Vyxen Overview:

Vyxen is a sophisticated social media app similar to reddit. The platform enables users to create profiles, share posts with attached media, follow other users, engage with content through likes, comments, and bookmarks, and communicate via a real-time messaging system. Vyxen also features a user verification system and subscription tiers, giving creators the option to monetize their content on the platform.

Tech Stack:

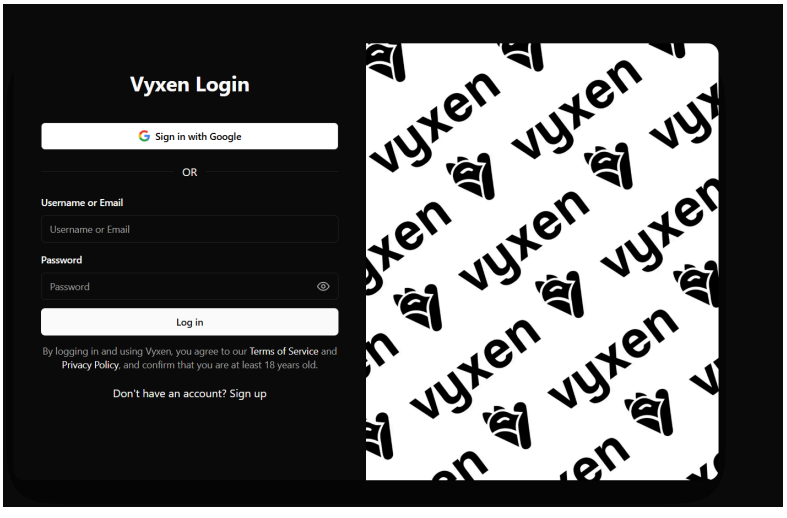
Vyxen is built on the Next.js framework. I felt this was a good choice since I could leverage the serverless API capabilities and have prior experience creating web apps with react. A few libraries I relied on while building out the frontend include Radix UI, Tailwind CSS, and Lucide react. These libraries provided a solid foundation for building consistent, responsive components while maintaining a clean design aesthetic.

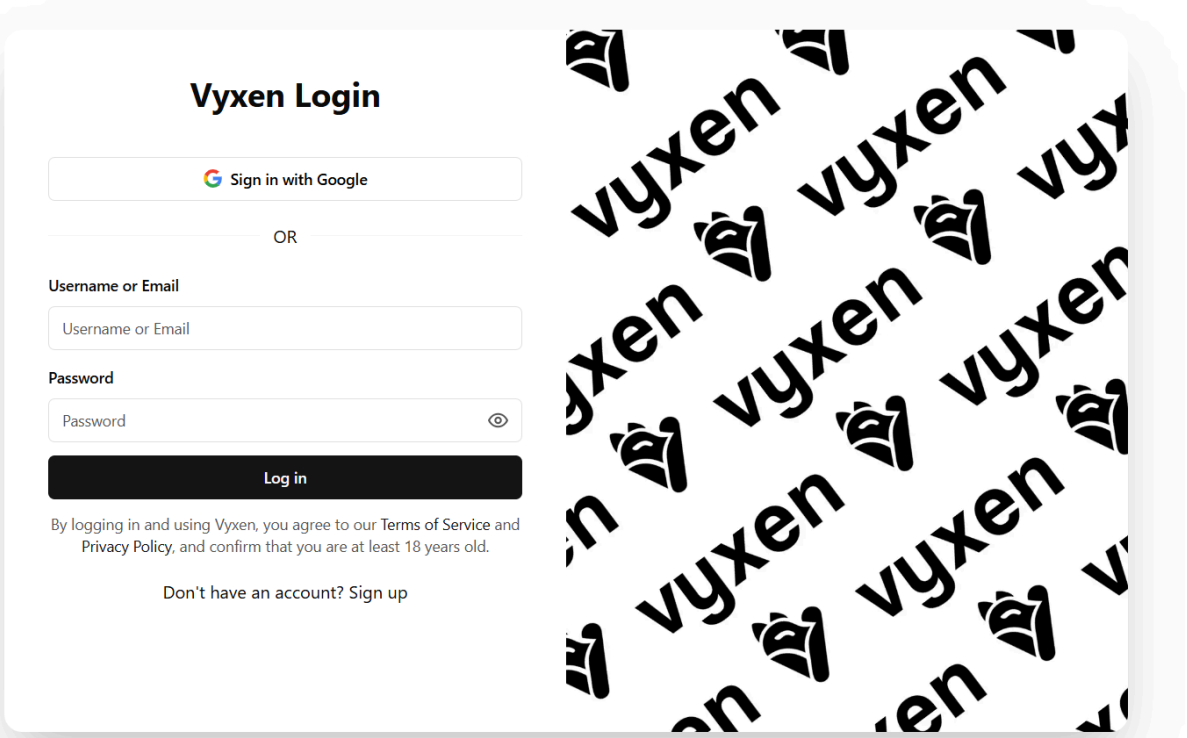
On the backend, Vyxen uses Prisma ORM to interact with a PostgreSQL database, which serves as the primary relational database for the app. The infrastructure uses several AWS services: S3 for media storage and delivery, DynamoDB works as our real-time database for the chat system, Lambda and API Gateway work together to run our websocket messaging system. Upstash Redis powers the personalized feed algorithm with caching of user preferences, content scores, and feed diversity tracking, which greatly improves performance and helps ease the load on our main database. This caching layer is necessary for delivering real time personalized content recommendations based on user interactions and engagement patterns.

