



Developer Handoff

Last Updated: 3 July 2026, 13:59

Prepared for ThreeLochsCommunity
Project Three Lochs App

© We Are Affective Ltd 2026. All rights reserved.
Private and Confidential.

Contents

Introduction	3
Design Phase Deliverables	3
What Has Been Built	6
What Matters Most	8
Known Constraints & Decisions	12
System Copy Requirements	17
Acceptance Criteria	20
Emotional Criteria	20
Brand Criteria	21
Interaction Criteria	22
Copy Criteria	23
Performance Criteria	23
Pre-Launch Emotional Review	24
What It Covers	24
Who Is Involved	25
What It Produces	25
The Sign-Off Process	26
Post-Launch Review Framework	26

Introduction

This document exists at the handover point between two different kinds of work. The design phase answered questions about who this product is for, what it should feel like, and why each decision was made. The build phase answers different questions: how it gets constructed, tested, and shipped. What tends to get lost in that transition is not the functionality. Functional requirements travel well. What gets lost is intent. A button that works is not the same as a button that works and feels right. This document captures the layer between those two things, the reasoning behind the design, the emotional logic that shaped each decision, and the criteria by which the built product will be judged to have succeeded or fallen short.

This is not a functional specification. User stories, acceptance criteria, API requirements, and technical architecture live in separate documents and are maintained by the team best placed to own them. This is also not a project plan, a design system, or a QA checklist. Those things exist and they matter. This document does something none of them do: it explains what the product is trying to make a resident feel, and why that goal should be treated as a constraint rather than an aspiration. When a build decision needs to be made and the functional spec does not cover it, this document is where to look.

The development team, QA, and anyone responsible for translating design into a working product should read this before writing a line of code and return to it whenever a decision requires a judgement call. The design phase produced strategy documents, tone and voice guidelines, and visual direction, this document sits alongside all of them and gives them teeth in the build context. Where those outputs describe what Three Lochs is, this document describes what that means for the people building it. Every section is written with the end user in mind, because the resident on Waterside Walk who opens this app for the first time is the only meaningful judge of whether the build succeeded.

Design Phase Deliverables

The documents produced during the design phase do not function as separate artefacts. They form a system, and that system has a logic to it. The strategy work established who the resident is emotionally, what they carry into every interaction with the community, and what needs to shift. The brand and experience work translated that understanding into voice, visual language, and design principles. Together they describe not just what the product should do but what it should feel like to use, and why every decision that shaped it was made the way it was. No single document in this list tells the full story. Read as a set, they describe a product with a coherent emotional architecture from first principles to final pixel.

Developers should begin with the Design Principles and Voice and Tone documents before anything else. These two establish the logic that governs every judgement call in the build. The Visual Design

System and Component Library are the technical translation of that logic into buildable form. Everything else provides context that deepens understanding and prevents well-intentioned decisions from quietly contradicting the intent.

Document	Category	Primary Audience	Summary
Resident Experience Strategy	Research & Strategy	Design, Development, QA	The foundational strategy document. Defines the emotional reality of the Three Lochs resident today, the gap between expectation and experience, and the arc the product needs to move people along. Establishes the core problem the app solves and the principles that govern every decision downstream.
Emotional Journey Map	Research & Strategy	Design, Development, QA	Maps the resident's emotional state across six key lifecycle moments, from first open through sustained engagement. Identifies the specific points where trust is built, where it is most likely to break, and what the product must do at each stage to keep the arc moving in the right direction.
Aspiration Gap Report	Research & Strategy	Design, Product, Strategy	Documents the distance between how residents currently experience HOA administration and how they should. Captures the three resident archetypes who arrive at the app, the internal dialogue they carry, and the belief the product needs to install by the end of the first session. Required reading for anyone making decisions about onboarding.
Design Principles	Research & Strategy	Development, QA, Design	Five principles that govern every product decision: every loop opened must be closed; the app speaks before you are asked; every screen belongs to the resident; tone is the product; the first session must end with something real. Each principle includes a measurable signal and a practical description of what passing and failing look like in the build.
Voice and Tone Guidelines	Brand & Experience	Development, QA, Content	Defines the voice of Three Lochs precisely: calm, specific, ahead, assured, personal. Includes the formality register, sentence rhythm, approved phrases, and a definitive list of language that is never permitted in the product. Any string of copy that will appear in the built product should be checked against this document before it ships.

Document	Category	Primary Audience	Summary
Brand Identity System	Brand & Experience	Design, Development	Establishes the visual identity of Three Lochs, the loch-blue, slate, and natural tone palette, the typographic approach, the landscape imagery direction, and the logo and mark usage rules. Provides the foundation from which the Visual Design System was built.
Resident Persona Profiles	Brand & Experience	Design, Development, QA	Detailed profiles for the three resident types who arrive at the product: the new resident in the honeymoon period, the task-focused long-term resident, and the frustrated resident who has already been let down. Each profile describes emotional state on arrival, what they need from the interaction, and what a successful experience looks like for them specifically.
Copy and Microcopy Library	Brand & Experience	Development, QA, Content	A complete library of approved copy for every state and interaction in the product, confirmation messages, error states, empty states, notification copy, onboarding text, and task completion screens. This is the source of truth for any string that appears in the built product. Deviations require sign-off.
Visual Design System	Technical	Development, QA	The single source of truth for all visual components, spacing, colour tokens, typography scales, and interaction states. Produced in Figma and exported for development handoff. Every component is named, documented, and cross-referenced to the relevant design principle where applicable. Developers should work from this document for all UI implementation.
Component Library and States	Technical	Development, QA	Documents every component in the product across all states: default, hover, active, disabled, loading, error, and success. Includes behaviour specifications for transitions and animations. The animation and transition constraints in this document are not decorative; they carry emotional weight and are treated as non-negotiable.

