

CONTROLLED WORKING DOCUMENT

ISTOCKHOMES

Designer & Developer Manual

One Foundation • Many Branded Doorways • One Trusted Network

The controlling principle

Build on Istockhomes. Do not build another Istockhomes. A website, plugin or mobile app is a branded doorway into the shared Foundation; it is not a replacement identity system, database, marketplace or payment engine.

Owner and final authority: Brad Camp, Istockhomes Marketing Ltd.

Consolidated Edition 1.0 | 2 September 2026

Classification: Internal development guide. No credentials, private keys, tokens or private user data.

Document control

Field	Controlled value
Purpose	A single build manual for designers, developers and AI coding systems creating Istockhomes-connected websites, WordPress packages, Shopify modules, Android apps and Apple apps.
Authority	Brad Camp is the final business and architecture authority.
Relationship to Control Centre	This manual explains how to build doorways. The current Istockhomes Project Control Centre remains the governing source for production locations, build locks, status, deployment and rollback.
Edition	Consolidated Edition 1.0. Earlier conversational chapter numbering is retired.
Change rule	Do not convert a proposed example into a production fact. Verify the current code, schema and route registry before release.
Secrets	Never place credentials, API keys or private configuration in this document or in a downloadable starter package.

How to use this manual

1. Read Chapters 1–3 before designing or coding. They define the purpose, boundary and evidence rules.
2. Complete the Build Manifest in Chapter 7. It becomes the source for an AI directive or human build brief.
3. Use Chapter 8 for a custom hosted website, or the single applicable platform chapter for WordPress, Shopify, Android or Xcode.
4. Resolve every VERIFY CURRENT item against the production registry and source code before deployment.
5. Run the acceptance tests in Chapter 12 and retain the changed-file list, rollback point and test evidence.

Status language

Label	Meaning	Builder action
LOCKED	Approved architecture rule.	Follow exactly; changing it requires Brad's written authority.
CONFIRMED	Supported by current controlled project evidence.	Use it, then smoke-test it in the target environment.
TRANSITION	More than one live route or implementation currently exists.	Use the approved adapter and do not consolidate without authorization.
PROPOSED	Desired contract, feature or download not yet proven live.	Treat as a build target, not a production dependency.
VERIFY CURRENT	Likely to change or incomplete in the available evidence.	Inspect the current control centre, source, schema or deployment before release.
DISABLED	Intentionally unavailable or fail-closed.	Do not simulate success or advertise the capability as live.

Contents

- 01 Why Istockhomes
- 02 The Foundation and the doorway model
- 03 Contracts: routes, data and protected boundaries
- 04 Members, profiles and verification
- 05 Listings, media and marketplace services
- 06 Shared trust, Live and AI assistants
- 07 The Boxcar Builder and Build Manifest
- 08 Custom hosted websites and the universal AI directive
- 09 WordPress theme and connector plugin
- 10 Shopify app and theme extension
- 11 Android and Xcode doorway apps
- 12 Release, acceptance and handoff

Appendices A: Controlled registry • B: Data contract worksheets • C: Copy-and-paste AI master directive • D: Glossary

CHAPTER 01

Why Istockhomes

The network is the product; each branded experience is an entrance.

Istockhomes exists to let many specialized marketplaces share one trusted operating Foundation. A luxury-home site, aircraft site, marine site, art marketplace, WordPress site, Shopify storefront and mobile app may look completely different while using the same member identity, verification, listing records, communications and transaction history.

The problem it solves

- Independent marketplaces repeatedly rebuild sign-up, security, search, messaging, verification, payments and trust.
- Duplicate databases split a member's identity and create conflicting records.
- Designers are forced to become backend architects, while backend changes can damage working products.
- A branded doorway solves the presentation problem while the shared Foundation solves identity, data and trust.

The promise

Audience	Value
Members	One Istockhomes account and profile across participating experiences.
Buyers and sellers	Consistent verification, listing ownership, communications and transaction history.
Designers	Freedom to create distinct brands without copying the Foundation.
Developers	A defined integration boundary, stable contracts and reusable starter packages.
Doorway owners	A branded business connected to the wider Istockhomes network.
Istockhomes	Network growth, governance and an approved share of successful transactions.

Design rule

Header = Identity. Content = Value. Footer = Trust. The brand may own the visual experience; Istockhomes owns the shared Foundation and certification relationship.