App Features:

Signup/Login:

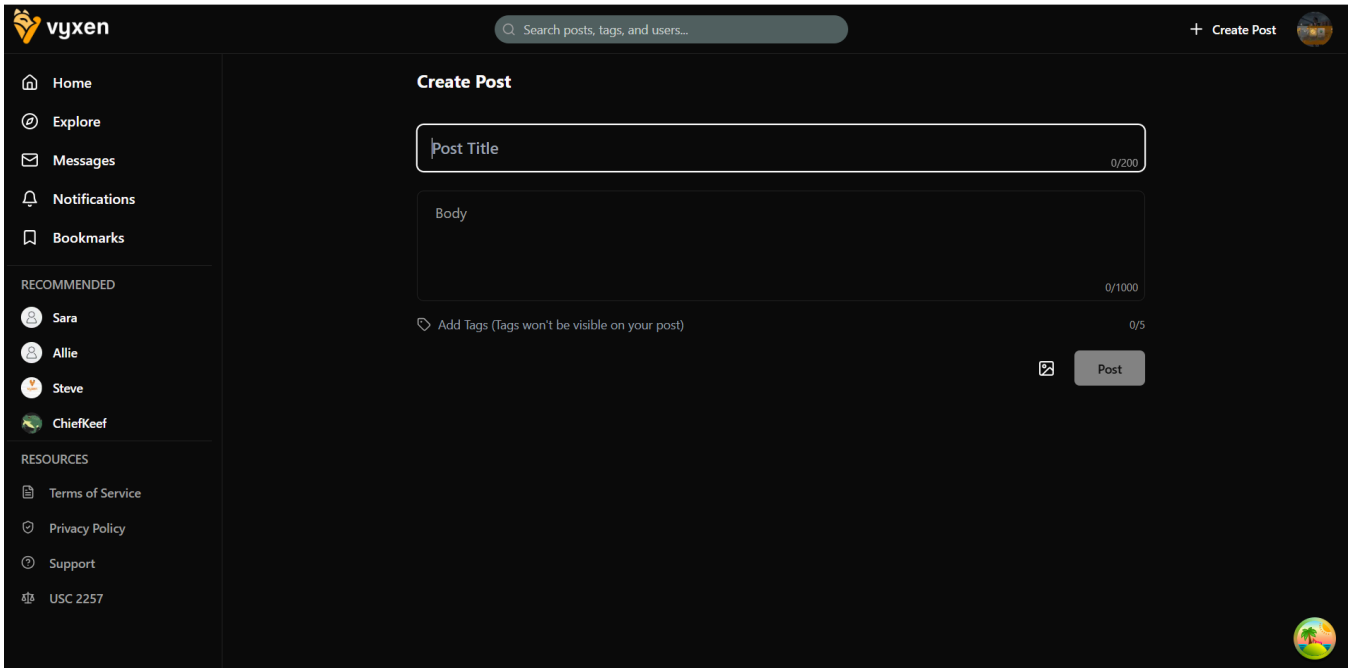
1. **Multiple Authentication Methods:** Users can sign up using email/password or Google OAuth integration
2. **Username or Email Login**: The login system supports either username or email for authentication
3. **Secure Password Requirements**: Password validation with minimum 8 characters, at least one capital letter, and one number
4. **Argon2 Password Hashing**: Implementation of secure password hashing using the Argon2 algorithm
5. **Server Side Validation**: Comprehensive validation on both client and server sides
6. **Unique Username Checking**: Checks to prevent duplicate usernames / emails
7. **Session Management**: Secure session creation and cookie management using the Lucia authentication library

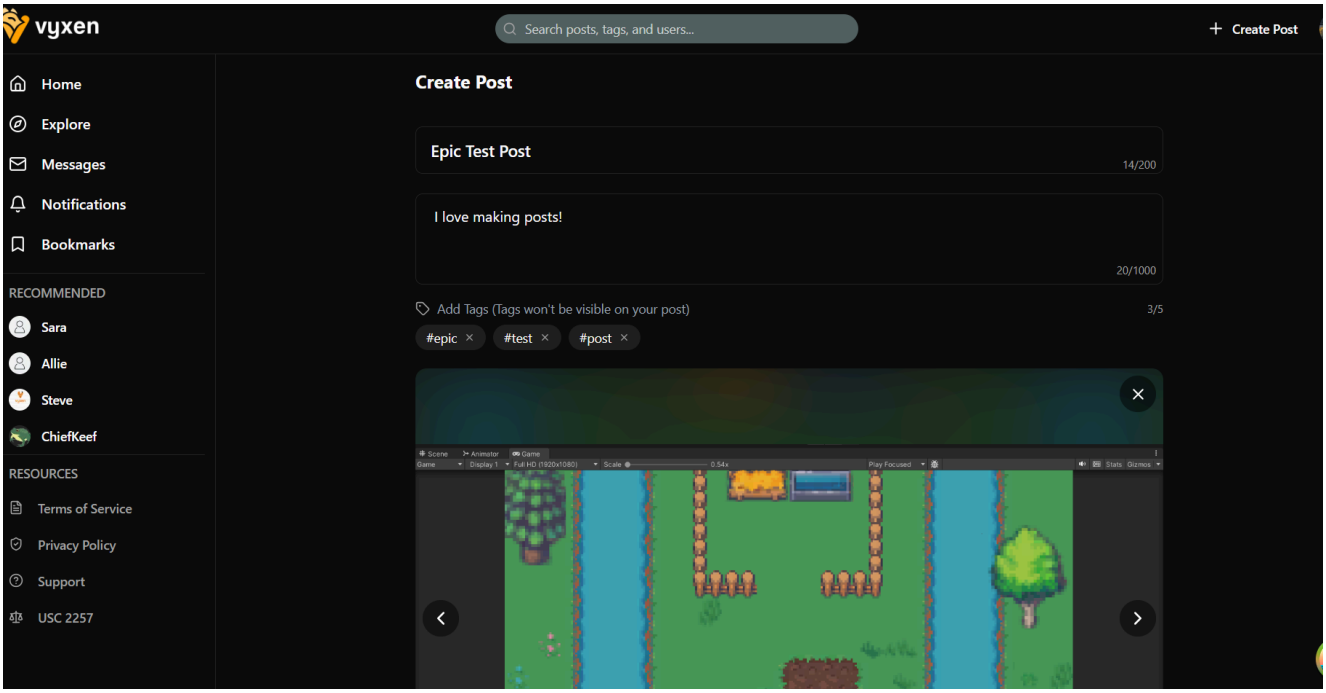




Content Creation:

1. **Post Structure:** Posts are created with a title field, body field and allow for media to be uploaded. Hidden tags can be added to the post as well to help with discovery and the for-you algorithm.
2. **Character Limits**: Title limited to 200 characters, content limited to 1,000 characters. Visual feedback showing character usage with warnings when approaching limits.
3. **Auto-formatting**: Sanitizes excessive spacing and line breaks and automatically detects and formats urls in posts.
4. **Multi-media Uploads**: Users can attach up to 20 images and 5 videos per post. Accepts various file formats including JPEG, PNG, GIF, WebP, SVG, MP4, MOV, and WebM.
5. **Media Preview**: Real-time preview of uploaded media with carousel navigation for multiple files.
6. **Paste Support**: Clipboard image pasting directly into the post editor.
7. **Validation Feedback**: Clear error messages for content limits and media restrictions such as file size too large or too many pieces of media on one post.
8. **Full-screen Viewer**: Expandable images for detailed viewing.
9. **Gallery Navigation**: Intuitive controls for browsing multiple images within a post.
10. **Video Playback**: Native controls for video content with autoplay options.
11. **Lazy Loading**: Videos and images load only when needed.
12. **Debounced Actions**: Performance optimization for engagement tracking.
13. **Optimistic Updates**: UI updates immediately before server confirmation.
14. **Server-side Validation**: All user input validated both client and server side.
15. **Access Control**: Authentication checks before post creation.
16. **Content Sanitization**: Input cleaning to prevent XSS and injection attacks.





Post Page:

Full Post View: While viewing posts on the feed, if a post has media and long text, they won't be able to view the entire post straight from the feed. You can click on posts to be brought to the posts dedicated page to view the whole post and comment easier.

Comment System:

- **Threaded Conversations:** Rich comment system with support for nested replies up to 3 levels deep.
- **Real-time Interactions:** Comments can be liked, with instant UI updates via optimistic rendering.

Reply Tagging: When replying to a comment, the username is automatically added with @ mention.

Expandable Threads: Collapsible reply sections with count indicators for improved readability.

Character Limits: 300 character limit for comments with real-time character counter.

User Engagement Tools:

Like System: One-click like/unlike functionality with visual indicators and counters.

Bookmark Feature: Save posts for later reference with persistent state across sessions.

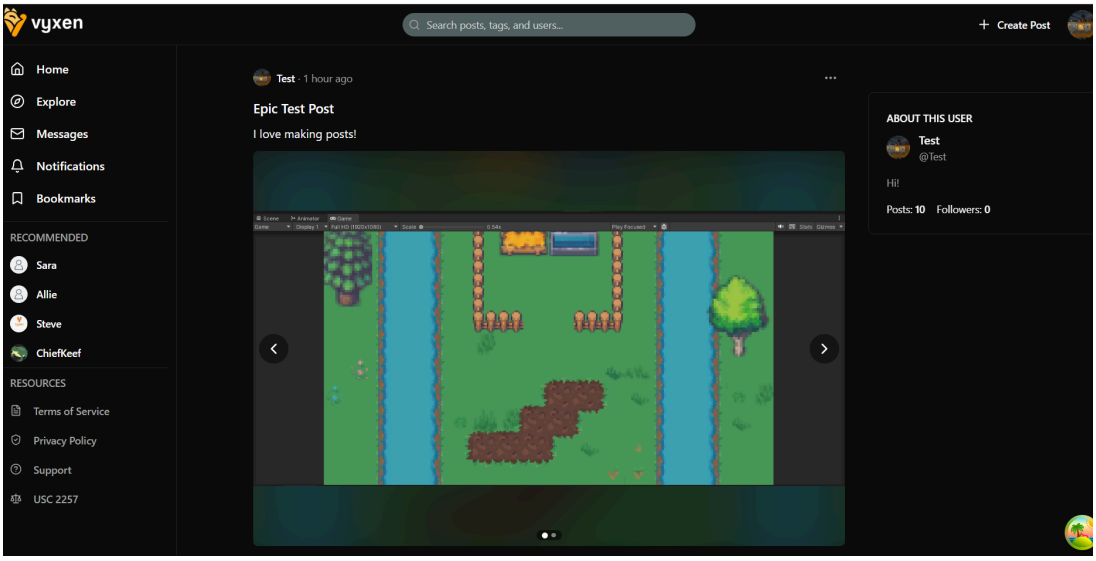
Comment Count: Real-time comment counters showing engagement levels.