Document	Category	Primary Audience	Summary
Notification Logic and Trigger Map	Technical	Development, QA	Specifies every notification type in the product, the trigger condition, the timing, the copy state, and the action it surfaces. Built directly from the design principle that the app speaks before the resident has to ask. Any notification that is not in this document has not been designed and should not be shipped without a design review.
Accessibility Specification	Technical	Development, QA	Documents the accessibility requirements for the product across contrast ratios, tap target sizing, screen reader labelling, and motion reduction support. These are requirements, not guidelines. The product does not ship without meeting them.
QA Acceptance Criteria	Technical	QA, Development	Translates the design principles into testable criteria. Includes both functional acceptance criteria and experience criteria, the specific emotional states and behavioural outcomes the product must produce to be considered complete. QA sign-off requires both sets of criteria to be met.

What Has Been Built

Three Lochs is a community, not a platform. The brand carries the character of a well-run place that takes quiet pride in itself: calm, assured, personal, and always one step ahead. It speaks the way a general manager of a small exceptional hotel speaks, with warmth that does not tip into familiarity and authority that does not need to announce itself. Nothing about it is generic. Nothing about it performs.

The resident is a homeowner who made a significant investment and quietly expects the community behind it to match the quality of the place itself. What they fear most is being ignored: submitting something and hearing nothing, paying dues and feeling like a debtor rather than a member, finding out something important from a neighbour before hearing it from the HOA. Success for them is frictionless and invisible. They complete what they came to do in under two minutes, receive confirmation that something has been handled rather than merely logged, and carry on with their day without having to think about the administration of where they live.

The emotional arc, First contact: the resident arrives braced. Every prior experience of HOA communication has trained them to expect friction, impersonal language, or silence. The first sixty seconds of the app must interrupt that expectation completely. Their name, their address, their neighbourhood, their dues status, and what is happening in the community today, all present before

follow up. The cumulative effect of these moments is trust, earned incrementally through every closed loop and every proactive notification., After sustained use: they stop thinking about the HOA as an entity they have to manage. They say "we" when they talk about Three Lochs. The app has moved them from payer to member, from someone who owns a house in the community to someone who lives in it.

Critical moments, The home screen on first open. This is where the prior expectation either changes or does not. The screen must show the resident's name, their neighbourhood, their dues status, and something happening in the community that is relevant to them today. A generic dashboard here confirms every suspicion they arrived with. A personalised one does something no onboarding flow can manufacture., The first task completion. Not the setup, the first thing they actually do, pay dues, book the pool, check an announcement. The confirmation screen at the end of that task is where the product's personality lands hardest. "All sorted for June" versus "transaction confirmed" are not two versions of the same thing. They produce entirely different beliefs about what kind of product this is., The first maintenance request. This is the most emotionally loaded interaction in the product. A resident submits something they care about and waits. What happens next either proves the product works or confirms what they already suspected. A named person and an expected completion date within the hour is passing. A reference number and silence is not., The first resolved notification. The moment an unprompted notification arrives telling the resident their reported issue has been fixed is the turning point in the relationship. It closes the loop they were bracing to chase. Every future interaction is easier because of it., Any moment involving money or rule enforcement. Dues reminders, fee changes, violation notices, and arrears communications are emotional landmines. The resident who receives a dues reminder in administrative language after they already paid will not forgive it quickly. Every financially adjacent interaction must be handled with the specific care the Voice and Tone Guidelines describe.

Design principles, Every loop opened must be closed. Nothing submitted disappears. Every request receives a named owner, a timeframe, and a resolution notification when the work is done. The measurable signal: zero maintenance requests older than seven days without a visible status update., The app speaks before you have to ask. Proactive communication is not a feature. It is the primary emotional signal that someone is paying attention. Dues confirmations, pool closures, maintenance updates, and community news all arrive before the resident has thought to look for them., Every screen belongs to the resident, not the organisation. Navigation is organised around what residents do, not around the board's administrative categories. If a screen does not answer a question a resident actually has, it does not belong on their device., Tone is the product. A notification that completes its function in bureaucratic language has still failed. Every string of copy in the product is subject to the Voice and Tone Guidelines. Warm, specific, calm, and personal are not descriptors. They are requirements., The first session must end with something real. Onboarding is not complete when the account is set up. It is complete when the resident has done something, made a payment, saved an event, submitted a preference, and received a confirmation that felt like a close rather than a receipt.

What Matters Most

These are not preferences. They are not stretch goals or design aspirations that can be descoped when the build gets tight. The items in this section are load-bearing. Remove or compromise any one of them and the emotional architecture of the product does not degrade gracefully. It collapses at exactly the moment a resident needs it most. The test for whether something belongs on this list is simple: if it fails, does the product still deliver on its core promise? For each item below, the answer is no.

Non-Negotiable	What It Is	Why It Matters	What Happens If It Is Compromised
Personalised home screen on first open	The home screen must display the resident's first name, their property address, their neighbourhood, their current dues status, and at least one piece of community information relevant to them today, before any interaction has taken place.	This is the single moment that either interrupts the resident's prior expectation of HOA administration or confirms it. Every workshop session returned to the same finding: residents arrive braced. A generic dashboard tells them nothing has changed. A personalised screen, one that already knows who they are and what matters to them today, is the first proof that this product is different. That proof cannot be deferred to session two. It has to land in the first sixty seconds or the foundation of trust is never laid.	If the home screen is generic, or if personalisation is delayed behind onboarding steps, the resident's prior expectation holds. They file the app in the same mental category as every other HOA portal they have encountered. Recovering from that first impression requires multiple successful subsequent interactions. Some residents will not give the product enough sessions to recover. The new resident, the easiest person to win, is the most likely casualty of a personalisation failure here.

