Eötvös Loránd University Faculty of Informatics

Department of Media and Educational Technology

PixelRatings

Supervisor:

Abonyi-Tóth Andor

Senior Lecturer, PhD.

Author:

Naz Demirsoy

Computer Science BSc.

Budapest, 2024

Thesis Registration Form

Student's Data:

Student's Name: Demirsoy Naz

Student's Neptun code: BLD0YM

Course Data:

Student's Major: Computer Science BSc

I have an internal supervisor

Internal Supervisor's Name: *Andor Abonyi-Tóth*

Supervisor's Home Institution: ***Department of Media and Educational Technology***

Address of Supervisor's Home Institution: ***1117, Budapest, Pázmány Péter sétány 1/C.***

Supervisor's Position and Degree: *Senior Lecturer, Ph.D.*

Thesis Title: PixelRatings

Topic of the Thesis:

(Upon consulting with your supervisor, give a 150-300-word-long synopsis os your planned thesis.)

My Thesis topic is a mobile app for game review and rating system where users can register and rate the games they have played. The system will store user data in a database, including their username, password, and personal information. Once registered, users can search for games, read reviews, and rate games on a scale of 1 to 10. The system will then calculate the average rating for each game and update it.

To ensure safety and prevent misuse, users will not be able to create new games. Instead, the app will use an API to pull game data from an existing database. The app will also display new releases, enabling users to stay up-to-date with the latest games.

The app will be designed with two types of users in mind: those who have registered and those who have not. Users who have not registered will still be able to browse games and read reviews, but they will not be able to rate or review games themselves.

Overall, this app will provide a platform for game enthusiasts to share their experiences and opinions, as well as discover new games to play. It will also serve as an informative resource for anyone interested in the world of gaming.

Budapest, 2023. 05. 11.

Acknowledgements

First of all, I would like to express my sincere gratitude to my supervisor, Abonyi-Tóth Andor, for their invaluable mentorship, constant support, and insightful critiques throughout my research journey.

To my lovely family, words can't fully say how grateful I am. I am who I am today because of all you've given up and how much you believe in me. Your constant love, encouragement, and support have always guided me. Every success I have shows the strength and inspiration I get from you. You are with me in everything I do.

To my friends, particularly my roommates, your companionship during these years has been invaluable. Your steadfast support in both the good times and the challenging moments, offering comfort, love, and solace, has been indispensable. Your faith in me during times of self-doubt has been a source of immense gratitude. I cherish our shared memories and eagerly anticipate those yet to come.

In summary, this thesis is a tribute to the enduring support and contributions of all those mentioned. Without your guidance, love, and belief in me, this achievement would not have been attainable. I am deeply thankful for your integral role in my academic journey and for aiding me in reaching this significant milestone.

Table of Contents

Chapter 1

Introduction

PixelRatings app is your new go-to destination for discovering and sharing insights about video games. Whether you're a seasoned gamer or just dipping your toes into the world of gaming, PixelRatings is designed to be easy, fun, and informative. Our app is all about connecting you with a community of gamers and providing a space where you can explore, rate, and discuss your favorite video games. With PixelRatings, you'll have the power to make informed decisions about what to play next, based on ratings and reviews from gamers just like you.

At PixelRatings, we believe that everyone should have access to honest and insightful game reviews. That's why our app is user-friendly and inclusive, welcoming both registered users who want to actively participate in our gaming community and those who prefer to browse and read reviews without signing up. You'll find a wide range of games to explore, and our easy-to-navigate interface makes it simple to find exactly what you're looking for. Whether you're in search of the latest game releases or want to find out what others think about a classic title, PixelRatings is here to enhance your gaming experience.

1.1 Purpose

PixelRatings: Your Gateway to the Gaming World - PixelRatings is designed to be a comprehensive and engaging platform for video game enthusiasts and casual players alike. Our app serves as a dynamic space where users can delve into the vast universe of video games, offering a rich blend of community-driven reviews and detailed game insights. The purpose of PixelRatings is to empower gamers with the tools and information they need to make informed decisions about the games they play and to share their experiences with a like-minded community.

At the heart of PixelRatings lies the desire to create a user-friendly, interactive, and informative environment where gamers can feel at home. Whether you are looking to explore the latest releases, find in-depth reviews on timeless classics, or simply want to rate a game you've recently played, PixelRatings provides a one-stop solution. By leveraging the collective wisdom of our gaming community, we aim to build a platform where every game is given a fair and accurate representation based on real player experiences.

Key Objectives:

1. Community Engagement and Interaction: PixelRatings is more than just an app; it's a community. We are committed to fostering a vibrant and interactive space where gamers can connect, share, and learn from each other. Our platform encourages users to actively participate by rating and reviewing games, thus contributing to a rich database of player perspectives and experiences. This communal approach not only enriches the quality of content on our platform but also creates a sense of belonging among users.
2. Informed Gaming Choices: We understand that choosing a game to play can be overwhelming given the plethora of options available. PixelRatings aims to simplify this decision-making process by providing comprehensive reviews and ratings from fellow gamers. Our app collates these user-generated insights to present an average rating for each game, thereby guiding users towards choices that align with their interests and preferences.
3. Discoverability of Games: With a constantly updated catalog of games, including the latest releases, PixelRatings is a treasure trove for discovering new and exciting gaming experiences. We strive to keep our users at the forefront of the gaming world, ensuring they are always informed about new additions and popular titles. This feature is particularly beneficial for users who wish to expand their gaming horizons and explore genres or titles they might not have considered before.
4. Inclusivity and Accessibility: PixelRatings is designed to be inclusive and accessible to all users, regardless of their level of gaming expertise or involvement. While registered users can actively contribute by rating and reviewing games, non-registered users also have the ability to browse and read reviews. This inclusive approach ensures that PixelRatings caters to a wide spectrum of users, from hardcore gamers to those who are simply curious about the gaming world.

1.2 Target audience

PixelRatings is designed with a diverse and inclusive audience in mind, appealing to everyone from hardcore gamers to those who are just discovering the joy of gaming. Our app is an ideal platform for anyone with an interest in video games, regardless of their gaming proficiency or genre preference. This includes individuals who are passionate about exploring new game releases, seasoned players looking for in-depth reviews and insights, and casual gamers who enjoy occasional gaming and wish to stay informed about popular trends.

1.3 Overall Structure and Key Features

PixelRatings is designed with a clear and user-friendly structure, making it effortless for users to navigate and utilize its various features. The app is segmented into several key areas, each tailored to enhance the user experience in exploring and interacting with the world of video games. These areas include the Home Screen, Game Discovery Screen, Review and Ratings Screen, and the User Profile Screen.

1. Home Screen: As the central hub of PixelRatings, the Home Screen provides a welcoming interface that showcases trending games, latest releases, and top-rated titles. This screen is the starting point for users, offering a quick overview of what's popular and new in the gaming world. It’s designed to be engaging and informative, allowing users to easily access different sections of the app.
2. Game Discovery Screen: This section is dedicated to helping users explore and find games. It features robust search functionality, category filters, and sorting options to tailor the discovery process according to user preferences. Whether users are looking for a specific title or wanting to browse through different genres, this screen makes the exploration process both efficient and enjoyable.
3. Review and Ratings Screen: Central to the app's purpose, this screen allows users to read detailed reviews and view ratings for each game. Registered users can submit their own ratings and reviews, contributing to the community-driven database. This interactive feature encourages users to share their experiences and opinions, providing valuable insights for others.

4. User Profile Screen: For registered users, this personal space within the app allows them to manage their account, view their review history, and customize their preferences. It’s a personalized dashboard that keeps track of their contributions and preferences, enhancing their engagement with the app.

5. Favorite Screen: A personalized section where users can save and view their favorite games, creating a tailored and easily accessible collection of preferred titles.

Each section of this documentation will delve deeper into these key areas, providing a detailed overview of their functionalities, design considerations, and user interaction flow. The aim is to offer both users and developers a comprehensive understanding of PixelRatings, ensuring a seamless and enjoyable experience in the exciting world of gaming. As you explore this documentation, you’ll gain insights into how each feature of PixelRatings contributes to creating a vibrant and informative platform for all gaming enthusiasts.

Chapter 2

System Requirements

2.1 Hardware requirements

For the PixelRatings mobile application to function at its best, users should consider the following hardware specifications for their devices:

1. Device Type: The application is compatible with both Android and iOS platforms, including smartphones and tablets. This broad compatibility ensures accessibility for a wide range of users.

2. Memory (RAM): A minimum of 2GB RAM is recommended. This ensures smooth multitasking and an overall fluid user experience, allowing the app to run without lags or interruptions.

3. Processor: A device with a quad-core processor (or better) is advisable. A more powerful processor will efficiently manage the computational demands of the app, supporting quick loading times and responsive interactions.

4. Storage Space: Users should have around 30MB of free space on their device. This space is needed to install PixelRatings and accommodate the data that the app will generate during use, such as user preferences and cached information.

It's important to note that the performance of PixelRatings can vary based on the operating system version and the specific model of the device. Generally, newer models and recent OS versions will offer a more optimized and enjoyable experience with the app. Users are encouraged to consider these specifications to fully enjoy all the features and functionalities that PixelRatings has to offer.

2.2 Software requirements

To ensure smooth operation and optimal performance of the PixelRatings mobile application, it is recommended to have the following software specifications:

Android or iOS operating system: The PixelRatings app can be deployed on devices running Android 5.0 (Lollipop) or later and iOS 10.0 or later. Ensure that your target devices meet these minimum operating system requirements.

