Sprint Period:

13.11.2023 - 26.11.2023

Team Overview:

- Radu (Lead Android Developer)
- Eduard (Android Developer)
- Cosmin (Android Developer)

Planned backlog items:

- Storage liberation and storage access
- UX and info screens
- UI rework
- Parallax effect

Implemented backlog items:

- folder synchronization (incomplete)

Implemented user stories:

- saving games in a user accessible folder, for easy files backup, copy-pasting and modifications (incomplete)

Items added after start:

- none

items removed before end:

- none

backlog Items pushed to future sprint:

- UX and info screens, UI rework, parallax effect
- remaining work for storage liberation and acess

User stories covered in this sprint and their acceptance criteria:

Storage liberation and access:

- As a user, I want to be able to access all of my save files from a folder I created, not the default Android/data folder which is inaccessible to most file managers out there.
- Acceptance criteria:
    - folder synchronization: The app should be able to sync folders. When a new file is added to the synced folder, it should automatically appear in the other synced folders. Similarly, changes made to a file in one folder should reflect in all other synced folders.
      Given a user has created a folder within the app for storing save files
      When the user adds a new file to the synced folder
      Then the file should automatically appear in all other synced folders and changes made to a file in one synced folder should be reflected in all other synced folders

- file accessibility: The files saved in the synced folders should be easily accessible by the user. This includes being able to open, read, and modify these files without any restrictions.
  Given a user has created a folder within the app for storing save files
  When the user attempts to access the saved files in the synced folders
  Then the user should be able to easily open, read, and modify these files without encountering any restrictions

- modifications: Users should be able to rename, delete, move, or modify files within the synced folders. Any such modifications should reflect in all other synced folders.
  Given a user has created a folder within the app for storing save files
  When the user renames, deletes, moves, or modifies files within the synced folders
  Then these modifications reflect in all other synced folders

- error handling: The app should handle errors gracefully. For instance, if a file cannot be copied due to insufficient space or lack of permissions, the app should notify the user and continue with the remaining operations.
  Given a user interacts with the app to perform file operations within the synced folders
  When an error occurs such as insufficient space or lack of permissions during copying
  Then the app notifies the user and proceeds with the remaining operations

- performance: The app should ensure that the synchronization process does not significantly impact the performance of the device or slow down other operations.
  Given the app is synchronizing folders
  When ensuring performance
  Then the synchronization process should not significantly impact the device's performance or slow down other operations

Sprint review:

From the sprint review meeting we all agreed to not use a new framework, because current devs are not comfortable with learning new technology on top of being pressured by the sprint deadlines.
Even though we finished part of the storage liberation and access feature, the project is far from an MVP point yet.

Sprint retrospective:

What went well:

- We are managing our schedule effectively.
- New android dev (Cosmin) enjoys learning the new technology.

What went not so well:

- For some, it’s difficult to learn Kotlin in such a short time.

Improvements:

- First sprint, nothing to improve here.