GitHub: <link>

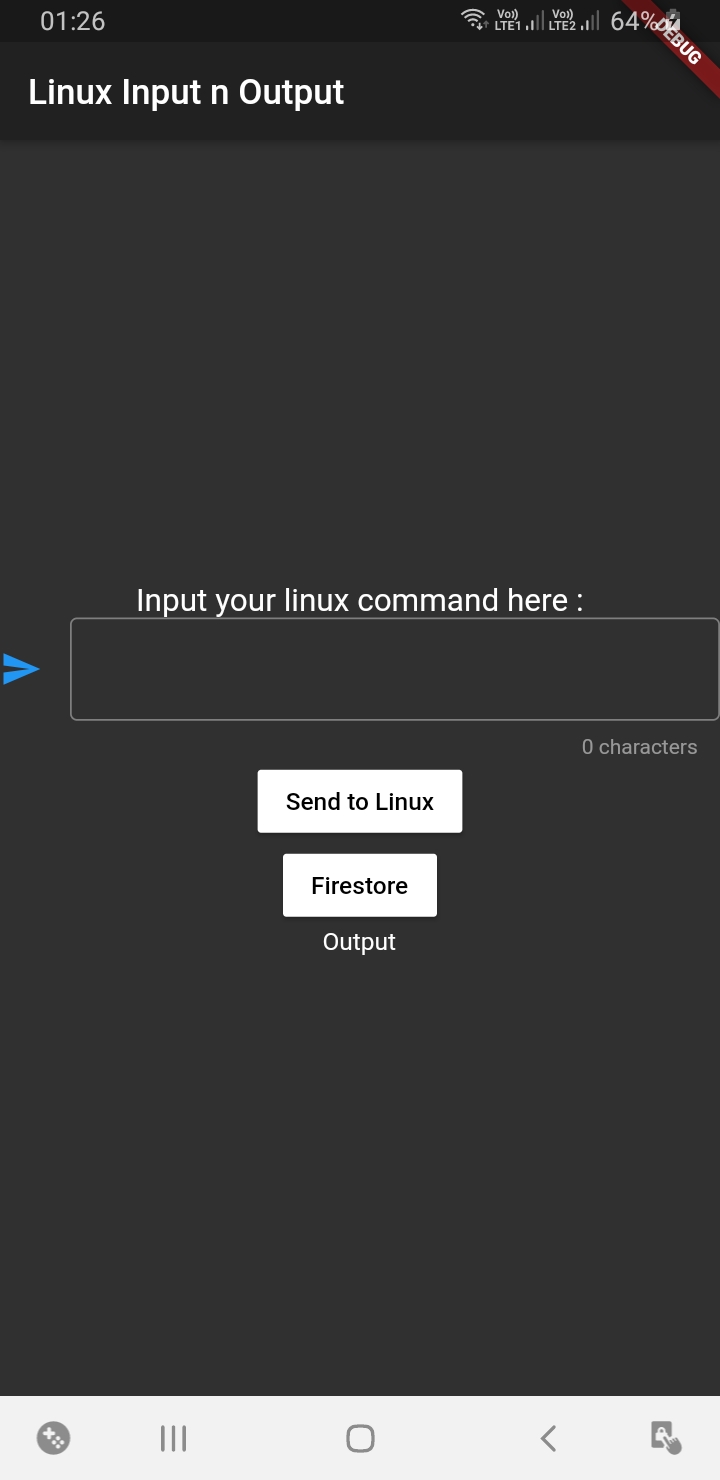
It will be a treat to all the Linux users out there if they could just operate their Linux systems at the tap of their mobile phones right?!

With this app, I tried to integrate my flutter app with linux and firestore, which also stores the outputs as received, which can be viewed in the future.

Problem Statement:

* Create an app that can run any linux command using API and the output will be saved in Firestore. From Firestore, get this output and print on screen.

First, we create a basic UI of the app as follows:



* We can create this UI with the help of column widget.
* The button are made using RaisedButton and output is shown using the Text widget only.

