



PORTFOLIO

Wonhee Lee

Portfolio Website:	whd793.github.io
Resume pdf (Korean):	whd793.github.io/resume-kor.pdf
Resume pdf (English):	whd793.github.io/resume-eng.pdf
Github:	github.com/whd793
Blog:	https://wonblog.vercel.app/
Linkedin:	llinkedin.com/in/whd793
Email:	whd793@gmail.com
Phone:	+82-10-2551-5567

WONHEE LEE

Education		Certificate		Link	
2019.12	Purdue University B.S. - Web Programming & Design	2023.01	JavaScript (Intermediate) Certificate / HackerRank	Portfolio Website:	whd793.github.io
Experience		2023.01	React Certificate / HackerRank	Resume Korean:	whd793.github.io/resume-kor.pdf
		2023.01	React Hooks / LinkedIn	Github:	github.com/whd793
		2021.08	CompTIA Network+ (N10-007) Cert Prep / LinkedIn	Blog:	https://wonblog.vercel.app/
		2021.07	MySQL Advanced Topics / LinkedIn	LinkedIn:	llinkedin.com/in/whd793
		2021.07	NoSQL Essential Training / LinkedIn	Resume English:	whd793.github.io/resume-eng.pdf
		2021.06	Learning SQL Programming / LinkedIn	Email:	whd793@gmail.com
2024.07 ~ 2024.12	Twiqli - Sms Web startup (Fullstack Developer)	2021.05	Advanced Node.js: Scaling Applications / LinkedIn		
2023.08 ~ 2024.03	Pockapp Games - Startup (Game Developer)	2021.01	React.js Essential Training / LinkedIn		
2022.07 ~ 2023.07	Voodoo.io - Game Developer				
2020.01 ~ 2020.03	Gom Shabu - Freelance - Frontend Developer				
2019.06 ~ 2019.10	PRPC - Freelance Frontend Developer				

SKILLS

Frontend

- HTML, CSS3, Javascript, React.js, Redux, Typescript, Next.js, Tailwind CSS, Shadcn, Zustand

Backend

- Node.js, Express.js

DB

- MongoDB, AWS, PostgreSQL, MySQL, Firebase, Redis, Prisma, DrizzleORM, GraphQL

Others

- OpenAI API, Python, C#, Unity, Java, Git, Github, Docker

PROJECTS

- 01 AI Interview & Feedback Assistant 2024.05 - 2024.07 | Frontend & Backend
- 02 PodStream - Podcast Radio Platform 2024.01 - 2024.04 | Frontend & Backend
- 03 SMS Subscription & Marketing Service 2024.07 - 2024.12 | Frontend & Backend
- 04 E-commerce Site 2024.11 - 2025.01 | Frontend & Backend
- 05 Speedy Hero App - 47,000 Installs 2017.01 - 2017.05 | Game Development
- 06 'Hyper Casual Game' Projects 2023.08 ~ 2024.03 | Game Development

01 AI Interview & Feedback Assistant

<https://ai-interview-app-u24p.vercel.app/>

Problem

During my job search, I experienced firsthand the importance of interview preparation and the lack of effective practice methods. The biggest challenge was obtaining objective feedback when practicing alone and the limitations in preparing job-specific questions.

Solution

To address these issues, I developed an AI-powered mock interview platform. Rather than simply repeating predetermined questions, the goal was to provide customized questions based on the user's job and career level, and to offer real-time feedback.

Development

Period:

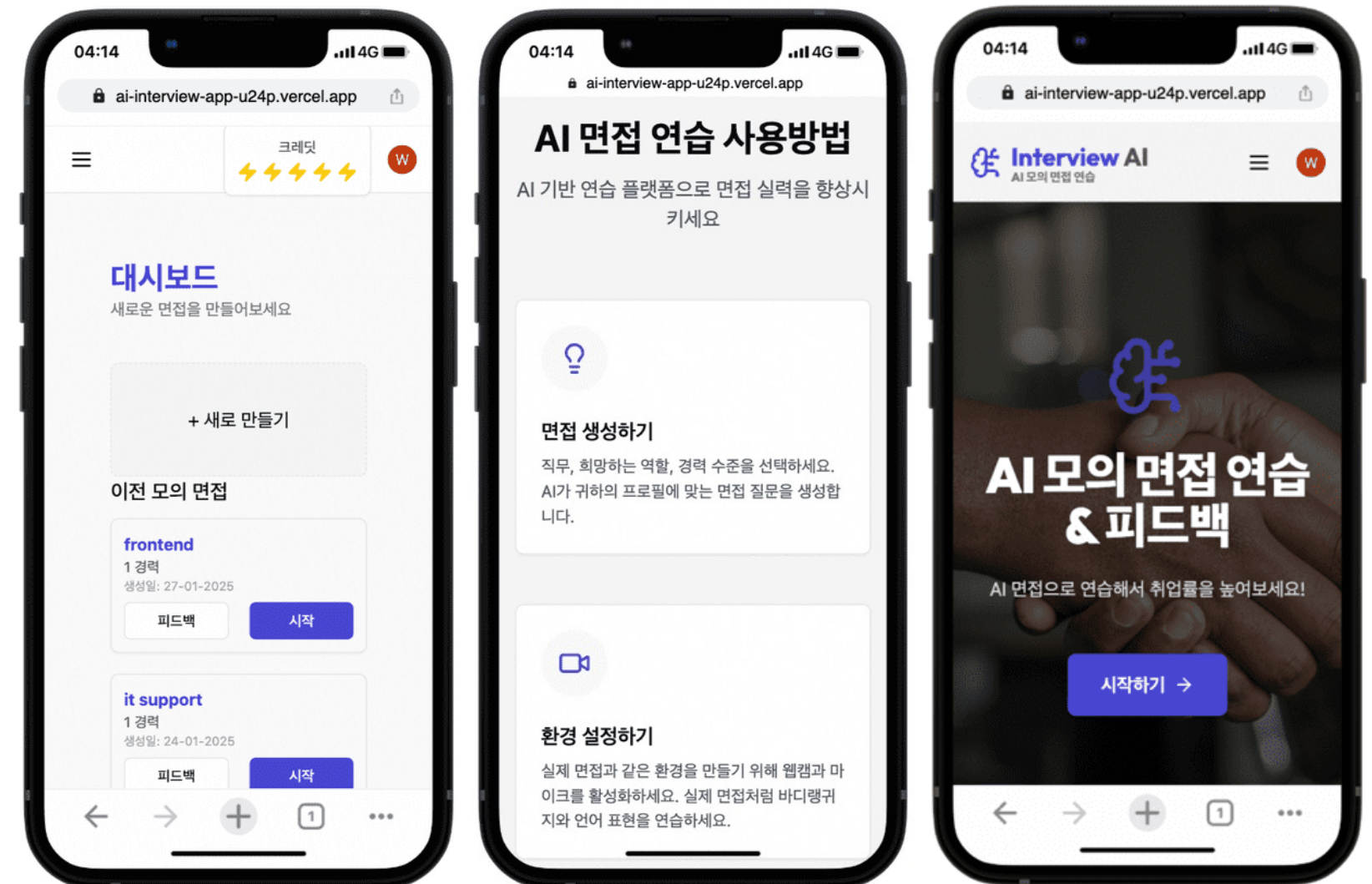
- Planning, design: 2024.05.01 ~ 2024.05.15
- Development: 2024.05.16 ~ 2024.06.28
- Testing and deployment: 2024.06.29 ~ 2024.07.05

Team:

PM, Designer, Developer)

My Role:

- Project Manager, Frontend, Backend
- Service planning and direction setting
- Main logic implementation (frontend, backend), API design, DB architecture



01

AI Interview & Feedback Assistant

Development Environment:

Github <https://github.com/whd793/AI-Interview-App>

Demo <https://ai-interview-app-u24p.vercel.app/>

Frontend

- Next.js 13 (Server-side rendering and routing with App Router)
- React (Component-based UI development and state management with Hooks)
- Tailwind CSS (Responsive design and custom UI implementation)
- Lucide React (Icon system)
- Clerk (User authentication system)
- Shadcn UI (Reusable component library)

Backend

- PostgreSQL (Relational database design and implementation)
- Drizzle ORM (Type-safe database queries)

State Management

- React Context for language settings
- useState for local state management
- useEffect for real-time updates

APIs and Integration

- Google Gemini AI API (Custom interview question generation and answer evaluation)
- Speech Recognition API for voice input
- WebRTC (Real-time voice recognition and processing)

01

AI Interview & Feedback Assistant

Core Features:

Comprehensive Analysis Dashboard:

- Interview data visualization using Recharts library
- Individual interview performance tracking and detailed analysis
- Answer quality evaluation and improvement point derivation
- Job-specific performance analysis and customized learning direction
- Growth process monitoring over time

AI Interview System:

- Custom interview question generation by job type using Google Gemini AI
- Real-time answer collection and text conversion through Speech Recognition API
- Multi-language support (Korean, English, Japanese, Chinese)
- AI-based real-time evaluation and feedback on answers

Interview Management System:

- Past interview storage and review functionality
- Comprehensive feedback history management
- Detailed performance metrics
- Identification of improvement areas and learning direction guidance

Credit System

- User-specific credit management system
- Automatic credit recharge (1 credit per minute)
- Real-time timer implementation using React Hooks and Moment.js
- Credit usage and recharge history management

01

AI Interview & Feedback Assistant

Technical Challenges:

Real-time Processing Optimization:

- Resolving speech recognition and AI response processing delay issues
- Real-time data synchronization using WebSocket
- Browser compatibility issues

State Management Optimization:

- Global state management using React Context and Custom Hooks
- Resolving concurrency issues in the credit system
- Performance improvement through caching strategies

Frontend Optimization:

- Initial loading time reduction through Code Splitting
- Image optimization and lazy loading implementation
- Minimizing re-rendering through component memoization

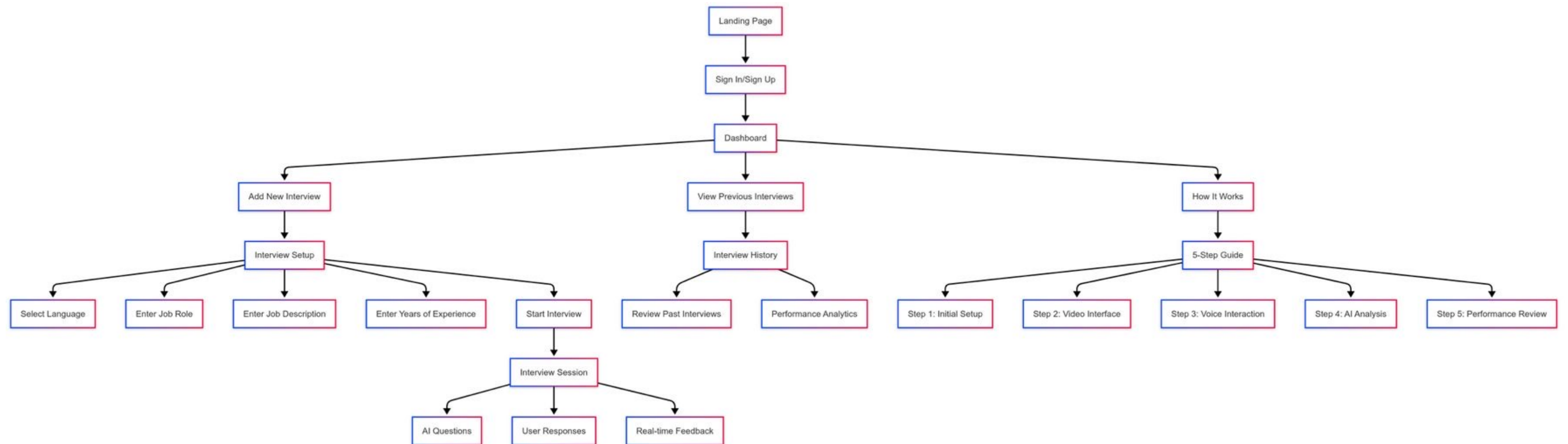
Multi-language Processing:

- Dynamic language switching system implementation
- Optimization of AI model's multi-language processing
- Font and layout adaptation for different languages

01

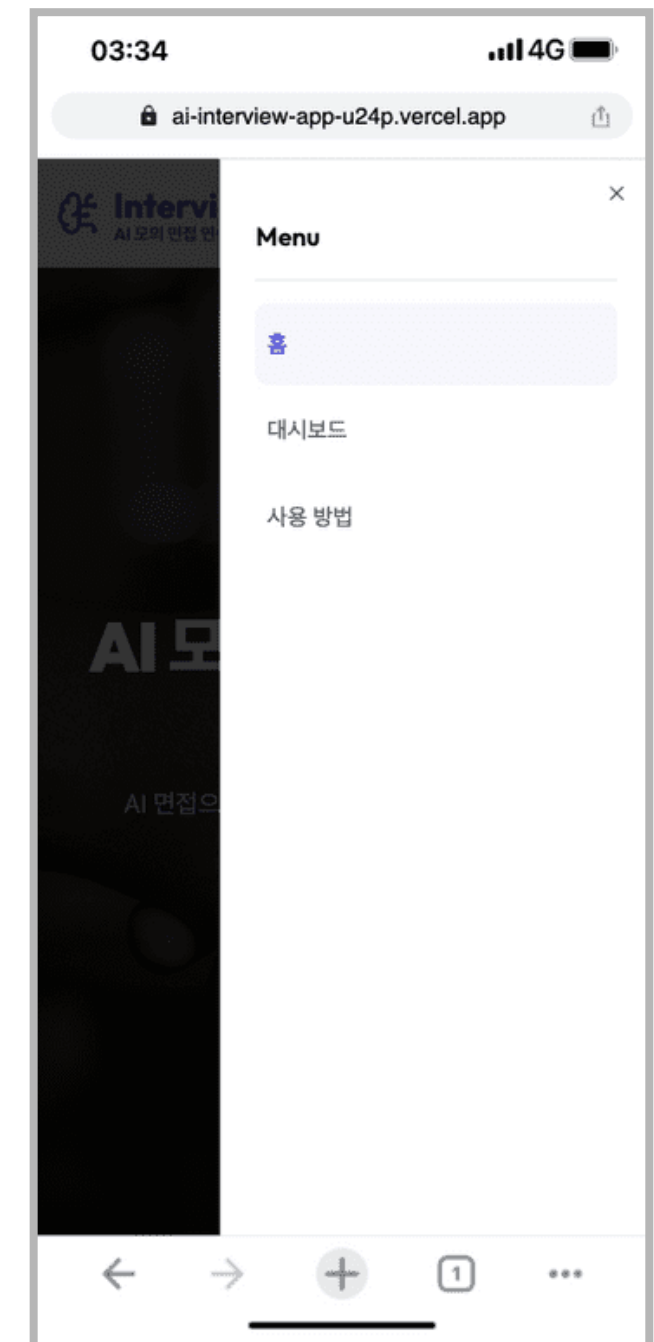
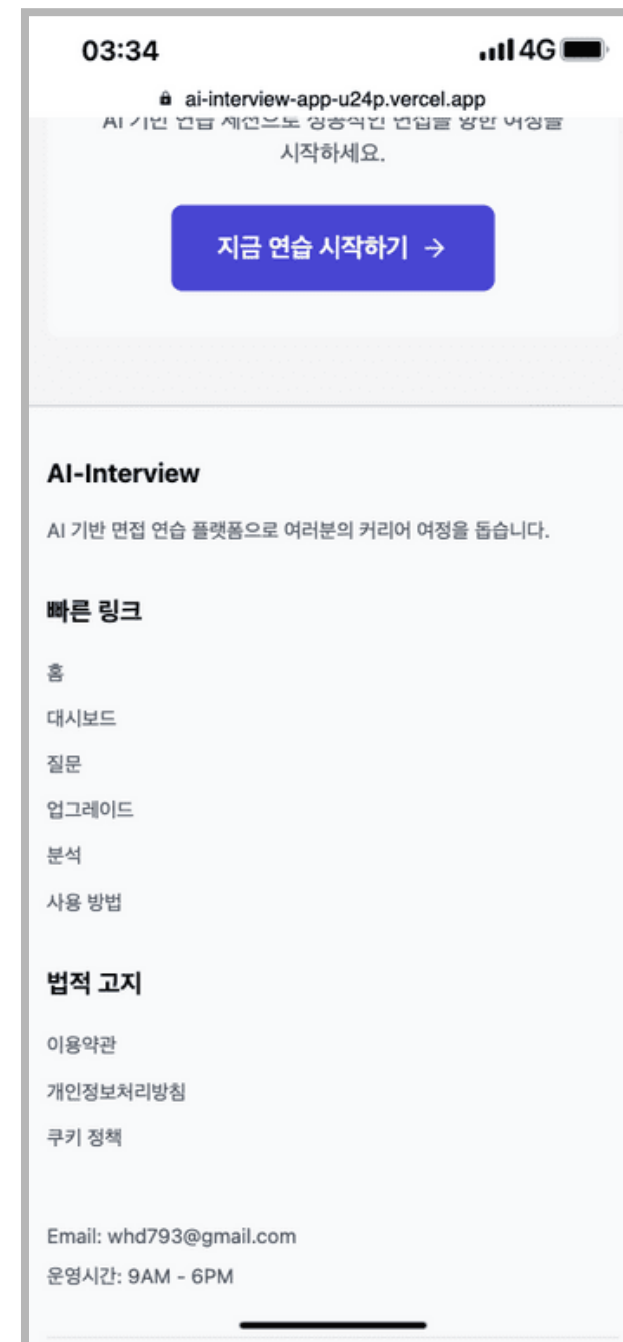
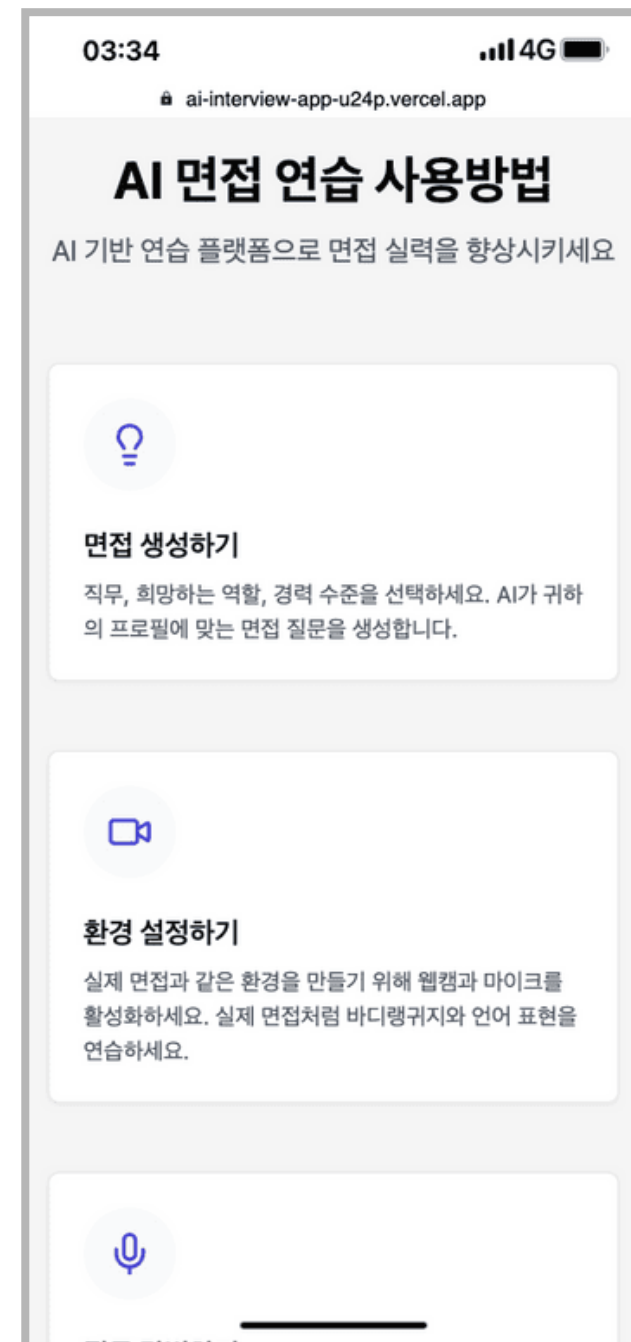
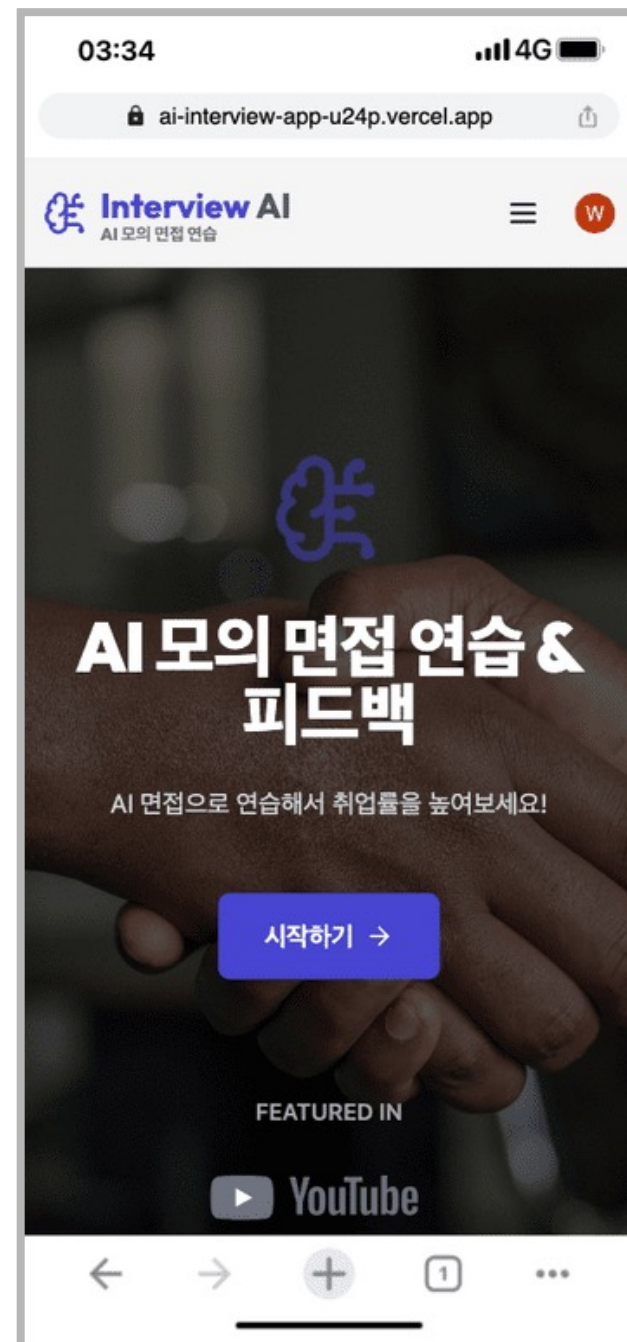
AI Interview & Feedback Assistant

User Flow



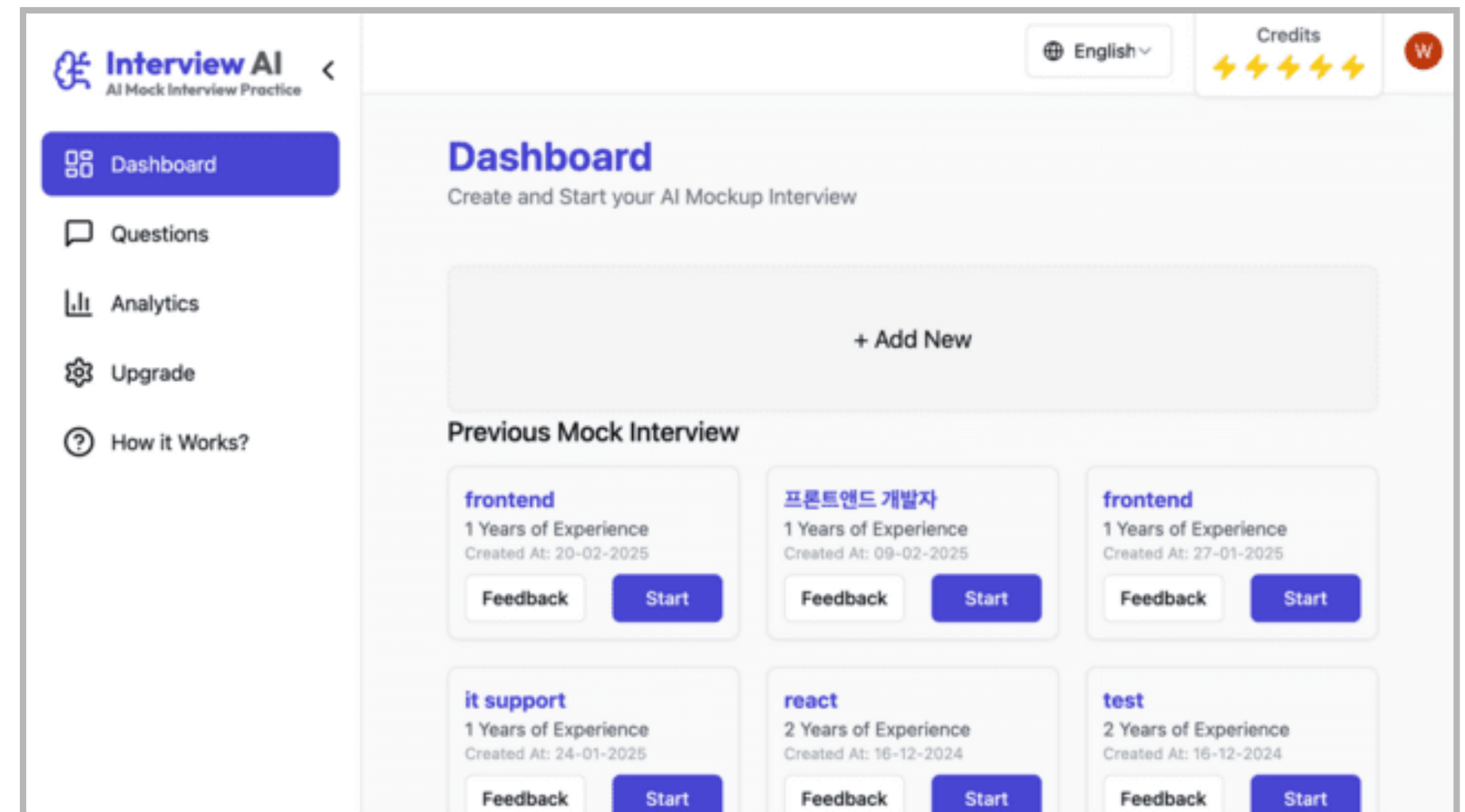
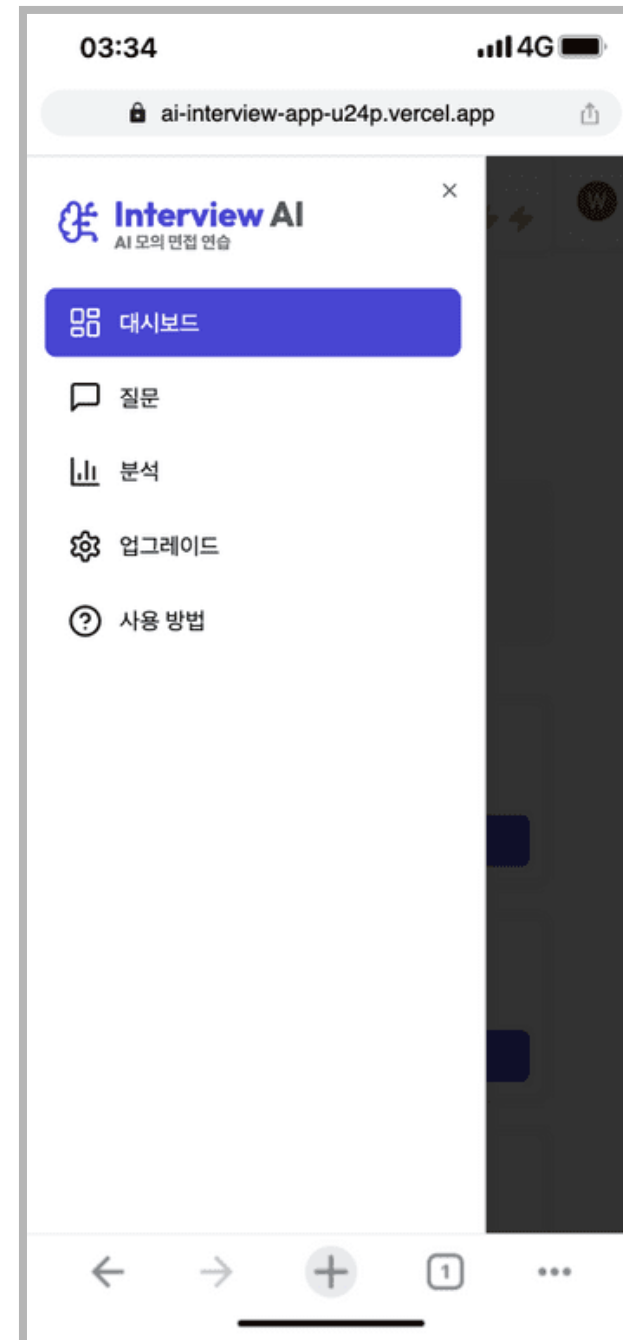
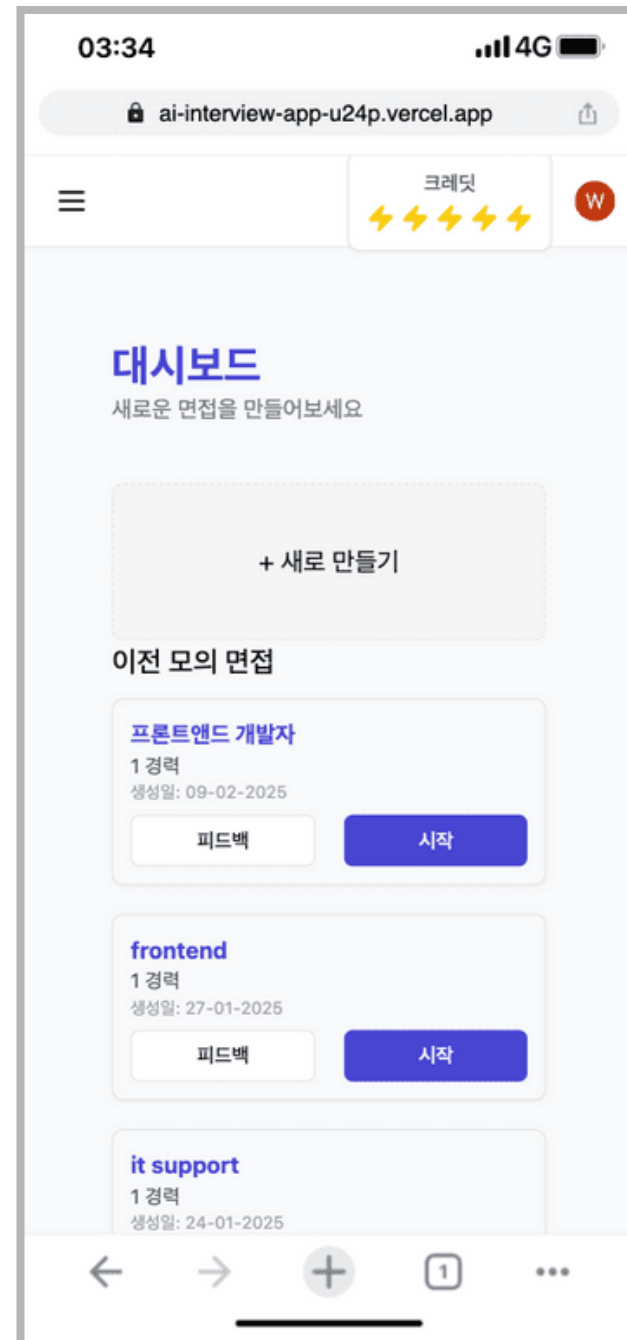
01 AI Interview & Feedback Assistant

Homepage:



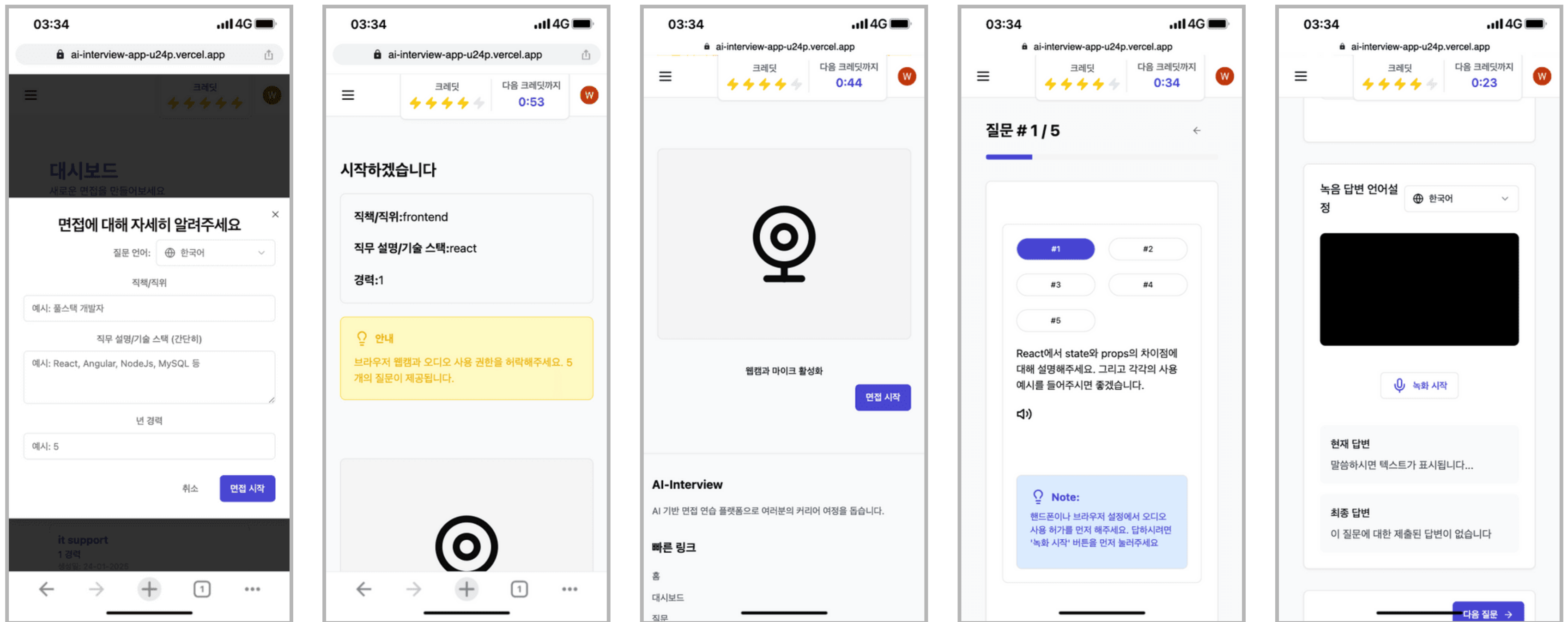
01 AI Interview & Feedback Assistant

Dashboard, Multi-language Support, Credit Management:



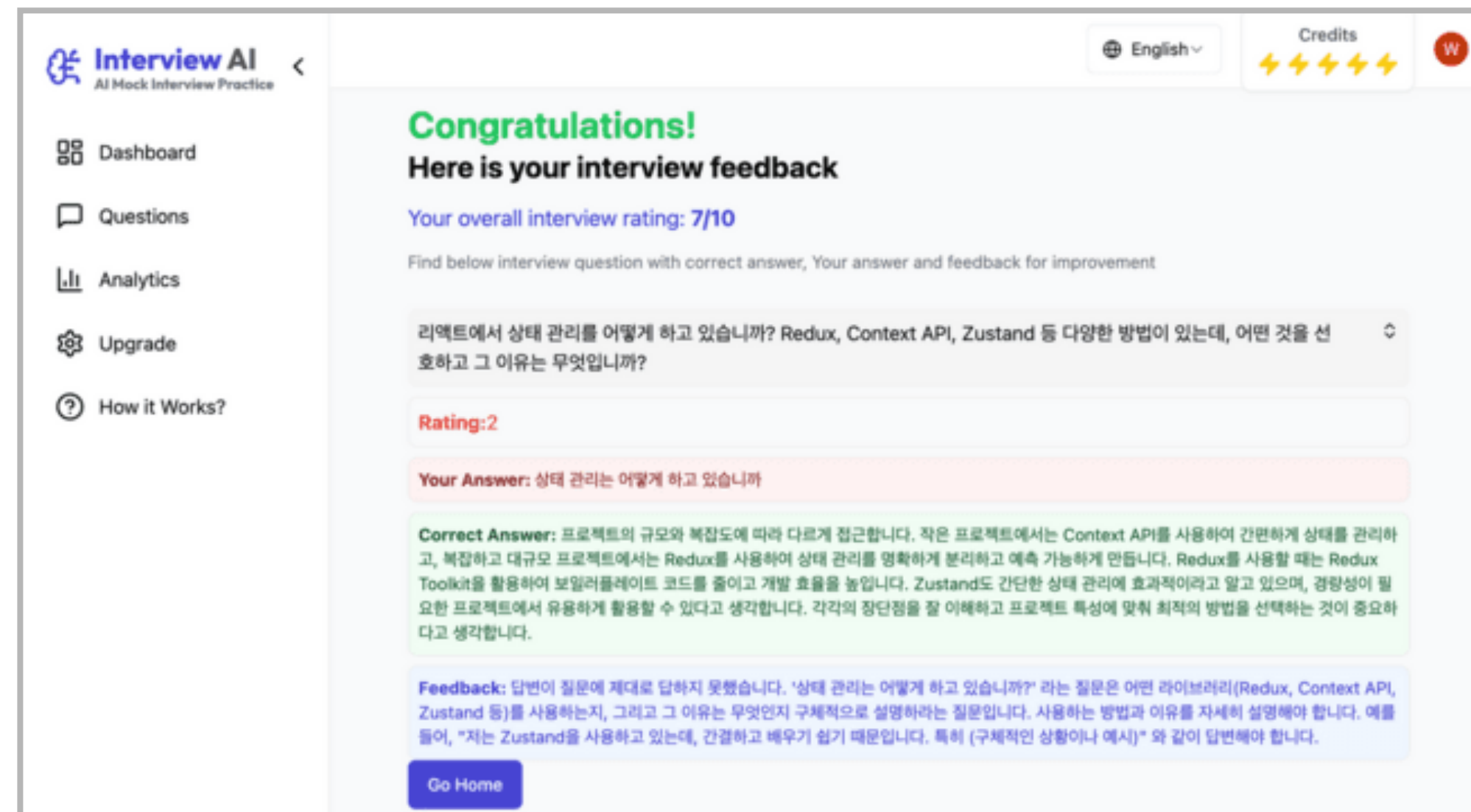
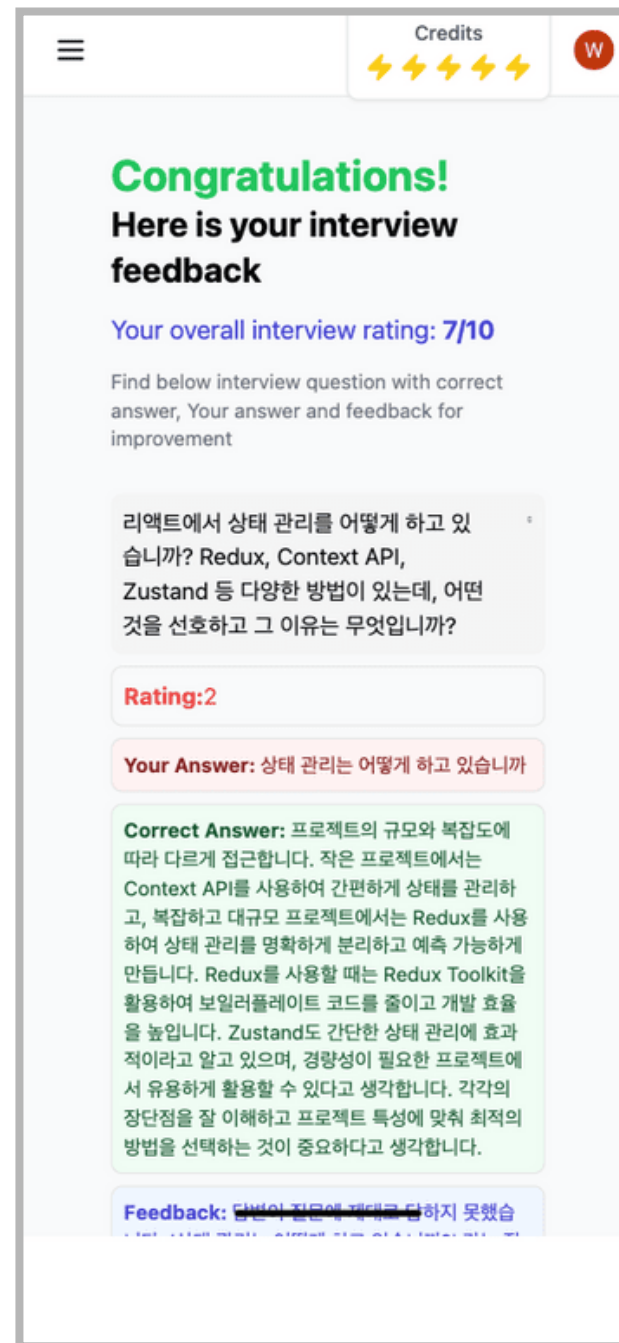
01 AI Interview & Feedback Assistant

AI-powered Job-specific Interview Questions and Voice Recognition:



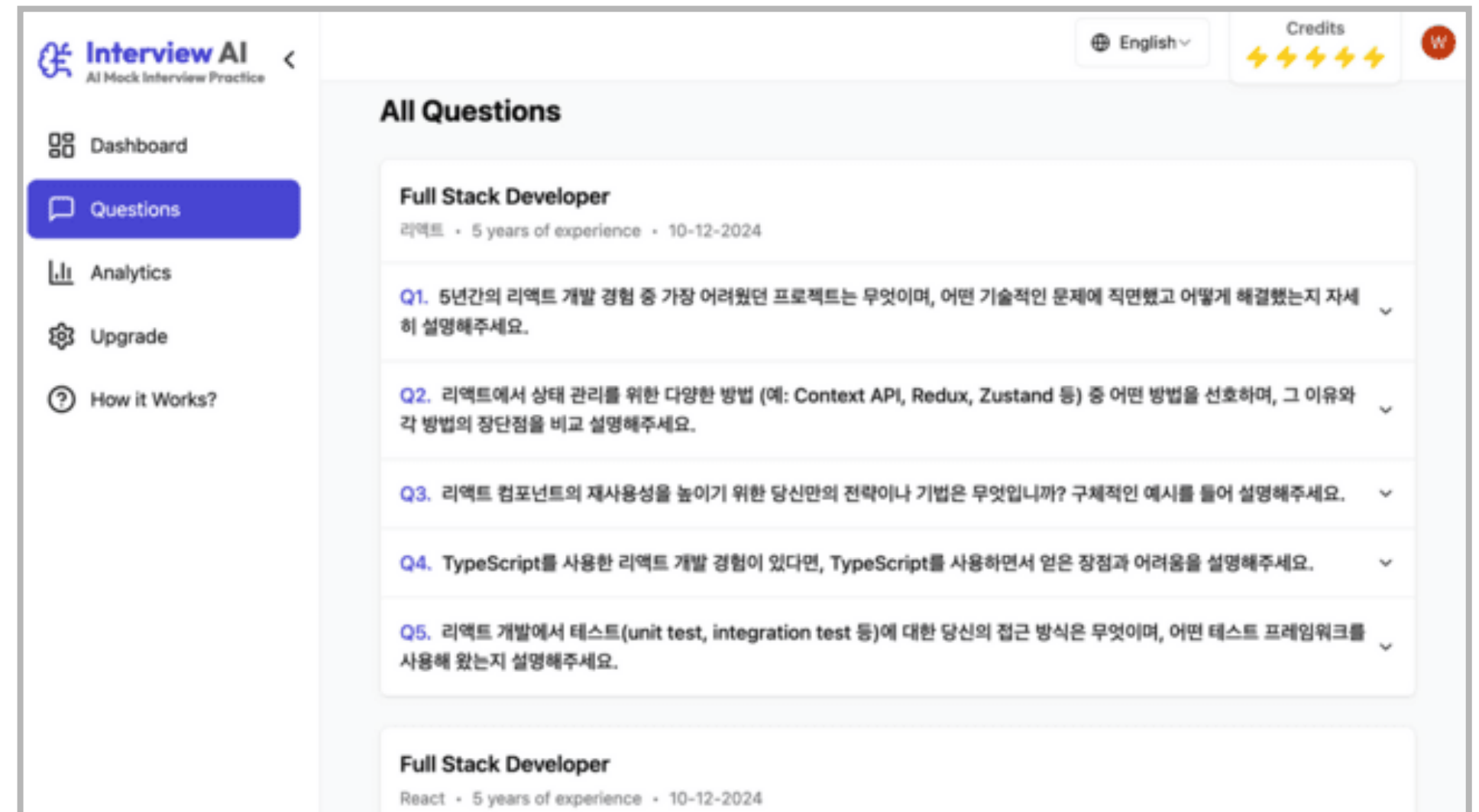
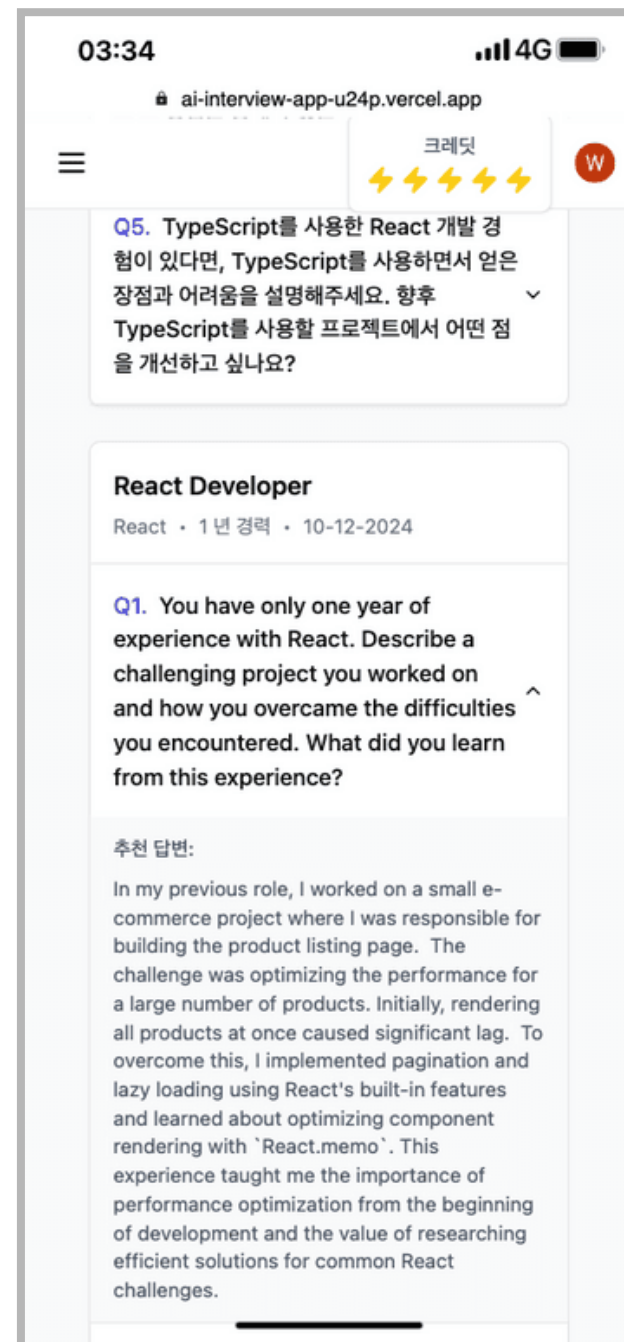
01 AI Interview & Feedback Assistant

