reusable launch process for external pilots later.	Not a bolt-on
Clapham Cycle Web Builder site replacement	LAUNCH_BLOCKING_GAP	Built, Needs Hardening	Explicitly in Clapham Cycle go-live scope: the CorePOS-generated website is intended to replace the existing Clapham Cycle website after controlled readiness and migration work.	Provides the first real proof case for the later commercial Web Builder add-on.	Not a generic bolt-on for Clapham Cycle go-live; broader Web Builder packaging for future stores remains a bolt-on/commercial add-on.
Cash-up, daily close, trade close, cash oversight	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for end-of-day operation.	Required for every store.	Not a bolt-on
CRM, customer management, customer bikes, and service context	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required for Clapham Cycle retail and workshop continuity.	Required for bike-shop differentiation.	Not a bolt-on
Customer communication basics and account-visible communication history	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required at basic level for workshop and customer follow-up.	Required for customer-service maturity.	Not a bolt-on
Deployment, diagnostics, support packs, backups, integrity	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required before a real shop trial.	Required before any external pilot.	Not a bolt-on
Financial reporting, store analysis, BI, and management dashboards	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for daily shop management and confidence.	Required for every store; advanced analytics can become add-on scope later.	Not a bolt-on
Inventory, stock movements, stocktake,	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required for Clapham Cycle	Required for every store.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
transfers, replenishment			stock confidence.		
POS, till, in-store payments, receipts, quotations, layaways	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for Clapham Cycle soft go-live.	Required for every store.	Not a bolt-on
Procurement, purchase orders, receiving, supplier returns	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required for core stock replenishment and supplier workflow.	Required for every bike shop.	Not a bolt-on
Refunds, exchanges, gift cards, liabilities, and stored value	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required for controlled shop operation, especially liability review.	Required for wider rollout trust.	Not a bolt-on
Roles, permissions, login, PIN auth, sensitive-action controls	CORE_FUNDAMENTAL_NEEDS_HARDENING	Foundation Exists	Required for safe staff operation.	Required for rollout and support.	Not a bolt-on
Rota, staff scheduling, closed days, working hours	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required where booking capacity depends on working patterns.	Required for rollout; advanced HR/payroll remains out of scope.	Not a bolt-on
Staff performance reporting	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Useful for launch visibility and manager coaching; attribution review, drilldowns, product classification, and cost-snapshot foundations now exist, but formal pay/incentive decisions still need policy sign-off.	Commercial management feature with bike-shop-specific retail, workshop, line attribution, review workflow, scorecards, and add-on reporting now maturing.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Store Info, opening hours, public business presence, basic Google Business Profile foundation	CORE_FUNDAMENTAL_NEEDS_HARDENING	Foundation Exists	Core public-presence foundation; advanced review/marketing automation is not launch scope.	Commercially important local-presence trust layer.	Not a bolt-on
Supplier integration foundation and Madison baseline path	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Core as a capability, but live automation is not required for day-one use.	Commercial differentiator and paid-supplier connector foundation.	Additional connectors can become paid add-ons.
System configuration, first-run onboarding, dashboard settings	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required to make the trial reproducible.	Required for external onboarding later.	Not a bolt-on
Workshop diary, scheduler, capacity, and online workshop booking	CORE_FUNDAMENTAL_NEEDS_HARDENING	Active Hardening	Required as a core workshop scheduling capability; embed packaging is not launch-blocking.	Required for commercial trust and customer-facing maturity.	Not a bolt-on
Workshop jobs, estimates, approvals, parts, collection, print	CORE_FUNDAMENTAL_NEEDS_HARDENING	Built, Needs Hardening	Required for Clapham Cycle.	Required for every bike-shop rollout.	Not a bolt-on
Release verification, merge readiness, PR process, dashboard publishing	CORE_FUNDAMENTAL	Trial Ready	Required for release safety, though not a shop-floor feature.	Required for maintainable product operations.	Not a bolt-on
Staff records and staff directory	CORE_FUNDAMENTAL	Foundation Exists	Required for staff access and operational records.	Required for every store.	Not a bolt-on
HQ, multi-location, location governance	COMMERCIAL_HARDENING	Foundation Exists	Not required for single-store Clapham Cycle go-live beyond safe defaults.	Important for later multi-store rollout.	Not a bolt-on
Persistent customer account, profile/preferences, workshop portal unification, rental readback	COMMERCIAL_HARDENING	Built, Needs Hardening	Helpful for customer service, but not a blocker for core in-shop operation.	Important for wider customer-facing product maturity.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Stripe online payments, public payment status, refunds, reconciliation	COMMERCIAL_HARDENING	Active	Supporting for Hardening	Required for commercial aonline/payment trust.	Not a bolt-on
Customer-facing storefront, website checkout, online orders	BOLT_ON	Foundation Exists	Not required for Clapham Cycle core go-live.	Important later commercial/customer-facing product area.	Bolt-on / commerce package candidate.
HR personnel foundation and future payroll	BOLT_ON	Foundation Exists	Outside Clapham Cycle core go-live.	Optional HR bolt-on foundation now includes personnel files, review queue, candidate offers, lifecycle event snapshots, regenerated-link controls, withdrawal evidence, new-starter capture, and internal contract snapshot/signing evidence; payroll remains future specialist work.	Paid add-on candidate.
Quote Approval public widget	BOLT_ON	Foundation Exists	Widget packaging is not counted toward Clapham Cycle core until Thomas explicitly adds it.	Merged customer-facing widget maturity for wider rollout.	Bolt-on / paid add-on candidate.
Rental / hire internal platform for bikes and bike boxes	BOLT_ON	Built, Needs Hardening	Explicitly not required for Clapham Cycle core go-live unless Thomas adds it to launch scope.	Bike-shop differentiator and bolt-on candidate.	Bolt-on / paid add-on candidate.
Rental / hire public request, status, widget flow, account and secure-link readback	BOLT_ON	Foundation Exists	Not required for Clapham Cycle core go-live.	Useful customer-facing bolt-on.	Bolt-on / paid add-on candidate.
Rental / hire staff booking	BOLT_ON	Foundation Exists	Not counted for Clapham Cycle	Merged bolt-on hardening for rental	Bolt-on / paid add-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
amendments			core readiness because rental is bolt-on scope.	operations.	candidate.
Website builder, richer public widgets, template publishing	BOLT_ON	Foundation Exists	Commercial Builder packaging remains outside generic Clapham core scoring; the Clapham-specific generated website replacement is tracked separately as go-live scope.	Commercial add-on / packaging area.	Bolt-on / paid add-on candidate.
Additional supplier connectors beyond Madison baseline	PAID_ADD_ON	Planning	Outside Clapham Cycle core go-live.	Paid supplier connector expansion after Madison/general sync foundations are proven.	Paid add-on candidate.
Advanced analytics beyond core financial/store analysis	PAID_ADD_ON	Foundation Exists	Outside Clapham Cycle core go-live beyond basic reports.	Potential premium analytics add-on.	Paid add-on candidate.
Advanced Google Reviews, review requests, marketing automation, Maps/Places depth	PAID_ADD_ON	Planning	Outside Clapham Cycle core go-live.	Potential paid local-marketing add-on after public presence foundation is stable.	Paid add-on candidate.
Phone system / telephony integration	PAID_ADD_ON	Foundation Exists	Outside Clapham Cycle core go-live.	Optional paid add-on candidate.	Paid add-on candidate.
QuickBooks integration, mapping, posting, reconciliation controls	PAID_ADD_ON	Active Hardening	Not required for Clapham Cycle core go-live unless Thomas explicitly adds it to day-one scope.	Important paid add-on / accounting trust feature.	Paid add-on candidate.
External single-store pilot after Clapham stability	NOT_STARTED	Planning	Must not precede 3-6 months stable Clapham Cycle operation.	Next rollout stage after Clapham Cycle proves stable.	Not a bolt-on

MODULE	SCOPE	STATUS	CLAPHAM CYCLE	COMMERCIAL	BOLT-ON
Promotions, vouchers, loyalty, advanced discounting	FUTURE_IDEA_OUT_OF_CURRENT_SCOPE	Planning	Outside Clapham Cycle core go-live unless Thomas explicitly adds it.	Future retail growth idea.	Future add-on possibility.

#### CATEGORY OVERVIEW

### Feature areas

CATEGORY	MATURITY	TRIAL	COUNT
Operational / Staff Workflow	72%	71%	8
Customer Facing	75%	66%	8
Bike-shop-specific	76%	69%	4
Platform / Commercial	66%	62%	7

#### TOP GAPS

### Most important unfinished work

- **Workshop UX and behaviour audit:** A single workshop readiness audit is not yet codified, so UX debt is spread across several surfaces. high
- **Procurement and receiving maturity:** Supplier API ordering, supplier returns/RMA, supplier documents, and procurement accountability still need more operational polish. high
- **Supplier / Madison integration:** Live sync, supplier API ordering, and normalization maturity are still intentionally incomplete. high
- **Permissions and sensitive action hardening:** Sensitive action review is still distributed across many modules rather than a single hardening ledger. high
- **Diagnostics, support packs, and operational readiness:** Support-safe troubleshooting flows still need tighter packaging for real trial use. high
- **Track My Bike / Workshop Progress View:** Portal polish, customer wording, and broader regression coverage still need hardening before it feels fully trial-safe. high
- **Communication Centre: email, SMS, WhatsApp:** Channel reliability, support tooling, and operator trust signals still need more maturity than the raw transport foundation. high
- **Customer account / portal:** Confirmed rental booking readback and storefront order-history readback are account-only and view-only; rental self-service, public confirmation, secure readback tokens, customer-visible receipt/invoice controls, and broader longitudinal account history remain future work. high

## NEWEST IDEAS

### Fresh thinking worth keeping

- **Communication Centre: email, SMS, WhatsApp:** Track communication maturity by channel trust, orchestration depth, and customer self-service coverage.
- **Customer account / portal:** Split portal maturity into identity, self-service, and ongoing customer relationship stages.
- **Customer bike lifecycle CRM:** Create a bike-lifecycle maturity band that combines reminders, history, portal access, and workshop continuity.
- **Customer-facing storefront:** Track storefront maturity by browse/catalogue, checkout, fulfilment, and post-purchase support.
- **Deployment / Windows / Caddy / production serving:** Treat deployment readiness as its own recurring review lane rather than burying it inside docs.
- **Diagnostics, support packs, and operational readiness:** Track support-bundle completeness as a first-class maturity signal beside implementation.
- **Google Business Profile / Reviews / Maps:** Add a dedicated Google presence maturity lane only after controlled posting has operational evidence or public local-marketing value becomes real.
- **HR personnel foundation add-on:** Keep Problem Cycle-specific bonus calculation separate from the reusable commercial HR module unless a configurable framework is deliberately commissioned.

## RECOMMENDED NEXT TASKS

### Immediate follow-through

- **Workshop UX and behaviour audit:** Define a workshop hardening checklist covering queue, job detail, check-in, bookings, collection, and portal handoff. **high**
- **Workshop UX and behaviour audit:** Add a compact workshop regression bundle for the highest-risk operator flows. **high**
- **Procurement and receiving maturity:** Turn the procurement gap audit into a living maturity checklist with explicit completion criteria. **high**
- **Procurement and receiving maturity:** Add targeted regression coverage for discrepancy and exception-resolution paths. **high**
- **Supplier / Madison integration:** Run supervised Madison/procurement pilot sessions and use readiness decisions, the operating dashboard, and bounded evidence reports to decide whether narrow manager-led limited use remains justified. **high**
- **Permissions and sensitive action hardening:** Map the highest-risk manager/admin actions into an explicit permissions hardening checklist. **high**
- **Diagnostics, support packs, and operational readiness:** Keep the dashboard's trial-readiness section concise as diagnostics, integrity, and support-pack maturity continue to grow. **high**
- **Track My Bike / Workshop Progress View:** Add a focused portal hardening checklist covering progress wording, approvals, attachments, and reply flows. **high**
- **Communication Centre: email, SMS, WhatsApp:** Translate post-service follow-up consent, suppression, review-request wording, Google separation, and approval policy into explicit product-data requirements before any outbound workflow is considered. **high**
- **Communication Centre: email, SMS, WhatsApp:** Clarify the maturity ladder for workshop notifications versus broader omnichannel communication. **high**

FIND FEATURES FAST

## Filter the feature map

### Evidence warnings (1)

- Missing evidence file for rental-bike-box-hire-internal-platform:  
prisma/migrations/20260613120000\_rental\_hire\_pilot\_readiness\_closeout/migration.sql

## Operational / Staff Workflow

72% average maturity across 8 features.

71% trial

8 features

## Supplier / Madison integration

The repo has real supplier-integration groundwork through reviewed Madison draft PO creation, supplier-generated draft review hardening, manual supplier submission recording, merged manager issue intake/resolution, pilot-session tracking, readiness decision records, a limited-use operating dashboard, and bounded evidence reports for management review. Current closeout says this is supervised-testing ready and conditionally suitable for narrow manager-led limited operational use, but it is intentionally conservative and not yet a live sync or supplier API ordering system.

High

Active Hardening

80%

**Target:** operational, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against supplier integration routes, services, docs, closeout, and smoke coverage.

### Feature maturity

80%

#### Evidence links (19)

- `frontend/src/pages/SupplierIntegrationReadinessPage.tsx` — Supplier integration readiness UI exists.
- `frontend/src/pages/SupplierIntegrationPilotSessionsPage.tsx` — Supplier integration pilot-session UI exists for supervised Madison/procurement trial measurement and final recommendation.
- `frontend/src/pages/SupplierIntegrationIssuesPage.tsx` — Supplier integration issue intake/resolution UI exists for Madison/procurement trial findings.
- `frontend/src/pages/SupplierIntegrationReadinessDecisionPage.tsx` — Supplier integration readiness decision UI distinguishes computed recommendation, manager final decision, blockers, evidence, and history.
- `frontend/src/pages/SupplierIntegrationLimitedUseDashboardPage.tsx` — Supplier integration limited-use dashboard aggregates provider state, readiness decision, blockers, review due, recent evidence, safe activity, and bounded management evidence reports.
- `frontend/src/pages/PurchaseOrderPage.tsx` — Supplier-generated draft PO review evidence and sign-off UI exists.
- `src/routes/supplierIntegrationRoutes.ts` — Supplier integration routes exist.
- `src/services/supplierIntegrationReadinessDecisionService.ts` — Readiness decision service computes conservative recommendations from pilots, unresolved issues, blockers, preview/link evidence, and preserves manager overrides.
- `src/services/supplierIntegrationLimitedUseEvidenceReportService.ts` — Limited-use evidence report service generates bounded sanitized report/Markdown summaries and records report generated/copied/exported audit events without raw payloads.
- `src/services/supplierIntegrationPilotService.ts` — Pilot service validates linked supplier/procurement evidence, computes metrics, records lifecycle events, and keeps mutation boundaries intact.
- `src/services/supplierIntegrationIssueService.ts` — Supplier integration issue service validates linked records, triage/fix-plan transitions, and verification evidence before verified resolution.
- `scripts/supplier_integration_issue_intake_smoke_tests.js` — Issue intake/resolution smoke coverage verifies access control, linking, triage, audit, closure/reopen, and no-mutation boundaries.
- `scripts/supplier_integration_pilot_tracking_smoke_tests.js` — Pilot tracking smoke coverage verifies lifecycle, evidence linking, blockers, audit, and no-mutation boundaries.
- `scripts/supplier_integration_readiness_decision_smoke_tests.js` — Readiness decision smoke coverage verifies recommendations, overrides, linked evidence, audit events, permissions, history, and no-mutation boundaries.

