TimeHero

Team Members:

Andrew Nguyen, Justin Nguyen, Nathan Hoang, Kaleb Perkins

What is TimeHero?

Imagine an app that helps you take control of your screen time while motivating you to achieve more. Set a limit, and we keep you on track. Want more screen time? Complete beneficial tasks across different categories: fitness, learning, or mindfulness to earn extra time. Stay focused with built-in tools like timers and notifications. If you leave the app while a task is running, we detect it to keep you honest.

The Problem:

- High screen time is a growing concern for individuals across all age groups.
- Excessive screen use leads to procrastination, distraction, and affects well-being.
- Popular apps like Kids360 and OurPact lack motivation or engagement.

Our Vision:

- Time Hero is a productivity app designed to help you take control of your screen time while motivating you to accomplish more.
- Gamified productivity app rewarding users to earn screen time by completing a set of productive tasks.
- By completing positive tasks, users are encouraged to find balance and purpose in their day.

Target Audience

FOR (Parents and People Struggling with High Screen Time)

WHO (Want to manage screen time and boost productivity)

TimeHero is a (productivity mobile app)

THAT (Allows users to earn screen time by completing productive tasks)

UNLIKE (Existing screen time limiting apps)

OUR PRODUCT (Rewards users for healthy and productive behaviors by allowing them to earn more screen time through positive actions)

Implementation

Frontend: React Native for cross-platform mobile app development. ExpoGo enables efficient coding while providing access to native device features .

- State Management: Redux to manage tasks, user data, and time tracking.

Backend: Node.js with Express, SQLite for data persistence.

- APIs: RESTful APIs for user info, task progress, and leaderboard updates.

Key Features

Gamified Productivity System

- Earn screen time by completing tasks in categories like fitness, learning, or mindfulness. Tasks are rewarded based on completion, encouraging balance and productivity.

Dynamic Leaderboard

- Friendly competition with friends by comparing earned screen time.

Task Categories with Customization

- Predefined tasks like "30 minutes outside" or "45 minutes of schoolwork." Option to add or customize tasks to suit individual needs.

Screen Time Management Tools

- Track earned and redeemed screen time. Redeem using a timer-based interface.

Accountability Features

- App detects when users leave tasks mid-way and prevents progress updates.

Intuitive User Interface

- Simple, engaging screens like Home, Leaderboard, and Time Screen Pages. Interactive buttons, progress bars, and animations enhance usability.

Core Features

Robust

- Database with tables for users, tasks, and leaderboards. API endpoints for fetching user data, tasks, and leaderboard rankings.

State Management with Redux

- Efficient handling of user data, completed tasks, and screen time accumulation. Real-time updates for key components like the leaderboard and time tracker.

Customizable Task Framework.

- Core logic for defining tasks with properties like time, metric, and steps. Flexibility to add new task types without breaking existing functionality.

Authentication and User Profiles

- Secure user login and profile management. Track completed tasks, accumulated screen time, and leaderboard participation.

Gamification Logic

- Algorithms for rewarding screen time based on task completion. Leaderboard logic to rank users by accumulated time.

Video Demonstration

https://youtu.be/ms2Y8qIq3Do

Contributions

Andrew Nguyen: Frontend task management and Redux state, Backend APIs for tasks and user data, database schema, Sign in and home screen

Justin Nguyen: Frontend and Backend leaderboard, home screen, and time screen

Nathan Hoang: Frontend and Backend leaderboard, home screen, and task list

Kaleb Perkins: Frontend and Backend First Time Sign-in Slideshow

Challenges and Solutions

- Managing asynchronous call order
- Correcting data models
- State management
- Merge conflicts while collaborating

Questions?