AI-based Real-time Evaluation and Feedback:



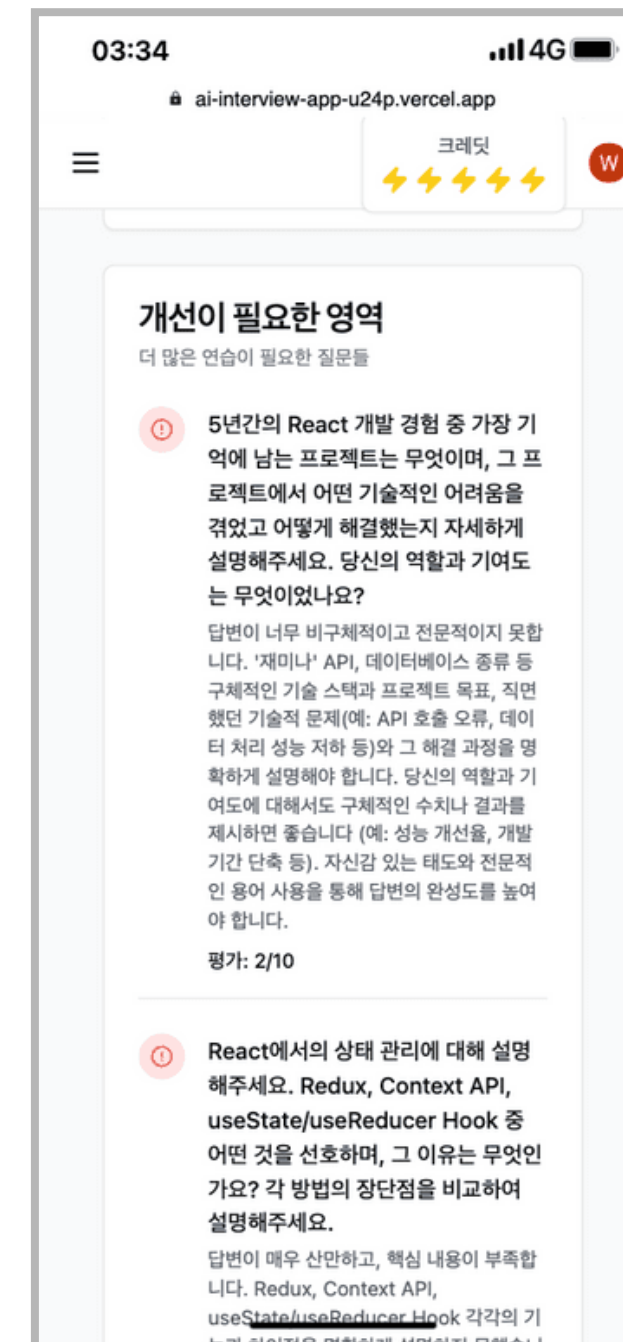
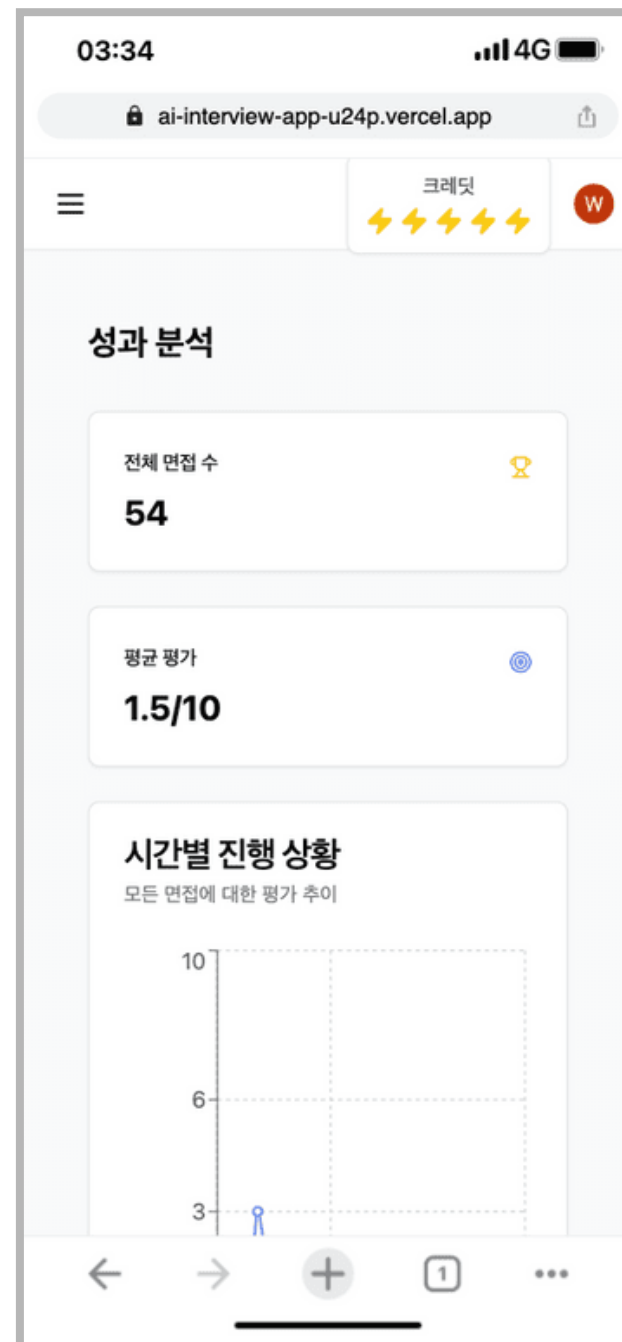
01 AI Interview & Feedback Assistant

Interview History and Feedback Management:



01 AI Interview & Feedback Assistant

Comprehensive Analysis Dashboard and Performance Tracking:

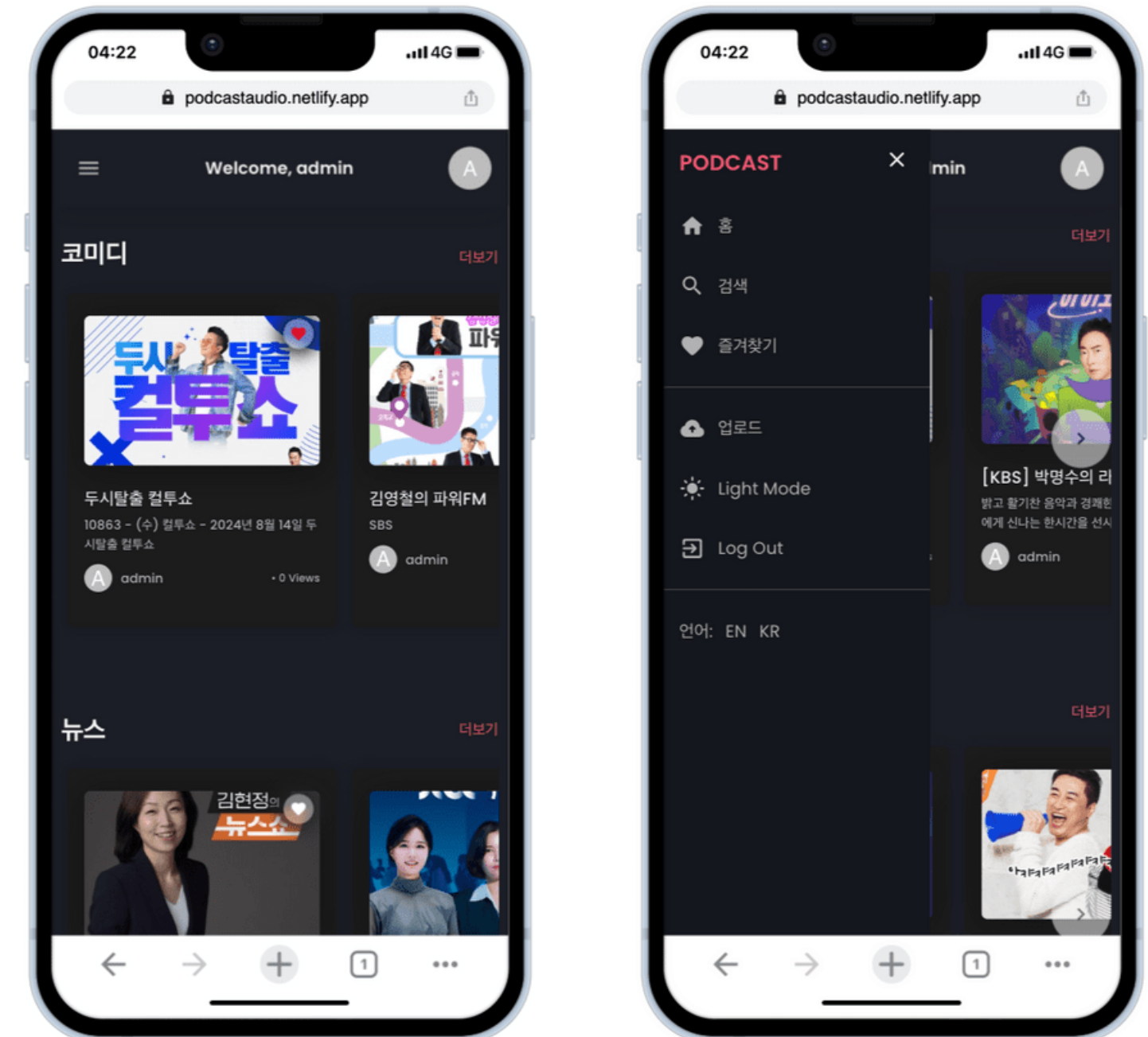


02 PodStream - Podcast Radio Platform

Intro

PodStream is a streaming platform that allows users to easily share and listen to audio and video podcasts. Developed with React and Node.js, it provides an intuitive UI/UX and stable streaming service. Through this project, I was able to apply and learn various technologies and methodologies needed for modern web application development. I particularly focused on user experience and performance optimization to implement an application at the actual service level.

Development Period:	• Planning and design: 2024.01.15 ~ 2024.01.29 • Development: 2024.01.30 ~ 2024.03.30 • Testing and deployment: 2024.04.01 ~ 2024.04.15
Team:	PM, Designer, Developer
My Role:	• Project Manager, Frontend, Backend • Service planning and direction setting • Main logic implementation, API design, DB architecture
Github	https://github.com/whd793/podcast-app
Demo	https://podcastaudio.netlify.app/



02 PodStream - Podcast Radio Platform

Development Environment:

Frontend:

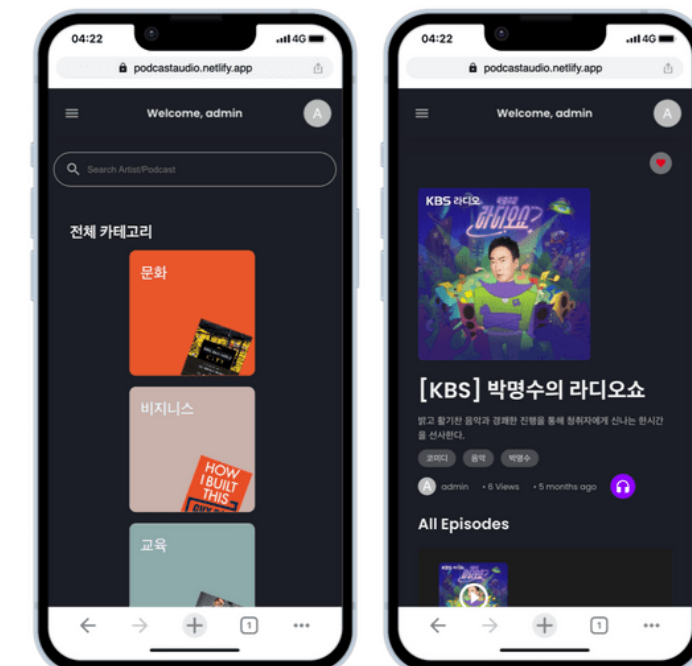
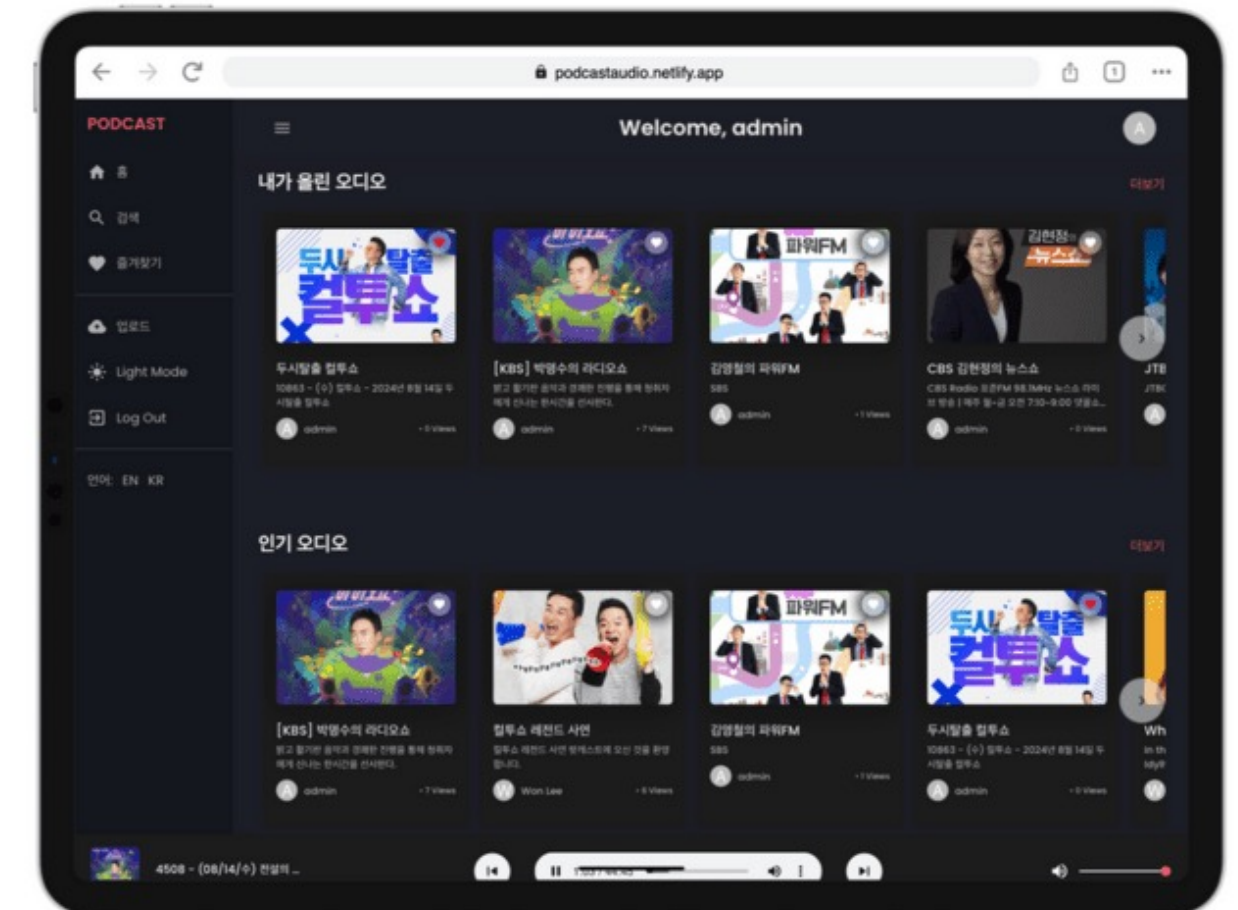
- React (Component-based architecture for efficient UI implementation)
- Redux Toolkit (State management)
- Styled-Components (Styling)
- Material-UI (UI components)
- i18next for Korean/English multi-language support

Backend:

- Node.js (RESTful API, main logic)
- Express.js
- JWT and Kakao OAuth for user authentication

DB:

- MongoDB (Database)
- AWS S3 (File storage)



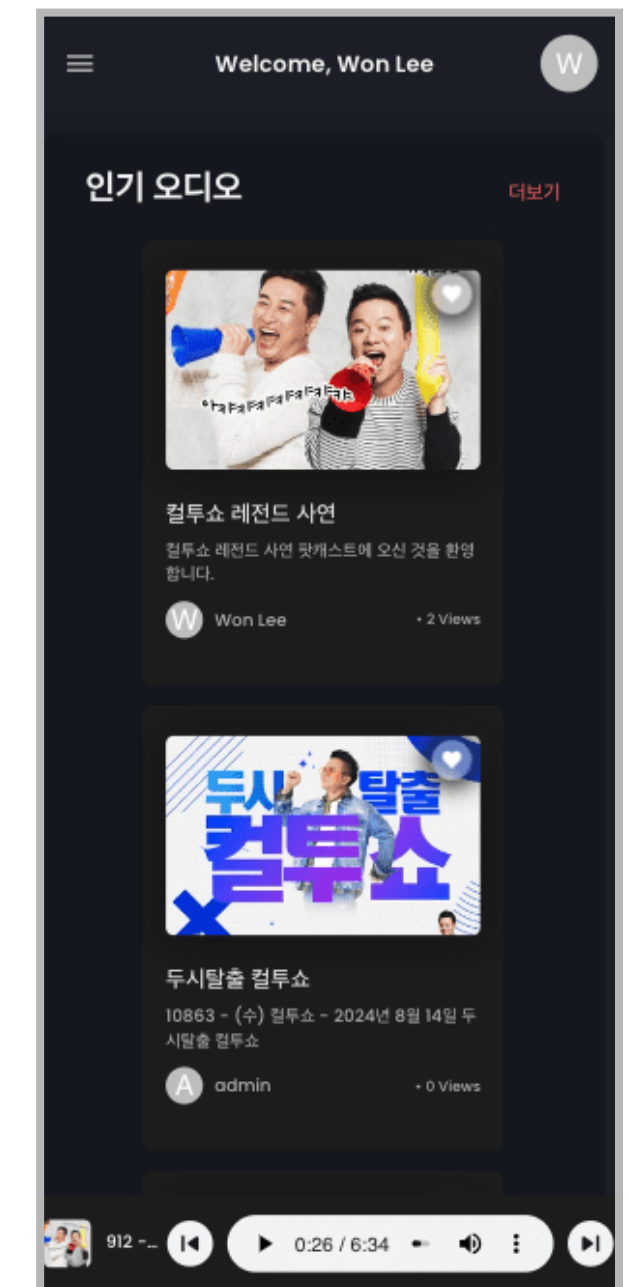
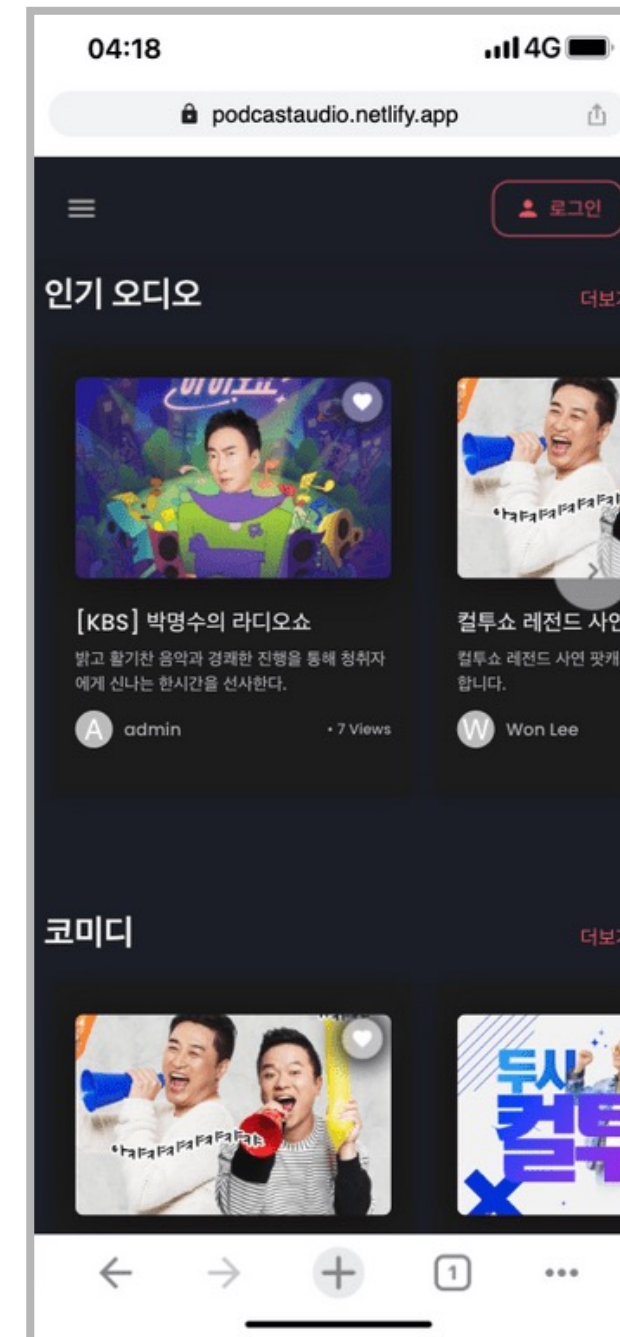
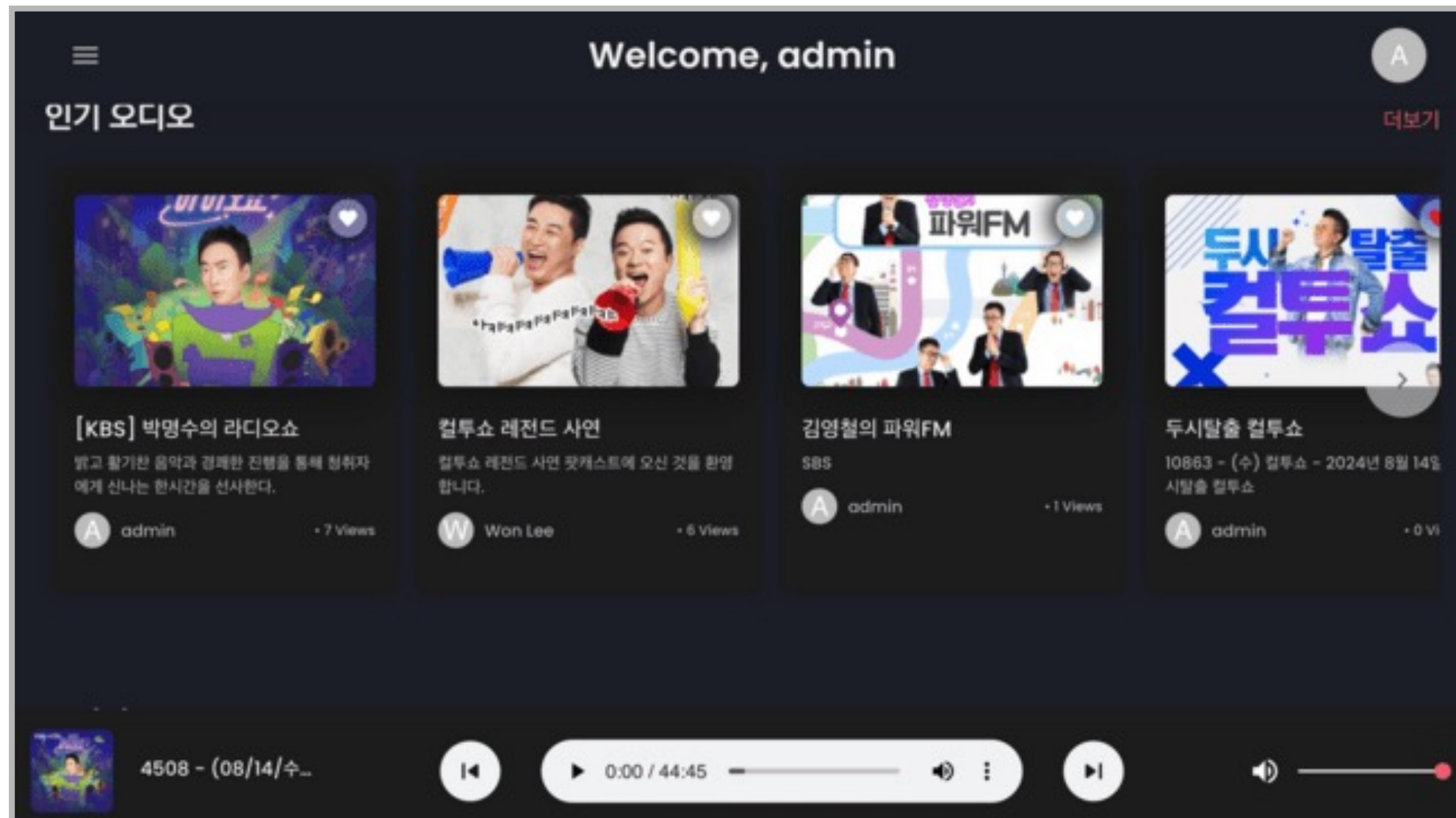
02 PodStream - Podcast Radio Platform

Core Features:

- **User Authentication and Security System:**
 - JWT token-based security authentication system
 - Social login integration with Kakao OAuth
 - User session management and permission-based access control
- **Multimedia Streaming System:**
 - Custom audio player with real-time playback progress tracking and synchronization
 - Advanced player features such as playback speed control and volume control
 - User-specific playback history storage and resume listening functionality
 - Background playback support in mobile environments
 - Previous/next episode navigation
- **Cloud-based Media Management System:**
 - Large media file upload/download system using AWS S3
- **Content Management:**
 - Podcast upload and management
 - Episode addition and modification
 - Category-based content classification
 - Favorites functionality
- **UI/UX and Multi-language Support:**
 - Efficient state management using Redux Toolkit
 - Korean/English multi-language processing system using i18next
 - Responsive design supporting both mobile and desktop environments
- **Performance Optimization:**
 - Initial loading time optimization through lazy loading
 - Prevention of unnecessary re-rendering using React.memo
 - Improved rendering performance through component memoization
 - Efficient state management using Redux Toolkit

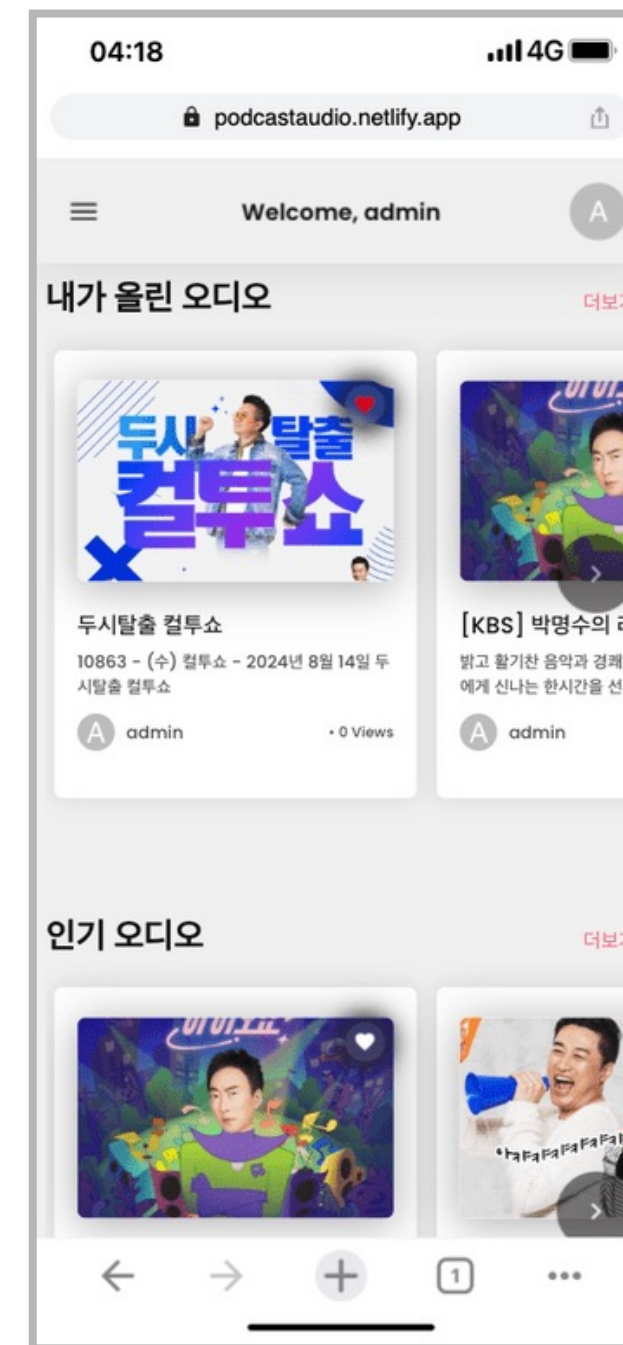
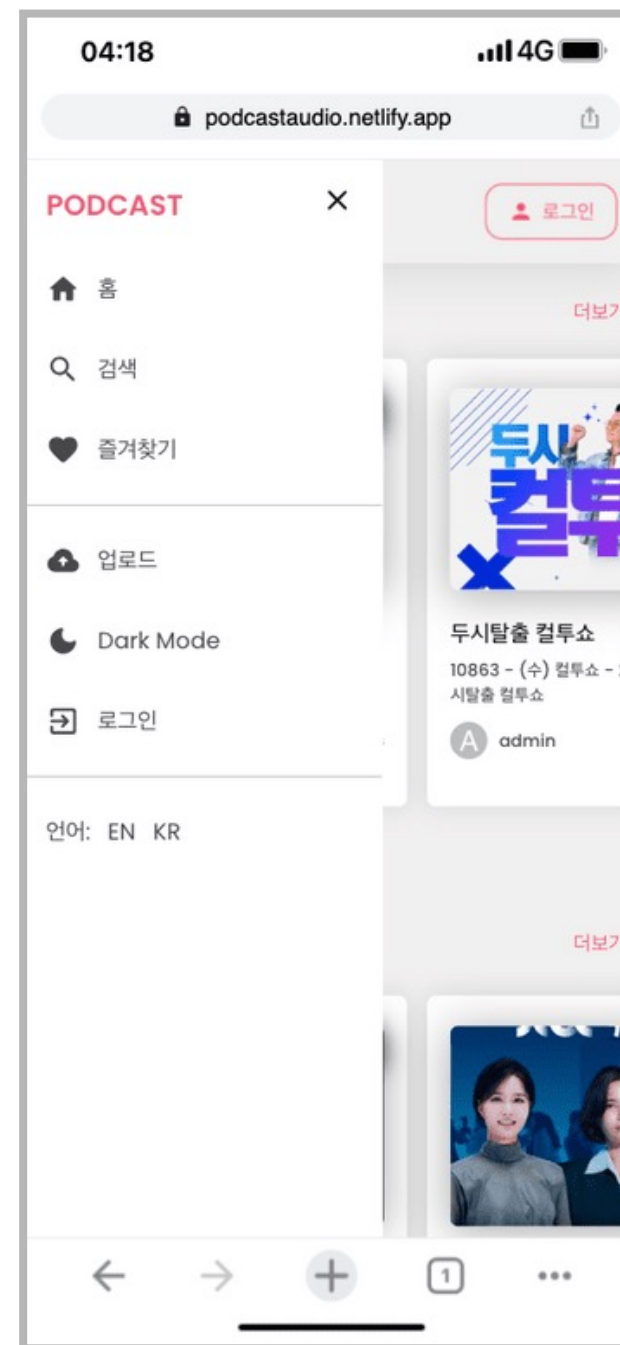
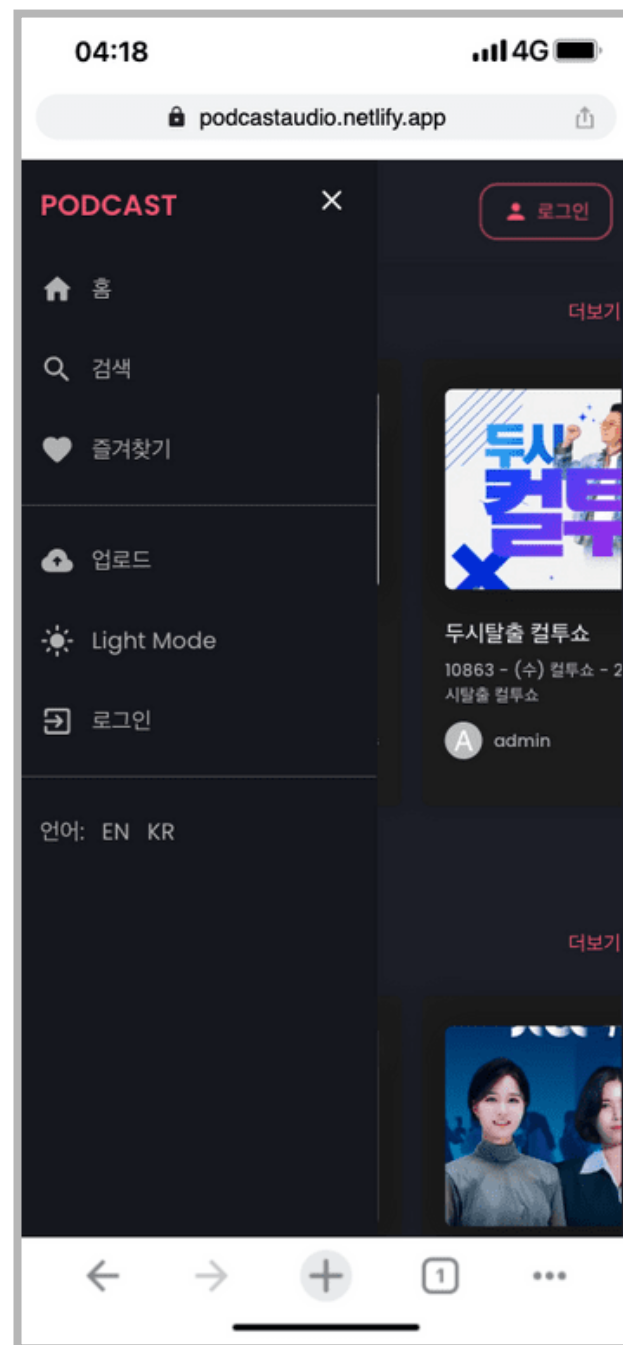
02 PodStream - Podcast Radio Platform

Homepage and Popular Audio, Category-based Content Classification, Multimedia Streaming System:



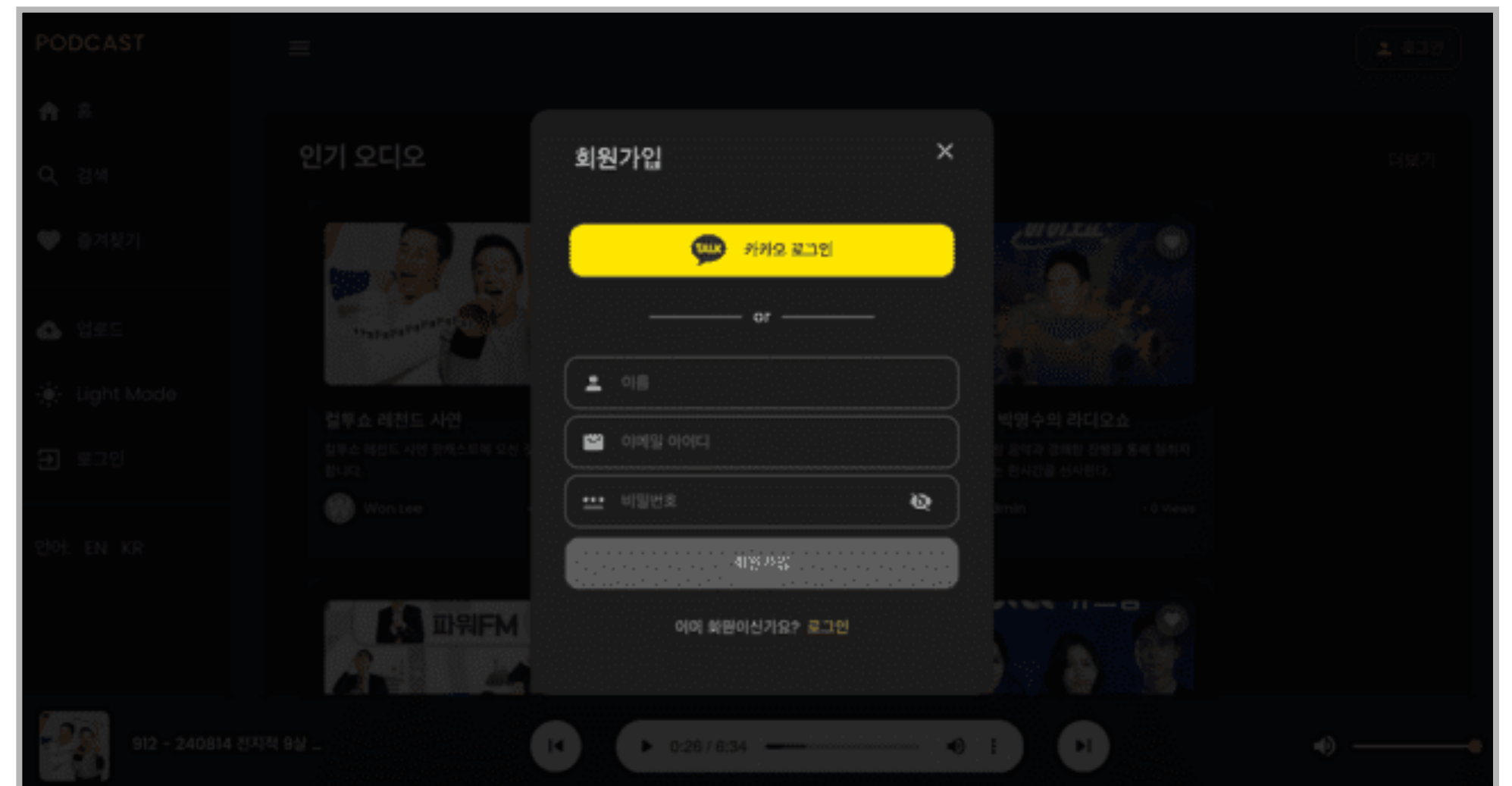
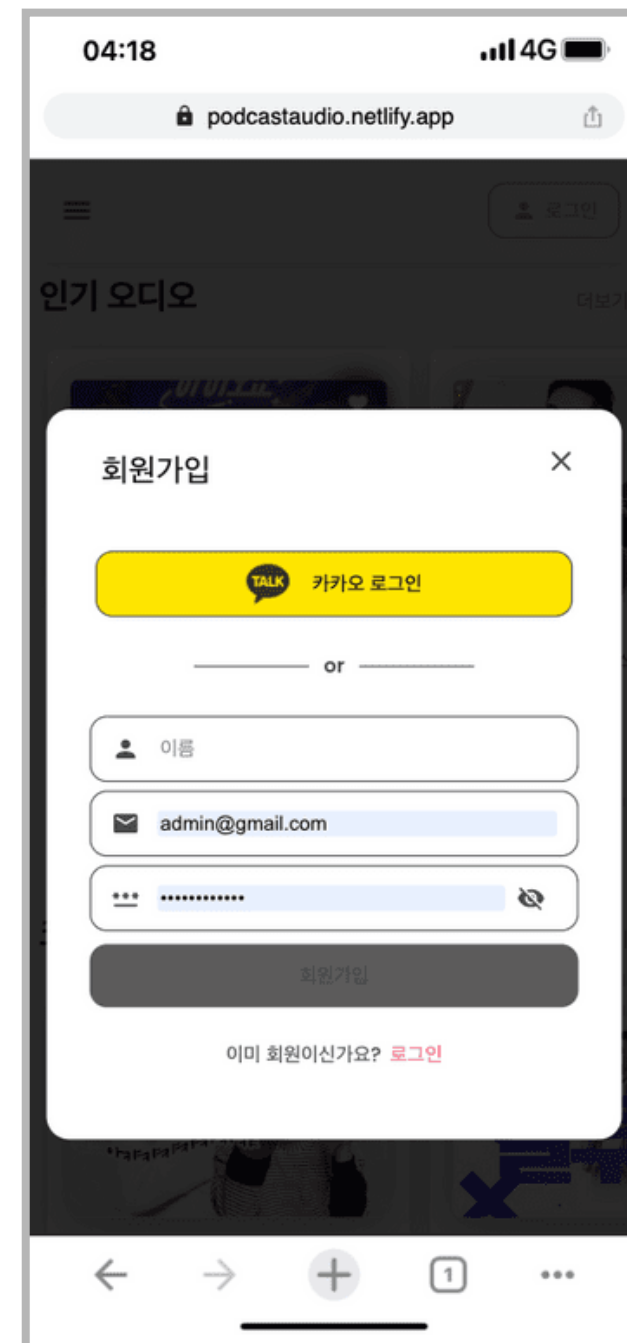
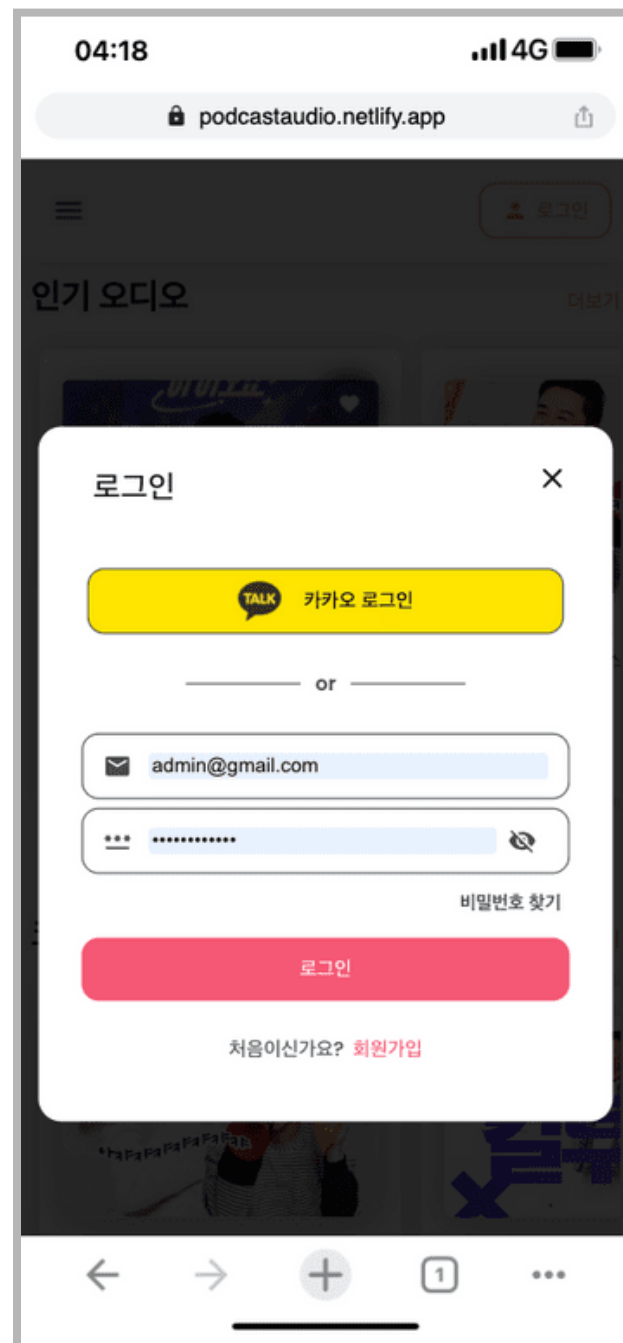
02 PodStream - Podcast Radio Platform