Non-Negotiable	What It Is	Why It Matters	What Happens If It Is Compromised
<p>Closed-loop maintenance notifications</p>	<p>Every maintenance request must result in a named assignee and an estimated completion date within the hour of submission, a status notification at each stage of progress, and an unprompted resolution notification when the work is complete.</p>	<p>The maintenance request flow is where the product is most directly tested against the reason it was built. The workshops identified the same failure mode repeatedly: submission followed by silence. A resident who submits something and hears nothing does not conclude that the system is slow. They conclude that nobody received it. That assumption, hardened across multiple interactions, produces the resigned, low-expectation resident who stops engaging and goes back to calling the office. Closing the loop, visibly, specifically, with a name and a resolution, is the behaviour that separates a concierge from a portal. It is also the turning point in the resident relationship: the first resolved notification is the moment the product stops being a promise and becomes a proof.</p>	<p>If the loop stays open, either because resolution notifications are not triggered, because status updates are not pushed, or because the assignee field defaults to a team rather than a person, the product fails at its most critical test. A reference number in place of a name tells the resident their problem has been logged. It does not tell them it will be fixed. Those are completely different feelings. A resident who submits a request, receives a ticket number, and hears nothing for seven days has not had a neutral experience. They have had a confirming one. It confirms everything they expected before they opened the app.</p>

Non-Negotiable	What It Is	Why It Matters	What Happens If It Is Compromised
<p>Voice and tone applied to every built string</p>	<p>Every piece of copy in the product, including confirmation messages, error states, empty states, notification text, and onboarding copy, must comply with the Voice and Tone Guidelines before it ships. No string is exempt.</p>	<p>Tone is not a layer applied over the product. It is the product. Two notifications carrying identical information produce entirely different emotional outcomes depending on how they are written. "Your payment has been processed" and "All sorted for June" are functionally equivalent and emotionally opposite. The first reminds the resident they are an account. The second closes the interaction with warmth and gives them their time back. The brand is built on the premise that Three Lochs speaks like the general manager of a small exceptional hotel: calm, specific, personal, and always one step ahead. That voice is not a marketing decision. It is a design constraint. Any string that sounds like it was generated by a system or written for a procurement document breaks the spell instantly and permanently.</p>	<p>If default system copy ships without review, or if confirmation screens and error states are written by developers working without the tone guidelines, the product fractures at the moments that matter most. Task completion screens, error messages, and notification copy are precisely where the product's personality lands hardest. A resident who completes a dues payment and reads "transaction confirmed" has not had the experience the product was designed to deliver. They have had a generic one. Generic is indistinguishable from every platform Three Lochs is specifically positioned against.</p>

Non-Negotiable	What It Is	Why It Matters	What Happens If It Is Compromised
<p>Proactive notification triggers</p>	<p>The notification system must send status updates, reminders, and confirmations before the resident would think to look for them. Dues confirmations must arrive before the due date, not after a missed payment. Maintenance updates must be pushed at each stage without the resident opening the app. Amenity closures must be communicated the morning they occur, not when a resident arrives at a locked gate.</p>	<p>Proactive communication is the primary signal that someone is paying attention on the resident's behalf. The Heartbreak Scale sessions were unambiguous on this point: silence after submitting anything is the single biggest trust-killer in the current experience. Residents fill the vacuum with worst-case assumptions. The app's job is to make that vacuum impossible. A notification that arrives before the resident has formed the question does something no amount of warm design can replicate: it demonstrates, through behaviour rather than language, that Three Lochs is already one step ahead. That demonstration is what converts a task-completing product into a trusted one.</p>	<p>If notification triggers are delayed, batched, or only sent in response to resident action, the product behaves like a portal rather than a concierge. The resident who checks the app to see whether their request was received has already been let down. The resident who opens the gate to find the pool closed without warning has had an experience that no subsequent notification can fully repair. Reactive communication is the current standard. The product was built specifically to replace it. Shipping a reactive notification system inside a product that promises to be proactive is not a partial success. It is a direct contradiction of the core promise.</p>

Non-Negotiable	What It Is	Why It Matters	What Happens If It Is Compromised
<p>Dues and payment copy framed as membership, not billing</p>	<p>Every financially adjacent interaction, including dues reminders, payment confirmations, autopay setup, and arrears communications, must use the language and framing defined in the Copy and Microcopy Library. Dues are membership. Payment is confirmation that everything is in order. Nothing in the payment flow uses billing, outstanding, debtor, or compliance language.</p>	<p>Residents of a premium community feel the language of debt collection viscerally. The workshops returned to this repeatedly. A dues reminder that arrives in administrative language, even for a resident who always pays on time, changes the emotional register of the entire interaction. It reframes the resident as a payer rather than a member. That reframing undoes something the product has worked hard to build. The belief the product needs to install, "I am a member of something well-run, and that reflects well on me", is incompatible with a payment experience that feels like a utility bill. The copy in payment flows is not incidental. It is where the membership framing either holds or fails.</p>	<p>If payment copy defaults to system language, or if the autopay setup flow reads as a billing configuration rather than a membership confirmation, the emotional architecture breaks at the most frequent touchpoint in the product. Dues are paid monthly. If that interaction feels like a transaction every time, no amount of warm notification copy or personalised home screens can compensate. The cumulative effect of twelve monthly payment interactions written in billing language is a resident who, despite a functional app, still feels like an account holder. That is the exact identity the product was built to replace.</p>

Known Constraints & Decisions

The decisions documented in this section were made during a design phase that involved structured workshops, instinct-led exercises, and considered conversation between We Are Affective and the ThreeLochsCommunity team. Each decision has context behind it: a resident emotional state it is responding to, a trust failure it is preventing, or a brand position it is protecting. That context does not travel automatically into the build phase. This document carries it forward deliberately.

