# Penguin Details Checker

A Mobile Application Realized with React Native (üK 335)

# What are we Solving?

- Researchers used to carry pen and paper in the arctic, researching penguins
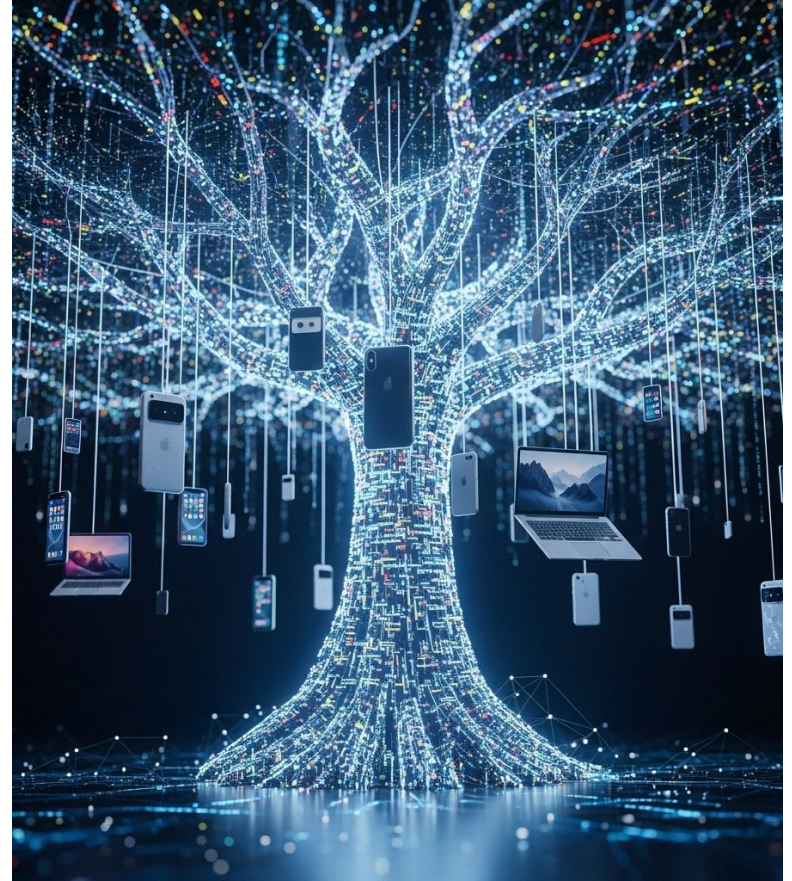- Paper gets wet.
- Pens Freeze

# Our Solution!

- A Mobile Application to track penguin species like never before
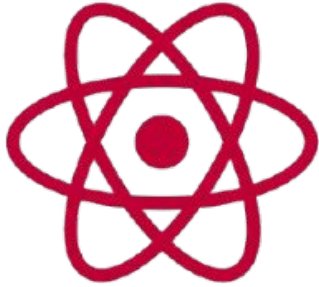- Access detailed information on demand
- Worldwide accessibility

# Let's talk **Technology**

# Our Approach:
# Hybrid App Development

- Best in Class Performance / Cost Tradeoff
- Dev team's familiarity with React
- Fast Iteration speed due to Expo