- scripts/supplier\_integration\_limited\_use\_dashboard\_smoke\_tests.js — Limited-use dashboard smoke coverage verifies permission checks, blockers, recent evidence, sanitized output, and read-only boundaries.
- scripts/supplier\_integration\_limited\_use\_evidence\_report\_smoke\_tests.js — Limited-use evidence report smoke coverage verifies permissions, bounded filters, report contents, sanitization, audit events, and no-mutation boundaries.
- src/services/supplierIntegrations/madisonProvider.ts — Madison provider groundwork exists.
- docs/supplier\_integrations.md — Integration scope and limits are documented.
- docs/supplier\_integration\_limited\_use\_closeout.md — Closeout states that wider shop use and unattended automation are still blocked despite the merged limited-use path.

#### Remaining gaps (2)

- Live sync, supplier API ordering, and normalization maturity are still intentionally incomplete. **high**
- Operational confidence now has pilot telemetry, readiness decisions, an operating dashboard, and evidence reports, but readiness still depends on repeated supervised trial evidence before any automation discussion.

**medium**

#### Recommended next tasks (2)

- Run supervised Madison/procurement pilot sessions and use readiness decisions, the operating dashboard, and bounded evidence reports to decide whether narrow manager-led limited use remains justified. **high**
- Use operating dashboard findings to prioritize unresolved Madison questions, stale preview/link evidence, and receiving discrepancy follow-up. **medium**

#### Ideas and dependencies

##### Ideas

- Track supplier integration maturity separately for preview tooling, approval tooling, and live automation.

##### Dependencies

- Procurement and receiving maturity
- Permissions hardening

## Procurement and receiving maturity

Purchasing and receiving are implemented enough to use, reviewed Madison suggestions can create draft POs, supplier-generated drafts now have review hardening, reviewed drafts can be manually preflighted/exported/recorded, confirmed receipt discrepancies now have internal manager resolution history, and managers now have merged supplier integration issue resolution, pilot-session tracking, readiness decisions, and a limited-use operating dashboard with evidence, blockers, confidence, next review, and recent activity; supplier API ordering and procurement exception maturity remain hardening work.

High

Active Hardening

79%

**Target:** operational, trial, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit plus procurement readiness and limited-use closeout docs already committed on main.

### Feature maturity

79%

#### Evidence links (11)

- frontend/src/pages/PurchasingPage.tsx — Purchasing workspace exists.
- frontend/src/pages/SupplierReceivingPage.tsx — Receiving workspace exists.
- frontend/src/pages/SupplierIntegrationPilotSessionsPage.tsx — Manager/admin supplier integration pilot sessions track supervised trial evidence, metrics, blockers, confidence, and recommendation.
- frontend/src/pages/SupplierIntegrationIssuesPage.tsx — Manager/admin supplier integration issue intake and resolution dashboard exists.
- frontend/src/pages/SupplierIntegrationReadinessDecisionPage.tsx — Manager/admin supplier integration readiness decision gate records conservative go/no-go review.
- frontend/src/pages/SupplierIntegrationLimitedUseDashboardPage.tsx — Manager/admin limited-use operating dashboard shows current readiness, blockers, review due, and recent Madison/procurement evidence.
- src/routes/purchaseOrderRoutes.ts — Purchase-order routes exist.
- prisma/migrations/20260615120000\_procurement\_receiving\_discrepancy\_resolution/migration.sql — Receipt discrepancy resolution model records manager review history without creating supplier credits, replacements, invoices, or accounting entries.
- scripts/procurement\_receiving\_discrepancy\_smoke\_tests.js — Procurement receiving smoke covers discrepancy resolution permissions, validation, audit, and no stock mutation.
- docs/procurement\_readiness\_gap\_audit.md — Gap audit already documents remaining maturity work.
- docs/supplier\_integration\_limited\_use\_closeout.md — Closeout confirms the current path is suitable for supervised testing and only narrow manager-led limited operational use.

#### Remaining gaps (2)

- Supplier API ordering, supplier returns/RMA, supplier documents, and procurement accountability still need more operational polish. high
- Manager trust signals now have merged pilot sessions, issue queues, readiness decisions, and a limited-use operating dashboard, but broader procurement exception resolution remains split across specialist workflows.

medium

#### Recommended next tasks (2)

- Turn the procurement gap audit into a living maturity checklist with explicit completion criteria. **high**
- Add targeted regression coverage for discrepancy and exception-resolution paths. **high**

#### Ideas and dependencies

##### Ideas

- Show procurement maturity as its own dashboard slice with explicit remaining exception classes.
- Use the supplier operating dashboard evidence as an input to future procurement exception reporting.

##### Dependencies

- Supplier / Madison integration
- Diagnostics and operational readiness

## POS / till / retail workflow

Retail checkout is a strong foundation and one of the most operationally credible areas in the repo.

High

Active Hardening

76%

**Target:** operational, trial, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against POS routes, pages, and smoke coverage.

### Feature maturity

76%

#### Evidence links (4)

- frontend/src/pages/PosPage.tsx — React POS surface exists.
- src/routes/tillRoutes.ts — Till APIs are present.
- scripts/pos\_m28\_smoke\_tests.js — POS smoke coverage exists.
- scripts/till\_m37\_smoke\_tests.js — Till workflow smoke coverage exists.

#### Remaining gaps (2)

- Trial-facing runbooks and operator guidance are thinner than the implementation depth. medium
- Broader end-to-end coverage for mixed tender and exception cases can still improve confidence. medium

#### Recommended next tasks (2)

- Expand retail regression coverage around mixed tenders, refunds, receipts, and edge-case operator states. medium
- Add a POS/trade-close readiness checklist to the support docs set. medium

#### Ideas and dependencies

##### Ideas

- Track recurring POS trial friction separately from feature completion so operator confidence has its own signal.

##### Dependencies

- Permissions hardening
- Diagnostics and support readiness

## Deployment / Windows / Caddy / production serving

Deployment guidance and production frontend serving exist, but this is still operationally sensitive enough to keep under active hardening.

High

Active Hardening

72%

**Target:** operational, trial, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against deployment docs and serving verification scripts.

### Feature maturity

72%

#### Evidence links (4)

- docs/production\_setup.md — Production setup guide exists.
- docs/windows\_print\_agent.md — Windows print-agent guidance exists.
- docs/production\_frontend\_serving.md — Frontend serving docs exist.
- scripts/verify\_production\_frontend\_serving.js — Production serving verification script exists.

#### Remaining gaps (2)

- Deployment confidence still depends on disciplined manual process rather than a single hardened release dashboard. medium
- Windows/Caddy/operator troubleshooting maturity is documented but not yet deeply operationalized in the new dashboard layer. medium

#### Recommended next tasks (2)

- Expose deployment readiness warnings and evidence paths directly in the dashboard output. medium
- Add a concise trial-store release checklist pointer from the dashboard summary. medium

#### Ideas and dependencies

##### Ideas

- Treat deployment readiness as its own recurring review lane rather than burying it inside docs.

##### Dependencies

- Managed printing / print agent
- Verification / multi-agent tooling

## Diagnostics, support packs, and operational readiness

Diagnostics, integrity, and support assets are strong enough to count as implemented, but still need better operational packaging.

High

Built, Needs Hardening

72%

Target: operational, trial

Last reviewed: 2026-05-15. Source: Repo audit against diagnostics pages, support docs, and smoke coverage.

### Feature maturity

72%

#### Evidence links (4)

- frontend/src/pages/AdminDiagnosticsPage.tsx — Admin diagnostics page exists.
- frontend/src/pages/OpsHealthPage.tsx — Ops health page exists.
- docs/pilot\_support\_pack.md — Trial support pack doc exists.
- scripts/admin\_diagnostics\_smoke\_tests.js — Diagnostics smoke coverage exists.

#### Remaining gaps (2)

- Support-safe troubleshooting flows still need tighter packaging for real trial use. high
- The new compact dashboard is more shareable, but support-safe review habits and trend history are still not routine enough yet. medium

#### Recommended next tasks (2)

- Keep the dashboard's trial-readiness section concise as diagnostics, integrity, and support-pack maturity continue to grow. high
- Add a documented checklist for what should be reviewed before asking a trial shop to self-diagnose. medium

#### Ideas and dependencies

##### Ideas

- Track support-bundle completeness as a first-class maturity signal beside implementation.

##### Dependencies

- Deployment / production serving
- Verification / multi-agent tooling

## Stocktake and inventory accuracy

Stocktake is clearly implemented and should be tracked as a hardening problem, not a missing feature.

High

Built, Needs Hardening

70%

Target: operational, trial

Last reviewed: 2026-05-15. Source: Repo audit against stocktake pages, routes, services, and smoke tests.

### Feature maturity

70%

#### Evidence links (4)

- frontend/src/pages/InventoryStocktakesPage.tsx — Dedicated stocktake session UI exists.
- src/routes/stocktakeSessionRoutes.ts — Session-style stocktake routes exist.
- scripts/stocktake\_m18\_smoke\_tests.js — Legacy stocktake coverage exists.
- scripts/stocktake\_m26\_smoke\_tests.js — Session/finalize stocktake coverage exists.

#### Remaining gaps (2)

- Stocktake UX still needs a cleaner operator narrative for review, finalize, and exception handling. medium
- Inventory accuracy diagnostics are present, but stocktake-specific readiness reporting is still thin. medium

#### Recommended next tasks (2)

- Add stocktake operational guidance to the support docs with review/finalize troubleshooting. medium
- Track variance-resolution quality separately from the raw ability to run a stocktake session. medium

#### Ideas and dependencies

##### Ideas

- Surface stocktake confidence trends and repeat variance hotspots on the dashboard layer.

##### Dependencies

- Diagnostics and support-pack readiness
- Procurement and receiving maturity

## Workshop UX and behaviour audit

Workshop operations are broad and visibly implemented, but workflow friction and coverage still need deliberate hardening before trial use feels calm.

High

Built, Needs Hardening

66%

Target: operational, trial

Last reviewed: 2026-05-15. Source: Repo audit against current workshop pages, services, and E2E coverage.

### Feature maturity

66%

#### Evidence links (4)

- frontend/src/pages/WorkshopPage.tsx — Main workshop workspace exists.
- frontend/src/pages/WorkshopJobPage.tsx — Detailed job workflow, timeline, attachments, and conversation UI exists.
- frontend/src/pages/WorkshopCheckInPage.tsx — Structured intake/check-in flow exists.
- e2e/workshop/workshop-scheduler.spec.js — Workshop scheduling has dedicated end-to-end coverage.

#### Remaining gaps (2)

- A single workshop readiness audit is not yet codified, so UX debt is spread across several surfaces. high
- Automation depth is still lighter than the breadth of the workshop UI surface. medium

#### Recommended next tasks (2)

- Define a workshop hardening checklist covering queue, job detail, check-in, bookings, collection, and portal handoff. high
- Add a compact workshop regression bundle for the highest-risk operator flows. high

#### Ideas and dependencies

##### Ideas

- Add a release-facing workshop scorecard that tracks friction hotspots by sub-flow.

##### Dependencies

- Diagnostics and support-pack maturity
- Customer portal hardening

## Permissions and sensitive action hardening

Role-aware infrastructure is broad, but repo-owned tracking should keep calling out the difference between route guards and true operational safety.

High

Foundation Exists

61%

**Target:** operational, trial, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against auth middleware, routes, UI, and smoke coverage.

### Feature maturity

61%

#### Evidence links (4)

- `src/middleware/staffRole.ts` — Shared role guard middleware exists.
- `src/routes/authRoutes.ts` — Authentication surface exists.
- `frontend/src/pages/PinSettingsPage.tsx` — PIN self-service/admin UI exists.
- `scripts/pin_auth_foundation_smoke_tests.js` — PIN auth smoke coverage exists.

#### Remaining gaps (2)

- Sensitive action review is still distributed across many modules rather than a single hardening ledger. **high**
- Staff-facing wording and escalation guidance for denied/guarded actions can still be improved. **medium**

#### Recommended next tasks (2)

- Map the highest-risk manager/admin actions into an explicit permissions hardening checklist. **high**
- Add support-doc guidance for common permission failures and safe escalation. **medium**

#### Ideas and dependencies

##### Ideas

- Track sensitive action maturity separately from login/auth maturity so hardening work is visible.

##### Dependencies

- Diagnostics and operational readiness
- Support agent / help system

## Customer Facing

75% average maturity across 8 features.

66% trial

8 features

## Web builder / templates / embeddable widgets

This should be treated as partial only. Public site, Store Info wiring, styling settings, the workshop booking iframe, the Track My Bike secure-link iframe, the Quote Approval secure-link iframe, a read-only Store Info/opening-hours iframe, a request-first rental/hire iframe, a central Public Widgets hub, the Web Builder V1 data/admin shell, the Workshop-first template, the Retail + workshop template, the Premium service template, the first public catalogue contract, guarded generated-site purchasable product browsing, anonymous storefront basket intent, public preview/share links, and bounded template image blocks now exist. The shell has AppConfig-backed setup state, Store Info dependency visibility, widget reuse guidance, manager/admin preview routes, public-safe stakeholder share links, manager-controlled dedicated-route public publishing for available templates, a first-run checklist, support-safe generated-site diagnostics, constrained theme controls, and image-block controls for template-defined slots. Template 1 now has first content, section, SEO, metadata, accessibility, mobile/performance, support/onboarding, purchasable product cards, search/category filters, basket-token checkout handoff, compatibility SKU/quantity checkout fallback, and bounded imagery passes. Public checkout can carry generated-site basket-token or SKU/quantity intent while collecting customer contact and shipping details for server-side checkout validation. Task 35A now implements optional signed-in storefront checkout and basket checkout account attachment plus view-only account-owned web-order history, while preserving guest checkout and rejecting guest-order inference by email, phone, name, postcode, basket token, or order text. Task 36 now enriches paid storefront order payment-received emails with customer-safe receipt/invoice details, staff-visible send state, and explicit manager resend without changing payment/accounting truth. Task 37 records an EasyPost/aggregator-first courier decision while keeping direct Royal Mail, DPD, DHL, and other branded adapters behind pilot evidence gates. Task 38 adds server-owned UK shipping-rate/address validation and trusted shipping charges. Task 39 adds customer-safe collection/dispatch notifications from authoritative fulfilment/provider state with staff-visible send/review/failure state and no payment/accounting mutation. Task 41 is runtime-complete for explicit storefront product/variant publishing controls: active products stay hidden by default, `online-store/products` is the manager-owned public product control surface, the public catalogue contract includes `publicationAuthority`, browse-only products avoid checkout handoff, category/featured/sort ordering stays server-owned, and basket/direct checkout revalidates against publishing state. Task 42 is runtime-complete for bounded storefront product media: manager-owned upload/external-URL controls validate PNG/JPEG/WebP media, public catalogue v3 exposes only approved public-safe `primaryImage` data with alt text, and generated product cards render fixed-aspect image or text fallback states. Task 43 is runtime-complete for storefront merchandising diagnostics and launch readiness: Web Builder now reports public product availability, category readiness, featured-product exposure/suppression, product-media/fallback state, browse-only/checkout state, and Clapham pilot launch blockers without changing public catalogue or checkout authority. Template 2 now has a publishable retail/workshop body with product browsing, basket intent, and widget reuse on the same dedicated route and readiness gates. Template 3 now has a publishable premium-service body and uses the same manager-controlled readiness gates on the dedicated route. The repo now has a Web Builder delivery plan, task queue, product browsing integration audit, storefront merchandising/product-media/publish-controls plan, customer-account checkout/order-history contract, storefront courier-adapter decision audit, Clapham migration evidence records, Clapham route-inventory/decision-matrix evidence, Clapham route-map candidate/approval-packet/remaining-blocker evidence, Clapham migration verification/support planning, and a commercial/pilot readiness audit. Thomas approved Task 29 repo implementation on 2026-05-16, moving the Clapham generated-site replacement out of blocked planning and toward implementation/pilot readiness; DNS/custom-domain/production launch evidence remains pending. The commercial audit keeps the Clapham Cycle replacement as the proof case, Public Widgets as the widget-only option for stores with existing websites, and broader Web Builder packaging as a future commercial bolt-on. A general CMS/page builder, unrestricted media library, supplier media auto-import, billing/entitlement system, custom-domain automation, and generic external-store pilot process do not exist yet.