Commercial principle

Joining and building may be free under the applicable Istockhomes program. Istockhomes and an approved member-developer may receive a defined share when a qualifying transaction completes. Percentages, categories, territories, payout eligibility and legal terms belong in an executed agreement and the accepted transaction snapshot—not in front-end code.

CHAPTER 02

The Foundation and the doorway model

One source of truth, accessed through approved contracts.

A doorway contains brand presentation and client-side behavior. It calls the Foundation for protected capabilities. It does not carry the Foundation inside its ZIP, theme, plugin or app bundle.

Layer	Owns	Must not own
Doorway UI	Brand, layout, navigation, accessibility, display state and approved local preferences.	Canonical members, listings, verification, offers, payments or assistant secrets.
Connector/adaptor	Authenticated API calls, response normalization, error handling and capability detection.	A shadow business database or business-rule fork.
Istockhomes Foundation	Identity, authorization, shared records, verification, business rules and protected services.	Doorway-specific visual styling.
Shared database	Canonical IDs, relationships, timestamps, status and transaction evidence.	Theme settings or disposable UI cache unless explicitly approved.

Locked architecture rules

- One canonical Istockhomes identity per member; do not create an independent membership system.
- One canonical business record for each listing, message, offer and transaction; caches are not sources of truth.
- The server decides authorization. Hiding a button is not security.
- The Foundation and secure configuration are protected. A doorway may only use approved APIs, includes or adapters.
- No client package contains database credentials, payment secrets or AI provider keys.
- Make the smallest safe change. Preserve working systems, create a rollback point and record changed files.

Request flow

Doorway UI

-> approved connector / API route

```

-> server authentication and authorization
  -> shared business services
    -> canonical Istockhomes database
  <- normalized response + permissions
<- safe display state

```

Never bypass the connector

A theme, Shopify block or mobile screen must never connect directly to MySQL, Stripe, PayPal or OpenAI. The downloadable package is a doorway, not the engine.

CHAPTER 03

Contracts: routes, data and protected boundaries

Builders code to a published contract, not to assumptions about the database.

Knowing the database fields matters, but distributing database credentials or encouraging direct SQL does not. Builders need a versioned data contract: canonical field names, types, required status values, permissions and API responses. The live schema remains authoritative and must be exported or documented before a package is certified.

App and App2

Area	Current rule	Status
/public_html/App/	Protected Foundation. Do not overwrite or refactor through a doorway project.	LOCKED
https://istockhomes.com/App/api/index.php	Production iOS evidence currently points here.	TRANSITION
/App2/	Primary web/development layer under the controlled deployment process.	CONFIRMED
/App2/api/	Exists in the web layer, but a universal canonical mobile migration is not proven.	TRANSITION
/public_html/secure/	Server-only configuration. Never package, expose or modify without authority.	LOCKED

App/App2 routing conflict

Do not choose a route because it looks newer. Resolve the current route for each client from the controlled registry or adapter. Consolidation is a separate authorized migration with regression testing.

Minimum API envelope

```
{
  "ok": true,
  "data": {},
  "error": null,
  "meta": { "request_id": "...", "contract_version": "v1" }
}
```

A failed request returns ok=false, a stable machine-readable error code, a safe message and the request ID. Never return stack traces, SQL, secrets or private identifiers.

Required contract groups

Group	Typical capabilities	Release requirement
Identity	register, login, logout, current member, account deletion request	Exact route, fields, token/session method and error codes verified.
Profiles	read/update person or business profile, avatar/logo	Server ownership and field-level permissions verified.
Verification	status, required steps, evidence submission	States and unlock rules documented.
Listings	search, view, create, update, archive/delete, media	Category schemas and owner authorization documented.
Messaging	threads, send, unread state	Membership and listing scope enforced.
Offers/transactions	create offer, accept/reject, status history	State machine and idempotency verified.
Shared services	footer/trust, Live entitlement, assistants	Hosted/include rules and failure behavior verified.

Database publication rule

The technical owner should publish a sanitized schema dictionary generated from the real database. It must list each public contract field, not credentials, internal secrets or unnecessary private columns. Migrations are owned by the Foundation team; a doorway package does not run arbitrary production migrations.

CHAPTER 04

Members, profiles and verification

Every doorway recognizes the same person without duplicating the person.

