OBJECTIVE

The objective of this project is to make an mobile expense tracker application which is customised according to Operating System i.e Android and IOS.

The application is developed on flutter SDK which runs on the language Dart. Flutter is a cross-platform SDK which helps to maintain a single-code base for both android and IOS.

Our objective is to make an application on flutter and customize it according to the Operating System and see the difference in their UI and performance in both operating systems.

We will also understand how an application interacts with different operating systems.

Through this project we will be to learn the basics of app development using the flutter framework.

INTRODUCTION

About application and what it is based upon..

This project is based on an expense tracker application to manage daily expenditures and keep track of your incomes and savings. It shows statistical analysis of average expenditure in a day, monthly and weekly expenditure which will help the user to maintain a record of his spendings and save accordingly.

PROPOSED WORK

In the project, we have built a mobile application customised for IOS and Android.

Our application includes 4 pages.

Users can simply click the add transaction button to add a new entry and specify the amount and the date.

Also,we have provided an option to add comments and a picture with it.

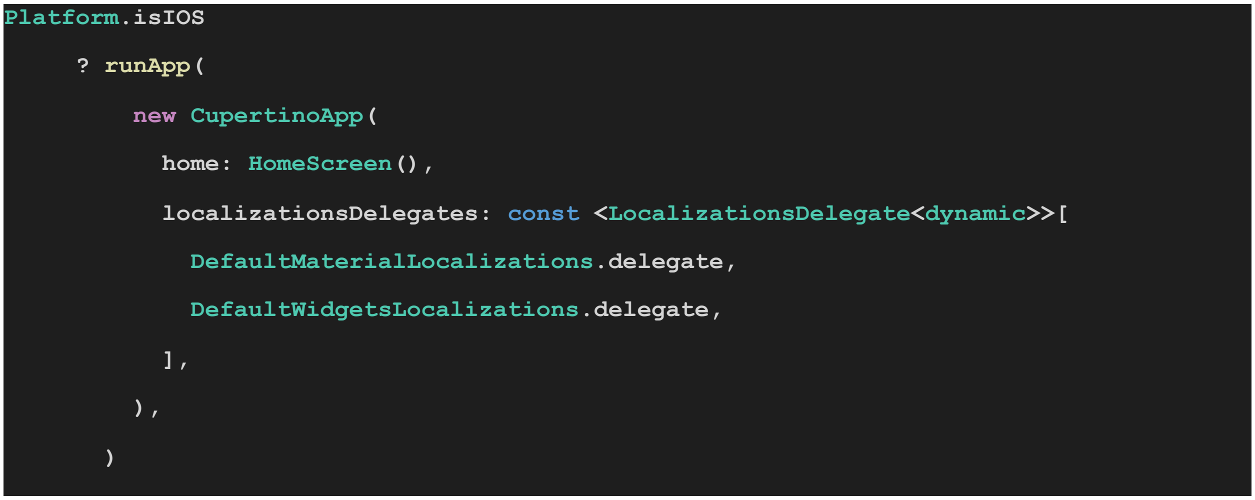
The user can delete the record by simply clicking the delete icon.

On clicking a transaction, the user will navigate to a new page which will show the details of the transaction like amount,title,date,a picture and a comment if added.

The second bottomNavigationTab is used to show the statistical analysis of the transactions which will show average expenditure per day ,expenditure in a week and expenditure in the last 30 days.