Light & Dark Mode, Korean/English Multi-language:



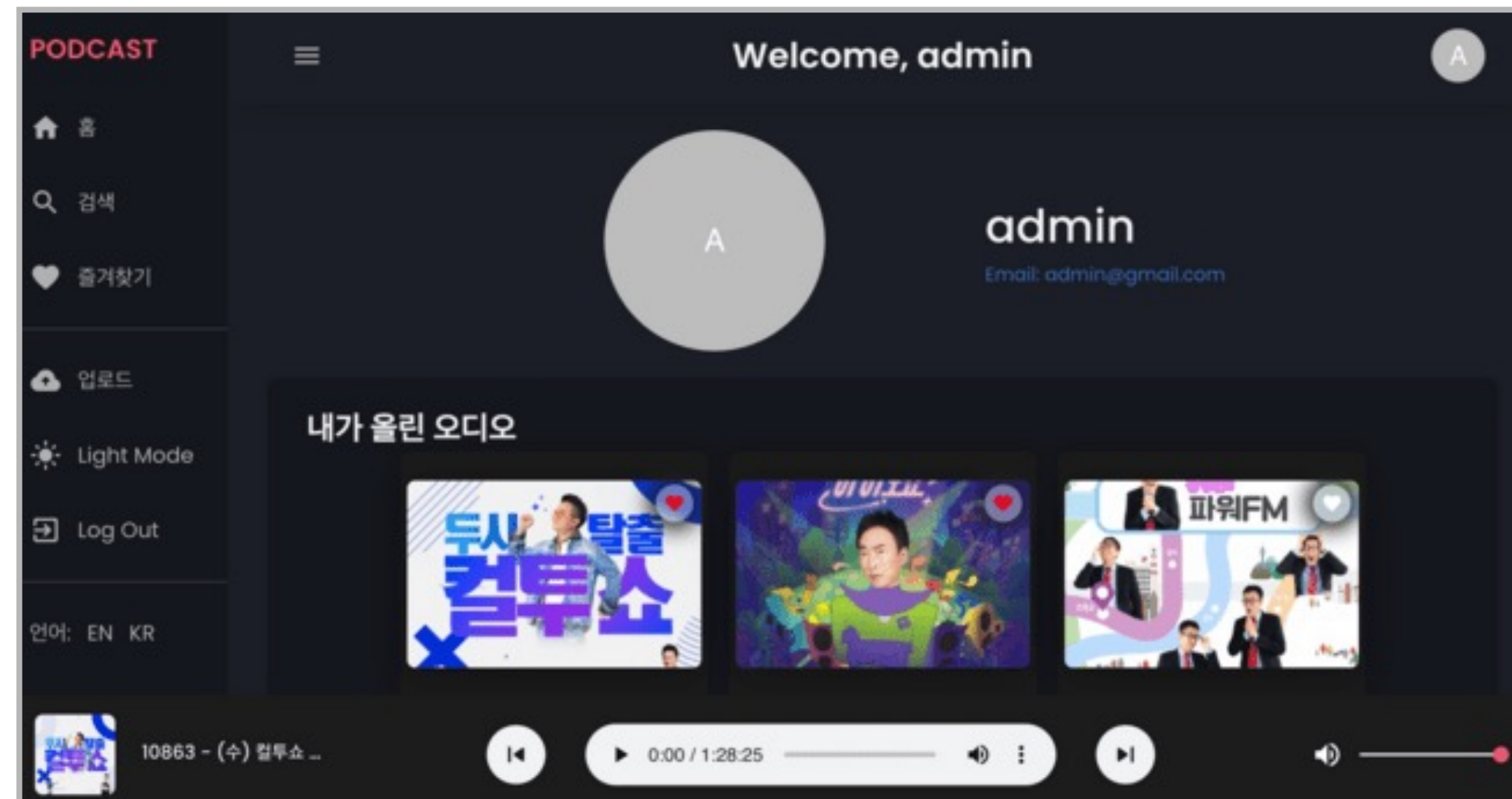
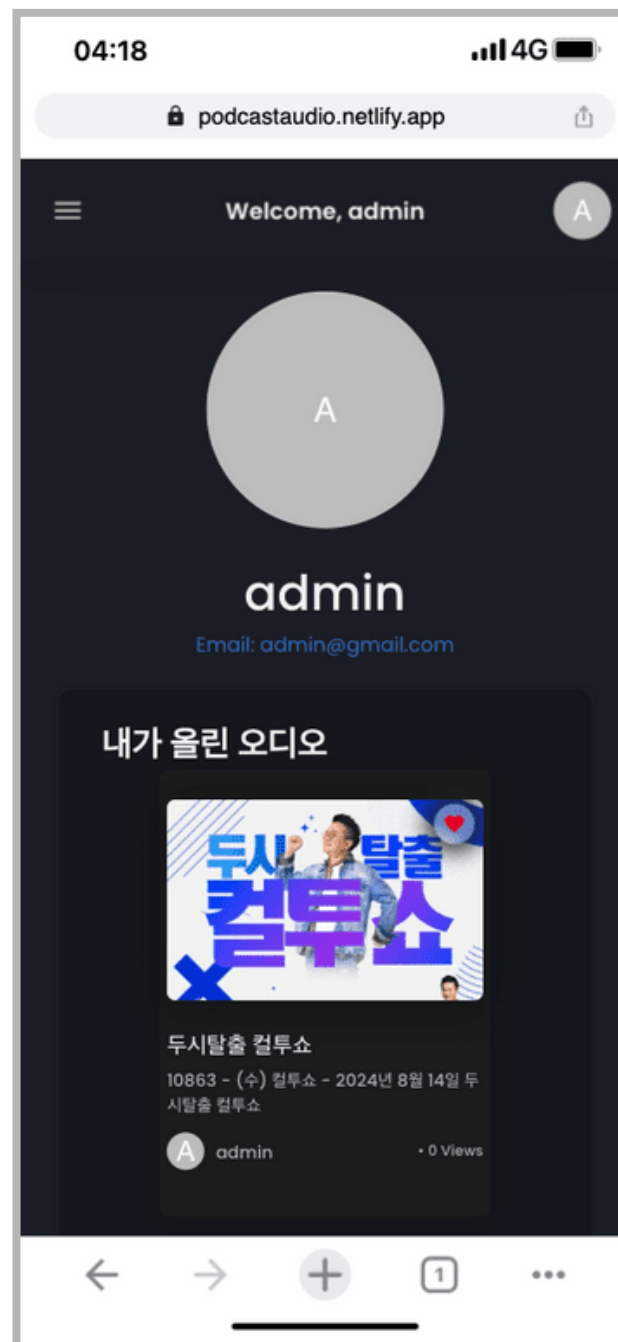
02 PodStream - Podcast Radio Platform

Registration & Login:



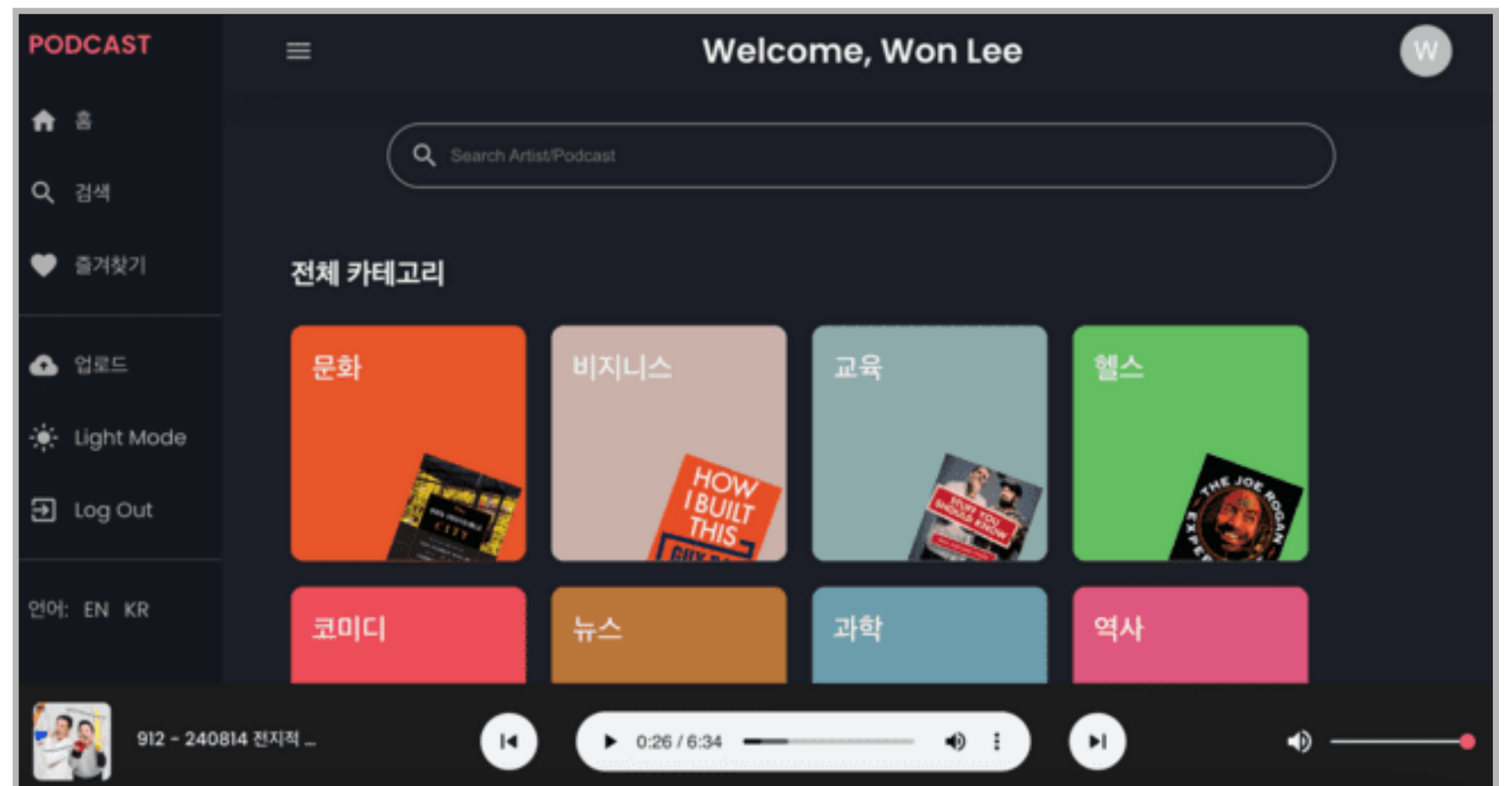
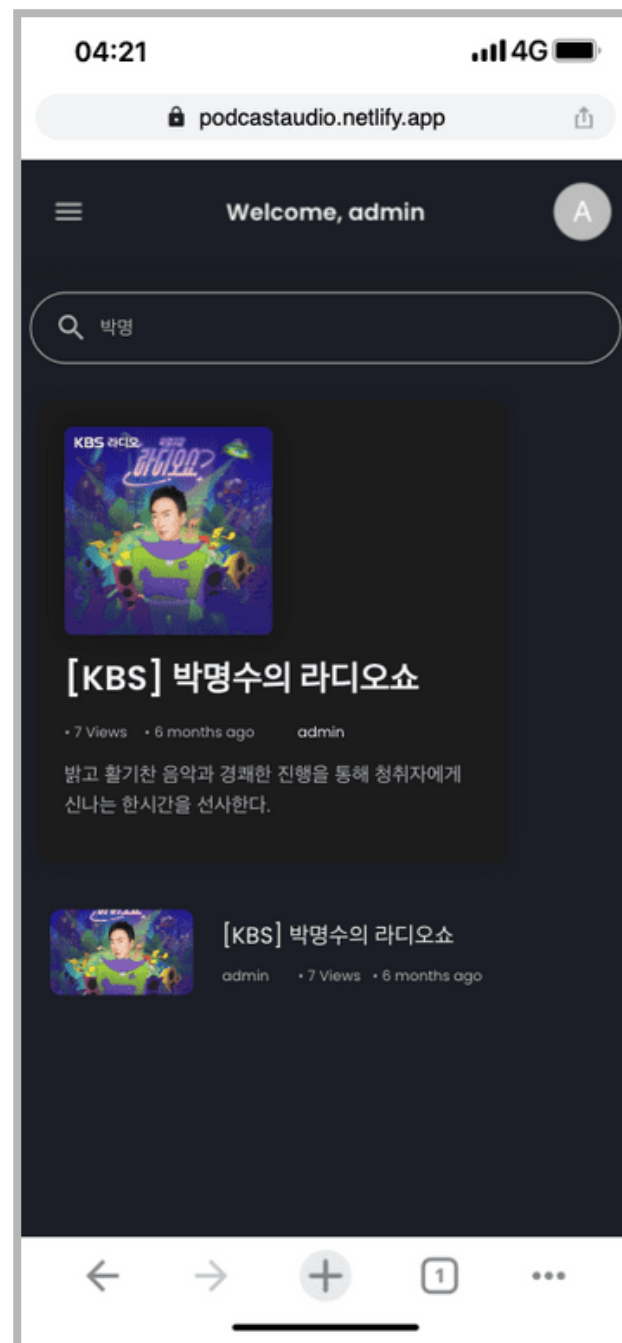
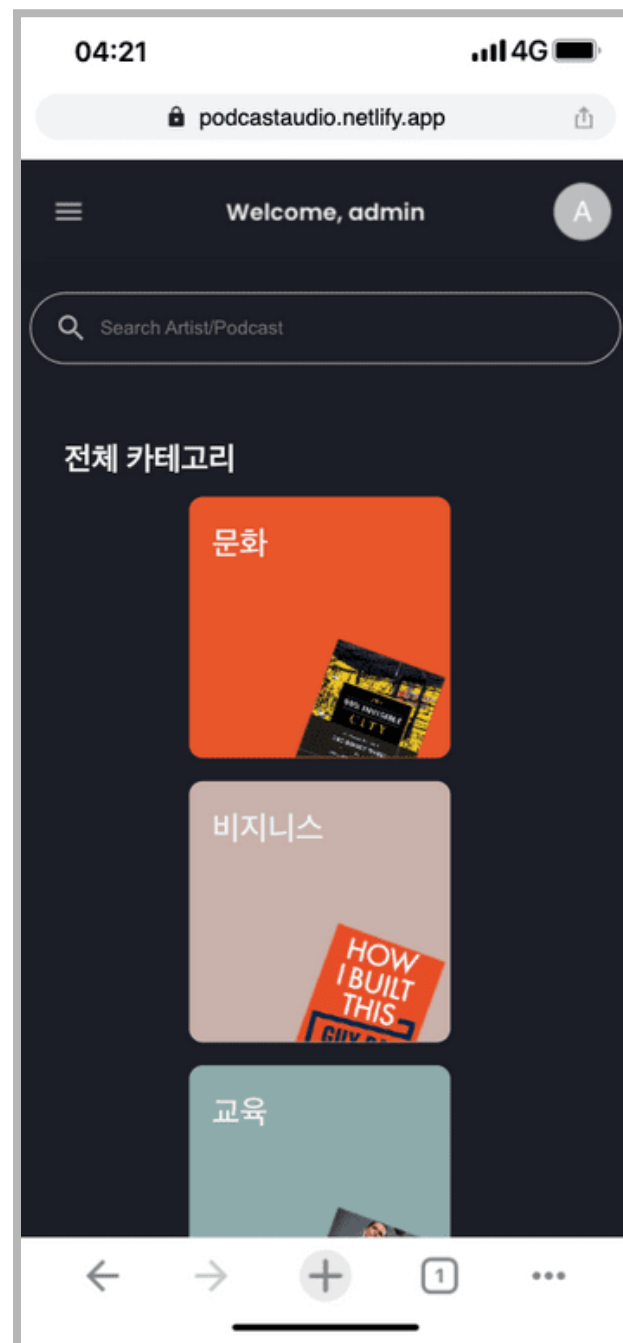
02 PodStream - Podcast Radio Platform

Profile Page, Podcast Upload and Management:



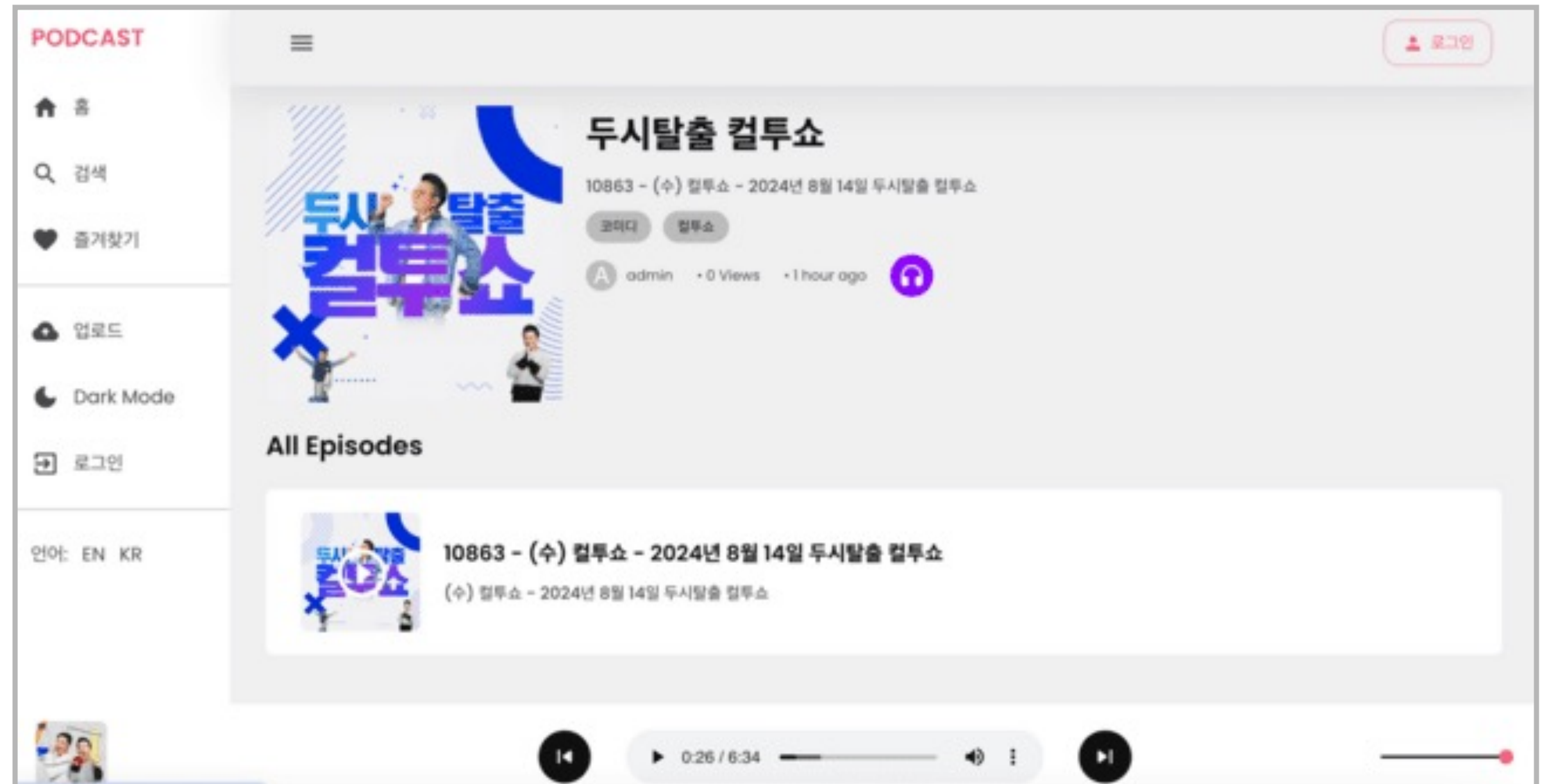
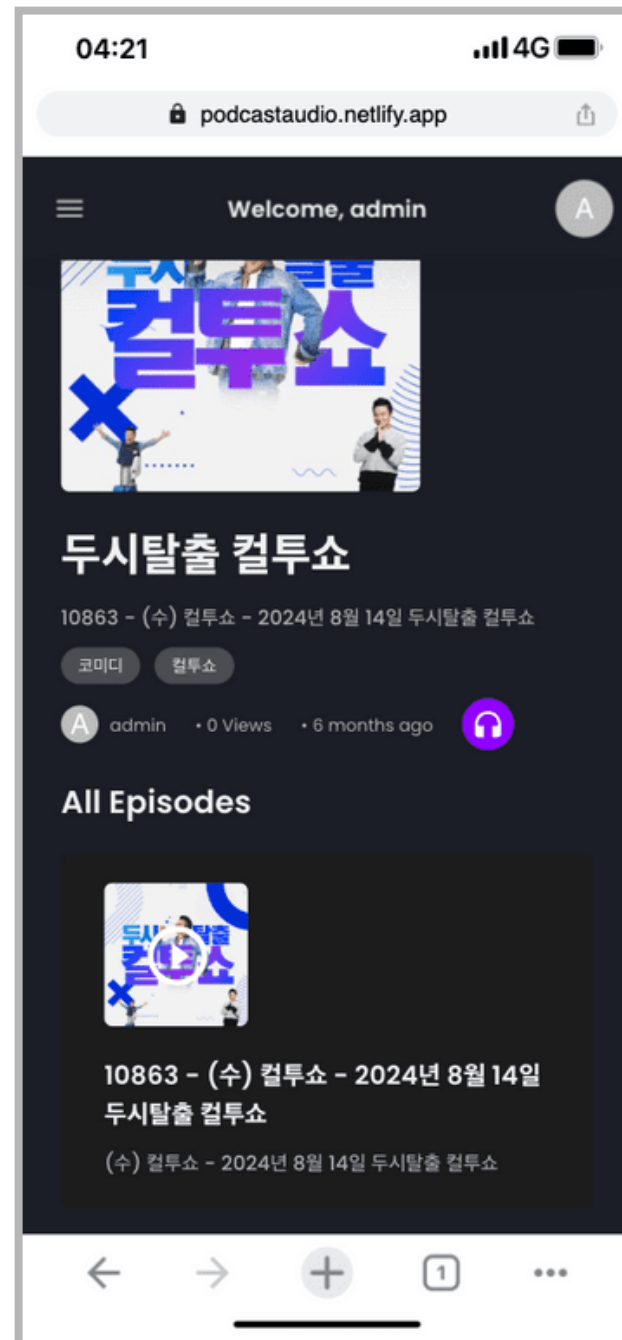
02 PodStream - Podcast Radio Platform

Search & Category Page:



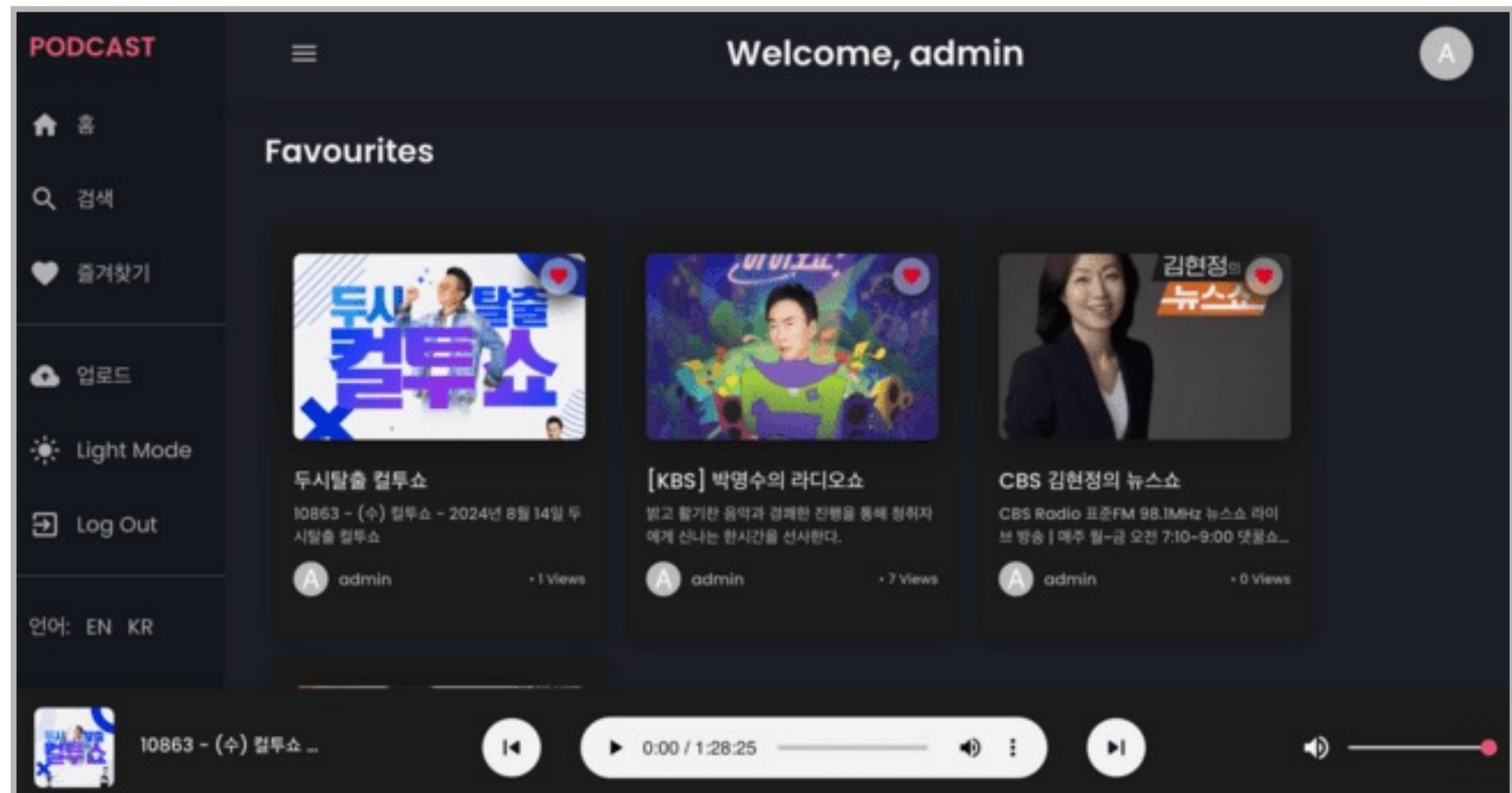
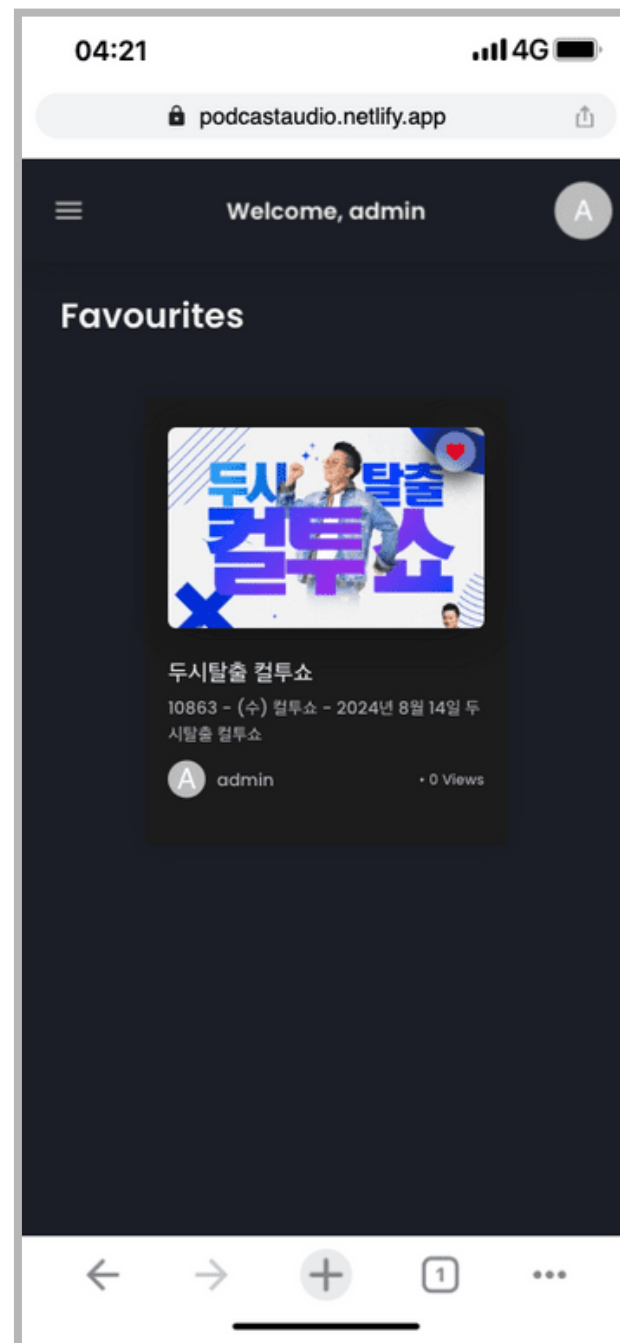
02 PodStream - Podcast Radio Platform

Radio Details and Episode Page:



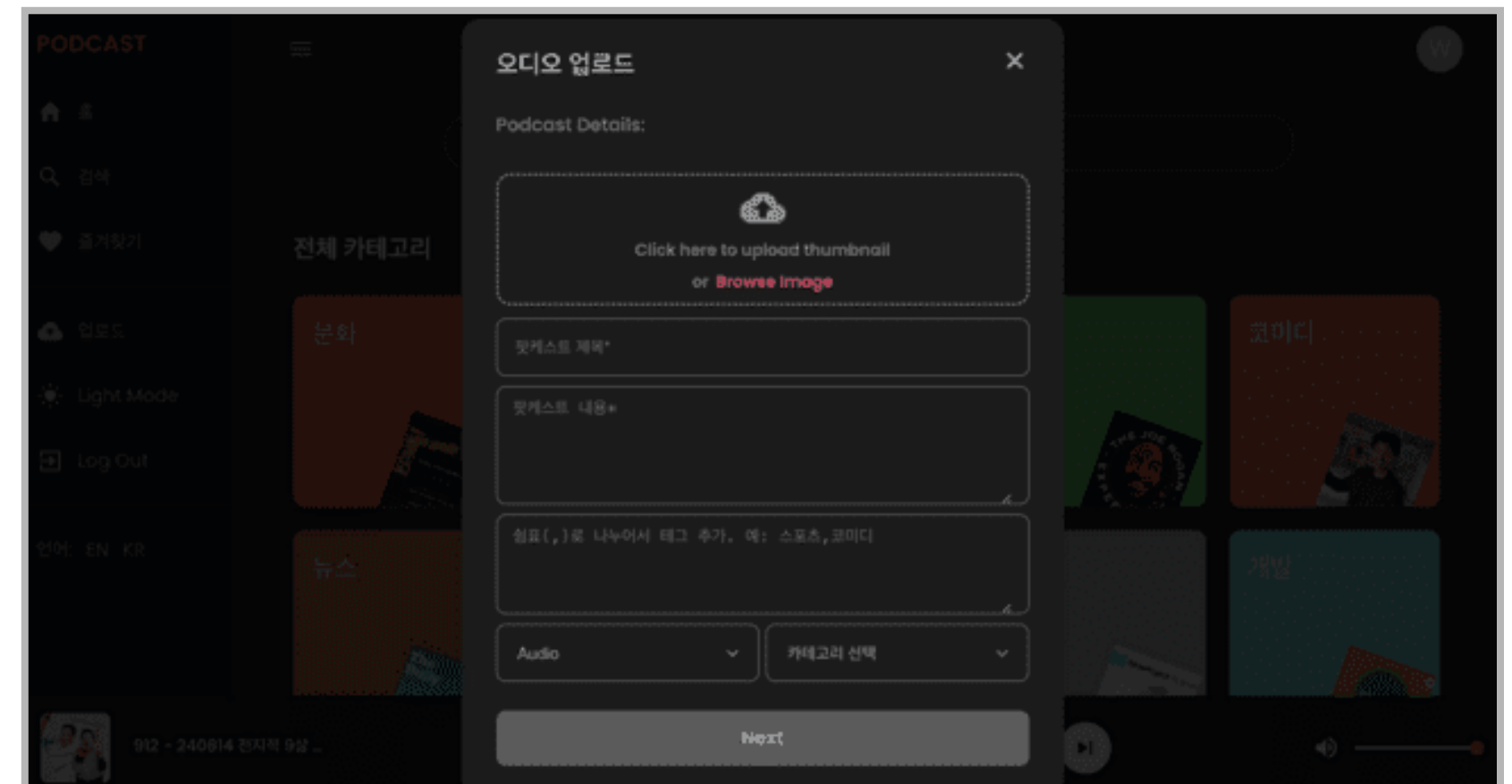
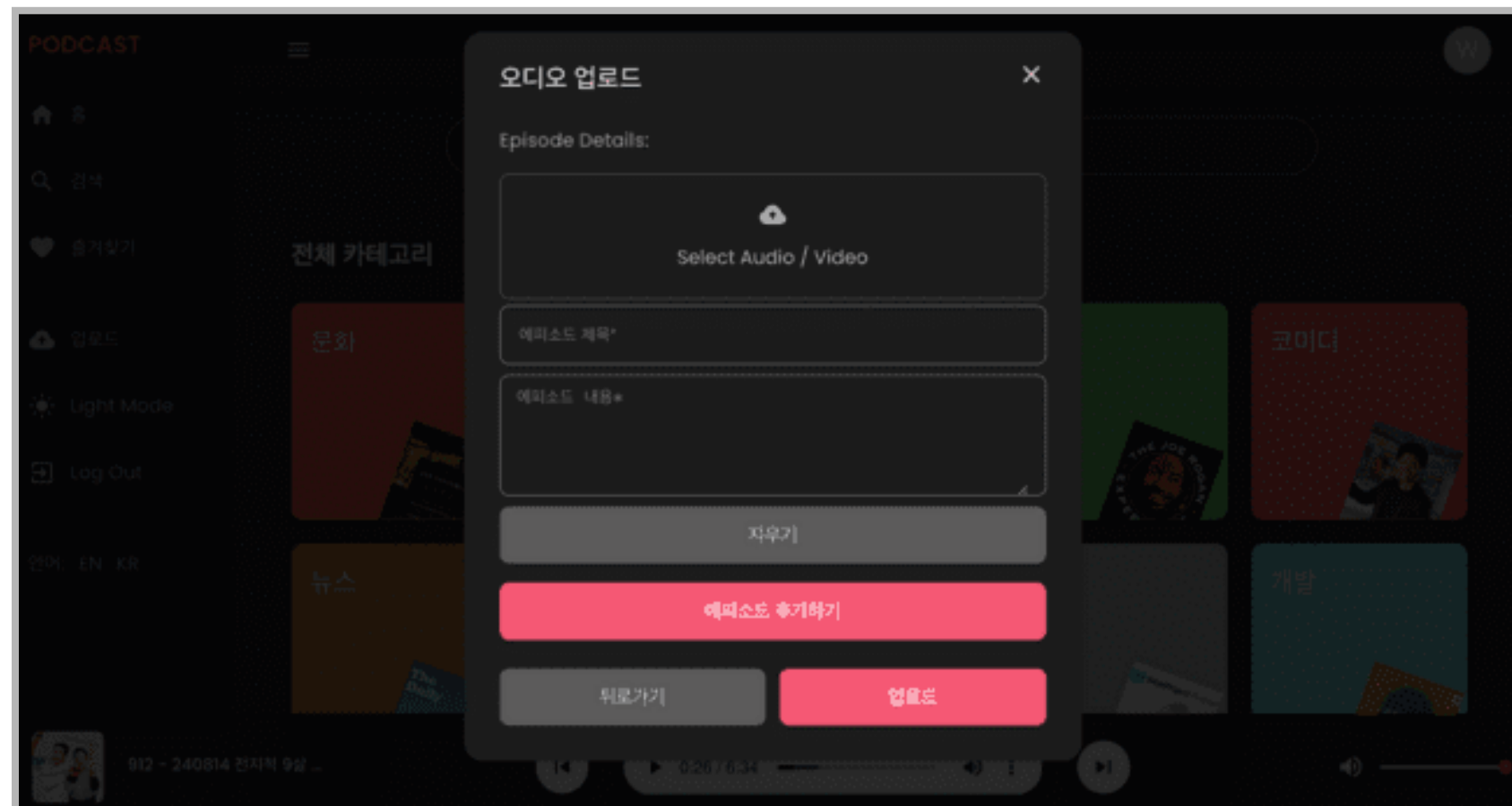
02 PodStream - Podcast Radio Platform

Favorites:



02 PodStream - Podcast Radio Platform

Audio Upload:

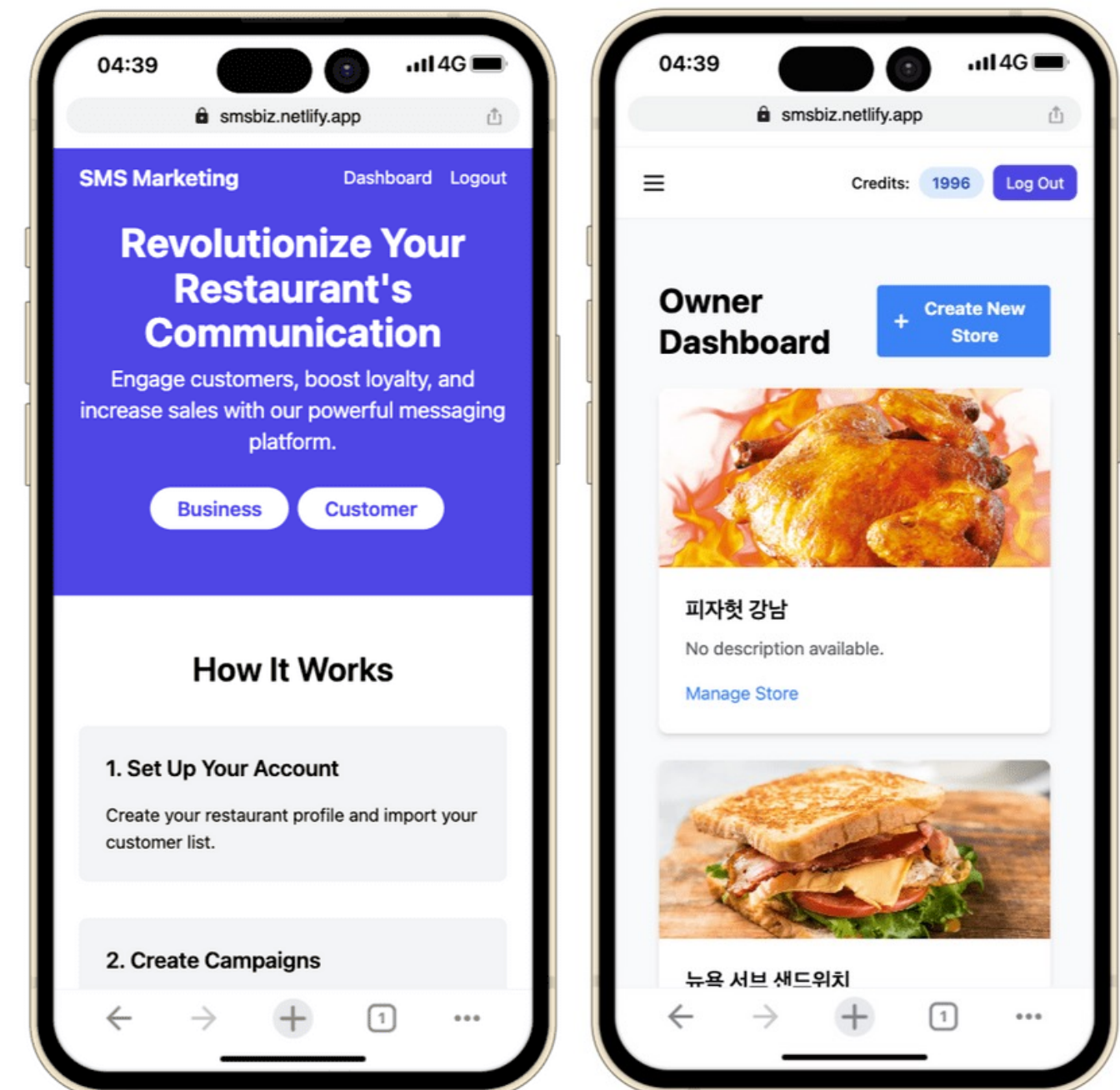


03 SMS Subscription & Marketing Service for Offline Stores

Intro

I developed an integrated marketing platform to address the challenges of digital transformation in the restaurant industry. By talking with restaurant operators in the field, I could directly identify the difficulties they face in customer communication, marketing, and loyalty program operations. I aimed to create a solution that benefits both restaurant operators and customers, particularly to overcome inefficient operations and customer management limitations due to the lack of digital tools.