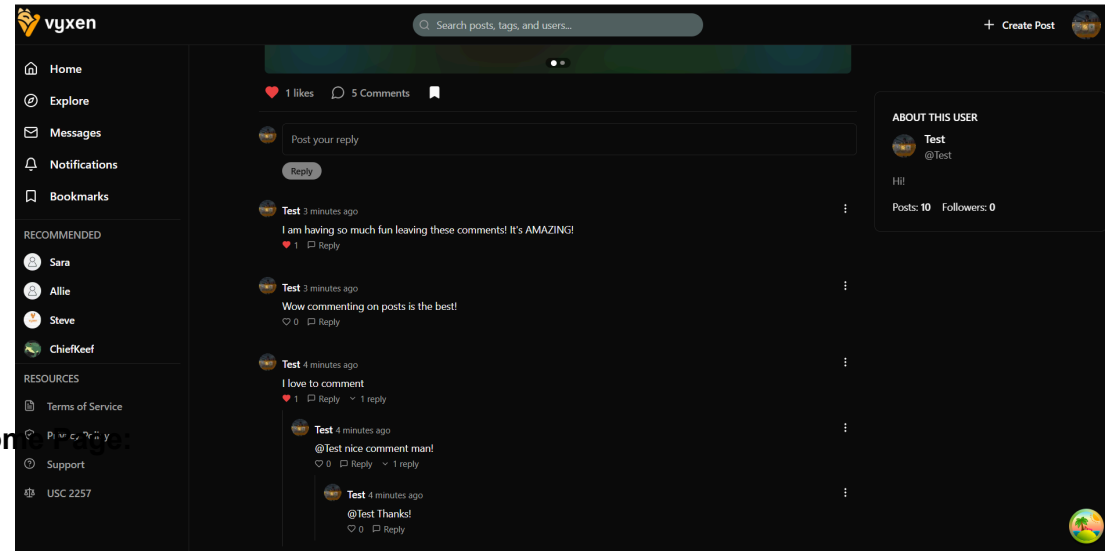
Engagement Analytics: All likes, comments, bookmarks, and view time are taken into consideration to figure out the quality of a post and who would be interested in the post. We track all post engagement to make the for-you page as addictive and interesting to users as possible.

Infinite Scrolling: "Load more comments" functionality for posts with a large comment section.

Content Moderation: Users and site admins have the ability to delete posts and have moderation abilities over all comments. Other users can report inappropriate content as well.

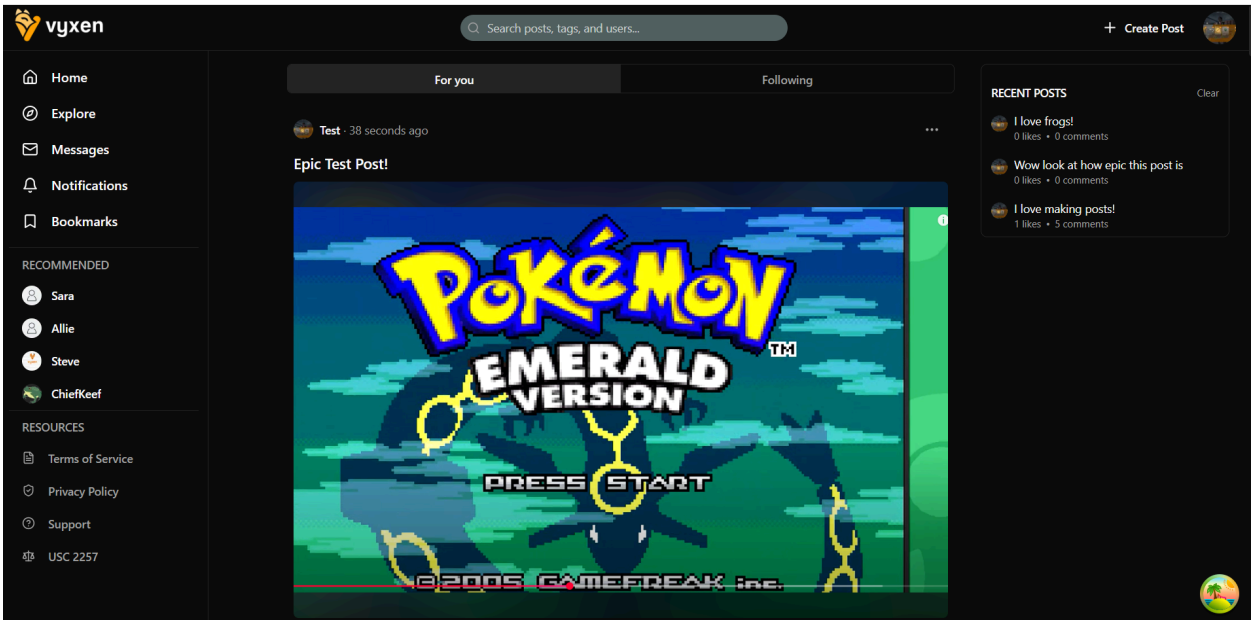
About This User Section: Sidebar with quick access to post author's profile with follow/unfollow functionality. This sidebar shows some information about the post author including follower and post count.