Account lifecycle

1. Register through the approved identity endpoint.
2. Verify the email or other required factor.
3. Complete the personal or business profile required for the intended action.
4. Sign in and receive the approved session or token for that client.
5. Check verification and capability status server-side before protected actions.
6. Allow sign out on every authenticated client.
7. Provide an account deletion route and explain consequences before confirmation.

State	Meaning	UI behavior
guest	No authenticated member.	Browse permitted public content; prompt to sign in for protected actions.
registered_unverified	Account exists but required verification is incomplete.	Show next verification step; do not pretend the member is verified.
verified	Required account verification has passed.	Enable only the capabilities returned by the server.
restricted	Server has limited actions.	Show a safe explanation and support path.
deletion_pending	Deletion request accepted but not completed.	Explain access and retention status from the server response.
deleted	Account no longer active.	End sessions and prevent normal sign-in unless policy permits recovery.

First-visit personalization

A doorway such as Paruse may welcome an authenticated member by name—for example, Diana can say “Hello Brad.” The display name must come from the authenticated Istockhomes profile, be escaped before display and follow the member’s privacy settings. A “first visit” must be recorded server-side per member and doorway when the behavior must persist across devices; a cookie alone is insufficient.

Account deletion is mandatory

An iOS or Android app that supports account creation must also provide an accessible deletion flow. The client requests deletion through the approved Foundation endpoint; it never deletes database rows directly.

CHAPTER 05

Listings, media and marketplace services

Category-specific experiences share a common listing core.

Core listing contract

Field	Type / example	Rule
id	immutable identifier	Created by Foundation; never generated as a client-side source of truth.
owner_id	member identifier	Derived from authenticated member; never accepted blindly from the UI.
category	homes, aircraft, marine, vehicles, art...	Validated against the controlled category registry.
title	string	Required, length-limited and escaped.
description	rich/plain text	Sanitized server-side; presentation may be transformed safely.
price	decimal + currency	Never use floating point for transaction math.
location	structured fields + coordinates	Respect privacy and category rules.
status	draft, active, paused, sold, archived...	Use only server-defined values and transitions.
media	ordered media objects	References canonical uploads; no duplicate private media store.
created_at / updated_at	server timestamp	Server-generated and returned in a stable format.
can_manage	boolean/capability	Returned by server; controls UI but server still enforces authorization.

Category extensions

Homes may need bedrooms, bathrooms and property type. Aircraft may need manufacturer, model, year, registration rules, total time and engine information. Marine, vehicles and art require different fields. Keep the common listing core stable and publish each category extension as a versioned schema; do not add speculative columns from a doorway.

Canonical uploads

Item	Controlled value
Physical path	/public_html/App/uploads/
Public URL	https://istockhomes.com/App/uploads/
App2 writer	App2/includes/public-uploads.php
Rule	No new public upload may be stored inside App2. Uploads must survive App2 replacement and rollback.

- Validate MIME type, extension, size and destination server-side.
- Deny directory listing and script execution in upload directories.
- Return canonical media objects and URLs from the Foundation.
- Verify image persistence after deployment and rollback.

Offers and transaction states

PAYMENT_PENDING -> PAID_SECURED -> DELIVERED -> RELEASE_READY
-> RELEASE_IN_PROGRESS -> RELEASED -> COMPLETED

Exception states may include cancelled, expired, abandoned, payment_failed, disputed, refunded, reversed and release_failed. The exact current state machine and transition authority must be verified before a client exposes transaction controls.

Payments are capability-gated

If checkout or payout is DISABLED, a doorway must fail closed and clearly state that the action is unavailable. Never fabricate a paid, secured or released state. Fee and payee terms are frozen into the accepted transaction snapshot.

CHAPTER 06

Shared trust, Live and AI assistants

Connect shared services once; render them consistently everywhere.

Certified footer

The shared footer is part of the Istockhomes trust layer, not decorative boilerplate. For same-server PHP pages, the current stated include target is:

```
require_once $_SERVER['DOCUMENT_ROOT'] . '/App2/footer.php';
```

Public reference: <https://istockhomes.com/App2/footer.php>

- Place the footer once near the end of every required page.
- Do not copy and edit its markup into each doorway.

- Preserve “Certified by Istockhomes” or the currently approved equivalent.
- Run footer.ping or the current trust verification test after deployment.
- External hosts, WordPress and Shopify cannot assume a local PHP include; they require the approved external footer contract or hosted trust component.

