HomeQuest

Hyeonseo Shim
Department of Information Systems
Hanyang University
Seoul, Republic of Korea
hyeonseo321@gmail.com

Sihyun Jang
Department of Information Systems
Hanyang University
Seoul, Republic of Korea
jjj99nine2@hanyang.ac.kr

Jin Yun
Department of Information Systems
Hanyang University
Seoul, Republic of Korea
jinijini0402@hanyang.ac.kr

Yeonsoo Cho
Department of Public Administration
Hanyang University
Seoul, Republic of Korea
meriel1010@hanyang.ac.kr

Abstract—This study proposes a family-centered smart home service that transforms everyday appliance and wearable data into shared motivation and emotional connection. The system aggregates data from LG ThinQ devices, analyzes behavior patterns, and generates daily or weekly AI briefings that present both family progress and personal achievements. Through the Family Challenge System, members receive action-based goals in areas such as household chores, energy saving, and wellness. Upon achieving these goals, individual points are accumulated, and each ongoing family challenge is visualized through a character interface. Members can earn cumulative points by completing challenges and view family rankings based on their performance. This process turns raw data into a game-like and visually engaging experience that fosters participation and achievement within the household, extending smart homes beyond efficiency to become a data-driven yet emotionally meaningful space for family connection.

TABLE I. *Role Assignments*

Role	Name	Task Description
Backend Developer	Yeonsoo Cho	The Backend Developer builds and manages server-side architecture, databases, and APIs that support the application's core features. This includes designing efficient data models, implementing security and authentication, and ensuring reliable data exchange between systems. The developer enhances performance, scalability, and fault tolerance while maintaining documentation and monitoring. Through collaboration with frontend and AI teams, the backend developer ensures stable and efficient platform operations.
AI Developer	Hyeonseo Shim	The AI Developer designs and implements intelligent systems that enhance the app's personalization and decision-making capabilities. This involves collecting and preprocessing data, training and fine-tuning machine learning models, and deploying them through scalable APIs or pipelines. The developer evaluates model performance, iterates for improvement, and ensures seamless integration with backend and frontend systems. By combining data science and engineering practices, the AI Developer enables the application to deliver smarter, more adaptive user experiences and insights based on real-time data.
Frontend Developer	Sihyun Jang	The UI/UX Designer is responsible for understanding user needs and transforming them into intuitive, user-centered designs. This role involves conducting user research, creating wireframes and interactive prototypes, and developing a consistent design system that aligns with the app's identity. The designer ensures visual harmony, accessibility, and usability across all interfaces while collaborating closely with developers to maintain design accuracy during implementation. By continuously refining layouts and interaction flows, the designer contributes to creating a seamless and engaging overall user experience.
Frontend Developer	Jin Yun	The Frontend Developer builds responsive, interactive, and visually precise user interfaces from design specifications. This includes creating components with modern frameworks, ensuring cross-device compatibility, and integrating backend APIs. The developer improves performance and accessibility, conducts testing and debugging, and maintains clean code. Collaborating with designers and backend teams, the frontend developer delivers a smooth and cohesive user experience from design to deployment.

I. INTRODUCTION

A. Background

1) Smart homes have become deeply integrated into our daily lives, offering convenience and automation in many ways. However, most smart home data only shows the results of what has already happened through numbers and notifications. While users can review their activity outcomes, the data rarely helps them recognize or improve their living patterns. In fact, as smart homes provide more information, the gap between data and meaningful behavioral change continues to grow. We wanted to close this gap.

2) This realization naturally led us to focus on the idea of home as a shared space. Daily life is not created by individuals alone, but through the interactions and relationships among family members. Yet current smart home systems

manage data on an individual basis, failing to reflect this interconnectedness. We believe smart homes should evolve into systems that allow families to recognize each other's actions and grow together through shared feedback.

3) From this perspective, we propose Familylog. This service analyzes data collected from LG ThinQ appliances and wearable devices to generate personalized reports and practical challenges that fit each family's lifestyle. Small actions such as finishing the laundry, taking a walk together, or saving energy are recorded and connected, gradually becoming meaningful data-driven experiences in the family's daily life. Beyond simply improving efficiency, our goal is to present a new direction for smart homes where data inspires action and action strengthens connection.

B. Problem Statement (Client's Needs)

(a) Lack of Family Connection and Behavioral Change
Current smart home systems are primarily operated through individual user accounts, managing each member's data separately. As a result, even though home appliances are shared devices used by the entire family, the collected data remains divided by individual and fails to connect the activities of different members. This separation makes it difficult to understand collective household patterns or identify shared goals, preventing the data from contributing to meaningful habit improvement or behavioral change. While smart homes effectively enhance personal efficiency and convenience, they show clear limitations in promoting family interaction and collaborative experiences.

(b) Need for Meaningful Data Experience
Current smart home systems, powered by IoT and AI technologies, collect and provide large amounts of data. However, users receive this information repeatedly without meaningful context. Notifications about laundry completion, energy consumption, or step counts accumulate over time, leading to information overload and increasing user fatigue. This constant flow of data diminishes interest and engagement, causing users to perceive the smart home merely as an automated tool rather than a supportive companion.

(c) Lack of Emotional Connection
The current smart home data environment focuses mainly on storing and displaying information, failing to capture the emotional flow or meaningful changes in daily life. In this fragmented structure, even when family members experience behavioral or lifestyle changes, the emotional meaning behind these actions is not shared, weakening the overall context of family life. While smart homes emphasize functional convenience, they lack mechanisms that foster emotional interaction and connection among family members. As a result, positive changes often fade unnoticed, and the family's growth and shared experiences remain buried within disconnected data.

C. Common Concepts and Object Definition

- Family: A container that groups family members. Each family entity maintains its own set of policies, including reward and penalty rules, cutoff times, and notification preferences.

- Member: An individual family participant who can be linked to one or more LG devices or wearables.

- Goal: A quantitative target defined at either the individual or family level (e.g., "Do laundry twice a week," "Reduce family energy use by 10%"). Each goal includes parameters such as duration (daily, weekly, or monthly), baseline values, measurement units, success criteria, and weighting or contribution rules.

- Challenge: An executable package composed of one or more goals. Challenges can be individual- or family joint-, and include details such as start and end dates, rewards and penalties, and verification methods.

- Metric: A standardized measurement unit used to quantify performance or behavior. Examples include laundry cycles (count), energy consumption (kWh), step count (steps), and cleaning area (m²).

- Event: A single recorded action triggered by a device, app, or voice command—for instance, completing a laundry cycle, starting or ending a robot vacuum session, pressing a confirmation button, or saying "I'm done."

D. Research on Related Software and Algorithm

(a) Samsung SmartThings Energy [1]
1) Function: Samsung SmartThings Energy automatically collects and analyzes power consumption data from home appliances to help users manage energy usage efficiently. Within the app, weekly and daily consumption patterns and reduction rates are visualized, accompanied by notifications such as "Power usage decreased by 8% this week" and energy-saving tips. In addition, the system includes functions that automatically set energy-saving goals and provide personalized recommendations for more efficient usage.