Revisiting these decisions without that context is not neutral. A developer who adjusts a copy pattern for convenience, simplifies an interaction to reduce build complexity, or defaults to system-generated language for an edge case is not making a small technical choice. They are making an emotional one, on behalf of a resident who will feel the consequence. The table below names the decisions, explains why they were made, and describes specifically what not to do. Where a decision

Decision	Rationale	What Not To Do
<p>The home screen is personalised on first open, showing the resident's name, neighbourhood, dues status, and a relevant piece of community information before any interaction</p>	<p>Residents arrive carrying every bad HOA experience they have ever had. The first sixty seconds either interrupts that expectation or confirms it. Personalisation at this moment is not a feature; it is the first proof that this product is different. A generic dashboard tells the resident nothing has changed.</p>	<p>Do not build a default home screen that shows community-wide information and surfaces personalisation only after onboarding is complete. Do not defer the resident's name, address, or dues status behind a loading state or a navigation step. Do not use property ID or account number anywhere a name could appear.</p>
<p>Maintenance requests must return a named assignee and an estimated completion date, never a reference number alone</p>	<p>A ticket number tells a resident their problem has been logged. A name and a date tells them it will be fixed. Those are completely different feelings. The workshops identified silence after submission as the single biggest trust-killer in the current resident experience. A named person makes accountability specific and visible.</p>	<p>Do not allow the confirmation state of a maintenance request to default to "your request has been received, reference TL-XXXX." Do not use team names ("a member of the facilities team") where a person's name should appear. Do not ship a default state that shows an open ticket with no assignee and no estimated date.</p>
<p>Every loop opened by a resident must be closed with an unprompted resolution notification</p>	<p>Submission and resolution are two separate events. The product's job is to connect them visibly, before the resident has to ask. The first time a resident receives an unprompted notification telling them their issue has been resolved is the moment the relationship changes. It does not just close one loop. It retires the anxiety around every future interaction.</p>	<p>Do not mark a maintenance ticket as resolved in the admin dashboard without triggering a resident-facing notification. Do not rely on the resident opening the app to discover that something has been fixed. Do not batch resolution notifications or delay them for operational convenience.</p>

Decision	Rationale	What Not To Do
<p>All notification copy must be proactive, not reactive</p>	<p>A notification that arrives after the resident has already formed the question has failed. Dues confirmations must arrive before the due date. Pool closures must be communicated the morning they occur. Maintenance updates must be pushed at each stage without the resident initiating. This is the primary behavioural signal that someone is paying attention on the resident's behalf.</p>	<p>Do not build a notification system that only fires in response to resident action. Do not send amenity closure notifications after the closure has begun. Do not batch proactive updates into a daily digest. Do not treat push notification delivery as a nice-to-have that can be scoped out under time pressure.</p>
<p>Every string of copy in the built product must comply with the Voice and Tone Guidelines before it ships</p>	<p>Tone is not applied over the top of the product. It is the product. "Transaction confirmed" and "All sorted for June" are functionally identical and emotionally opposite. The first reminds the resident they are an account. The second closes the interaction with warmth and returns their time to them. Confirmation screens, error states, empty states, and notification copy are precisely where the product's personality lands hardest.</p>	<p>Do not ship default system copy on any state without a tone review. Do not use passive voice in any resident-facing string. Do not use the following under any circumstances: "please be advised," "your request has been logged," "reference number," "a member of our team will be in touch," "outstanding balance," "failure to comply," "this is an automated message." Do not write error states as warnings.</p>
<p>Payment and dues interactions must be framed as membership confirmation, not billing</p>	<p>Residents pay monthly. If each of those twelve interactions per year feels like a utility bill, no amount of warm notification copy compensates. The emotional reframe, from payer to member, from obligation to confirmation that everything is in order, is incompatible with billing language. "All sorted for June" and "outstanding balance due" are not two tones applied to the same transaction. They describe two entirely different relationships.</p>	<p>Do not use "outstanding," "overdue," "balance due," "payment required," or "account in arrears" in any copy visible to a resident who has not yet missed a payment. Do not build the autopay setup flow as a billing configuration screen. Do not send a dues reminder to a resident who has already paid. Do not default to billing language in edge cases or error states on the grounds that it is more legally precise.</p>

Decision	Rationale	What Not To Do
<p>Visual language must carry the Three Lochs landscape identity on every screen, not just the home screen</p>	<p>Visual consistency is emotional consistency. A home screen that feels warm and personal, followed by a form that looks like it belongs to a different product, breaks the sense of being somewhere coherent. The palette, the typography, the spacing, and the imagery direction are not decorative choices. They are the mechanism by which every screen communicates that the resident is still inside Three Lochs, not inside generic software that Three Lochs happens to use.</p>	<p>Do not apply the brand palette to primary screens and revert to system defaults for utility screens, forms, or error states. Do not use generic icons, default loading spinners, or system alert dialogs that fall outside the component library. Do not treat the maintenance request flow, the payment flow, or any functional journey as a zone where brand standards are relaxed.</p>
<p>Navigation must be organised around resident tasks, not administrative categories</p>	<p>Every HOA platform on the market was built for the administrator. The information architecture, the default views, the labelling, all of it organised around what the organisation needs to manage. Three Lochs inverts this completely. If a screen does not answer a question a resident actually has, it does not belong on their device. Navigation labels must reflect what residents do, not what the board needs to track.</p>	<p>Do not organise primary navigation around administrative categories such as "tickets," "ledger," "compliance," or "accounts." Do not surface board-level metrics on resident-facing screens. Do not label the dues section "billing." Do not use internal system terminology in any label, heading, or navigation item visible to residents.</p>
<p>The first session must end with a completed action, not a setup confirmation</p>	<p>Onboarding is not complete when the account is configured. It is complete when the resident has done something real: made a payment, saved an event, confirmed a preference, or received a piece of information they did not have to ask for. The prior expectation the resident carries will not shift because the app looks good. It shifts because the app delivers something. One completed action in session one changes the baseline for every subsequent session.</p>	<p>Do not design an onboarding flow that ends on a "you're all set" screen with no action completed. Do not make the first meaningful resident action available only after a multi-step setup process. Do not treat the welcome experience as complete once account details have been entered. Surface something relevant to the resident's neighbourhood within the first sixty seconds, before they have had to search for it.</p>