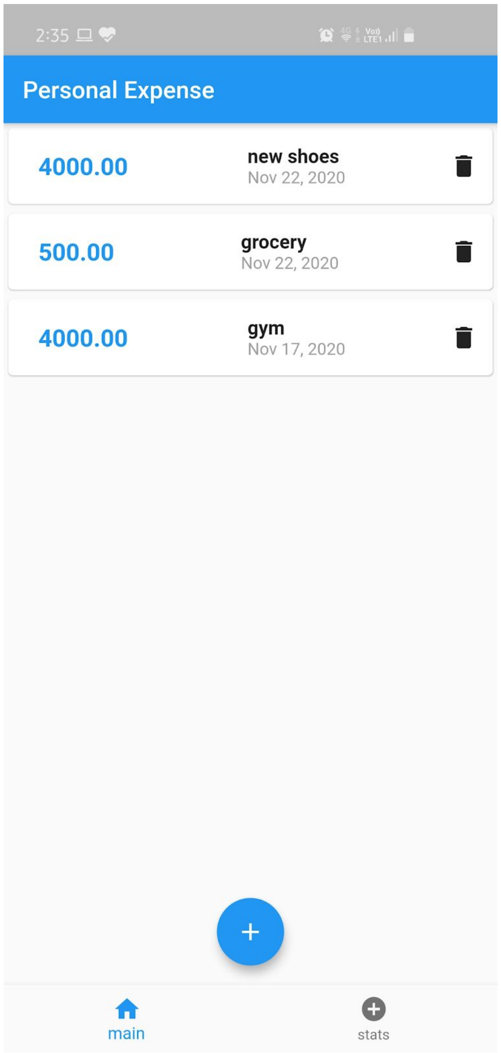
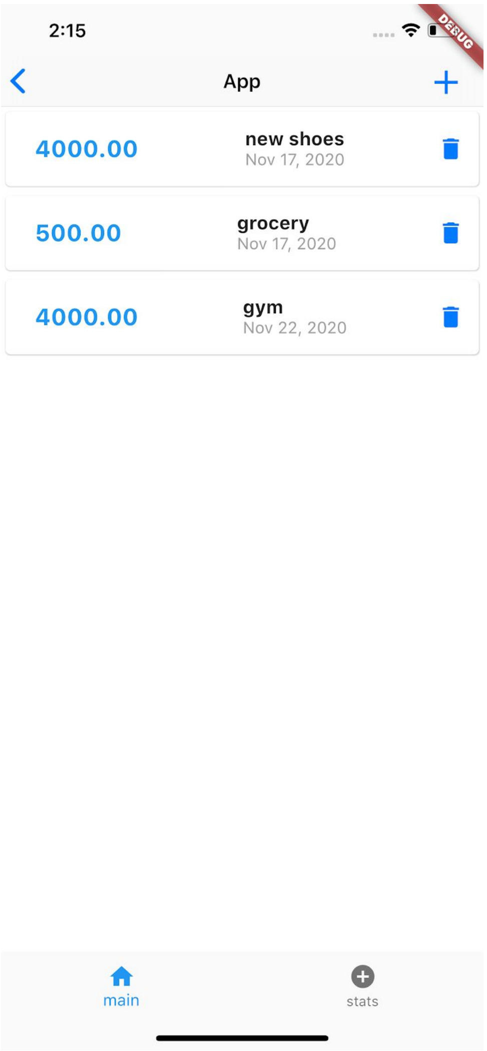
Platform.isIOS is a very useful function that comes with library ‘dart.io’ .**This is one of the most important features of our application as it allows you to customize applications according to the OS while maintaining a single-code base.**