2) Similarity with Our Service: Similar to our service, SmartThings Energy analyzes lifestyle patterns based on appliance data and presents them in report form. Both services share the same objective of promoting efficient household behavior through data-driven and personalized feedback mechanisms.

3) Difference from Our Service: While SmartThings Energy focuses primarily on energy efficiency and power consumption management, our service integrates data related to appliances, health, emotions, and lifestyle habits to provide a comprehensive AI-generated family lifestyle report. In other words, it goes beyond a single-domain approach (energy management) to evolve into a holistic AI platform that offers family-oriented lifestyle improvement and emotional feedback, establishing a clear point of differentiation.

(b) Strive (Group Fitness Challenge App) [2]

1) Function: Strive is a social fitness challenge application that allows users to set exercise goals and participate in team-based competitions or collaborations with family members or friends. Users can set objectives such as step counts or workout durations and monitor their progress through a real-time leaderboard. Upon completing missions, users earn badges and titles, and family members can encourage one another through motivational notifications. Additionally, the app enables continuous performance evaluation by comparing individual results with team averages.

2) Similarity with Our Service: Like the Family Challenge System in our service, Strive adopts a team-based goal structure and promotes participation through both competitive and cooperative mechanics. Both platforms use behavioral data to generate visual feedback and employ gamification strategies that enhance engagement and motivation.

3) Difference from Our Service: Strive is limited to fitness-related data derived from wearable devices, whereas our service expands the scope by combining appliance, health, and lifestyle data to support challenges across various daily activities such as cleaning and energy saving. Thus, instead of functioning merely as a fitness-oriented competition platform, our system aims to build an intelligent challenge ecosystem that integrates family communication and overall lifestyle improvement within the household context.

(c) FamilyAlbum (by Mixi, Japan) [3]

1) Function: FamilyAlbum is a private family-oriented social networking platform that allows members to share photos and videos and interact through comments and emojis. The AI system automatically categorizes uploaded content by date and generates features such as "1-second highlight videos" or "On this day last year," helping families relive shared memories. Furthermore, its feed-based interface facilitates easy intergenerational

communication and emotional connection.

2) Similarity with Our Service: Our service also employs a family-centered feed structure and AI-based summarization functions to promote communication and emotional interaction among members. Like FamilyAlbum, it visualizes daily records in an easily accessible and aesthetically organized format, offering a similar user experience focused on familial engagement.

3) Difference from Our Service: While FamilyAlbum primarily emphasizes emotional sharing through photo and video records, our service goes a step further by transforming AI-collected lifestyle data into narrative-driven visual stories within the family feed. This distinction positions our service not merely as a record-oriented platform but as an AI-driven family relationship enhancement system, combining emotional exchange with data insights to create a new form of intelligent family communication platform.

(d) Reinforcement Learning [4]

1) Function: Reinforcement Learning (RL) is a machine learning paradigm in which an agent learns to make optimal decisions through continuous interaction with its environment. By receiving rewards or penalties in response to its actions, the agent gradually develops a policy that maximizes cumulative long-term rewards. Through repeated trial-and-error processes, the system is able to discover effective strategies even without prior knowledge of the environment's structure.

2) Application to our Service: In the proposed family challenge service, RL can be applied to the challenge recommendation and difficulty adjustment algorithm. The system observes user engagement metrics—such as completion rates, satisfaction feedback, and participation frequency—and interprets them as reward signals. A successfully completed and positively rated challenge yields a high reward, whereas prematurely abandoned or negatively rated challenges result in low rewards. Based on these reward patterns, the RL agent updates its policy to recommend future challenges that are more likely to match the family's interests and ability levels.

Furthermore, RL enables dynamic difficulty adjustment, where overly easy or overly difficult challenges are penalized, while moderately challenging and successfully completed ones are rewarded. This allows the system to automatically tailor the level of difficulty to each family's performance patterns.

3) Expected Outcomes: By integrating reinforcement learning, the service can (1) continuously personalize challenges to fit each family's behavioral tendencies, (2) increase long-term engagement and motivation, and (3) evolve into a self-adaptive recommendation system that improves its performance as more interaction data is accumulated.

II. REQUIREMENTS

A. User Authentication

a) Initial Screen: Upon startup, the application initializes with the login screen as the default interface. This screen enables returning users to access their accounts and provides options for new user registration or credential recovery. The interface includes input fields for user ID and password, with the password field masked to ensure privacy. Prior to server transmission, client-side validation checks are applied, including ID format verification and minimum password requirements. All login data are transmitted securely, and based on the server response, users are either redirected to the main dashboard or presented with a corresponding error message.

b) User Input: After the login screen is displayed, the user must provide their account credentials. The ID field requires a unique identifier, often in the form of an email address, while the password field accepts a private passphrase that remains hidden as it is typed. Before the credentials are submitted, the system performs simple checks such as verifying email structure or confirming that the password meets minimum length and complexity rules.

c) Validation and Authentication: Once the credentials are entered, the system validates them against stored records. The password is first converted into a secure hash so that the original value is never transmitted or stored. If the account ID exists and the hash matches the one saved in the database, the login is successful and access to the main interface is granted. If the password does not match, the user is notified and prompted to try again, while a non-existent ID results in a clear message indicating that the account cannot be found.

B. Data Collection from Devices

The system integrates and manages data collected from various IoT appliances and wearable devices. Each dataset serves as the foundation for analyzing user lifestyle patterns and generating behavioral indicators at the family level.

(a) Household Tasks Data Collection
The system automatically collects household activity data from multiple LG ThinQ appliances. All task-related events—such as washing, drying, dishwashing,

and cleaning—are automatically logged, transmitted to the backend, normalized, and stored.

- Washing Machine: Each completed washing cycle is recorded as a single household task event. Cycle start and end times are captured to track overall laundry frequency.

- Dryer: Every completed drying cycle is logged in the same manner, allowing monitoring of laundry-related activity frequency.

- Dishwasher: Each dishwashing operation is registered as a completed task. These records are used to measure participation and contribution in kitchen-related chores.

- Robot Vacuum: Both the number of cleaning sessions and the total cleaned area are recorded. These metrics reflect not only frequency of use but also contribution to household cleanliness.

If data from any device are missing or incomplete, the system flags the inconsistency and notifies the user to ensure accuracy and transparency of the calculated indicators.

(b) Energy Consumption Data
The system periodically collects and stores power consumption data from LG ThinQ appliances. This information is used to analyze energy-saving performance relative to a baseline or user-defined family goal.

- Refrigerator: Continuous power usage is tracked to monitor efficiency and detect unnecessary consumption.

- Air Conditioner: Operating modes, usage duration, and hourly power draw are collected to evaluate peak-time usage and the effectiveness of energy-saving modes.

- Washing Machine: Energy consumption per washing cycle is measured in kilowatt-hours (kWh), allowing comparison between normal and eco modes.

- TV: Power consumption is logged during both active and standby states to identify idle usage patterns such as nightly standby power.

If any energy data are delayed or unavailable, the system identifies the gap and computes savings indicators only from verified data sources.

(c) Exercise and Wellness Data
The system gathers health and activity information from wearable devices such as LG Health or compatible smartwatches. Collected data are synchronized daily or weekly to generate wellness indicators.