Medium

Foundation Exists

93%

**Target:** customer-facing, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against public-site pages, Store Info wiring, Web Builder Template 1, and styling settings.

### Feature maturity

93%

#### Evidence links (111)

- [frontend/src/pages/CustomerSitePage.tsx](#) — Public site scaffold exists.
- [frontend/src/pages/StylingSettingsPage.tsx](#) — Styling settings UI exists.
- [src/routes/publicStoreInfoRoutes.ts](#) — Public Store Info route exists.

- frontend/src/App.tsx — Public /widgets/workshop-booking, /widgets/track-my-bike, /widgets/quote-approval, /widgets/store-info, and /widgets/hire-request routes exist as iframe widget surfaces.
- frontend/src/pages/TrackMyBikeWidgetPage.tsx — Track My Bike widget renders a compact secure-link helper without public-site or staff/admin chrome.
- frontend/src/pages/QuoteApprovalWidgetPage.tsx — Quote Approval widget renders a compact secure-link helper without public-site or staff/admin chrome.
- frontend/src/pages/StoreInfoWidgetPage.tsx — Store Info / opening-hours widget renders public-safe shop contact, directions, hours, special hours, and closures without public-site chrome.
- frontend/src/pages/PublicHireRequestWidgetPage.tsx — Rental / hire request widget renders the existing public hire request intake flow in compact iframe-safe request-only mode.
- src/config/publicWidgetMetadata.ts — Public widget suite metadata now centralizes live widget ids, routes, titles, dimensions, customer boundaries, and future Web Builder reuse notes.
- src/routes/publicWidgetRoutes.ts — Manager-only public-widget routes expose workshop readiness, Track My Bike readiness, Quote Approval readiness, Store Info widget readiness, rental/hire request readiness, and aggregated hub metadata.
- frontend/src/pages/PublicWidgetsPage.tsx — Public Widgets hub exists as the central current/future widget management surface with five live widgets, per-widget customer boundaries, and Web Builder handoff guidance.
- frontend/src/pages/SystemSettingsPage.tsx — Store Info settings link to the hub while keeping Store Info-specific widget readiness context.
- src/config/webBuilderTemplateRegistry.ts — Web Builder template registry records the Workshop-first, Retail + workshop, and Premium service templates as available with expected widget references.
- src/services/webBuilderService.ts — Web Builder service reads/writes AppConfig-backed V1 setup state, surfaces Store Info dependencies, gates manager-controlled publishing for available templates, exposes support-safe generated-site diagnostics, exposes manager-only preview payloads, supports scoped public preview/share links, and serves the public generated-site payload only when enabled and ready.
- src/services/webBuilderMediaService.ts — Web Builder media service validates managed PNG/JPEG/WebP image uploads by MIME type, extension, size, and image byte signature before storing them under the managed Web Builder media path.
- src/routes/webBuilderRoutes.ts — Manager-only Web Builder routes expose admin state, template registry, config save endpoints, media image upload/external-URL preparation endpoints, and template preview endpoints for previewable templates.
- src/routes/publicWebBuilderRoutes.ts — Public Web Builder route exposes the manager-selected available generated site only through the public-safe generated-site API.
- frontend/src/pages/WebBuilderSharedPreviewPage.tsx — Shared preview page renders public-safe stakeholder review content for the selected generated-site setup without manager/admin chrome.
- src/routes/publicCatalogueRoutes.ts — Public catalogue route exposes the first read-only generated-site-safe catalogue contract at /api/public/catalogue/products.
- src/services/publicCatalogueService.ts — Public catalogue service filters active public-eligible products/variants, supports purchasable-only filtering, excludes non-stock/internal service SKUs, uses base retail price, derives coarse default-location availability from stock ledger minus active reservations, and omits sensitive stock/supplier/cost fields.
- frontend/src/api/publicCatalogue.ts — Frontend public catalogue client consumes only the public /api/public/catalogue/products contract for generated-site browsing.
- frontend/src/pages/WebBuilderPage.tsx — Web Builder shell lets managers select a template, prepare setup copy, review sections, see Store Info dependency state, confirm widget reuse boundaries, open previewable templates, see a first-run generated-site readiness checklist and generated-site diagnostics, and control the generated-site publish state.

- `frontend/src/pages/WebBuilderTemplatePreviewPage.tsx` — Template preview routing renders the correct previewable template body with Store Info-backed public content and widget route references without staff/admin chrome.
- `frontend/src/features/webBuilder/WebBuilderTemplateBody.tsx` — Shared Web Builder template rendering now applies safe theme preset attributes for preview and public generated-site output.
- `src/services/webBuilderService.ts` — Web Builder config normalization now persists only supported theme preset ids and rejects arbitrary theme keys or raw CSS-style payloads.
- `frontend/src/pages/WebBuilderPage.tsx` — Web Builder shell exposes manager-editable theme controls for accent colour, typography preset, spacing density, and button style without adding a general design editor.
- `frontend/src/pages/WebBuilderPage.tsx` — Web Builder shell exposes image-block controls for template-defined slots, managed uploads, constrained external HTTPS image URLs, alt text, decorative flags, captions, previews, and clear/remove actions.
- `frontend/src/features/webBuilder/TemplateImageBlock.tsx` — Template image-block component renders the selected image-block config consistently in manager preview, public preview/share, and public generated-site contexts.
- `frontend/src/features/webBuilder/WorkshopFirstTemplateBody.tsx` — Workshop-first Template 1 has a polished customer content hierarchy, action-led section order, credible service copy, product section placement, secure follow-up wording, conditional section navigation/footer rendering, stable H1 handling, labelled generated-site sections, clear link names, visible focus styling, and stable Store Info logo dimensions.
- `frontend/src/features/webBuilder/RetailWorkshopTemplateBody.tsx` — Retail + workshop Template 2 balances retail range highlights, workshop service messaging, product browsing, hire request links, secure existing repair links, and Store Info-backed visit details as a publishable generated-site template.
- `frontend/src/features/webBuilder/PremiumServiceTemplateBody.tsx` — Premium service Template 3 frames independent-shop story, premium service highlights, service journey, secure follow-up links, request-first workshop/hire actions, and Store Info-backed visit details as a publishable generated-site template.
- `frontend/src/features/webBuilder/GeneratedProductBrowser.tsx` — Generated product browser renders purchasable public catalogue product cards with search/category filters, coarse availability, fulfilment wording, lazy-created anonymous basket intent, server-recalculated basket estimates, basket-token checkout handoff, and compatibility SKU/quantity fallback without using client-side totals.
- `frontend/src/pages/PublicWebsiteCheckoutPage.tsx` — Public checkout can prefill basket-token or repeated `lineSku/lineQuantity` intent, collect customer contact and shipping details, and submit SKU/quantity lines for server revalidation.
- `src/routes/publicStorefrontBasketRoutes.ts` — Public storefront basket routes expose anonymous `create/restore/get/add/update/remove/checkout` actions under `/api/public/online-store/baskets` without staff authentication.
- `scripts/storefront_basket_smoke_tests.js` — Storefront basket smoke coverage verifies customer-safe basket estimates, coarse quantity blockers, concurrent same-SKU add handling, checkout blocking, and private-field exclusion.
- `scripts/storefront_basket_service_tests.ts` — Storefront basket service tests verify checkout retry recovery, external-reference recovery, and checked-out mutation guards with mocked Stripe checkout.
- `frontend/src/pages/WebBuilderGeneratedSitePage.tsx` — Dedicated public generated-site page renders the manager-selected available template only when publishing is enabled, otherwise showing a current-site fallback; it owns generated-site title, description, canonical metadata, and focus handoff.
- `frontend/src/styles.css` — Generated-site styles now support constrained accent, typography, spacing, and button presets through internal CSS variables while preserving mobile overflow and tap-target safeguards.
- `scripts/web_builder_v1_smoke_tests.js` — Web Builder smoke coverage verifies manager-only access, registry metadata, Template 2 and Template 3 preview and publish behavior, theme default/persistence/partial-patch validation, manager enable/disable publishing, public generated-site rendering for available templates, preview endpoint access, public preview/share behavior, widget reuse references, and Public Widgets compatibility.

- `scripts/public_catalogue_smoke_tests.js` — Public catalogue smoke coverage verifies customer-safe product contract behavior, including purchasable-only filtering, for generated-site product browsing foundations.
- `e2e/workshop/workshop-public.spec.js` — Public Web Builder E2E coverage verifies generated-site checklist and diagnostics visibility, theme controls and themed preview/public attributes, Template 2 and Template 3 preview plus public rendering, metadata, focus handoff, labelled generated-site navigation, heading presence, widget links, generated product cards, product filters, basket-token checkout handoff, compatibility SKU/quantity checkout fallback, private-data boundaries, and mobile overflow/tap-target behavior at 320px, 360px, and tablet widths.
- `docs/support/store-info-settings.md` — Store Info usage is documented.
- `docs/support/public-widgets.md` — Public widget embedding and boundaries are documented.
- `docs/support/web-builder.md` — Web Builder support docs explain the admin shell, template registry, public preview/share links, bounded image blocks, Template 1 onboarding checklist, Template 2 publish readiness, Store Info dependency, widget reuse contract, publish-disabled review flow, escalation blockers, and non-goals.
- `docs/customer_facing_web_builder_widget_architecture_plan.md` — A planning baseline now exists for Web Builder plus shared embeddable widget architecture.
- `docs/web_builder_delivery_plan.md` — Delivery plan records product principles, safety boundaries, widget reuse, multi-agent rules, verification strategy, and readiness gates.
- `docs/web_builder_task_queue.md` — Task queue records completed milestones, ordered next tasks, dependencies, file ownership, parallel-safety, verification, and acceptance criteria.
- `docs/web_builder_commercial_pilot_readiness_audit.md` — Commercial/pilot readiness audit separates Clapham proof-case readiness, standalone Public Widgets usage, package boundaries, support/onboarding gates, entitlement assumptions, and external pilot entry/exit criteria.
- `docs/web_builder_product_browsing_integration_audit.md` — Product browsing audit records the safe contract and first guarded Template 1 runtime consumer for generated-site product sections.
- `docs/clapham_existing_site_widget_trial_plan.md` — Existing-site widget trial plan defines how to test the widget suite on the current Clapham website before generated-site migration.
- `docs/clapham_generated_site_review_evidence.md` — Clapham generated-site review evidence record captures pending review/sign-off gates for /site/generated before live launch.
- `docs/clapham_existing_site_widget_trial_evidence.md` — Clapham existing-site widget-trial evidence record captures pending pass/fail observations for external widget testing.
- `docs/clapham_public_site_migration_risk_rollback.md` — Clapham migration risk/rollback record captures route ownership, critical public link preservation, deployment assumptions, and rollback gates.
- `docs/clapham_existing_site_public_route_inventory.md` — Clapham existing-site public route inventory captures the live Saledock route shape, sitemap scale, public identity, tested URLs, asset hosts, and pending manual confirmations for launch planning.
- `docs/corepos_public_route_inventory_for_clapham_migration.md` — CorePOS public route inventory records public route and API ownership that a generated-site migration must preserve.
- `docs/clapham_public_route_migration_decision_matrix.md` — Clapham route decision matrix records route-family recommendations and PENDING DECISION items before live launch.
- `docs/clapham_public_site_route_map_proposal.md` — Clapham route-map proposal records proposed launch route targets, canonical behavior, rollback notes, and verification expectations without approving live launch.
- `docs/clapham_public_site_migration_remaining_blockers.md` — Clapham remaining-blockers checklist separates Thomas's accepted Task 29 repo implementation risk from hard no-go live launch gates for verification, production-serving, live/staging evidence, deployment ownership, rollback rehearsal, and support ownership.
- `docs/clapham_public_site_migration_verification_package.md` — Clapham migration verification package records Task 29 test groups, route families, production-serving checks, rollback rehearsal evidence, and

pass/fail evidence tables.

- docs/clapham\_public\_site\_launch\_day\_support\_checklist.md — Clapham launch-day support checklist records support ownership, monitoring, customer-safe failure handling, rollback communication, evidence capture, and no-go escalation language.
- scripts/clapham\_route\_ownership\_smoke\_tests.js — Clapham route ownership smoke coverage verifies generated-site route declarations, preserved public workflow routes, widget routes, public APIs, production SPA fallback exclusions, legacy printable route exclusions, operational endpoint ownership, and the approval/launch-gate state without changing runtime behavior.
- scripts/clapham\_migration\_evidence\_consistency\_tests.js — Clapham migration evidence consistency coverage verifies implementation approval versus live launch posture, protected route coverage, verification-package guardrails, route recrawl facts, and feature-map evidence alignment without changing runtime behavior.
- docs/clapham\_public\_route\_recrawl\_2026-05-15.md — Clapham public route recrawl refresh confirms both public hosts, robots and sitemap behavior, representative live/404/redirect route statuses, external redirects, and canonical-host uncertainty without approving live launch.
- docs/clapham\_public\_site\_final\_route\_map\_candidate.md — Clapham final route-map candidate is now the accepted Task 29 repo implementation basis after Thomas's 2026-05-16 approval while preserving DNS/custom-domain/production launch blockers.
- docs/clapham\_public\_site\_migration\_approval\_packet.md — Clapham migration approval packet records Thomas Witherspoon's 2026-05-16 decision status as APPROVED TO IMPLEMENT TASK 29 while leaving live launch evidence pending.
- docs/roadmap-events/2026-05-16-clapham-public-site-migration-implementation-approval.json — Roadmap event records Task 29 implementation approval and launch-readiness non-goals.
- docs/roadmap-events/2026-05-16-web-builder-commercial-pilot-readiness-audit.json — Roadmap event records the commercial/pilot readiness audit and its non-goals.
- docs/roadmap-events/2026-05-16-web-builder-storefront-phase1.json — Roadmap event records the storefront phase-1 milestone and its ecommerce non-goals.
- docs/roadmap-events/2026-05-16-web-builder-storefront-basket-foundation.json — Roadmap event records the storefront basket foundation and its ecommerce non-goals.
- docs/web\_builder\_customer\_account\_checkout\_order\_history\_contract.md — Task 35 contract defines the optional account checkout and account-linked order-history runtime boundary for generated-site storefront work.
- docs/roadmap-events/2026-05-16-web-builder-customer-account-checkout-order-history-contract.json — Roadmap event records the Task 35 customer-account checkout/order-history contract and its runtime non-goals.
- frontend/src/pages/CustomerAccountOrdersPage.tsx — Customer account web-order history page renders account-owned order list/detail states as view-only and warns that guest orders are not inferred from contact/order text.
- src/services/customerAccountService.ts — Customer account order-history service returns orders only for the authenticated account's linked customer id.
- src/services/onlinePaymentEmailService.ts — Paid storefront payment-received emails now include customer-safe receipt/invoice summary lines, totals, delivery/collection wording, and refund context without creating a second automatic email path.
- frontend/src/pages/OnlineStoreOrdersPage.tsx — Manager order detail exposes storefront receipt/invoice email send state, safe failure context, and an explicit resend action.
- docs/web\_builder\_storefront\_courier\_adapter\_decision\_audit.md — Task 37 records an EasyPost/aggregator-first storefront courier decision and direct-carrier deferral gates while keeping public rate shopping, address validation, direct carrier adapters, and delivery promises out of scope.