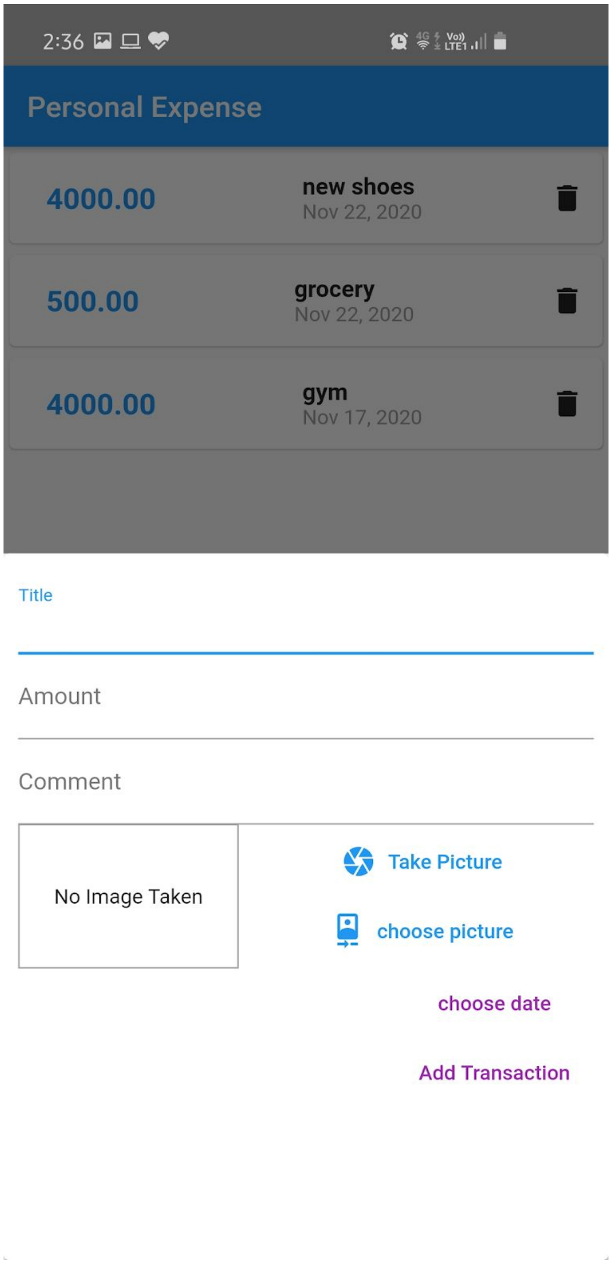
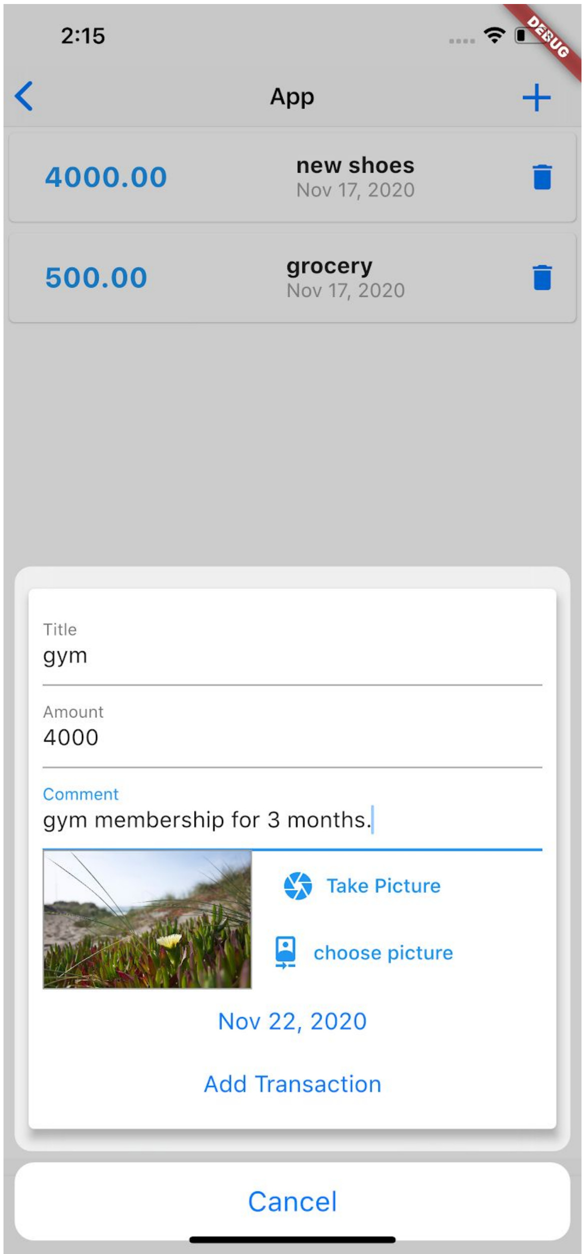
Here is a screenshot of our code showing how Platform.isIOS is implemented