Development Period:	• Planning and design: 2024.07.23 ~ 2024.08.13 • Development: 2024.08.14 ~ 2024.12.11 • Testing and deployment: 2024.12.12 ~ 2024.12.26
My Role:	• PM, Frontend, Backend • Service planning and direction setting • Main logic implementation, API design, DB architecture
Github	https://github.com/whd793/sms-startup
Demo	https://smsbiz.netlify.app/



03

SMS Subscription & Marketing Service for Offline Stores

Development Period:

Frontend:

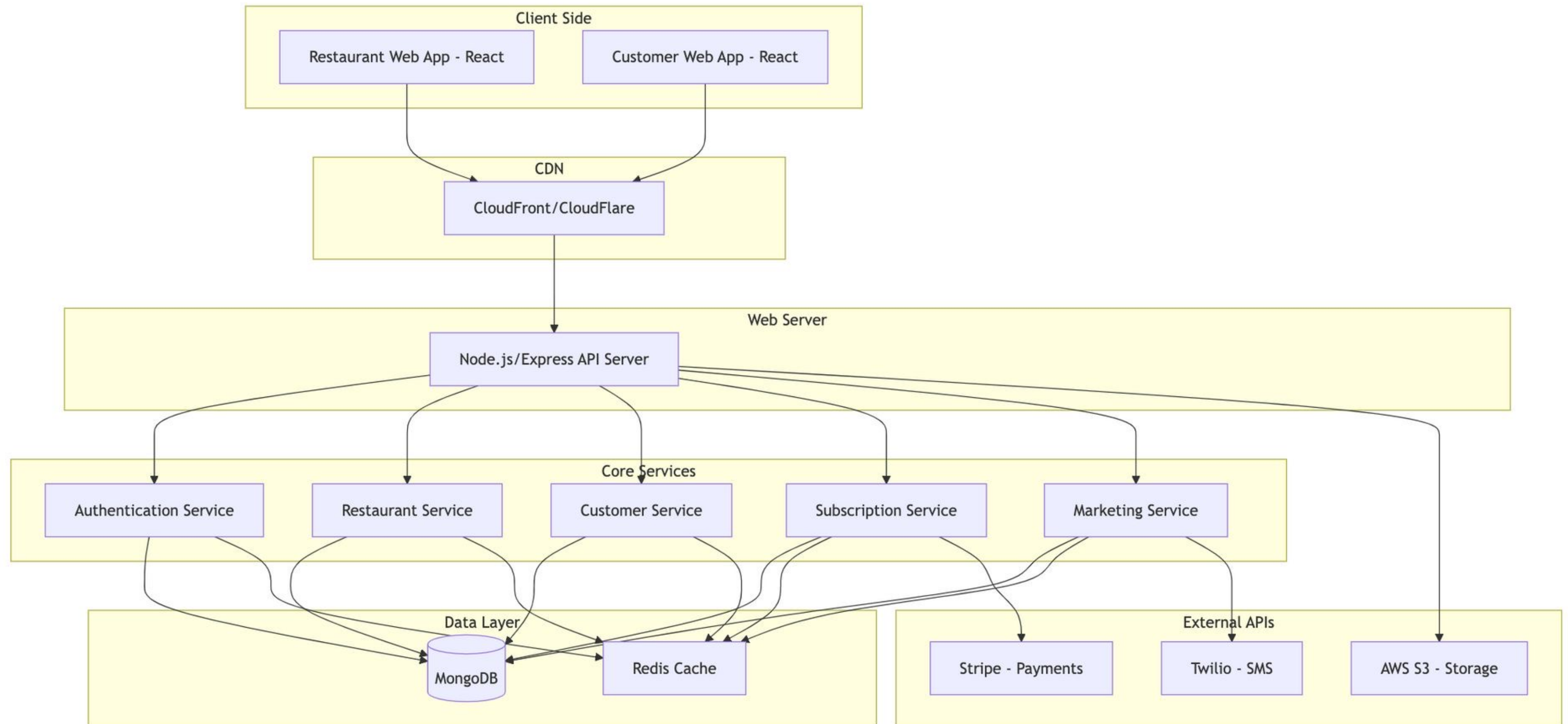
- Core framework: React 18
- State management: React Query, Context API
- Routing: React Router
- API communication: Axios
- Styling: Tailwind CSS, Framer Motion
- Form management: React Hook Form, Yup
- Payment module: Stripe Elements

Backend:

- Runtime: Node.js 18
- Framework: Express.js
- Database: MongoDB/Mongoose
- Caching: Redis
- Authentication: JWT, bcrypt
- File storage: AWS S3
- SMS service: Twilio API
- Payment system: Stripe API

03 SMS Subscription & Marketing Service for Offline Stores

System Architecture



03

SMS Subscription & Marketing Service for Offline Stores

Core Features:

Business Owner Features:

Account Management:

- Registration and login
- profile management
- password change
- account deletion

Text Message Management:

- Marketing message creation and scheduled sending
- message template storage
- image attachment
- recurring sending
- Twilio API integration

Stamp System:

- Stamp accrual rules configuration,
- daily limits and cooldown system,
- reward exchange system,
- customer stamp status tracking,
- NFC tag-based stamp collection

Subscriber Management:

- Subscriber list viewing,
- subscriber statistics,
- subscription cancellation management

Restaurant Management:

- Restaurant information
- registration/editing/deletion,
- profile image upload,
- automatic QR code generation,

03

SMS Subscription & Marketing Service for Offline Stores

Core Features:

Business Owner Features:

Coupon System:

- Coupon creation and management,
- QR code-based coupon authentication,
- coupon expiration management,
- usage tracking,
- targeting options

Search and Filtering:

- Advanced search functionality,
- multi-filtering options,
- sorting features,
- real-time search results

Image Processing System

- AWS S3-based image upload

03

SMS Subscription & Marketing Service for Offline Stores

Core Features:

Customer Features:

- **Account Management:**
 - Phone number-based registration,
 - OTP authentication,
 - profile settings,
 - account settings management
- **Restaurant Subscription:**
 - Subscription via QR code scan,
 - subscribed restaurant list viewing,
 - subscription cancellation
- **Coupon Management:**
 - Viewing issued coupons,
 - coupon usage and management,
 - expiring coupon alerts
- **Stamp Collection:**
 - Stamp collection via NFC tag/QR code,
 - viewing collection status,
 - reward redemption
- **Playlist Features:**
 - Favorites,
 - saving preferred restaurants,
 - custom playlist creation,
 - playlist management

03 SMS Subscription & Marketing Service for Offline Stores

Core Features:

I developed a subscription system using the Stripe API. Users can select and manage various subscription plans, with features including credit card payments, plan upgrades/downgrades, and credit purchases.

Subscription System:

- Free/paid subscription plan management system
- Automatic proration system for plan changes
- Subscription renewal, upgrade, downgrade logic implementation
- Free plan and 3 paid subscription plans (Basic, Pro, Premium)
- Immediate application and balance settlement for plan upgrades
- Automatic change after current subscription period for downgrades
- Subscription cancellation and reactivation functionality

Credit System:

- Credit system (subscription/purchase) design and implementation
- Automatic credit renewal and carryover system (Webhook)
- Basic credits provided by subscription plan

03

SMS Subscription & Marketing Service for Offline Stores

Core Features:

Payment System Integration:

- PCI-compliant payment process using Stripe Elements and secure card payment system
- Multiple payment method management
- Automatic payment and failure recovery system
- Default payment method setting
- Payment history viewing and receipt/invoice download
- Automatic payment failure recovery system
- Automated refund processing
- Payment data synchronization system

03 SMS Subscription & Marketing Service for Offline Stores

Technical Problem Solving:

Frontend:

- For user interface performance optimization, I utilized component memoization for rendering optimization, actively applying memoization techniques such as `React.memo`, `useCallback`, and `useMemo`. I also developed custom hooks to enhance business logic reusability.
- For state management, I used Context API to manage global state, with particular effort to maintain consistency of important data such as authentication state.

03 SMS Subscription & Marketing Service for Offline Stores

Technical Problem Solving:

Backend:

- Implemented toast notifications for loading states, error handling, and success/failure messages to provide immediate feedback to users
- Ensured data consistency through MongoDB transaction processing
- Optimized query performance through indexing
- Implemented automatic recovery mechanisms for payment failures
- For security, implemented token-based authentication with JWT and secure password hashing with bcrypt. Particularly implemented an OTP system in the user authentication process for two-factor authentication
- Applied rate limiting to prevent brute force attacks
- Ensured fast response times even with large-scale data processing through database query optimization, caching strategy establishment, and pagination implementation. Particularly used Redis to cache frequently accessed data and efficiently manage sessions

03

SMS Subscription & Marketing Service for Offline Stores

Technical Problem Solving:

Concurrency Control Problem Solving:

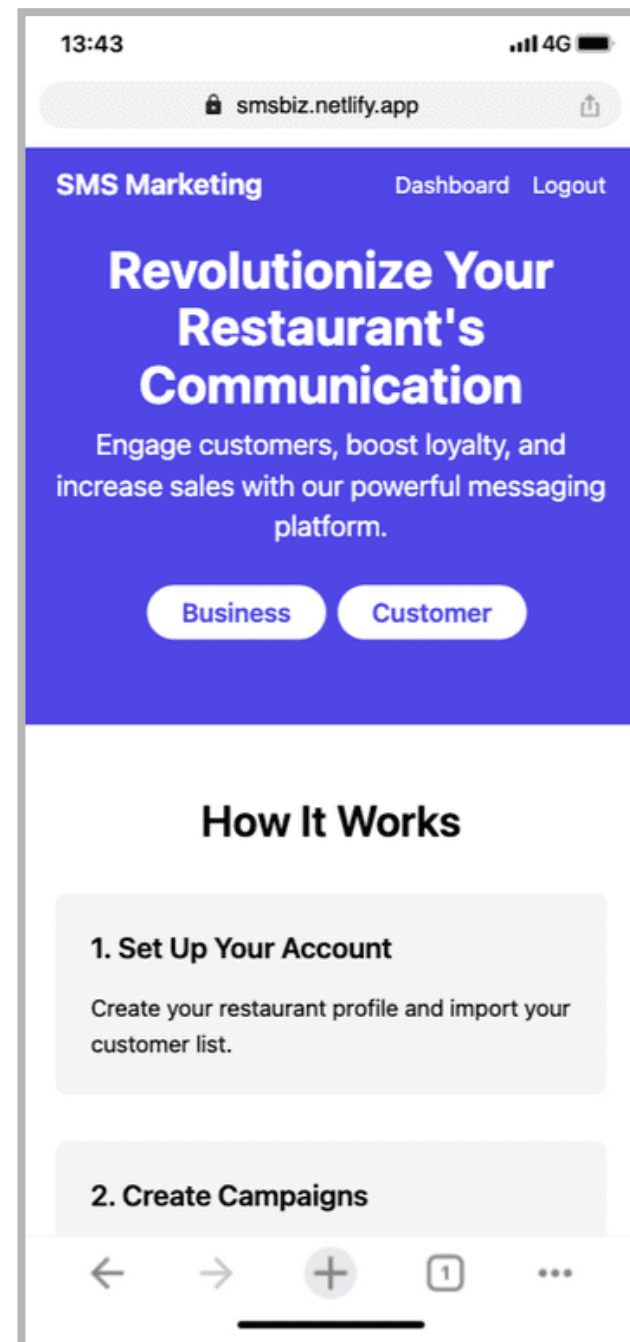
- Implemented lock mechanisms for concurrent access control
- Used unique idempotency keys for payment processing
- Applied database-level transaction locks

Data Consistency: When subscription plans were changed, issues with payment and data update consistency arose. To solve this:

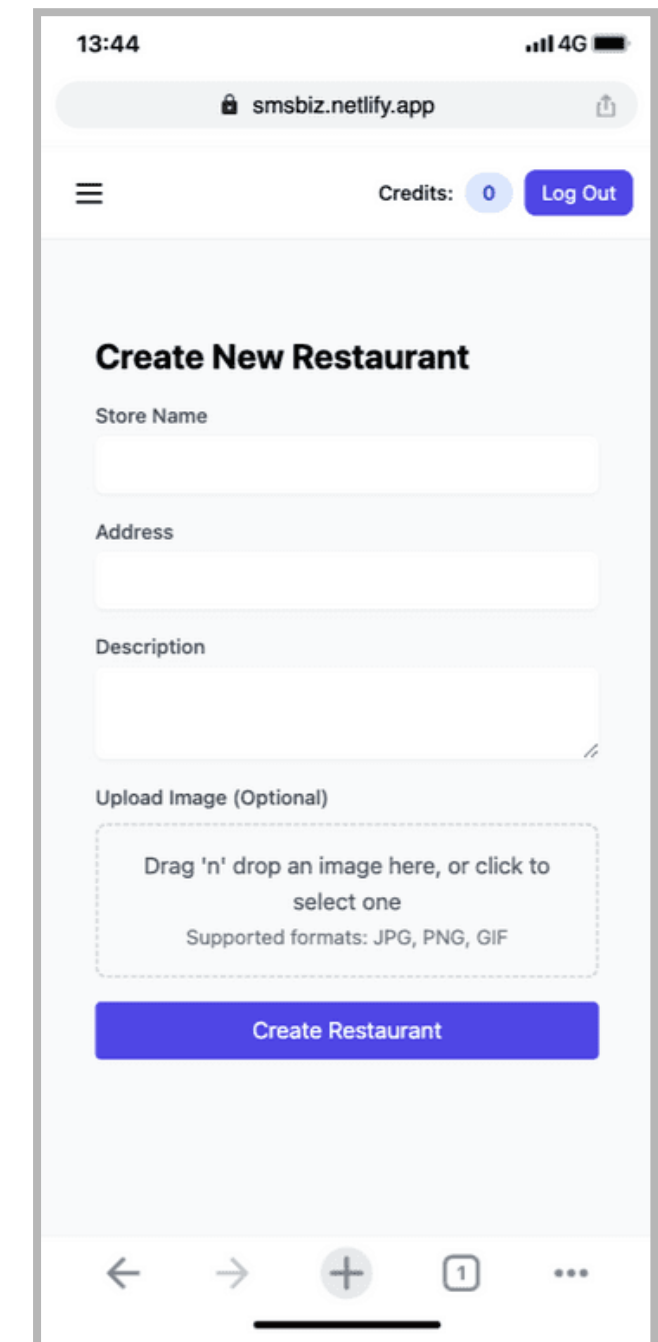
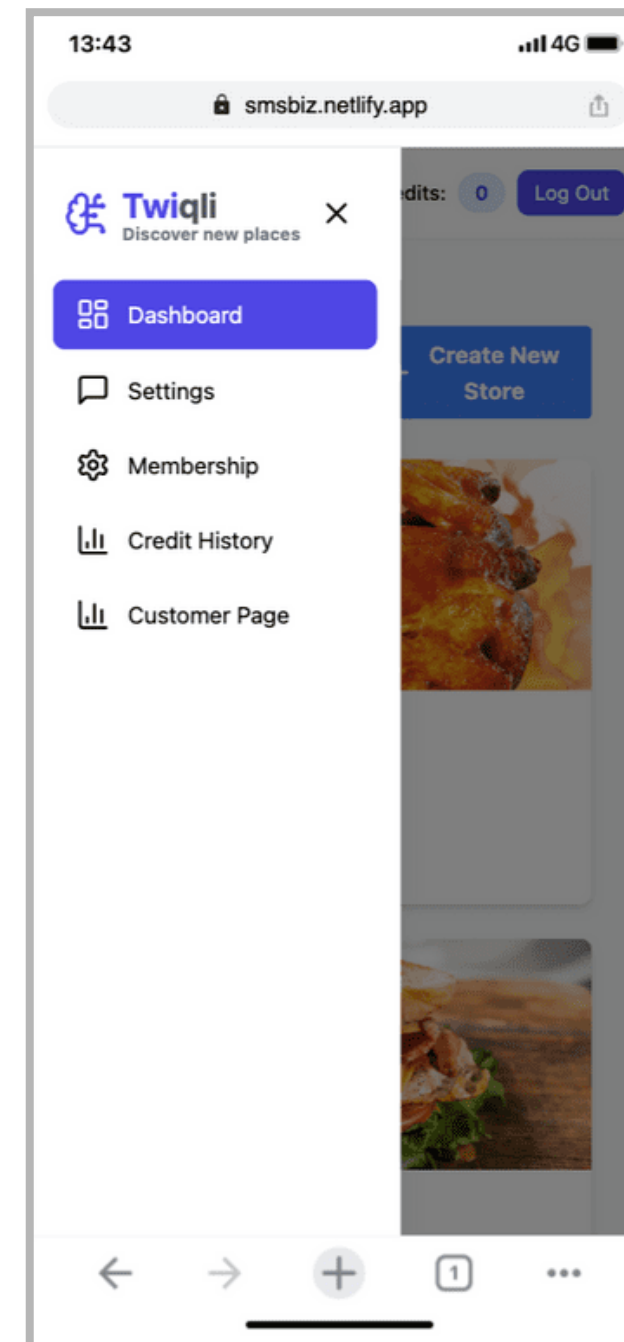
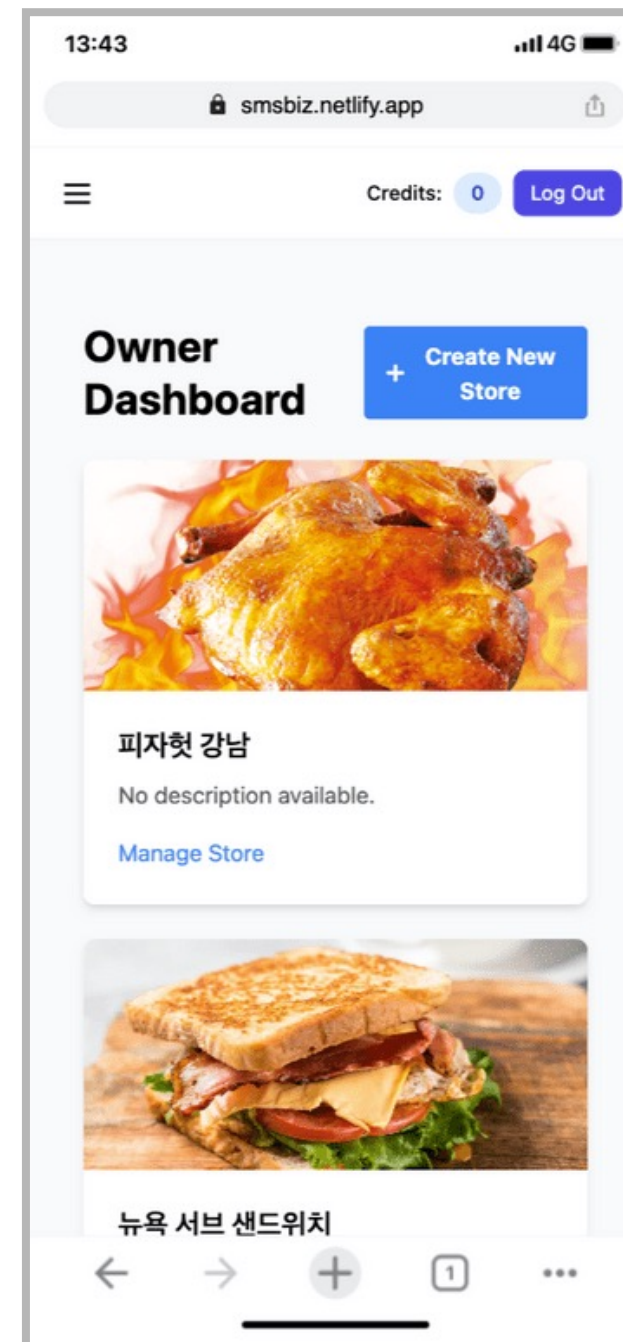
- Introduced MongoDB transactions
- Implemented automatic rollback mechanisms for payment failures

03 SMS Subscription & Marketing Service for Offline Stores

Home Page



Admin dashboard, Resgister/Delete/Edit Business, Profile Image Upload



03 SMS Subscription & Marketing Service for Offline Stores

Admin Account Settings - Profile, Password, Delete

13:44 smsbiz.netlify.app Credits: 0 Log Out

Owner Account Settings

Profile Information

Name
admin

Email
admin@gmail.com

✓ Update Profile

Update Password

Current Password
Enter current password

New Password
Enter new password

Confirm New Password
Confirm new password

✓ Update Password

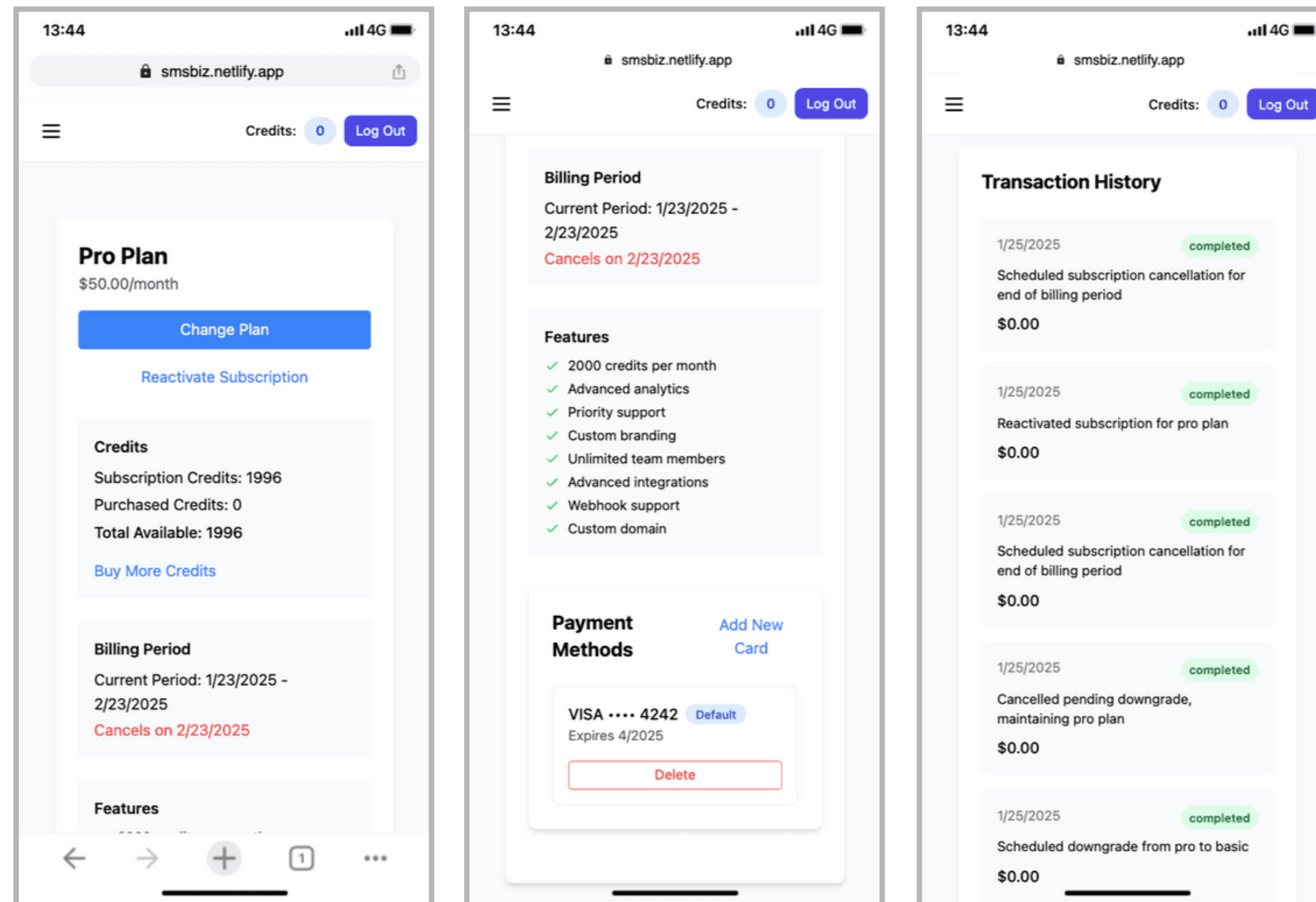
Danger Zone

Once you delete your account, there is no going back. Please be certain. This will also delete all your restaurants and associated data.

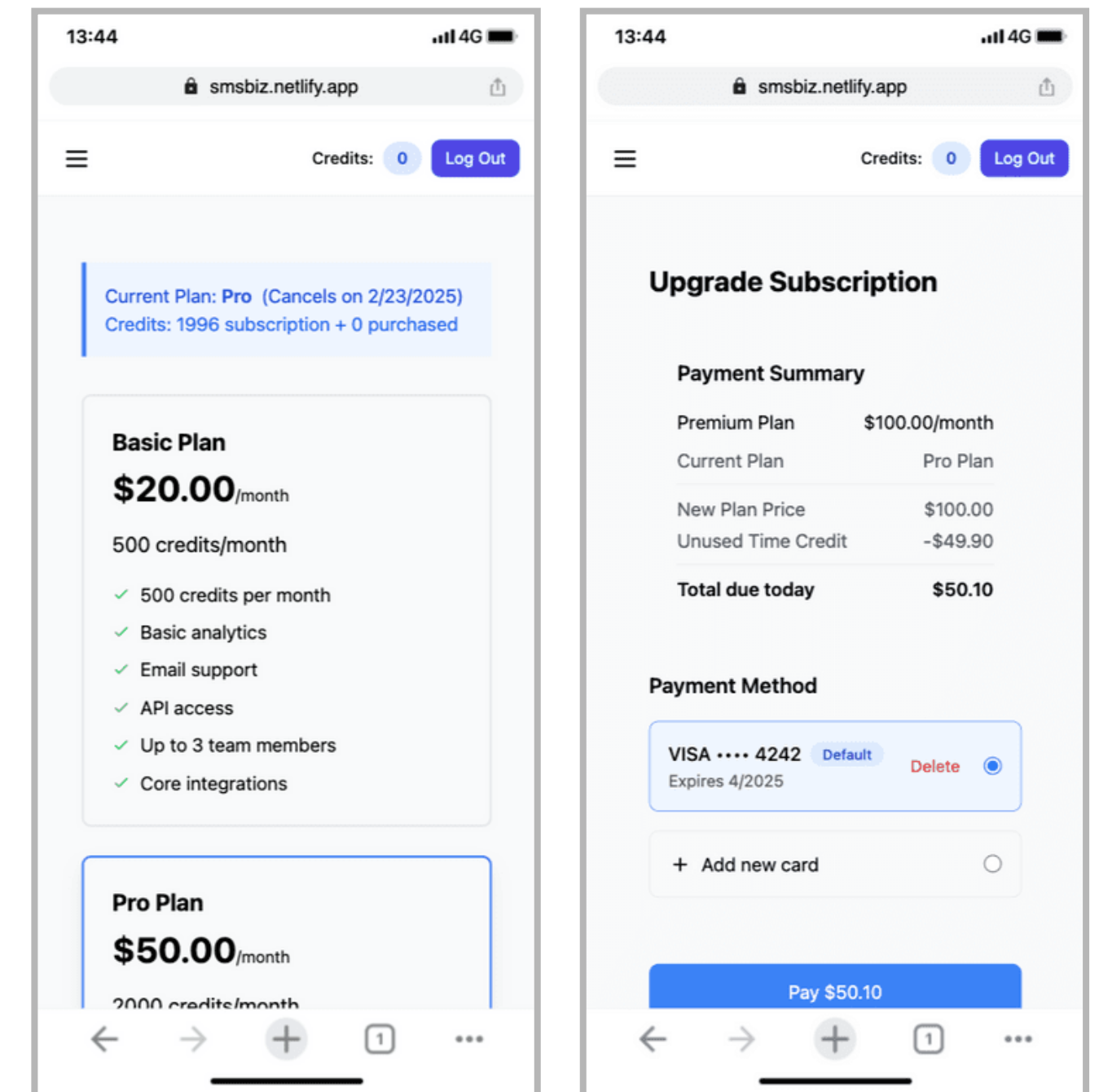
Delete Account

03 SMS Subscription & Marketing Service for Offline Stores

Subscription System, Plan proration, Subscription Transaction History



Purchase subscription plans, payment methods



03

SMS Subscription & Marketing Service for Offline Stores

Credit History

13:44

smsbiz.netlify.app

Credits: 0

Log Out

Credit History

mm/dd/yyyy

mm/dd/yyyy

All Types

Date	Type
1/25/2025	message_sent
1/24/2025	message_sent
1/24/2025	message_sent
1/24/2025	message_sent

13:44

smsbiz.netlify.app

Credits: 0

Log Out

Credit History

mm/dd/yyyy

mm/dd/yyyy

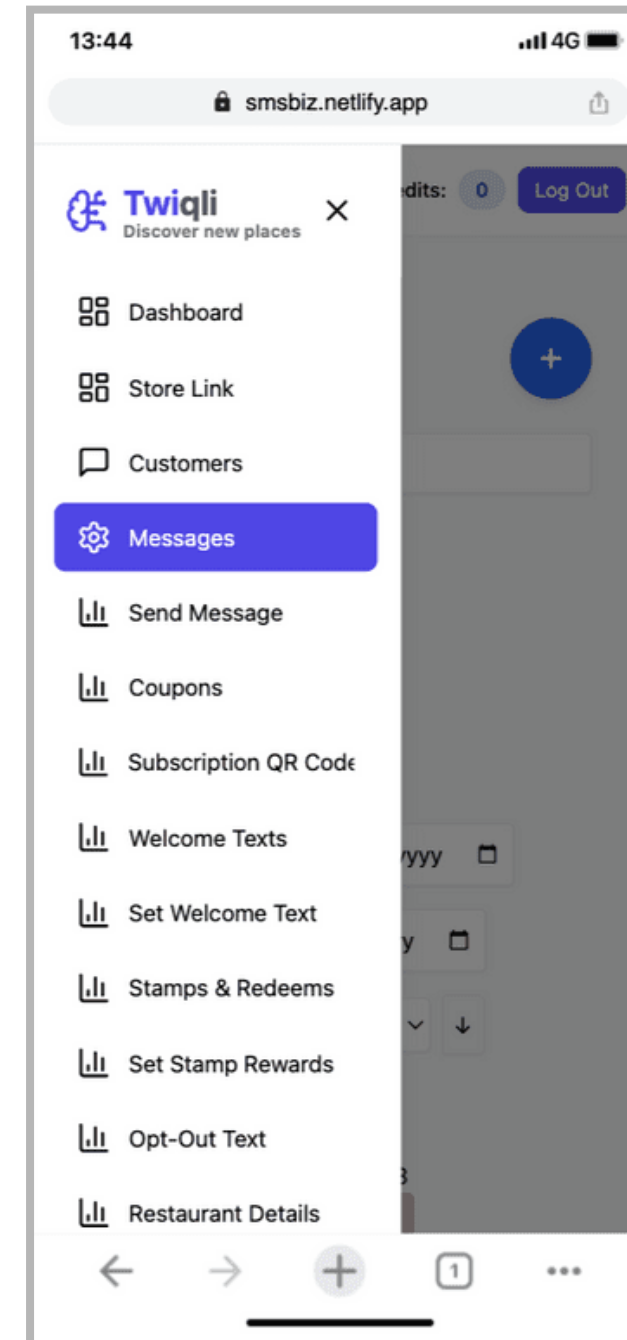
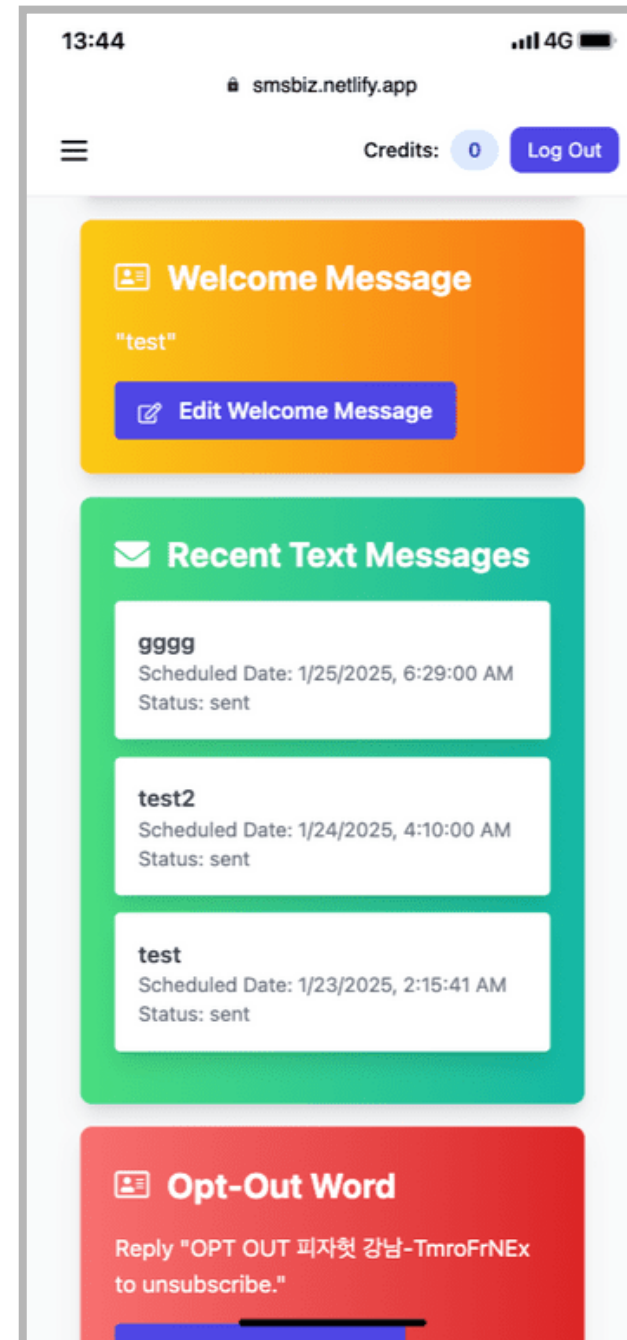
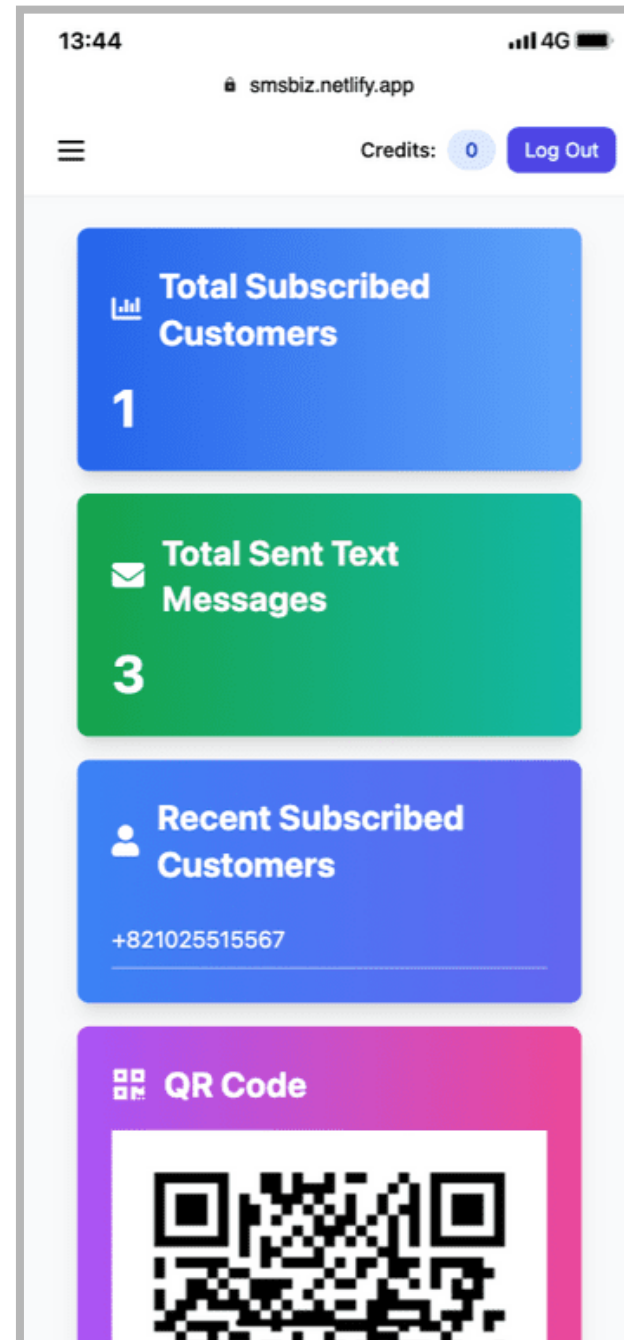
All Types

Description	Credits	Balance
Credits used for message: gg...	-1	1996
Credits used for message: koko...	-1	1997
Credits used for message: test2323...	-1	1998
Credits used for message: test2...	-1	1999

Credits

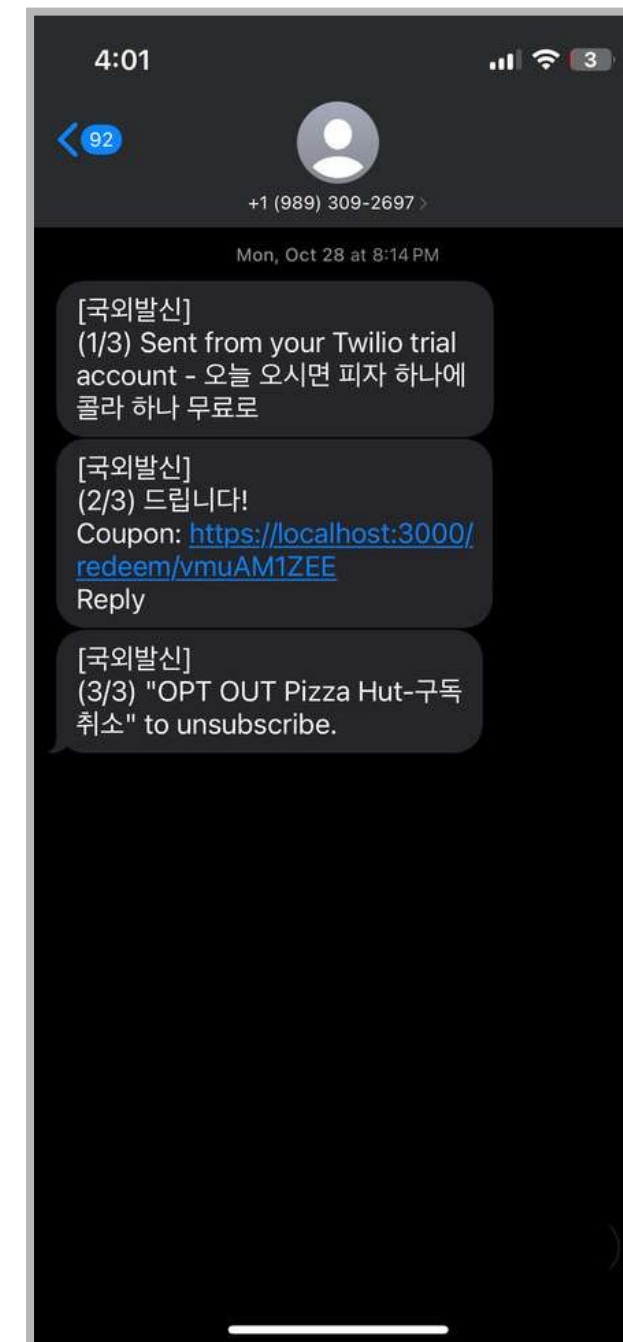
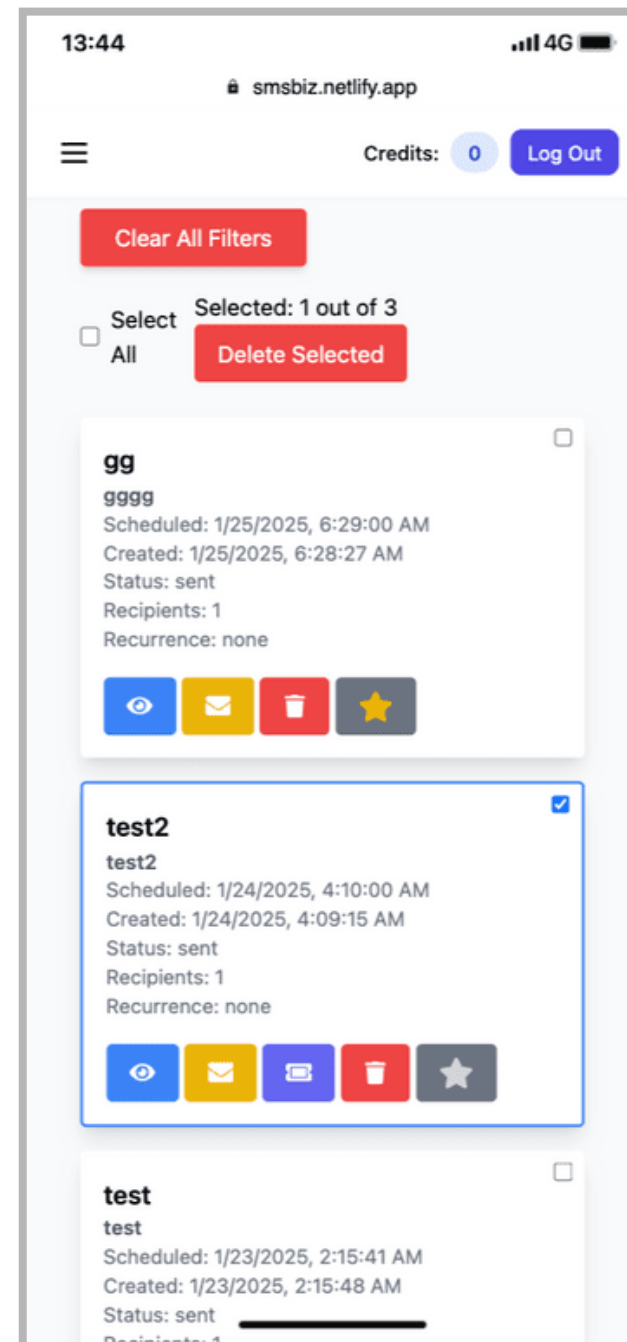
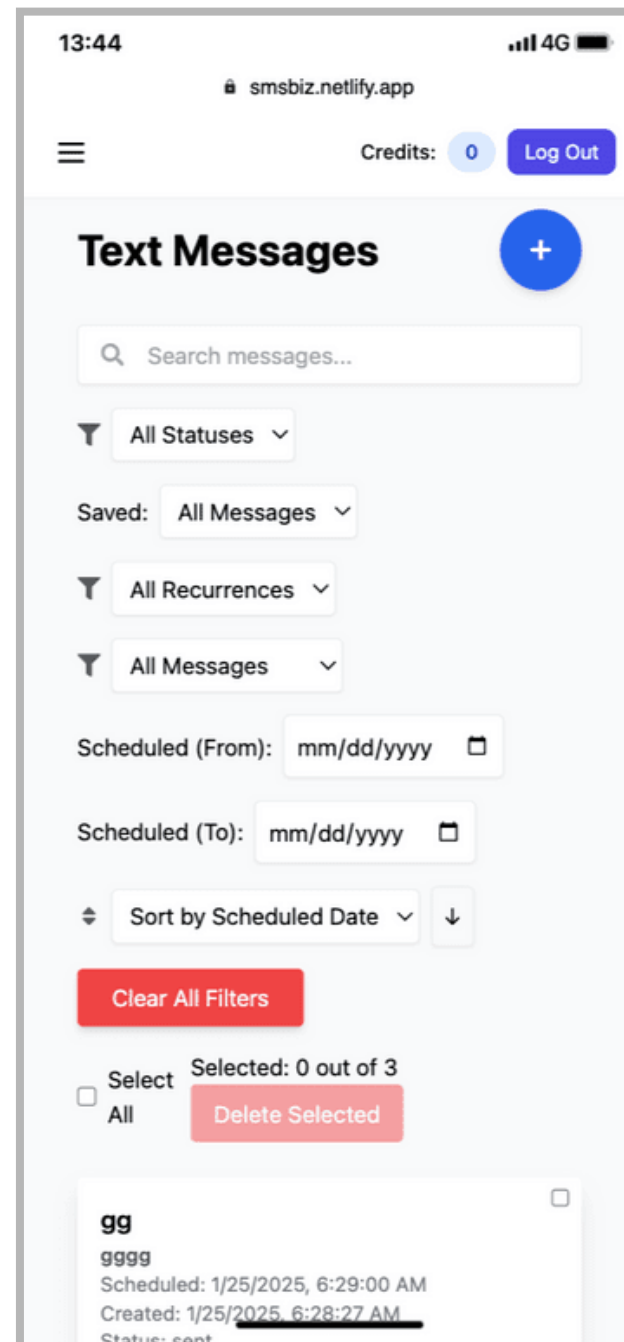
03 SMS Subscription & Marketing Service for Offline Stores

Business dashboard



03 SMS Subscription & Marketing Service for Offline Stores

Marketing message management,



03 SMS Subscription & Marketing Service for Offline Stores

Create new message

13:44 smsbiz.netlify.app Credits: 0 Log Out

Create New Message

Title
수요일 쿠폰

Message/Coupon Content 12/160
오늘 피자 1 + 1!

Scheduled Date
February 20, 2025 1:48 PM

Recurrence
Weekly

Recipients
☒ All Subscribers

Select Subscribers
Search phone number...
☒ +821025515567

13:44 smsbiz.netlify.app Credits: 0 Log Out

Message Type:
☐ Text
Send news, promo code, or events.
☒ Text + Coupon
Generate unique one-time coupon for each subscriber.

Coupon Expiration Days (Optional):
30

Coupon Expiration Date:
3/22/2025

Upload Image (Optional)
Drag 'n' drop an image here, or click to select one
Supported formats: JPG, PNG, GIF

Schedule Message

01:48 PM
Coupon:
localhost:3000/redeem/(unique coupon code)
Reply "OPT OUT 피자헛 강남-TmroFrNEx" to unsubscribe.

13:44 smsbiz.netlify.app Credits: 0 Log Out

Schedule Message

01:48 PM
Coupon:
localhost:3000/redeem/(unique coupon code)
Reply "OPT OUT 피자헛 강남-TmroFrNEx" to unsubscribe.