- docs/roadmap-events/2026-05-16-web-builder-storefront-courier-adapter-decision-audit.json — Roadmap event records the Task 37 storefront courier-adapter decision audit.
- src/services/shipping/storefrontShippingService.ts — Task 38 adds a server-owned storefront shipping-options service that validates UK addresses, requests quote-only provider rates, and revalidates the selected option before checkout creates a paid web order.
- src/routes/publicOnlinePaymentRoutes.ts — Public online-payment routes expose /api/public/online-store/shipping-options before the public checkout route.
- prisma/migrations/20260617120000\_web\_order\_shipping\_quote\_snapshot/migration.sql — WebOrder now persists the selected shipping provider/service/rate/address-validation snapshot used for the trusted payment total.
- docs/roadmap-events/2026-05-16-web-builder-storefront-shipping-rates-address-validation-foundation.json — Roadmap event records the Task 38 storefront shipping-rate/address-validation foundation.
- src/services/webOrderNotificationService.ts — Task 39 centralizes storefront collection/dispatch notification decisions, idempotent send state, and support-safe skipped/failed/review-required outcomes.
- src/services/shipping/providerSyncService.ts — Provider sync can surface authoritative tracking/exception states into web-order notification decisions without letting notifications mutate provider truth.
- docs/roadmap-events/2026-05-16-web-builder-storefront-collection-dispatch-notifications.json — Roadmap event records the Task 39 storefront collection/dispatch notification foundation.
- docs/web\_builder\_storefront\_merchandising\_product\_media\_publish\_controls\_plan.md — Task 40 records the implementation-ready plan for explicit manager-reviewed public product publishing, browse-only versus purchasable states, server-owned merchandising order, and public-safe product media boundaries.
- docs/roadmap-events/2026-05-16-web-builder-storefront-merchandising-product-media-publish-controls.json — Roadmap event records the Task 40 storefront merchandising/product-media/publish-controls planning milestone.
- prisma/migrations/20260619120000\_storefront\_product\_publishing\_controls/migration.sql — Task 41 adds durable storefront product and variant publishing controls plus server-owned category, featured, and sort-order fields while keeping active products hidden by default.
- src/services/storefrontProductPublishingService.ts — Storefront product publishing service provides the manager-owned runtime for explicit product/variant visibility, browse-only versus purchasable state, and category/featured/sort ordering.
- src/controllers/storefrontProductPublishingController.ts — Storefront product publishing controller validates manager publishing updates before they reach the service layer.
- src/routes/onlineStoreRoutes.ts — Online Store routes now include the manager-only product-publishing API under /online-store/products alongside the existing orders surface.
- frontend/src/pages/OnlineStoreProductsPage.tsx — The /online-store/products manager page exposes the explicit public product/variant publishing control surface without making active products public automatically.
- frontend/src/api/onlineStoreProducts.ts — Frontend online-store product API helper consumes the manager publishing contract for the new control surface.
- src/services/publicCatalogueService.ts — Public catalogue v3 now reports publicationAuthority and mediaAuthority, includes browse-only public products without checkout handoff, applies server-owned category/featured/sort ordering, and exposes only approved public-safe primaryImage data.
- src/services/orderService.ts — Public website checkout revalidates direct SKU checkout lines against current storefront publishing state before order creation.
- src/services/publicStorefrontBasketService.ts — Public basket checkout revalidates basket lines against current storefront publishing state before handoff.
- scripts/public\_catalogue\_smoke\_tests.js — Public catalogue smoke coverage includes publishing-state visibility, browse-only non-checkout behaviour, publicationAuthority, and server-owned ordering assertions.

- `scripts/storefront_basket_smoke_tests.js` — Storefront basket smoke coverage includes publishing-state revalidation for basket checkout.
- `scripts/stripe_website_checkout_smoke_tests.js` — Website checkout smoke coverage includes publishing-state revalidation for direct checkout.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-product-publishing-runtime-foundation.json` — Roadmap event records the Task 41 product publishing runtime foundation and its non-goals.
- `prisma/migrations/20260620120000_storefront_product_media_runtime/migration.sql` — Task 42 adds durable storefront product media records with source, primary selection, public-safe review state, alt text, focal point, and upload metadata.
- `src/services/storefrontProductMediaService.ts` — Storefront product media service validates PNG/JPEG/WebP uploads, HTTPS external URLs, local/private/scriptable URL boundaries, public-safe approval, alt text, and primary selection.
- `src/controllers/storefrontProductMediaController.ts` — Storefront product media controller exposes manager-only `list/upload/external-url/update/delete` handlers for product media.
- `frontend/src/pages/OnlineStoreProductsPage.tsx` — The manager product publishing page includes product media upload, external URL, approval, public-safe, alt text, focal point, and primary selection controls.
- `src/services/publicCatalogueService.ts` — Public catalogue v3 reports `mediaAuthority` and exposes only approved public-safe `primaryImage` data with alt text while excluding review notes and unapproved media.
- `frontend/src/features/webBuilder/GeneratedProductBrowser.tsx` — Generated product browser renders approved primary product images with fixed-aspect fallbacks for text-only cards.
- `requests/online_store_product_publishing.http` — Manual request examples cover product media upload, external URL registration, approval, blocking, primary selection, and removal.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-product-media-runtime-foundation.json` — Roadmap event records the Task 42 storefront product media runtime foundation.
- `src/services/webBuilderService.ts` — Task 43 adds generated-site storefront diagnostics for public product availability, category readiness, featured-product exposure/suppression, `product-media/fallback` state, `browse-only/checkout` state, and Clapham launch blockers.
- `frontend/src/pages/WebBuilderPage.tsx` — The Web Builder first-run checklist surfaces product, category, product-media, and launch-evidence rows from storefront diagnostics.
- `scripts/web_builder_v1_smoke_tests.js` — Web Builder smoke coverage verifies the Task 43 diagnostics and private-data leakage boundaries.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-merchandising-diagnostics-launch-readiness.json` — Roadmap event records the Task 43 storefront merchandising diagnostics and launch-readiness milestone.

#### Remaining gaps (5)

- The three initial templates are publishable on the dedicated generated-site route with constrained theme presets, public preview/share links, and bounded image blocks; multi-page generated sites, unrestricted media management, advanced per-template theming, and a general CMS/page builder remain future work. high
- Clapham Cycle generated-site replacement is now approved for Task 29 repo implementation and has evidence-capture, route-map candidate, approval packet, verification, support planning, route-ownership smoke, source-evidence consistency, and live-route recrawl records, but live DNS/custom-domain/production launch evidence, policy/external-route/high-value legacy redirect decisions, existing-site widget trial rehearsal, rollback rehearsal, deployment ownership, and manager/admin launch sign-off remain unfinished. high
- The commercial/pilot audit defines package boundaries, but billing, entitlement checks, custom-domain provisioning, generic future-store onboarding, support SLA/owner model, and an external pilot process are not implemented. high

- Task 43 now provides storefront merchandising diagnostics, but exact stock quantities, direct carrier adapters, product/package-specific parcel rules, delivery promises, unrestricted CMS media, and deeper storefront behavior remain future work. **medium**
- Workshop booking, Track My Bike, Quote Approval, Store Info, and rental request iframes are live, but script-loader embeds, auto-resize, and per-origin allowlist UI remain future work. **medium**

#### Recommended next tasks (7)

- Fill or accept the generated-site review, existing-site widget-trial, risk/rollback, deployment-owner, support-owner, production-serving, and remaining launch evidence records before DNS/custom-domain/production launch. **high**
- Use the commercial/pilot readiness audit to design a generic future-store onboarding packet, support runbook, entitlement/pricing boundary, custom-domain launch process, and external pilot entry/exit matrix before scheduling a non-Clapham store. **high**
- Resolve remaining policy-page, external club/bike-fit, high-value legacy redirect, deployment, and rollback ownership decisions before live launch. **high**
- Review generated-site image choices, alt/decorative wording, aspect fit, external image hosts, and mobile layout before live launch. **high**
- Reuse the public catalogue and basket-token contracts in future templates only with purchasable-only filtering, server-side estimates, and server-side price/stock authority intact. **medium**
- Do not start Task 44 Clapham production launch rehearsal/sign-off until real Clapham review, widget trial, deployment, rollback, production-serving, support-owner, and final manager/admin evidence is available. **high**
- Use docs/web\_builder\_task\_queue.md as the source of sequencing truth for generated-site image review, diagnostics, and public-site migration work. **medium**

#### Ideas and dependencies

##### Ideas

- Track website maturity as layers: Store Info wiring, templates, content blocks, widget composition, and publish workflow.

##### Dependencies

- Customer account / portal
- Online workshop booking
- Customer-facing storefront
- Google Business Profile / Reviews / Maps

## Customer account / portal

The first customer account + workshop portal unification milestone is implemented, with the dashboard now prioritising customer next actions before the supporting account sections. The account is the customer workshop home while quote/progress and booking-manage token flows remain valid deep links, saved bikes can open a focused customer-safe profile/history/detail view with service reminder readiness, customers can manage a conservative phone/contact-channel preference foundation without changing identity or staff CRM data, Messages & updates gives read-only visibility into safe workshop communications, authenticated account rental booking visibility now shows customer-safe staff-confirmed hire summaries, and manager-side post-service follow-up readiness/decisions/preview-copy history/outcomes can respect contact permissions without customer-facing automation.

High

Built, Needs Hardening

80%

**Target:** customer-facing, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against customer account pages, routes, and smoke coverage.

### Feature maturity

80%

#### Evidence links (17)

- `frontend/src/pages/CustomerAccountLoginPage.tsx` — Customer account login UI exists.
- `frontend/src/pages/CustomerAccountDashboardPage.tsx` — Customer account dashboard exists and renders a prioritised next-actions section.
- `frontend/src/pages/CustomerAccountProfilePage.tsx` — Customer account profile/preferences page exists for safe phone and contact-channel preference updates.
- `frontend/src/pages/CustomerAccountCommunicationHistoryPage.tsx` — Customer account Messages & updates page exists for read-only safe communication history.
- `frontend/src/pages/CustomerAccountBikeDetailPage.tsx` — Customer account bike detail page exists and shows read-only service reminder readiness.
- `frontend/src/pages/CustomerAccountRentalsPage.tsx` — Customer account rental list/detail pages show customer-safe staff-confirmed hire booking summaries as read-only account views.
- `frontend/src/pages/CustomerCommunicationQueuePage.tsx` — Manager communication queue uses customer contact permissions in no-send post-service follow-up readiness, decision, durable preview/copy history, manual preview, gap filter, and manual outcome panels.
- `src/routes/customerAccountRoutes.ts` — Customer account routes include dashboard, profile/preferences, communications, bike-detail, and rental booking visibility endpoints.
- `src/services/customerAccountService.ts` — Dashboard, profile/preferences, communication-history, bike-detail, and rental booking responses are shaped for safe customer display across prioritised next actions, profile fields, active jobs, links, bikes, history, messages, reminder readiness, and staff-confirmed hire summaries.
- `scripts/customer_account_identity_smoke_tests.js` — Customer account identity smoke coverage verifies next-action priority, profile/preference access and validation, communication-history filtering, reminder-readiness labels/no-send side effects, safe links/fallbacks, ownership isolation, internal decision/preview/outcome exclusion, internal-field exclusion, and empty-account state.
- `scripts/customer_account_rental_visibility_smoke_tests.js` — Customer account rental visibility smoke coverage verifies own-booking list/detail access, cross-customer isolation, safe DTO exclusions, public request-status safety, and no payment/accounting mutations.
- `e2e/workshop/workshop-public.spec.js` — Customer account browser coverage exercises the dashboard profile/preferences link, communication-history empty state, preference save flow, and bike-detail reminder readiness.
- `docs/rental_hire_customer_booking_visibility_foundation.md` — Customer rental booking visibility doc records account-only readback and explicit no self-service/payment/upload/signing/internal-data boundaries.

- docs/customer\_facing\_web\_builder\_widget\_architecture\_plan.md — Architecture plan now records customer account + workshop portal unification as implemented.
- docs/web\_builder\_customer\_account\_checkout\_order\_history\_contract.md — Task 35 contract records how account-aware storefront checkout and order-history runtime work should attach by authenticated customer id while preserving guest checkout and token-link compatibility.
- src/services/customerAccountService.ts — Customer account service now includes account-owned web-order list/detail queries scoped by the authenticated account's customer id.
- frontend/src/pages/CustomerAccountOrdersPage.tsx — Customer account order-history UI exists for customer-safe storefront order list/detail readback with no guest-order inference or order-state mutation.

#### Remaining gaps (3)

- Confirmed rental booking readback and storefront order-history readback are account-only and view-only; rental self-service, public confirmation, secure readback tokens, customer-visible receipt/invoice controls, and broader longitudinal account history remain future work. high
- Richer service reminder/review-request consent, verified email-change, customer reply behavior from the account, outbound reminder sending, and outbound follow-up/review-request workflows are still future customer-account work despite the staff-only preview and outcome aids. high
- Customer-facing rollout guidance, support docs, and trust/diagnostics signals still need commercial hardening. medium

#### Recommended next tasks (3)

- Harden customer-account profile/privacy review coverage before expanding identity, consent, reminder-sending, or message payloads. high
- Keep customer account order history view-only while customer-visible receipt/invoice controls and post-purchase storefront support mature. high
- Decide how far customer-account bike profiles should go into attachments and maintenance recommendations before expanding broader account history. medium

#### Ideas and dependencies

##### Ideas

- Split portal maturity into identity, self-service, and ongoing customer relationship stages.

##### Dependencies

- Track My Bike / Workshop Progress View
- Online workshop booking

## Track My Bike / Workshop Progress View

This is not missing. The repo already has a secure workshop portal with progress, quote approval, conversation, attachments, timeline, account-dashboard deep links, an iframe Track My Bike widget foundation, and an iframe Quote Approval widget foundation that open existing secure links without public lookup.

High

Built, Needs Hardening

77%

**Target:** customer-facing, trial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against public workshop routes, portal UI, and workshop conversation/attachment services.