Decision	Rationale	What Not To Do
<p>Sensitive communications, including rule violations, architectural rejections, and arrears notices, must always include a clear next step</p>	<p>Difficult news delivered without a path forward generates more anxiety than the news itself. A violation notice with no explanation and no appeal route, a fee increase with no budget context, an architectural rejection with no resubmission guidance: each of these is honest in content and harmful in form. The product must deliver truth calmly and always with the next step already visible.</p>	<p>Do not ship any communication that conveys a negative outcome without a clear, specific next action attached. Do not send violation or arrears notices in legal register when plain language is possible. Do not present an architectural decision as final without surfacing the review or appeal process. Do not send a fee increase notification without linking to the supporting budget information in the same message.</p>
<p>The visual density, energy level, and layout must remain calm and uncluttered across all screens</p>	<p>The Opposite Ends sessions produced clear and specific direction: low visual density (2), low energy (2), card-based layout (3), with generous spacing. This is not a discovery app or a social product. Residents open it to do something and close it. The interface should reflect that: present, calm, easy to leave once the task is done. Badges, counters, and engagement mechanics that belong to a different category of product do not belong here.</p>	<p>Do not introduce notification badges, streak counters, gamification elements, or social proof mechanics. Do not increase information density on any screen on the grounds that it is more efficient. Do not use high-energy animation, bright accent colours outside the defined palette, or interaction patterns borrowed from consumer social apps. Do not add supplementary content panels or suggested actions that extend session time beyond the resident's intent.</p>
<p>Sensitive resident data, including dues status, arrears, and maintenance history, must never be visible to other residents</p>	<p>The workshops identified surveillance as an emotional landmine. A resident who feels watched rather than served by how much the app knows about their property will disengage and will not return. The community noticeboard feature, the estate-wide maintenance visibility, and the resident directory all require careful scoping to ensure that what is visible is communal and what is private remains so.</p>	<p>Do not make individual dues status, payment history, or arrears information visible in any shared view. Do not surface a specific resident's maintenance requests in the community-wide ticket feed without stripping identifying information. Do not allow the resident directory to default to opt-in visibility without explicit, informed consent from each resident. Do not build any admin view that exposes private resident data in a context where it could be seen by other homeowners.</p>

System Copy Requirements

System copy is where brand voice is most often lost. Not because the intent was wrong, but because these strings get written late, under pressure, by whoever has access to the codebase at the time. Error messages get borrowed from other projects. Empty states get placeholder text that never gets replaced. Confirmation screens default to whatever the framework outputs. By the time QA sees them, they have been in the build long enough to feel fixed.

That cannot happen here. The Voice and Tone Guidelines are unambiguous: tone is the product. A maintenance confirmation that reads "your request has been logged, reference TL-2847" does not just sound wrong. It directly contradicts the core promise of the app. Every category of system copy in the table below requires review against the Copy and Microcopy Library before it ships. No exceptions, no placeholders in production, no strings that have not been touched by someone who has read the tone guidelines.

Who reviews system copy and when. All resident-facing copy, including every state in the table below, is reviewed by the Three Lochs content lead and signed off against the Copy and Microcopy Library before the component enters QA. Copy review is not a final-pass activity. It happens at component completion, before QA begins. Any string that reaches QA without sign-off is returned for review. Developers should not write resident-facing copy independently. The Copy and Microcopy Library contains approved strings for every state listed here. Use them as written. If a scenario arises that the library does not cover, raise it as a copy request before writing a fallback.

Category	Examples	Tone Guidance	Review Required
Confirmation messages	Dues payment confirmed, maintenance request submitted, amenity booking made, RSVP saved, profile updated	This is where the product's personality lands hardest. Every confirmation must feel like a close, not a receipt. "All sorted for June" not "payment confirmed." "Picked up, Sarah will be with you by Thursday" not "your request has been assigned." The resident should leave the screen feeling that something has been handled, not merely processed. Active voice, first names where a person is involved, specific dates not timeframes, short sentences.	Yes, before component enters QA

Category	Examples	Tone Guidance	Review Required
Error messages	Payment failed, booking unavailable, photo upload error, form validation, session timeout, connectivity loss	Errors must not alarm. They must be calm, specific about what happened, and immediately clear about what the resident should do next. No passive voice. No "an error has occurred." No apologetic filler without a resolution path. "That didn't go through, try again or contact us directly" is the register. Never a technical error code visible to the resident. Never a dead end.	Yes, before component enters QA
Empty states	No upcoming events, no maintenance requests, no bookings yet, empty notification feed, no neighbours in directory	Empty states are an opportunity, not a gap. A screen that simply says "nothing here yet" wastes the moment. The tone should be warm and forward-pointing: "No events this week, check back soon" or "You have not booked the pool yet, it is open until 8pm today." Never a blank screen. Never a message that implies the app is broken or the community is inactive.	Yes, before component enters QA
Loading states	App initialising, home screen loading, payment processing, image uploading, search in progress	Loading copy should be brief and human. "Getting your home screen ready" is the outer edge of acceptable. Most loading states should carry no copy at all beyond a progress indicator, because copy during a wait can feel like an excuse. Where copy is needed, it must not raise anxiety. Never "please wait" on a payment screen. Never anything that implies uncertainty about the outcome.	Yes, before component enters QA
Onboarding copy	Welcome screen, setup steps, permission requests, neighbourhood confirmation, first-session prompts	Onboarding copy sets the tone for the entire relationship. It must feel like an arrival, not a form. "Welcome to Three Lochs" is the right register. The resident's name and neighbourhood should appear as early as technically possible. Permission requests for notifications must explain the benefit in resident terms: "We will let you know when something needs your attention, before you have to ask." Never transactional. Never a list of features. The first session ends with something done, so the final onboarding screen should prompt action, not summarise setup.	Yes, before first session flow is built