Coupon Preview
Please show this to one of the employees
Message Content:

13:44 smsbiz.netlify.app Credits: 0 Log Out

Coupon Preview
Please show this to one of the employees

Message Content:
Expiration Date:
3/22/2025

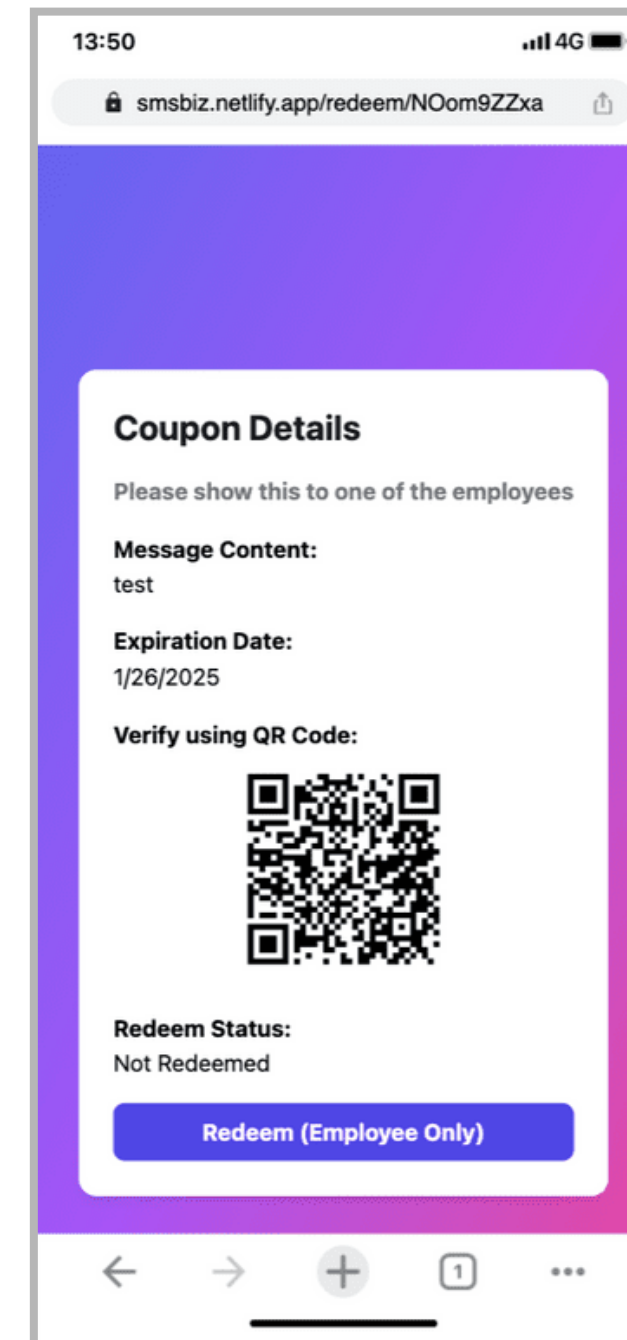
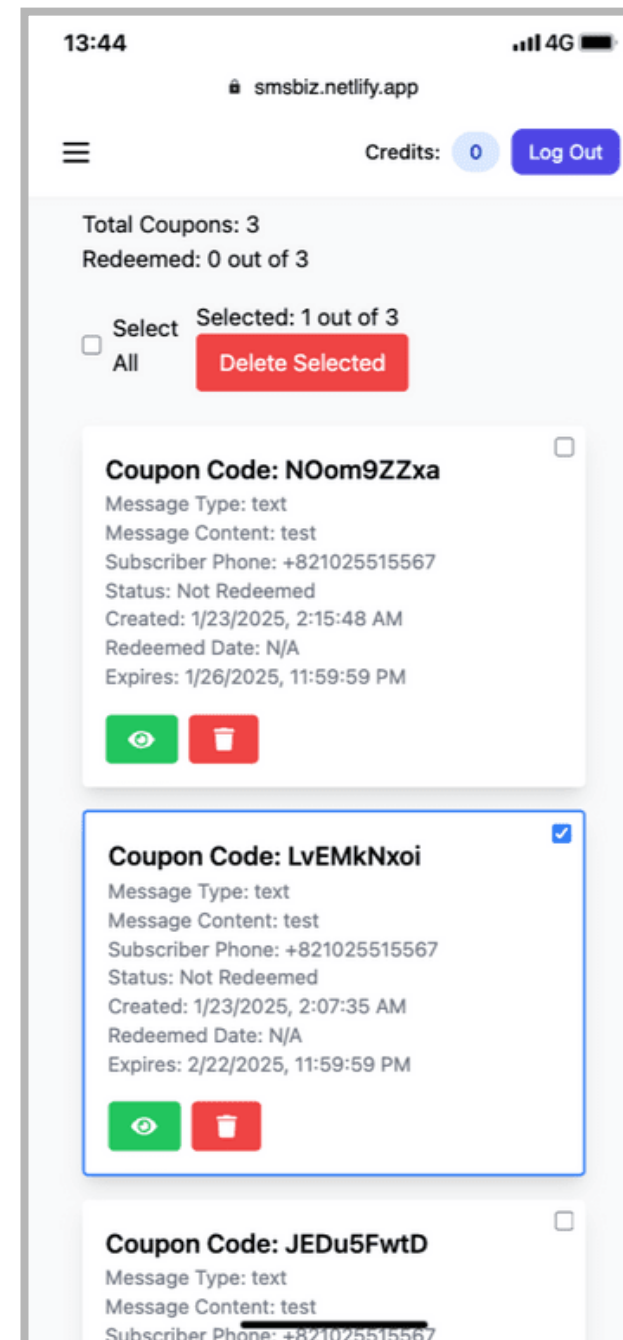
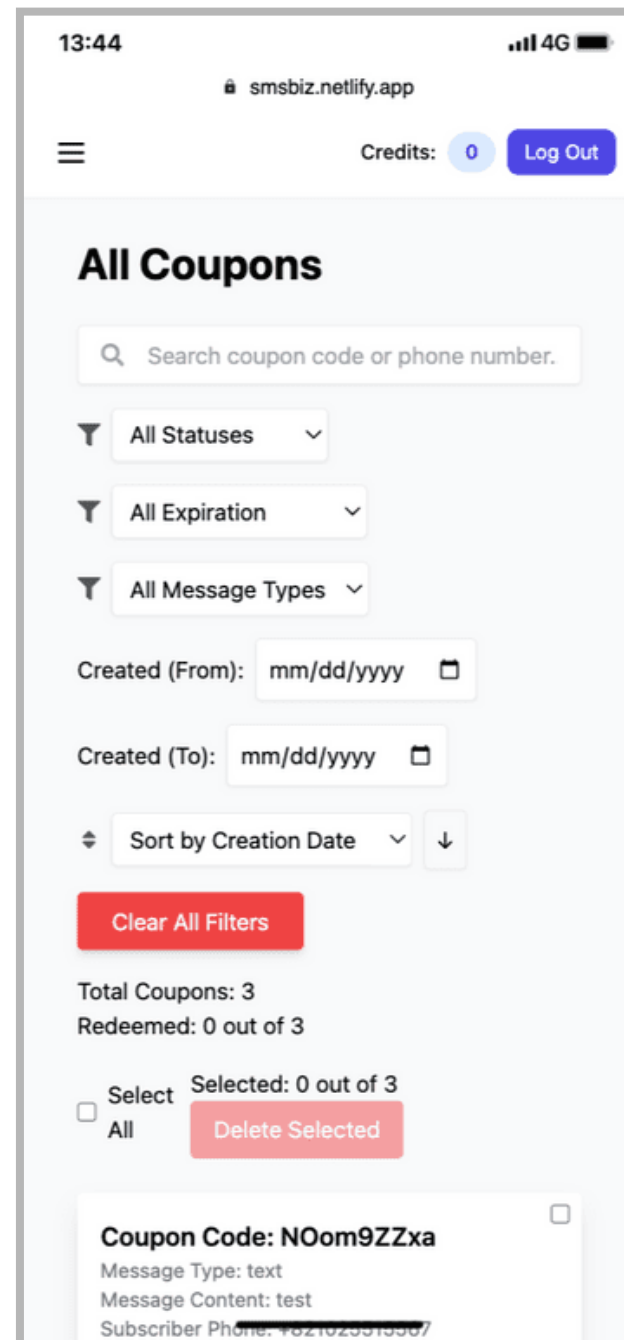
Verify using QR Code:
[QR Code]

Redeem Status:
Not Redeemed

Redeem (Employee Only)

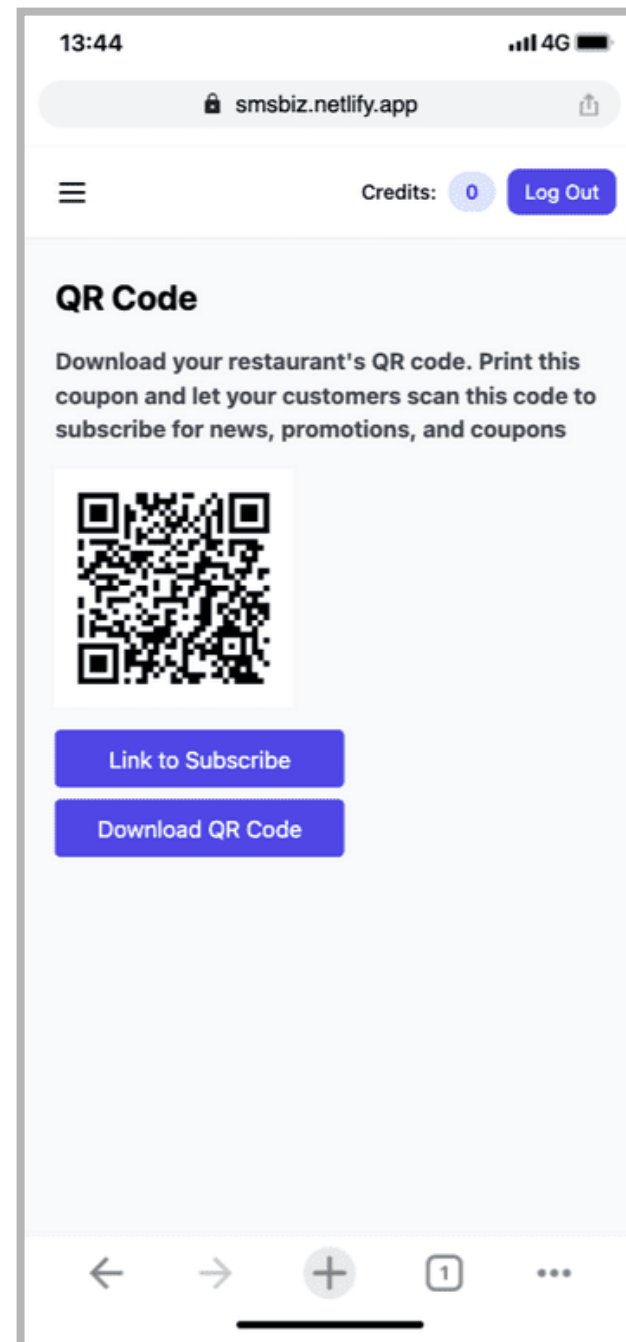
03 SMS Subscription & Marketing Service for Offline Stores

Coupon Management System



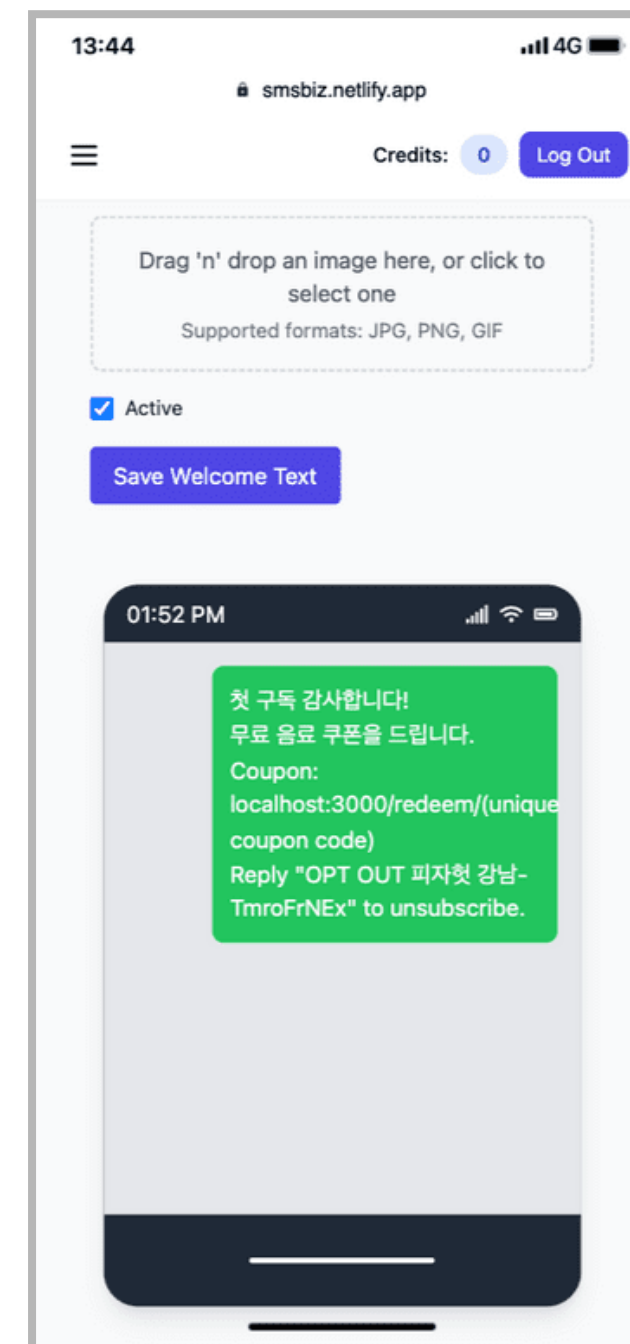
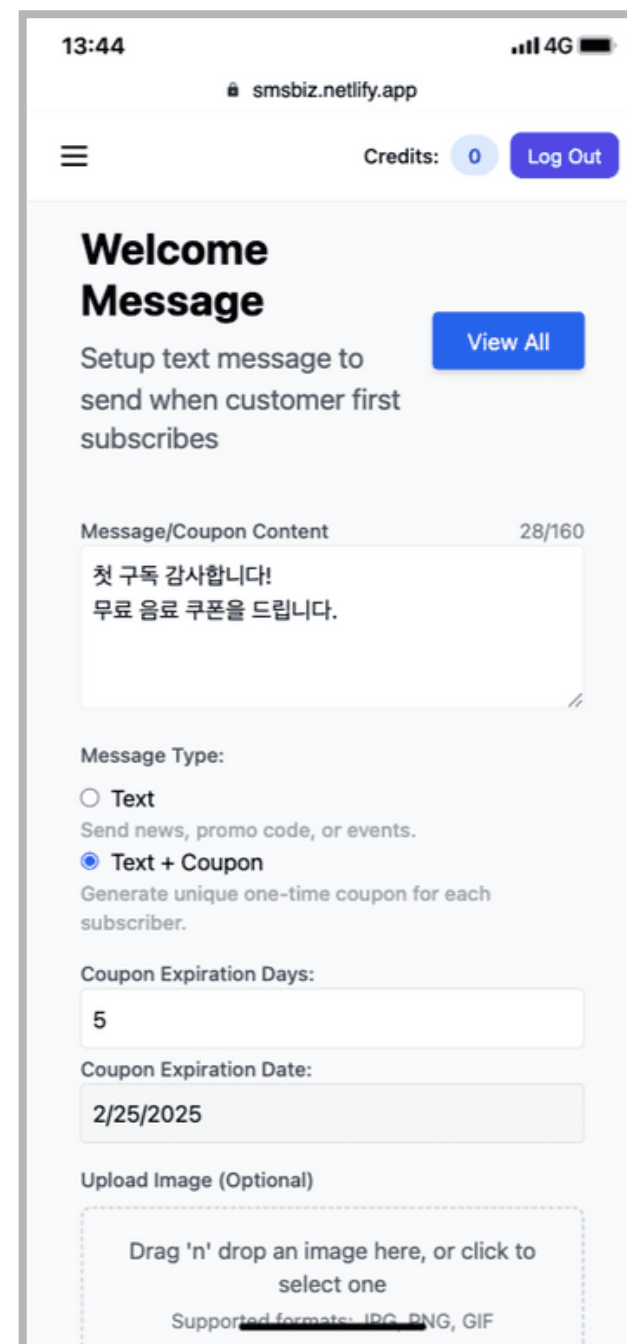
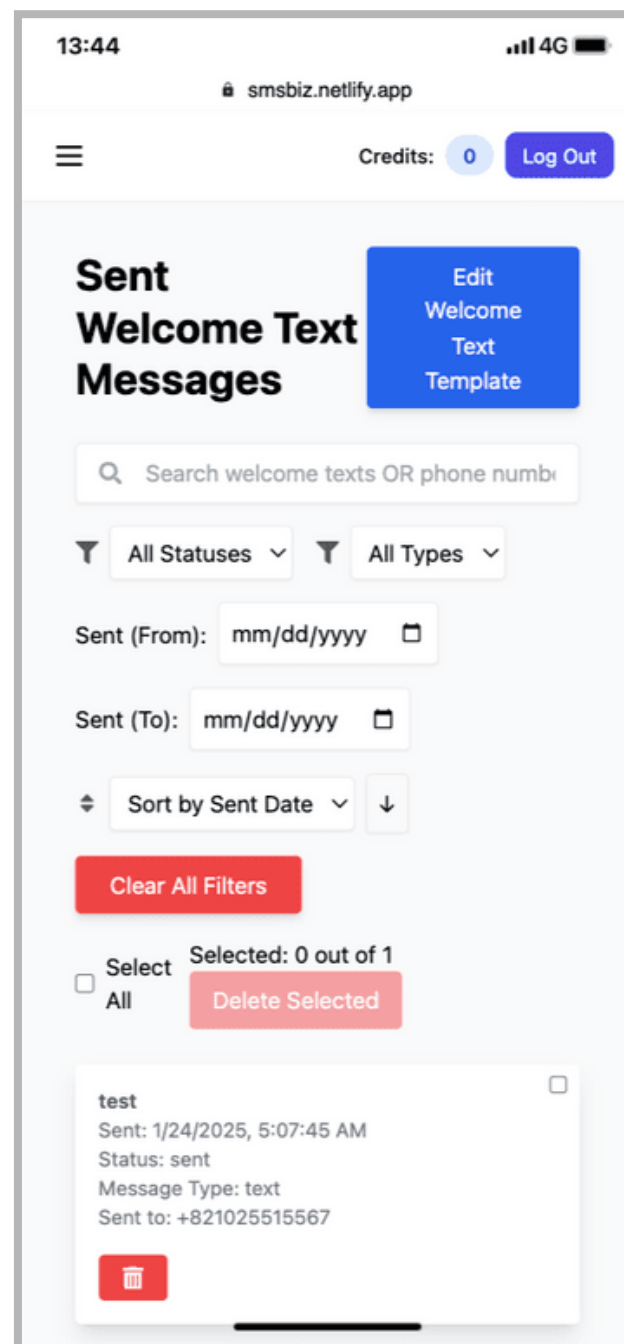
03 SMS Subscription & Marketing Service for Offline Stores

Business QR code generation



03 SMS Subscription & Marketing Service for Offline Stores

New subscriber message, coupon generation, template



03 SMS Subscription & Marketing Service for Offline Stores

Stamp Management

13:44

smsbiz.netlify.app

Credits: 0 Log Out

Stamp Reward Setup

Stamp can only be collected using NFC scan. Please follow our guidelines for setting up the NFC scan service. If you have any troubles, we will help you so please contact ____

Stamp Settings

☐ Enable Stamp System

Max Stamps Per Day

1

Cooldown Period (minutes)

10

Save Settings

Create New Stamp Reward

Stamp Cost

Reward Description

Expiration Days After Redeem (Optional)

Create Stamp Reward

13:44

smsbiz.netlify.app

Credits: 0 Log Out

Create New Stamp Reward

Stamp Cost

Reward Description

Expiration Days After Redeem (Optional)

Create Stamp Reward

Coupon Preview

Please show this to one of the employees

Coupon Description:

Expiration Days:
No expiration

Verify using QR Code:

Redeem Status:
Not Redeemed

Redeem (Employee Only)

13:44

smsbiz.netlify.app

Credits: 0 Log Out

Coupon Preview

Please show this to one of the employees

Coupon Description:

Expiration Days:
No expiration

Verify using QR Code:

Redeem Status:
Not Redeemed

Redeem (Employee Only)

Existing Stamp Rewards

피자 한판 무료

Stamp Cost: 20

Expiration Days: N/A

13:44

smsbiz.netlify.app

Credits: 0 Log Out

Coupon Preview

Please show this to one of the employees

Coupon Description:

Expiration Days:
No expiration

Verify using QR Code:

Redeem Status:
Not Redeemed

Redeem (Employee Only)

Existing Stamp Rewards

피자 한판 무료

Stamp Cost: 20

Expiration Days: N/A

free drink

Stamp Cost: 5

Expiration Days: 15

03 SMS Subscription & Marketing Service for Offline Stores

Opt Out subscription cancellation

13:44 4G

smsbiz.netlify.app

Credits: 0 Log Out

Change Your OPT OUT Message

Subscribers can opt out of text message by replying: "OPT OUT (your keyword)"

Your Keyword:

피자헛 강남 구독취소

Update

Edit restaurant details

13:44 4G

smsbiz.netlify.app

Credits: 0 Log Out

Edit Restaurant Details

Restaurant Name

피자헛 강남

Address

Description (optional)

Profile Image

Drag 'n' drop an image here, or click to select one

Supported formats: JPG, PNG, GIF

Image Preview

13:44 4G

smsbiz.netlify.app

Credits: 0 Log Out

Supported formats: JPG, PNG, GIF

Image Preview

Current image Remove

Update

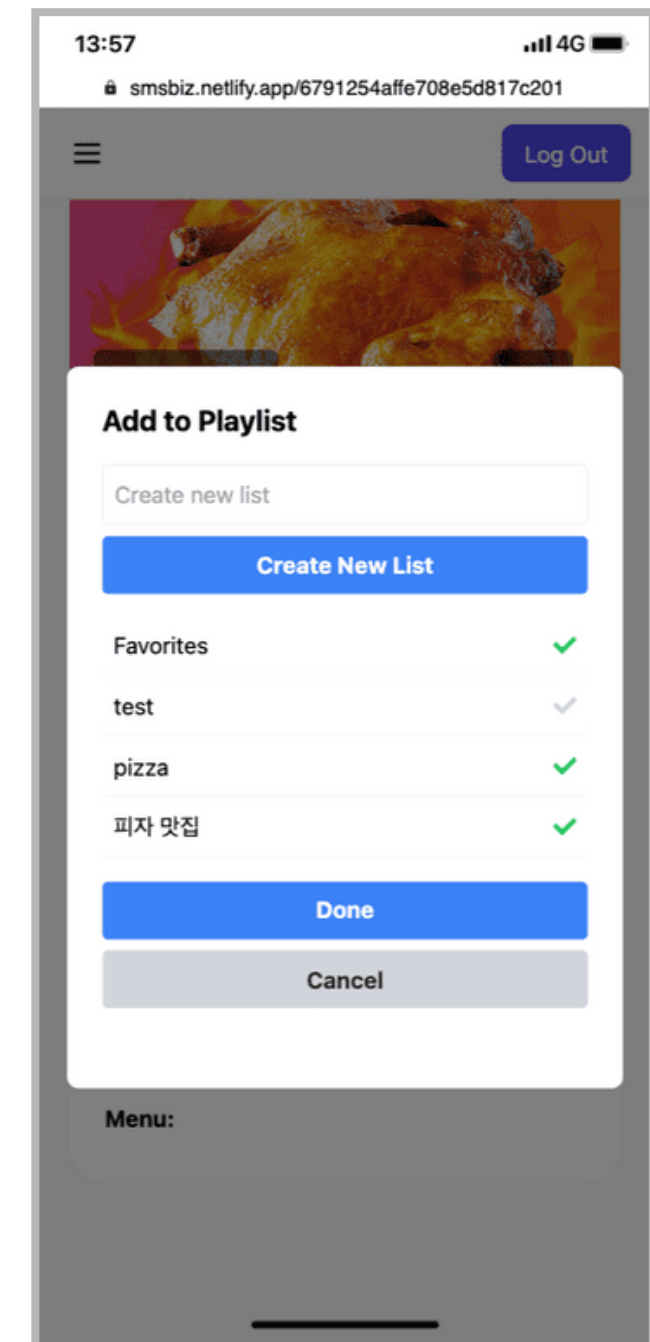
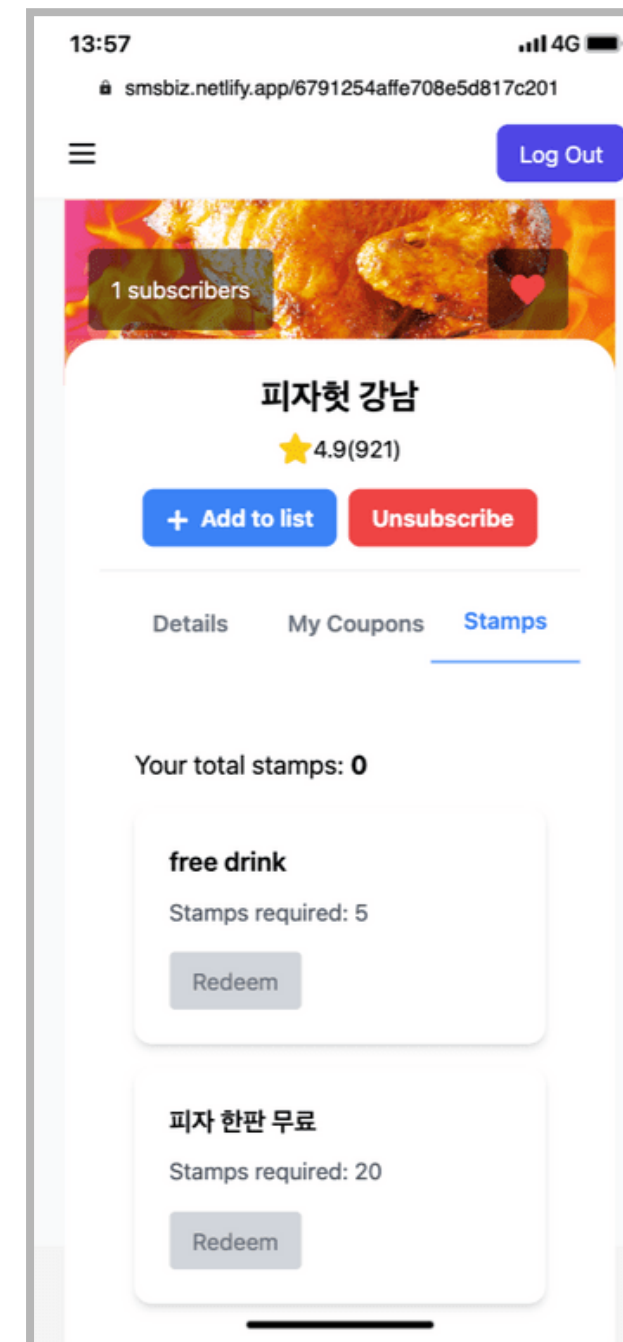
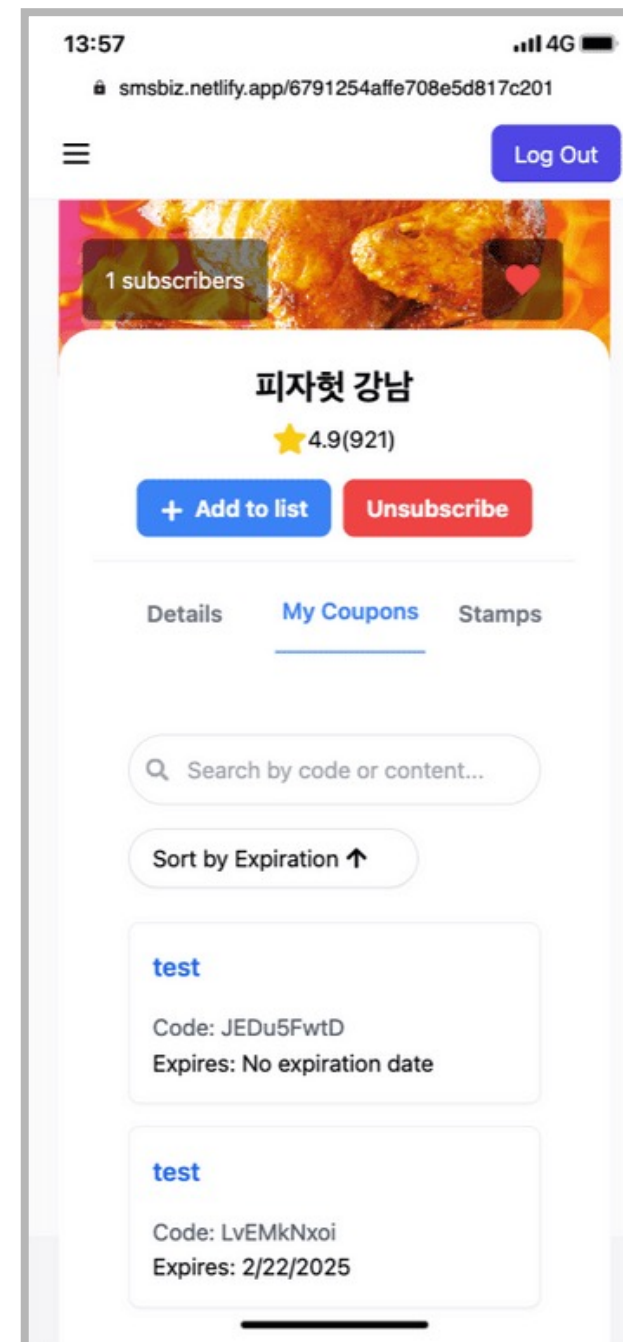
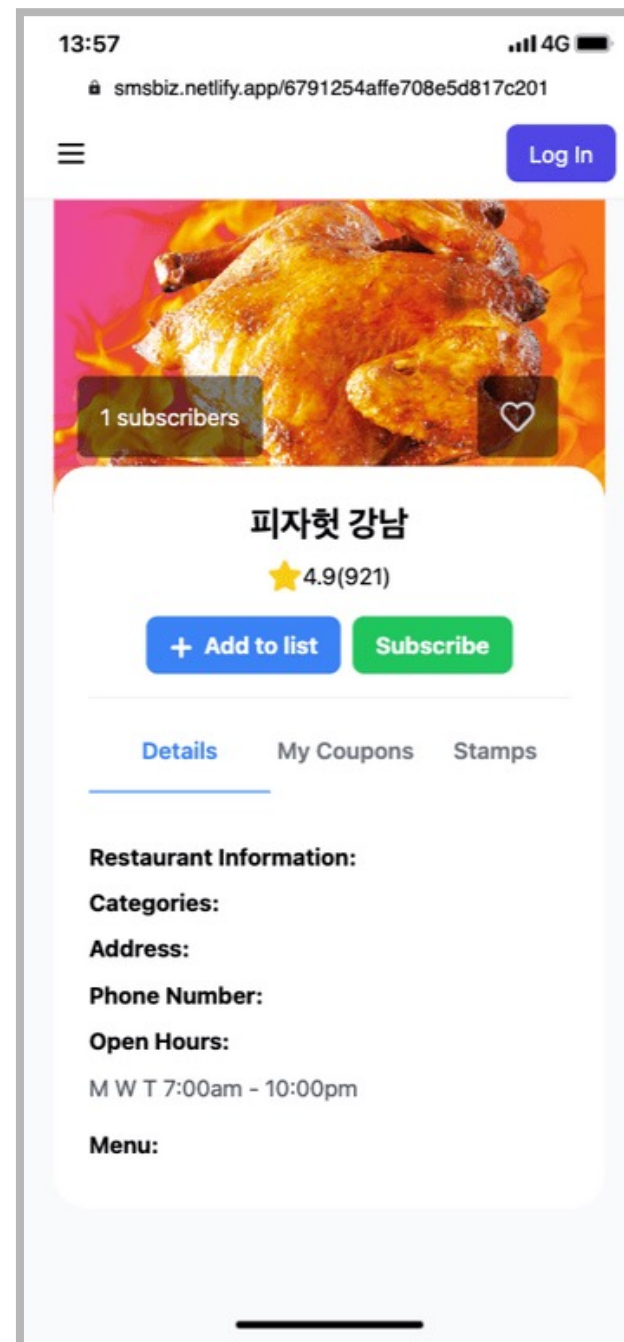
Delete Restaurant

Warning: This action cannot be undone.

Delete Restaurant

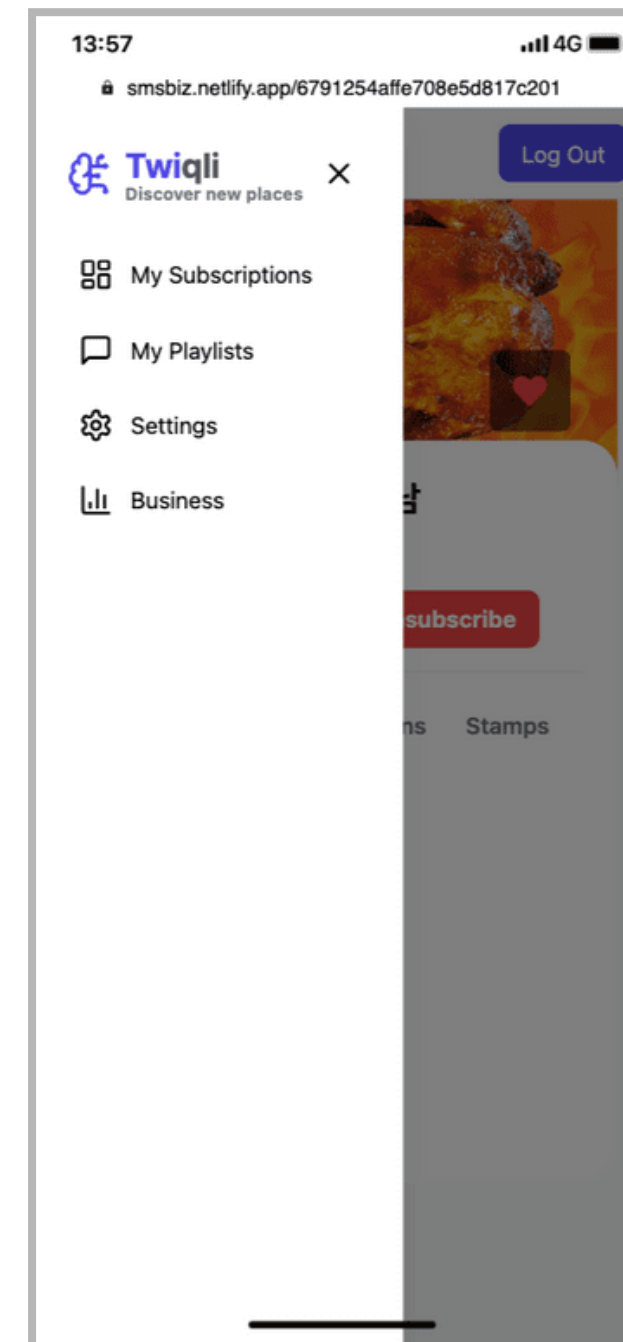
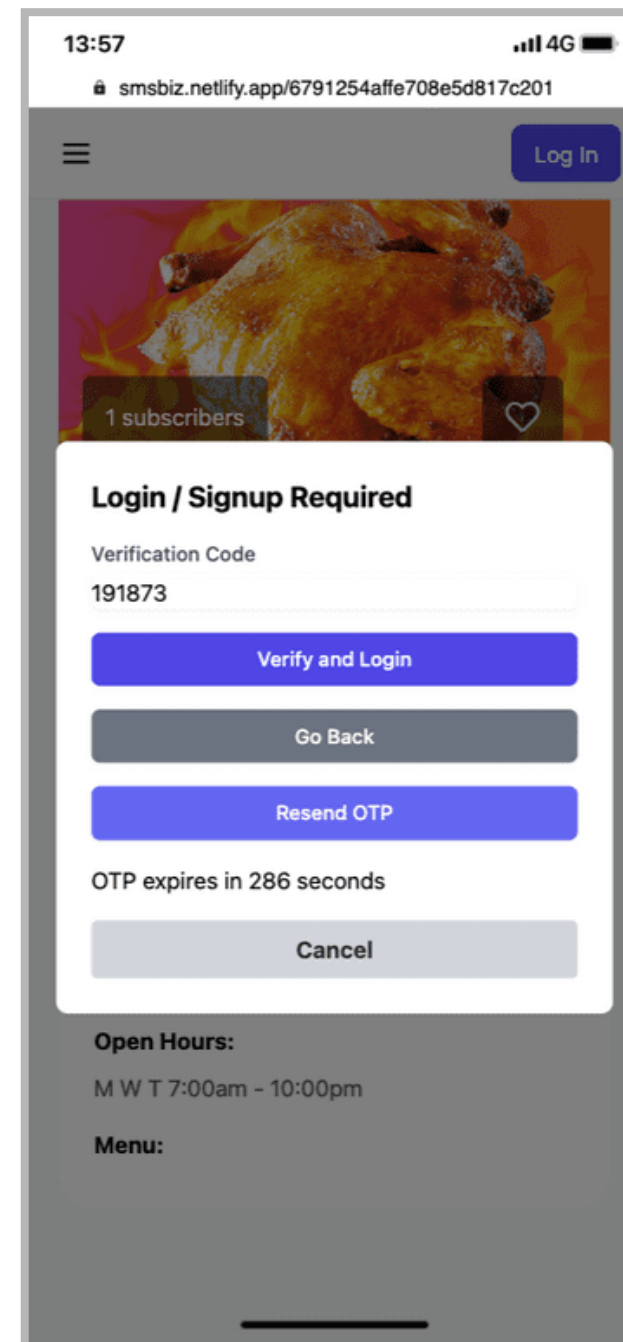
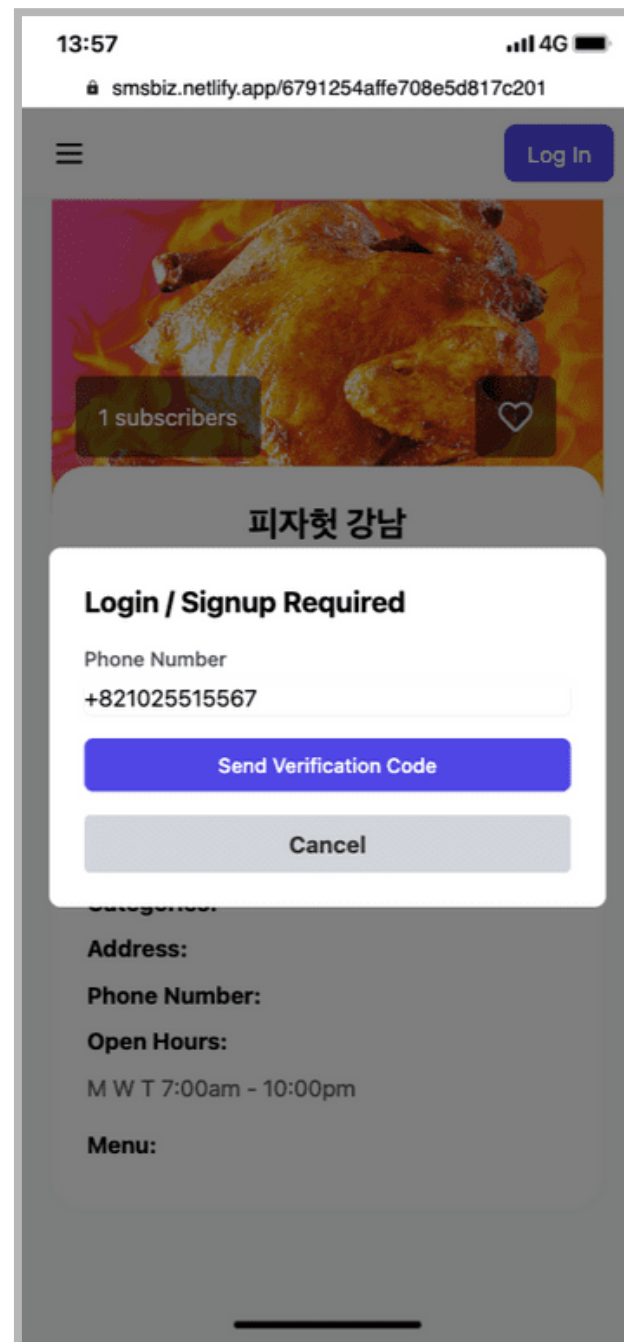
03 SMS Subscription & Marketing Service for Offline Stores

Customer page, subscribe, playlist, coupons, stamps



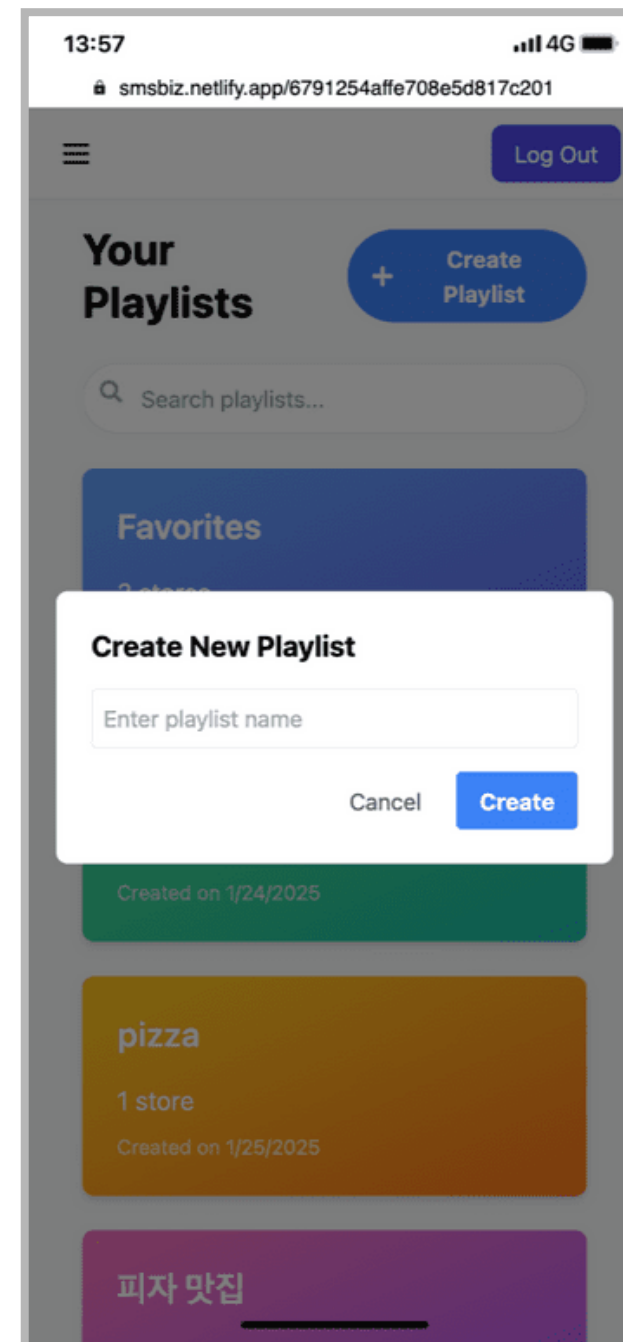
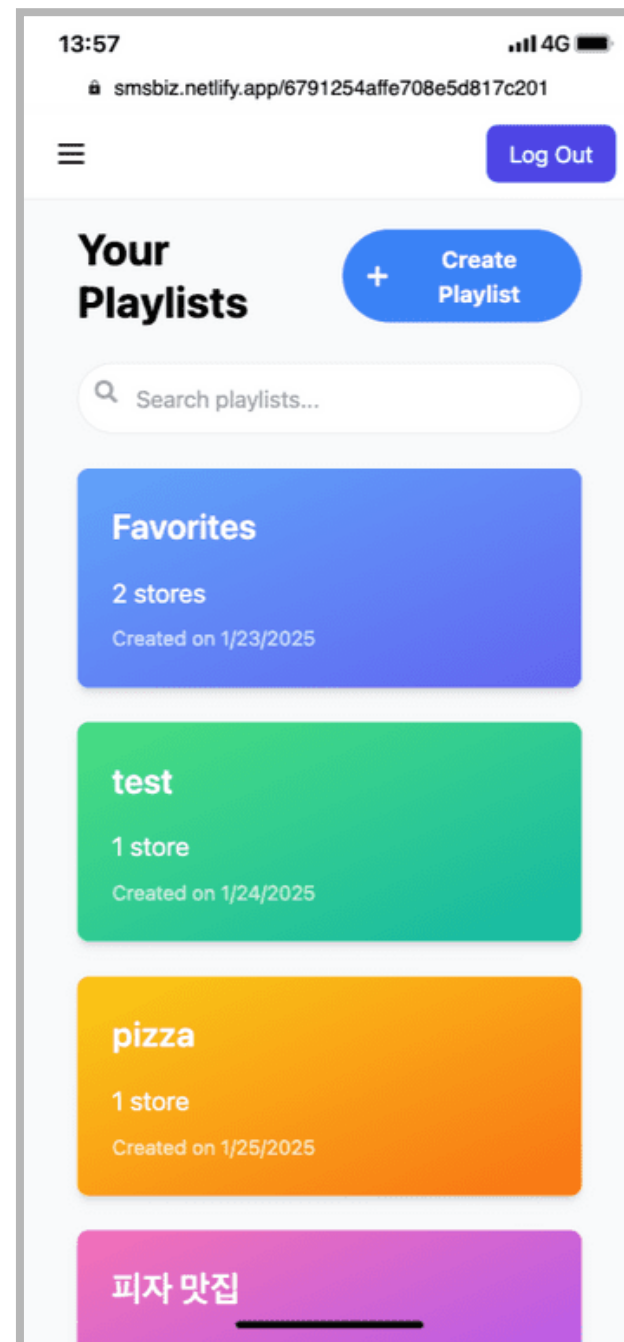
03 SMS Subscription & Marketing Service for Offline Stores