Verify the include contract

Confirm required variables, assets, allowed hosts, caching behavior and footer.ping against the current footer implementation. A URL shown here does not prove that remote embedding is permitted.

Live and paid video

The user-facing entry has been specified as <https://istockhomes.com/Live/>. Controlled project evidence may use a lowercase /live/ root. Paths are case-sensitive on many production servers, so the registry must identify the canonical root before release. A doorway links to or embeds only the approved Live entry point and passes entitlement through the Foundation; it does not recreate video rooms or payment logic.

AI assistants

Assistant	Primary role / doorway
Diana	Reception, general guidance and Paruse experience.
Madison	Homes and housing.
Katherine	Aircraft.
John	Vehicles.
Selyna	Marine.
Estee	Art.
Harold	Legal guidance within approved scope.

Known App2 components include /App2/includes/openai-assistant-chat.php, /App2/api/diana-chat.php and the controlled assistant roster. These are implementation evidence, not permission for a doorway to call provider APIs directly.

- The server selects the assistant by controlled slug and validates authorization.
- The client sends only approved context and displays safe responses.
- Conversation continuity belongs to the Foundation contract.
- Never place an OpenAI key in JavaScript, WordPress, Shopify, Android or Xcode packages.
- The locked target is one production OpenAI key in /public_html/secure/OpenAI-One-Key.php; migration remains controlled and old key files are not deleted until the authorized completion gate passes.

CHAPTER 07

The Boxcar Builder and Build Manifest

A form turns a business idea into a constrained, repeatable doorway package.

The Boxcar Builder collects business, brand, feature and platform decisions, then produces a Build Manifest. An AI or human developer works from that manifest plus this manual. The manifest does not grant access to protected code or secrets.

Required manifest fields

Section	Required fields
Owner	Verified Istockhomes member ID, legal/business name, contact, agreement status.
Doorway	Name, domain, target market, category, territory, languages and audience.
Brand	Logo assets, colors, typography, tone, photography direction and accessibility target.
Platform	Hosted web, WordPress, Shopify, Android or Xcode; versions and hosting constraints.
Features	Browse/search, listing view, account, create/edit listing, messages, assistant, Live, offers and payments.
Contracts	Approved environment, API base/adaptor, auth mode, footer method, assistant slug and category schema version.
Commerce	Agreement ID, fee policy reference, developer attribution and payout eligibility; never raw percentages without approval.
Release	Package name/version, environments, rollback method, acceptance owner and support contact.

Generated package contents

- README-FIRST.md with prerequisites and activation steps.
- build-manifest.json with non-secret configuration.
- Platform source code and a clear editable design layer.
- .env.example or configuration template containing names only, never values.
- API client/connector isolated from presentation code.
- Error, loading, empty, offline and restricted states.
- Automated lint/build tests and a manual acceptance checklist.

- CHANGED-FILES.md, VERSION.md, LICENSE/TERMS reference and rollback instructions.

Activation gate

A package remains inactive until the doorway owner/developer is an approved Istockhomes member, the agreement is recorded, the environment is registered and the Foundation issues the approved non-secret client configuration.

CHAPTER 08

Custom hosted websites and the universal AI directive

Build a full branded site without creating a second marketplace backend.

Recommended project shape

```
/public
/src/components
/src/pages
/src/services/istockhomes
/src/styles
/config
/tests
README-FIRST.md
build-manifest.json
.env.example
```

Implementation sequence

1. Validate the manifest and list every unresolved VERIFY CURRENT value.
2. Select a maintainable stack supported by the host; do not change the Foundation stack.
3. Build the brand system and responsive page shells with mock contract data.
4. Implement one Istockhomes connector module for identity, listings and shared services.
5. Add server-verified capability gates for member actions.
6. Connect listing media using canonical returned URLs.
7. Add the certified footer using the environment-appropriate method.
8. Connect the assigned assistant and Live entry only after contracts are verified.
9. Test guest, unverified, verified, owner, non-owner, restricted and failure states.
10. Package only the doorway; exclude caches, credentials, private logs and Foundation code.

