# OPSC6312 PART 1 RESEACH

**Sky Pals**

## INTRODUCTION:

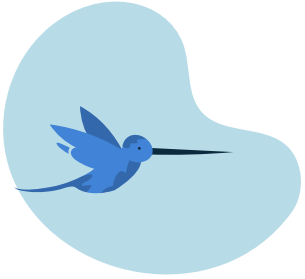
This research project focuses on exploring the impact of technology on the popular outdoor activity of bird watching. It examines three prominent bird watching mobile applications: eBird, Birda, and Picture Bird, which have gained popularity among bird enthusiasts worldwide. The research aims to assess these apps in terms of their user interfaces, bird identification tools, geolocation features, citizen science contributions, and community engagement capabilities. The goal is to provide valuable insights for both beginners and experienced bird watchers and offer guidance for app developers interested in the intersection of technology and wildlife observation.

## RESEARCH ON EACH APP:

### eBird

#### Overview

eBird Mobile makes it easy to record the birds you see in the field, and seamlessly link these observations with eBird--a global online database of bird records used by hundreds of thousands of birders around the world. This free resource makes it easy to keep track of what you see, while making your data openly available for scientific research, education, and conservation. eBird Mobile is the only app that passes information directly from the iOS device to your Bird account on the web.



### Strengths

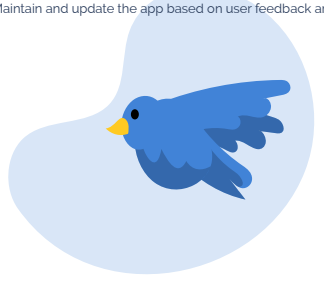
1. Global Birdwatching Community: eBird is a global platform that brings together birdwatchers from around the world, facilitating the exchange of bird sightings and creating a robust database.
2. Scientific Credibility: Backed by the Cornell Lab of Ornithology, eBird is known for its scientific rigor and data quality, making it a trusted resource for researchers and conservationists.
3. User-Friendly: The app and website are user-friendly, appealing to both beginners and experts in birdwatching, making it accessible to a wide audience.
4. Real-Time Data: eBird provides real-time bird sighting data, helping users plan birdwatching trips and discover recent sightings in their area.
.5 Contribution to Conservation: The platform plays a significant role in bird conservation efforts by collecting data on bird populations and helping identify areas that need protection.

### Weaknesses:

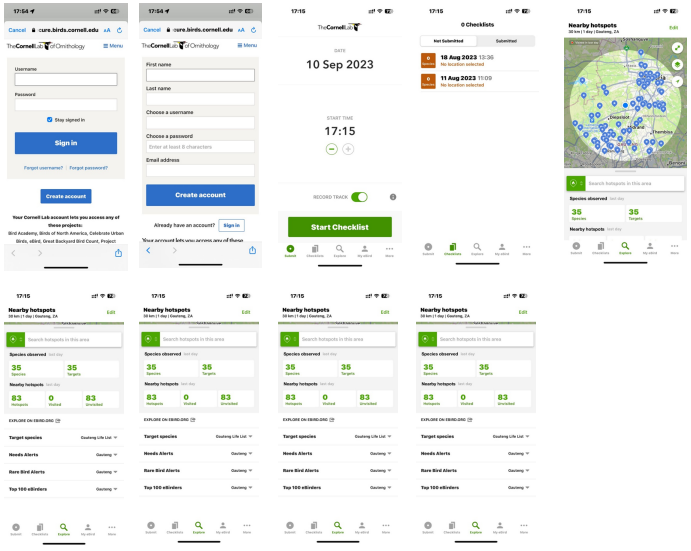
1.Data Accuracy: The accuracy of eBird data depends on the expertise of users, potentially leading to inaccuracies in species identification and location.
2.Data Bias: Certain areas and species may be overrepresented due to high birdwatching activity, while underrepresented regions may lack comprehensive data.
3.Privacy Concerns: Users are required to disclose their location, raising privacy concerns for some individuals who may not want to share their whereabouts.
4. Language Barrier: eBird primarily operates in English, limiting accessibility for non English speakers and potentially excluding valuable contributions from diverse regions.
5.Dependency on Connectivity: The app relies on internet connectivity, which can be challenging in remote areas, hindering birdwatchers' ability to use it effectively.
6. Does not have logout button.