2.3 Installation Guide

To install and set up the PixelRatings application on your environment, follow these steps:

1. Install Node.js and React Native CLI:

a. PixelRatings is built using React Native. Start by installing Node.js from https://nodejs.org. Node.js is essential for running the JavaScript codebase and managing packages.

b. After installing Node.js, install React Native CLI by running `npm install -g react-native-cli` in your command line. This will install the React Native command line interface globally on your machine.

2. Set up Your Development Environment:

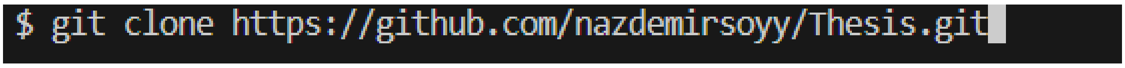
a. Choose an Integrated Development Environment (IDE) that is compatible with React Native. Options include Visual Studio Code (https://code.visualstudio.com/) or Android Studio (https://developer.android.com/studio) for Android app development.

b. Install extensions or plugins in your IDE that are relevant for React Native and Expo development to streamline your development process.

3. Clone the PixelRatings Repository:

a. Ensure Git is installed on your machine for version control. You can download and install it from https://git-scm.com/downloads.

b. Clone the PixelRatings repository from its GitHub repository. In your terminal or command prompt, navigate to the directory where you want to place the project and run the command git clone https://github.com/nazdemirsoyy/Thesis.git



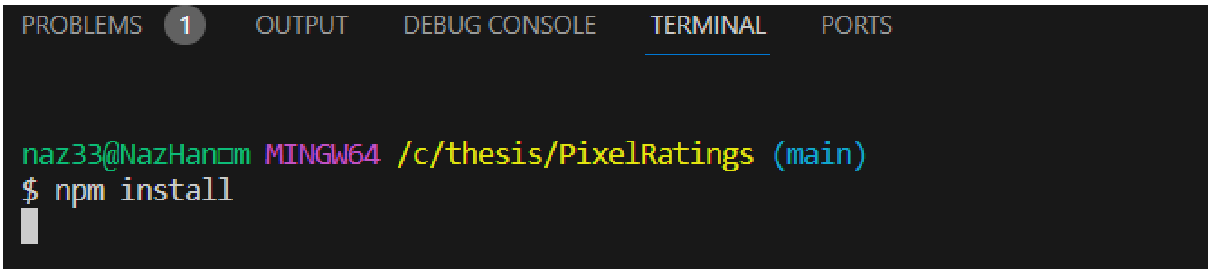
```
$ git clone https://github.com/nazdemirsoyy/Thesis.git
```

Figure 1: Cloning the Repository

4. Install Dependencies:

a. Navigate to the project directory (PixelRatings_app) in your terminal or command prompt.

b. Run the command npm install to install all the dependencies listed in the project’s package.json file.



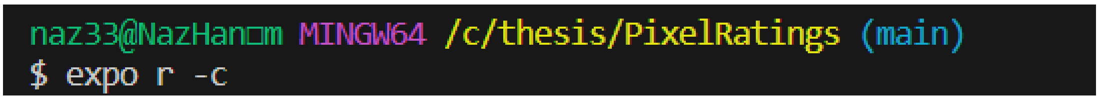
```
naz33@NazHan□m MINGW64 /c/thesis/PixelRatings (main)
$ npm install
```

Figure 2: Installing Dependencies

5. Run the Application with Expo:

a. In the project directory, run the command expo r -c. The -c flag clears the React Native packager's cache, which can be useful for resolving potential caching issues.

b. Expo will start and open a new tab in your default web browser with the Expo developer tools. From here, you can run the app on an iOS or Android emulator, or on a physical device using the Expo client app.



```
naz33@NazHan□m MINGW64 /c/thesis/PixelRatings (main)
$ expo r -c
```

Figure 4: Starting the App with Expo

6. Testing on Physical Devices (Optional):

a. To test the application on a physical device, download the Expo client app from the App Store (iOS) or Google Play Store (Android).

b. Scan the QR code displayed in the Expo developer tools using the Expo client app to open and run PixelRatings on your device.

Following these steps will allow you to install and run the PixelRatings application on your development environment, enabling you to explore its features and functionalities.

2.4 Starting the Application

Once the PixelRatings mobile application is successfully installed on your device, you can start using it by following these steps:

1. Locate the PixelRatings app icon on your device's home screen or app drawer.
2. Tap on the PixelRatings app icon to launch the application.
3. The PixelRatings app will open, presenting you with the main screen and its various features and functionalities.

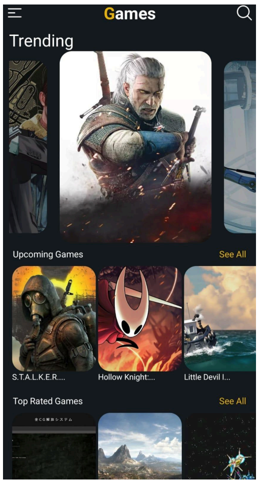


Figure 5: Application Screen

By following these steps, you can easily start using the PixelRatings mobile application and embark on a journey of mindfulness and well-being.

Note: The application may require certain permissions, such as access to the device's data, storage, speakers depending on the specific features being utilized. Grant the necessary permissions as prompted by the app to enable full functionality.

2.4 Dependencies

Dependencies are essential libraries and components that the PixelRatings app relies on to provide specific functionalities and integrate seamlessly with various services. These external code resources enhance the app's capabilities, streamline the development process, and ensure efficient operation across different platforms and environments.
The dependencies used in the PixelRatings application are as follows:

- "@expo/vector-icons": "^13.0.0",
- "@expo/webpack-config": "^19.0.0",
- "@react-native-async-storage/async-storage": "1.18.2",
- "@react-native-firebase/storage": "^18.5.0",
- "@react-navigation/stack": "^6.3.18",
- "axios": "^1.6.2",
- "axios-mock-adapter": "^1.22.0",
- "babel-preset-expo": "^9.5.2",
- "cors": "^2.8.5",
- "deprecated-react-native-prop-types": "^5.0.0",
- "expo": "~49.0.8",
- "expo-camera": "~13.4.4",
- "expo-linear-gradient": "^12.3.0",
- "expo-splash-screen": "~0.20.5",
- "expo-status-bar": "~1.6.0",
- "express": "^4.18.2",
- "firebase": "^10.5.0",
- "heroicons": "^2.0.18",
- "metro-react-native-babel-preset": "^0.77.0",
- "mongoose": "^7.6.2",
- "nativewind": "^2.0.11",
- "react": "18.2.0",

- "react-dom": "18.2.0",
- "react-native": "0.72.6",
- "react-native-document-picker": "^9.0.1",
- "react-native-heroicons": "^3.2.0",
- "react-native-image-picker": "^7.0.2",
- "react-native-progress": "^5.0.0",
- "react-native-ratings": "^8.1.0",
- "react-native-reanimated": "~3.3.0",
- "react-native-reanimated-carousel": "^3.5.1",
- "react-native-safe-area-context": "4.6.3",
- "react-native-screens": "~3.22.0",
- "react-native-snap-carousel": "^3.9.1",
- "react-native-svg": "13.9.0",
- "react-native-user-avatar": "^1.0.8",
- "react-native-web": "^0.19.9",
- "tailwindcss": "^3.3.2",
- "test": "^3.3.0",
- "twrnc": "^3.6.4"

Each of these dependencies plays a critical role in the functionality of PixelRatings. They provide a wide range of features, from UI components and navigation to backend integration and database management. By leveraging these dependencies, PixelRatings ensures a robust, efficient, and user-friendly experience for its audience.

Chapter 3
User Documentation

3.1 Opening screen

The opening screen is the initial screen displayed when the Pixel Ratings app is launched. You can tap anywhere on the screen to proceed to the main page.

Figure 6: Opening Screen

3.2 Home & navigation

The home page is the main hub of the Pixel Ratings app that offers various features. It displays 3 options for Trending, Upcoming and Top Rated games. Users can easily navigate to different screens and games by tapping on the corresponding icons.

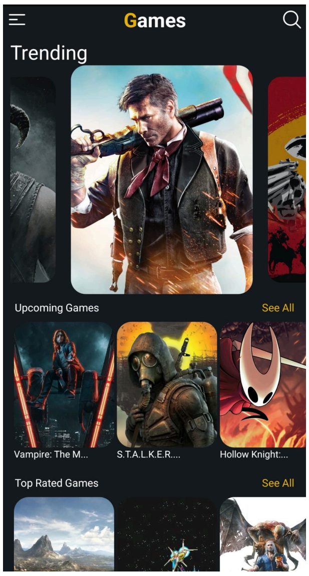


Figure 7: Home page

The navigation bar is a common component used throughout the Pixel Ratings app. Tap on the corresponding icon to navigate between the Home, Profile and About pages.