Minimum pages

Page	Required behavior
Home	Brand value, featured content and clear entry to search/sign-in.
Search/results	Filters mapped to the category contract, pagination and empty/error states.
Listing detail	Canonical data/media, ownership-safe actions, assistant/Live entry when enabled.
Sign in/register	Approved Foundation identity flow; no duplicate account store.
Dashboard	Profile, member listings, messages and capabilities returned by the server.
Create/edit listing	Category schema, media upload and validation; server authorizes every change.
Trust/legal	Certified footer, privacy, terms, contact and required platform disclosures.

The complete copy-and-paste directive is provided once in Appendix C. Fill the manifest values, attach the approved starter package and current contract registry, then give the package to the AI model.

CHAPTER 09

WordPress theme and connector plugin

Presentation stays in the theme; Istockhomes integration stays in the plugin.

Architecture

Package	Responsibility
Istockhomes Connector plugin	Settings, authentication adapter, API client, caching, blocks/shortcodes, webhooks, capability checks and safe logging.
Custom theme	Brand layout, templates, typography, responsive behavior and styling for plugin-provided components.
Istockhomes Foundation	Canonical identity, listings, media, verification, communications and transactions.

Build instructions

1. Create a namespaced plugin with an activation check for PHP and WordPress versions.
2. Provide an administrator settings screen for non-secret environment/client values; store secrets server-side only when the approved contract requires them.
3. Use WordPress HTTP APIs, nonces for local actions, capability checks, sanitization and escaping.
4. Wrap remote calls in one client class with timeouts, normalized errors and limited caching.
5. Expose Gutenberg blocks or shortcodes for listing search, detail, member state, assistant launcher and certified footer.
6. Build the custom theme against those blocks; never call the Foundation directly from templates.
7. Implement uninstall behavior that removes disposable plugin settings only; never delete Istockhomes records.
8. Package plugin.zip and theme.zip separately with installation and rollback instructions.

External footer rule

A WordPress site on another host cannot use the same-server PHP require statement. Use the approved hosted footer or remote trust-component contract after it is verified. Do not scrape footer.php or copy its HTML.

WordPress AI directive extension

Append this to Appendix C: “Target WordPress. Produce two separate installable ZIPs: a namespaced Istockhomes Connector plugin and a presentation-only custom theme. Use Gutenberg blocks where practical. No theme

template may connect directly to Istockhomes, MySQL, payment providers or AI providers. Include activation, settings, uninstall, rollback and WordPress coding-standard checks.”

CHAPTER 10

Shopify app and theme extension

Shopify presents the storefront; an approved app connects the Istockhomes network.

A Shopify theme ZIP alone is not an appropriate secure integration layer. Use an embedded or approved Shopify app backend plus a theme app extension/app blocks. Keep ordinary Shopify customer identity separate from Istockhomes membership until an approved account-linking contract exists.

Required deliverables

- Shopify app source with OAuth/authentication, session handling, configuration and webhook verification.
- Theme app extension with blocks for discovery, listing cards/detail links, assistant entry and certified trust component.
- Optional presentation theme ZIP, kept independent from secure app logic.
- Data minimization, privacy webhooks and deletion handling appropriate to the app’s distribution model.
- Install, development-store test, production configuration, rollback and uninstall documentation.

Build instructions

1. Register the app and approved redirect URLs for the target environment.
2. Use Shopify’s supported authentication and session model; verify HMAC/webhook signatures.
3. Keep Istockhomes calls on the app server, never in Liquid with secrets.
4. Use app blocks so merchants can place components without modifying theme code.
5. Proxy only approved public/member requests and normalize errors.
6. Do not treat a Shopify customer as an Istockhomes member without explicit verified linking.
7. On uninstall, revoke/expire local sessions and remove disposable app data; do not delete canonical Istockhomes data.
8. Test on a development store before distribution or production installation.

Shopify AI directive extension

Append this to Appendix C: “Target Shopify. Build an installable Shopify app plus a theme app extension; create a separate theme ZIP only if the manifest requests it. Use Shopify-supported authentication, signed webhooks and app blocks. No Liquid or browser JavaScript may contain secrets or call protected Foundation services directly. Implement required privacy/uninstall handling and document development-store testing.”

CHAPTER 11

Android and Xcode doorway apps

Two native starter packages, one Foundation contract and the same activation discipline.

Shared mobile rules

- The starter ZIP contains client source and design assets only—no App, App2, database, payment engine, assistant engine or secrets.
- Use the platform secure store for session material; never ship provider or database credentials.
- Use one API client layer and environment configuration.
- Support sign in, sign out, account deletion request, profile, listings and capability-controlled actions.
- Show loading, empty, offline, expired-session, restricted and server-error states.
- Use canonical media URLs and safe upload methods returned by the Foundation.
- Register developer attribution and doorway identity during activation.