- Step Count: The total number of steps taken per day is recorded as the baseline metric for physical-

activity tracking.

- Heart Rate: Continuous or periodic heart-rate measurements are logged to monitor exercise intensity and overall wellness, while abnormal spikes or missing data are filtered out to maintain accuracy.

- Active Minutes: The duration of light, moderate, and vigorous activity is aggregated to represent both the quality and sustainability of physical exercise.

In cases of synchronization failure or incomplete records, the system highlights the issue and alerts the user, ensuring transparency and reliability of wellness indicators.

C. Family Challenge System

(a) Challenge Creation and Recommendation

- The system provides action-based goal templates that allow users to select intuitive, behavior-oriented goals without entering numerical values.

- Each goal can be adjusted in intensity using a single slider control.

- The system supports three types of challenges:
  - Household Tasks (Weekly): Encourages family members to establish shared home-care routines.
  - Energy Saving (Weekly): Reinforces sustainable lifestyle habits.
  - Exercise & Wellness (Daily): Helps individuals maintain a balanced daily health rhythm.

- To generate relevant goals, the system analyzes user or family data from the previous two to four weeks to identify areas of low activity.

- Based on this analysis, it automatically recommends appropriate challenges together with the following information: the goal intensity level (Easy / Normal / Boost), an estimated success rate, and the intended duration period.

(b) Challenge Types & Engagement Mechanics
The challenge system is designed to promote repeated engagement by introducing short, rewarding, and interactive missions. Each challenge type motivates users through different mechanisms such as routine formation, family collaboration, surprise bonuses, and friendly competition.

1) Daily Challenge
Daily Challenges convert everyday habits into simple, time-bound missions that can be completed within a single day.

- Grants +4 points upon successful completion

- Automatically refreshed each morning based on the user's routine

- Encourages habit formation with instant feedback

- Upon completion, the user receives a point notification and visual confirmation (e.g., "+4 points")

2) Family Relay Challenge
A cooperative mission where each family member must complete their assigned portion in sequence. If any member fails, the entire challenge is not completed.

- Mission is passed sequentially (e.g., Parent → Sibling → Child)

- Successful completion rewards +6 points to each participating member

- Failure results in a shared notification, encouraging retry and family communication

3) Double Reward Day
Introduces an unexpected incentive by randomly selecting a challenge that awards twice the normal points.

- Designated challenge rewards 2× standard points

- Displayed with a highlighted label such as "2X Today"

- Encourages immediate participation and daily app access

4) Speed Challenge
When multiple family members undertake the same mission, only the first person to complete it receives a special bonus.

- Base reward plus a bonus for the first finisher

- Recognition is displayed on the family board (e.g., "First to Complete: Walking Challenge")

- Promotes light competition and individual initiative

(c) Challenge Execution and Evaluation

- Users can access the "Today's Recommended Challenges" section on the home screen, select one of three suggested cards, and start a challenge immediately by tapping the [Start Challenge] button.

- The system automatically evaluates challenge results at both the individual and family levels. Evaluation occurs on a daily or weekly basis depending on the challenge type.

- For family-level challenges, all members share a common goal, and each participant's contribution

is aggregated to calculate total achievement.

- Two calculation models are supported:
  - Aggregate Type: Individual completion counts from all members are summed to determine overall family progress.
  - Average Type: The average achievement rate of all members is used to evaluate family performance.
- Each member's data are weighted equally. If a member's data are missing, the system substitutes the family's average value and flags the record as *"Incomplete."*
- A challenge is considered successful when the final achievement rate meets or exceeds a predefined threshold (e.g., $\geq$ N times or $\geq$ 90%).

D. Tap1. Home Map & Character Space

(a) Home Appliance Map Interface

When the user opens the app, a home-shaped appliance map appears. The map is divided into living spaces such as the living room, kitchen, bedroom, and bathroom. Each space contains the relevant home appliances and the challenges linked to that area. For example:

- Living Room – Tidiness Challenge
- Kitchen – Energy-Saving Cooking Challenge
- Bathroom – Water-Saving Challenge

Users can freely explore the map by zooming in or navigating through it. Each area displays real-time icons such as "refrigerator in use" or "laundry completed." When tapped, a pop-up window shows the current challenge progress and related activity details.
UX Perspective: The map is designed so that family members can intuitively recognize their own actions within the familiar context of a home environment, helping them participate naturally and feel more engaged.

(b) Individual Character Setup

ach family member creates and customizes their own character. Basic avatars can be personalized with simple options such as clothing and hairstyle. The character represents each individual on the family board, and their actions and points are visually reflected in real time. For instance, completing an exercise challenge makes the character perform a running motion, while completing an energy-saving challenge brightens the character's surroundings. During character creation, users enter their name, family role, favorite color, and main challenge. All settings are synchronized across devices so that every family member sees the same screen.

(c) Real-Time Character Status

When a user taps on a character, they can instantly view that member's current challenge progress in real time.

The following information is displayed:

- Current Points – today's earned points, total accumulated points, and the contribution ratio within the family
- Earned Badges – e.g., "Runner of the Week (2nd Week of October)", "Walking King"
- Ongoing Challenge Progress – e.g., "Family Running Challenge (3.8km / 5km completed)"

When a challenge is completed, the user's points increase immediately, and a "+4p" floating balloon appears above the character's head for one second. The family progress gauge rises simultaneously, showing an animated fill effect to represent the team's growth. Even when the app is closed, if another family member completes a challenge, a small glowing light appears near the user's character with a notification such as "Mom finished the dishwashing challenge! +4pt" Upon successful completion of a challenge, the character's badge chest opens with a short popup message: "'Walking King' badge earned!" Tapping the badge chest reveals detailed information on when and through which challenge each badge was obtained. All of this data is synchronized in real time (within 1 second) via a WebSocket event stream, ensuring that every family member's screen reflects the same status simultaneously.

(d) IoT Integration for Challenge Updates

Each challenge automatically updates based on real data from connected IoT devices or sensors. For example, when the washing machine finishes an energy-saving cycle, points for the Energy-Saving Laundry Challenge are instantly reflected. Examples of connectable devices include:

- Washing machine – Energy-Saving Laundry Challenge
- Water purifier – Proper Hydration Challenge
- Air conditioner – Optimal Temperature Challenge
- Smartwatch – Daily Step Challenge

Delays in data synchronization should be minimized. If data cannot be retrieved, the last known status is temporarily displayed. Users can manually choose which devices are connected and configure them through the Device Connection Management menu.

(e) Character Animation and Progress Visualization

Each character's state changes in real time according to the challenge completion rate. The progress from 0% to 100% is represented in three stages.

Progress	State	Example Motion
0–30%	Idle	Sitting, yawning
31–70%	Active	Walking, tidying motion
71–100%	Complete	Jumping, clapping

When a challenge is completed, celebratory fireworks

and short sound effects play, and a speech bubble appears above the character indicating completion. If there is no progress for a certain period of time, the character shows boredom through gestures such as yawning or slouching. Characters interact with one another – performing gestures such as high-fives or cheering motions – to express family unity and enhance the sense of shared achievement.

