

Houston Command App: Development Roadmap, Features, and Decisions



Landing Site (houstoncmd.com)

Introduction to Houston

- Highlight why Houston is great for developers
- Emphasize its flexibility, clean architecture, and ease of use

Code Generation Showcase

- Display generated code snippets with callouts to key features
- Explain how clean architecture allows swapping backend data sources (e.g., Supabase, Django, ServerPod) with minimal changes

Demo of Blueprint Creation

- Walk through creating a blueprint from start to finish
- Show the resulting UI in the Flutter app and backend CRUD screens

Roadmap

- Outline what's next for Houston (e.g., CLI improvements, hosting options)

Developer Benefits

- Opinionated patterns for efficiency and responsiveness
- Flexibility in backend choices (Supabase, Django, ServerPod)
- CLI for heavy-lifting code generation

Example Projects

- Showcase projects built with Houston (e.g., Surf Music, Storyplace, DashHost)

Hero Video

- High-quality sales video with voiceover, demonstrating key features



FAQ and Contact

Address common questions and provide links to GitHub, Discord, and support



Call to Action

Encourage users to clone the repo or get started with Houston



Documentation

Link to MK Docs-based documentation with video tutorials and search functionality

App (dash.houstoncmd.com)

Core Features



Project/App Creation

- Users can create an app with multiple blueprints
- Each blueprint includes models and properties



Blueprint Management

- Intuitive UI for creating and managing blueprints
- Generate code directly from blueprints



Code Export

- Export generated code as a zip file
- Option to build and host a Flutter web app for prototyping



Hosting Options

- Use Supabase for demo versions (low infrastructure overhead)
- Consider Django for future scalability and customization



The Houston Command app provides an intuitive interface for developers to create, manage, and export code from blueprints.

Development Phases



Key Features for Developers

Flexibility

- Swap backend data sources (Supabase, Django, ServerPod) without changing frontend code
- Support for RESTful APIs and custom backend implementations

Flutter-First

- Supports iOS, Android, desktop, and web apps
- Works with WebAssembly for web targets



Code Generation

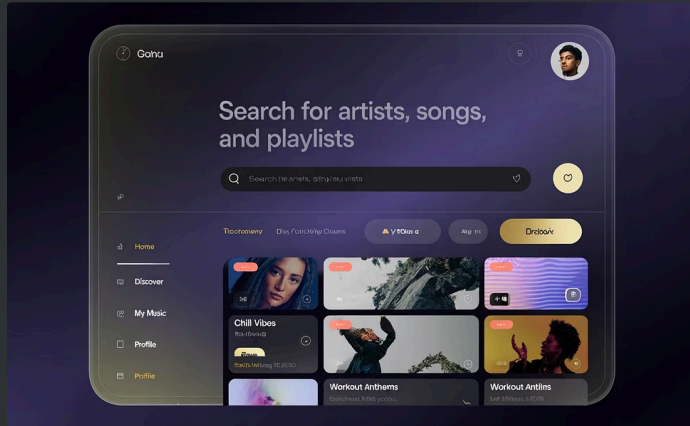
- CLI generates scaffolding code for frontend (Flutter) and backend (Django, Supabase, ServerPod)
- Includes models, migrations, API endpoints, serializers, and admin interfaces

Clean Architecture

- Separation of concerns for easy maintenance and scalability
- Intuitive patterns for managing state and CRUD operations

Houston Command provides developers with a powerful set of tools designed to streamline app development while maintaining flexibility and clean architecture principles.