Android package

Item	Controlled / target value
Existing app package	com.istockhomes.app — VERIFY CURRENT before using as a derivative identifier.
Existing APK reference	https://istockhomes.com/Downloads/Istockhomes.apk — VERIFY CURRENT availability and version.
Starter ZIP target	https://istockhomes.com/Downloads/Istockhomes-Android-Doorway-Starter.zip — PROPOSED until published and validated.
Release artifact	Signed Android App Bundle (AAB) for Play distribution; APK may be produced for controlled testing.
Signing	Each derivative app uses its authorized permanent release key; never include the private key in the starter ZIP.

1. Download and checksum the approved starter ZIP.
2. Open it in the supported Android Studio version and build the unchanged baseline.
3. Create the authorized application ID, display name and signing configuration.
4. Fill the non-secret manifest; keep environment values out of source where appropriate.
5. Restyle screens and assets without changing the connector contract.
6. Run unit, UI and authenticated sandbox tests.
7. Build a signed release AAB/APK, retain mapping files and record versionCode/versionName.
8. Complete Play Console privacy, data safety, deletion and testing requirements.

Xcode package

Item	Controlled / target value
Existing project	Istockhomes-Mobile-V3M — VERIFY CURRENT repository/release.
Existing bundle evidence	Istockhomes.IstockhomesXcodeV1; Apple ID 6760603059 — do not reuse for a new brand without authority.
Existing route evidence	https://istockhomes.com/App/api/index.php with preferApp2=false — TRANSITION.
Starter ZIP target	https://istockhomes.com/Downloads/Istockhomes-Xcode-Doorway-Starter.zip — PROPOSED until published and validated.
Distribution	Archive in Xcode, validate, upload to App Store Connect and test through TestFlight before review.

1. Download and checksum the approved starter ZIP.
2. Open the project in the supported Xcode version and run the unchanged baseline.
3. Create the authorized bundle ID, app record, team, signing and capabilities.
4. Fill the non-secret Build Manifest and verified API adapter selection.
5. Replace brand assets, colors, copy and category forms while preserving shared service boundaries.
6. Test sign in/out, listing create/update, media, expired sessions and account deletion.
7. Archive, validate and upload to App Store Connect.
8. Complete privacy nutrition labels, screenshots, support/privacy URLs and TestFlight acceptance before submission.

Download link release gate

Do not publish either starter URL in customer-facing material until the ZIP exists, builds unchanged, contains no credentials, has a checksum/version, and passes the baseline acceptance suite.

CHAPTER 12

Release, acceptance and handoff

A build is complete only when it is reproducible, reversible and proven against the Foundation.

Pre-build report

- Canonical workspace or starter version.
- Applicable locked directives.

- Protected systems and files affected.
- Rollback point.
- Complete proposed changed-file list.
- Every unresolved VERIFY CURRENT item and its owner.

Core acceptance matrix

Area	Required proof
Build	Clean install/build from documented prerequisites; no missing files or private dependencies.
Public web	Home, search, listing detail, navigation and mobile layout load without fatal errors.
Identity	Register where enabled, verify, sign in, refresh/reopen, sign out and account deletion request.
Authorization	Owner/non-owner controls differ correctly and server rejects bypassed unauthorized requests.
Listings	Create, validate, upload, view, edit and archive/delete using the approved contract.
Persistence	Existing listing/profile/media records remain visible after deployment and rollback rehearsal.
Trust	Correct brand header, certified footer and footer.ping/trust check.
Assistant	Correct assistant identity, first and second turns, authenticated name handling and safe failure state.
Live	Correct canonical entry, entitlement behavior and unavailable state; no unapproved payment simulation.
Transactions	Only when enabled: state transitions, idempotency, ownership, fee snapshot and error recovery.
Security	No secrets in package/logs; TLS; input validation; output escaping; signed callbacks/webhooks.
Accessibility	Keyboard/focus, labels, contrast, scalable text and screen-reader basics on supported clients.

Handoff package

- Installable source package and release artifact, where applicable.
- README-FIRST.md and platform setup guide.
- Completed Build Manifest with secret values removed.
- Contract and schema versions used.