E. Tap2. Collecting point & Reward Market

(a) Personal Point Display (Top-Right Indicator)
Each individual's personal point balance is displayed at the top-right corner of the screen, next to their profile icon. This serves as a quick visual reference for personal progress, separate from the shared family pool.

- When a user completes a challenge, their personal points increase instantly, appearing with a floating "+X" animation above the top bar.

- For example, if a user earns 4 points from a dishwashing challenge, the indicator momentarily displays "+4p" before updating the total.

Even without opening any menu, users can always see how many points they personally contributed today.

(b) View Total Family Points
Upon accessing the Point Collecting tab, users can view the current total family points at the top of the screen, displayed within a circular gauge or progress bar. Each member's contribution is represented through subtle markers or initials inside the bar, reinforcing a sense of shared effort rather than isolated progress.

(c) Reward Catalog
Scrolling further reveals the Reward Catalog, containing LG product discount coupons, gift vouchers, and seasonal offers. Each reward is displayed as an interactive reward card, including:

- Reward Image or Icon

- Required Points (e.g., 4,000P)

- Short Benefit Description – e.g., "₩30,000 Coupon – LG Puricare"

- Redeem Button – active only when sufficient points are available

Tapping on a reward card opens a detailed popup with terms, expiration date, and a confirmation button ("Redeem Now").

(d) Real-Time Point Updates

- When a user redeems a reward, the family's total point gauge instantly decreases, accompanied by a soft deduction animation (e.g., coins dispersing).

- A confirmation message appears: "LG Coupon Redeemed! Check 'My Coupons' to use it anytime."

- If the user lacks enough points, the Redeem button is replaced with a subtle message: "Not enough points – keep collecting!"

F. Tap3. Family Ranking

(a) Real-Time Family Ranking

- Families are ranked in real time based on their total accumulated points.

- Within the LG ThinQ app, users can view their family's current rank, total score, and any benefits unlocked through point-based rewards.

- By checking their real-time rank, users can feel a sense of friendly competition and shared motivation with other families.

G. Notification & Communication Layer

a. Mobile Push Notifications

- The system must provide mobile push notification functionality to continuously maintain user engagement and awareness.

  - Notifications must include challenge reminders, progress summaries, and system status alerts.

  - Notifications must be concise and non-intrusive, designed so that users can check information without feeling burdened.

b. Smart Speaker Announcements

- The system must provide voice notification functionality through smart speakers.

  - Users must be able to listen to family-level activity summaries without any manual operation.

  - Voice briefings must be provided on a daily and weekly basis.

- In homes connected with smart speakers, the system must deliver information through the following functions:

  - (a) Daily Briefing: Provides a voice summary of the previous day's main achievements and today's recommended challenges.

  - (b) Achievement Announcement: Broadcasts a voice message that the whole family can hear when a challenge is successfully completed.

- The notification system must extend the visually centered app experience into an auditory communication channel, allowing users to receive infor-

mation hands-free while cooking, cleaning, or exercising.

- All notification channels must be designed to maintain the flow of challenge participation without requiring continuous user attention.

H. System Architecture & Data Flow

a. IoT and Wearable Device Integration

- The system must be integrated in real time with IoT and wearable devices, automatically collecting data on household activities, energy consumption, and wellness.

- All collected data must be automatically transmitted to the backend system through the cloud API without user intervention.

- The following devices and data categories are included:
    - LG ThinQ Appliances: Includes washing, drying, dishwashing, refrigerator energy usage, air conditioner operating modes, TV on/standby states, and robot vacuum cleaning sessions and area coverage data.
    - Wearable Devices (e.g., LG Health, Smartwatches): Continuously provide health and activity data streams including step count, heart rate, and activity time by intensity. Each device must periodically or on-demand transmit activity logs, energy usage, cleaning sessions, step count, heart rate, and activity intensity data.

- All device and application events must be transmitted securely in real time. Timestamp assignment and queuing are performed in the backend system before data normalization and aggregation processing.

b. Data Processing Structure

- To efficiently process data from various sources, the system must adopt a hybrid structure combining event streaming and batch processing.

- The data collection and processing stages are as follows:
    - (1) Event Capture: Continuously collect raw data from IoT and wearable devices.
    - (2) Queue Management: Temporarily store incoming data in a scalable message queue to handle spikes in device traffic.
    - (3) Batch Reconciliation: Periodically verify and normalize accumulated events.
    - (4) Data Delivery: Deliver refined and structured data to the AI analytics pipeline, where it is aggregated and stored in the database for

visualization.

- The data processing workflow must ensure scalability, persistence, and fault tolerance throughout all operations.

c. Privacy and Security Controls

- Data Encryption and Transmission Security: All communications between IoT devices, wearable sensors, and backend servers must be encrypted using the HTTPS/TLS protocol to prevent eavesdropping or tampering during transmission.

- Secure Storage and Access Control: When stored on the server, data must be protected using AES-256 encryption. Access privileges must be clearly separated through Role-Based Access Control (RBAC) among users, administrators, and service modules.
    - User Level: Users can access only their own family data through authenticated sessions.
    - (b) Administrator Level: Administrators can access only anonymized aggregated data, which is limited to system maintenance and AI model improvement purposes.
    - (c) Service Module Level: Each application module can access only the necessary data subset, controlled by scoped API keys and signed tokens.

- Through this structure, the system ensures data privacy, access control, encrypted storage, and secure transmission paths, maintaining a highly reliable security layer even within a real-time IoT and AI-integrated environment.

III. DEVELOPMENT ENVIRONMENT

A. Platform Configuration and Selection Rationale

HomeQuest is designed as a web application based on Firebase Hosting, utilizing key services of Google Firebase such as Authentication, Cloud Firestore, Hosting, and Cloud Functions. Through this integrated architecture, user authentication, data management, server execution, and deployment are all handled within a single platform. Firebase fully supports HomeQuest's real-time data synchronization and family feedback structure, enabling rapid prototyping and significantly reducing the overall development cycle.

- *Integrated Backend Environment:* Firebase allows user authentication, data management, and deployment to be handled in a single unified environment, eliminating the need for separate server infrastructure.

- *Real-Time Data Synchronization:* By leveraging Firestore's real-time listener feature, challenge progress, point updates, and reward exchanges among family mem-

bers are synchronized across all devices within one second.

- *Cloud-Based Stability and Scalability:* Built upon Google Cloud infrastructure, Firebase ensures stable data processing and transmission even as the number of IoT and wearable devices increases, without performance degradation.

Overall, Firebase was selected as the optimal platform to achieve HomeQuest's goals of real-time collaboration and consistent family-level data management.

B. Programming Languages and Frameworks

- *Frontend:* JavaScript / React.js Offers high compatibility with the Firebase SDK and provides efficient state management for real-time data reflection.

- *Backend:* Node.js (Firebase Cloud Functions) Enables serverless function-level execution without the need for traditional server management.

- *AI Module:* Python Used to develop recommendation models and behavior pattern analysis based on family activity data.

C. Database Structure Selection (NoSQL / Firestore)

HomeQuest is a smart home platform centered on real-time data sharing and challenge-based family interaction. To realize this concept, Google Firebase's Cloud Firestore was selected as the main database. Firestore, built on a NoSQL structure, provides flexibility in data modeling and high-speed synchronization capabilities.