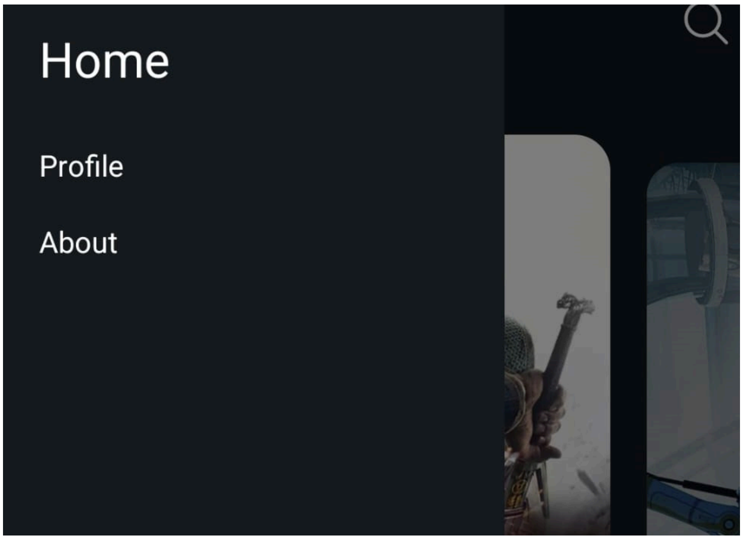


Figure 8: Navigation bar

3.3 Logging in & signing up

The sign-up page allows new users to create an account by providing their name, email address, and password. After filling in the required information, users can press the "Sign Up" button to register. If successful, they will receive a confirmation message and be redirected to the home page. In case of any errors, such as invalid email or password, appropriate error messages will be displayed.

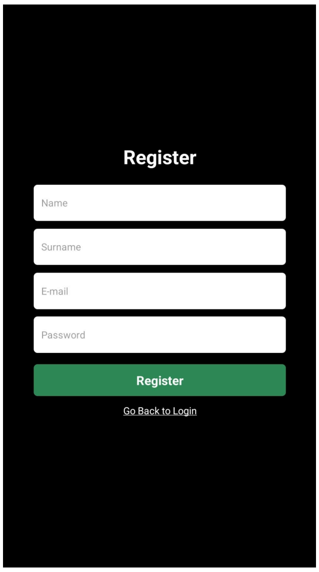


Figure 9: Login Page

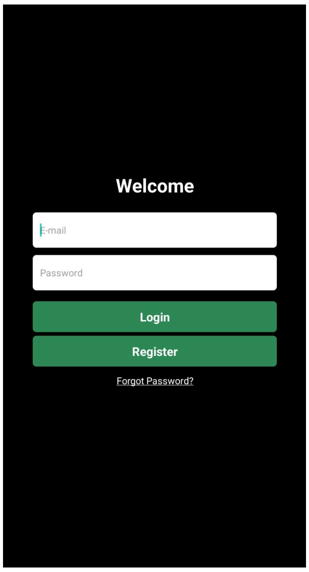


Figure 10: Register Page

The login page allows registered users to access their accounts by entering their email address and password. After providing the required credentials, users can press the "LogIn" button to authenticate. If the provided information is correct, they will be directed to the home page. If there are any authentication errors, such as incorrect email or password, appropriate error messages will be shown to the user.

3.4 Forgot password

The forgot password page allows users to reset their password by providing their email address. Upon submitting the email address, a reset password email will be sent to the user's email, which contains the link and instructions on how to reset their password.

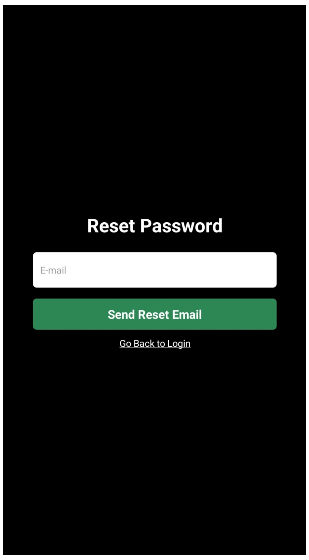
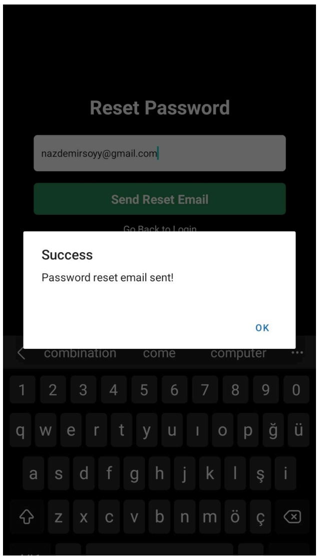


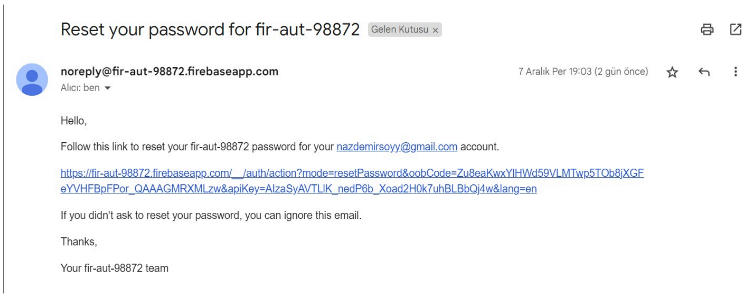
Figure 11: Forget Password Screen



Success

Password reset email sent!

Figure 12: Reset email sent successfully



Reset your password for fir-aut-98872

noreply@fir-aut-98872.firebaseapp.com
Alıcı: ben

7 Aralık Per 19:03 (2 gün önce)

Hello,

Follow this link to reset your fir-aut-98872 password for your nazdemirsoyy@gmail.com account.

https://fir-aut-98872.firebaseapp.com/__/auth/action?mode=resetPassword&oobCode=Zu8eaKwxYIHWd59VLMTwp5TOb8jXGFeYVHFBpFPor_QAAAGMRXMLzw&apiKey=AIzaSyAVTLIK_nedP6b_Xoad2H0k7uhBLBbQj4w&lang=en

If you didn't ask to reset your password, you can ignore this email.

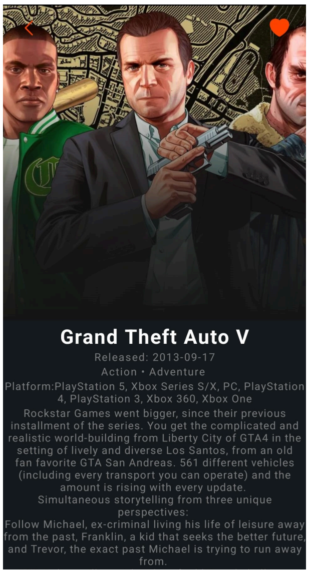
Thanks,

Your fir-aut-98872 team

Figure 13: Reset email sent by firebase

3.5 Game page

The Game Screen page allows users to showcase detailed information about games, including their name, release date, genre, and available platforms. It also features key functionalities such as the ability to favorite games, submit ratings, and leave comments, enhancing user interaction with the app.



Grand Theft Auto V

Released: 2013-09-17

Action • Adventure

Platform:PlayStation 5, Xbox Series S/X, PC, PlayStation 4, PlayStation 3, Xbox 360, Xbox One

Rockstar Games went bigger, since their previous installment of the series. You get the complicated and realistic world-building from Liberty City of GTA4 in the setting of lively and diverse Los Santos, from an old fan favorite GTA San Andreas. 561 different vehicles (including every transport you can operate) and the amount is rising with every update.

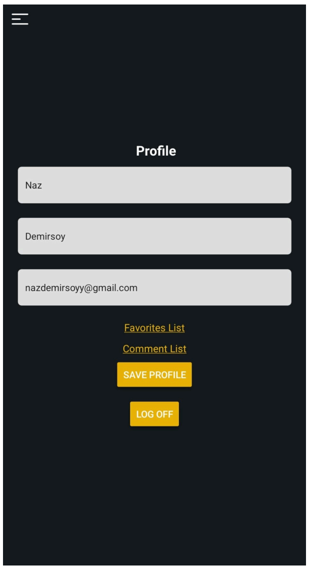
Simultaneous storytelling from three unique perspectives:

Follow Michael, ex-criminal living his life of leisure away from the past, Franklin, a kid that seeks the better future, and Trevor, the exact past Michael is trying to run away from.

Figure 14: Game Screen

3.6 Profile

The profile page allows users to view and manage their personal information and account settings. It displays user details such as First Name, Last name and email. Users can edit their information. Additionally, users can log out of their account or access other features like the Favorites List and Comment List.



Naz

Demirsoy

nazdemirsoyy@gmail.com

Figure 15: Profile Screen

3.7 Favorites page

The Favorites page enables users to easily access their favorite games. By simply clicking on the game icon, they can swiftly navigate to the Game Screen. Within the Game Screen, users have the option to either add or remove games from their favorites list by clicking on the heart icon located in the top right corner.

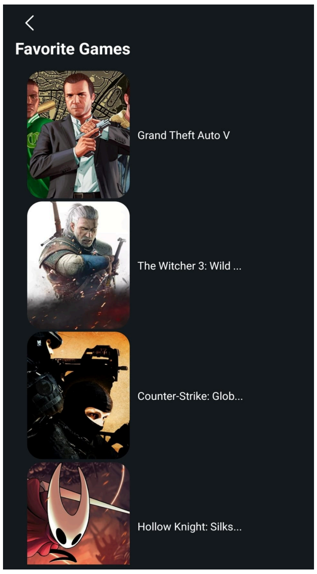
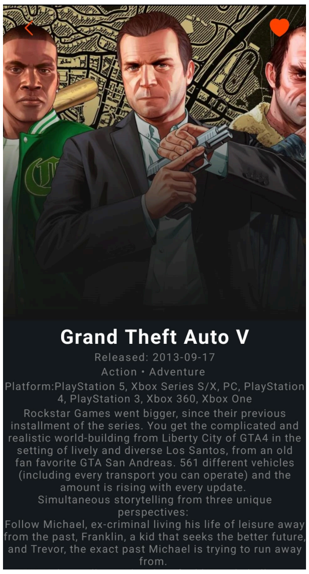


Figure 16: Favorites Screen



Grand Theft Auto V

Released: 2013-09-17

Action • Adventure

Platform:PlayStation 5, Xbox Series S/X, PC, PlayStation 4, PlayStation 3, Xbox 360, Xbox One

Rockstar Games went bigger, since their previous installment of the series. You get the complicated and realistic world-building from Liberty City of GTA4 in the setting of lively and diverse Los Santos, from an old fan favorite GTA San Andreas. 561 different vehicles (including every transport you can operate) and the amount is rising with every update.