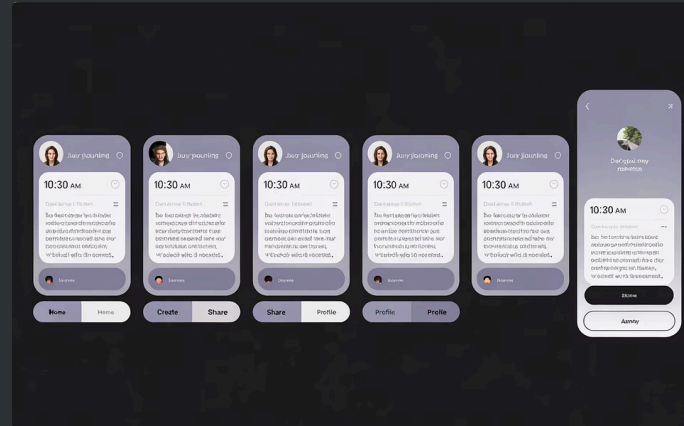
Example Projects



Surf Music

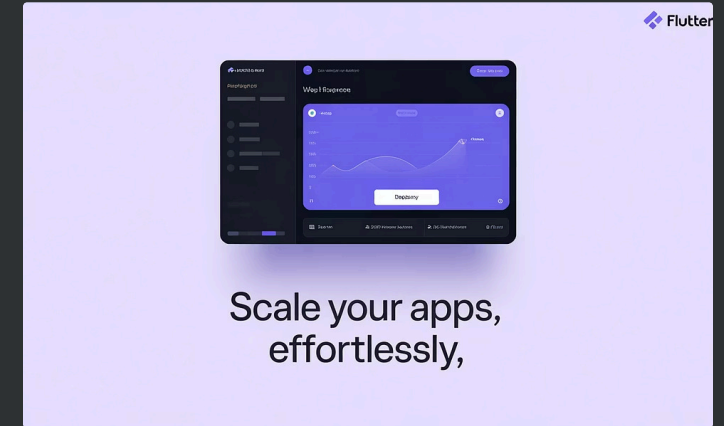
A B2B music marketplace built with Houston.

- Features media playback, social following, and AI integrations



Storyplace

A social journaling app available on iOS and Android.



DashHost

A web hosting platform for Flutter apps, with a dashboard built in Houston.

These real-world applications demonstrate the versatility and power of Houston Command for building complex, feature-rich applications across different domains.



Design and Aesthetic



The landing site should have a sleek, sci-fi aesthetic (e.g., Houston in space, Mars Attack vibes).

Visual Elements

- Use animations or videos to demonstrate blueprint creation and code generation
- Include diagrams to explain the architecture and components