### Implementation:

1.Project Setup: Start by creating a new Android project in Android Studio, defining basic project settings.
2.User Interface Design: Use Android Studio's visual UI designer to create the app's interface, including screens for bird sightings and maps.
3.Programming Logic: Write Java or Kotlin code to implement core app features and functionalities.
4.Data Storage: Choose a data storage method like SQLite, Room, or Firebase for managing bird sighting data.
5. Maps Integration: Incorporate Google Maps API or another mapping service for displaying birding hotspots.
6.Network Communication: Use libraries like Retrofit or Volley to communicate with eBird servers via HTTP requests.
7. User Authentication: Implement user authentication, possibly using Firebase Authentication or OAuth, for user accounts.
8. Real-Time Updates: Ensure real-time updates of recent bird sightings, possibly using Firebase Realtime Database.
9. Testing and Debugging: Thoroughly test and debug the app using Android Studio's tools on emulators and devices.
10. Optimization and Performance: Optimize app performance, addressing memory management and responsiveness.
11. Localization: Support multiple languages and regions for a global audience.
12. Security: Implement security measures, including secure data communication and user privacy considerations.
13. Deployment: Prepare the app for release, generate signed APKs, and publish on app stores.
14. Continuous Improvement: Maintain and update the app based on user feedback and evolving requirements.



### Screenshots



## Birda

### Overview

Birda is a free bird app & community for anyone curious about nature and birds, no matter your level of knowledge or previous bird watching experience. We want to help you enjoy birds, so you are inspired to fight to protect them. Regardless of how you spell birdwatching (or bird watching) if you like fun & nature, Birda is for you!

### Strengths

1. User interface is easy and friendly to use.
2. App has integrated groups and challenges
3. App contains a discover page that has guide lines that assist the users
4. App has an achievement system and summary that contains all the events, birds and places the users have been.

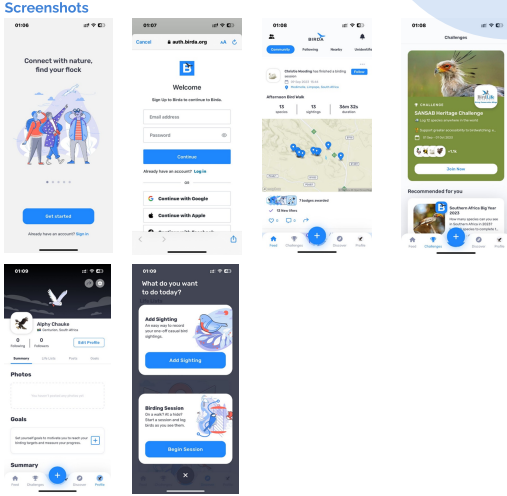
### Weaknesses:

1. Dependency on Connectivity: The app relies on internet connectivity, which can be challenging in remote areas, hindering birdwatchers' ability to use it effectively.

### Implementation:

1.Project Setup: Start by creating a new Android project in Android Studio, defining basic project settings.
2.User Interface Design: Use Android Studio's visual UI designer to create the app's interface, including screens for bird sightings and maps.
3.Programming Logic: Write Java or Kotlin code to implement core app features and functionalities.
4.Data Storage: Choose a data storage method like SQLite, Room, or Firebase for managing bird sighting data.
5. Maps Integration: Incorporate Google Maps API or another mapping service for displaying birding hotspots.
6.Network Communication: Use libraries like Retrofit or Volley to communicate with eBird servers via HTTP requests.
7. User Authentication: Implement user authentication, possibly using Firebase Authentication or OAuth, for user accounts.
8. Real-Time Updates: Ensure real-time updates of recent bird sightings, possibly using Firebase Realtime Database.
9. Testing and Debugging: Thoroughly test and debug the app using Android Studio's tools on emulators and devices.
10. Optimization and Performance: Optimize app performance, addressing memory management and responsiveness.
11. Localization: Support multiple languages and regions for a global audience.
12. Security: Implement security measures, including secure data communication and user privacy considerations.
13. Deployment: Prepare the app for release, generate signed APKs, and publish on app stores.
14. Continuous Improvement: Maintain and update the app based on user feedback and evolving requirements.

### Screenshots




## Picture Bird

### Overview

Picture Bird is a useful and smart bird identifier app that can recognize any bird species by photo or sound. Simply take upload a picture of a bird or record a bird's sound, and you can get everything you want to know about it.

### Strengths

1. User-Friendly: The app and website are user-friendly, appealing to both beginners and experts in birdwatching, making it accessible to a wide audience.
2. Ai: The app has Ai technology embedded in it that assist in identifying birds.