### Feature maturity

77%

#### Evidence links (11)

- `src/routes/publicWorkshopQuoteRoutes.ts` — Public workshop portal routes exist.
- `frontend/src/pages/WorkshopQuotePage.tsx` — Customer portal UI exists.
- `src/services/workshopConversationService.ts` — Portal conversation service exists.
- `src/services/workshopAttachmentService.ts` — Portal attachments service exists.
- `frontend/src/pages/CustomerAccountDashboardPage.tsx` — Customer account dashboard links into quote/progress and collection views.
- `frontend/src/pages/TrackMyBikeWidgetPage.tsx` — Track My Bike iframe widget exists as a secure-link helper and does not display repair data inside the iframe.
- `frontend/src/pages/QuoteApprovalWidgetPage.tsx` — Quote Approval iframe widget exists as a secure-link helper and does not display estimate data inside the iframe.
- `src/services/publicWidgetService.ts` — Public Widgets readiness includes Track My Bike and Quote Approval URLs, iframe snippets, secure portal reuse checks, no-public-lookup boundaries, and limitations.
- `scripts/workshop_m13_smoke_tests.js` — Public Widgets smoke coverage includes manager-only Track My Bike and Quote Approval snippet/readiness metadata plus safe access-flow flags.
- `e2e/workshop/workshop-public.spec.js` — Public workshop E2E covers the Track My Bike and Quote Approval iframe routes, lack of public chrome, and secure portal link preparation.
- `docs/support/public-widgets.md` — Support docs explain Track My Bike and Quote Approval embedding, secure-link use, and privacy boundaries.

#### Remaining gaps (2)

- Portal polish, customer wording, and broader regression coverage still need hardening before it feels fully trial-safe. **high**
- The first iframe widgets are secure-link helpers only; they do not show compact validated progress or quote summaries inside the iframe. **medium**

#### Recommended next tasks (2)

- Add a focused portal hardening checklist covering progress wording, approvals, attachments, and reply flows. **high**

- Decide whether later progress or quote widgets should add server-validated token handoff/rendered summaries or stay as link-out helpers while the secure portal remains the decision engine. **medium**

#### Ideas and dependencies

##### Ideas

- Track portal maturity separately from workshop internal maturity so customer-facing readiness is visible.

##### Dependencies

- Communication Centre
- Workshop service history and customer portal

## Online workshop booking

Public workshop booking now has a server-side capacity contract, staff diagnostics, a hardened iframe-safe widget route, and a central Public Widgets manager hub for readiness, snippets, preview, and live companion widget slots.

Medium

Active Hardening

76%

**Target:** customer-facing, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against public booking pages, routes, and roadmap guidance.

### Feature maturity

76%

#### Evidence links (13)

- `frontend/src/pages/PublicWorkshopBookingPage.tsx` — Public workshop booking page exists and now supports compact widget mode for `/widgets/workshop-booking`.
- `frontend/src/pages/PublicWorkshopBookingManagePage.tsx` — Public booking manage page exists.
- `src/routes/workshopBookingRoutes.ts` — Workshop booking routes exist.
- `src/services/workshopAvailabilityService.ts` — Booking availability now centralizes Store Info hours, special hours, rota closed days, lead-date checks, daily caps, and capacity decisions.
- `src/services/workshopBookingService.ts` — Public booking creation and manage-token rescheduling enforce the central availability rules server-side.
- `frontend/src/pages/WorkshopBookingsPage.tsx` — Staff booking readiness panel surfaces closures, capacity blocks, and setup warnings.
- `src/services/publicWidgetService.ts` — Manager-only widget service emits workshop readiness plus the aggregated Public Widgets hub list.
- `frontend/src/pages/PublicWidgetsPage.tsx` — Public Widgets hub surfaces live widget readiness, URLs, snippets, previews, support notes, and future widget slots when present.
- `frontend/src/pages/SystemSettingsPage.tsx` — Store Info settings keep a concise Website Widgets summary and link to the hub.
- `scripts/workshop_m13_smoke_tests.js` — M13 smoke coverage includes widget hub auth, snippet generation, live widget metadata, capacity blocking, and staff-only diagnostics.
- `e2e/workshop/workshop-public.spec.js` — Public workshop E2E covers iframe widget routes and the manager-only Public Widgets hub.
- `docs/ROADMAP.md` — Roadmap records workshop booking capacity hardening and the iframe workshop booking widget foundation.
- `docs/support/public-widgets.md` — Support docs explain the external-site iframe snippet, preview, platform guidance, troubleshooting, and security boundaries.

#### Remaining gaps (2)

- The first widget is still iframe-only; there is no host-page script loader, auto-resize bridge, per-origin admin allowlist UI, or advanced theme configuration yet. medium
- Staff override policy, service-category duration tuning, and confirmation wording still need more operational polish before broad public rollout. medium

## Recommended next tasks (2)

- Use trial feedback from the Public Widgets hub to decide whether iframe auto-resize, script-loader support, per-origin allowlist UI, or richer widget theming should come next. **medium**
- Add staff override/confirmation guidance for exceptions that customers cannot self-book online. **medium**

## Ideas and dependencies

### Ideas

- Track online booking maturity separately from the web-builder/widget layer to avoid over-crediting the website shell.

### Dependencies

- Web builder / templates / embeddable widgets
- Workshop UX and behaviour audit

## Communication Centre: email, SMS, WhatsApp

This is also not missing. Notification orchestration and channel services already exist, customers now have a read-only account view of safe workshop messages and delivered notification copies, and managers have a no-send post-service follow-up loop with readiness, internal review decisions, durable preview/copy history, manual preview assistance, manual outcome logging, operational ageing/reporting, fixed ageing-policy guidance, consent/suppression/review-request governance policy, gap filters, and dashboard attention surfacing for completed workshop jobs; the remaining work is product-data design, trust, rollout, and outbound policy implementation discipline.

High

Built, Needs Hardening

73%

**Target:** customer-facing, trial, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against notification orchestration, delivery services, and customer communication UI.

### Feature maturity

73%

#### Evidence links (13)

- `src/services/notificationService.ts` — Notification orchestration service exists.
- `src/services/emailService.ts` — Email delivery service exists.
- `src/services/smsService.ts` — SMS delivery service exists.
- `src/services/whatsappService.ts` — WhatsApp delivery service exists.
- `frontend/src/pages/CustomerAccountCommunicationHistoryPage.tsx` — Customer account communication-history page shows read-only safe workshop messages and delivered shop updates.
- `frontend/src/pages/CustomerCommunicationQueuePage.tsx` — Manager communication queue now includes no-send post-service follow-up readiness, operations ageing, internal decision, durable preview/copy history, manual preview, manual outcome panels, gap filters, and print-review-pack controls for completed workshop jobs.
- `frontend/src/pages/ManagementDashboardPage.tsx` — Management dashboard includes a read-only follow-up attention tile linking to the Customer Communication Queue without send actions.
- `src/services/reports/customerReports.ts` — Customer reports include shaped post-service follow-up readiness and operations reporting, manager-only decision/outcome history, durable preview/copy events, ageing buckets, priority/deferred/stale item lists, gap filters, and gated manual preview output using contact preferences and duplicate communication evidence without raw message/provider fields.
- `docs/post_service_follow_up_remaining_work_plan.md` — Remaining-work plan now records completed durable preview/copy event tracking, reporting integration, history UI, gap filters, staff runbook polish, review-pack printing, and dashboard surfacing while preserving no-send boundaries.
- `docs/post_service_follow_up_ageing_policy.md` — Ageing policy documents fixed post-service follow-up buckets, the 8-day stale threshold for pending work, deferred/no-response interpretation, timestamp basis, and no-send boundaries.
- `docs/customer_post_service_follow_up_policy.md` — Governance policy defines consent, suppression, review-request wording, Google separation, approval/audit, duplicate evidence, and outbound boundaries before any future follow-up or review-request workflow.
- `scripts/customer_account_identity_smoke_tests.js` — Customer account smoke coverage verifies communication-history ownership, safe-link shaping, customer exclusion from internal decision/preview/outcome endpoints, and internal/suppressed/provider metadata exclusion.
- `scripts/customer_reminders_m124_smoke_tests.js` — Customer reminders smoke coverage verifies post-service follow-up readiness labels, operations ageing/counts, manager auth, decision/outcome

validation/history, durable preview/copy events, gap filters, draft-preview gating, internal-field exclusion, and no-send side effects.

#### Remaining gaps (2)

- Channel reliability, support tooling, and operator trust signals still need more maturity than the raw transport foundation. **high**
- Direct messaging, delivery-state explanations, customer reply behavior, dedicated consent/suppression product data, and outbound follow-up/review-request implementation remain future work beyond the no-send decision, preview/copy, outcome-logging, ageing-policy, and governance-policy foundation. **medium**

#### Recommended next tasks (3)

- Translate post-service follow-up consent, suppression, review-request wording, Google separation, and approval policy into explicit product-data requirements before any outbound workflow is considered. **high**
- Clarify the maturity ladder for workshop notifications versus broader omnichannel communication. **high**
- Add support-facing delivery diagnostics and operator guidance to the dashboard links and docs. **medium**

#### Ideas and dependencies

##### Ideas

- Track communication maturity by channel trust, orchestration depth, and customer self-service coverage.

##### Dependencies

- Track My Bike / Workshop Progress View
- Customer account / portal

## Customer-facing storefront

Website checkout, online-store foundations, the first public catalogue contract, a guarded generated-site purchasable product browsing consumer, the Task 34 anonymous storefront basket foundation, the Task 35A customer-account checkout/order-history runtime foundation, the Task 36 storefront receipt/invoice email foundation, the Task 37 EasyPost/aggregator-first courier decision audit, the Task 38 shipping-rate/address-validation runtime foundation, the Task 39 collection/dispatch notification foundation, the Task 40 storefront merchandising/product-media/publish-controls plan, the Task 41 storefront product publishing runtime foundation, the Task 42 product media runtime foundation, and the Task 43 merchandising diagnostics/launch-readiness foundation now exist. Product browsing exposes customer-safe product identity, explicit product/variant publishing state, publicationAuthority, browse-only public product states without checkout handoff, purchasable-only filtering, search/category filters, basket-token checkout handoff, compatibility SKU checkout fallback, base retail price display, coarse availability, fulfilment wording, approved public-safe primary images, fixed-aspect fallback cards, and server-owned category/featured/sort ordering. Basket preview stores SKU/quantity intent only, recalculates customer-safe estimates server-side, blocks unavailable lines and coarse over-limit quantities, handles concurrent same-SKU adds, guards line mutations against checkout races, atomically claims checkout before creating the public website checkout, and revalidates basket lines against current publishing state. Checkout can be retried against the same basket token when a successful response is lost. Checkout can capture contact and shipping details while still sending SKU/quantity intent for server-side validation, and direct SKU checkout revalidates against current publishing state before order creation. Shipping checkout now requests server-owned UK shipping options, requires a selected validated option before Stripe opens, revalidates that option server-side, persists the selected provider/service/address-validation snapshot on WebOrder, and includes the trusted shipping charge in WebOrder/Stripe totals. Collection/dispatch notifications now cover ready-for-collection, collected, dispatched, tracking-available, delivery-exception, and cancellation/refund fulfilment states using authoritative CorePOS fulfilment/provider state, idempotent sends, and staff-visible skipped/failed/review-required outcomes. Signed-in direct storefront checkout and basket checkout can attach new WebOrders by authenticated customer session only. Customer account order history is scoped by WebOrder.customerId, not email, phone, name, postcode, basket token, or order-text matching, and stays view-only without payment/refund/shipment/stock/accounting mutation. Paid storefront orders enrich the existing idempotent Stripe payment-received email with customer-safe receipt/invoice body content and manager-visible resend/failure review without adding payment/accounting mutation. Web Builder diagnostics now explain public product availability, category readiness, featured-product exposure or suppression, product media fallback/review state, browse-only versus checkout-ready state, and Clapham launch blockers without changing public catalogue, media, basket, checkout, payment, fulfilment, or route authority. This is still not a finished storefront product because direct carrier adapters, product/package-specific parcel rules, exact stock quantities, exact delivery promises, DNS/custom-domain launch, and broader post-purchase support remain separate milestones.

Medium

Foundation Exists

71%

**Target:** customer-facing, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against website checkout, online-store routes, and smoke coverage.

### Feature maturity

71%

#### Evidence links (75)

- `frontend/src/pages/PublicWebsiteCheckoutPage.tsx` — Public website checkout page exists.
- `frontend/src/pages/OnlineStoreOrdersPage.tsx` — Staff-side online store orders page exists.
- `src/routes/onlineStoreRoutes.ts` — Online store routes exist.
- `src/routes/publicCatalogueRoutes.ts` — Public catalogue route exposes `/api/public/catalogue/products` without staff authentication or manager-only online-store data.
- `src/services/publicCatalogueService.ts` — Public catalogue service provides server-owned product/variant eligibility, purchasable-only filtering, price source, coarse availability, fulfilment eligibility, checkout handoff metadata, and sensitive-field exclusion for generated-site browsing.
- `frontend/src/features/webBuilder/GeneratedProductBrowser.tsx` — Generated-site product browsing consumes the public catalogue and storefront basket contracts, renders purchasable product cards and coarse availability, supports search/category filtering, lazy-creates anonymous basket intent on first add, and links to checkout by basket token without client-side totals.

- `frontend/src/pages/PublicWebsiteCheckoutPage.tsx` — Public checkout can accept generated-site basket-token or `lineSku/lineQuantity` intent, capture customer contact and shipping details, and still submits SKU/quantity lines for server-side checkout revalidation.
- `src/routes/publicStorefrontBasketRoutes.ts` — Public storefront basket routes expose anonymous basket `create/restore/get/add/update/remove/checkout` actions under `/api/public/online-store/baskets`.
- `src/services/publicStorefrontBasketService.ts` — Public storefront basket service stores SKU/quantity intent, recalculates customer-safe totals and availability warnings, blocks unavailable lines and coarse over-limit quantities, handles same-SKU concurrent adds, guards checked-out basket mutations, persists checkout recovery metadata, and atomically claims checkout before public website checkout creation.
- `frontend/src/api/publicStorefrontBasket.ts` — Frontend public basket API helper normalizes the server basket contract for generated-site browsing and checkout surfaces.
- `scripts/online_store_smoke_tests.js` — Online store smoke coverage exists.
- `scripts/public_catalogue_smoke_tests.js` — Public catalogue smoke coverage verifies catalogue eligibility, purchasable-only filtering, price, availability, fulfilment, filtering, pagination, and sensitive-field boundaries.
- `scripts/storefront_basket_smoke_tests.js` — Storefront basket smoke coverage verifies anonymous basket lifecycle behavior, server estimates, unavailable-line and coarse quantity blockers, concurrent add handling, checkout blocking, checked-out mutation rejection, and private-field boundaries.
- `scripts/storefront_basket_service_tests.ts` — Storefront basket service tests verify mocked-Stripe checkout recovery, external-order-reference recovery, and mutation blocking after checkout.
- `e2e/workshop/workshop-public.spec.js` — Public Web Builder E2E coverage verifies generated-site product filters, basket creation, multiple basket lines, basket-token checkout review, and compatibility SKU/quantity checkout fallback.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-phase1.json` — Roadmap event records the storefront phase-1 milestone.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-basket-foundation.json` — Roadmap event records the storefront basket foundation milestone.
- `docs/web_builder_customer_account_checkout_order_history_contract.md` — Task 35 contract defines optional signed-in storefront checkout, strict account order-history ownership, customer-safe order-state wording, privacy exclusions, and runtime verification expectations.
- `docs/roadmap-events/2026-05-16-web-builder-customer-account-checkout-order-history-contract.json` — Roadmap event records the Task 35 customer-account checkout/order-history contract milestone.
- `src/controllers/publicOnlinePaymentController.ts` — Public checkout passes authenticated customer account context into website checkout creation without accepting customer ids from the browser.
- `src/controllers/publicStorefrontBasketController.ts` — Storefront basket checkout passes authenticated customer account context into basket checkout without changing guest basket-token checkout.
- `src/services/orderService.ts` — Online store order creation accepts server-derived customer id attachment for signed-in checkout while preserving checkout authority for price, reservation, payment, and fulfilment.
- `src/services/customerAccountService.ts` — Customer account order-history queries `list/detail` WebOrders only where the order customer id matches the authenticated account customer id.
- `frontend/src/api/customerAccountOrders.ts` — Frontend customer-account order API helper consumes the account order `list/detail` endpoints.
- `frontend/src/pages/CustomerAccountOrdersPage.tsx` — Customer account web-order `list/detail` page is view-only, says guest orders are not matched by `contact/order` text, and avoids `payment/refund/shipment/accounting` mutation.
- `docs/roadmap-events/2026-05-16-web-builder-customer-account-checkout-order-history-runtime.json` — Roadmap event records the Task 35A runtime foundation for account-owned storefront order history.