- Test report with environment, date and failures/limitations.
- Changed-file list, migrations (Foundation-owned only), rollback and recovery steps.
- License, agreement and developer-attribution references.
- Known issues and support ownership.

Definition of done

The package installs or builds from a clean environment, connects only through approved contracts, passes the applicable tests, contains no secrets, has a rollback path and is accepted by the designated owner. “The screen looks right” is not completion.

Appendix A — Controlled registry

This registry centralizes values that were previously repeated. “VERIFY CURRENT” means the release owner must compare the value with the current Control Centre, source and production environment.

Capability	Location / value	Status
Protected Foundation	/public_html/App/	LOCKED
App API router	https://istockhomes.com/App/api/index.php	TRANSITION; current iOS evidence
App2 web layer	/App2/	CONFIRMED
App2 API	/App2/api/	TRANSITION
Certified footer	https://istockhomes.com/App2/footer.php	CONFIRMED target; include contract VERIFY CURRENT
Footer include	require_once \$_SERVER['DOCUMENT_ROOT'] . '/ App2/footer.php';	Same-server PHP only
Live	https://istockhomes.com/Live/ versus /live/	VERIFY CURRENT case-sensitive canonical root
Public uploads	/public_html/App/uploads/ and https://istockhomes.com/App/ uploads/	LOCKED
App2 upload writer	App2/includes/public-uploads.php	LOCKED
Diana API	/App2/api/diana-chat.php	CONFIRMED evidence; contract VERIFY CURRENT
Assistant include	/App2/includes/openai-assistant- chat.php	CONFIRMED evidence
OpenAI target secret	/public_html/secure/OpenAI-One- Key.php	LOCKED target; controlled migration
Android APK	https://istockhomes.com/ Downloads/Istockhomes.apk	VERIFY CURRENT
Android starter ZIP	https://istockhomes.com/ Downloads/Istockhomes-Android- Doorway-Starter.zip	PROPOSED
Xcode starter ZIP	https://istockhomes.com/ Downloads/Istockhomes-Xcode- Doorway-Starter.zip	PROPOSED

Appendix B — Data contract worksheets

Complete from the real schema/API before certifying a doorway. Do not fill missing values by guesswork.

Identity contract

Operation	Route/method	Request fields	Response/errors
Register	VERIFY CURRENT	VERIFY CURRENT	VERIFY CURRENT
Login	VERIFY CURRENT	VERIFY CURRENT	VERIFY CURRENT
Me/profile	VERIFY CURRENT	—	VERIFY CURRENT
Logout	VERIFY CURRENT	—	VERIFY CURRENT
Deletion request	VERIFY CURRENT	VERIFY CURRENT	VERIFY CURRENT

Listing contract

Field	DB/API type	Required/validation	Privacy/permission
id	VERIFY CURRENT	Server-generated	Public or scoped
owner_id	VERIFY CURRENT	Authenticated server value	Owner/admin
category	VERIFY CURRENT	Registry value	Public
title	VERIFY CURRENT	Length/required	Public
price/currency	VERIFY CURRENT	Decimal/ISO currency	Public
status	VERIFY CURRENT	State transition	Capability-based
media	VERIFY CURRENT	Canonical objects	Policy-based

Service contract

Service	Entry/route	Authentication	Failure behavior
Certified footer	Appendix A	Host/contract dependent	Fail visibly or approved fallback
Live	VERIFY CURRENT	Entitlement contract	Unavailable; never fake access
Assistant	Controlled slug/route	Member/session as required	Safe retry/support message
Messages	VERIFY CURRENT	Member + scope	No cross-thread data
Transactions	VERIFY CURRENT	Member + capability	Fail closed

Appendix C — Copy-and-paste AI master directive

How to use

Replace every bracketed value, attach the approved starter package and current sanitized contract registry, then paste this directive into the chosen AI model. The AI must stop on missing production facts rather than inventing them.

You are the implementation developer for an Istockhomes-connected branded doorway.