Category	Examples	Tone Guidance	Review Required
Notification copy	Dues reminder, maintenance status update, event reminder, pool closure, community announcement, resolution confirmation	Notifications are the primary proactive signal that the app is paying attention on the resident's behalf. Every notification must arrive before the resident has formed the question it answers. Copy must be specific: a name, a date, a place on the estate. "Tom from the facilities team will be with you by Thursday morning" not "your request is being processed." Dues notifications must use membership framing: "All set for July, nothing to do until August." Pool closures must give the reopening date in the same sentence. Resolution notifications must thank the resident for reporting without sounding performative. Never a notification that ends on an open question.	Yes, before notification triggers are built
Success states	Payment complete, booking confirmed, request submitted, profile saved, event RSVP confirmed	Success screens are the moments residents carry away from the app. They inform the belief about what kind of product this is. The copy must feel warm and complete: "You are all set for the Birchwood summer social on Saturday" not "RSVP confirmed." "The lochside path report has been picked up" not "submission successful." Always a forward-facing close: what happens next, when the resident will hear back, or simply that there is nothing more they need to do. Never a success screen that just says "done."	Yes, before component enters QA
Help and support text	FAQs, contact prompts, rule explanations, document descriptions, tooltip copy, in-app guidance	Help copy must answer the question behind the question. A resident looking at the bylaws section is not browsing policy. They are anxious about whether they are allowed to do something. The tone must be reassuring and plain: "Not sure if your plans need board approval? Here is how to check." Never legal register when plain English is possible. Never a wall of text when two sentences will do. Contact prompts must name a person or a team, not a generic inbox. "Speak to the management team directly" is less trustworthy than "call Sarah at the management office."	Yes, before help content is populated

Category	Examples	Tone Guidance	Review Required
Rule violation and enforcement copy	Parking notices, architectural non-compliance alerts, noise complaint acknowledgements, arrears communications	These are the emotional landmines identified explicitly in the workshops. A violation notice that arrives without context, without a next step, or in legal language will damage the resident relationship regardless of whether the underlying issue is legitimate. Every enforcement communication must include a plain-language explanation of the issue, a specific and achievable next step, and a contact name for questions. Arrears copy must never use "outstanding balance," "failure to comply," or passive voice. It must open with the path to resolution, not with the consequence of inaction.	Yes, senior review required before any enforcement communication template is built

One rule applies across all of the above: if the Copy and Microcopy Library contains an approved string for a given state, use it exactly as written. If a new scenario arises during the build that the library does not cover, do not write a fallback independently. Raise it as a copy request. The library will be updated to cover it. A string written in the build phase without tone review is not a placeholder. Once it is in the product, it is the product.

Acceptance Criteria

Emotional Criteria

These criteria describe specific feelings the product must create at specific moments. QA cannot verify these through automated testing alone. Each one requires a structured review session with at least one person who has read the Emotional Journey Map and is not a member of the build team., [] A resident opening the app for the first time feels recognised before they have done anything. Their name, neighbourhood, dues status, and something relevant to today are visible on the home screen without any navigation., [] A resident who completes their first dues payment closes the confirmation screen feeling that something has been handled, not processed. The screen must not feel like a receipt., [] A resident who submits a maintenance request feels, within the hour, that a specific person has picked it up. The confirmation must produce confidence, not the low-level anxiety of wondering whether anyone received it., [] A resident who receives a resolution notification for a reported issue without having followed up feels the loop close. The notification must arrive unprompted and must not require them to open the app to discover the outcome., [] A resident navigating between the home screen and any task flow does not feel a shift in register. The warmth

distinguish clearly between informational communication and a demand., [] A resident opening the app during their first week feels they have arrived somewhere, not logged in somewhere. The welcome experience must feel like an arrival specific to Three Lochs and to their neighbourhood, not a generic setup flow., [] A resident who encounters an error does not feel alarmed. The error state must be calm, specific about what happened, and immediately clear about what to do next., [] A resident receiving a notification about a rule, a community change, or a fee adjustment does not feel threatened. Difficult news must arrive with the next step already visible., [] A resident who completes a session and closes the app feels their time was respected. No interaction should leave them feeling they spent longer than the task required.

Brand Criteria

These criteria confirm that the visual and verbal identity of Three Lochs has been applied consistently across the built product. Any deviation from the Visual Design System or Voice and Tone Guidelines is a build defect, not a design preference., [] The loch-blue, slate, and natural tone palette is applied correctly across all screens, including utility screens, forms, error states, and loading states. No screen defaults to system colours outside the defined palette., [] The landscape imagery direction is applied on the home screen and carried consistently through the product. Generic stock photography does not appear anywhere in the built product., [] Typography scales and weights match the Visual Design System exactly. No screen uses a system default font or a weight not defined in the design system., [] Spacing and breathing room match the Opposite Ends specification: low visual density, generous whitespace, card-based layout. No screen introduces additional information density on grounds of efficiency., [] The resident's first name appears wherever the design specifies it. Property ID, account number, and unit reference never appear in any resident-facing string., [] Every confirmation, notification, error, empty state, and success screen has been reviewed against the Copy and Microcopy Library and signed off by the Three Lochs content lead before entering QA., [] None of the following phrases appear anywhere in the built product: "please be advised," "your request has been logged," "reference number," "a member of our team will be in touch," "outstanding balance," "failure to comply," "this is an automated message," "for your records," "as per our records," "in due course," "do not hesitate to contact us.", [] Passive voice does not appear in any resident-facing string. Every sentence has a subject doing something., [] Named individuals appear in maintenance assignment notifications and support contacts. Team names do not substitute for person names in any context where the design specifies a named contact., [] Dues and payment copy uses membership framing throughout. The words "billing," "ledger," "debtor," "outstanding," and "overdue" do not appear in any copy visible to a resident who has not missed a payment., [] The tone register across all copy matches the Voice and Tone Guidelines specification: calm, specific, ahead, assured, personal. Any string that could have been written by any HOA platform is returned for rewrite.

