**Feed App Documentation**

Created by Márton Molnár

I regret to inform you that I did not write any tests, because I had a problem with Junit when using Android Studio and I did not have the time to fix it.

**Project structure**

**A képen szöveg, képernyőkép, szoftver, Operációs rendszer látható

Előfordulhat, hogy az AI által létrehozott tartalom helytelen.**

**Classes**

* **Product:** Holds the data of the fetched products
* **ProductRepository:** Repository interface
* **ProductRepositoryImpl:** Implements the above interface. The API path, limit and skip variables can be set with the primary constructor Has the following functions:
  + **getProducts**: Returns a list of products. I chose to return a list because the limit can be set higher than one, in which case the function fetches multiple products at once from the API.
  + **resetRepo**: Resets the skip variable, which means that on a new start command, the fetching will start from the beginning.
  + **incrementIdCounter**: Increments the idCounter variable, which is needed for displaying items in a LazyColumn
  + **getIdCounter**: Returns the value of the idCounter variable.
* **ProductViewModel**: The ViewModel of tha application. Responsible for processing the commands as well as starting and stopping the coroutine to fetch data from the API. Has the following functions:
  + **startCommand**: Starts the fetching coroutine. If the app is not paused, updates the flow with the new Products.
  + **stopCommand**: Stops the fetching and resets the repo.
  + **pauseCommand**: Pauses the updating of the flow.
  + **resumeCommand**: Resumes the updating of the flow.
  + **processCommand:** Processes the incoming commands and calls the correct functions.
  + **addCommand**: Adds a command to the flow,
* **ProductViewModelFactory**: Needed for passing the Repo to the ViewModel constructor
* **MainActivity**: The only activity of the application. Responsible for creating the repo and the ViewModel instances.
* **MainScreen**: The main composable of the application. Contains the FeedContent, a TextField and a Button
* **FeedContent**: Displays the FeedCards inside a LazyColumn
* **FeedCard**: Display the elements of the feed.