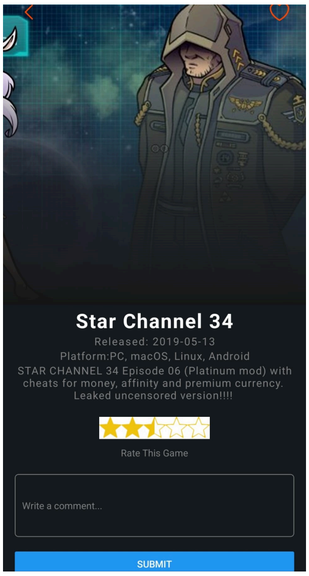
Simultaneous storytelling from three unique perspectives:

Follow Michael, ex-criminal living his life of leisure away from the past, Franklin, a kid that seeks the better future, and Trevor, the exact past Michael is trying to run away from.

Figure 17: Favorited Game Screen

3.8 Comments page

On the Game Screen, users have the ability to both rate and comment on games. Subsequently, in the Comments page, users can review their own comments and ratings for each game, along with the date of their comment submission. The page also provides a feature for sorting comments based on their posting date.



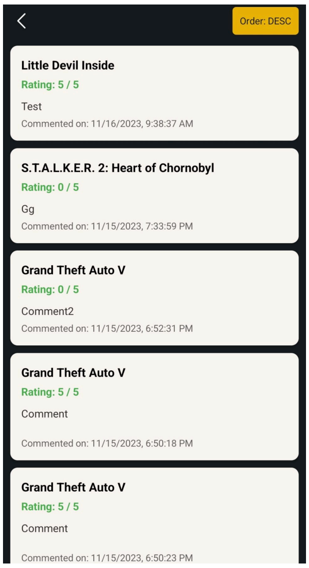
Star Channel 34

Released: 2019-05-13

Platform:PC, macOS, Linux, Android

STAR CHANNEL 34 Episode 06 (Platinum mod) with cheats for money, affinity and premium currency. Leaked uncensored version!!!!

Figure 18: Comment Box



Little Devil Inside
Rating: 5 / 5
Test
Commented on: 11/16/2023, 9:38:37 AM

S.T.A.L.K.E.R. 2: Heart of Chornobyl
Rating: 0 / 5
Gg
Commented on: 11/15/2023, 7:33:59 PM

Grand Theft Auto V
Rating: 0 / 5
Comment2
Commented on: 11/15/2023, 6:52:31 PM

Grand Theft Auto V
Rating: 5 / 5
Comment
Commented on: 11/15/2023, 6:50:18 PM

Grand Theft Auto V
Rating: 5 / 5
Comment
Commented on: 11/15/2023, 6:50:23 PM

Figure 19: Comment screen

3.9 Search page

Users can search for games using text input by entering the game's name or selecting a genre. When a successful search is conducted, the results will showcase the game poster along with its name. In the event of an error, a failure image will be displayed.

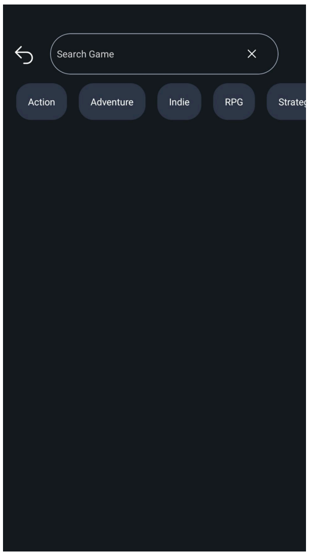


Figure 20: Search Screen

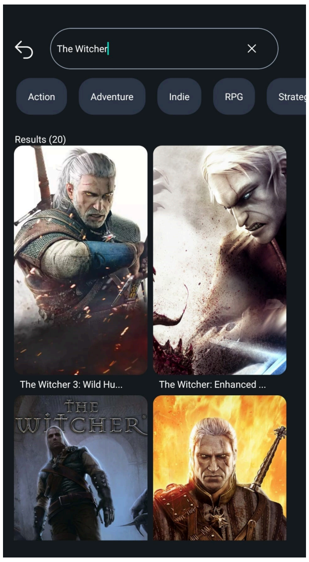


Figure 21: Successful Search Result

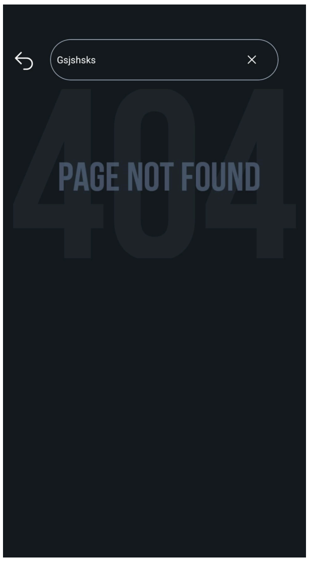
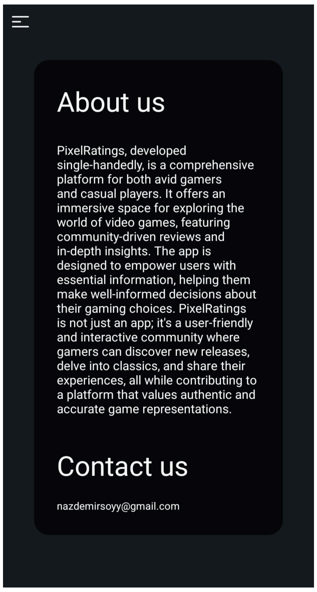


Figure 22: Unsuccessful Search Result

3.10 About page

The About screen in the app succinctly presents an overview of the app's features and purpose. It also includes the developer's contact information for user feedback and inquiries.



About us

PixelRatings, developed single-handedly, is a comprehensive platform for both avid gamers and casual players. It offers an immersive space for exploring the world of video games, featuring community-driven reviews and in-depth insights. The app is designed to empower users with essential information, helping them make well-informed decisions about their gaming choices. PixelRatings is not just an app; it's a user-friendly and interactive community where gamers can discover new releases, delve into classics, and share their experiences, all while contributing to a platform that values authentic and accurate game representations.

Contact us

nazdemirsoyy@gmail.com

Figure 23: About Page

Chapter 4

Developer Documentation

4.1 Program structure

The PixelRatings app features a straightforward and effective program structure designed for easy maintenance, scalability, and organized code. The architecture emphasizes separating different parts of the app, such as user interface, data management, and API connections, to ensure each section works well independently while still contributing to the overall functionality. This approach makes the app robust and easier to update or expand in the future.

4.1.1 Logical structure

PixelRatings employs a structured architecture, adapting the Model-View-ViewModel (MVVM) pattern, suitable for its React Native framework, ensuring a clear separation of concerns:

Models: These are classes representing data structures, like Game, User, Review, etc. They manage the app's data and business logic. Each model encapsulates its respective data and includes methods for handling data manipulation and access.

Views: The Views in PixelRatings are implemented using React Native components. Each screen in the app corresponds to a View component. Their primary role is rendering the user interface and displaying data to users. They interact with ViewModels to get data and update the UI based on user interactions.

ViewModels: Serving as the link between Models and Views, ViewModels process and prepare data for the Views. They handle the business logic and state management aspects of the app. ViewModels retrieve data from the Models, process it, and then expose it to the Views. They also handle user inputs and actions, updating the Models as needed.

Services: This layer includes services for handling external interactions like API calls, database operations, and authentication. Services provide a streamlined way for ViewModels to request and receive data or perform operations outside the app's core logic.

Utility Classes: These are helper classes and functions that provide common functionalities across the app. They include utilities for tasks like data formatting, validation, network connectivity checks, and other reusable code snippets to support the app's functionality.

This architecture ensures that PixelRatings remains modular, maintainable, and scalable, with each component fulfilling its role efficiently within the React Native ecosystem.

4.1.2 Physical structure

PixelRatings, built using React Native, has a well-organized physical structure optimized for performance and scalability:

Data Structures: The app employs efficient data structures to store and organize data, such as game details, user profiles, reviews, and ratings. These structures are designed to be intuitive and align with the app's features, ensuring quick access and manipulation of data.

Databases: PixelRatings uses Firebase Firestore as its primary database solution. Firestore's real-time data synchronization, scalability, and offline capabilities make it an ideal choice for managing the dynamic and interactive data needs of the app. This includes user authentication data, game ratings, user comments, and favorites list.

Modules: The app's codebase is divided into distinct modules, each handling specific functionalities like user authentication, game details display, reviews management, and user settings. This modular approach aids in maintaining a clean and organized code structure, simplifying updates and scalability.