Interaction Criteria

These criteria define specific interaction patterns that must work as designed. They are not aspirational. Each one has been specified in the Interaction Specifications document. Deviation from the specified behaviour is a defect., [] The home screen loads with the resident's personalised data visible before any scroll. Name, neighbourhood, dues status, and at least one relevant community item are present above the fold on all tested device sizes., [] Any common task, paying dues, booking an amenity, submitting a maintenance request, can be completed within three taps from the home screen. Tap-path analysis must confirm this for all three tasks before QA sign-off., [] The maintenance request flow accepts a photo upload, surfaces a named assignee and estimated completion date in the confirmation state, and triggers a push notification. All three must function correctly for the interaction to pass., [] The dues payment flow presents as a guided two-step sequence. The final screen uses the approved membership confirmation copy. The flow does not present as a billing form at any step., [] The amenity booking flow confirms availability in real time and surfaces the confirmation with the resident's name, the amenity, the date, and a forward-facing close. No booking confirmation screen ends without telling the resident what happens next., [] Push notifications open to the relevant screen when tapped. A maintenance update notification opens the request. A dues confirmation opens the payment history. An event reminder opens the event. Deep linking must function correctly for every notification type., [] Notification opt-in is presented during onboarding with copy that explains the resident benefit before asking for permission. The permission request does not use system-default language., [] The resident directory is presented as an opt-in feature with explicit, informed consent. A resident's information does not appear in the directory unless they have actively chosen to include it., [] Status updates on maintenance requests are pushed to the resident at each stage: received, assigned, in progress, resolved. None of these stages are visible only when the resident opens the app. All are pushed proactively., [] The community noticeboard surfaces estate-wide maintenance information with individual identifying details stripped. No resident's specific request, dues status, or property history is visible in any shared view., [] Transitions between screens use the animation specifications defined in the Component Library and States document. No screen-to-screen transition uses a system default that falls outside the specification., [] The onboarding flow ends with a completed action. The final onboarding screen prompts the resident to do something, not confirms that setup is complete. At least one of the following must be completable before the first session ends: dues setup, amenity booking, event RSVP, or notification preference confirmation., [] Error states on payment screens do not use language that implies the resident did something wrong. The error state is calm, specific, and resolves with a clear next step., [] Sensitive financial information, including dues history, arrears status, and payment amounts, is not accessible to any user other than the resident to whom it belongs and authorised board members in the admin view.

Copy Criteria

All system copy must pass tone review before the component it belongs to enters QA. The process below is not optional and does not run in parallel with QA. Copy review happens first.

Review process, [] The Three Lochs content lead holds a copy of the Voice and Tone Guidelines and the Copy and Microcopy Library at all times during the build phase., [] Every resident-facing string is submitted for review at component completion, before QA begins. Strings submitted after QA has started are returned and the component is held., [] Strings with an approved equivalent in the Copy and Microcopy Library are used exactly as written. No paraphrasing, no editorial adjustment by the build team., [] Strings covering scenarios not in the library are raised as copy requests. A fallback is not written independently. The library is updated to cover the scenario before the string ships., [] Enforcement and violation copy requires a senior review in addition to content lead sign-off. No enforcement template is built without both approvals.

Copy categories requiring review, [] Confirmation messages across all flows: payment, booking, maintenance submission, RSVP, profile update., [] Error messages across all flows: payment failure, booking unavailable, upload error, form validation, session timeout, connectivity loss., [] Empty states: no events, no bookings, no requests, empty notification feed, empty directory., [] Loading states: any screen where loading copy is specified in the design., [] Onboarding copy: all steps from welcome through to first-session action prompt., [] Notification copy: all types listed in the Notification Logic and Trigger Map document., [] Success states across all flows., [] Help and support text: FAQs, contact prompts, rule explanations, tooltip copy., [] All enforcement and violation communication templates.

Performance Criteria

These thresholds are not benchmarks. They are minimum requirements. Each one is set at the point where a delay or failure breaks the emotional pacing of the experience. Below these thresholds the product functions. At or above them the product delivers on its promise., [] The home screen loads personalised resident data within 2 seconds on a standard 4G connection. A personalised home screen that loads slowly is more damaging than a fast generic one: it raises and then delays the first moment of recognition., [] Maintenance request confirmation, including assignee name and estimated completion date, appears within 3 seconds of submission. A delay here lands in exactly the silence the product was built to eliminate., [] Push notifications for maintenance status updates are delivered within 60 minutes of the status change in the admin system. Notifications delayed beyond this window undermine the proactive communication principle., [] Dues payment processing confirmation appears within 5 seconds of the resident confirming payment. A long wait on a payment screen generates anxiety the copy cannot resolve., [] Amenity availability reflects real-time status. A resident who books a pool slot that was already taken has had an experience that no warm confirmation copy can repair., [] The app does not produce a visible loading state when navigating

between primary screens that have already been loaded in the current session. Transitions between the home screen and top-level sections must feel immediate., [] Images, including landscape photography on the home screen, load progressively and do not produce a blank container state visible for more than 1 second. A blank space where the landscape should be breaks the visual identity at the moment it matters most., [] The app remains functional and displays appropriate, copy-reviewed messaging when the device has no network connection. A resident in a dead-signal area on the lochside trail should not see a system crash or a raw error state., [] Session initialisation, the period between the resident tapping the app icon and the home screen being interactive, completes within 3 seconds on a device up to three years old. First-open speed is part of the first impression.

Pre-Launch Emotional Review

The pre-launch emotional review is a structured walkthrough of the complete resident experience conducted after build is stable and before any release candidate is signed off. It is not a QA session. QA will have run by this point. This review asks a different question: does the built product actually feel like Three Lochs?

That distinction matters. A product can pass every functional acceptance criterion and still fail the emotional ones. Interactions complete correctly but feel transactional. Copy has been reviewed but sounds slightly off in context. A transition works as specified but lands with less warmth than the design intended. None of those failures show up in a test log. They show up when a real person opens the app and the feeling is wrong. The pre-launch emotional review exists to catch that category of failure before a resident does.

What It Covers

The review is structured around the Feel Factor layers: first open, first task, first loop closure, and sustained engagement. The walkthrough follows the same arc the Emotional Journey Map describes, from the moment a new resident opens the app for the first time through to the experience of a long-term resident returning to complete a routine task. It does not test individual components in isolation. It moves through the product the way a resident would, attending to the cumulative feeling as much as to individual moments.