- `src/services/onlinePaymentEmailService.ts` — Payment-received emails for paid storefront orders now include customer-safe order lines, totals, delivery or collection wording, refund context, and non-VAT-invoice wording while preserving one automatic email path.
- `src/services/onlinePaymentService.ts` — Explicit payment receipt resend claims the email state atomically, rejects in-flight resend attempts, and returns safe delivery status without raw provider failure text.
- `src/services/orderService.ts` — Online store order detail exposes latest payment email state and adds the manager-controlled storefront receipt email resend service.
- `frontend/src/pages/OnlineStoreOrdersPage.tsx` — Manager order detail shows storefront receipt email send/review state and a resend action for paid orders.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-receipt-invoice-email-foundation.json` — Roadmap event records the Task 36 storefront receipt/invoice email foundation and its payment/accounting non-goals.
- `docs/web_builder_product_browsing_integration_audit.md` — Audit separates checkout groundwork from public catalogue, product browsing, price, stock, fulfilment, and delivery boundaries and records the first guarded generated-site consumer.
- `docs/web_builder_storefront_courier_adapter_decision_audit.md` — Task 37 records an EasyPost/aggregator-first storefront courier decision and direct Royal Mail, DPD, DHL, or other branded carrier deferral gates.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-courier-adapter-decision-audit.json` — Roadmap event records the Task 37 storefront courier-adapter decision audit and its non-goals.
- `src/services/shipping/storefrontShippingService.ts` — Storefront shipping service validates UK shipping addresses, asks the configured provider for quote-only rates, and revalidates the selected option before checkout creates a paid web order.
- `src/services/shipping/easyPostProvider.ts` — EasyPost provider exposes address validation and rate-quote methods without buying labels during checkout.
- `src/services/orderService.ts` — Public website checkout requires a selected shipping option for shipping fulfilment, revalidates it server-side, persists the selected shipping snapshot, and includes the trusted shipping charge in the Stripe amount.
- `frontend/src/pages/PublicWebsiteCheckoutPage.tsx` — Public checkout UI lets customers check server-returned delivery options, choose a service, and blocks Stripe until shipping orders have a selected option.
- `prisma/migrations/20260617120000_web_order_shipping_quote_snapshot/migration.sql` — `WebOrder` stores selected shipping provider, service, rate reference, currency, address-validation status/message, quote reference, expiry, and metadata.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-shipping-rates-address-validation-foundation.json` — Roadmap event records the Task 38 shipping-rate/address-validation foundation and its non-goals.
- `src/services/webOrderNotificationService.ts` — Storefront notification service sends customer-safe collection/dispatch messages from authoritative fulfilment/provider states, records idempotent send state, and exposes skipped/failed/review-required outcomes for staff review.
- `src/services/shipping/providerSyncService.ts` — Provider sync contributes tracking/exception state for notification decisions while keeping provider, fulfilment, payment, refund, stock, and accounting truth separate from email delivery.
- `frontend/src/pages/OnlineStoreOrdersPage.tsx` — Manager order detail exposes collection/dispatch notification state and review context alongside existing storefront order support controls.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-collection-dispatch-notifications.json` — Roadmap event records the Task 39 collection/dispatch notification foundation and its non-goals.
- `docs/web_builder_storefront_merchandising_product_media_publish_controls_plan.md` — Task 40 plan defines explicit public product states, manager product-publishing controls, browse-only behaviour, server-owned category/featured order, and product media boundaries.

- docs/roadmap-events/2026-05-16-web-builder-storefront-merchandising-product-media-publish-controls.json — Roadmap event records the Task 40 storefront merchandising/product-media/publish-controls planning milestone.
- prisma/migrations/20260619120000\_storefront\_product\_publishing\_controls/migration.sql — Task 41 adds durable storefront product and variant publishing fields plus server-owned category, featured, and sort-order fields.
- src/services/storefrontProductPublishingService.ts — Storefront product publishing service owns manager-reviewed product/variant publication state, browse-only versus purchasable behavior, and server-owned merchandising order.
- src/controllers/storefrontProductPublishingController.ts — Storefront product publishing controller validates manager product publishing requests and keeps transport validation outside the service layer.
- src/routes/onlineStoreRoutes.ts — Online Store routes expose the manager-only `/online-store/products` product publishing API alongside existing order management.
- frontend/src/pages/OnlineStoreProductsPage.tsx — `/online-store/products` gives managers explicit product and variant publishing controls without treating active inventory as automatically public.
- frontend/src/api/onlineStoreProducts.ts — Frontend online-store product API helper consumes the manager product publishing contract.
- src/services/publicCatalogueService.ts — Public catalogue v3 filters by explicit publishing state, reports publicationAuthority and mediaAuthority, includes browse-only public products without checkout handoff, applies server-owned category/featured/sort ordering, and exposes only approved public-safe primaryImage data.
- frontend/src/api/publicCatalogue.ts — Frontend public catalogue client understands the v3 publicationAuthority, mediaAuthority, primaryImage, and browse-only checkout-handoff boundaries.
- src/services/orderService.ts — Direct public website checkout revalidates SKU lines against current storefront publishing state before creating orders.
- src/services/publicStorefrontBasketService.ts — Basket checkout revalidates line eligibility against current storefront publishing state before checkout handoff.
- scripts/public\_catalogue\_smoke\_tests.js — Public catalogue smoke coverage verifies explicit publishing visibility, hidden-by-default active products, browse-only non-checkout behavior, publicationAuthority, and server-owned ordering.
- scripts/storefront\_basket\_smoke\_tests.js — Storefront basket smoke coverage verifies checkout revalidation against publishing state.
- scripts/stripe\_website\_checkout\_smoke\_tests.js — Website checkout smoke coverage verifies direct checkout revalidation against publishing state.
- requests/online\_store\_product\_publishing.http — Manual request examples cover manager product publishing runtime controls.
- docs/support/web-builder.md — Support docs describe the runtime product publishing, product media, merchandising diagnostics, and launch-readiness boundaries.
- docs/web\_builder\_task\_queue.md — Task queue now records Task 43 as runtime-complete and keeps Clapham production launch rehearsal/sign-off blocked behind real launch evidence.
- docs/roadmap-events/2026-05-16-web-builder-storefront-product-publishing-runtime-foundation.json — Roadmap event records the Task 41 storefront product publishing runtime foundation.
- prisma/migrations/20260620120000\_storefront\_product\_media\_runtime/migration.sql — Task 42 adds durable storefront product media records for manager-approved product-card imagery.
- src/services/storefrontProductMediaService.ts — Storefront product media service validates uploads and external URLs, enforces public-safe approval plus alt text, and manages primary media selection.

- `src/controllers/storefrontProductMediaController.ts` — Storefront product media controller exposes manager-only product media handlers alongside the online-store product publishing surface.
- `frontend/src/pages/OnlineStoreProductsPage.tsx` — ``/online-store/products`` now includes product media controls for upload, external URL, review, public-safe state, alt text, focal point, primary selection, and removal.
- `frontend/src/features/webBuilder/GeneratedProductBrowser.tsx` — Generated-site product cards render approved primary images with fixed-aspect fallbacks for products without public media.
- `scripts/online_store_smoke_tests.js` — Online Store smoke coverage verifies product media upload, external URL validation, approval, primary selection, and deletion controls.
- `requests/online_store_product_publishing.http` — Manual request examples include product media upload, external URL, approval, blocking, primary selection, and deletion flows.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-product-media-runtime-foundation.json` — Roadmap event records the Task 42 product media runtime foundation.
- `src/services/webBuilderService.ts` — Task 43 adds manager-only generated-site storefront diagnostics for product availability, category readiness, featured-product exposure/suppression, product-media/fallback state, browse-only/checkout state, and Clapham launch blockers.
- `frontend/src/pages/WebBuilderPage.tsx` — Web Builder first-run checklist rows now summarize product, category, product-media, and launch-evidence readiness from generated-site diagnostics.
- `scripts/web_builder_v1_smoke_tests.js` — Web Builder smoke coverage verifies Task 43 diagnostics while asserting supplier cost and private smoke markers stay out of the response.
- `docs/roadmap-events/2026-05-16-web-builder-storefront-merchandising-diagnostics-launch-readiness.json` — Roadmap event records the Task 43 storefront merchandising diagnostics and launch-readiness milestone.

#### Remaining gaps (2)

- Task 43 now provides merchandising diagnostics and launch-readiness visibility, but exact stock quantities, direct carrier adapters, product/package-specific parcel rules, delivery promises, DNS/custom-domain launch, and broader e-commerce trust remain future work. high
- Storefront maturity should not be conflated with narrow checkout/payment groundwork. high

#### Recommended next tasks (5)

- Keep generated-site product browsing reuse constrained to the public catalogue contract, purchasable-only filtering, basket-token checkout handoff, and server-side estimates. medium
- Pilot the shipping-rate/address foundation with real Store Info origin readiness and EasyPost sandbox credentials before promising live courier services. high
- Do not start Clapham production launch rehearsal/sign-off until real generated-site review, widget trial, production-serving, deployment-owner, support-owner, rollback, and manager/admin evidence is available. high
- Sequence product/package parcel rules, direct carrier adapters, exact delivery promises, and broader post-purchase support as separate follow-on milestones. high
- Keep storefront readiness reporting separated into browse/catalogue, checkout, fulfilment, and post-purchase support layers. medium

#### Ideas and dependencies

**Ideas**

- Track storefront maturity by browse/catalogue, checkout, fulfilment, and post-purchase support.

**Dependencies**

- Web builder / templates / embeddable widgets
- Stripe / payment trust

## Google Business Profile / Reviews / Maps

Readiness, OAuth, read-only reviews, internal reply approvals, explicit approved-draft reply posting, and local operations/runbook diagnostics are real; this remains an internal Google presence foundation rather than a public/local-marketing product.

Medium

Foundation Exists

63%

**Target:** customer-facing, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against readiness UI, OAuth, read-only reviews, internal reply drafts, approved-draft posting hardening, operations diagnostics tests, and Google planning docs.

### Feature maturity

63%

#### Evidence links (14)

- `frontend/src/pages/GoogleBusinessProfileReadinessPage.tsx` — Readiness UI exists.
- `src/services/googleBusinessProfileOAuthService.ts` — OAuth groundwork exists.
- `src/services/googleBusinessProfileReviewInboxService.ts` — Read-only reviews inbox service exists.
- `src/services/googleBusinessProfileReviewReplyDraftService.ts` — Internal reply draft approval, approved-draft posting, failure classification, and duplicate/update-flow blocking service exists.
- `src/services/googleBusinessProfileOperationsStatusService.ts` — Local Google Reviews operations status service exists.
- `prisma/migrations/20260604120000_google_review_reply_drafts/migration.sql` — Durable reply draft history migration exists.
- `prisma/migrations/20260605120000_google_review_reply_posting/migration.sql` — Reply posting state metadata migration exists.
- `docs/support/google-business-profile-operations-runbook.md` — Manager operations runbook exists.
- `docs/google_business_profile_integration_plan.md` — A dedicated plan exists.
- `docs/google_business_profile_production_readiness_audit.md` — A production-readiness audit and controlled live-trial checklist exist.
- `scripts/google_business_profile_readiness_tests.ts` — Readiness tests exist.
- `scripts/google_business_profile_review_reply_drafts_tests.ts` — Reply draft service tests exist.
- `scripts/google_business_profile_review_reply_posting_tests.ts` — Approved-draft posting service and hardening tests exist.
- `scripts/google_business_profile_operations_tests.ts` — Operations status tests exist.

#### Remaining gaps (2)

- Bulk/automatic reply posting, posted-reply update/delete, public review display, review requests, Maps/Places, and local-marketing automation remain intentionally unimplemented. high
- This area should stay explicitly lower-maturity until customer-facing value is more than configuration readiness.

medium

#### Recommended next tasks (3)

- Run the first single-location controlled production live trial using the production-readiness audit before any broader Google feature work. **high**
- Keep approved-draft posting, future posted-reply update/delete, review requests, and public display as separate tracked states. **high**
- Plan any future posted-reply update/delete workflow as a separate explicit, audited, manager-approved milestone. **medium**

#### Ideas and dependencies

##### Ideas

- Add a dedicated Google presence maturity lane only after controlled posting has operational evidence or public local-marketing value becomes real.

##### Dependencies

- Web builder / templates / embeddable widgets
- Store Info and public-site readiness

## Rental / hire customer-facing flow

The first public rental slice now exists as a request-first, customer-safe enquiry flow with staff-side quote/response handling and an embeddable iframe widget. Authenticated customer accounts and staff-generated secure readback links can now show customer-safe summaries of staff-confirmed hire bookings, but the flow remains deliberately short of self-service confirmation, reservation, amendments, cancellation, payment, upload, or digital signing.

Medium

Foundation Exists

62%

**Target:** customer-facing, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against rental foundation pages, smoke coverage, and readiness docs.