External APIs: PixelRatings integrates external APIs to fetch game-related data. This setup allows the app to provide up-to-date and comprehensive information about various games, enhancing the user experience.

Utility and Helper Functions: The app includes utility modules that contain reusable code and helper functions. These are crucial for tasks such as data formatting, validation, and network requests, providing common functionalities across the app.

Component Library: Utilizing a library of reusable UI components, PixelRatings ensures consistency and efficiency in its user interface development. This library includes custom buttons, input fields, rating stars, and more, tailored to the app's design system.

This physical structure of PixelRatings, with its focus on modular design, efficient data management, and integration of external services, ensures that the app is robust, maintainable, and scalable within the React Native framework.

4.2 UI/UX design

The PixelRatings app's UI/UX design is meticulously crafted to cater to game enthusiasts, focusing on providing a seamless and engaging user experience. Here are the key design principles it incorporates:

Minimalism: The app adopts a minimalist design to ensure a clutter-free interface. This approach emphasizes the content – game images, ratings, and reviews – and presents it in a clean, straightforward manner. The minimalist design reduces distractions, making it easier for users to focus on the games and their respective community feedback.

Consistency: PixelRatings maintains a consistent design language across all its screens. Uniform typography, color schemes, and button styles are used throughout the app, creating a cohesive and harmonious user experience. This consistency enhances user comfort and familiarity with the app's interface.

Intuitive Navigation: The app features intuitive navigation, allowing users to effortlessly move between different sections such as game libraries, review areas, and user profiles. This user-friendly navigation ensures that features and information are easily accessible, enhancing the overall user experience.

Responsive Layout: With a responsive layout, the app adapts smoothly to various devices and screen sizes. This flexibility is essential in providing a great user experience across different platforms, whether users are on a smartphone, tablet, or desktop.

Visual Feedback: PixelRatings provides clear visual feedback for user interactions. Animations and visual cues are used to confirm actions like rating a game or writing a review. This feature not only makes the app more interactive but also reassures users that their inputs have been successfully recorded.

Overall, the UI/UX design of PixelRatings is focused on creating an environment that is both aesthetically pleasing and functional. The design choices are aimed at enhancing the user experience, making it enjoyable for users to explore new games, revisit favorites, or engage with the gaming community on the app.

4.3 React Native and JavaScript

React Native, a popular open-source framework developed by Facebook, is used for building PixelRatings. It allows for the creation of cross-platform applications using JavaScript, a widely-used programming language. Here are the key benefits of using React Native in the development of PixelRatings:

Cross-platform Development: React Native enables the development of applications for multiple platforms, such as iOS and Android, using a single JavaScript codebase. This feature is instrumental in ensuring consistency in functionality and appearance across different devices.

Live Reload and Hot Reloading: React Native enhances the development process with features like live reload and hot reloading. These allow developers to immediately see the impact of the latest changes without needing to restart the app, thereby speeding up development and debugging.

Component-based UI Development: The framework employs a component-based architecture, making the UI highly modular and reusable. This approach leads to more manageable code and simplifies the development process, as components can be independently developed and assembled.

Performance: React Native applications are known for their near-native performance. While the core of the application is written using JavaScript, React Native enables portions of the code to be written in native languages, ensuring smooth and efficient performance.

Access to Native Capabilities: React Native provides the ability to integrate with native device features such as camera, geolocation, and sensors. This integration is crucial for apps like PixelRatings, which may require advanced functionalities to enhance the user experience.

Extensive Ecosystem and Community Support: The framework boasts a robust ecosystem with numerous libraries and plugins, which can extend the app’s capabilities without the need for custom development. Additionally, the large community around React Native offers extensive support, resources, and best practices.

In summary, React Native's cross-platform capabilities, efficient development process, component-based architecture, performance, access to native features, and strong ecosystem make it an excellent choice for building PixelRatings. This approach facilitates the creation of a feature-rich, high-performance, and user-friendly app that appeals to gaming enthusiasts on various platforms.

4.4 Frontend components

4.4.1 Opening screen

The opening screen is the initial screen displayed when the PixelRatings app is launched. It provides a visually appealing welcome experience for users before they proceed to the main content.

4.4.2 Home page

The page consists of the following components:

- Game Highlights Carousel: A rotating carousel that showcases featured games, offering users a glimpse of popular or new titles.
    - Upcoming Games Section: Dedicated to highlighting games that are soon to be released. This section uses the GameList component with the title "Upcoming Games" and is populated with data from the upcoming array.
    - Top Rated Games Section: Showcases games that have received the highest ratings from the community. This section also utilizes the GameList component, titled "Top Rated Games", and is fed with data from the topRated array.
    - Trending Games Section: Highlights games that are currently popular within the community. This section uses the GameList component, appropriately labeled "Trending Games", and displays content from the trending data array.
- Search Bar: Allows users to search for games by title or genre. This feature enhances discoverability within the app.
- Navigation Bar: Includes icons for home, profile and about, allowing users to easily navigate to different sections of the app.

4.4.3 Navigation bar

The navigation bar consists of the following components:

- Home: When tapped, it navigates the user to the home page.
- Profile: When tapped, it navigates the user to the profile screen.
- About: When tapped, it navigates the user to the about screen.

4.4.4 Login page

The page consists of the following components:

- Email input field: Users should enter their email address here.
- Password input field: Users should enter their password.
- Login button: Initiates the login process when tapped. If the login is successful, a success message is displayed and the user is redirected to the home page. If an error occurs during login, an error message is displayed based on the error.
- Forgot password link: When tapped, it navigates to the forgot password screen in order to reset password.
- Sign up link: Directs the user to sign up in case they don't have an account yet. When tapped, it navigates to the signup screen for creating a new account.

4.4.5 Register page

The page consists of the following components:

- Name input field: Users should enter their First name and Last name here.
- Email input field: Users should enter their email address here.
- Password input field: Users should enter their password here.
- Sign up button: Initiates the sign up process when tapped. If the login is successful, a success message is displayed and the user is redirected to the home page. If an error occurs during signup, an error message is displayed based on the error.
- Login link: Directs the user to login case they already have an account. When tapped, it navigates to the login screen for signing in to an existing account.

4.4.6 Forgot password page

The page consists of the following components:

- Email input field: Users should enter their email address here.
- Reset password button: Initiates the sign up process when tapped. If the reset email is sent successfully, a success message is displayed and the user is redirected to the home page. If an error occurs during the password reset, an error message is displayed based on the error.
- Back button: Directs the user back in case they decide not to reset their password. When tapped, it navigates back to the login screen.

4.4.7 Profile page

The page displays different components based on whether a user is signed in or not.

When the user is signed in:

- Information cards: Display user information in a card format. The information displayed are as follows: First name, Last name and Email
- Edit details: Allows the user to edit their First name, Last name and Email by simply tapping the input field and changing. The user can save the changes by tapping to Save Profile Button.
- Log Off button: When tapped, logs the user out of the app.
- Navigation bar: Allows the user to navigate to other sections of the app.

When the user is not signed in:

- Login button: Directs the user to login case they already have an account. When tapped, it navigates to the login screen for signing in to an existing account.
- Register button: Directs the user to sign up in case they don't have an account yet. When tapped, it navigates to the register screen for creating a new account.
- Navigation bar: Allows the user to navigate to other sections of the app.

4.4.8 Game Screen

The page consists of the following components:

- Detailed Game Information: Provides comprehensive details about each game.
- User Interaction: Enables users to mark games as favorites and share their ratings and comments. Interacts with Firebase to retrieve user data, manage favorites, and update user ratings and comments.

4.4.9 All Game Screen

The page consists of the following components:

- Game Listing: Displays a list of games with images and names.
- Navigation Integration: Enables navigation to the detailed game screen upon selecting a game.

4.4.10 Favorite Game Screen

The page consists of the following components:

- Favorite Games Listing: Displays a personalized list of the user's favorite games.
- Navigation to Game Details: Each game in the list can be tapped to navigate to its detailed view.
- User Authentication Check: Ensures that only logged-in users can view their favorite games.

4.4.11 Search Screen

The page consists of the following components:

- Dynamic Search: Allows users to search for games by title and by genre.
- Navigation Integration: Enables seamless navigation between search results and game detail screens.
- Loading State Management: Provides feedback to the user during data fetching. Fetches the list of game genres on component mount and updates the genres state.

4.4.12 Comments and Ratings Screen

The page consists of the following components:

- Personalized Data Display: Shows the user's comments and ratings for games.
- Dynamic Sorting: Allows users to sort their comments in ascending or descending order.
- Integration with Firebase: Retrieves user-specific data from Firebase.
- Helper Functions:
  - formatDate: Formats the timestamp of comments into a readable date string.
  - toggleOrder: Toggles the sorting order between ascending and descending.
  - getSortedData: Sorts the gamesWithComments based on the current order direction.

4.5 Firebase

Firebase serves as a crucial backend solution for the PixelRatings mobile application, chosen for its extensive range of services and robust features that significantly enhance the app's functionality and reliability. As a comprehensive platform offered by Google, Firebase provides essential services like authentication, cloud storage, real-time database, and hosting. Its scalable and secure infrastructure assures smooth performance and effective data management.

In the context of PixelRatings, Firebase is utilized for several key functionalities:

User Authentication: Using getAuth from Firebase, the app handles user authentication processes, ensuring secure access to the app's features.

Data Storage and Management: The app leverages getFirestore to interact with Firebase's cloud database. This feature is crucial for storing and retrieving data related to user profiles, game lists, and user preferences.