The specific moments the review assesses are as follows., First open: does the home screen produce the right feeling within sixty seconds? Name, neighbourhood, dues status, and a relevant community item all present before any interaction. The resident feels recognised, not logged in., First task completion: does a dues payment, amenity booking, or maintenance submission end with a confirmation that feels like a close rather than a receipt? The copy is checked in context, not against the library in isolation. A string that passed tone review can still land wrong when surrounded by the rest of the screen., Maintenance request flow: does the confirmation state produce confidence? Is a named assignee and a specific date visible? Does the screen feel like someone picked it up, or like it

went somewhere unspecified?, Notification experience: are proactive notifications arriving at the right moments in the right register? This is reviewed in a test environment with triggered notifications, not assumed from the specification., Onboarding arc: does the first session end with something completed? Does the welcome experience feel like an arrival at Three Lochs specifically, or a generic setup flow?, Copy in context: every category of system copy reviewed in its actual screen context. Confirmation messages, error states, empty states, enforcement templates. The review is looking for anything that has drifted from the tone specification or that reads differently when surrounded by real data rather than placeholder content., Visual consistency across utility screens: forms, error states, loading states, and edge cases checked against the Visual Design System. The review is particularly focused on screens that received less design attention during the build, as these are where brand standards most commonly slip., Sensitive communication handling: any screen or notification touching money, rule enforcement, or resident disputes is reviewed with specific attention to tone. These are the emotional landmines identified in the workshops and they require a higher level of scrutiny than routine confirmation copy.

Who Is Involved

The review requires four roles to be present. All four must attend for the session to proceed., A We Are Affective strategist or experience lead who participated in the design phase. This person holds the emotional intent of the design and is the primary voice on whether individual moments are landing correctly., The Three Lochs client lead, the person on the ThreeLochsCommunity side with authority to make decisions about what ships. Their presence is required because some findings will require a judgement call about whether an issue is blocking or post-launch, and that call cannot be made in their absence., The development lead responsible for the build. Not a representative of the team. The person who knows the codebase well enough to say, in the room, whether a finding can be addressed before launch or not., A QA representative who has worked on the product throughout the build phase and can confirm whether a finding represents a regression, a gap in the original specification, or a new issue surfaced by the review.

The session is conducted on a physical device, not a browser or a simulator. Where the product is cross-platform, it is reviewed on each platform. The walkthrough takes between two and three hours for a first run. A second session may be required if the number of blocking findings warrants it.

What It Produces

The review produces a single sign-off document: the Pre-Launch Emotional Review Report. It is structured around three categories of finding.

The first category covers what is working. These are the moments where the emotional intent has been delivered correctly in the build. They are documented not as a courtesy but because the report needs to establish a baseline of what passing looks like. A finding against a confirmed passing moment in a future review is a regression, not a new issue.

The second category covers what needs adjustment. These are findings where the intent is present but the execution is slightly off. Copy that is close but not quite in register. A transition that works but feels marginally abrupt. A confirmation screen where the layout de-emphasises the warm close. Adjustments in this category are assigned to a named person with a completion date before launch.

The third category covers what blocks launch. A blocking finding is one where the emotional architecture of the product fails at a critical moment, and no workaround exists that does not require a fix. The criteria for a blocking finding are deliberately tight. Not every imperfection blocks launch. A finding blocks launch if it directly contradicts one of the five design principles, produces a resident-facing experience that the workshops specifically identified as a trust failure, or sends a communication that cannot be retrieved once a resident has received it, for example, an enforcement notice in the wrong tone.

The report names every finding, assigns it a category, records who is responsible for resolution, and documents the specific evidence from the walkthrough that produced the finding.

The Sign-Off Process

Sign-off requires agreement from all four reviewers. A single dissenting voice on a blocking finding holds the release. This is intentional. The development lead cannot override the We Are Affective strategist on an emotional finding. The client lead cannot override the development lead on a technical feasibility question. The review is a shared judgement, and the sign-off reflects that.

The threshold for release is as follows. All blocking findings must be resolved and verified before a release candidate is issued. Adjustment findings must have a named owner and a confirmed completion date within fourteen days of launch. Anything without an owner and a date moves to the blocking category.

Once all four reviewers have agreed that the threshold has been met, the sign-off document is countersigned and dated. That document travels with the product into the post-launch review cycle and becomes the baseline against which the thirty-day review is conducted.

If the review produces more than three blocking findings, a second session is scheduled after remediation rather than accepting fixes on trust. A product with multiple emotional failures corrected under launch pressure carries a higher risk of regression than one that is reviewed again after the fixes have settled.

Post-Launch Review Framework

The sign-off document is not the end of the review process. It is the beginning of an ongoing one. The product will behave differently in the hands of real residents than it did during the pre-launch walkthrough. Patterns will emerge that no walkthrough can anticipate.

Three review sessions are scheduled at launch.

The thirty-day review is primarily a listening session. Call volume data, notification open rates, session recordings, and any direct resident feedback are reviewed against the measurable signals defined in the Design Principles. The primary question is whether the product is behaving as designed in the hands of real residents, or whether real-world usage is surfacing gaps the pre-launch review did not catch.

The ninety-day review adds the first longitudinal data. Are residents returning? Are they completing tasks faster over time, which would indicate the product is building the habit the journey map describes? Are inbound calls to the management office decreasing? This is the session where the early behavioural signals either confirm the design decisions or begin to question them.

The six-month review is a fuller assessment. It looks at the identity shift the product was built to produce. Are residents talking about Three Lochs differently? Is the relationship between resident and HOA changing in the ways the workshops predicted? This review may produce recommendations for second-phase design work if the data suggests that specific parts of the experience are not delivering the intended shift.

All three sessions are attended by We Are Affective and the Three Lochs client lead. The development team joins the thirty-day review and is invited to subsequent sessions where specific findings require a build response.