- *Flexible Data Structure:* Data entities such as families, members, and challenges are subject to frequent modification. Firestore's schema-free structure allows flexible and scalable data modeling to accommodate these changes efficiently.

- *Event-Oriented Architecture:* HomeQuest handles event-based data such as "family progress status" and "individual completion logs." Storing data in document-oriented structures rather than relational tables enables faster and more efficient query performance.

- *Real-Time Synchronization:* Firestore's real-time listener feature automatically updates all client interfaces when data changes occur. For example, when a family member completes a challenge, the global family leaderboard and character points are immediately updated across all devices.

- *Security Rule-Based Access Control:* Firestore employs rule-based access control, allowing user- and family-level permission management. By utilizing Firebase Security Rules, family data is securely isolated to prevent unau-

thorized access.

D. Cost Estimation

- Firebase (Spark Plan): Authentication, Firestore, and Hosting services – Free for small-scale usage.

- Development Tools: Visual Studio Code, GitHub, Node.js – Free and open source.

- Domain (Optional): homequest.dev – Approximately ₩10,000–₩15,000 per year.

Firebase's free Spark Plan provides authentication, database, and hosting services at no cost during the development and testing stages.

In the production stage, a **pay-as-you-go** billing system applies depending on usage volume (database reads/writes, storage, and traffic).

E. Development Environment

- Operating System: Windows 11 / macOS Sonoma

- Node.js: v20.x

- npm: v10.x

- Firebase CLI: v13.x

- Integrated Development Environment (IDE): Visual Studio Code

- Browser: Latest version of Google Chrome

- Version Control: Git + GitHub

- Deployment Environment: Firebase Hosting (Google Cloud Infrastructure)

F. Use of Commercial Cloud Platform

HomeQuest utilizes Google Cloud–based Firebase to manage the server, database, authentication, and hosting within a single cloud ecosystem.

Firebase's serverless architecture and auto-scaling capability ensure stable performance even under heavy load or increased IoT device connections.

Moreover, Firebase inherits Google Cloud's TLS encryption and security infrastructure, ensuring that IoT and wearable data are transmitted and stored securely.

This cloud-based structure technically supports HomeQuest's philosophy of real-time data sharing and emotional connection among family members, while maintaining high standards of system reliability, scalability, and data security.
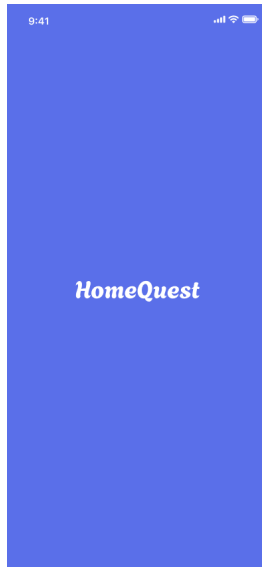
IV. SPECIFICATION

A. Initial Screen

Fig. 1. Initial Screen

When the app is launched, the service logo "HomeQuest" is displayed at the center of the screen. This serves as a welcome and identity screen, allowing users to recognize the brand before entering the main interface.
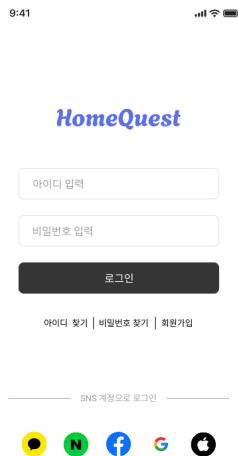
B. Log In

Fig. 2. Log In

1) ID Input
  - Description: Guides users to enter their login ID, either as an email address or phone number.

- Input Format: Accepts formats like user@example.com or 01012345678.

- Validation: Verifies whether the input follows a valid email or phone number format.

2) Password Input
  - Description: Prompts users to enter their password.

  - Input Format: Accepts text input, displayed as "•" symbols for privacy.

  - Validation: Checks for minimum length and required character combinations.

3) Log In Button
  - Action: Attempts to log in with the provided credentials.

  - On Success: Redirects to the main Home screen.

  - On Failure: Displays an error message ("Invalid ID or password").

4) Additional Options
  - Menu: Find ID / Find Password / Sign Up.

  - SNS Login: Supports quick login via Kakao, Naver, Facebook, Google, and Apple accounts.

C. Sign-Up

Fig. 3. Sign-Up

1) ID Input
  - Description: Allows users to create a new ID for login.

- Input Format: Alphanumeric string (e.g., home-quest01).

- Validation: Checks duplication through a "Check Availability" button.

2) Password Input
- Description: Users set a secure password for their account.

- Input Format: Combination of letters, numbers, and special characters (minimum 8 characters).

- Validation: Confirms whether the password matches the re-entered confirmation field.

3) Email Input
- Description: Collects email address for verification and account recovery.

- Input Format: Follows the format user@example.com.

- Validation: Checks email structure and allows domain selection.

4) Phone Number Input
- Description: Used for identity verification and notification purposes.

- Input Format: Numeric format such as 01012345678.

- Validation: Performs text message verification through "Send Code."

5) Sign-Up Button
- Action: Activates only when all fields are valid.

- On Success: Completes registration and redirects to the login screen.

- On Failure: Displays an error message for invalid or missing information.

D. Initial Setup

1) Profile Setup
- Description: This step allows users who have completed sign-up to enter their basic information.

- Input Fields: Nickname, Family Role

- Input Format:
  - Nickname: A string of 2–10 characters, consisting of Korean, English letters, or numbers.
  - Family Role: Free text input (e.g., Dad, Mom,

Fig. 4. Profile Setup

Son, etc.)
- Action:
  - When the user clicks the "Next" button, the entered information is saved, and the user proceeds to the next step.

- Validation:
  - After entering a nickname, users can click the "Check Duplication" button to verify if the nickname already exists.
  - If any input field is empty, the "Next" button is disabled.

2) Avatar Creation
- Description: This step allows users to create a 3D avatar that represents themselves.

- Feature: Users can swipe left or right to preview various avatar styles.

- Action: After selecting an avatar, the user clicks the "Next" button to save the selected avatar information and proceed to the next step.

3) Family Connection
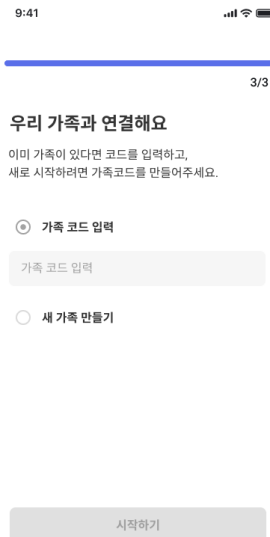
Fig. 5. Avatar Creation
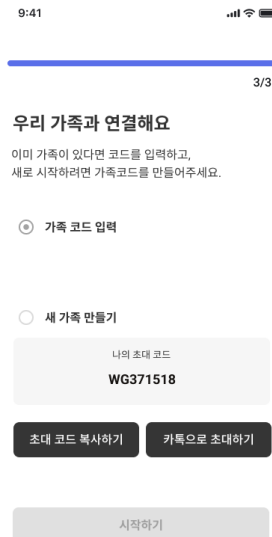

Fig. 6. Family Connection   Fig. 7. New Family Creation

- Description: This step allows users to either join an existing family group or create a new one.
- Feature Options:
  - Enter Family Code: Connect by entering the code provided by an existing family member.
  - Create New Family:
    * Create a new family group and automatically receive a unique family code.
    * The generated family code is displayed on the screen and can be shared using the

Copy button or the Invite via KakaoTalk option.