Favorite Games Management: The functions addGameToFavorites and removeGameFromFavorites utilize Firebase to update the user's favorite games list, showcasing Firebase's capability to handle dynamic data changes.

User Ratings and Comments: The function fetchRatingsFromUser retrieves user-submitted ratings and comments for games, demonstrating Firebase's ability to manage user-generated content effectively.

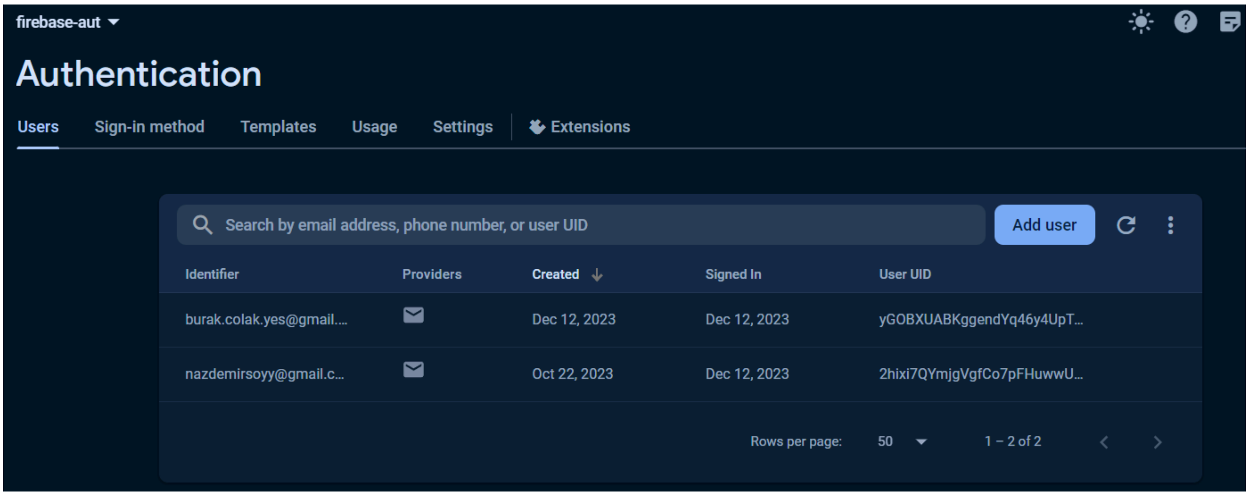
Real-time Data Updates: By employing Firebase's real-time database features, the app ensures that any changes in user data, such as favorite games or ratings, are promptly reflected across the platform.

Efficient Data Queries: With functions like doc, getDoc, setDoc, updateDoc, and arrayUnion, the app can efficiently query and manipulate data in the Firebase database, enhancing the user experience by providing timely and relevant game information.

The integration of Firebase into PixelRatings empowers the app with reliable backend services, enabling the developers to concentrate on refining the core features of the app while relying on Firebase for crucial backend functionalities.

4.5.1 Firebase authentication

Firebase Authentication is a pivotal feature within PixelRatings, offering robust and secure user authentication mechanisms. It supports multiple authentication methods, including email and password sign-in, and social logins, facilitating a flexible sign-in experience. Firebase Authentication ensures users can register, sign in securely, and manage their profiles with ease. The service emphasizes user data security, incorporating defenses against common security threats. With Firebase Authentication integrated into PixelRatings, users enjoy a smooth and personalized experience while the app upholds stringent security protocols to protect user information.



Identifier	Providers	Created ↓	Signed In	User UID
burak.colak.yes@gmail....	✉	Dec 12, 2023	Dec 12, 2023	yGOBXUABKggendYq46y4UpT...
nazdemirsoyy@gmail.c...	✉	Oct 22, 2023	Dec 12, 2023	2hixi7QYmjgVgfCo7pFHuwwU...

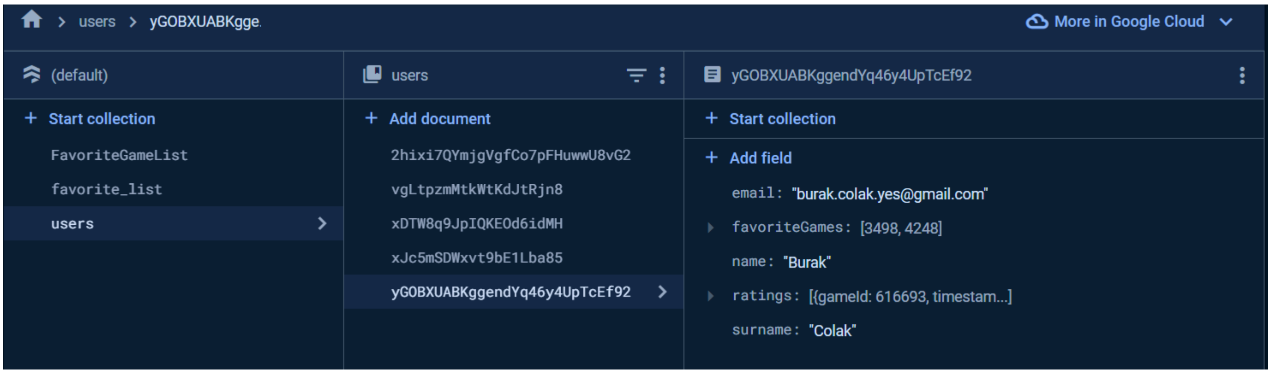
Rows per page: 50 1 – 2 of 2

Figure 24: Firebase authentication

4.5.2 Firestore cloud storage

Firebase Cloud Firestore serves as the primary backend for the Calmly app, enabling us to store and retrieve data seamlessly. We have structured our data into different collections to organize and manage specific information required by the application. The following collections are utilized:

- Users: This collection stores user-related information such as name, surname, email addresses, favorite games and ratings. It enables us to manage user accounts and retrieve personalized data.
  - favoriteGames: An array of numbers, possibly representing game IDs which are marked as favorites by the user.
  - ratings: An array that likely contains objects or identifiers for user-submitted ratings. This array includes at least one entry that has a numeric identifier and a timestamp, suggesting the user has rated at least one game.



FavoriteGameList
favorite_list
users

2hixi7QYmjgVgfCo7pFHuwwU8vG2
vgLtpzmMtkWtKdJtRjn8
xDTW8q9JpIQKEOd6idMH
xJc5mSDWxvt9bE1Lba85
yGOBXUABKggendYq46y4UpTcEf92

email: "burak.colak.yes@gmail.com"
favoriteGames: [3498, 4248]
name: "Burak"
ratings: [{gameId: 616693, timestam...]
surname: "Colak"