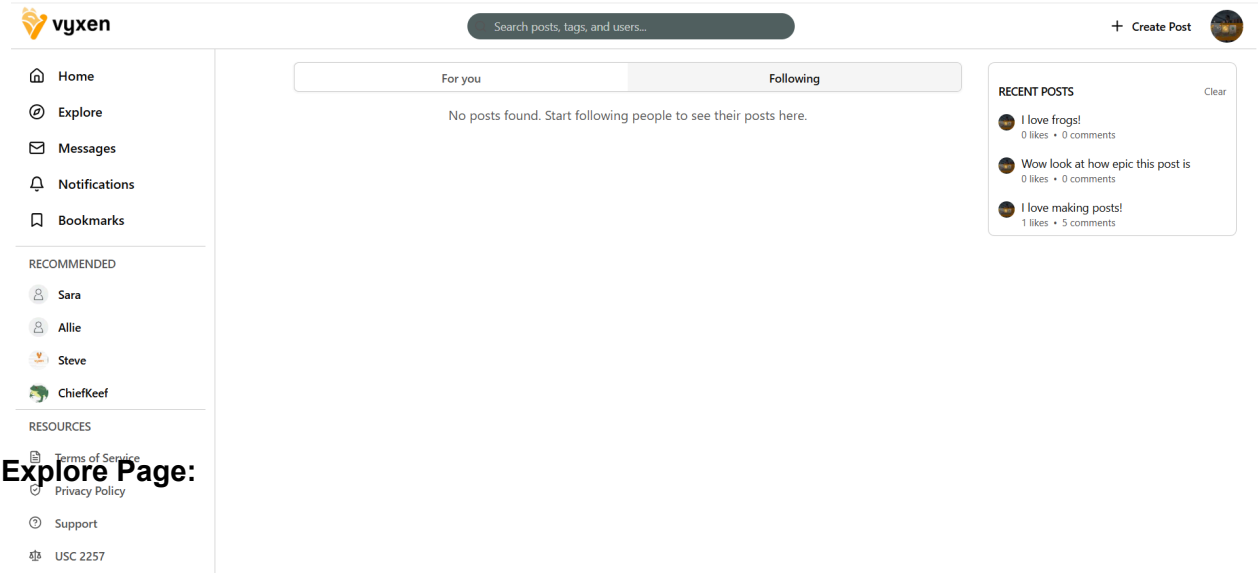




Home Page:

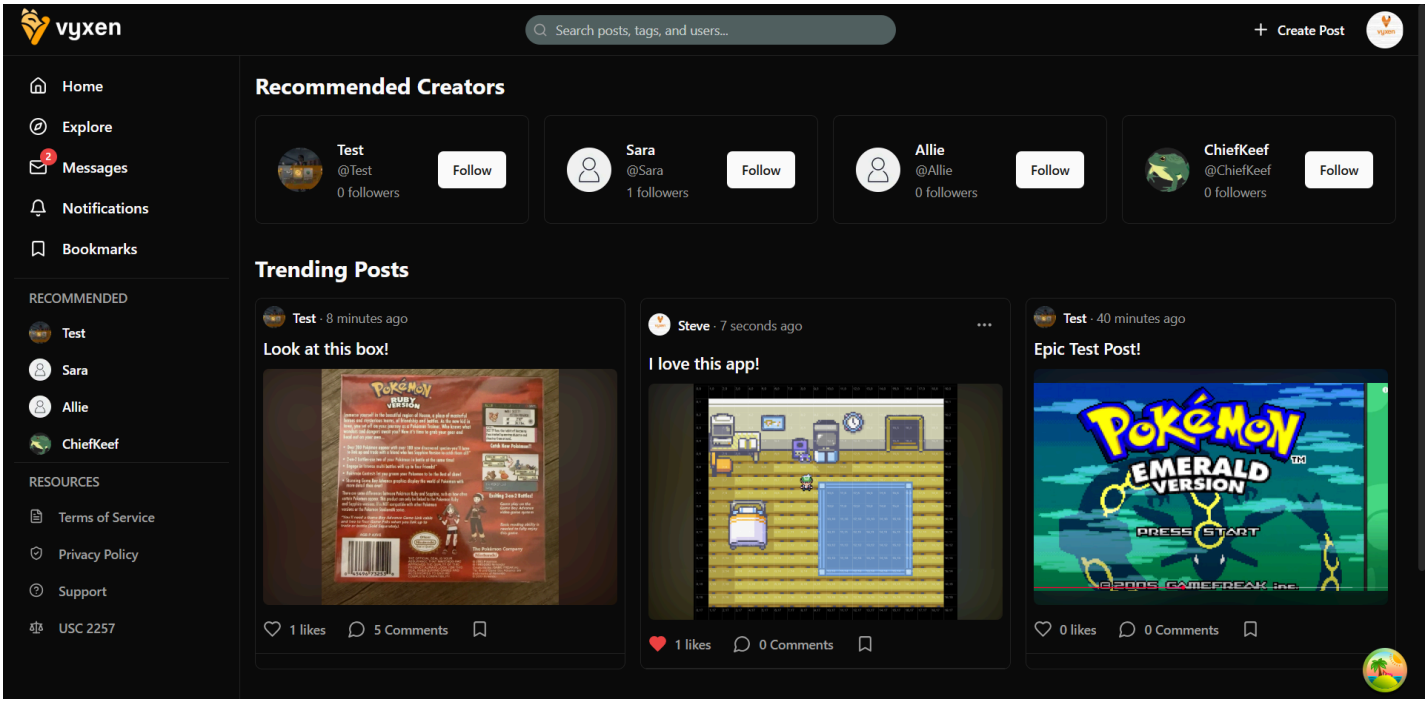
Dual Feeds: At the top of the home page the user can switch between the “For You” feed and “Following” feed. The for you feed will show personalized posts the algorithm thinks the users will enjoy. The following feed will show posts from followers in order from newest to oldest.
Recent Posts Sidebar: Sidebar panel showing recently viewed posts for quick navigation. Only appears on the desktop view.
Infinite Scrolling: Seamless content loading as the user scrolls, no pagination.
Sophisticated For you Feed: For you feed uses a sophisticated algorithm that rapidly scores thousands of posts in order to keep users entertained and engaged. Content is ranked on user interactions and preferences, creator affinity, post affinity, and recency. Likes, comments, saves, watch time are all taken into consideration when scoring posts. Each post has a recency score that gradually decays over time. New users with limited history have slightly modified post scoring. The for you feed uses multiple content diversity mechanisms to ensure the feed isn’t dominated by one creator / interest.
Engagement & Preference Learning: System automatically learns user interests from their interactions. Interest scores decay over time to keep preferences current. Different interactions have varying impacts. Post view time is accounted into users interests as well (the longer a user looks at a post the more likely they are to see similar content.)