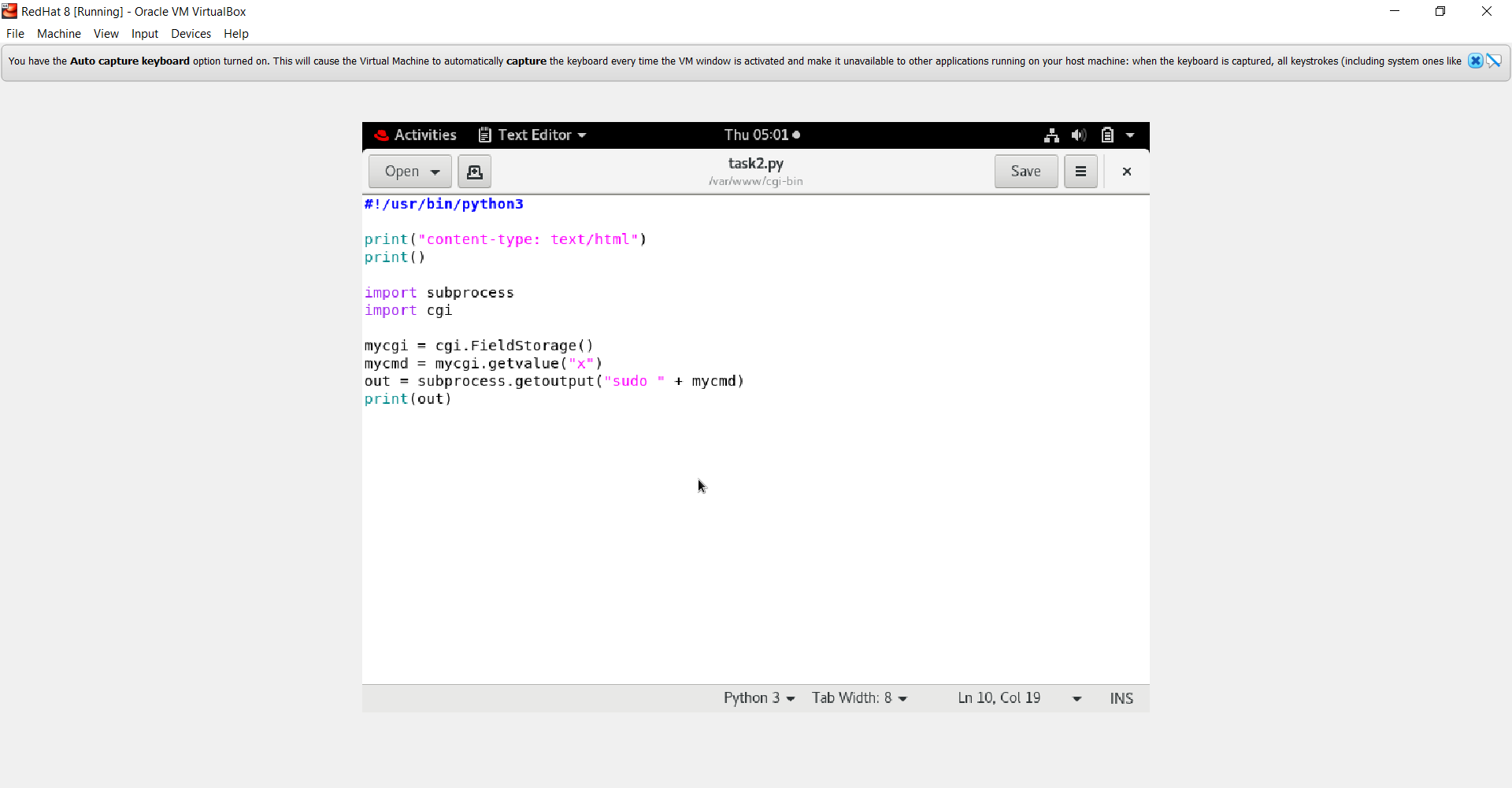
(Made the UI of Dark theme to be more appealing and soothing to the eyes!)

Now, we proceed with the main implementation of our app:

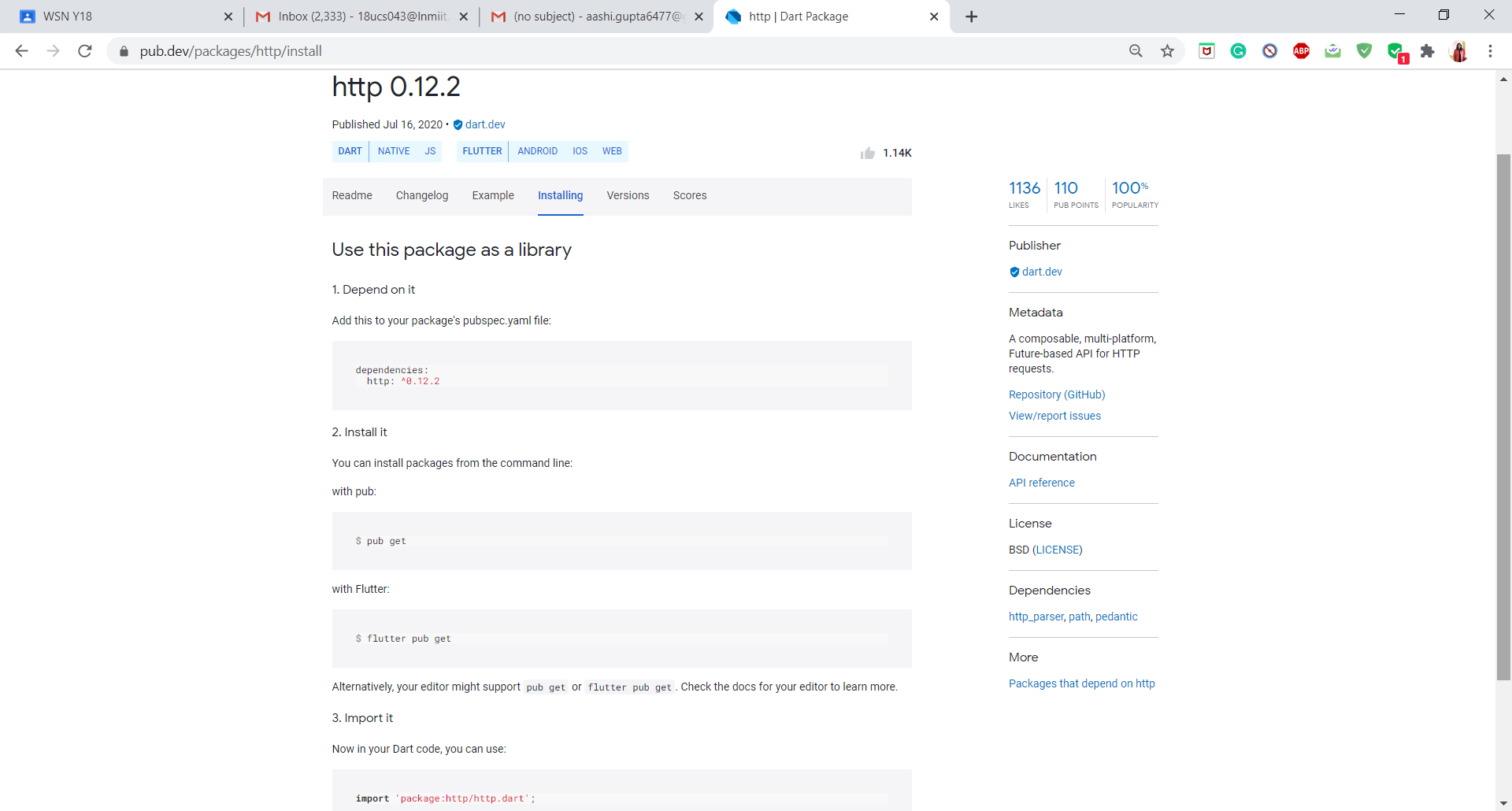
* **API to connect to Linux.**
* **Connect to Firestore from where we can either fetch our data or feed more data to it.**

**#Linux API**

* API are to be executable scripts written in the cgi-bin folder in linux.
* First go to the directory using : cd/var/www/cgi-bin
* Next, we create the following python script for our API by: gedit task2.py



* To make this file executable use: chmod +x task2.py
* To access this file from our flutter app, we need to establish a network between our app and linux server, via http protocol
* From pub.dev, we download a plugin c/a httpd and add it in our app as follows:



Then, we access our file, via a url and it returns the output from linux server by the following method created



