High-Level Feature Set

Real-time Collab

Link and email sharing

Node/plugin API

Node operations (filtering, searching)

Exporting (PDF, PNG)

User Dashboard for managing multiple files

First-party plugins

Make a Sandbox alpha version of the app

→ Push new features to sandbox on a weekly cycle

→ Use sandbox for A/B testing with real users

→ Iterate design and UX on sandbox

↓

After 5 iteration cycles, push beta version of the app

→ Enroll more users and get more feedback. Iterate on a 2-week cycle

→ After 3 cycles, open plugin API for third-party use

Sandbox Alpha 0.1 Features

User Actions: login (with Google), logout

User Dashboard with document creation, renaming and deletion

Node types: Text, Image, Video, Files, Links

Node actions: add, edit, delete

Sandbox Alpha 0.2 Features

View-only Link Sharing (second user only spectates, file owner can edit)

Pointer presence for active users

Sharing Menu

Node actions: reparent, lock, collapse

Sandbox Alpha 0.3 Features

Read-Write-Admin permission based sharing

Collaborative Editing (with mutual write blocks, i.e. only 1 user can edit 1 node at a time)

Catch up on any pending work

Sandbox Alpha 0.4 Features

Activity Pane

Document Actions: Filtering, Searching

Node Actions: comments, undo, redo

Feedback form

Sandbox Alpha 0.5 Features

Polish UI for beta launch

Settings Menu: dark mode, color theme, about, privacy policy

Add keyboard shortcuts

Website landing page

Catch up on any pending work

User Testing Timeline

Until Alpha v0.3, closed testing within internal team at LMS (target ~10-15 people)

From Alpha v0.4, closed testing within internal team and any volunteering friends/contacts (target ~30-50 people)

Starting at beta release, invite-only testing up to 100 people

A/B Testing Targets for Refining Design

UI vocabulary (what are nodes called? what is a document called?)

UI skins (colors, shadows, card styles)

Action flows (adding/deleting nodes, sharing documents)

UI element locations (where are specific buttons placed?)

Core concept refining (which node layouts appeal more to users, overall issues with program)