Subscribe to business, OTP verification

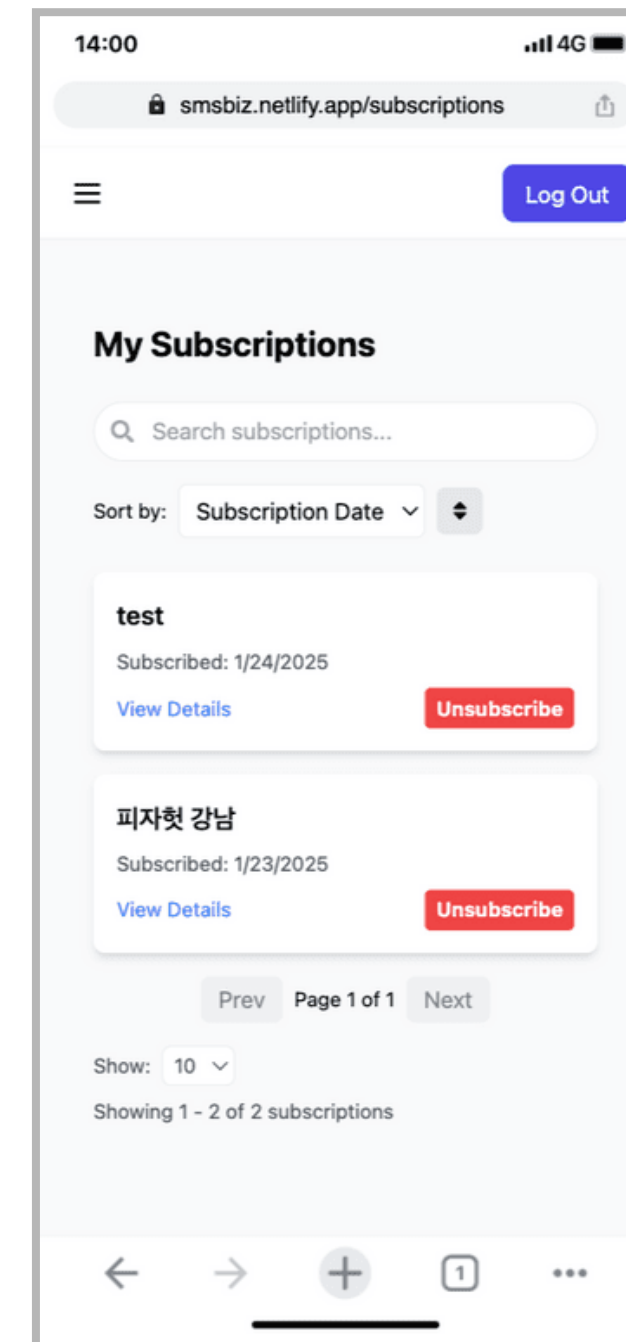


03 SMS Subscription & Marketing Service for Offline Stores

Playlist Management



Customer Subscriptions Management



03 SMS Subscription & Marketing Service for Offline Stores

Customer Account settings

14:00

smsbiz.netlify.app/subscriptions

Log Out

Account Settings

Profile Information

Name (Optional)

Your name

Email

Your email

✓ Update Profile

Update Password

Current Password

Enter current password

New Password

Enter new password

Confirm New Password

Confirm new password

14:00

smsbiz.netlify.app/subscriptions

Log Out

Update Password

Current Password

Enter current password

New Password

Enter new password

Confirm New Password

Confirm new password

✓ Update Password

Phone Number

Current Phone Number

New Phone Number

Enter new phone number

Send OTP

⚠ Danger Zone

14:00

smsbiz.netlify.app/subscriptions

Log Out

Confirm New Password

Confirm new password

✓ Update Password

Phone Number

Current Phone Number

New Phone Number

Enter new phone number

Send OTP

⚠ Danger Zone

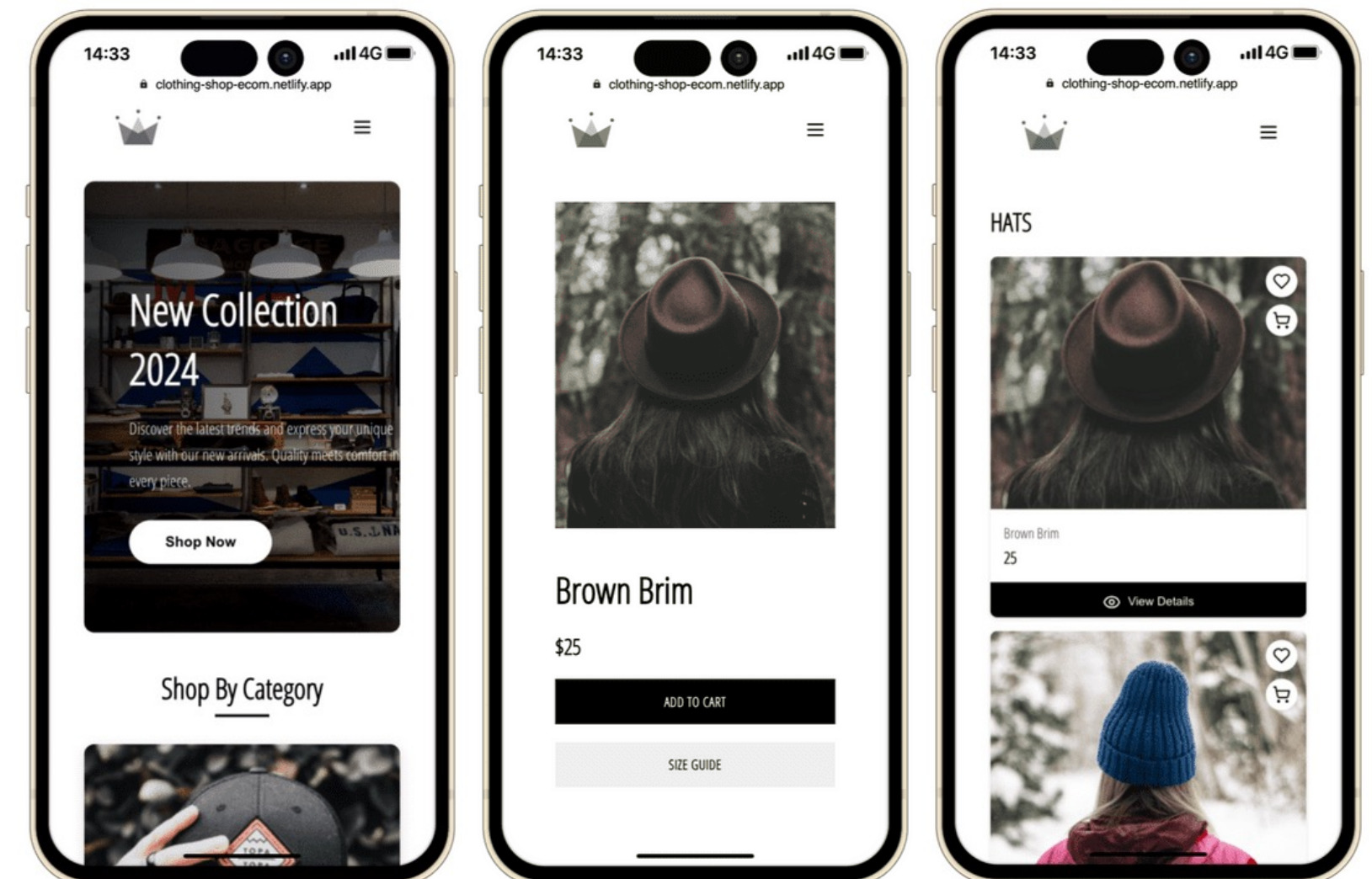
Once you delete your account, there is no going back. Please be certain.

Delete Account

04 E-commerce Site Intro

As an e-commerce platform, it provides a comprehensive shopping experience including real-time product updates, user authentication, shopping cart management, wishlist functionality, and product reviews.

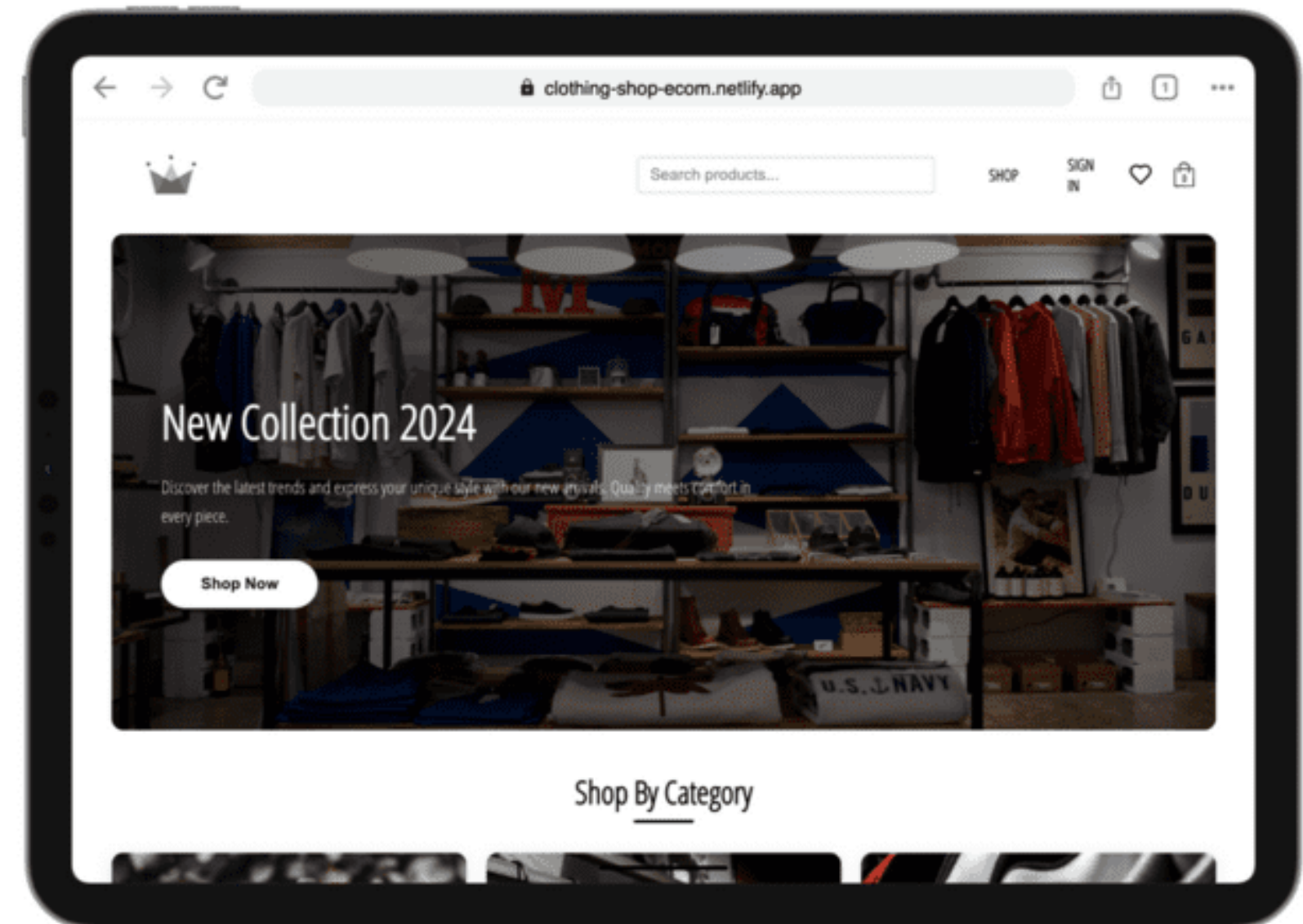
Development Period:	• Planning and design: 2024.11.21 ~ 2024.11.27 • Development: 2024.11.28 ~ 2025.01.02 • Testing and deployment: 2025.01.03 ~ 2025.01.10
My Role:	• Frontend, Backend • Service planning and direction setting • Main logic implementation, API design, DB architecture
Github	https://github.com/whd793/Shopping-ecommerce
Demo	https://clothing-shop-ecom.netlify.app/



04 E-commerce Site

Development Environment:

- Frontend Core: React 18, Redux Toolkit
- Styling: Styled-components
- Backend: Firebase (Auth, Firestore)
- State Management: Redux Toolkit, Redux-persist
- Routing: React Router v6



04 E-commerce Site

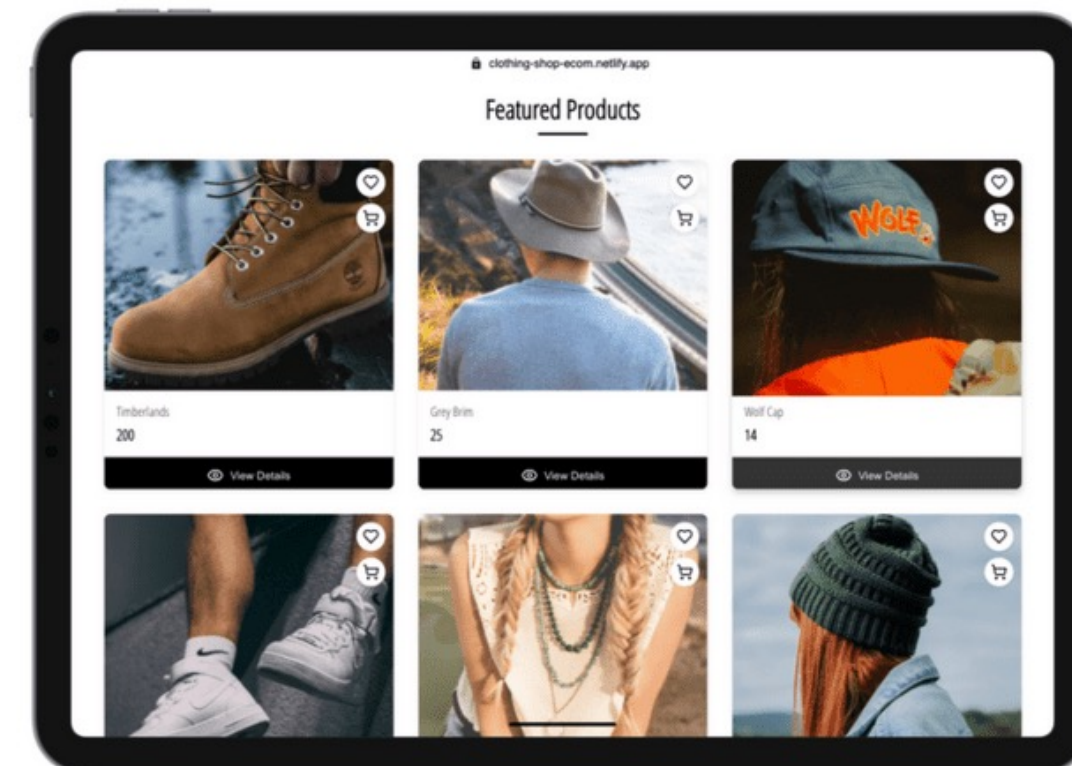
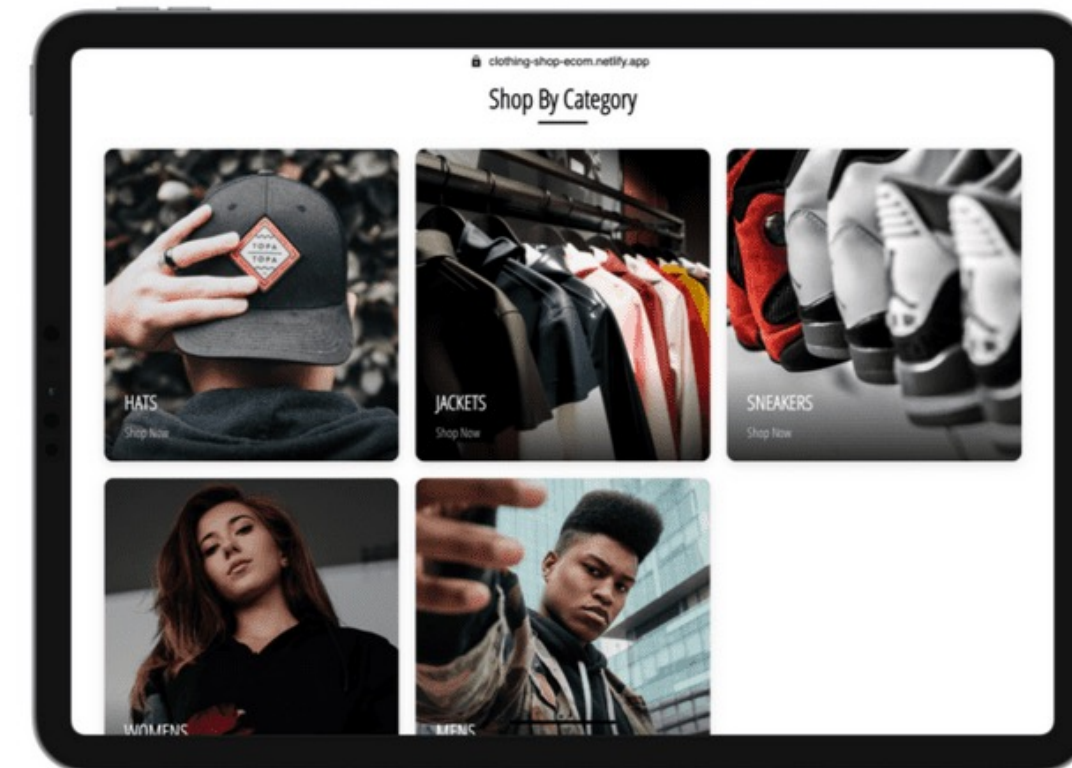
Core Features:

Global State Management System using Redux Toolkit:

- Efficient management of shopping cart, wishlist, and user authentication states
- Optimization of unnecessary re-rendering through Selector pattern
- Modularization of state management logic through Redux slice pattern
- Elimination of data duplication through Redux state normalization

Firebase-based User Authentication System:

- Email/password and Google OAuth login
- Firestore database design and query optimization
- Data access control through security rules configuration
- Real-time user session management implementation



04 E-commerce Site

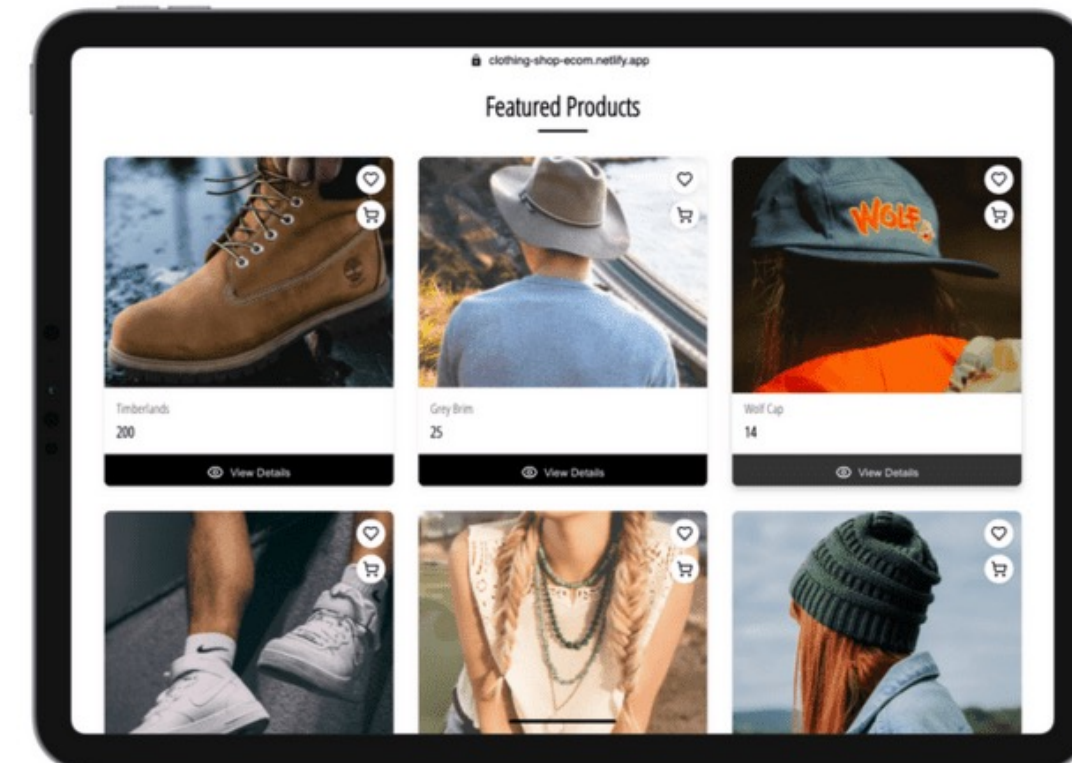
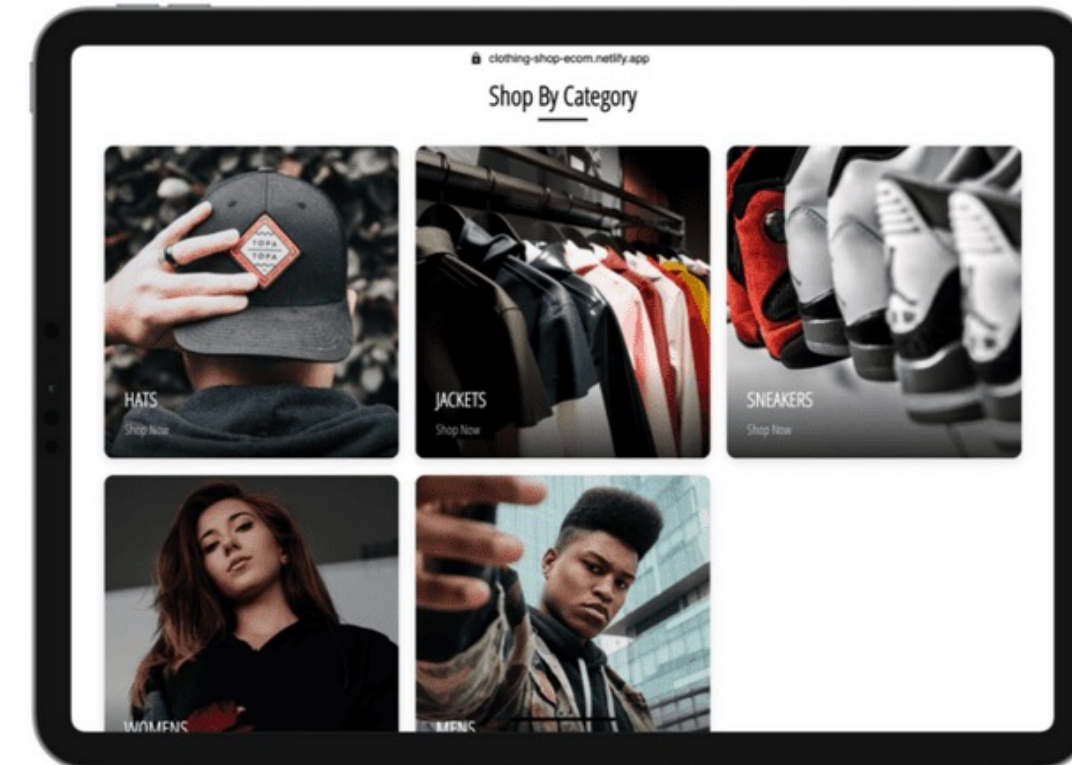
Core Features:

Product Management:

- Category-based product filtering
- Recommended products section
- Detailed product pages
- Product image optimization
- Recently viewed products tracking

Shopping Cart and Checkout System:

- Real-time shopping cart updates
- Quantity adjustment functionality
- Shopping cart state persistence
- Cart summary



04 E-commerce Site

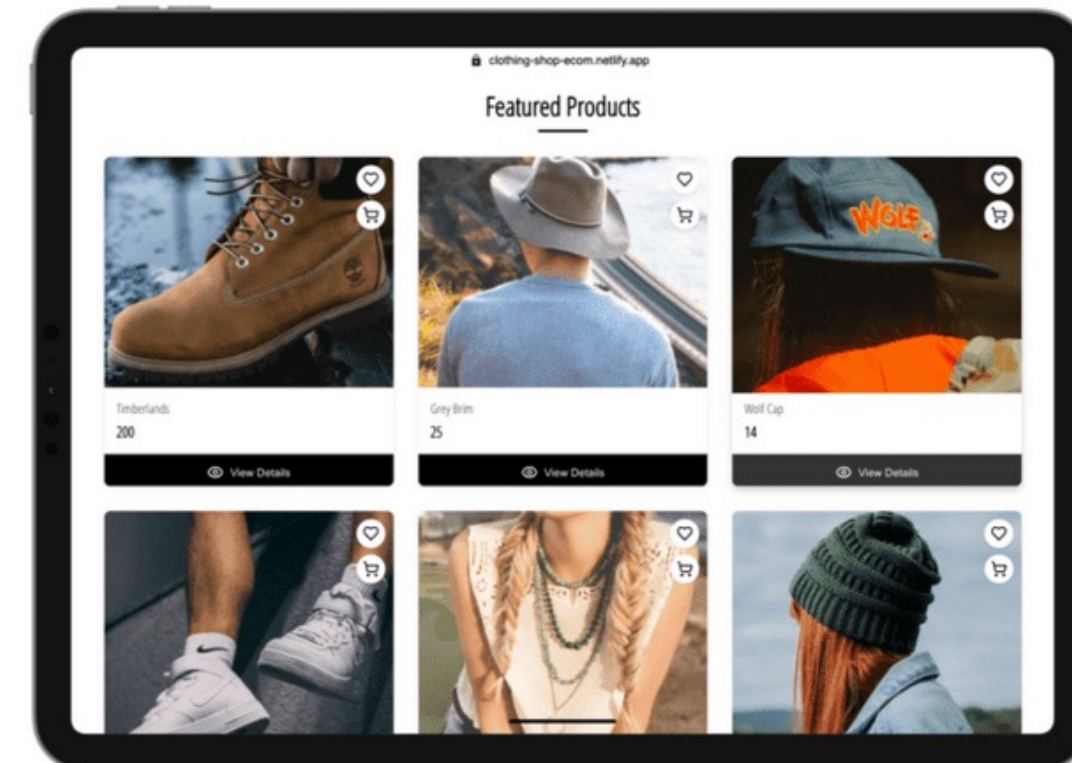
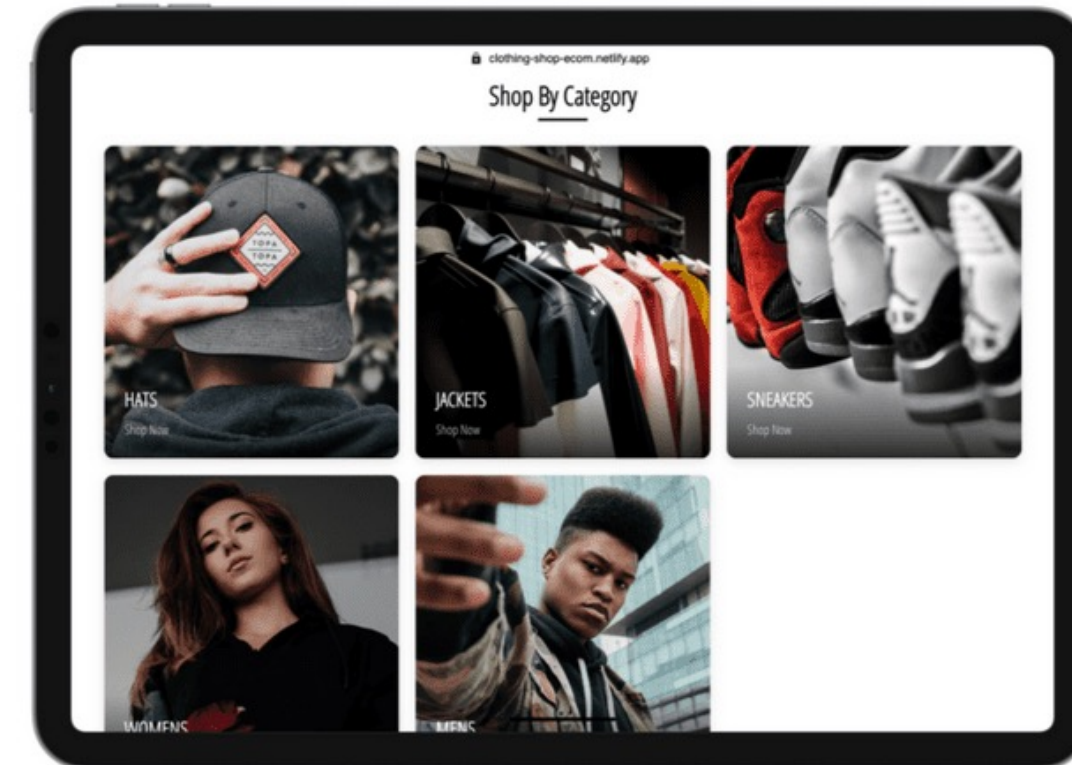
Core Features:

User Experience Optimization:

- Toast notification system for user action feedback
- Modern responsive design
- Interactive product cards

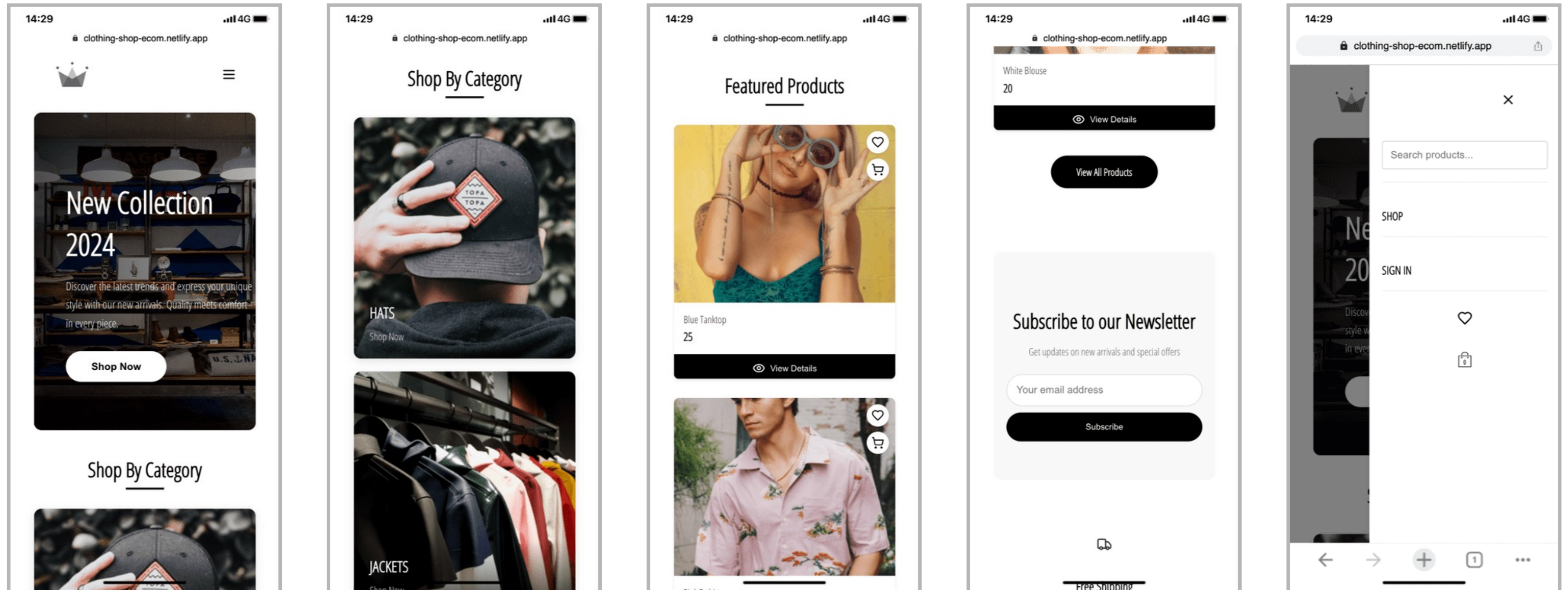
Performance Optimization:

- Initial loading time optimization through Code Splitting
- Component optimization using React.memo and useMemo
- Implementation of image lazy loading



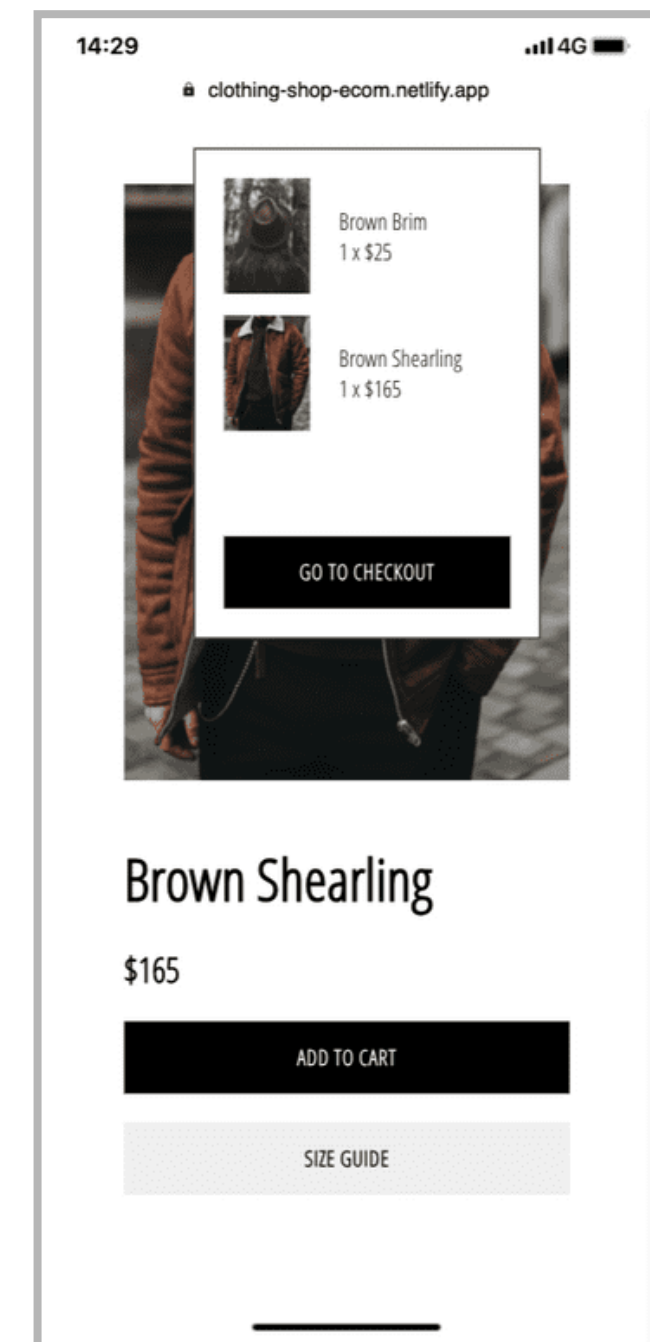
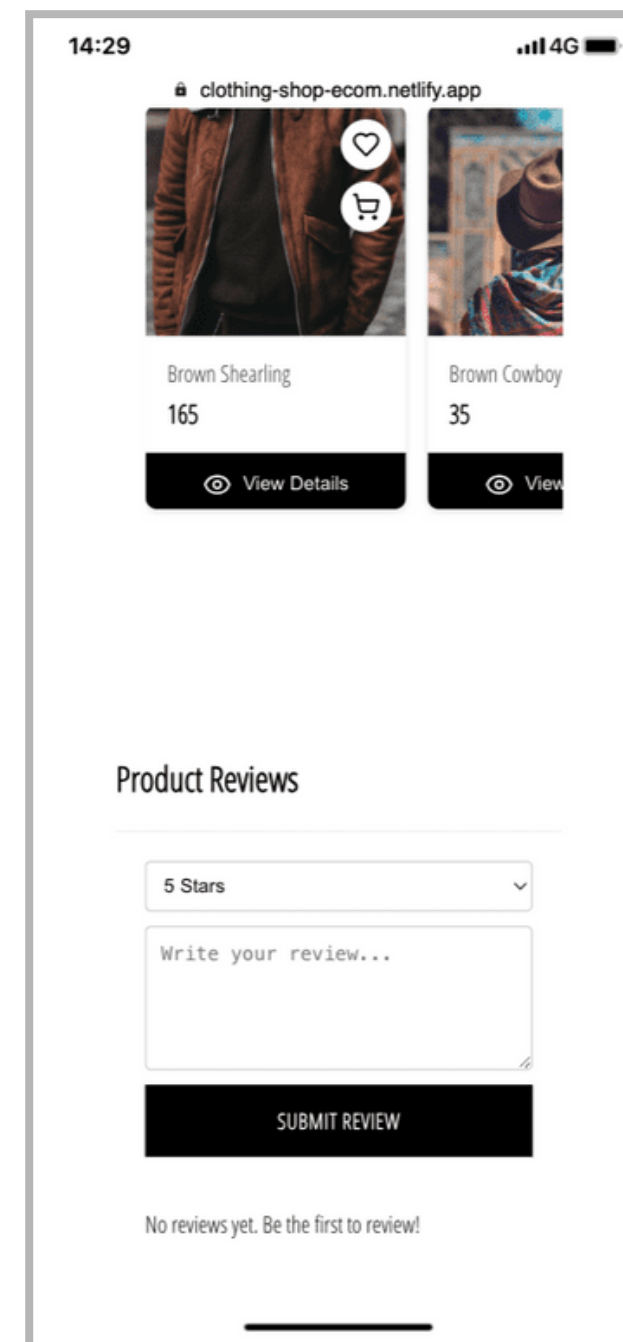
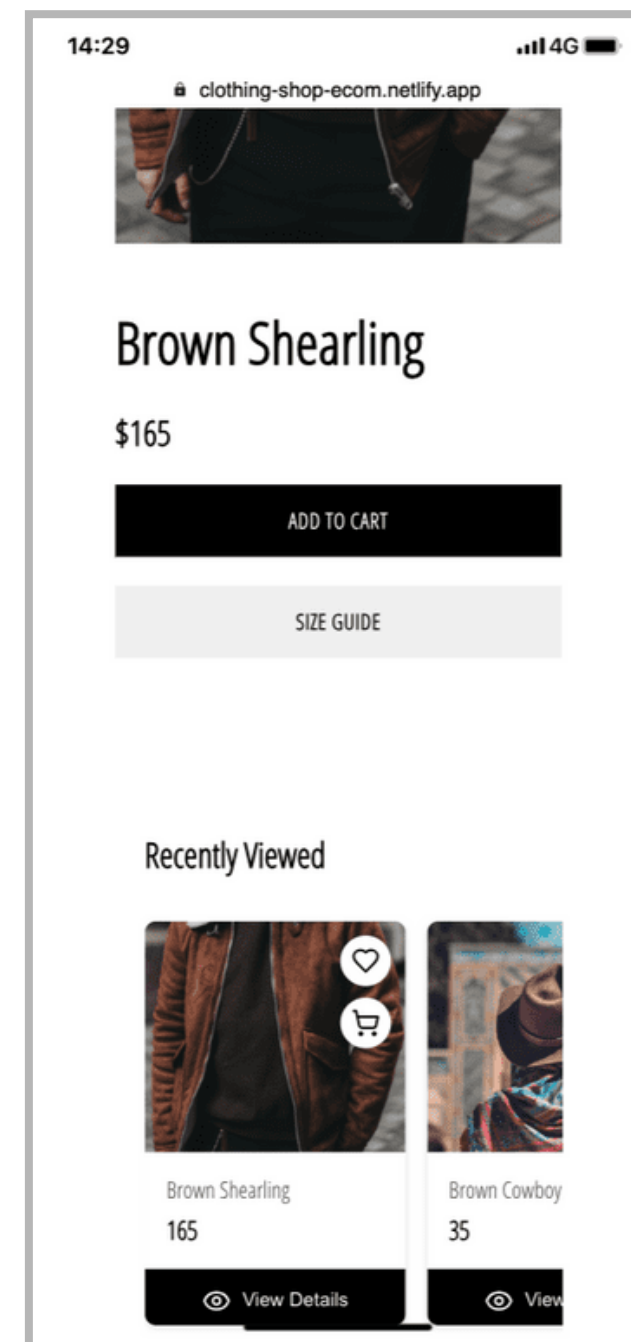
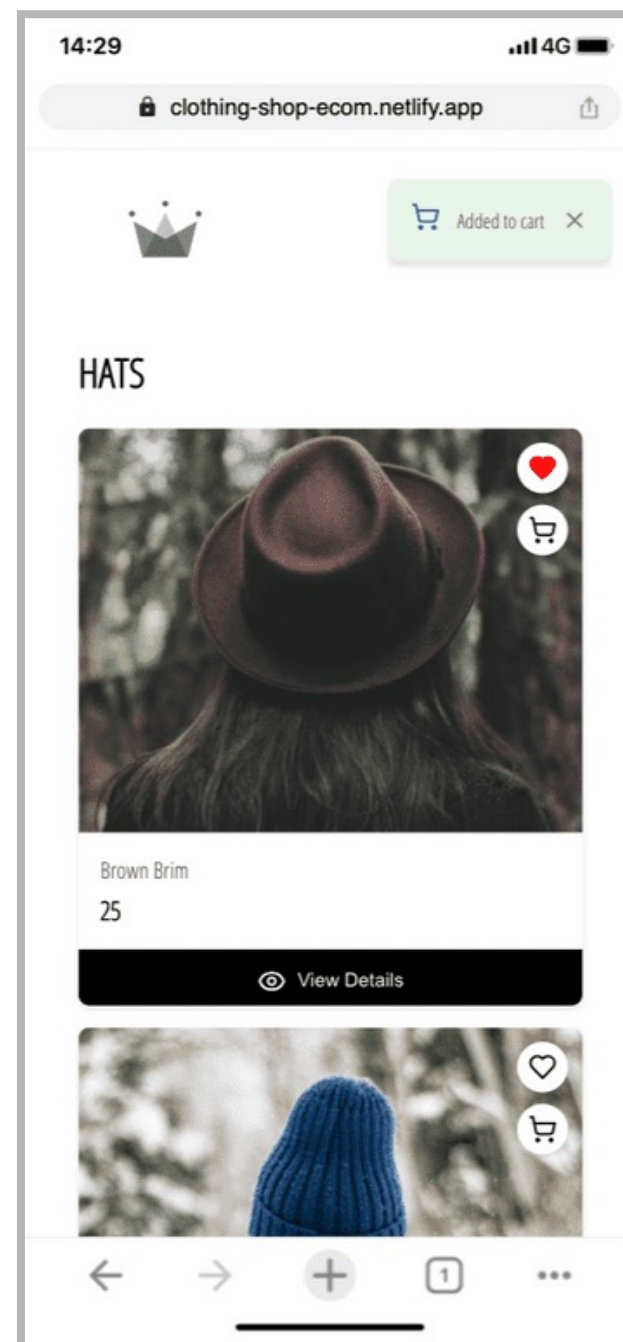
04 E-commerce Site

Homepage, Category Showcase, Recommended Products:



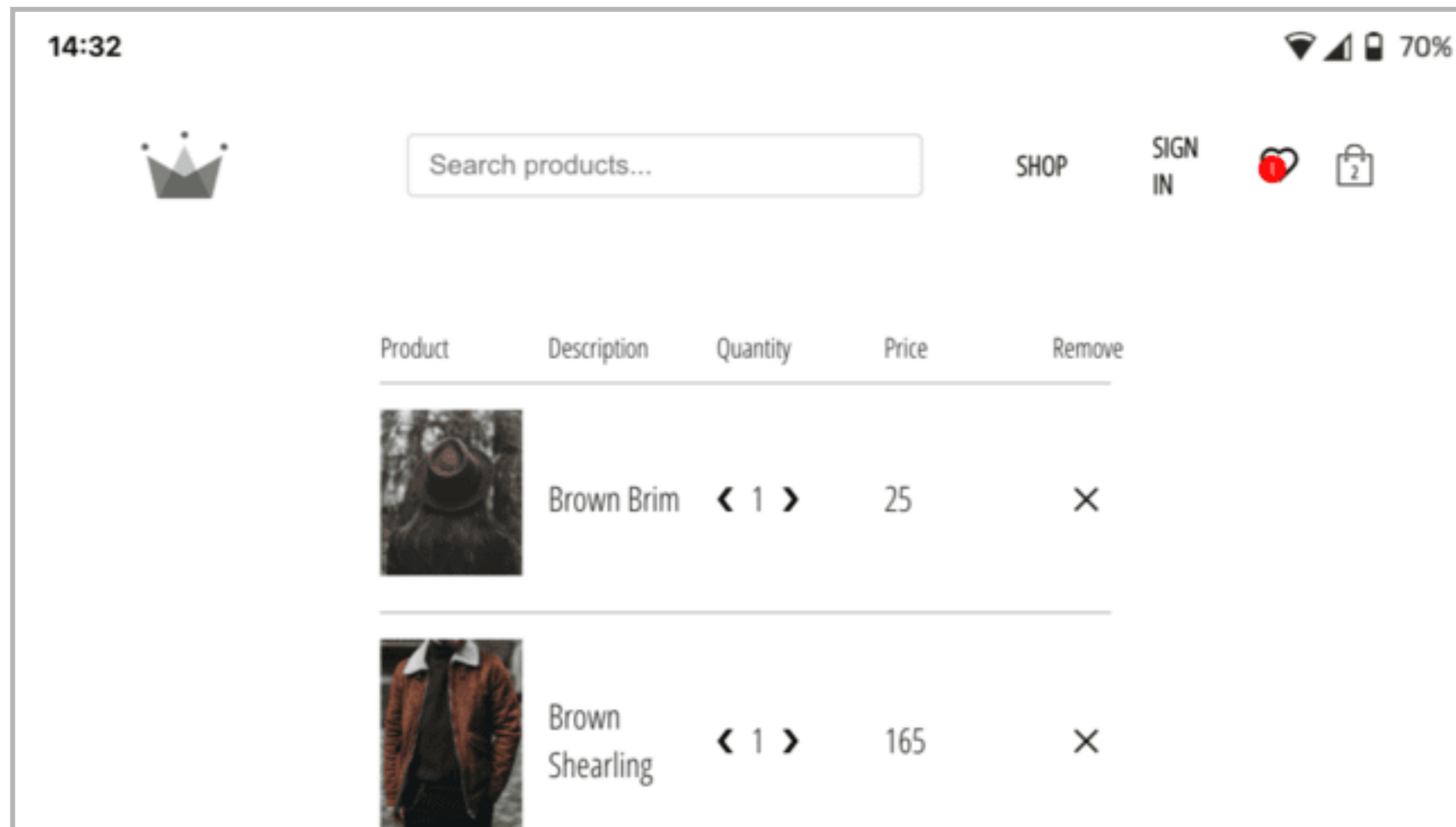
04 E-commerce Site

Detailed Product Page, Product Image Optimization, Recently Viewed Products Tracking, Product Review System:



04 E-commerce Site

Shopping Cart Summary, Quantity Adjustment Functionality:

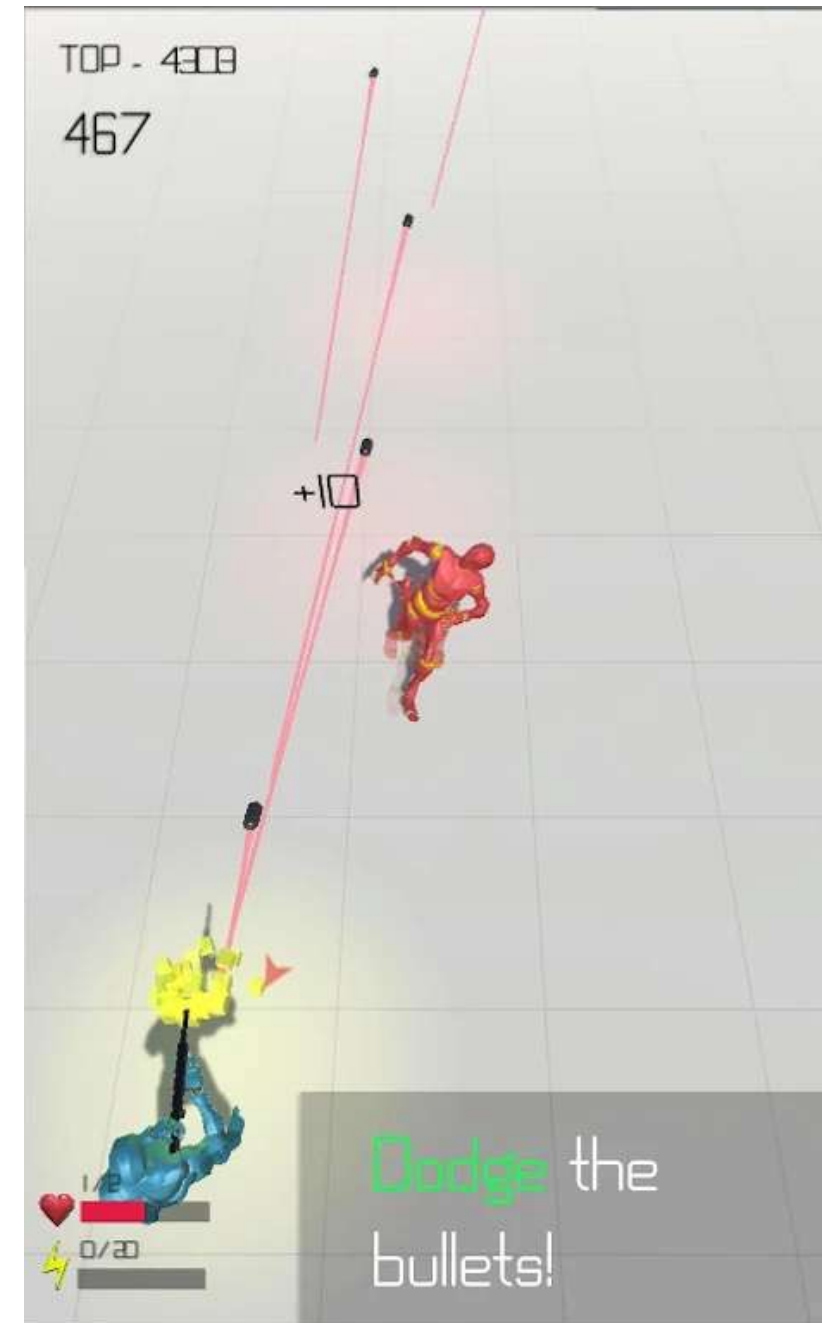


05 Speedy Hero - Mobile Game

Intro

Speedy Hero is a runner game where a character with super-speed ability runs endlessly avoiding obstacles. Players aim to achieve high scores by adjusting speed, avoiding obstacles, and outrunning AI enemies. The key technical features include disciplined use of physics engines, interaction with AI, and optimized graphics and performance.