- Action:
  - When "Enter Family Code" is selected, the code input field becomes active.
  - When "Create New Family" is selected, a code is automatically generated and displayed on the screen.
  - Once all selections are complete, the user clicks the "Start" button to proceed to the Home screen.
- Validation:
  - The family code must be 6–8 characters long, containing letters and numbers.
  - If the format is incorrect or the input field is empty, the "Start" button is disabled.
  - If an invalid code is entered, an error message is displayed.
  - If the group exceeds the maximum of 5 members, an error message is displayed.
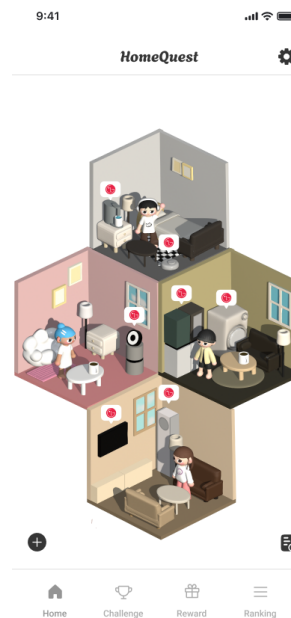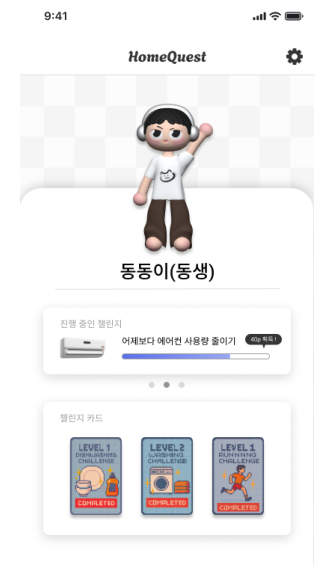
E. Home Page

Fig. 8. Home Page   Fig. 9. Character Detail Page

1) Room Navigation

- Description: The home screen is designed as an isometric map, where one character is placed in each room (e.g., living room, kitchen, laundry room, bedroom, etc.). Users can freely navigate the map

using drag and pinch gestures to move, zoom in, or zoom out.

- Components:
  - Isometric Map: Visually represents the overall house layout
  - Room icons
  - Appliance icons
  - Characters: avatars placed in each room
- Actions:
  - Tapping a room opens its Room Detail screen.
  - Smooth zoom-in and zoom-out animations are applied when moving between rooms.
  - The last viewed room is saved locally so that the same position is restored upon re-entry.

2) Character and Gauge Bar
- Description: Each room on the home screen displays the user's and family members' characters, with a gauge bar above each character indicating the challenge progress. The gauge bar interacts with the character's facial expressions or motions depending on the challenge status.
- Components:
  - Character: Avatar representing the user (includes basic idle motions)
  - Gauge Bar: UI showing challenge progress
  - Character Detail Tab
- Actions:
  - As the challenge progresses, the gauge gradually fills; upon completion, it reaches full capacity and triggers a firework animation.
  - Tapping the gauge bar opens the Character Detail Window as a popup.
  - The character's expressions and poses automatically change based on challenge progress.

3) Add Appliance Function
- Description: Users can add new LG appliances or replace existing devices in each room from the home screen. This function is linked to the user's actual LG (ThinQ) account, allowing only authenticated devices to be added.
- Components:
  - "+" Button: Displayed at the bottom-left corner of the Home tab
  - Appliance List Popup: Shows a list of available LG devices (name, model, connection status)
  - Connection Confirmation Modal: Displays

"Would you like to add this appliance?"

- Actions:
  - Clicking the "+" button opens a popup showing the list of available devices from the LG account.
  - After selecting a device and clicking "Add", the appliance icon appears in the corresponding room. Already-added devices are shown as disabled.
  - Added appliances are automatically linked to the challenge system and may display a Glow effect.
  - Appliance information is automatically saved both locally and on the server, ensuring persistence after re-entry.

4) Appliance Glow Interaction
- Description: LG appliances linked to active challenges are displayed with a soft glowing effect on the home screen. This allows users to easily recognize which appliances are currently associated with ongoing challenges.
- Components:
  - Appliance Icon: Represents the connected LG device
  - Glow Layer: Animated border shimmer effect
  - State Transition Logic: Controls visual effects for active and completed states
- Actions:
  - When a challenge is active, a subtle blinking effect is applied to the appliance icon.
  - Upon completion, a short bright flash indicates success before returning to normal state.
  - Animations are automatically optimized and simplified to brightness transitions in low-power mode.

5) Bottom Navigation Bar
- Description: A fixed bottom navigation bar allows users to switch between major screens of the app. Tabs provide quick access to Home, Challenge, Reward, and Ranking pages.
- Components:
  - Home: View the home screen and characters
  - Challenge: Access available challenge lists
  - Reward: Check points and reward history
  - Ranking: Compare user rankings
- Actions:

– Tapping a tab navigates to the corresponding page.

– The selected tab is displayed with a filled icon and bold text.

– Fade or slide animations are applied during transitions.
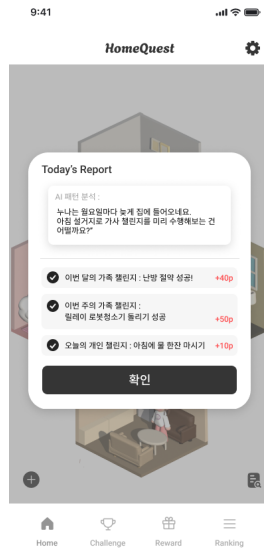
6) Home Report Overview

Fig. 10. Home Report

- Description: Displays the user's daily and weekly challenge progress through an AI-generated report. The section provides personalized insights based on appliance usage patterns and challenge performance.

- Components:

  – Report Card: Central container summarizing AI feedback and completed challenges

  – Challenge Log: Lists current and completed family and personal missions with corresponding points

  – Confirmation Button: Closes the report and returns to the main dashboard

7) Settings Page

  - Description: Allows users to manage personal information, linked devices, notification preferences, and language settings in a unified interface.