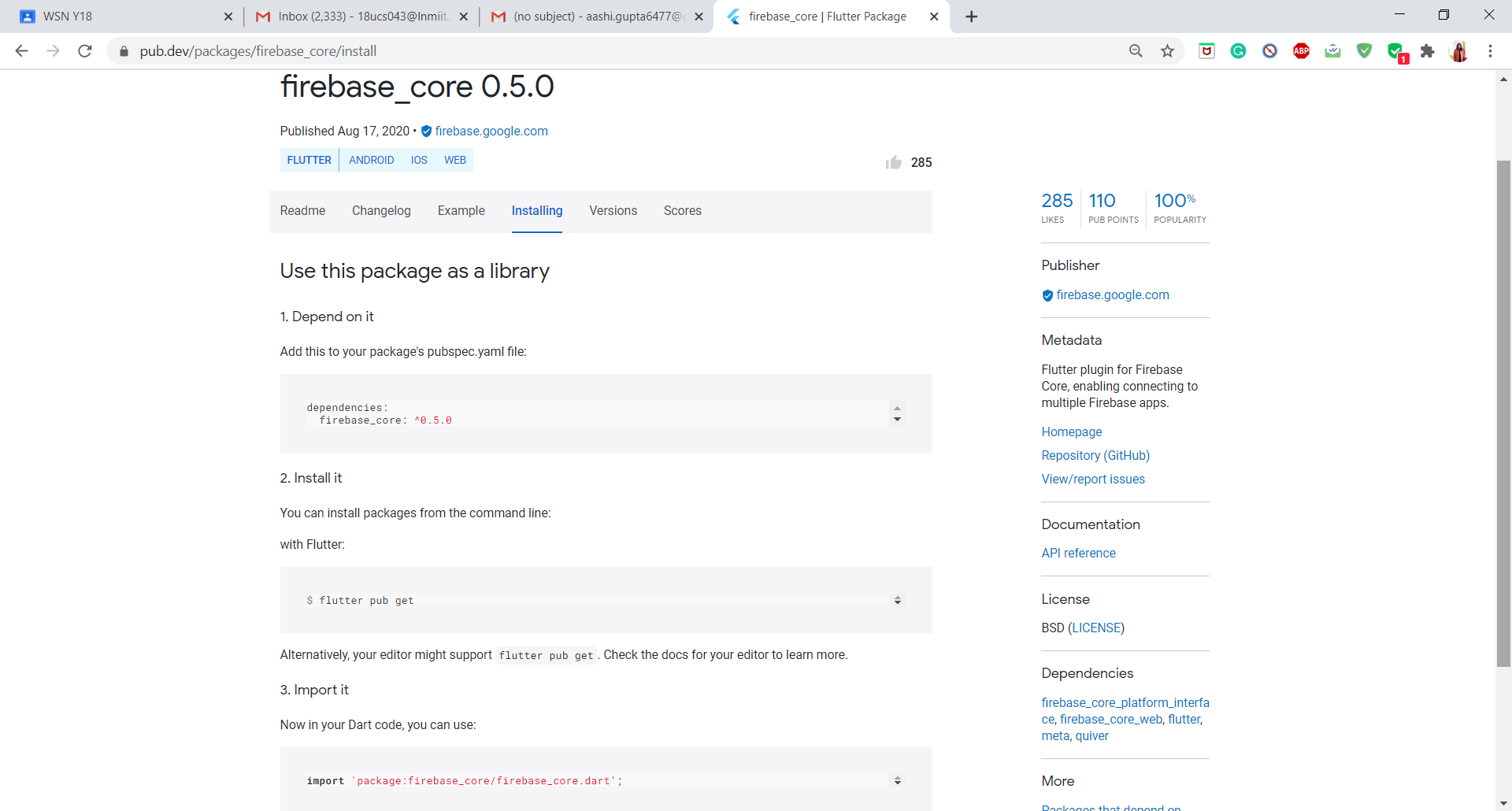
I made a provision to display the output on the screen as soon as command is executed on the Text Widget. To change its contents on the fly, we use the setState() method for the same as shown above.

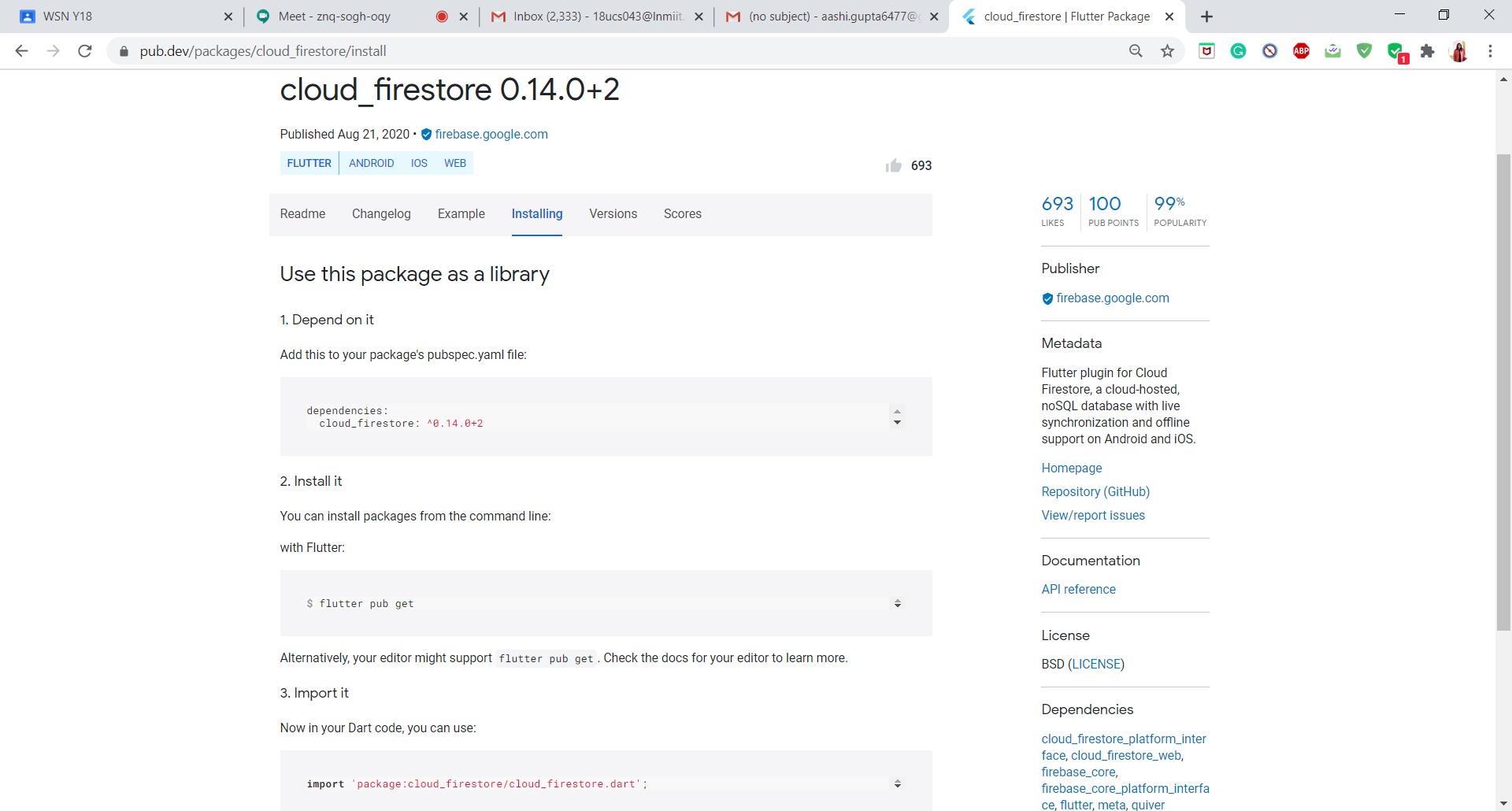
This completes our part of integrating Flutter with Linux