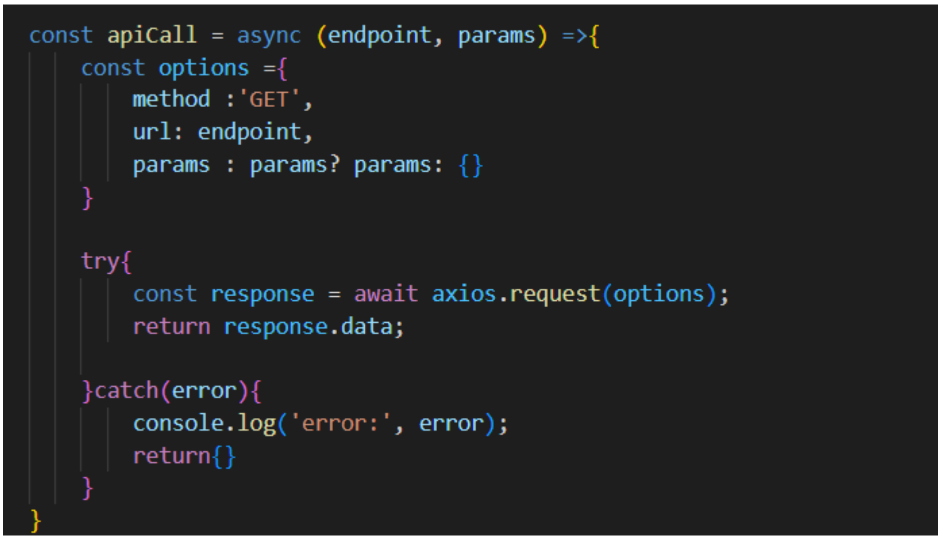
Figure 25: Firebase Cloud Storage

4.6 RAWG Video Games Database API

4.6.1 Api Calls

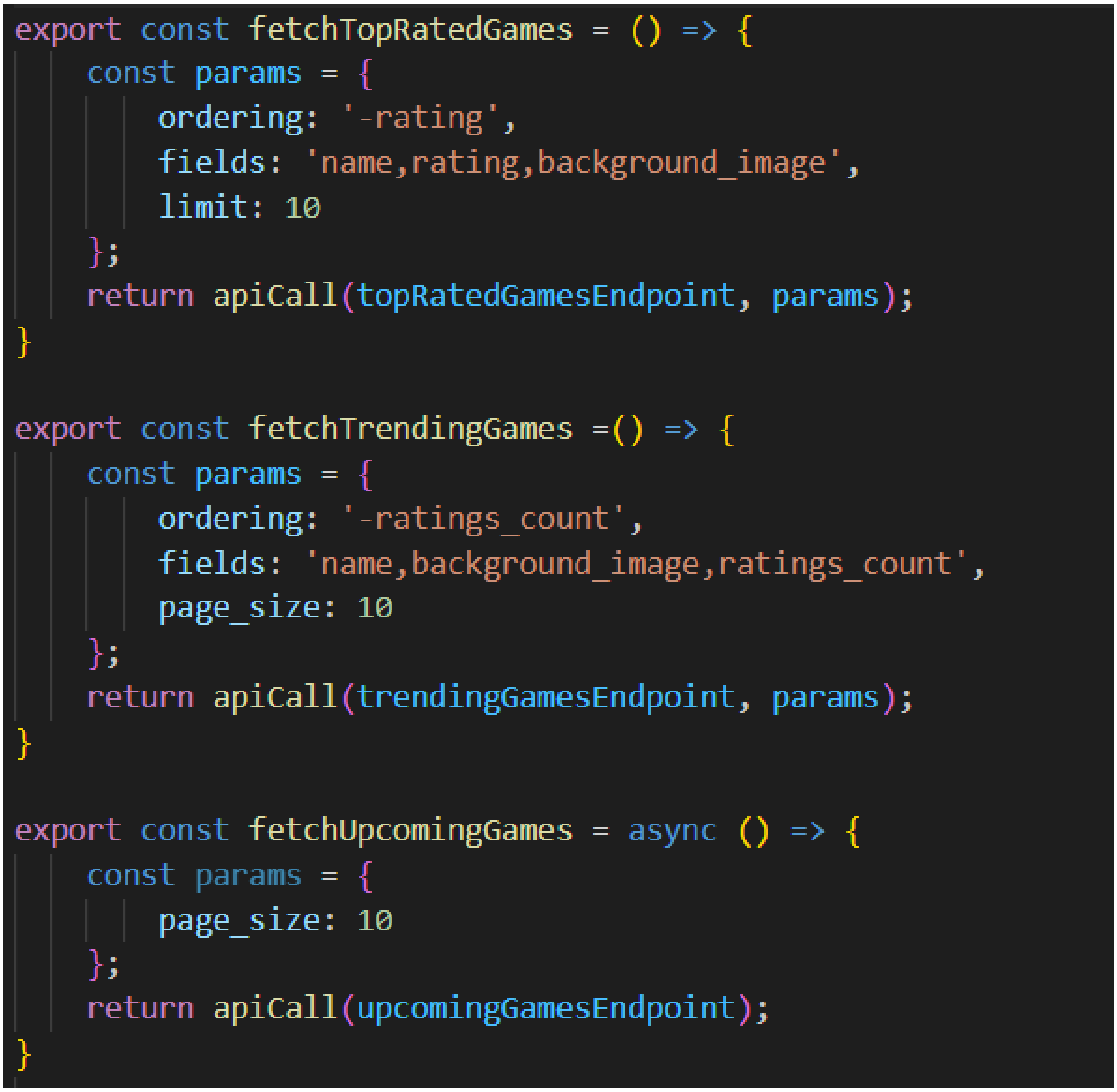
The apiCall function is a generic method to perform GET requests to various API endpoints. It accepts two parameters: endpoint for the API URL and params for any query parameters.

1. fetchTopRatedGames: Fetches a list of the top-rated games, sorted by rating.
2. fetchTrendingGames: Retrieves a list of trending games, sorted by the count of ratings.
3. fetchUpcomingGames: Acquires a list of upcoming games.



```
const apiCall = async (endpoint, params) =>{
    const options ={
        method :'GET',
        url: endpoint,
        params : params? params: {}
    }

    try{
        const response = await axios.request(options);
        return response.data;

    }catch(error){
        console.log('error:', error);
        return{}
    }
}
```

Figure 26: apiCall Function

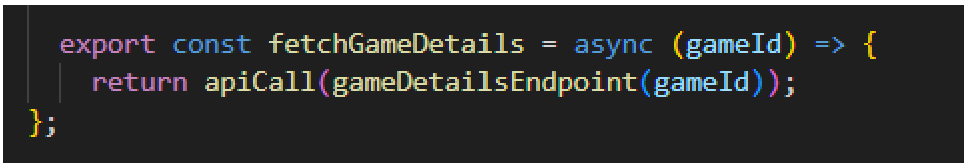


```
export const fetchTopRatedGames = () => {
    const params = {
        ordering: '-rating',
        fields: 'name,rating,background_image',
        limit: 10
    };
    return apiCall(topRatedGamesEndpoint, params);
}

export const fetchTrendingGames =() => {
    const params = {
        ordering: '-ratings_count',
        fields: 'name,background_image,ratings_count',
        page_size: 10
    };
    return apiCall(trendingGamesEndpoint, params);
}

export const fetchUpcomingGames = async () => {
    const params = {
        page_size: 10
    };
    return apiCall(upcomingGamesEndpoint);
}
```

Figure 27: Api calls for Top Rated, Trending and Upcoming games

4.6.2 Fetch Game Details

fetchGameDetails is a function used in the PixelRatings app to retrieve detailed information about a specific game by its unique identifier.

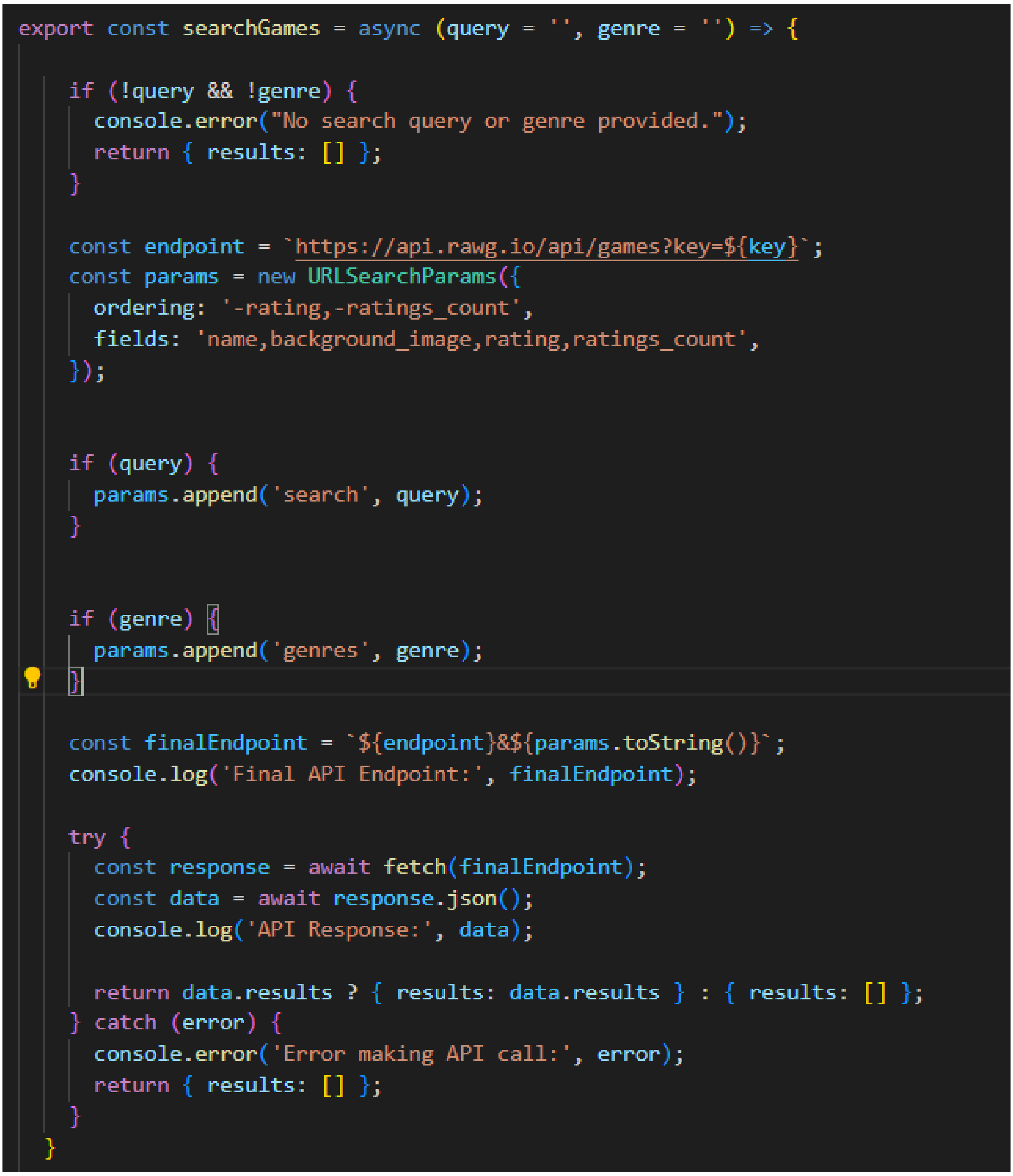


```
  export const fetchGameDetails = async (gameId) => {
    return apiCall(gameDetailsEndpoint(gameId));
};
```

Figure 28: Fetch Game Detail

4.6.3 Search Game

The searchGames function is designed to search the RAWG Video Games Database for games based on user-defined queries or genres.