  - Components:

    – Profile Header: Displays the user's avatar, name, and email information

    – Settings List: Menu for device connection, family invitation, notifications, and language
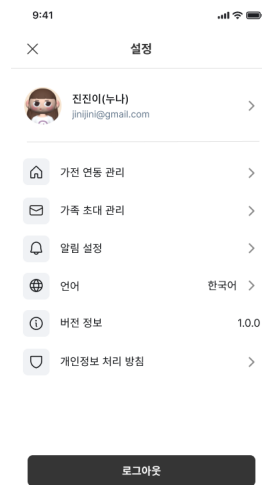

Fig. 11. Settings Page

selection

– Version and Policy Section: Shows current app version and privacy policy access

– Logout Button: Provides account sign-out functionality
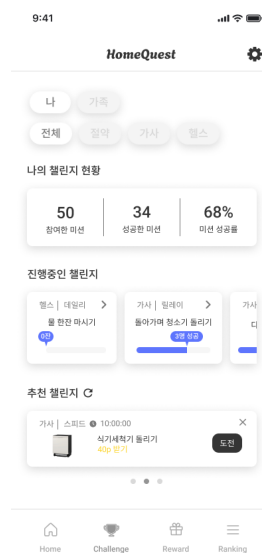
F. Challenge Page

Fig. 12. Challenge Page

1) Ongoing Challenges

  - Description: Displays all currently active challenges in a card-based layout.

  - Components:

– Category tags (e.g., Health, Energy Saving, Household)

– Challenge name (e.g., "Walk 10 Minutes," "Laundry Relay Challenge")

– Progress bar (%) showing completion rate

• Action: Tapping a card opens the detailed challenge screen.
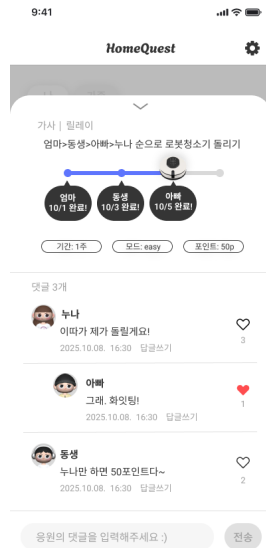
2) Challenge Detail

Fig. 13. Challenge Detail

• Description: Shows detailed progress by participant and mission information.

• Components:
  – Progress timeline (e.g., "Mom completed on 10/1," "Dad completed on 10/4")
  – Duration, difficulty, and reward points
  – Comment section for family encouragement messages with "Like" feature

• Action:
  – Comments are added in real-time when the "Send" button is pressed.
  – Each comment includes username, date, and like count.

3) Recommended Challenges
  • Description: Suggests new challenges based on user activity data.
  • Components:

– Challenge name, short description, reward points, and "Start" button

– Example: "Reduce air conditioner use (+20p)," "Do stretching (+5p)"

• Action: Pressing "Start" adds the challenge to the ongoing list.
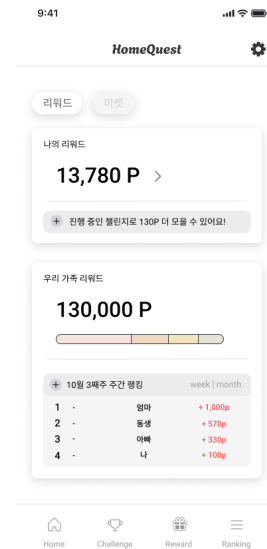
G. Reward Page

Fig. 14. Reward Page

1) Reward Overview
  • Description: The Reward tab serves as the main screen where users can view their current points as well as the total accumulated family points. Users can switch between the "Reward" and "Market" tabs using the top tab buttons, with the Reward screen displayed by default.
  • Components:
    – Tab buttons: "Reward", "Market"
    – Personal Reward Card
    – Family Reward Card
  • Actions:
    – Selecting the "Reward" tab keeps the current screen active.
    – Selecting the "Market" tab switches to the Market screen.
    – Clicking the " > " button on Personal Reward card opens the Timeline screen.

2) My Reward

- Description: Displays the user's current point balance and the additional points that can be earned from ongoing challenges. Users can tap the "View Details (>)" button to check their detailed point history.
- Components:
  - Current point balance (e.g., 13,780P)
  - Message indicating additional earnable points (e.g., "You can earn 1,300P more from ongoing challenges!")
  - View Details button (>)
- Actions:
  - When "View Details" is clicked, the user is redirected to a timeline view showing point acquisition history, including challenge name, date, and earned points.
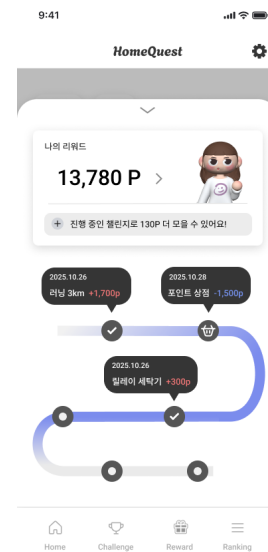
3) Family Reward
- Description: Displays the total accumulated family points by summing up all members' contributions. A contribution bar visually represents each member's input, and the weekly and monthly rankings show each member's point growth over time.
- Components:
  - Total family points (e.g., 130,000P)
  - Contribution bar
  - Ranking list showing member name, rank, and points earned
    * Period filter tabs: "Weekly/Monthly"
- Actions:
  - Switching between "Weekly / Monthly" changes the ranking view accordingly.
  - A point increase indicator (e.g., +1,000P) is shown beside each member's name for the selected period.
  - Data automatically refreshes every Monday and on the 1st day of each month.

4) Timeline
- Description: The Timeline page displays the detailed history of all point activities in chronological order. Users can view their entire point flow including challenge participation, reward boxes, and market exchanges at a glance.
- Components:
  - Point history list (Challenge name / Date / Point change)
  - Filter buttons: "All / Earned / Used"
  - Period selection dropdown (e.g., This Week /

Fig. 15. Reward Timeline

This Month / All)
  - Summary bar: Displays total points earned and used for the selected period
- Actions:
  - The list is sorted in descending order, showing the most recent records first.
  - Filters allow users to toggle between earned and used point views.
  - Selecting a specific period displays only data from that timeframe.
  - Infinite scrolling automatically loads previous records as the user scrolls.

5) Reward Market
- Description: Accessible by tapping the "Market" button at the top of the Reward tab, this screen allows users to browse and redeem items or coupons using their accumulated points.
- Components:
  - Point balance display
  - Product card list (Image, Name, Required Points, Exchange Button)
  - Category filter (e.g., Discount Coupons / Gift Icons, etc.)
- Actions:
  - Tapping the "Exchange" button opens a confirmation popup: "Would you like to redeem this item for 2,000P?"
  - Upon confirmation, the corresponding points are deducted, and a redemption success mes-
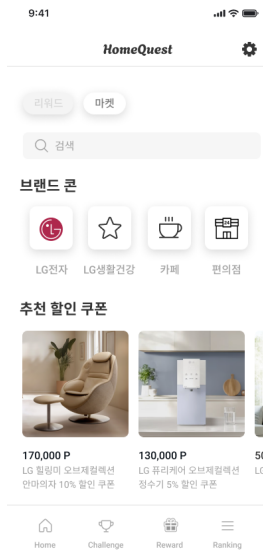
Fig. 16. Reward Market

sage appears.
- All redemption records are automatically added to the user's personal reward history in the My Reward Details section.

H. Family Ranking Page

Fig. 17. Family Ranking Page

Fig. 18. Ranking Info Page

1) My Family Rank
   - Description: Displays the family's current ranking within their residential area.