# Core Tech Stack


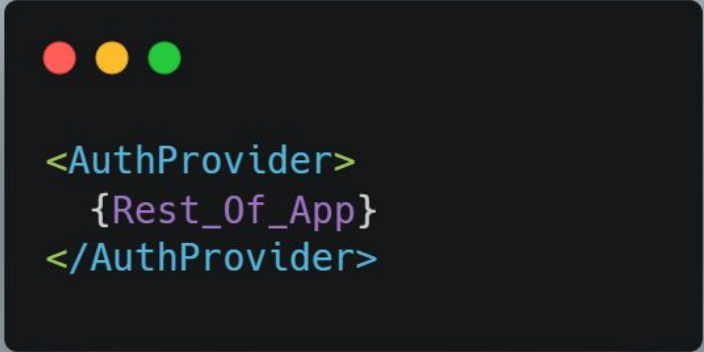React Native
Supported by
∧ Expo

# Additional Libraries


REACT QUERY


React Navigation


paper

# Implementation Highlights
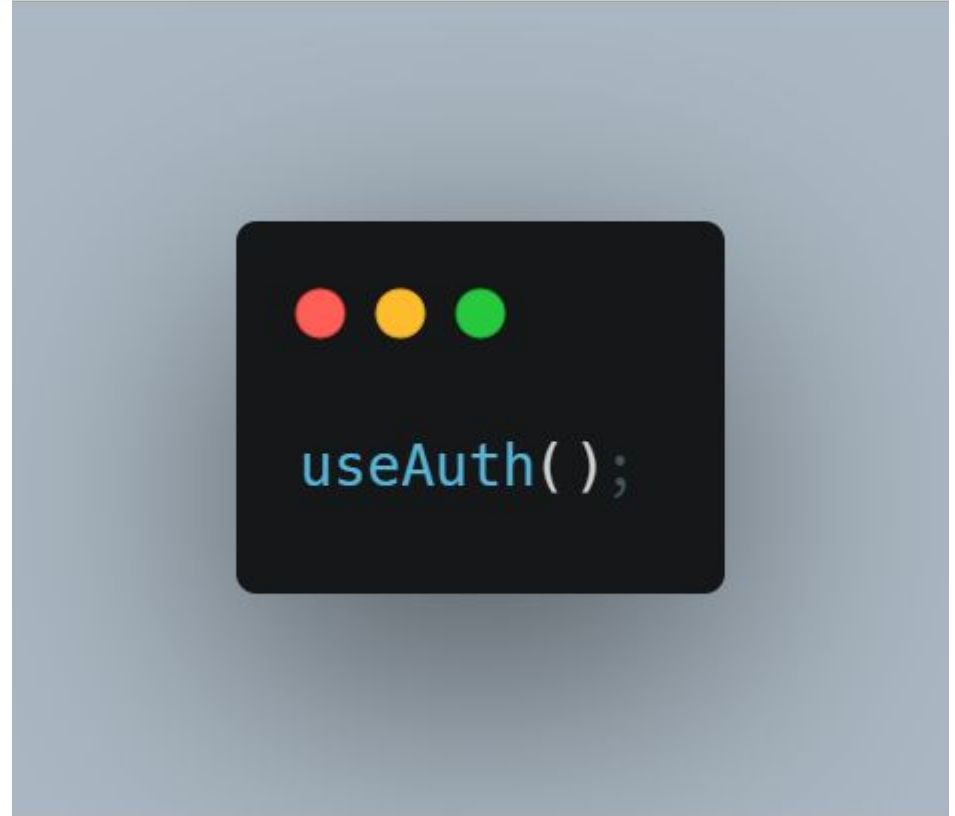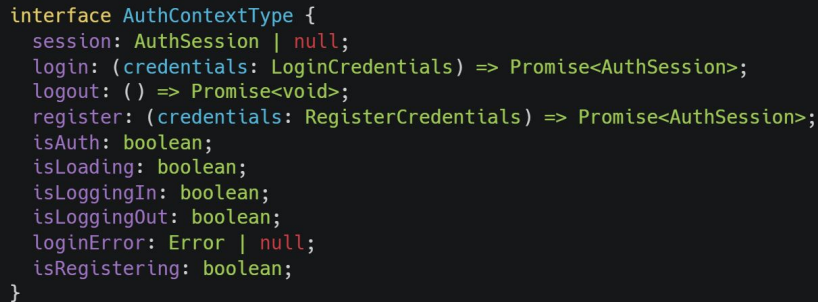
- **Custom useAuth Hook & Provider**
- Custom use…Penguin Hooks

# Implementation Highlights

- **Custom useAuth Hook & Provider**
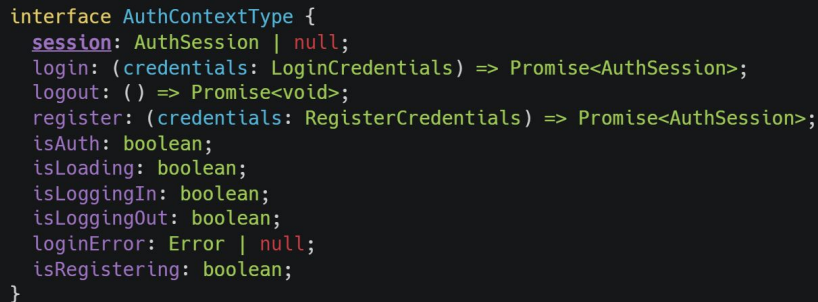- Custom use…Penguin Hooks

# Implementation Highlights

- **Custom useAuth Hook & Provider**
- Custom use…Penguin Hooks

```
interface AuthContextType {
  session: AuthSession | null;
  login: (credentials: LoginCredentials) => Promise<AuthSession>;
  logout: () => Promise<void>;
  register: (credentials: RegisterCredentials) => Promise<AuthSession>;
  isAuth: boolean;
  isLoading: boolean;
  isLoggingIn: boolean;
  isLoggingOut: boolean;
  loginError: Error | null;
  isRegistering: boolean;
}
```