**#Firestore**

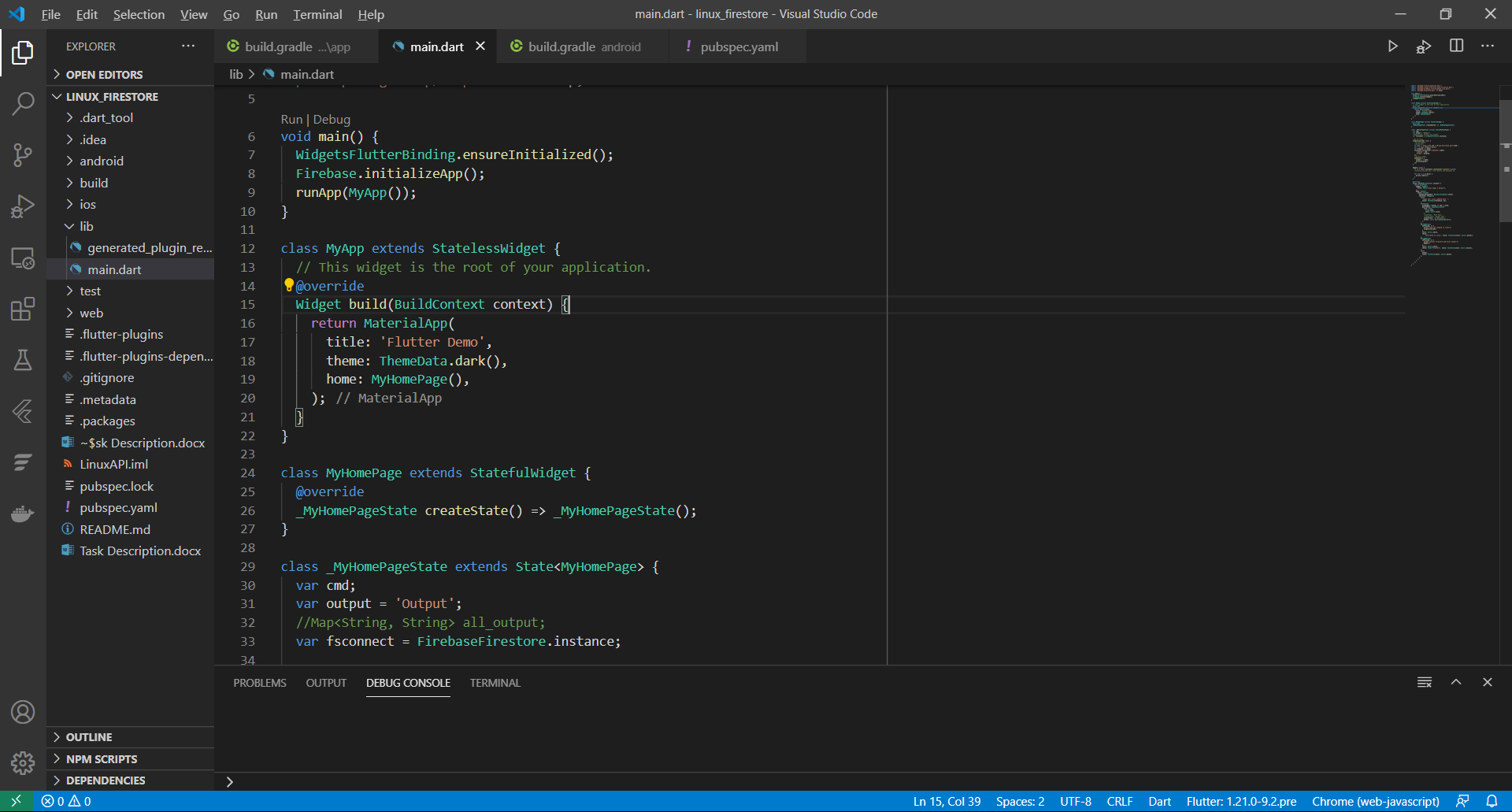
Next, comes the part of retrieving data stored in firestore:

* First of all, to connect to firestore, we need to go to console,firebase.google.com to create your account their and use the firebase android services provided by google. We thus connect to firestore as per the steps.
* But we will not be able to work on the firestore without the use of a Firestore Plugin
* Go to pub.dev and download the cloud\_firestore plugin as download it as shown in our app

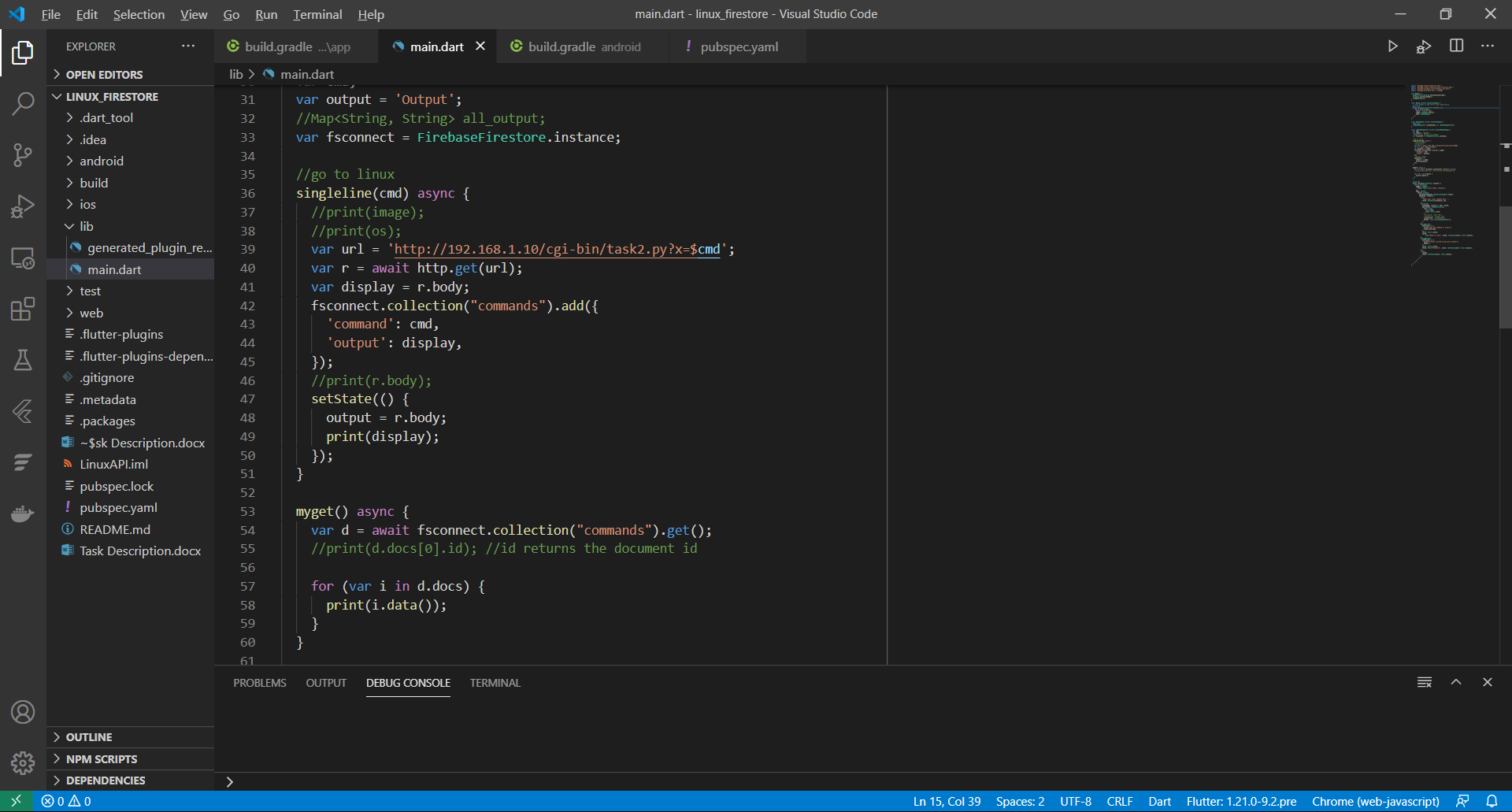




To use the firestore with our app, we use 2 methods to initialize our firestore variables, which are to be written in main() method as follows:



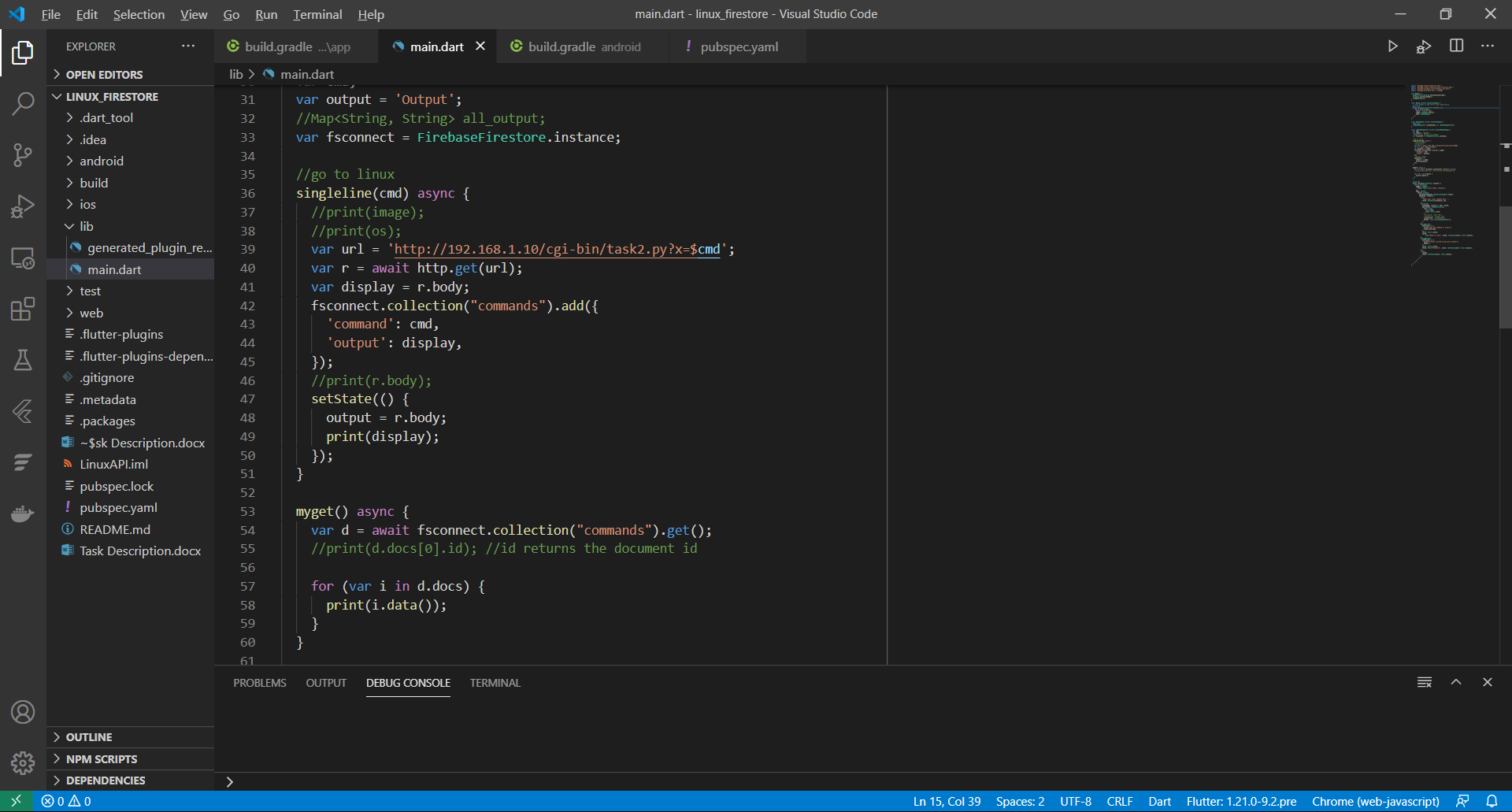
As seen in following snap, we can see that a variable called as fsconnect is used.