I learned and applied various technologies necessary for Unity-based game development, such as physics engines, AI, graphics optimization, asynchronous loading, input systems, and UI/UX design. Through this project, I experienced the entire process from planning to prototype development, optimization, and deployment. I particularly grew as a developer by considering various solutions for performance optimization and player experience enhancement.



05 Speedy Hero - Mobile Game

Intro

Technologies : Unity (C#), NavMesh AI, Universal Render Pipeline (URP), ShaderLab, Raycasting, Object Pooling, Game Analytics, Facebook Analytics

Development Period:

- Planning and design: 2017.01.24 ~ 2017.02.07
- Development: 2017.02.08 ~ 2017.04.12
- Testing, deployment, optimization: 2017.04.13 ~ 2017.05.04

Team PM, Designer, Developer

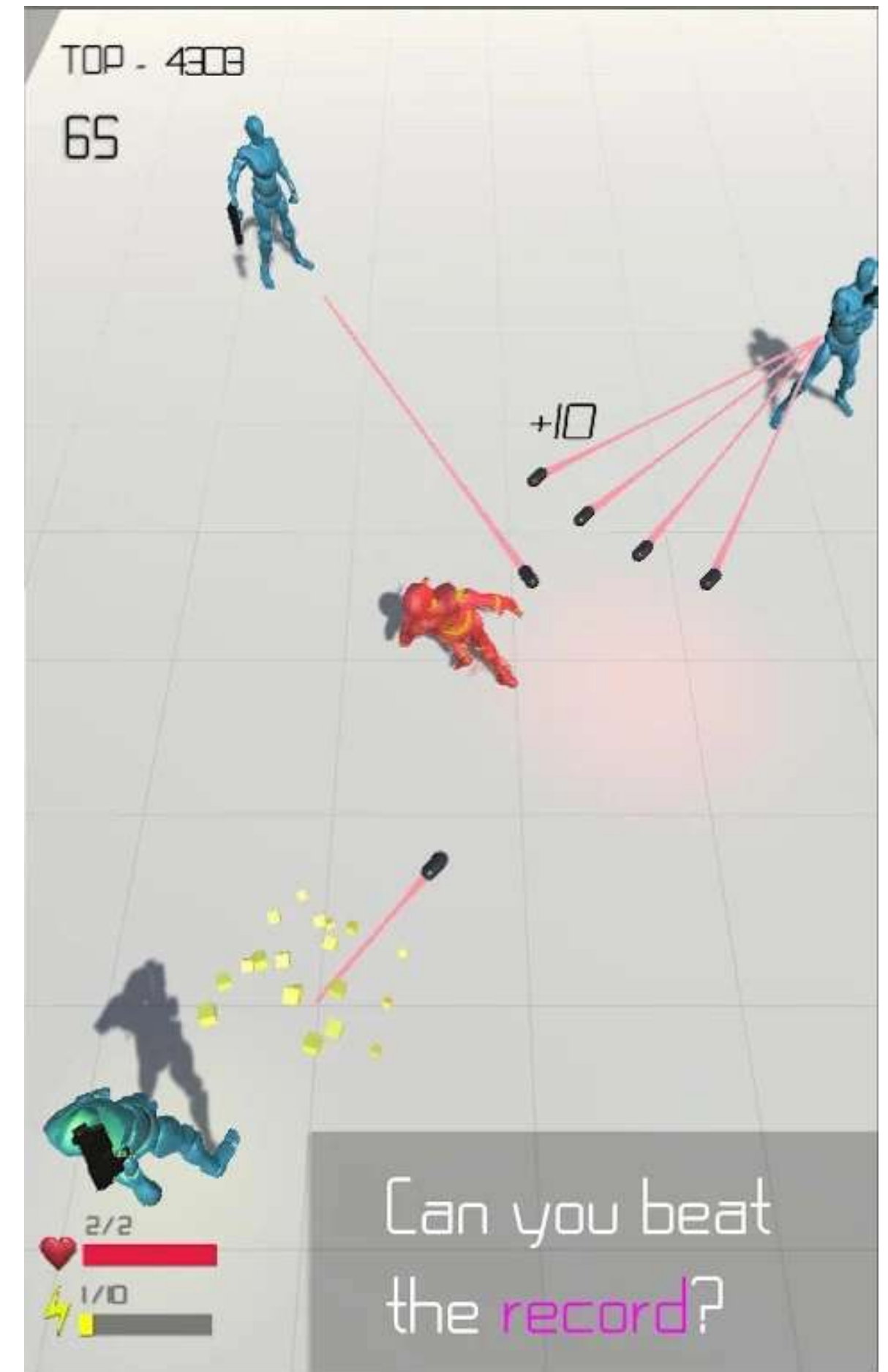
Roles:

- Game Design: Planning and implementation of game concept, characters, level design
- Programming: Implementation of game logic and interactions using Unity and C#
- Testing and Deployment: Bug fixes and optimization work through testing on various devices

Results Over 50,000 downloads, 4.5/5 user rating,
Retention: D1 45%, D7 12%

Github <https://github.com/whd793/speed-hero-prototype>

Demo <https://www.youtube.com/watch?v=tH6Hlx6yb9I>



05

Speedy Hero - Mobile Game

Core Features:

Player Movement and Core Gameplay System Development:

- High-speed movement and collision handling: Smooth and realistic velocity-based motion using Unity's Rigidbody physics
- Character abilities: Dash and evasion mechanics using Rigidbody.AddForce and Raycasting
- Enhanced collision detection during high-speed movement with Raycast (Continuous Collision Detection applied)
- Development of DashController.cs for instant acceleration (Dash) functionality and cooldown implementation
- Dynamic obstacle generation: Procedurally generated obstacles providing a unique experience with each run

AI and Enemy NPC System Development:

- AI Pathfinding and pursuit system: Implementation allowing AI to dynamically chase the player using Unity's NavMesh
- Finite State Machine (FSM) application (Idle, Chase, Attack state transitions)
- NavMeshObstacle addition to avoid collisions with obstacles
- Dynamic Difficulty Adjustment (DDA): Enemy spawn rate adjustment based on player performance

05

Speedy Hero - Mobile Game

Core Features:

Graphics and Performance Optimization:

- Optimized rendering pipeline: Use of Unity's URP (Universal Render Pipeline) for performance improvement
- Custom shader development based on ShaderLab: Motion Blur effect during high-speed movement, trail effect (Afterimage Shader) implementation based on speed
- LOD (Level of Detail) system application: Performance improvement by reducing the detail of distant objects
- Asynchronous scene loading: Smooth scene transition by loading backgrounds and maps asynchronously
- Object Pooling technique: Utilizing a pool manager for object reuse instead of frequent creation/destruction

05

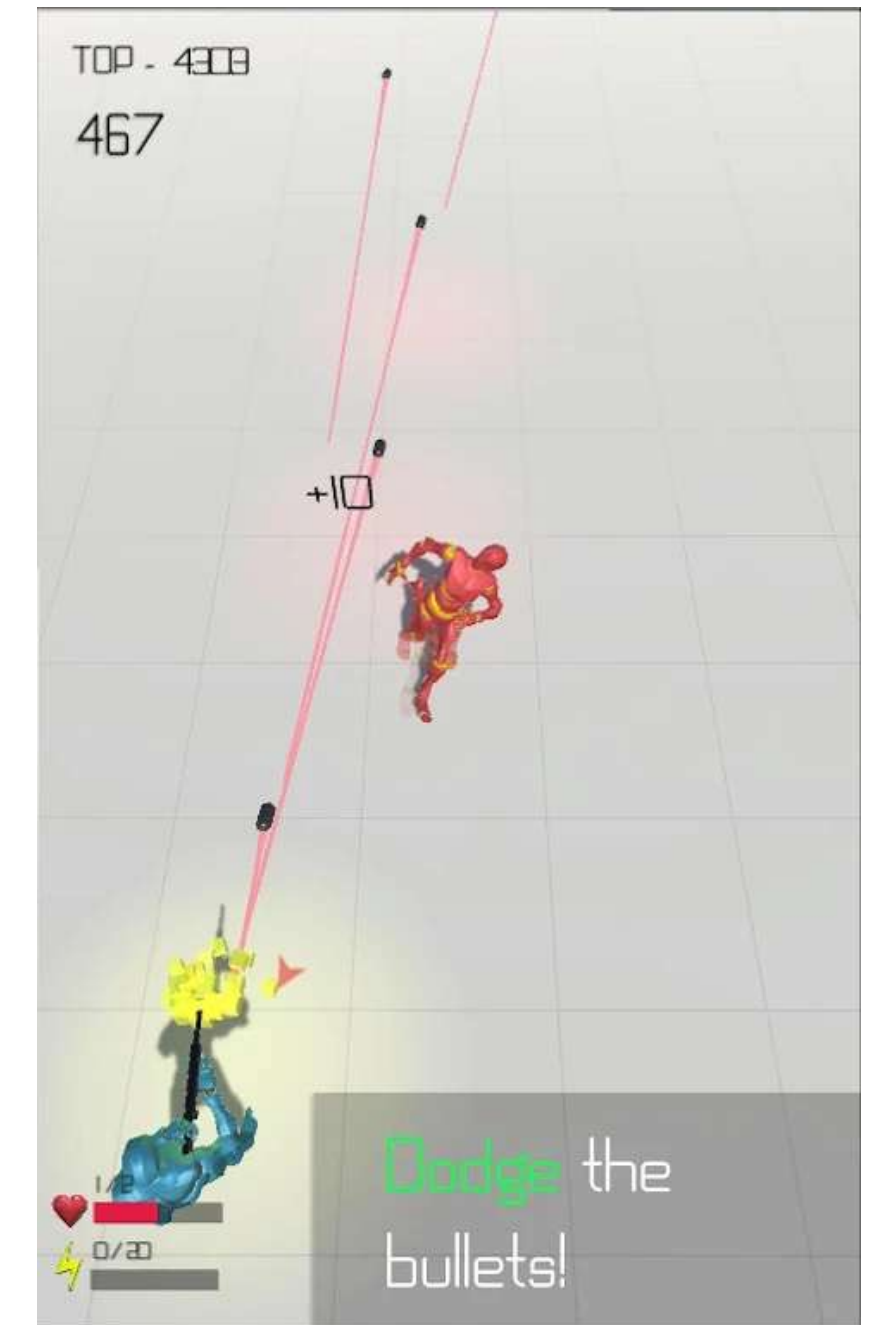
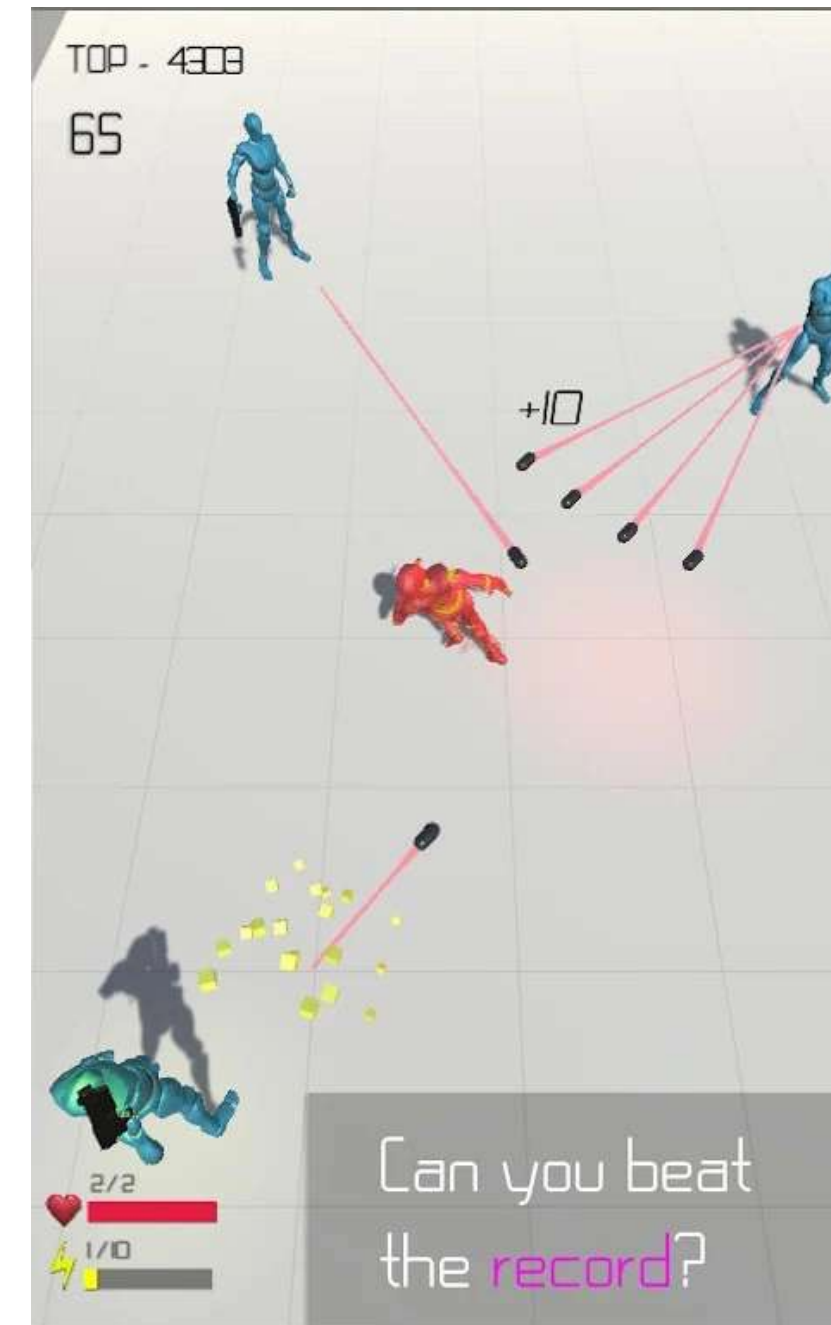
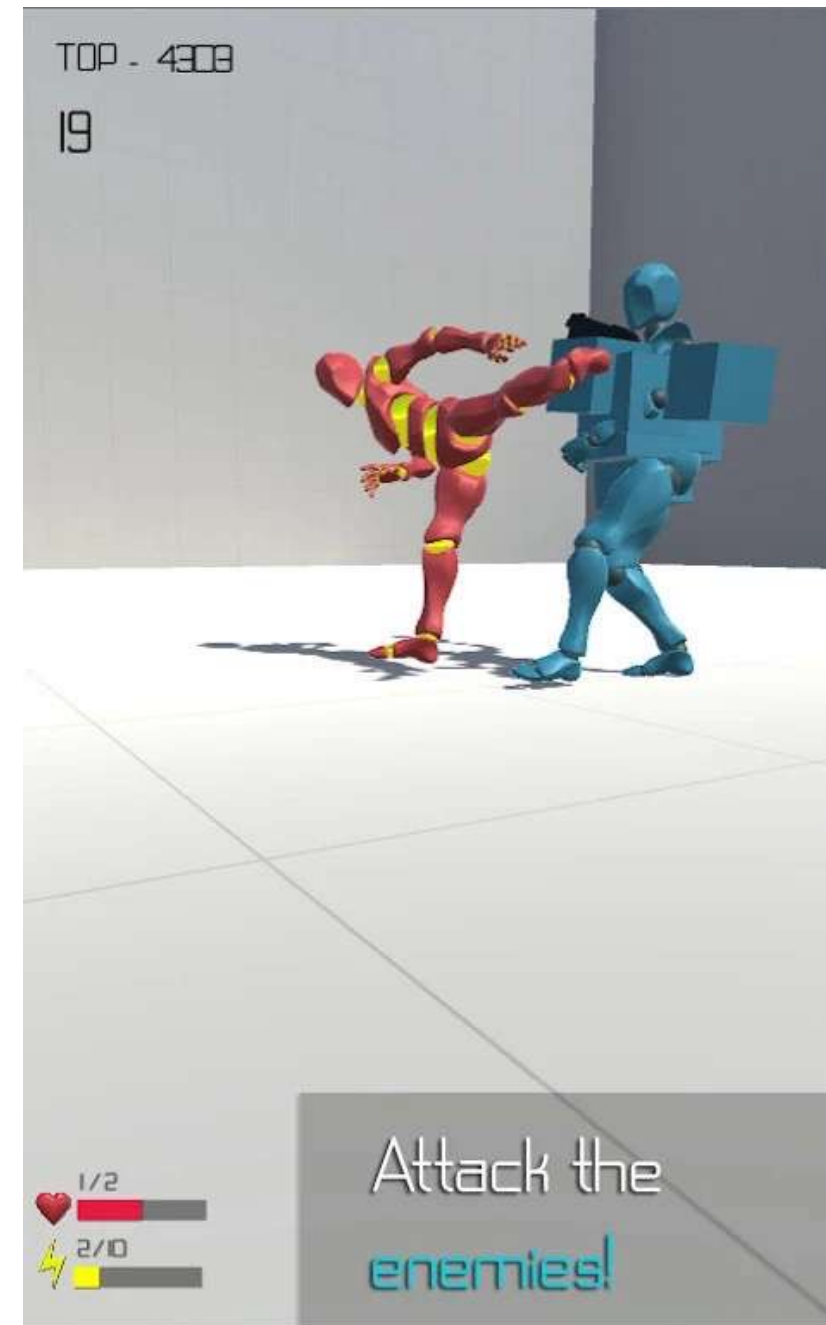
Speedy Hero - Mobile Game

Core Features:

Game Optimization and Functional Problem Solving:

- Quality settings optimization: Most games don't need high settings; in cost-intensive projects, disabled shadows and replaced them with fake circular images below characters
- LOD (Level of Detail) utilization: Scripts to simplify meshes according to camera distance, saving significant resources in terms of performance
- Caching use: Unity component getter functions used only once to fetch references
- Power-of-two textures: Always used power-of-two textures including UI elements (otherwise compression is disabled)
- Profiler use: Identifying performance issues through CPU usage, detecting rendering, physics, or graphics usage problems
- Compression tools: Many draw calls cause CPU and performance issues on Android; reduced the number of draw calls by merging scene meshes

05 Speedy Hero - Mobile Game



06

Hyper Casual Game Project

Intro

Successfully developed 22 hyper casual games using the Unity engine. During the development process, I conducted thorough market research and continuously improved gameplay mechanisms based on user feedback through rapid prototyping and iterative development cycles. I implemented sophisticated systems using C# programming and created modular, reusable code architecture to streamline the development process and maintain high code quality standards.

Through rigorous A/B testing and data analysis using Game Analytics and Facebook Analytics, I gained deep insights into user behavior patterns and monetization strategies. This data-driven approach allowed me to optimize key performance indicators, resulting in impressive retention rates, user session lengths, and user engagement.

This project taught me not only my technical capabilities in game development but also my ability to understand and respond to market demands. Through this experience, my skills in data analysis, user experience design, and business strategy formulation were significantly strengthened.



06 Hyper Casual Game Project

Technologies Used: Unity (C#), Game Analytics, Facebook Analytics

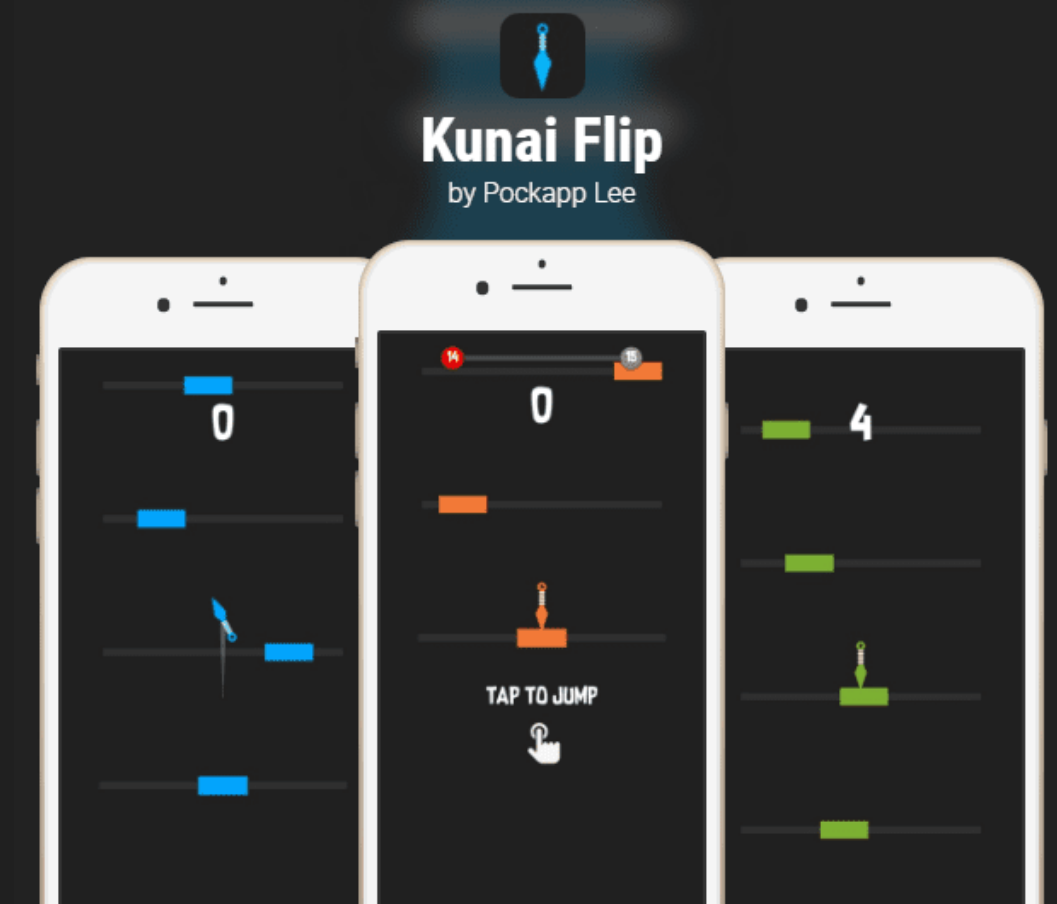
Development Period: 2023.08 ~ 2024.03

Team: PM, Designer, Game Developer

- My Role:
- Game Design: Planning and implementation of game concept, characters, level design
 - Programming: Implementation of game logic and interactions using Unity and C#
 - Testing and Deployment: Bug fixes and optimization work through testing on various devices



How high can you flip

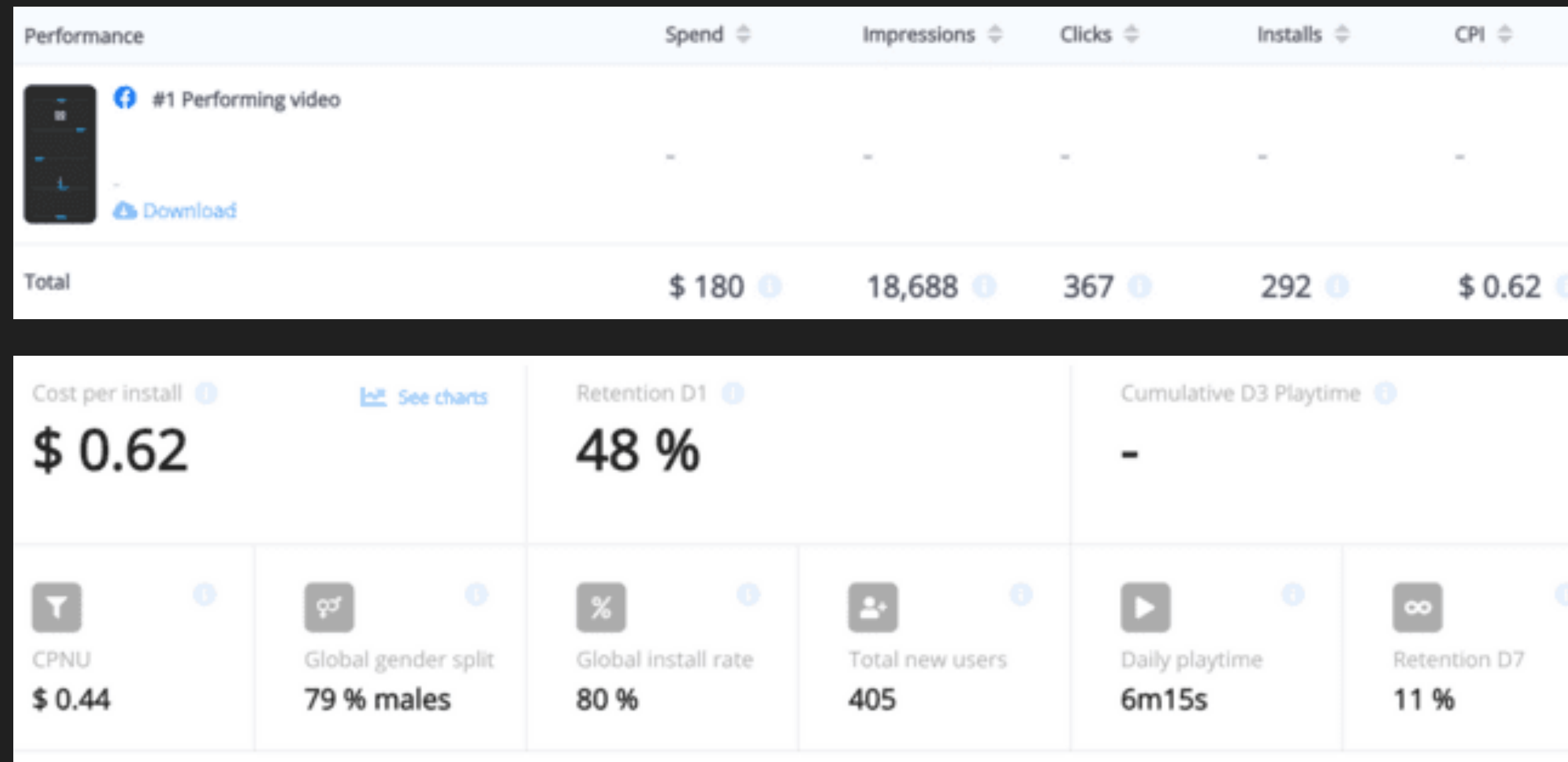


06 Hyper Casual Game Project

Case Study

I built a system to set and track key performance indicators (KPIs) for data-driven decision making. I analyzed user behavior using Game Analytics and Facebook Analytics, and optimized key game elements through A/B testing. As a result, I increased D1 retention to 48%. I also increased average session time to 5 minutes and successfully reduced CPI in the US market to \$0.62 through ad optimization.

Kunai Flip Analytics

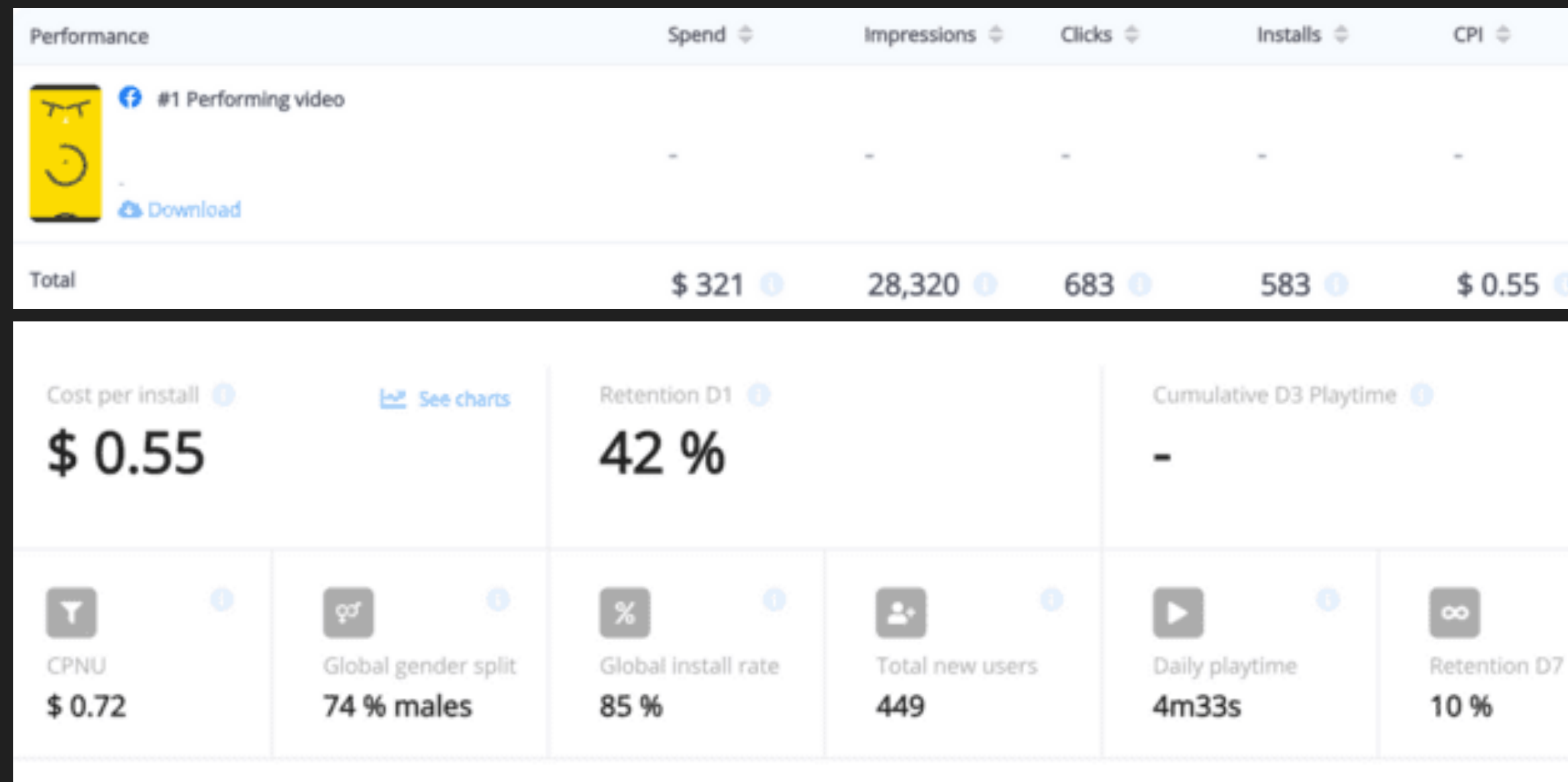


06 Hyper Casual Game Project

Case Study

I established a system for setting and tracking key performance indicators (KPIs) to enable data-driven decision making. Using Game Analytics and Facebook Analytics, I analyzed user behavior and optimized key game elements through A/B testing. As a result, I was able to increase D1 retention to 42%. Additionally, I increased average session time to 5 minutes and successfully reduced CPI in the US market to \$0.55 through ad optimization.

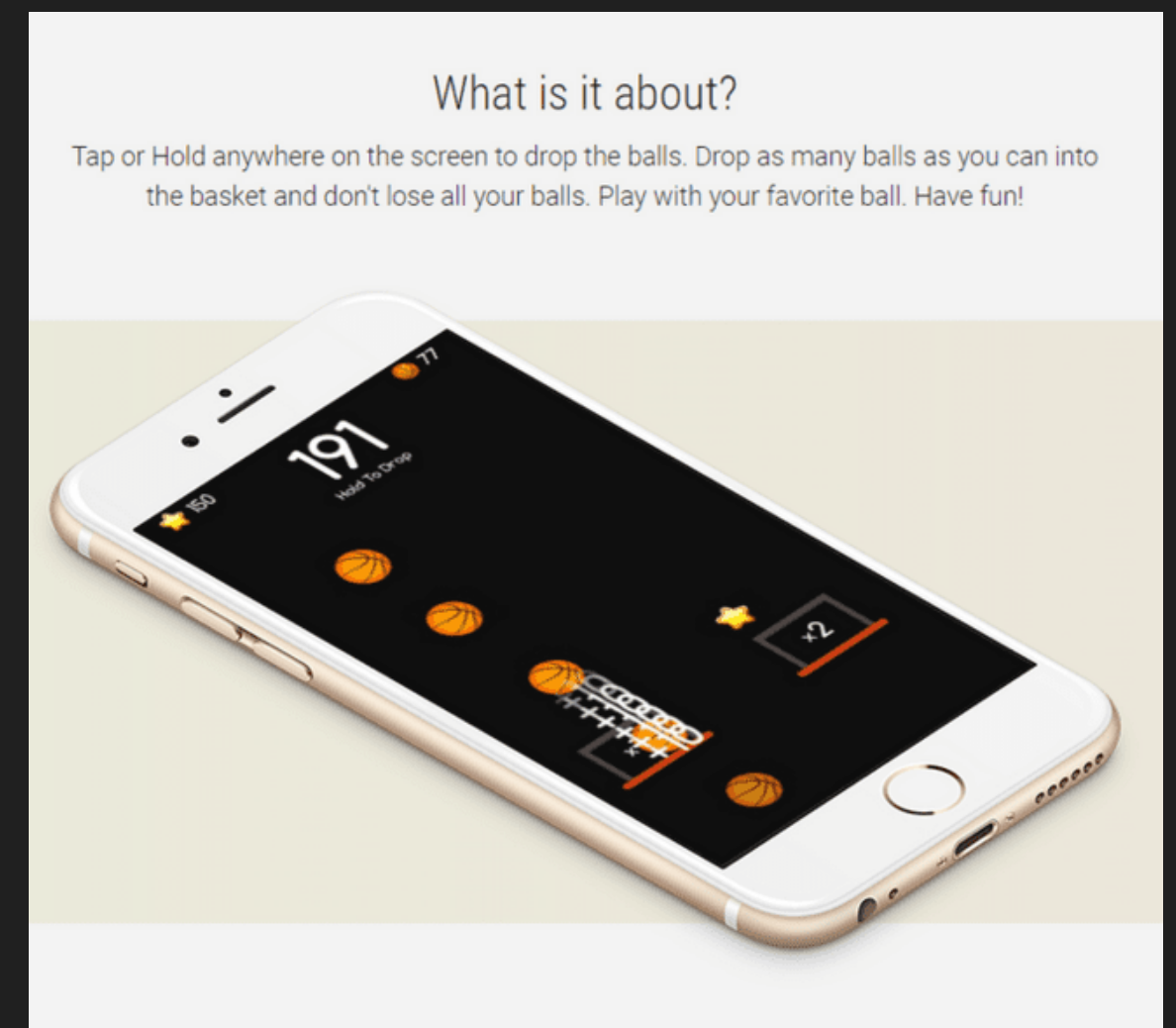
Dashy Arrow Analytics



06 Hyper Casual Game Project

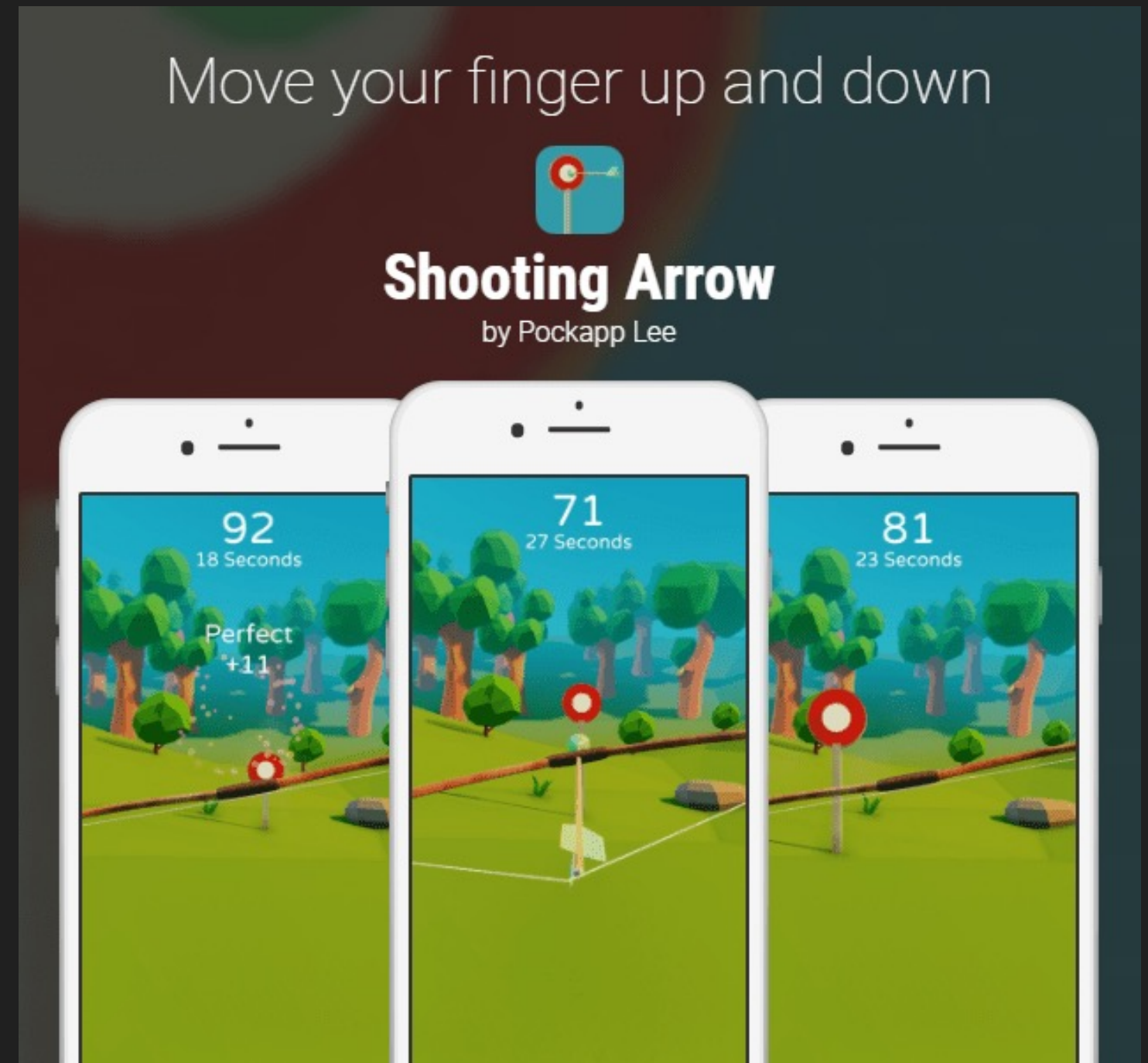
Game Optimization and Functional Problem Solving:

- **Quality Settings Optimization:**
 - Most games don't need high settings; in cost-intensive projects, disabled shadows and replaced with fake circular images under characters
- **LOD (Level of Detail) Utilization:**
 - Scripts to simplify meshes based on camera distance, saving substantial resources in terms of performance
- **Caching Usage:**
 - Unity component getter functions used only once to retrieve references
- **Power-of-Two Textures:**
 - Always used power-of-two textures including UI elements (otherwise compression is disabled)
 - Examples: Small images - 128x128, 256x256, Medium images - 512x512, 1024x1024
 - Image resolution determined proportionally to screen area occupied
 - Non-power-of-two images typically burden device memory 4 times more