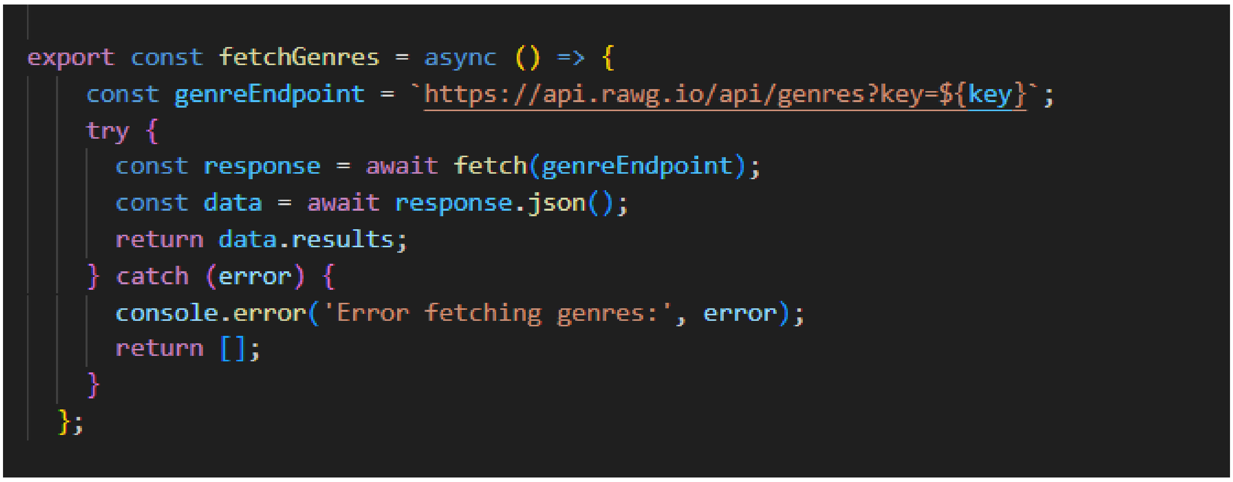


```javascript
export const searchGames = async (query = '', genre = '') => {

  if (!query && !genre) {
    console.error("No search query or genre provided.");
    return { results: [] };
  }

  const endpoint = `https://api.rawg.io/api/games?key=${key}`;
  const params = new URLSearchParams({
    ordering: '-rating,-ratings_count',
    fields: 'name,background_image,rating,ratings_count',
  });


  if (query) {
    params.append('search', query);
  }


  if (genre) {
    params.append('genres', genre);
  }

  const finalEndpoint = `${endpoint}&${params.toString()}`;
  console.log('Final API Endpoint:', finalEndpoint);

  try {
    const response = await fetch(finalEndpoint);
    const data = await response.json();
    console.log('API Response:', data);

    return data.results ? { results: data.results } : { results: [] };
  } catch (error) {
    console.error('Error making API call:', error);
    return { results: [] };
  }
}
```

Figure 29: Search call

4.6.4 Fetch Genres

fetchGenres is a function utilized within the PixelRatings app to retrieve a list of video game genres from the RAWG Video Games Database API.



```
export const fetchGenres = async () => {
    const genreEndpoint = `https://api.rawg.io/api/genres?key=${key}`;
    try {
      const response = await fetch(genreEndpoint);
      const data = await response.json();
      return data.results;
    } catch (error) {
      console.error('Error fetching genres:', error);
      return [];
    }
  };
```

Figure 30: Fetch Genres

4.7 Testing

4.7.1 Unit Testing

Sidebar Component:

- Purpose: To test the rendering and navigation functionality of the Sidebar component.
- Tests:
    - Rendering of Home, Profile, and About text.
    - Navigation to the Home, Profile, LoginPage, and Settings screens.

Game List Component:

- Purpose: To test the GameList component's rendering and navigation.
- Tests:
    - Correct rendering of titles and game data.
    - Navigation to the AllGames and individual Game screens.

TrendingGames Component:

- Purpose: To test the rendering of the TrendingGames component.
- Tests:
    - Rendering of the Trending text.

Register Component:

- Purpose: To test the Register screen's functionality including user input handling and navigation.
- Tests:
    - Correct rendering of input fields.
    - State updates on input changes.
    - Navigation back to the Login screen.

ForgotPassword Component:

- Purpose: To test the functionality of the ForgotPassword screen.
- Tests:
    - Rendering of the email input and buttons.
    - Email input state updates.
    - Handling of password reset email sending.
    - Navigation back to the Login screen.

UserComment Component:

- Purpose: To test the UserComment screen's functionality.
- Tests:
    - Rendering of the order button.
    - Toggling order on button press.
    - Navigation back to the previous screen.

FavoriteGameScreen Component:

- Purpose: To test the FavoriteGameScreen component.
- Tests:
    - Rendering of the Favorite Games title and game items.
    - Navigation back to the previous screen.

LoginPage Component:

- Purpose: To test the LoginPage's functionality including user authentication and navigation.
- Tests:
    - Rendering of input fields and buttons.
    - State updates on username and password input.
    - Navigation to Register and Forgot Password screens.
    - Handling of user login.

fetchFavoriteGames Function:

- Purpose: To test the fetchFavoriteGames function from Firebase service.
- Tests:
    - Fetching favorite games when the user document exists with favorite games.

Favorite Games Functions:

- Purpose: To test functions related to favorite games management (currently commented out in the test file).
- Tests:
    - Adding and removing a game from favorites.
    - Checking if a game is favorited by a user.

4.7.2 API Testing

fetchTopRatedGames Function:

- Purpose: Tests fetching top-rated games from the API.
- Scenarios:
    - Successful fetching of top-rated games.
    - Handling API failure by returning an empty object.

fetchTrendingGames Function:

- Purpose: Tests fetching trending games based on the number of ratings.
- Scenarios:
    - Successful fetching of trending games.
    - Handling API failure by returning an empty object.

fetchUpcomingGames Function:

- Purpose: Tests fetching upcoming games from the API.
- Scenarios:
    - Successful fetching of upcoming games.
    - Handling API failure by returning an empty object.

fetchGameDetails Function:

- Purpose: Tests fetching detailed information about a specific game.
- Scenarios:
    - Successful fetching of game details.
    - Handling API failure by returning an empty object.

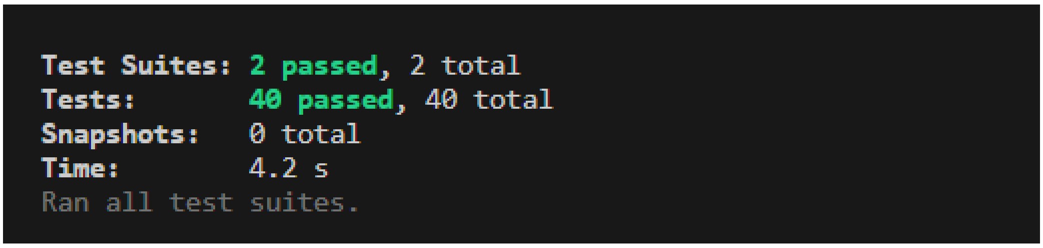
searchGames Function:

- Purpose: Tests the functionality of searching for games with a specific query or genre.
- Scenarios:

    - Performing a search with a genre.
    - Returning empty results for an empty query and genre.
    - Handling API failure by returning an empty array.

fetchGenres Function:

- Purpose: Tests fetching a list of game genres from the API.
- Scenarios:
    - Successful fetching of genres.
    - Handling API failure by returning an empty array.



```
Test Suites: 2 passed, 2 total
Tests:       40 passed, 40 total
Snapshots:   0 total
Time:        4.2 s
Ran all test suites.
```

Figure 31: Running Tests

Chapter 5
Conclusion

The development of PixelRatings, an innovative mobile application for game reviews and ratings, marks a significant milestone in my academic and professional journey. This project has not only showcased my commitment and skills in software development, but it has also greatly improved my knowledge and abilities in mobile app development, especially with React Native and JavaScript. The process of bringing PixelRatings to life has provided me with invaluable insights into the complexities of app design, user experience, and data management, thereby enriching my knowledge and shaping my future career path in the realm of technology

Through the design and development of PixelRatings, I have gained a profound understanding of the intricacies of mobile app development. This journey has allowed me to delve deeper into React Native, a framework for building native apps using React. My proficiency in JavaScript, a crucial language for web and mobile app development, has been notably enhanced. These skills are invaluable in the rapidly evolving field of technology and will undoubtedly aid in future projects and career opportunities.

PixelRatings serves as a powerful platform where gaming enthusiasts can converge to share their experiences, rate games, and stay updated with the latest trends in the gaming world. By integrating features like user registration, game rating, and review functionalities, the app offers a comprehensive and user-friendly interface for users.

Reflecting on the project, I recognize several areas for potential improvements and future enhancements for PixelRatings:

- Social Integration: Integrating social media platforms would allow users to share their reviews and ratings, thus enhancing user engagement and expanding the app's reach.
- Advanced Recommendation System: Implementing a more sophisticated algorithm that suggests games to users based on their rating history and preferences could greatly personalize the user experience.
- Enhanced Security Measures: Given the sensitivity of user data, continuously updating security protocols to safeguard personal information is imperative.
- User Interface and User Experience Improvements: Regular updates based on user feedback can help in refining the UI/UX to meet the evolving preferences of the users.

Chapter 6

Bibliography

[1] Visual Studio Code

https://code.visualstudio.com/

[2] Android Studio

https://developer.android.com/studio/

[3] Git

https://git-scm.com/

[4] React Native

https://reactnative.dev

[5] RAWG Video Game Database API

https://api.rawg.io/docs/

[6] Github repository

https://github.com/nazdemirsoyy/Thesis

[7] Firebase documentation

https://firebase.google.com/docs

[8] Expo

https://expo.dev