* This variable helps us to create an instance of the table/collection that we created in our firestore,
* So that we can connect to that collection and add more information with the add() method of that instance

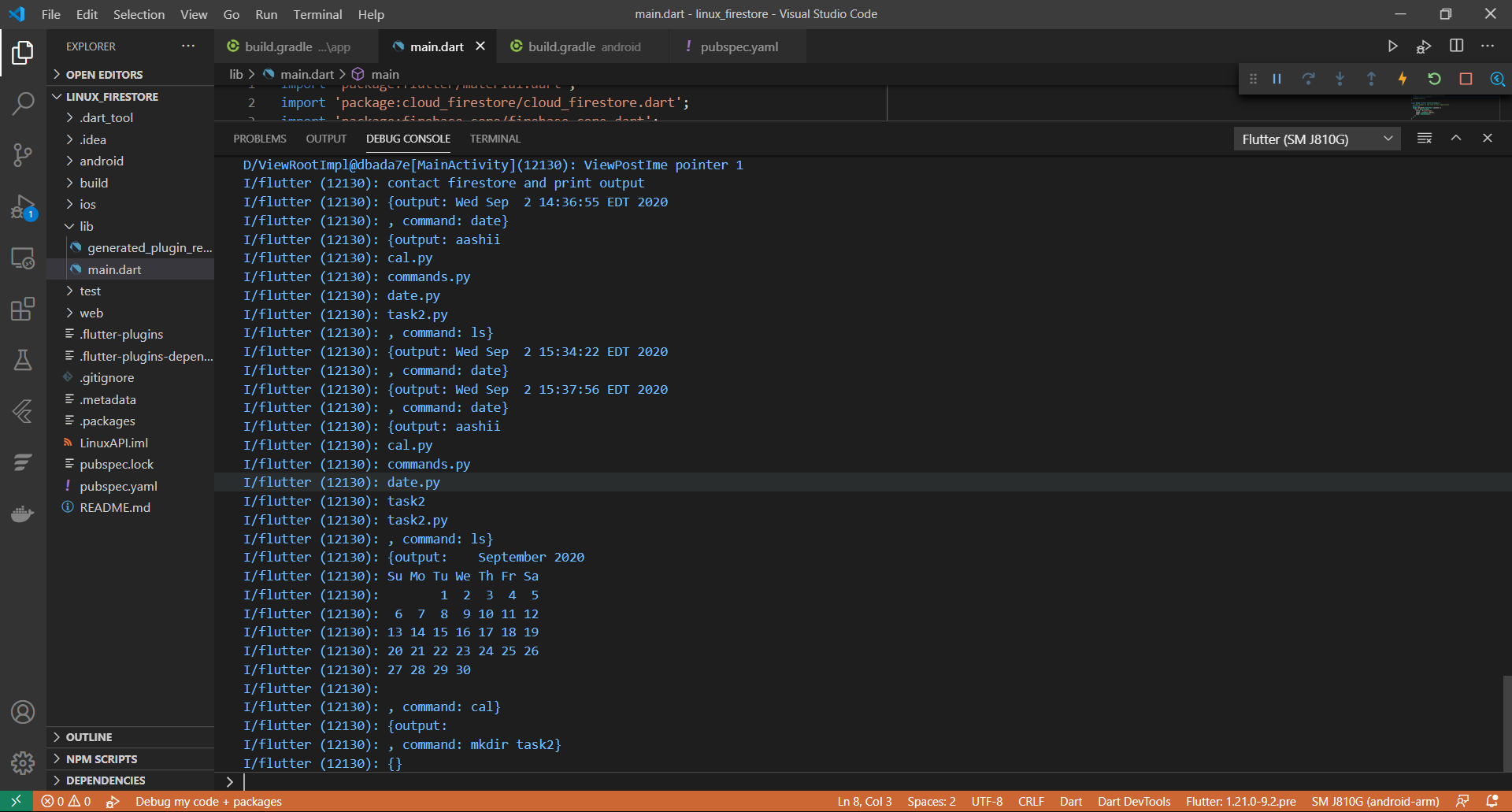
Here we add the command input by the user along with corresponding output

Now, when the button “Firestore” is clicked, we want the data to be retrieved by the output. For that we can simply use the get method to fetch the desired data



With this, we were able to achieve two things:

* Execute commands on Linux server from our app and see output on screen
* Store the outputs in a cloud-based database, from where all outputs can be retrieved and printed at once on the console as shown:



Final app looks like this for now:

<video>

Still I’m figuring out the part of fetching the data from firestore and displaying it on screen. So till then I’m printing it on console for now.