# Implementation Highlights

- **Custom useAuth Hook & Provider**
- Custom use…Penguin Hooks

```
interface AuthContextType {
  session: AuthSession | null;
  login: (credentials: LoginCredentials) => Promise<AuthSession>;
  logout: () => Promise<void>;
  register: (credentials: RegisterCredentials) => Promise<AuthSession>;
  isAuth: boolean;
  isLoading: boolean;
  isLoggingIn: boolean;
  isLoggingOut: boolean;
  loginError: Error | null;
  isRegistering: boolean;
}
```
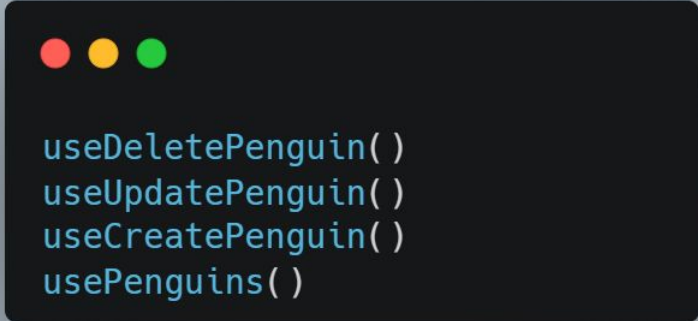
# Implementation Highlights

- **Custom useAuth Hook & Provider**
- Custom use…Penguin Hooks

```typescript
interface AuthSession {
  accessToken: string;
  user: User;
}
interface User {
  name: string | null;
  email: string;
  id: number;
}
```
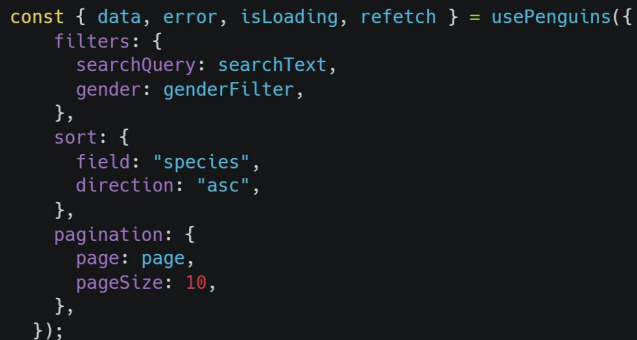
# Implementation Highlights

- Custom useAuth Hook & Provider
- **Custom use…Penguin Hooks**



```
useDeletePenguin()
useUpdatePenguin()
useCreatePenguin()
usePenguins()
```

# Implementation Highlights

- Custom useAuth Hook & Provider
- **Custom use…Penguin Hooks**

```
const { data, error, isLoading, refetch } = usePenguins({
    filters: {
        searchQuery: searchText,
        gender: genderFilter,
    },
    sort: {
        field: "species",
        direction: "asc",
    },
    pagination: {
        page: page,
        pageSize: 10,
    },
});
```

# Implementation Highlights

- Custom useAuth Hook & Provider
- **Custom use…Penguin Hooks**

```
const { data, error, isLoading, refetch }
= usePenguins()
```

# Security

- Backend Protected
- JWT Authenticated
- Expo Secure Store to protect from Local breaches.

# Privacy

- Zero Logs Policy
- No Telemetry

# Application Quality Assurance

---

- Taking advantage of our reliable Quality Assurance Engineers, we ran through manual tests
- All Tests are Passing

## Test Case Summary

| | |
|---|---|
| Happy Path Login | ✅ PASS |
| Error Path Login | ✅ PASS |
| Create Item | ✅ PASS |
| Read Item | ✅ PASS |
| Happy Path Update | ✅ PASS |
| Error Path Update | ✅ PASS |
| Happy Path Delete | ✅ PASS |

🚀 Live Demo Time 🚀

# Scenario

- Dr. Evelyn Reed, Penguin Researcher
- Discovers new penguin
- Create new entry

# Scenario

- Student researcher sees a penguin has been misspelled
- Student finds the Penguin
- Student edits the Penguin

# Scenario

- Sadly this penguin died…
- And we have to delete it

# Recap

- Worldwide Penguin Tracking
- Available on all Major Mobile Devices

# Next Steps

- Add Regression Tests
- Add Pictures of Penguins
- Penguin Location tracking

# Questions?