

SPEC-01: Hearthlink System Overview

1. Background

Hearthlink is a modular, secure, and extensible desktop/web platform composed of independent peer modules—Launcher, Alden UI, Alice, Mimic, Vault, Sentry, Synapse, and Core Services—coordinated via a universal system launcher. It emphasizes security-first architecture, dynamic persona-driven interactions, and comprehensive observability.

2. Requirements (MoSCoW)

Must have

- Universal System Launcher with branded Welcome Screen and module selector
- Secure inter-module communication over Synapse with schema validation and rate-limiting
- Centralized authentication (Core Services Auth) and RBAC enforcement across modules
- Secret management (Vault) with AES-256-GCM, HSM integration, and audit logs
- Real-time monitoring & alerting (Sentry) with <2 s ingestion latency and 90-day retention

Should have

- Persona & context management (Alice CBP, Mimic personas) for tailored interactions
- Configurable feature toggles and global settings via Core Services Config Store
- Project Command framework for SOP-driven workflows

Could have

- Self-hosted LLM inference with local workspace integration
- Multi-cluster federation for geo-redundant Synapse message replication

Won't have (this increment)

- Mobile-specific interfaces (deferred)
- GraphQL gateway or advanced A/B testing engines

3. Method

3.1 High-Level Architecture Diagram

```
@startuml
package "System Launcher" {
    [Launcher Stub] --> [Welcome Screen]
}
package "Modules" {
    [Alden UI]
```

```

[Alice]
[Mimic]
[Vault]
[Sentry]
[Synapse]
[Core Services]
}
package "Infrastructure" {
  [Kafka/NATS]
  [PostgreSQL/TimescaleDB]
  [Prometheus]
  [HSM/KMS]
}
[Welcome Screen] --> Modules
Modules --> Infrastructure
@enduml

```

4. UI Components & Wireframes

Design Concept Background:

- Refer to local asset: `G:\MythologIQ\Hearthlink\design-assets\Hearthlink-launcher-bg.png` for the full-screen background image on the Welcome Screen.

4.1 Welcome Screen (Launcher)

Design *Concept* *Mockup:* Refer to
`G:\MythologIQ\Hearthlink\design-assets\Hearthlink-launcher-bg-withicon-mock.png`

```

+-----+
|                                     |
|               Hearthlink Welcome   |
| [Background image with central emblem and module icons as shown] |
| Logo | Version vX.Y.Z |           | [Env: ▼ |
Prod] [Theme: 🌞/🌙] |
+-----+
| [Core] [Alden] [Alice] [Vault] [Mimic] [Synapse] [Sentry] |
|                                     |
+-----+
| [ Voice Icon ] Voice   [ Help Icon ] Help   [ Accessibility Icon ] Acc |
|                                     |
+-----+

```

Component	Function	API/Data Call
ModuleIconButton	Direct launch for each module (icon+label)	<code>window.launcher.launch(moduleId)</code>
VoiceIcon	Toggle system-wide voice HUD (default ON; state from settings)	<code>window.voice.toggle()</code>
HelpIcon	Open Help side drawer	<code>window.help.openDrawer()</code>
AccessibilityIcon	Open accessibility settings	<code>window.accessibility.open()</code>

4.2 Help Side Drawer & Bug Report Entry

... (existing content) ...

• **Troubleshooting & FAQs:** Add a third tab within the Help drawer, labeled “FAQs & Troubleshooting,” containing a curated list of common issues, self-service steps, and links into system docs.

4.3 System Settings Interface

```
+-----+
| System Settings (Full Screen)                                ||x||
+-----+
| Tabs: [General] [Security] [Voice] [Appearance] [Integrations] [Version
History]
+-----+
| • General:
|   - Default Module on Startup [Dropdown]
|   - Environment Default [Dropdown]
|   - Language [Dropdown]
+-----+
| • Security:
|   - Manage GitHub Token [Rotate] [View Permissions]
|   - Vault Encryption Key Status [View]
|   - mTLS Certificate Management [Upload/Download]
+-----+
| • Voice:
|   - Global Voice ON/OFF [Toggle]
|   - Voice Activation Phrase [Textbox]
|   - Privacy Mode (Local only) [Toggle]
+-----+
| • Appearance:
|   - Theme [Light/Dark/Auto]
|   - Launcher Background [Upload]
|   - Panel Layout Preferences [Reset to Default]
+-----+
```

```
| • Integrations:
|   - List Connected Services [Resync]
|   - Add New Integration [GitHub, Sentry, LLM]
|-----|
| • Version History:
|   - Changelog entries with Date, Version, Description
|   - Filter by module and category (feature, bugfix)
|-----|
| [Save Changes] [Cancel]
+-----+
```