### Feature maturity

62%

#### Evidence links (24)

- `frontend/src/pages/BikeHirePage.tsx` — Internal bike-hire page exists.
- `docs/rental_hire_readiness_gap_audit.md` — Gap audit exists.
- `scripts/bike_hire_foundation_smoke_tests.js` — Internal rental foundation has smoke coverage.
- `docs/customer_facing_web_builder_widget_architecture_plan.md` — Architecture plan keeps public rental behind internal hardening and frames the first public slice as request-first.
- `frontend/src/pages/PublicHireRequestPage.tsx` — Public rental request page exists.
- `frontend/src/pages/PublicHireRequestWidgetPage.tsx` — Embeddable rental / hire request widget exists at `/widgets/hire-request` and keeps the public flow request-first.
- `src/services/publicWidgetService.ts` — Public Widgets readiness includes rental/hire request URL, iframe snippet, request-only boundary checks, and safe limitations.
- `frontend/src/pages/PublicHireRequestStatusPage.tsx` — Public rental status lookup page exists.
- `frontend/src/pages/CustomerAccountRentalsPage.tsx` — Authenticated customer account rental pages show customer-safe staff-confirmed hire booking summaries without customer edit/payment/upload/signing controls.
- `frontend/src/pages/PublicHireBookingReadbackPage.tsx` — Secure rental readback page shows a token-scoped view-only hire booking summary without edit/payment/upload/signing controls.
- `src/services/customerAccountService.ts` — Customer account service maps rental bookings into customer-safe DTOs that omit internal ids, asset ids/tags, notes, diagnostics, evidence, payment refs, and workshop maintenance internals.
- `src/services/hireReadbackLinkService.ts` — Rental readback service hashes raw tokens, scopes them to one booking, expires/revokes links, and reuses customer-safe rental DTO shaping.
- `src/routes/customerAccountRoutes.ts` — Customer-account rental list/detail routes are authenticated and ownership-scoped.
- `src/routes/publicHireBookingRequestRoutes.ts` — Public hire readback endpoint validates token possession and returns customer-safe rental DTOs only.
- `scripts/bike_hire_customer_booking_request_intake_smoke_tests.js` — Public request intake plus staff response/quote flow smoke coverage exists.
- `scripts/bike_hire_readback_links_smoke_tests.js` — Secure rental readback smoke coverage verifies hash-only storage, create/list/revoke, token expiry, safe DTO exclusions, view counts, and no payment/accounting mutation.
- `scripts/customer_account_rental_visibility_smoke_tests.js` — Customer account rental visibility smoke coverage verifies own-booking access, cross-customer isolation, safe DTO exclusions, public status safety, and no payment/accounting mutations.

- e2e/workshop/workshop-public.spec.js — Public E2E covers secure rental readback page rendering and revoked-link safe messaging.
- e2e/rental/rental.spec.js — Rental E2E covers the iframe widget route, request-only wording, customer-safe confirmation, and status lookup link.
- scripts/workshop\_m13\_smoke\_tests.js — Public Widgets smoke coverage includes manager-only rental/hire request snippet/readiness metadata.
- docs/rental\_hire\_staff\_customer\_response\_quote\_flow.md — Staff response and quote workflow doc exists.
- docs/rental\_hire\_customer\_request\_status\_page.md — Public rental status lookup workflow doc exists.
- docs/rental\_hire\_customer\_booking\_visibility\_foundation.md — Customer rental booking visibility doc records the account-only, view-only boundary.
- docs/rental\_hire\_secure\_readback\_link\_foundation.md — Secure readback link doc records token handling, expiry, revocation, and no-self-service boundaries.

#### Remaining gaps (3)

- Customers still cannot confirm a rental themselves, reserve an asset, accept a quote live, pay a deposit, sign a waiver, upload evidence, cancel, reschedule, or manage the booking through self-service. **high**
- Customer-send automation remains future work; current secure readback links are generated and shared manually by staff. **medium**
- Manual communication remains the safe default because the rental-specific outbound-message and acceptance model is still intentionally conservative. **medium**

#### Recommended next tasks (2)

- Keep the public rental flow request-first until manual response, availability, waiver, and deposit boundaries are trusted enough for a later confirmation step. **high**
- Decide whether any later customer-send automation should reuse an existing safe communications subsystem or stay manual by default. **medium**

#### Ideas and dependencies

##### Ideas

- Track rental maturity as two linked streams: internal fleet operations and customer self-service.

##### Dependencies

- Rental / bike box hire internal platform
- Web builder / templates / embeddable widgets

## Bike-shop-specific

76% average maturity across 4 features.

69% trial

4 features

## Rental / bike box hire internal platform

Internal rental operations now cover booking, pricing, readiness, intake review, customer-response handoff, staff-only evidence references, versioned hire terms, staff-witnessed waiver acknowledgement records, staff-only reschedule/extension/early-return/asset-swap amendments, staff/manager-only deposit and damage outcome reviews, staff-only maintenance request links to existing internal workshop jobs, account-only customer-safe confirmed-booking visibility, staff-generated secure readback links, and manager pilot-readiness closeout decisions, but still need upload/public signing and broader operational finishing.

Medium

Built, Needs Hardening

85%

**Target:** operational, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against rental pages, services, docs, and smoke coverage.

### Feature maturity

85%

#### Evidence links (37)

- frontend/src/pages/BikeHirePage.tsx — Bike-hire staff UI exists.
- src/services/bikeHireService.ts — Bike-hire service exists.
- docs/rental\_hire\_internal\_foundation.md — Internal rental foundation doc exists.
- scripts/bike\_hire\_foundation\_smoke\_tests.js — Bike-hire smoke coverage exists.
- scripts/bike\_hire\_customer\_booking\_request\_intake\_smoke\_tests.js — Public request review, response, conversion, and diagnostics smoke coverage exist.
- docs/rental\_hire\_staff\_customer\_response\_quote\_flow.md — Staff response and quote flow doc exists.
- prisma/migrations/20260607120000\_rental\_hire\_photo\_document\_evidence\_foundation/migration.sql — Staff-only rental evidence migration exists.
- src/services/hireEvidenceService.ts — Rental evidence service exists.
- scripts/bike\_hire\_evidence\_foundation\_smoke\_tests.js — Rental evidence smoke coverage exists.
- docs/rental\_hire\_photo\_document\_evidence\_foundation.md — Rental evidence foundation doc exists.
- prisma/migrations/20260608120000\_rental\_hire\_versioned\_waiver\_acknowledgement\_foundation/migration.sql — Versioned rental waiver acknowledgement migration exists.
- src/services/hireWaiverAcknowledgementService.ts — Rental waiver template and acknowledgement service exists.
- scripts/bike\_hire\_waiver\_acknowledgement\_smoke\_tests.js — Rental waiver acknowledgement smoke coverage exists.
- docs/rental\_hire\_versioned\_waiver\_acknowledgement\_foundation.md — Rental waiver acknowledgement foundation doc exists.
- prisma/migrations/20260609120000\_rental\_hire\_booking\_amendment\_foundation/migration.sql — Rental booking amendment history migration exists.
- scripts/bike\_hire\_booking\_amendments\_smoke\_tests.js — Rental amendment smoke coverage exists.
- docs/rental\_hire\_reschedule\_extension\_asset\_swap\_foundation.md — Rental amendment foundation doc exists.
- prisma/migrations/20260610120000\_rental\_hire\_deposit\_damage\_outcome\_review\_foundation/migration.sql — Rental deposit and damage outcome review migration exists.
- src/services/hireOutcomeReviewService.ts — Rental deposit and damage outcome review service exists.
- scripts/bike\_hire\_deposit\_damage\_outcome\_smoke\_tests.js — Rental deposit and damage outcome smoke coverage exists.

- docs/rental\_hire\_deposit\_damage\_outcome\_review\_foundation.md — Rental deposit and damage outcome review foundation doc exists.
- prisma/migrations/20260611120000\_rental\_hire\_workshop\_maintenance\_request\_foundation/migration.sql — Rental maintenance request / workshop link migration exists.
- src/services/hireMaintenanceRequestService.ts — Rental maintenance request service exists.
- scripts/bike\_hire\_maintenance\_requests\_smoke\_tests.js — Rental maintenance request smoke coverage exists.
- docs/rental\_hire\_workshop\_maintenance\_job\_request\_foundation.md — Rental maintenance request foundation doc exists.
- prisma/migrations/20260612120000\_rental\_hire\_secure\_readback\_link\_foundation/migration.sql — Rental secure readback token migration exists.
- src/services/hireReadbackLinkService.ts — Rental secure readback link service exists.
- scripts/bike\_hire\_readback\_links\_smoke\_tests.js — Rental secure readback link smoke coverage exists.
- docs/rental\_hire\_secure\_readback\_link\_foundation.md — Rental secure readback link foundation doc exists.
- frontend/src/pages/CustomerAccountRentalsPage.tsx — Customer account rental list/detail pages provide account-only customer-safe readback for staff-confirmed bookings.
- frontend/src/pages/PublicHireBookingReadbackPage.tsx — Public token-scoped rental readback page exists.
- scripts/customer\_account\_rental\_visibility\_smoke\_tests.js — Customer account rental visibility smoke coverage verifies safe read-only DTOs and ownership isolation.
- docs/rental\_hire\_customer\_booking\_visibility\_foundation.md — Customer rental booking visibility foundation doc exists.
- prisma/migrations/20260613120000\_rental\_hire\_pilot\_readiness\_closeout/migration.sql — Rental pilot readiness decision migration exists. missing
- src/services/hirePilotReadinessService.ts — Rental pilot readiness service aggregates diagnostics into manager blockers, warnings, checklist, exception queues, and audit-only decisions.
- scripts/bike\_hire\_pilot\_readiness\_closeout\_smoke\_tests.js — Rental pilot readiness smoke coverage verifies manager-only access, decision audit, public boundary, and no financial mutation.
- docs/rental\_hire\_staff\_pilot\_readiness\_closeout.md — Rental staff pilot readiness closeout runbook exists.

#### Remaining gaps (3)

- Authenticated rental upload/download, customer-visible evidence, customer self-signing, public digital waiver signing, and public rental self-service remain intentionally unimplemented. high
- Operational polish, customer documents, dispute follow-up, internal fleet workshop-job creation policy, and later payment/accounting strategy are still incomplete. medium
- The internal platform is ahead of the customer-facing flow, which limits end-to-end commercial readiness. medium

#### Recommended next tasks (5)

- Run a controlled staff pilot and use pilot-readiness decisions, exception queues, and the runbook to decide what remains unsafe before wider use. high
- Track rental maturity by sub-capability: booking, amendments, pricing, agreement pack, return/condition, evidence, deposit/damage outcomes, maintenance requests, request handoff, and manual customer follow-up.

high

- Define a dedicated internal/fleet workshop-job pattern before allowing rental maintenance requests to create workshop jobs directly. **medium**
- Plan rental uploads and customer self-signing as separate explicit milestones with authenticated storage, legal retention rules, and no public leakage. **high**
- Add a clearer internal-vs-public maturity split in docs and reports. **medium**

#### Ideas and dependencies

##### Ideas

- Expose rental sub-feature maturity on the dashboard once the base tracker is established.

##### Dependencies

- Rental / hire customer-facing flow
- Managed printing / labels / print agent

## Service reminders

Service reminders are implemented for staff review, customer account bike detail pages show read-only reminder readiness using safe service history, active service schedules, and contact permissions, and the manager communication queue now adds no-send post-service follow-up readiness, operations ageing, fixed ageing-policy guidance, governance policy, internal decision records, durable preview/copy events, manual draft-preview assistance, manual outcome logging, gap filters, and dashboard attention surfacing for completed jobs. Outbound reminder and follow-up delivery remain intentionally unimplemented.

Medium

Built, Needs Hardening

75%

**Target:** operational, customer-facing, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against reminders page, service, and smoke coverage.

### Feature maturity

75%

#### Evidence links (13)

- `frontend/src/pages/ServiceRemindersPage.tsx` — Service reminders page exists.
- `src/services/reminderCandidateService.ts` — Reminder candidate service exists.
- `src/services/customerAccountService.ts` — Customer account bike detail responses include conservative reminder readiness without exposing reminder candidates or raw schedule records.
- `frontend/src/pages/CustomerAccountBikeDetailPage.tsx` — Customer bike detail UI shows service reminder readiness and profile/book actions.
- `src/services/reports/customerReports.ts` — Customer reports include post-service follow-up readiness and operations reporting, manager-only decision/outcome history, durable preview/copy events, ageing buckets, priority/deferred/stale item lists, gap filters, and gated manual preview text for completed jobs, contact preferences, duplicate follow-up evidence, and conservative staff actions.
- `frontend/src/pages/ManagerCommunicationQueuePage.tsx` — Manager communication queue surfaces post-service readiness, operations ageing, internal decision recording, durable preview/copy history, manual preview preparation, print-pack controls, and manual outcome logging with a no-message/no-review-request boundary.
- `docs/post_service_follow_up_remaining_work_plan.md` — Remaining-work plan now marks durable preview/copy evidence, reporting integration, history UI, gap filters, runbook polish, print-pack controls, and management dashboard surfacing as completed no-send foundations.
- `docs/post_service_follow_up_ageing_policy.md` — Ageing policy documents the current fixed post-service follow-up stale and bucket interpretation so reminder/follow-up readiness does not silently become automation policy.
- `docs/customer_post_service_follow_up_policy.md` — Governance policy records that current channel permissions are manual staff signals only and that service follow-up, service reminder, review-request, operational, and marketing contact require separate future semantics before automation.
- `docs/support/customers-and-bikes.md` — Support docs distinguish reminder readiness and post-service follow-up preview/copy/outcome reporting from outbound automation.
- `scripts/service_reminders_m103_smoke_tests.js` — Reminder smoke coverage exists.
- `scripts/customer_reminders_m124_smoke_tests.js` — Customer reminders smoke coverage verifies follow-up readiness labels, operations ageing/counts, manager auth, decision/outcome validation/history, durable preview/copy events, preview gap filters, preview gating/safe text, internal-field exclusion, and no outbound side effects.
- `scripts/customer_account_identity_smoke_tests.js` — Customer account smoke coverage verifies reminder-readiness labels, ownership, contact-preference states, internal-field exclusion, and no outbound side effects.

### Remaining gaps (2)

- Dedicated opt-in/out semantics, suppression product data, and outbound delivery remain future work; governance policy now exists but is not a sending implementation. **medium**
- Customer-facing readiness is intentionally lightweight and does not yet provide customer-managed schedules or full service-plan editing. **medium**

### Recommended next tasks (3)

- Translate consent, suppression, review-request copy governance, Google separation, and approval policy into data requirements before any explicit staff-approved outbound policy is implemented. **high**
- Document what makes reminder and post-service follow-up readiness trusted enough to become explicit staff-approved outbound policy. **medium**
- Keep reminder and follow-up maturity tied to communication, consent, review-request separation, and customer account privacy checks before enabling any sends. **medium**

### Ideas and dependencies

#### Ideas

- Add a reminder confidence signal that captures channel readiness and follow-through quality.

#### Dependencies

- Communication Centre
- Customer bike lifecycle CRM

## Customer bike lifecycle CRM

Customers, bikes, bike-linked history, service schedules, customer-visible reminder readiness, manager-only post-service follow-up readiness, internal follow-up decisions, durable preview/copy history, staff-prepared manual preview assistance, manual outcome logging, follow-up operations ageing/reporting, fixed ageing-policy guidance, consent/suppression/review-request governance policy, gap filters, and dashboard attention surfacing clearly exist, but the deeper CRM operating model is still maturing.

High

Built, Needs Hardening

72%

**Target:** operational, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against customer bike services, pages, and timeline coverage.