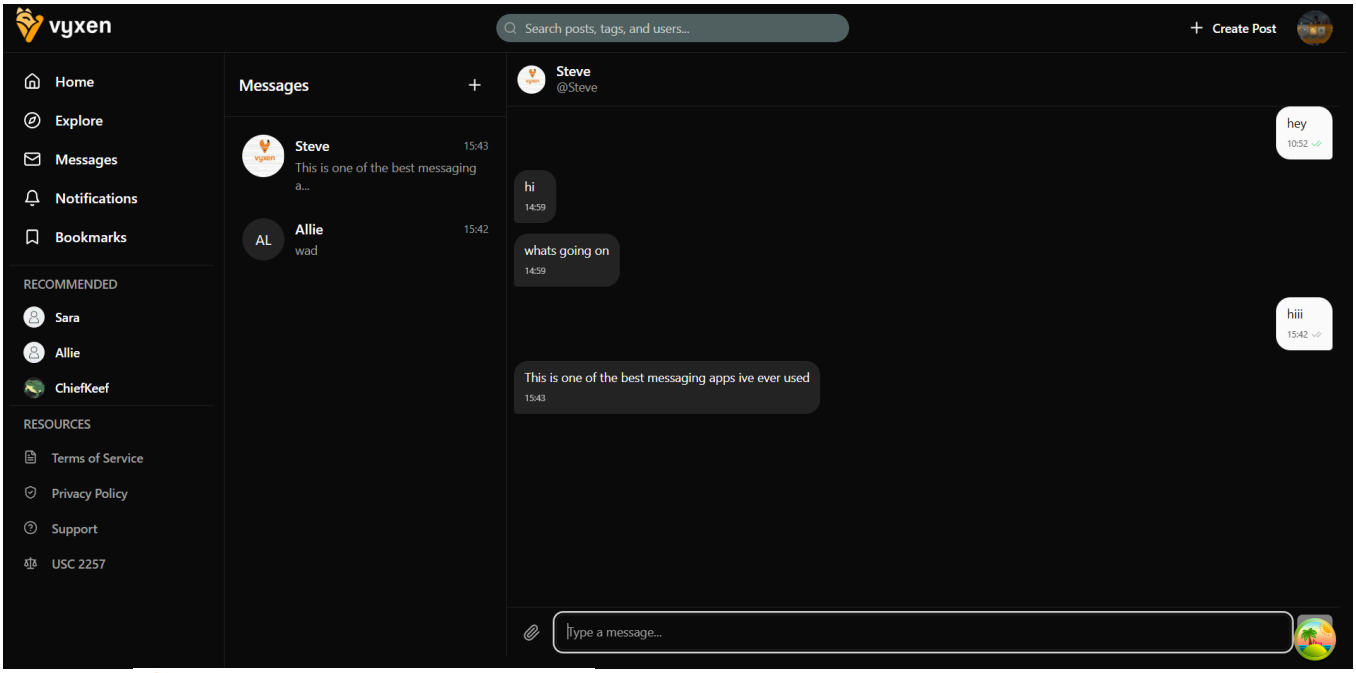
Explore Page:

Recommended Creators Section: Personalized creator recommendations based on user interests and creator affinity. Clean grid layout displaying creator cards with profile information and follower counts.
Trending Posts Section: Real-time aggregation of the most popular and engaging content across the platform.
Popular Creators Section:
Shows users popular creators based on recent growth and activity levels. Algorithm picks popular creators based on follower count, engagement rates and post performance.