- Components:
  - Shown as "Your family is ranked 36th."
  - Ranking basis: families living in the same district (e.g., Seongdong-gu, Seoul).
  - Refresh cycle: Rankings reset monthly on the 1st day of each month (seasonal ranking system).

2) Regional Leaderboard
   - Description: Visually displays the point rankings of families living in the same area.
   - Components:
     - Top 3 families displayed with medal icons (Gold, Silver, Bronze).
     - Ranks 4–10 display rank number, family name, points, and up/down arrows.
   - Visual Effects:
     - Rank changes are indicated with arrows (for up, for down).
     - When rising: The row background flashes red for 2 seconds.
     - When falling: The row background flashes blue for 2 seconds.
     - This animation plays automatically during ranking updates, allowing users to perceive changes instantly.
   - Action:
     - Pressing the "?" icon shows a popup explaining the ranking system.
     - Popup text: "Rankings reset on the 1st of every month. Families compete within the same area. Higher ranks increase reward box odds, which include coupons and bonus points."

3) Reward Box
   - Description: Provides visual feedback and rewards for seasonal achievements or ranking milestones.
   - Action:
     - Displays an animated reward box with a message: "Congratulations! +100p."
     - When the user presses "OK," points are instantly added to their account.
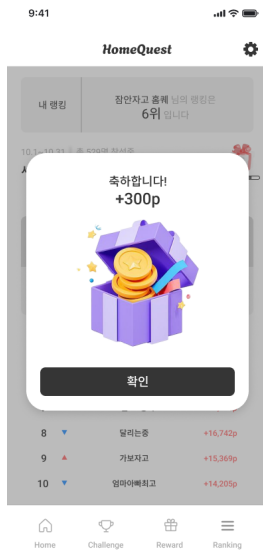
Fig. 19. Reward Box

I. Firestore Data Structure

0. Top-level Collections

```
/users/{userId}         // User base document
/families/{familyId}    // Family base document
/challenges/{challengeId}  // Challenge definitions (templates)
/products/{productId}   // Redeemable products
/randomBoxes/{boxId}    // Random box definitions
```

1. users

```
/users/{userId}
  nickName: string            // user nickname
  email: string
  avatarUrl: string
  familyId: string | null     // belonging family code
  roleInFamily: 'dad' | 'mom' | 'child' | null
                              // role in family
  orderInChild: number | null // child order (ex. 1=first child)
  label: string | null        // specific role for UI (ex. 'daughter', 'son')
  totalPoints: number         // user's accumulated points (PERSONAL!)
  createdAt: timestamp
  lastLoginAt: timestamp
```

1-1. User challenge progress

```
/users/{userId}/challengeProgress/{progressId}
  challengeId: string          // ref: /challenges/{challengeId}
  status: 'ONGOING' | 'DONE' | 'EXPIRED'
    // ONGOING: still in progress
    // DONE    : finished in time (Success)
    // EXPIRED: expired without completion (Failure)

  currentValue: number
    // accumulated challenge contribution value to date. (ex. 18,000 steps walked)
  targetValue: number
    // target value to complete the challenge. (ex. goal - walk 30,000 steps)
  startedAt: timestamp
  endedAt: timestamp | null

  // denormalized fields from challenge
  challengeTitle: string
  challengeCategory: string
```

1-2. User challenge contributions

```
/users/{userId}/challengeProgress/{progressId}/contributions/{contributionId}
  value: number                    // ex. 1 time, 3000 steps
  actionType: string               // concrete action ex. 'WALK' | 'CLEAN' | 'DRINK'...
  proofUrl: string | null          // proof image (optional)
  createdAt: timestamp
```

1-3. User purchases

```
/users/{userId}/purchases/{purchaseId}
  productId: string                // ref: /products/{productId}
  productName: string              // denormalized for history view
  price: number
  purchaseType: 'PERSONAL'
  purchasedAt: timestamp
```

2. families

```
/families/{familyId}
  name: string
  code: string                 // family join code
  region: string | null        // region for ranking UI
  familyPoints: number         // shared family points
  createdAt: timestamp
```

2-1. Family challenge progress

```
/families/{familyId}/challengeProgress/{progressId}
  challengeId: string
  status: 'ONGOING' | 'DONE' | 'EXPIRED'
  currentValue: number
    // accumulated challenge contribution value to date by FAMILY. (ex. 18,000 steps walked)
  targetValue: number
    // target value to complete the challenge by FAMILY. (ex. goal - walk 30,000 steps)
  startedAt: timestamp
  endedAt: timestamp | null

  // denormalized
  challengeTitle: string
  challengeCategory: string
```

2-2. Family challenge contributions

```
/families/{familyId}/challengeProgress/{progressId}/contributions/{contributionId}
  userId: string                   // actual challenge performer
  value: number                    // ex. 1 time, 3000 steps
  actionType: string               // concrete action ex. 'WALK' | 'CLEAN' | 'DISH' ...
  proofUrl: string | null
  createdAt: timestamp
```

2-3. Family purchases

```
/families/{familyId}/purchases/{purchaseId}
  productId: string
  productName: string
  price: number
  purchaseType: 'FAMILY'           // family-level purchase
  purchasedBy: string              // userId (which family member presses the purchase button)
  purchasedAt: timestamp
```

3. challenges (Templates)

```
/challenges/{challengeId}
  title: string
  description: string
  category: 'clean' | 'energy' | 'health' | 'etc'
                                  // challenge domain
  scope: 'PERSONAL' | 'FAMILY'    // challenge performer scope
  progressType: 'AGGREGATE' | 'COUNT' | 'ONCE'
  targetValue: number | null      // target value to complete the challenge ex. 30000
  unit: string | null             // ex. 'steps', 'times', 'kWh', 'photo' ...
  actionType: string | array<string>
    // concrete action(s)
    // ex. 'WALK' or ['CLEAN','DISH','LAUNDRY']

  durationType: 'DAILY' | 'WEEKLY' | 'MONTHLY'
  startAt: timestamp | null
  endAt: timestamp | null
  baseReward: number              // if success, point reward
  isActive: boolean
  createdAt: timestamp
```

4. randomBoxes

4-1. Random boxes definition

```
/randomBoxes/{boxId}
  name: string                    // box name
  gageMax: number                 // gage threshold for opening
  itemType: 'POINT' | 'COUPON' | 'PRODUCT'
                                  // reward type
  itemValue: number | string      // actual reward, ex. 3000, "Starbucks Gifticon"
```

4-2. Random boxes history

```
/families/{familyId}/randomBoxHistory/{historyId}
  boxId: string                   // opened box
  itemValue: number | string      // reward at that time
  openedBy: string                // who opened
  openedAt: timestamp
```

5. Challenge Comment

```
/challenges/{challengeId}/comments/{commentId}
  userId: string
  content: string
  createdAt: timestamp
  likes: number
```

REFERENCES

[1] Samsung SmartThings Energy. Available: https://www.smartthings.com/energy
[2] Strive App. Available: https://www.striveapp.com
[3] FamilyAlbum by Mixi. Available: https://family-album.com/
[4] Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. Proceedings of the 19th International Conference on World Wide Web (WWW 2010), 661–670. Available: https://doi.org/10.1145/1772690.1772758