### Feature maturity

72%

#### Evidence links (13)

- `src/services/customerBikeService.ts` — Customer bike service exists.
- `src/services/bikeServiceScheduleService.ts` — Bike-owned service schedule service exists for due tracking.
- `frontend/src/pages/CustomerAccountBikeDetailPage.tsx` — Customer account bike detail page shows customer-visible service reminder readiness.
- `frontend/src/pages/CustomerCommunicationQueuePage.tsx` — Manager communication queue shows post-service follow-up readiness, operations ageing, internal decision state, durable preview/copy history, manual preview preparation, gap filters, printable review-pack controls, and manual outcome logging for completed workshop jobs without sending messages or review requests.
- `frontend/src/pages/ManagementDashboardPage.tsx` — Management dashboard exposes a read-only post-service follow-up attention tile once durable preview/copy evidence is available.
- `docs/post_service_follow_up_remaining_work_plan.md` — Planning doc now records durable preview/copy event tracking, reporting integration, history UI, gap filters, runbook polish, print-pack controls, and dashboard surfacing as completed no-send foundations.
- `docs/post_service_follow_up_ageing_policy.md` — Post-service follow-up ageing policy defines fixed buckets, stale attention threshold, deferred/no-response handling, and future configurability constraints.
- `docs/customer_post_service_follow_up_policy.md` — Post-service follow-up governance policy separates service follow-up, service reminders, review requests, operational messages, and marketing contact while keeping outbound work blocked until product-data requirements exist.
- `frontend/src/pages/BikeHistoryPage.tsx` — Bike history page exists.
- `frontend/src/pages/CustomerTimelinePage.tsx` — Customer timeline UI exists.
- `scripts/customer_reminders_m124_smoke_tests.js` — Smoke coverage verifies completed-job follow-up readiness labels, operations ageing/reporting, contact-preference blockers, duplicate communication evidence, internal decision/outcome validation/history, durable preview/copy events, gap filters, draft-preview gating/safe text, and no-send side effects.
- `scripts/customer_account_identity_smoke_tests.js` — Customer account smoke coverage verifies bike reminder-readiness ownership, preference labels, and no-send side effects.
- `scripts/customer_timeline_m104_smoke_tests.js` — Customer timeline smoke coverage exists.

#### Remaining gaps (2)

- Lifecycle CRM value is present, but the repo does not yet expose a clear product-level maturity story for it.

medium

- Support and training material still emphasize individual pages more than the full customer-hike relationship

## Platform / Commercial

66% average maturity across 7 features.

62% trial

7 features

### Verification / multi-agent tooling

The repo has strong multi-worktree verification tooling, explicit merge-readiness guardrails, a merged canonical Codex submit wrapper, and a merged fast-gate watcher; this is one of the most mature internal platform areas.

High

Trial Ready

84%

**Target:** operational, trial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against verification wrappers, tests, and docs.

### Feature maturity

84%

#### Evidence links (7)

- scripts/verify\_parallel\_auto.sh — Automatic profile wrapper exists.
- scripts/verify\_parallel\_profile.sh — Manual profile wrapper exists.
- scripts/pre\_submit\_merge\_readiness.sh — Pre-submit merge-readiness guard exists.
- scripts/pr\_merge\_readiness\_final\_check.sh — PR-aware final merge-readiness guard exists.
- scripts/pr\_fastgate\_watch.js — PR fast-gate watcher exists.
- scripts/codex\_submit\_pr.sh — Canonical Codex PR submit wrapper exists.
- docs/PARALLEL\_VERIFY\_PROFILES.md — Parallel verification docs exist.

#### Remaining gaps (2)

- The verification model is strong internally, but the new submit/watch flow still needs repeated multi-agent usage and future trend-history depth before this area feels fully mature. medium
- Trial-ready here does not mean zero maintenance; it still needs careful visibility as repo processes evolve. low

#### Recommended next tasks (3)

- Track whether the merged submit/watch flow reduces stale-branch conflict and rerun churn across the next multi-agent batch. medium
- Keep the single-page dashboard aligned with verification workflow changes so maturity stays visible without manual repo spelunking. medium
- Keep test coverage close to any future workflow changes so maturity stays earned. medium

#### Ideas and dependencies

##### Ideas

- Surface verification maturity and repo process trust as a reusable internal KPI

- Surface verification maturity and repo-process trust as a reusable internal RFI.

### Dependencies

- Diagnostics and support-pack readiness
- Deployment / production serving

## Stripe/payment trust

Stripe foundations are deep and clearly implemented. Storefront payment-received email now carries customer-safe receipt/invoice summary content for paid web orders and supports explicit manager resend/failure review without changing payment/accounting truth, but operational trust is still a separate maturity step.

High

Active Hardening

76%

**Target:** operational, trial, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against Stripe routes, public status UI, docs, and extensive test coverage.

### Feature maturity

76%

#### Evidence links (6)

- `src/routes/onlinePaymentRoutes.ts` — Online payment routes exist.
- `src/routes/stripeWebhookRoutes.ts` — Stripe webhook routes exist.
- `frontend/src/pages/PublicPaymentStatusPage.tsx` — Public payment status page exists.
- `docs/online_payments_stripe.md` — Stripe foundation is documented.
- `src/services/onlinePaymentEmailService.ts` — Payment-received email delivery persists sanitized failure state and returns safe delivery status for storefront receipt/invoice resend responses.
- `scripts/stripe_website_checkout_tests.ts` — Stripe website checkout tests cover paid storefront receipt email content, duplicate webhook no-resend behavior, manual resend guardrails, and sanitized failure handling.

#### Remaining gaps (2)

- Production trust still depends on conservative rollout, observability, and support clarity rather than feature absence. high
- Customer-facing checkout and refund experiences should stay clearly separated from internal accounting maturity. medium

#### Recommended next tasks (2)

- Keep Stripe maturity split between implementation breadth and rollout trust in the dashboard summary. high
- Add a concise list of remaining operational confidence tasks for online payments. medium

#### Ideas and dependencies

##### Ideas

- Track payment trust by customer checkout, internal review, refund handling, and reconciliation confidence.

#### **Dependencies**

- QuickBooks / accounting trust
- Customer-facing storefront

## QuickBooks/accounting trust

QuickBooks foundations are strong, but they are intentionally conservative and still best described as trust-building hardening work.

High

Active Hardening

74%

**Target:** operational, trial, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against QuickBooks pages, services, smoke tests, and support docs.

### Feature maturity

74%

#### Evidence links (4)

- `frontend/src/pages/QuickBooksConnectionPage.tsx` — QuickBooks connection UI exists.
- `frontend/src/pages/QuickBooksAutomationPage.tsx` — QuickBooks automation UI exists.
- `src/services/quickBooksAutomationService.ts` — QuickBooks automation service exists.
- `scripts/quickbooks_manual_posting_smoke_tests.js` — QuickBooks posting smoke coverage exists.

#### Remaining gaps (2)

- Manager trust, review UX, and production trial confidence still need careful hardening before broader rollout. high
- Accounting maturity is broad enough that summary reporting needs to stay explicit about what is conservative by design. medium

#### Recommended next tasks (2)

- Expose QuickBooks trust maturity as review-led, not just feature-count complete. high
- Keep linking dashboard maturity to trial validation, evidence review, and stored-value reconciliation coverage. medium

#### Ideas and dependencies

##### Ideas

- Track accounting trust as a composite of connection, mapping, dry-run, posting, validation, and exception review.

##### Dependencies

- Stripe / payment trust
- Diagnostics and operational readiness

## Managed printing / labels / print agent

Printing foundations are meaningful and practical, but still operationally sensitive enough to keep under hardening.

Medium

Built, Needs Hardening

66%

**Target:** operational, trial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against print-agent source, managed print routes, docs, and smoke coverage.

### Feature maturity

66%

#### Evidence links (4)

- `print-agent/src/server.ts` — Print-agent server exists.
- `src/routes/managedPrintJobRoutes.ts` — Managed print routes exist.
- `docs/windows_print_agent.md` — Windows print-agent doc exists.
- `scripts/print_queue_smoke_tests.js` — Print queue smoke coverage exists.

#### Remaining gaps (2)

- Printer/device support and operational troubleshooting are still more foundation-grade than fully normalized.  
medium
- Printing readiness should stay linked to deployment/runtime confidence, not treated as isolated automation.  
medium

#### Recommended next tasks (2)

- Expose print-agent evidence and remaining setup friction directly in the dashboard outputs. medium
- Document the practical difference between print foundation and fully managed printing trust. medium

#### Ideas and dependencies

##### Ideas

- Track printing maturity by device support, dispatch reliability, and operator recovery steps.

##### Dependencies

- Deployment / Windows / Caddy / production serving

## Multi-location / HQ

HQ and multi-location foundations are real, but the roadmap still treats broader operational readiness as near-term future work.

Medium

Foundation Exists

54%

**Target:** operational, commercial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against HQ pages, routes, and smoke coverage.

### Feature maturity

54%

#### Evidence links (4)

- frontend/src/pages/HqOverviewPage.tsx — HQ overview UI exists.
- frontend/src/pages/HqUserLocationAccessPage.tsx — User/location access UI exists.
- src/routes/hqRoutes.ts — HQ routes exist.
- scripts/hq\_foundation\_smoke\_tests.js — HQ foundation smoke coverage exists.

#### Remaining gaps (2)

- Broader multi-store workflow depth is still limited beyond the current HQ groundwork. **high**
- Single-store safe defaults are good, but multi-store operating maturity is not yet broad enough to over-credit. **medium**

#### Recommended next tasks (2)

- Keep multi-location maturity explicitly below customer-facing and accounting scope claims until more workflows are location-safe. **high**
- Add clearer evidence links for what is HQ-ready versus still single-store-first. **medium**

#### Ideas and dependencies

##### Ideas

- Track multi-location maturity by governance, visibility, workflow safety, and commercial rollout depth.

##### Dependencies

- Permissions and sensitive action hardening
- Customer-facing storefront

## HR personnel foundation add-on

The HR bolt-on now has personnel files, a sanitized admin review queue, a candidate/offer flow, hash-only public offer acceptance links, offer lifecycle event snapshots, regenerated-link/withdrawal controls, new-starter capture, draft employee-profile conversion, and an internal contract snapshot/signing evidence foundation. Payroll, public evidence upload, external e-sign, operational activation, and legal/compliance automation remain future work.

Medium

Foundation Exists

53%

**Target:** commercial

**Last reviewed:** 2026-05-18. **Source:** Repo audit against HR plan, schema, admin/public API/UI, review-summary endpoint, document vault, contract signing foundation, support docs, and smoke coverage.

### Feature maturity

53%

#### Evidence links (19)

- docs/hr\_system\_plan.md — HR bolt-on plan records the hiring journey, compliance cautions, milestone sizing, payroll boundary, and recommended first slice.
- prisma/migrations/20260616120000\_hr\_foundation/migration.sql — Employee profile and HR document metadata schema exists.
- prisma/migrations/20260621120000\_hr\_candidate\_offer\_foundation/migration.sql — Candidate/offer schema stores pre-employment records and hash-only public token metadata.
- prisma/migrations/20260622120000\_hr\_offer\_lifecycle\_events/migration.sql — Offer lifecycle schema stores append-only offer event snapshots and withdrawal reasons.
- prisma/migrations/20260623120000\_hr\_contract\_signing\_foundation/migration.sql — Contract schema stores versioned templates, generated snapshots, document hashes, hash-only signing tokens, signer evidence, and append-only contract events.
- src/routes/hrRoutes.ts — Admin-only HR API routes include candidate, offer, new-starter, conversion, contract template/signing, employee, document, and review-summary endpoints.
- src/routes/publicHrRoutes.ts — Public HR API routes provide token-scoped offer decision, new-starter submission, and contract signing endpoints.
- src/services/hrService.ts — HR service handles profile validation, links, private document storage, checksums, audit events, and sanitized expiry/missing-basics review summaries.
- src/services/hrCandidateService.ts — Candidate service handles offer drafts, email attempts, resend/regenerate/withdraw lifecycle events, hash-only public links, public acceptance, new-starter capture, and draft employee conversion with explicit no-payroll/no-signature boundaries.
- src/services/hrContractService.ts — Contract service handles template rendering, immutable contract snapshots, document hashing, signing-link delivery, public signature evidence, event snapshots, and no-activation boundaries.
- frontend/src/pages/PersonnelPage.tsx — Admin-only Personnel page manages profiles, customer links, HR document upload/archive, and the review queue.
- frontend/src/pages/HrCandidatesPage.tsx — Admin-only Candidates & Offers page handles pre-employment workflow, offer lifecycle actions, new-starter capture, contract generation, signing-link delivery, and contract evidence before staff accounts or customer links exist.
- frontend/src/pages/PublicHrOfferPage.tsx — Public offer page supports token-scoped accept/decline without exposing admin HR records.
- frontend/src/pages/PublicHrNewStarterPage.tsx — Public new-starter page captures private starter details for admin review.

- frontend/src/pages/PublicHrContractSigningPage.tsx — Public contract-signing page supports token-scoped review and typed signature evidence without exposing admin HR records.
- scripts/hr\_foundation\_smoke\_tests.js — Focused HR smoke verifies access control, profile CRUD, user/customer links, document upload/download/archive, validation, and sanitized review-summary behavior.
- scripts/hr\_candidate\_offer\_smoke\_tests.js — Focused HR candidate smoke verifies admin-only access, token hashing, public accept/decline, regenerated-link invalidation, withdrawal revocation, offer event snapshots, one-use new-starter submission, and controlled conversion.
- scripts/hr\_contract\_signing\_smoke\_tests.js — Focused HR contract smoke verifies readiness blockers, template snapshots, hash-only signing links, no-store public preview, one-use signing evidence, event snapshots, withdrawal, and no operational activation.
- docs/support/personnel-and-hr-records.md — Support docs explain admin-only usage, review-queue limits, contract-signing evidence, and sensitive HR data boundaries.

#### Remaining gaps (2)

- External e-sign provider integration, public identity-document upload, right-to-work evidence approval workflow, leaver/disciplinary records, payroll exports, legal pay validation, and bonus calculations are not implemented. high
- Commercial rollout still needs HR-specific permissions, retention policy, encryption/key-management review, legal/compliance review, and operational support packaging. high

#### Recommended next tasks (2)

- Run Problem Cycle trial use against real candidate, offer, new-starter, and contract-signing needs before adding activation automation. high
- Decide whether the next HR slice should target operational activation hooks, HR permissions/retention hardening, or right-to-work evidence upload/review. high

#### Ideas and dependencies

##### Ideas

- Keep Problem Cycle-specific bonus calculation separate from the reusable commercial HR module unless a configurable framework is deliberately commissioned.

##### Dependencies

- Permissions and sensitive action hardening
- Diagnostics and support-pack readiness
- Email delivery and public token flows

## Support agent / staff help system

The support/help system is real enough to track, but still clearly early and intentionally constrained.

Medium

Foundation Exists

46%

**Target:** operational, trial

**Last reviewed:** 2026-05-15. **Source:** Repo audit against support-agent routes, services, UI, and docs.

### Feature maturity

46%

#### Evidence links (4)

- frontend/src/pages/SupportHelpPage.tsx — Staff help page exists.
- src/routes/supportAgentRoutes.ts — Support-agent routes exist.
- src/services/supportAgent/service.ts — Support-agent service exists.
- docs/support-agent-guardrails.md — Guardrails doc exists.

#### Remaining gaps (2)

- The support agent is still a guarded foundation rather than a mature daily-use assistant. medium
- Operational readiness depends on better curation of staff-safe help coverage and escalation boundaries. medium

#### Recommended next tasks (2)

- Track support-agent maturity by coverage, safety, and operator usefulness rather than mere route existence. medium
- Link help-system maturity to diagnostics and support-doc completeness in the dashboard summary. medium

#### Ideas and dependencies

##### Ideas

- Create a dedicated help-system maturity ladder for guarded tooling, staff docs, and escalation support.

##### Dependencies

- Diagnostics, support packs, and operational readiness
- Permissions and sensitive action hardening