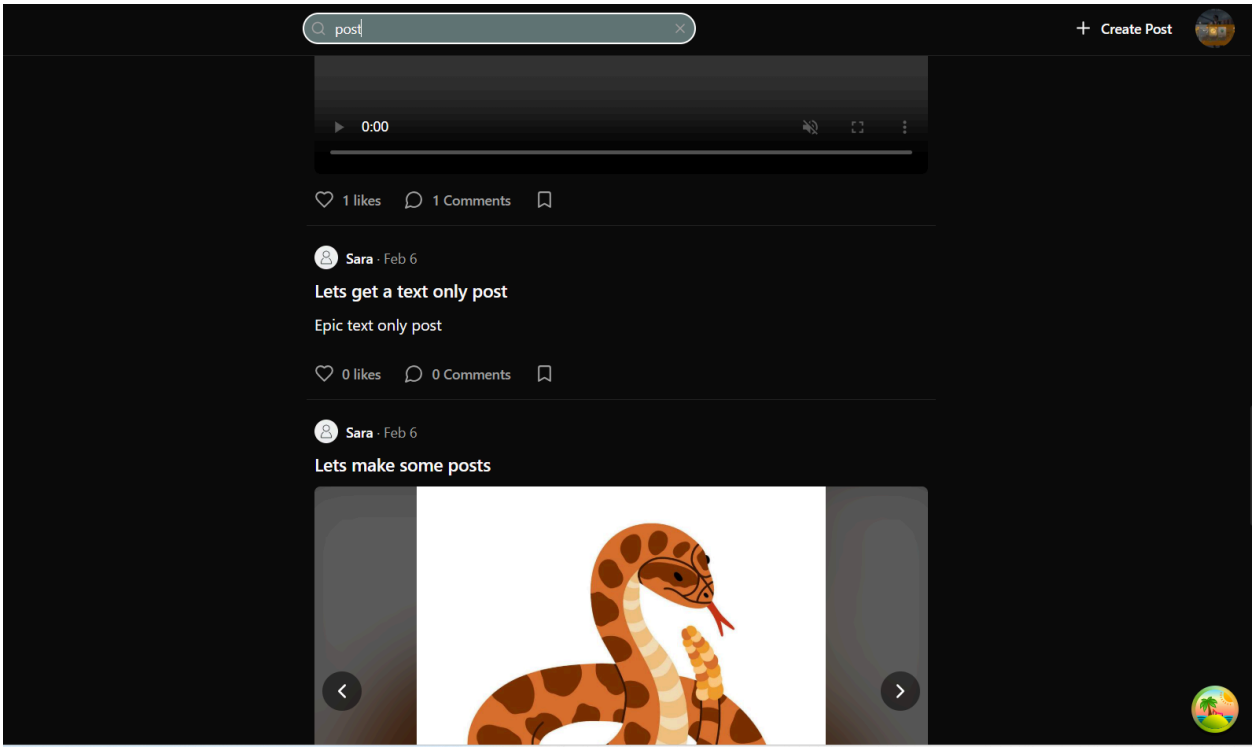


Massaging System:
Serverless WebSocket Architecture: The messaging infrastructure is built on AWS API Gateway WebSockets backed by AWS Lambda functions that handle connection events and message routing. The system uses AWS Lambda handlers for connection management, disconnection cleanup, and message distribution.
AWS DynamoDB Real-time Storage: Messages are persisted in DynamoDB tables with optimized access patterns using Global Secondary Indexes (GSIs) for efficient retrieval. The system uses separate tables for connection management and message history with TTL for automatic cleanup of stale data.
Real Time Communication: Messages are delivered instantly through a WebSocket connection with visual status indicators (sending, sent, delivered) providing immediate feedback. The system automatically reconnects if connection is lost.
Message Status & Typing Indicators: Each message displays its current status through an icon, showing if it's been delivered or read. Timestamps are included with each message. Real time typing indicators show when other participants are composing messages.
Direct & Group Conversations: Support for both private messaging and group conversations with multiple participants. Group chats show sender information with each message and support up to 20 participants. The creator of the group conversation can name the chat and kick / add members over time.

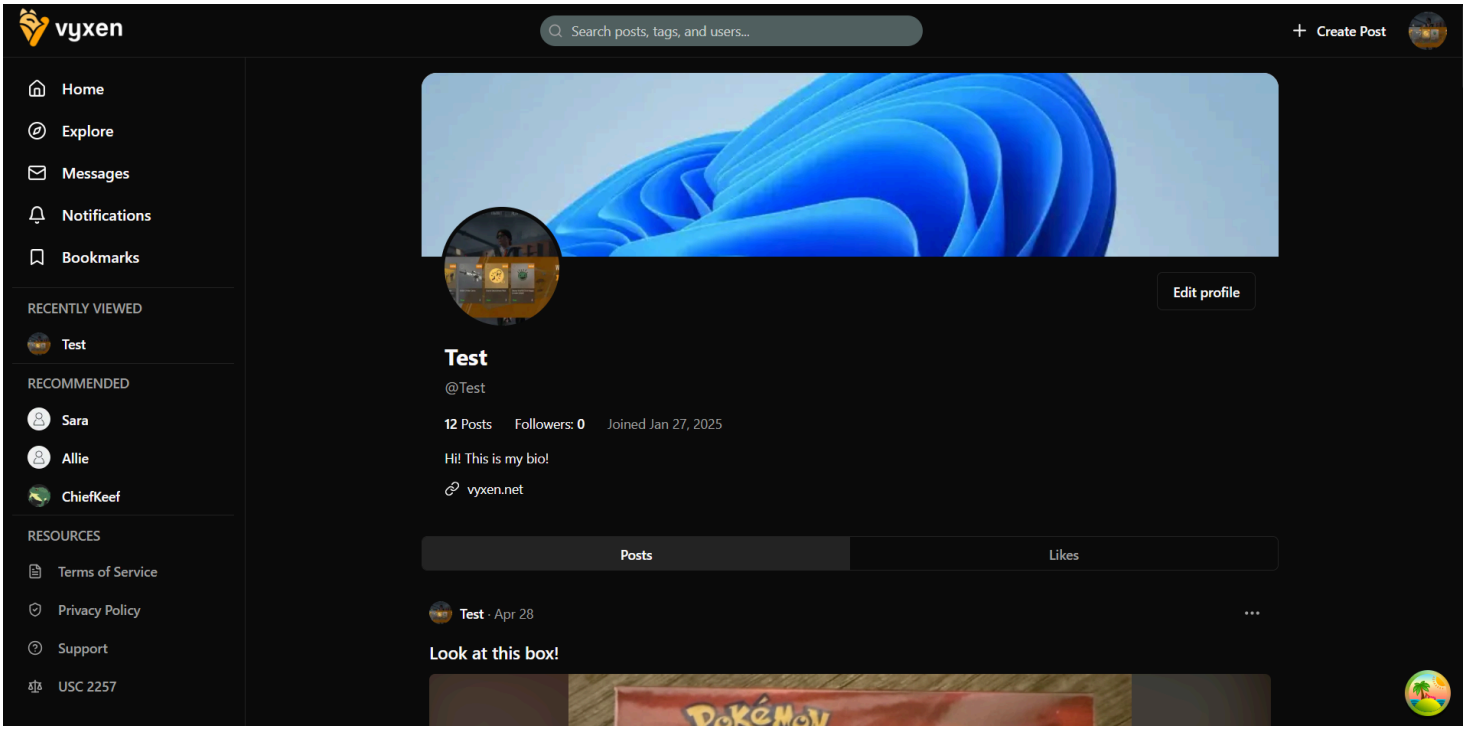
Media Sending: Users can easily share images, videos, and file attachments within conversations. Media previews are shown inline within the chat and users can choose to download media sent over the messaging system.
Message Navigation: Infinite scroll with automatic pagination for message history and a quick-scroll button that appears when users scroll up in conversation history.
Unread Message Management: Visual indicators show unread message counts both in the message list and within the app's navigation. The system automatically marks messages as read when viewed.
Chat List Organization: The chat list automatically sorts conversations with the most recent activity at the top. Each chat preview shows the last message content and timestamp along with unread indicators where applicable.