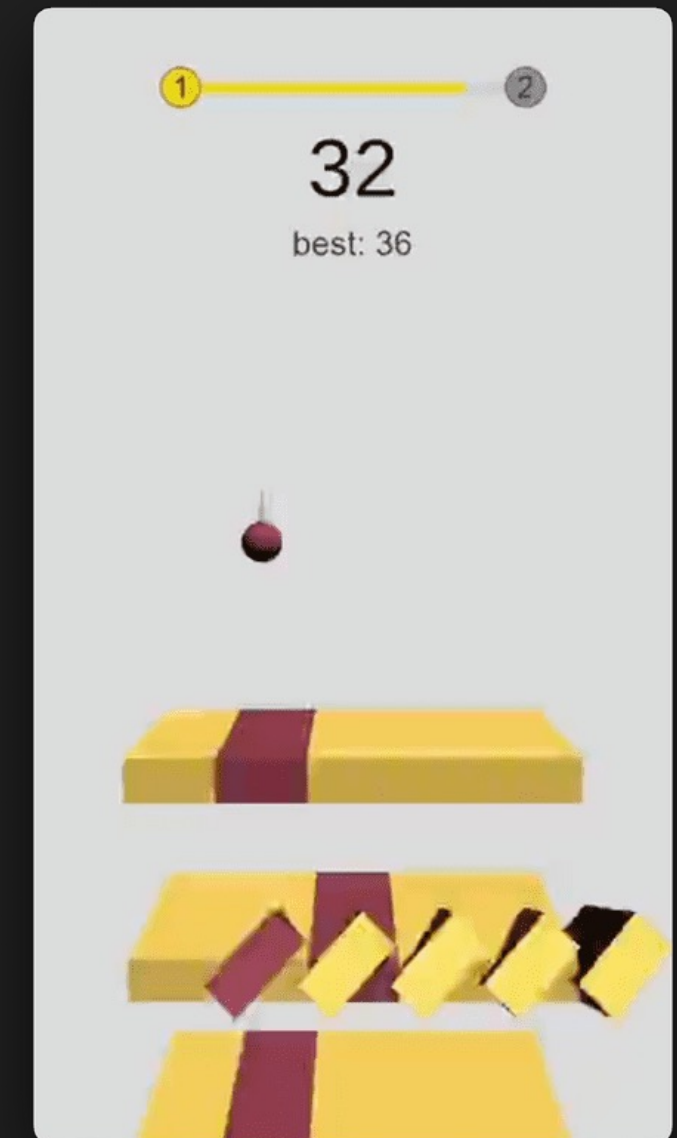
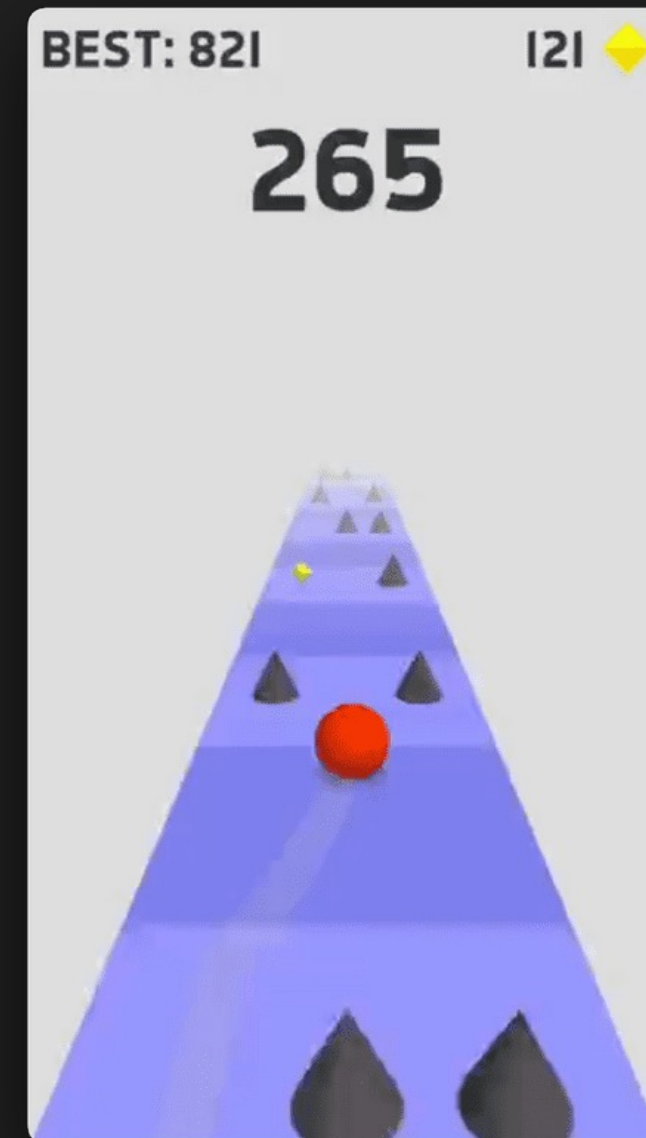
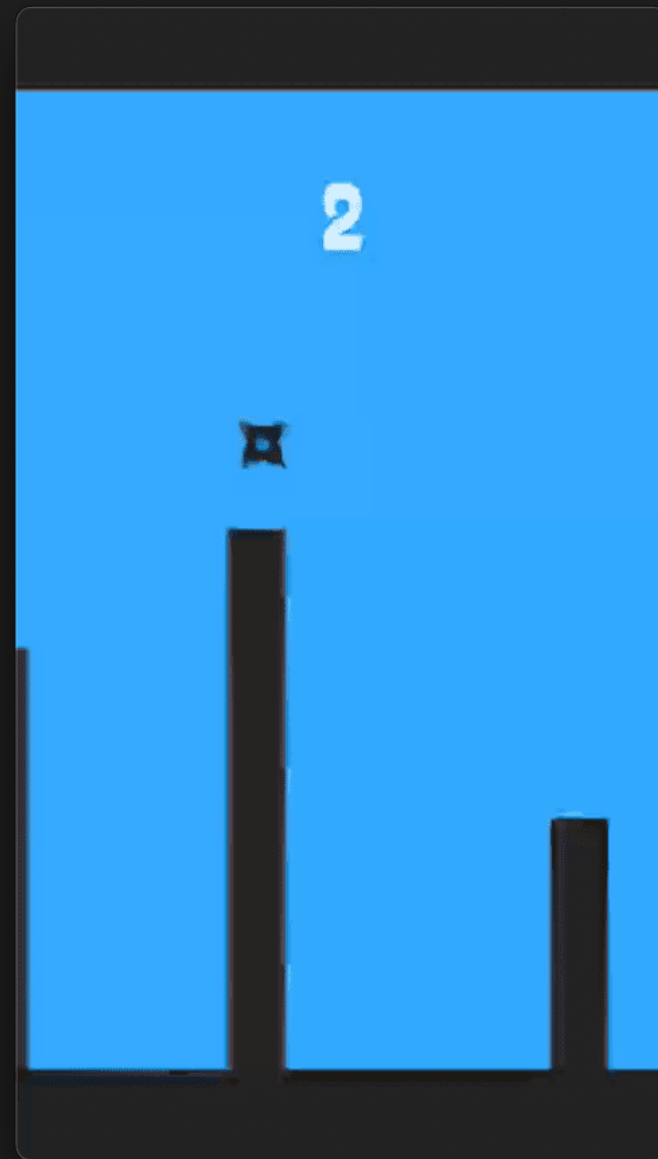
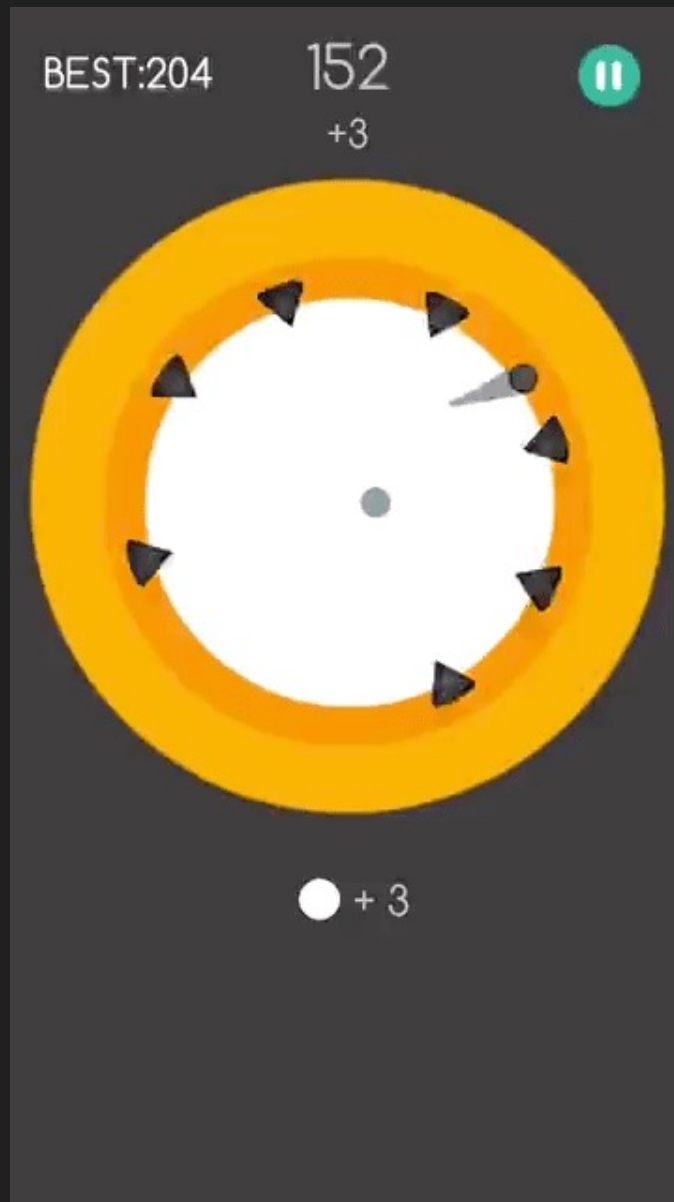


06 Hyper Casual Game Project

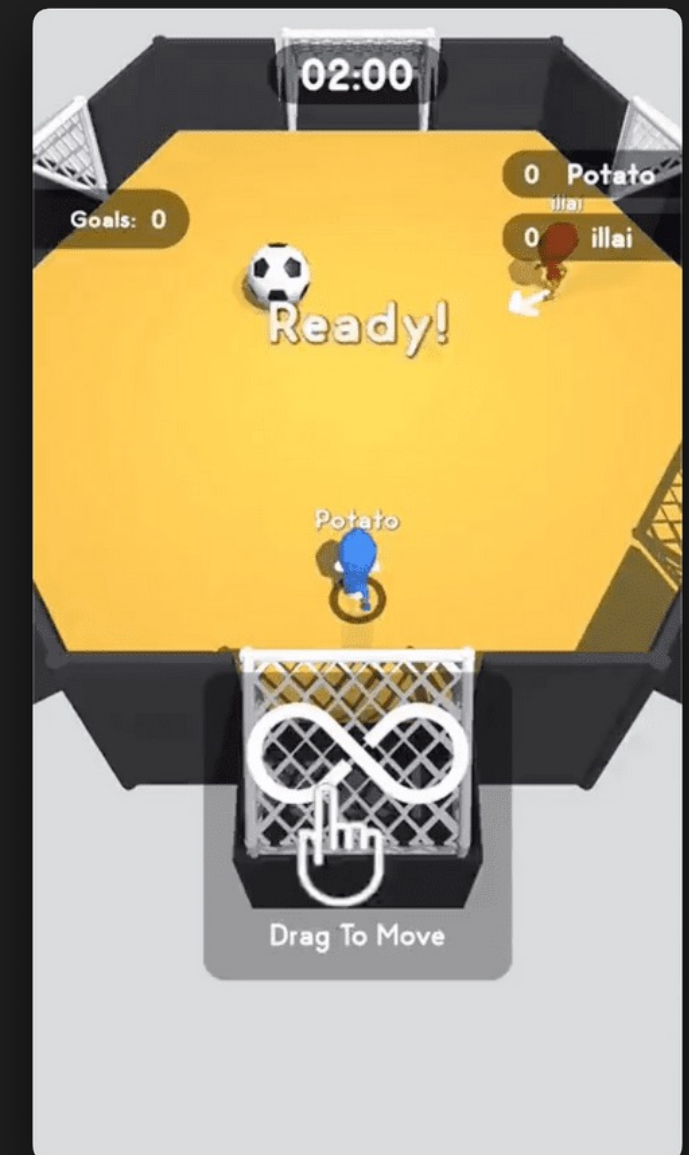
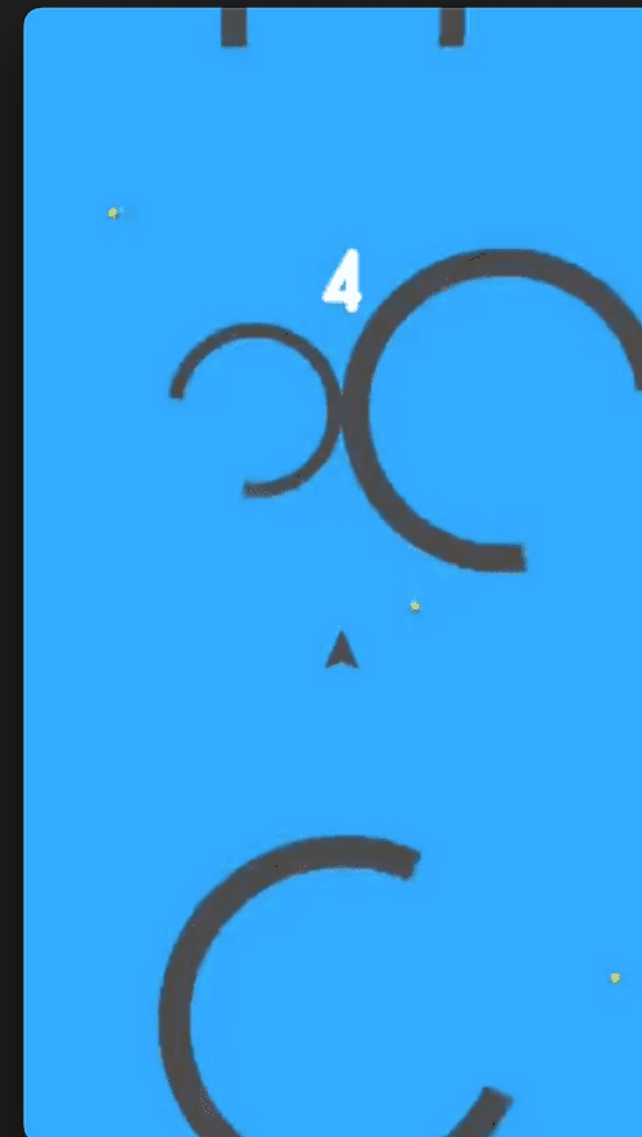
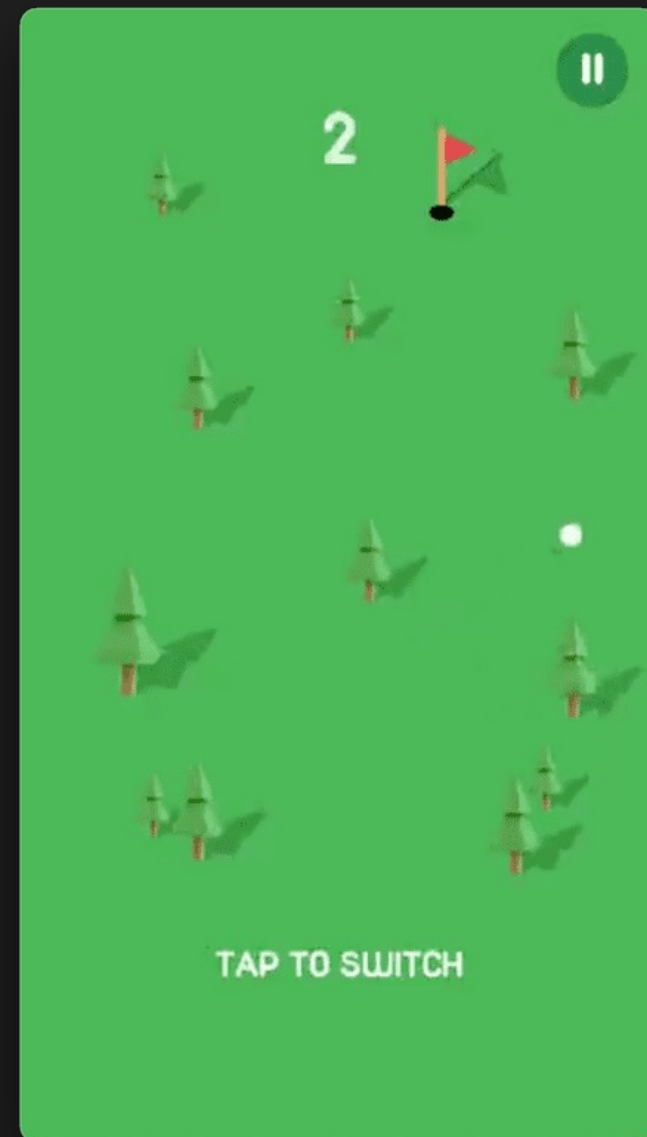
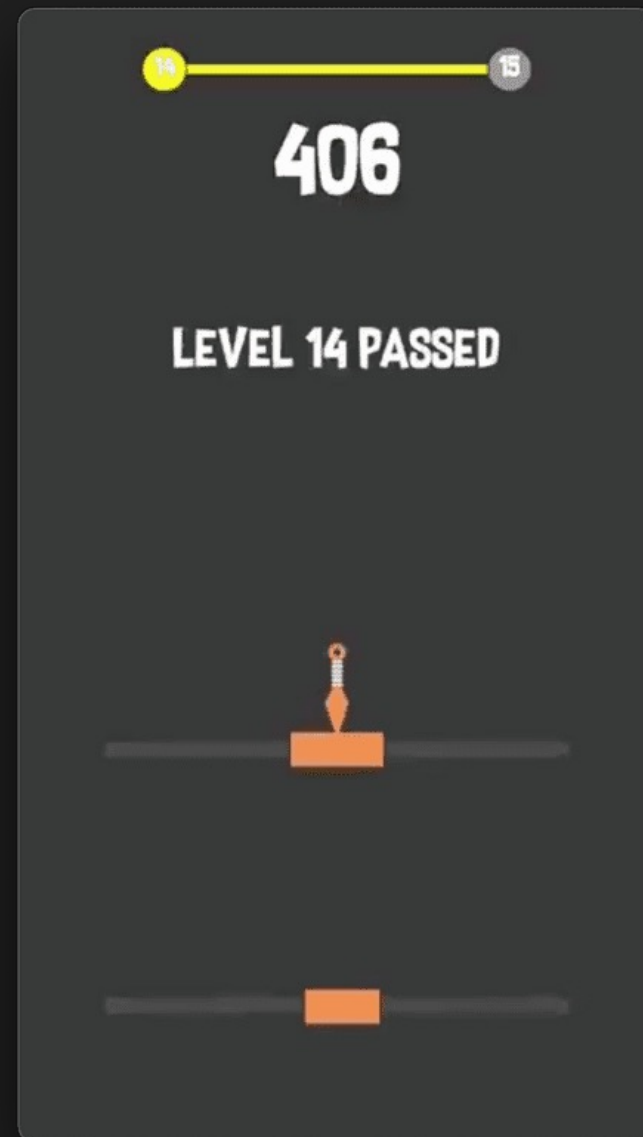
- **Android Haptic Optimization:**
 - Many Android devices overreact to haptics causing heating or lag;
 - used only light haptics on Android, avoided continuous haptic use
- **Compression Tools:**
 - Many draw calls cause CPU and performance issues on Android;
 - reduced draw call count by merging scene meshes



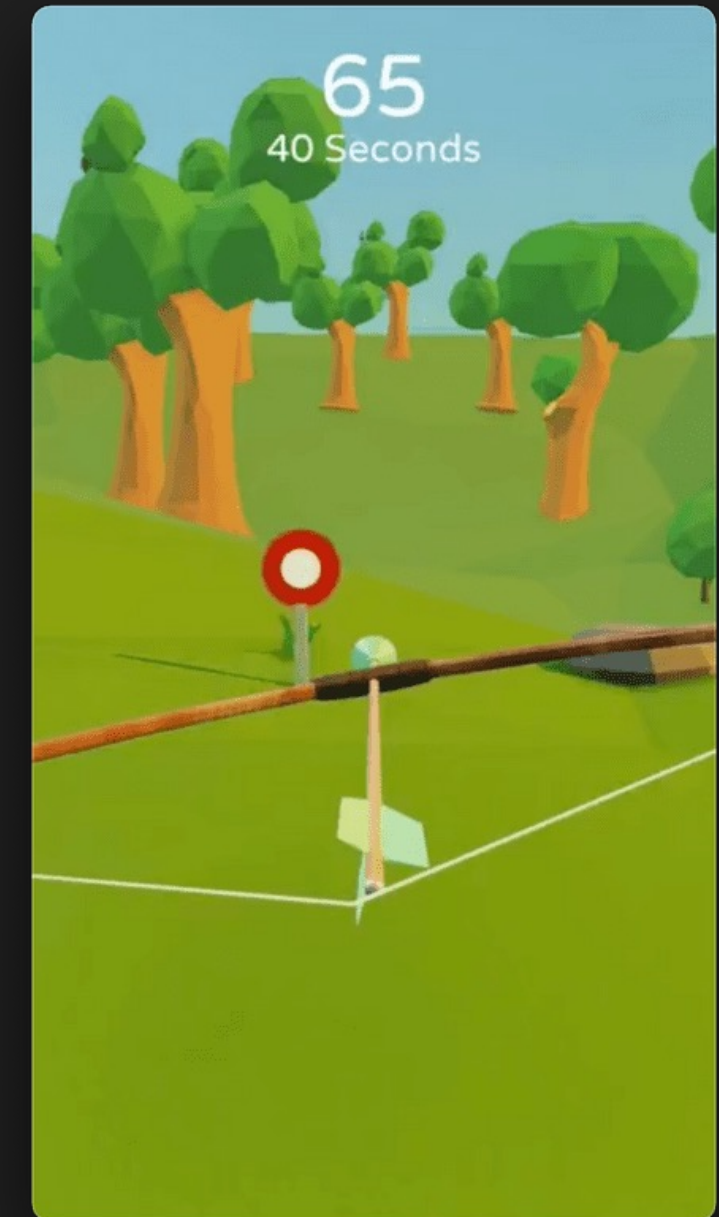
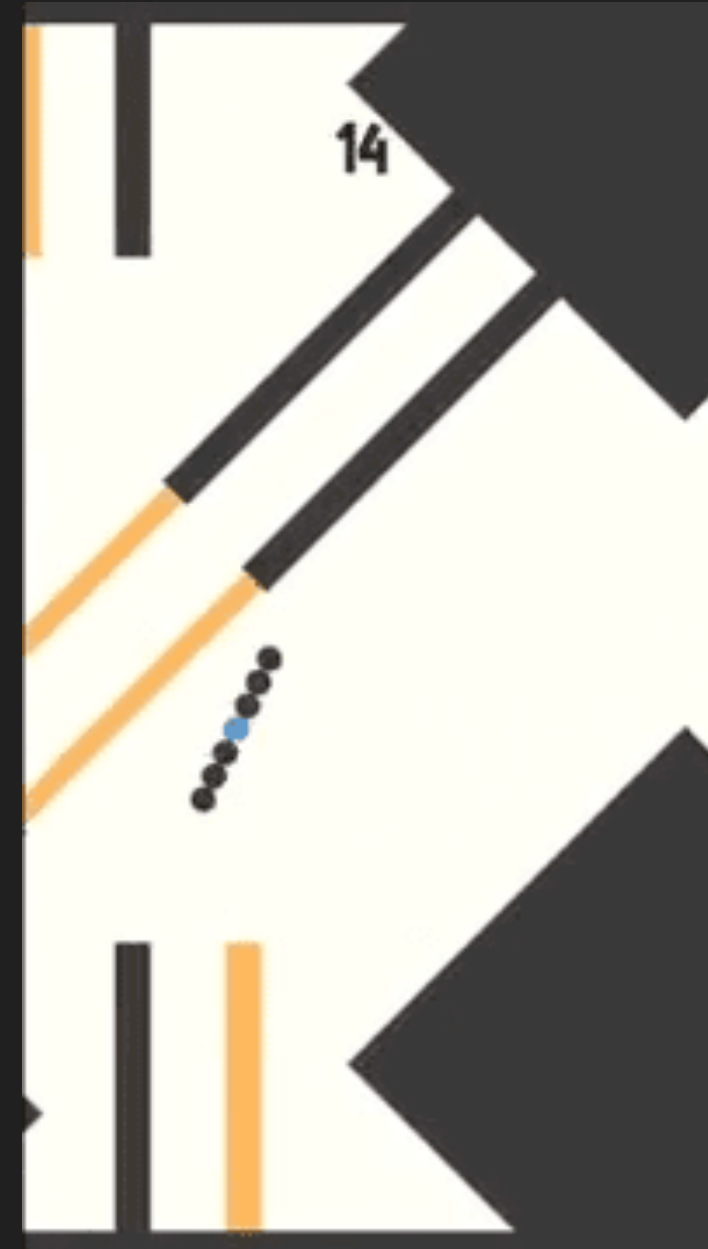
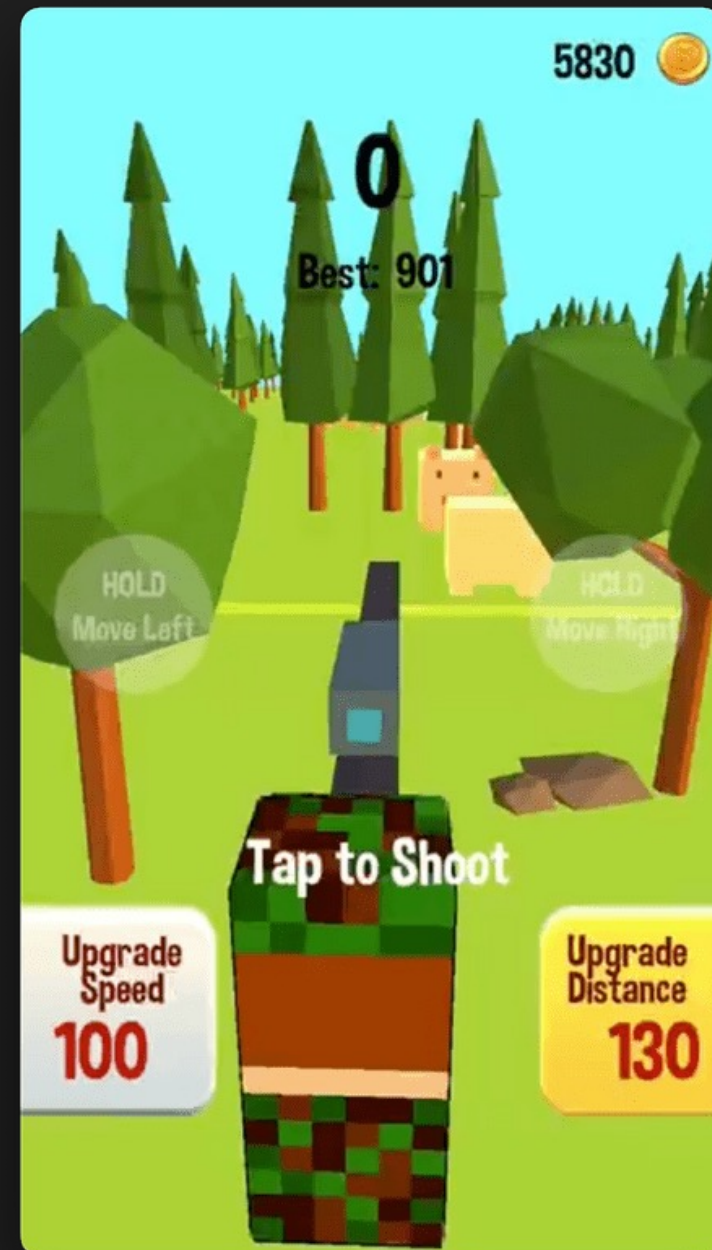
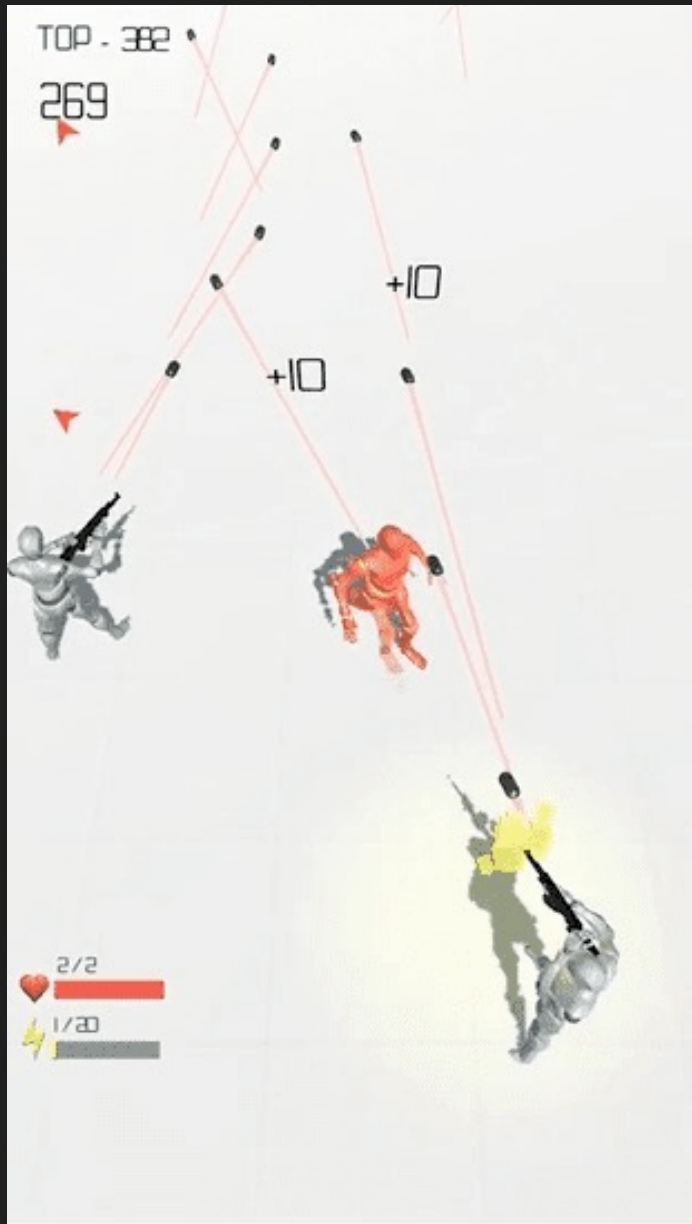
06 Hyper Casual Game Project



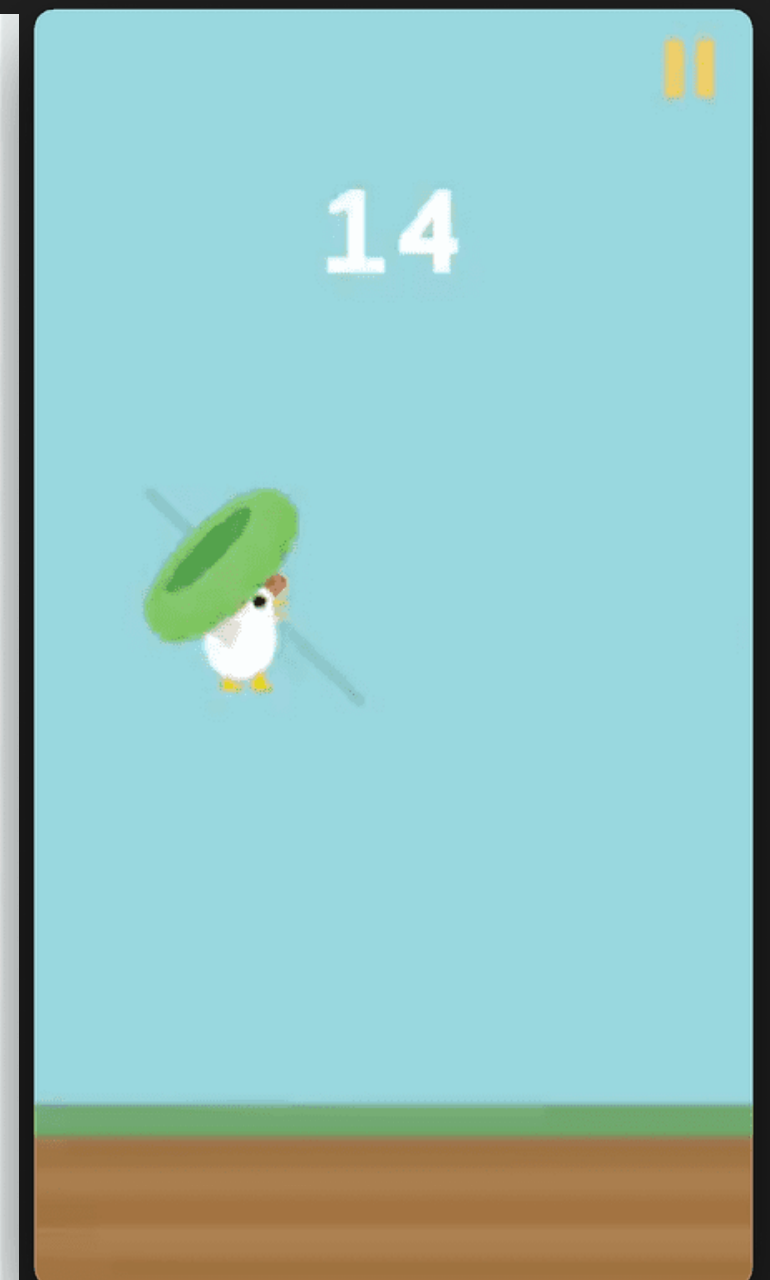
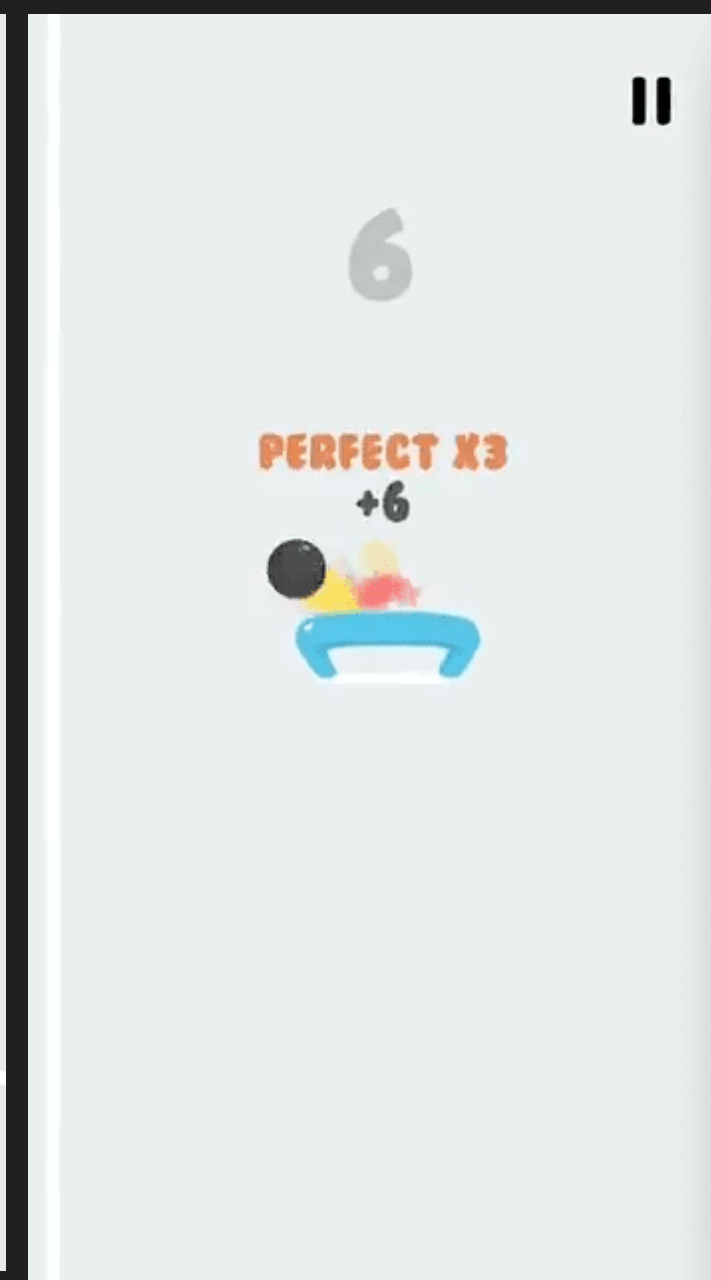
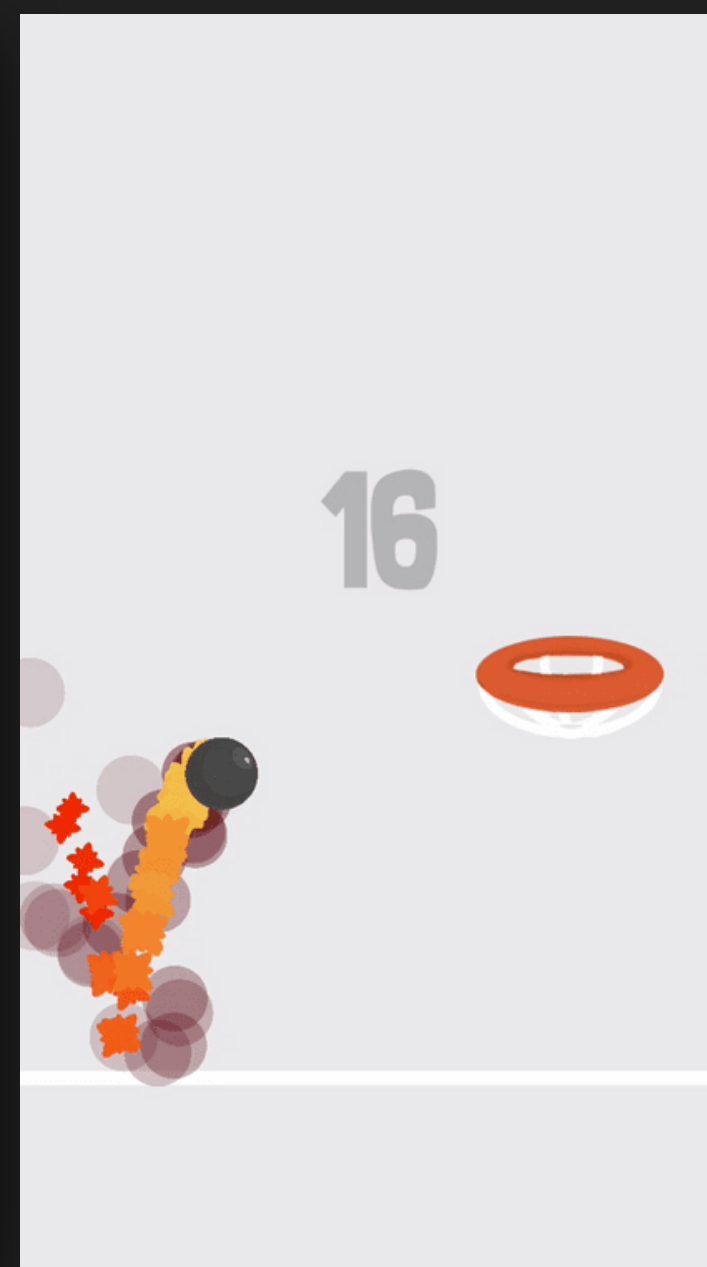
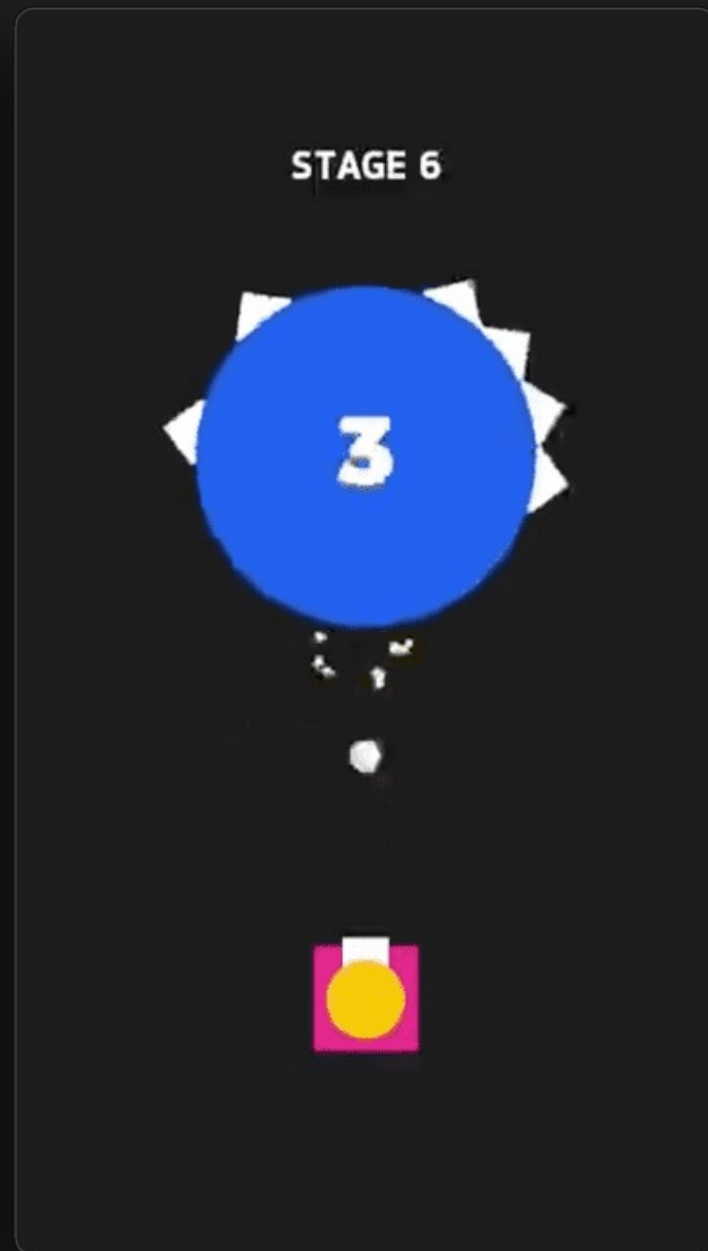
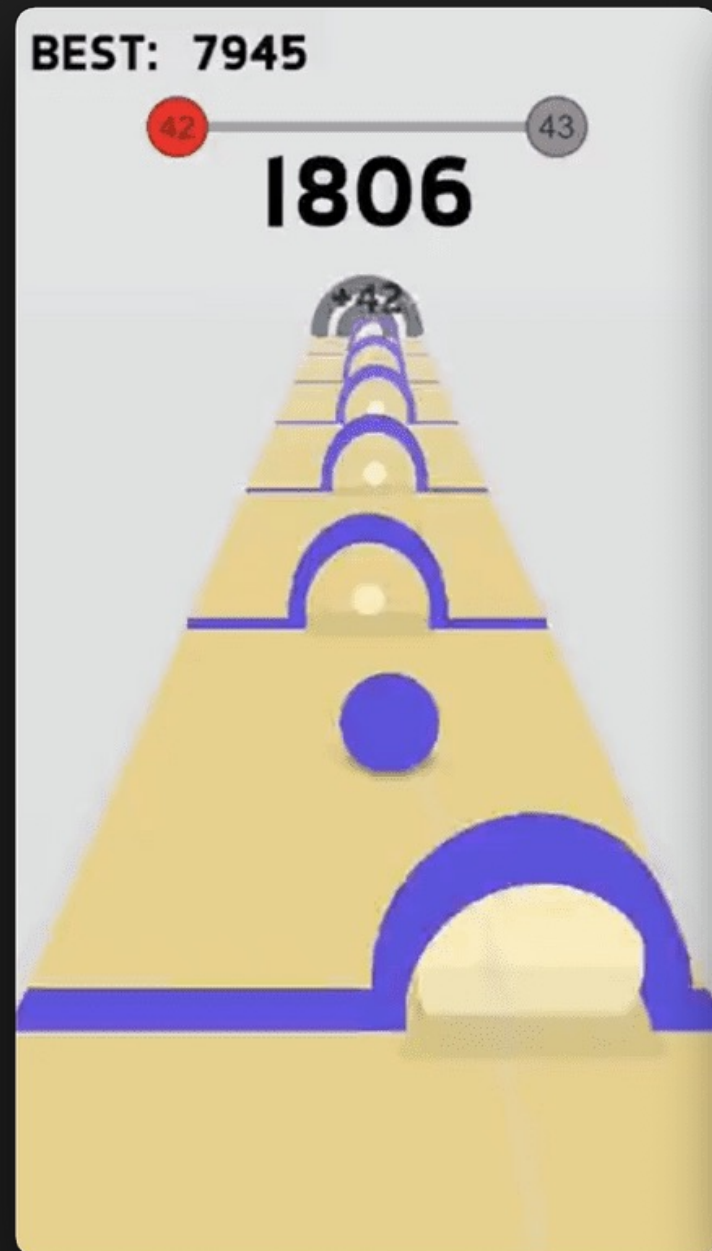
06 Hyper Casual Game Project



06 Hyper Casual Game Project



06 Hyper Casual Game Project



THANK YOU

To view more projects, please visit my portfolio website: whd793.github.io

Portfolio Website:	whd793.github.io
Resume pdf (Korean):	whd793.github.io/resume-kor.pdf
Resume pdf (English):	whd793.github.io/resume-eng.pdf
Github:	github.com/whd793
Blog:	https://wonblog.vercel.app/
Linkedin:	llinkedin.com/in/whd793
Email:	whd793@gmail.com
Phone:	+82-10-2551-5567