Component	Function	API/Data Call
SettingsTabs	Navigate between setting categories	N/A
DropDownField	Select options (module, env, language)	GET /system/settings, PUT /system/settings
ToggleSwitch	Enable/disable features (voice, privacy)	PUT /system/settings/{key}
UploadButton	Upload assets or certificates	POST /system/settings/upload
SaveChangesButton	Persist all modified settings	PUT /system/settings

6. Additional Considerations

- **Plugin & Extension Framework:** Offer a plugin marketplace UI within System Settings to discover, install, and manage extensions with automated compatibility checks. (Future conversation)
- **Audit Trail Explorer:** Vault already provides a unified log explorer in SPEC-02; ensure enhancements to filter across modules and integrate with Sentry events.
- **Usage Analytics Dashboard:** Core Services provides analytics endpoints; include a system overview panel linking to these metrics from SPEC-07's Analytics Query API.

These system-agnostic enhancements ensure Hearthlink remains extensible, transparent, and supportable as it scales.

```
+-----+
| System Settings (Full Screen)                                     ||x||
|-----|
| Tabs: [General] [Security] [Voice] [Appearance] [Integrations]
|-----|
| • General:
|   - Default Module on Startup [Dropdown]
|   - Environment Default [Dropdown]
```

```

|   - Language [Dropdown]
|-----|
| • Security:
|   - Manage GitHub Token [Rotate] [View Permissions]
|   - Vault Encryption Key Status [View]
|   - mTLS Certificate Management [Upload/Download]
|-----|
| • Voice:
|   - Global Voice ON/OFF [Toggle]
|   - Voice Activation Phrase [Textbox]
|   - Privacy Mode (Local only) [Toggle]
|-----|
| • Appearance:
|   - Theme [Light/Dark/Auto]
|   - Launcher Background [Upload]
|   - Panel Layout Preferences [Reset to Default]
|-----|
| • Integrations:
|   - List Connected Services [Resync]
|   - Add New Integration [GitHub, Sentry, LLM]
|-----|
| [Save Changes] [Cancel]
|-----+

```

Component	Function	API/Data Call
SettingsTabs	Navigate between setting categories	N/A
DropDownField	Select options (module, env, language)	GET /system/settings, PUT /system/settings
ToggleSwitch	Enable/disable features (voice, privacy)	PUT /system/settings/{key}
UploadButton	Upload assets or certificates	POST /system/settings/upload
SaveChangesButton	Persist all modified settings	PUT /system/settings

5. API Endpoints

Method	Path	Description	Auth Scope
GET	/system/settings	Retrieve current system-wide settings	system.settings.read
PUT	/system/settings	Update system-wide settings	system.settings.write

Method	Path	Description	Auth Scope
POST	/system/settings/upload	Upload certificates or assets	system.settings.write

6. Additional Considerations

- **Version History & Change Log:** Maintain a system-level changelog showing feature rollouts and bugfix versions for transparency.
- **Troubleshooting & FAQs:** Provide a cross-module FAQ and troubleshooting panel accessible from the banner or Help drawer, covering connectivity, authentication, and common issues.
- **Plugin & Extension Framework:** Offer a plugin marketplace UI within System Settings to discover, install, and manage extensions with automated compatibility checks.
- **Audit Trail Explorer:** Implement a unified log search dashboard pulling from Vault, Core Services, and Sentry, with filters by time, user, module, and severity.
- **Usage Analytics Dashboard:** Include an analytics view for tracking user sessions, API volumes, and error trends over time to inform operational decisions.

These system-agnostic enhancements ensure Hearthlink remains extensible, transparent, and supportable as it scales.