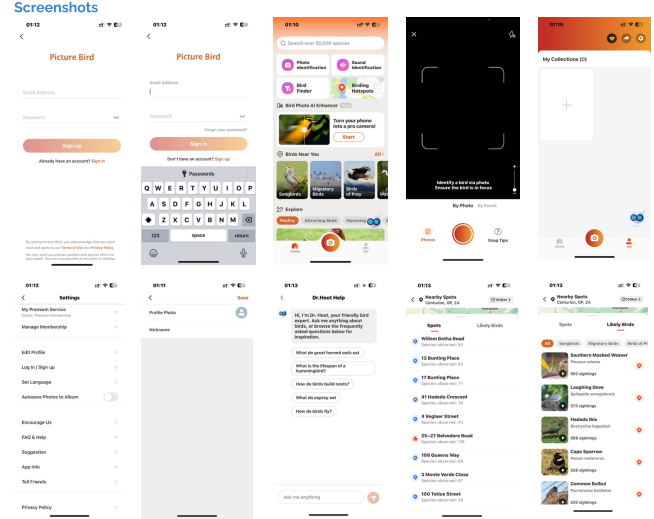
### Weaknesses:

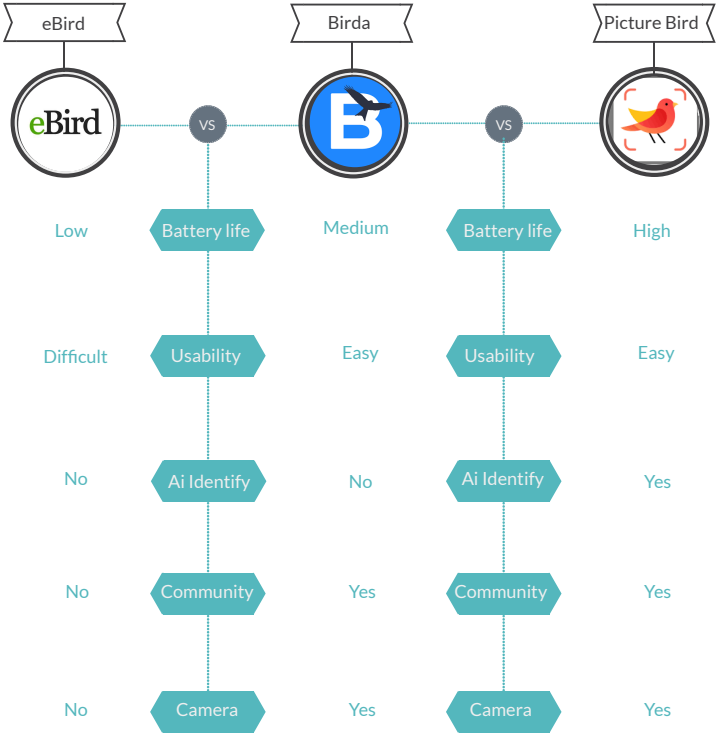
1. Dependency on Connectivity: The app relies on internet connectivity, which can be challenging in remote areas, hindering birdwatchers' ability to use it effectively.

### Implementation:

1.Project Setup: Start by creating a new Android project in Android Studio, defining basic project settings.
2.User Interface Design: Use Android Studio's visual UI designer to create the app's interface, including screens for bird sightings and maps.
3.Programming Logic: Write Java or Kotlin code to implement core app features and functionalities.
4.Data Storage: Choose a data storage method like SQLite, Room, or Firebase for managing bird sighting data.
5. Maps Integration: Incorporate Google Maps API or another mapping service for displaying birding hotspots.
6.Network Communication: Use libraries like Retrofit or Volley to communicate with eBird servers via HTTP requests.
7. User Authentication: Implement user authentication, possibly using Firebase Authentication or OAuth, for user accounts.
8. Real-Time Updates: Ensure real-time updates of recent bird sightings, possibly using Firebase Realtime Database.
9. Testing and Debugging: Thoroughly test and debug the app using Android Studio's tools on emulators and devices.
10. Optimization and Performance: Optimize app performance, addressing memory management and responsiveness.
11. Localization: Support multiple languages and regions for a global audience.
12. Security: Implement security measures, including secure data communication and user privacy considerations.
13. Deployment: Prepare the app for release, generate signed APKs, and publish on app stores.
14. Continuous Improvement: Maintain and update the app based on user feedback and evolving requirements.

### Screenshots






**Best features of all the apps that we want to use in our final app:**

1. The Chatbot from Picture Bird app.
2. The Ai photo identification from Picture Bird app.
3. The Sound recognition from Picture Bird app.
4. The Map from eBird App.
5. The Community page from Birda.

**Conclusion:**

In conclusion, the world of bird watching has embraced technology, and bird watching apps have become valuable companions for birders of all levels of expertise. This research project aims to provide a comprehensive comparison of three prominent bird watching apps, shedding light on their respective strengths and weaknesses. By doing so, we hope to empower bird enthusiasts with the information they need to make informed decisions about which app best suits their birding adventures. Additionally, we aim to contribute to the broader discourse on the intersection of technology, wildlife observation, and citizen science in the digital age.

Made with VISME