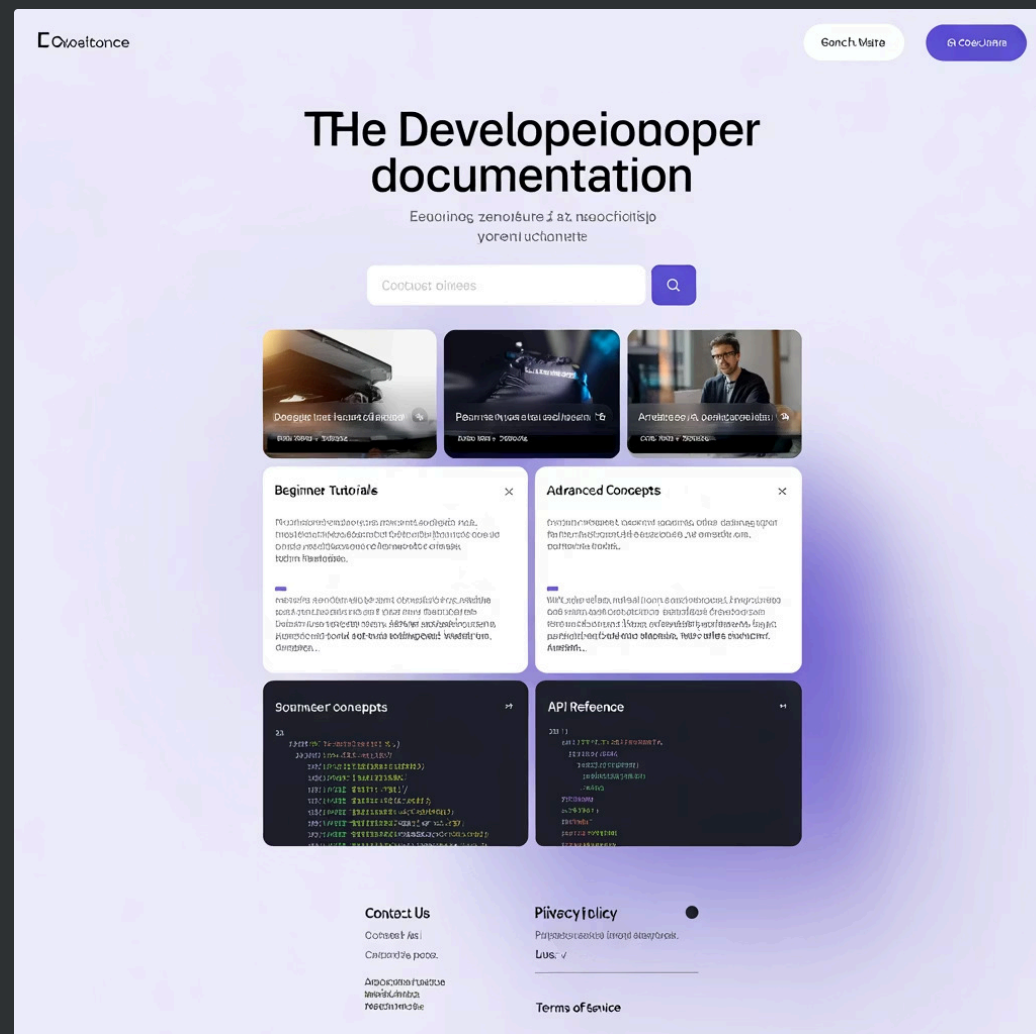
The design should reflect the cutting-edge nature of the platform while maintaining clarity and usability for developers.

Documentation and Community

Documentation

MK Docs Implementation

- Use MK Docs for markdown-based documentation
- Include video tutorials hosted on YouTube
- Add a search function for easy navigation



Community Contributions

Collaboration Platforms

- Use GitHub for issue tracking and pull requests
- Set up a Discord server for discussions and feedback
- Consider a Kickstarter or Fungible campaign for financial support



