

# Arcadia

Github - <https://github.com/kapilkumar2001/Arcadia-Auction>

Release Apk - <https://github.com/kapilkumar2001/Arcadia-Auction/releases>

## Introduction

This project was made by a team of 3 people. Our institute has frequent tournaments for CS GO. As an active player and spectator in them we felt that it lacked a place to see all the data and currently it was managed by google sheets which wasn't the best way to do it due to people misplacing links and asking the organisers for updates. We tried to make a one stop destination for all things related to the league/

## Features

We divided the app mainly into two different parts which were separated by the login credentials.

### Admin

- Admin can add teams on the App which are registered for the tournament.
- Admin can organise Live Auction for the teams and players on the App itself.
- Admin gets live updates about the Auction during live Auction.
- Admin can add match schedule on the App which will be displayed to the users.
- Admin can announce any updates throughout the App itself.

### User

- Players can Register for the tournament through the App itself.
- Users can Sign-in Anonymously.
- Users can see the Match Schedule on the App.
- Users can watch Match details (including time, score, winner team, MVP etc)
- Users can see the points table on the App.
- Users will get announcements from Organisers in case of match rescheduling or any important update on the App.

## Technologies and platforms used

- Flutter
- Provider
- Dart
- Firebase

## My Contributions

I Led the complete development of the app from the UI to the backend, I was mainly planning user flows and interactions , how the user reacts to what is visible on his screen, what buttons or navigation feels better.

State management is one of the things I have read and done a lot. We went ahead with provider as a state management - due to its simple nature and also considering that the other two members barely knew about working with a medium scaled app.

## Challenges Faced

We felt that things like making providers depend on each other is not the most easy task. Our business logic was a part of the app which made it worse to maintain and update the app.

State Management as a whole is something which needs to be addressed with a better architecture and a more robust solution like BLoC.

## Future Scope

We plan to make a new version of this app which supports multiple leagues and also using BLoC as a state management solution. Some more features which can be implemented are :

- Subscribe to match updates of a particular team.
- Live score updates with the use of API's provided by the CSGO server
- Player profile from steam's API
- Support for multiple Leagues.
- Moving to a custom backend solution.

## Note

- The app was mainly developed in a period of 7 days
- Actual players of the tournament used this app.
- This app was appreciated and had live data throughout the period of the tournament.

## Screenshots