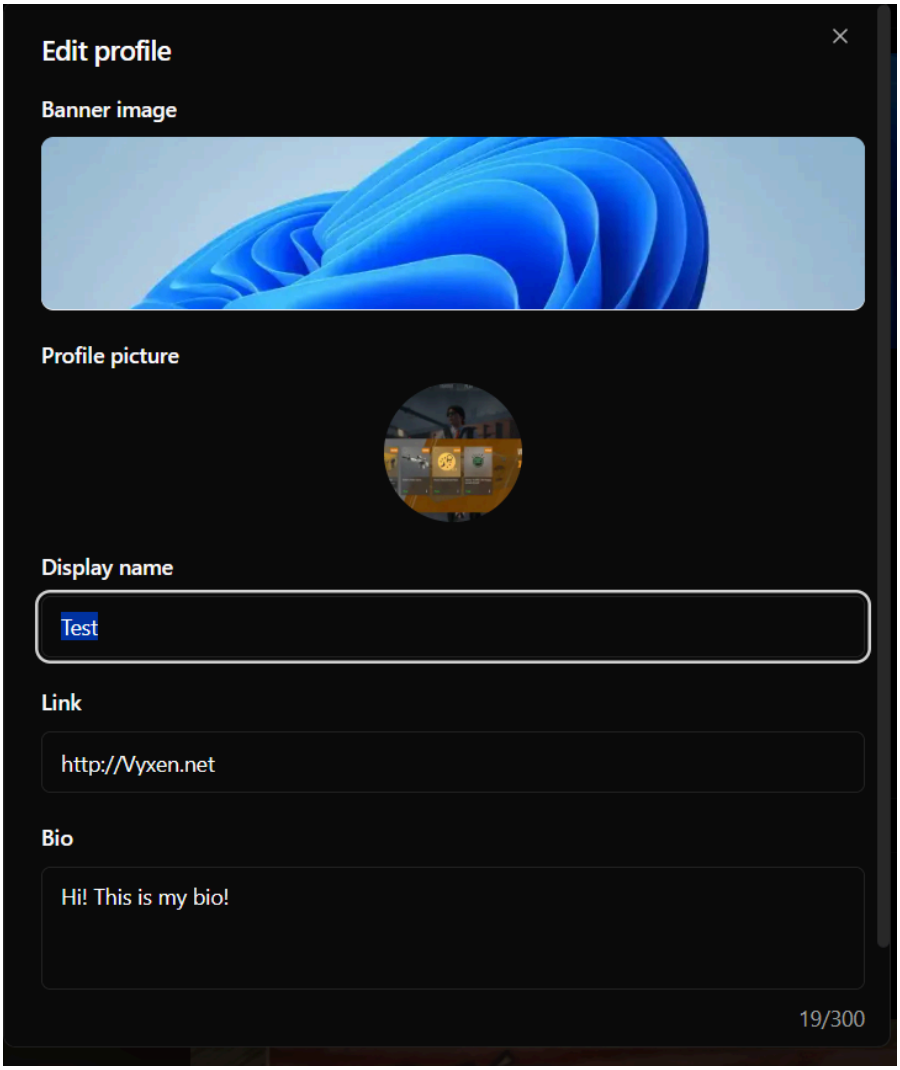


Search System:
Global Search: Users can search for posts, tags, and user profiles all from a single interface that appears in the navigation header.
Advanced Query Processing: Backend search utilizes PostgreSQLs full text search capabilities through Prisma ORM. The system performs searches across multiple data points including post titles, content, tags, and user information for comprehensive results.
Performance Optimized Backend: The search architecture uses database indexing on searchable fields to ensure fast query performance even with large data volumes.



User Profile Pages:
Content Tabs: Profile pages feature a tabbed interface with 2 separate feeds allowing visitors to easily switch between viewing the users content and content the user has liked.
Profile Customization: Each user profile can be customized with a large customizable banner image and a circular avatar. Users can also add a bio, website link, and change their display name.
Relationship Actions: The profile includes buttons that change based on the relationship between the current user and the profile being viewed. Users can follow/unfollow others with optimistic UI updates that reflect changes immediately.
Profile Analytics Tracking: The system captures profile views through a hook that records engagement metrics when profiles are viewed. This data contributes to the recommendation algorithms.
Posts and Following Tracker: Post and following count tracked and displayed on profile above bio section.


vyxen
Search posts, tags, and users...
Create Post
Home
Explore
Messages
Notifications
Bookmarks
RECENTLY VIEWED
Test
RECOMMENDED
Sara
Allie
ChiefKeef
RESOURCES
Terms of Service
Privacy Policy
Support
USC 2257
Edit profile
Test
@Test
12 Posts
Followers: 0
Joined Jan 27, 2025
Hi! This is my bio!
vyxen.net
Posts
Likes
Test · Apr 28
Look at this box!


Edit profile
Banner image
Profile picture
Display name
Test
Link
http://Vyxen.net
Bio
Hi! This is my bio!
19/300