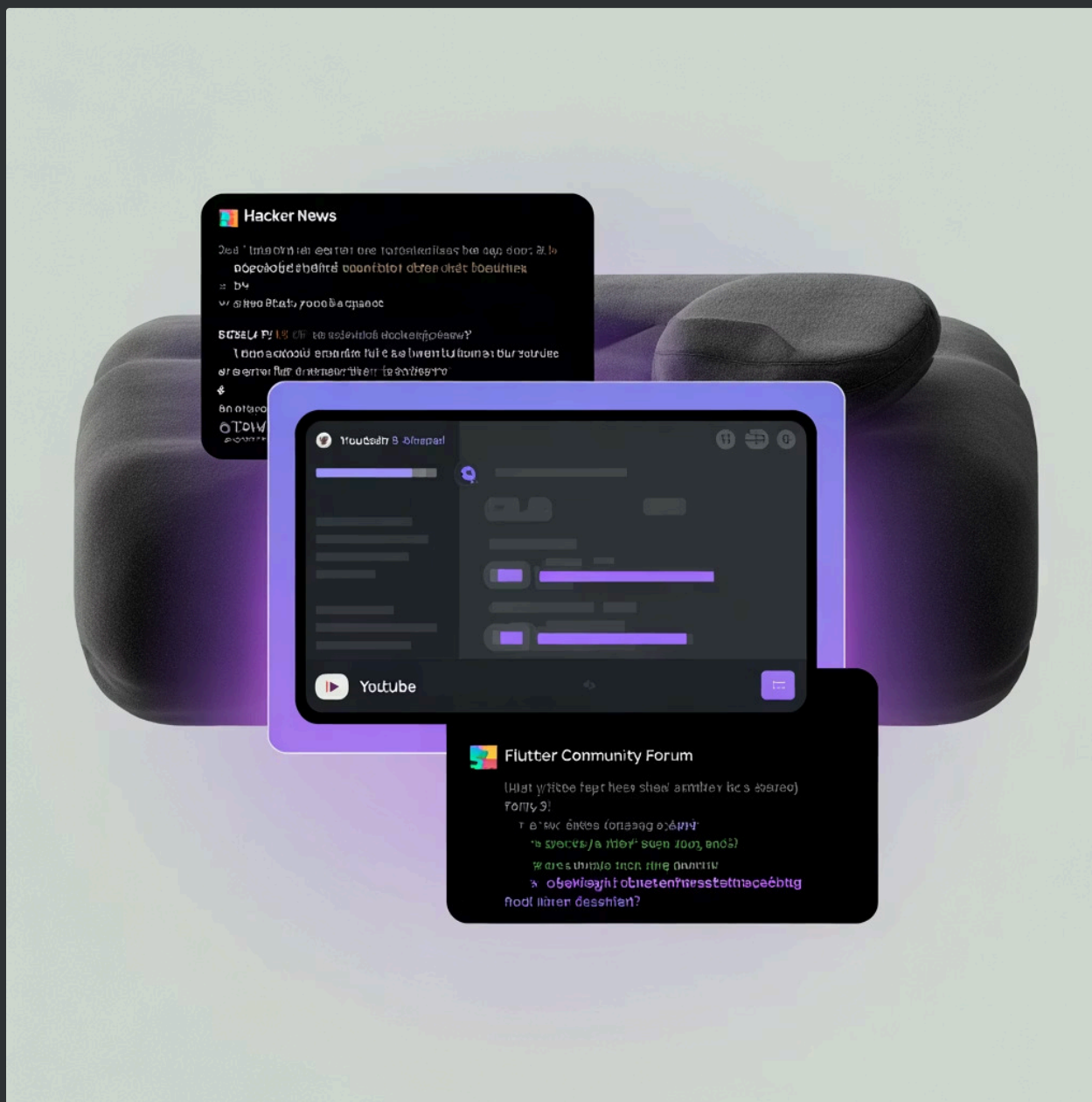
Marketing and Outreach

Channels

- Promote on Hacker News, YouTube, and Flutter communities
- Collaborate with Flutter influencers (e.g., Rory, Randall Schwartz)
- Aim for a feature on Fire Ship

Materials

- Create high-quality screen recordings with voiceovers
- Develop technical videos explaining the architecture and benefits



Effective marketing will focus on reaching developers through channels they already use and trust, with content that clearly demonstrates the value proposition of Houston Command.

High-quality technical content will be essential for building credibility within the developer community and showcasing the platform's capabilities.

Collaboration and Partner Requirements

Skills Needed

- Expertise in Django, frontend state management, and tooling
- Strong understanding of developer experience and clean architecture

Collaboration Process

- Use Discord for communication and GitHub for pull requests
- Hold vision board meetings to plan and execute features

Next Steps

Landing Site

- Finalize design and content
- Create hero video and demo animations

Documentation

- Set up MK Docs and create initial tutorials

Challenges and Learnings

- Time Management
 - Building the extra layer on top of existing tools will require significant effort
- Documentation
 - Time-consuming but essential. Need help from a dedicated contributor
- Scalability
 - Future hosting options (e.g., Django vs. Supabase) need careful consideration

Phase 1 Development

- Build the GUI for the CLI
- Host a demo of exported code

Community Engagement

- Launch Discord server and GitHub repository
- Begin marketing efforts on Hacker News and YouTube

This roadmap and feature list provide a clear path for developing Houston Command, ensuring it meets the needs of developers while maintaining flexibility and scalability.