PROJECT

- Doorway name: [NAME]
- Domain/store/app name: [DOMAIN OR APP]
- Platform: [HOSTED WEB / WORDPRESS / SHOPIFY / ANDROID / XCODE]
- Marketplace category: [CATEGORY]
- Audience and territory: [AUDIENCE / TERRITORY]
- Brand direction: [COLORS, TYPE, TONE, ASSETS]
- Required features: [FEATURE LIST]
- Assigned assistant slug: [SLUG OR NONE]
- Live capability: [ENABLED BY APPROVED CONTRACT / DISABLED]
- Offers/payments: [ENABLED BY APPROVED CONTRACT / DISABLED]
- Approved starter/version: [PACKAGE + VERSION]
- Approved contract/schema version: [VERSION]

CONTROLLING RULE

Build on Istockhomes. Do not build another Istockhomes. This project is a branded doorway into the shared Foundation. Do not create a second member system, listing database, verification system, messaging system, assistant backend, Live platform, transaction engine or payment system.

BEFORE CODING

1. Read the attached Istockhomes Designer & Developer Manual, Build Manifest, contract registry and starter README completely.
2. Report the workspace, locked directives, protected systems, rollback point, proposed changed files and every VERIFY CURRENT item.
3. Stop and ask for the missing controlled value if a route, schema field, auth method, secret-loading method, entitlement, payment capability or production path is not verified. Do not guess.

IMPLEMENTATION RULES

- Keep presentation separate from the Istockhomes connector/API client.
- Use only the approved route adapter for this client. Do not consolidate App/App2 or modify the Foundation.
- Never connect directly to the database, Stripe, PayPal, OpenAI or another protected provider.
- Never include credentials, tokens, private keys, signing keys or production personal data in source, logs, prompts or packages.
- The server controls identity, authorization, ownership, verification, state transitions and fee snapshots.
- Use canonical IDs, status values, timestamps and media URLs returned by the Foundation.
- Include loading, empty, offline, expired-session, restricted and safe error states.
- Include sign in, sign out and account deletion request when account creation or authenticated membership is available.
- Integrate the Certified by Istockhomes footer through the approved method for the target environment.
- Connect only the assigned assistant slug through the approved Foundation endpoint. Personalize from the authenticated profile; never invent a name.
- Link/embed Live only through the verified canonical entry and entitlement contract.
- If commerce is disabled, fail closed and do not simulate checkout, payment, escrow, delivery or payout success.
- Make the smallest safe change and preserve the starter's working contract layer.

DELIVERABLES

1. Complete installable/buildable source package for the selected platform.
2. README-FIRST with prerequisites, setup, activation, build, deployment and rollback.
3. Completed non-secret build-manifest.json and .env.example/configuration template.
4. Isolated connector/API client and documented public interfaces.
5. Automated lint/build tests plus the manual acceptance checklist.
6. CHANGED-FILES.md, VERSION.md, known limitations and support handoff.
7. No placeholder success responses and no claimed feature that has not been tested.

ACCEPTANCE

Prove clean installation/build, responsive/accessibility basics, identity lifecycle, owner/non-owner authorization, listings and media, footer/trust, assistant two-turn continuity, Live availability behavior, secure error handling and rollback. Test transactions only when the controlled environment says they are enabled.

Begin by returning the pre-build report and questions for unresolved controlled values. Do not write code until that report is complete.

Appendix D — Glossary

Term	Meaning
Foundation	Protected Istockhomes services and source-of-truth data.
Doorway	A branded website, store module or app that uses the Foundation.
Boxcar	A configured doorway package generated from an approved Build Manifest.
Build Manifest	Non-secret structured instructions defining brand, platform, contracts, capabilities and release ownership.
Connector	The single integration layer between a doorway and approved Istockhomes services.
Contract	The versioned route, field, state, permission and error behavior a client may rely on.
Certified footer	The shared Istockhomes trust element and its verification behavior.
Capability	A server-returned permission or enabled service, not a hard-coded UI assumption.
Canonical	The approved source or location currently controlling production behavior.
Fail closed	Deny or mark unavailable when authorization or state cannot be proven.
Transaction snapshot	The immutable accepted record of parties, fees, payees and split amounts for a transaction.
VERIFY CURRENT	A release blocker requiring evidence from the current controlled environment.

Final handoff statement

This consolidated manual replaces the earlier conversational chapter sequence. It removes repeated platform directives, centralizes paths and status, and separates locked production facts from proposed starter targets. It is ready to guide a designer or AI developer only when paired with the current Project Control Centre, sanitized schema/API contracts and an approved platform starter package.

Remember

One Foundation. One member identity. One database. Many branded doorways. Every release earns trust by preserving the working system and proving what it claims.