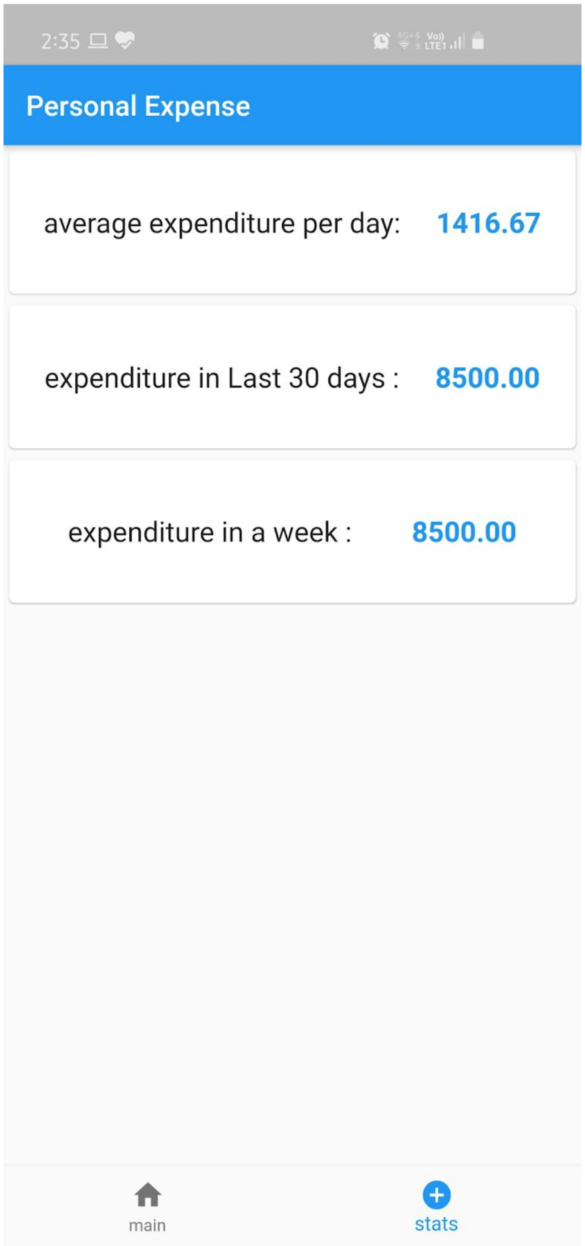


```
Platform.isIOS
     ? runApp(
         new CupertinoApp(
           home: HomeScreen(),
           localizationsDelegates: const <LocalizationsDelegate<dynamic>>[
             DefaultMaterialLocalizations.delegate,
             DefaultWidgetsLocalizations.delegate,
           ],
         ),
       )
```

```
    : runApp(MyApp());
```

ANDROID	IOS
 2:35 Personal Expense 4000.00 new shoes Nov 22, 2020 500.00 grocery Nov 22, 2020 4000.00 gym Nov 17, 2020 + main stats 1. This is the front page of android. 2. Android page Scaffold is used.	 2:15 DEBUG App 4000.00 new shoes Nov 17, 2020 500.00 grocery Nov 17, 2020 4000.00 gym Nov 22, 2020 main stats 1. This is the front page of IOS. 2. Cupertino Page Scaffold is used.

3. AppBar used is default android Appbar. 4. A FloatingActionButton is used to add new transactions.  2:36 Personal Expense 4000.00 new shoes Nov 22, 2020 500.00 grocery Nov 22, 2020 4000.00 gym Nov 17, 2020 Title Amount Comment No Image Taken Take Picture choose picture choose date Add Transaction For android, we used the 'showModelBottomSheet' widget.	3. AppBar is used as the default IOS app bar. 4. The trailing part of appBar is used to add new transactions.  2:15 DEBUG App 4000.00 new shoes Nov 17, 2020 500.00 grocery Nov 17, 2020 Title: gym Amount: 4000 Comment: gym membership for 3 months. Take Picture choose picture Nov 22, 2020 Add Transaction Cancel



Personal Expense

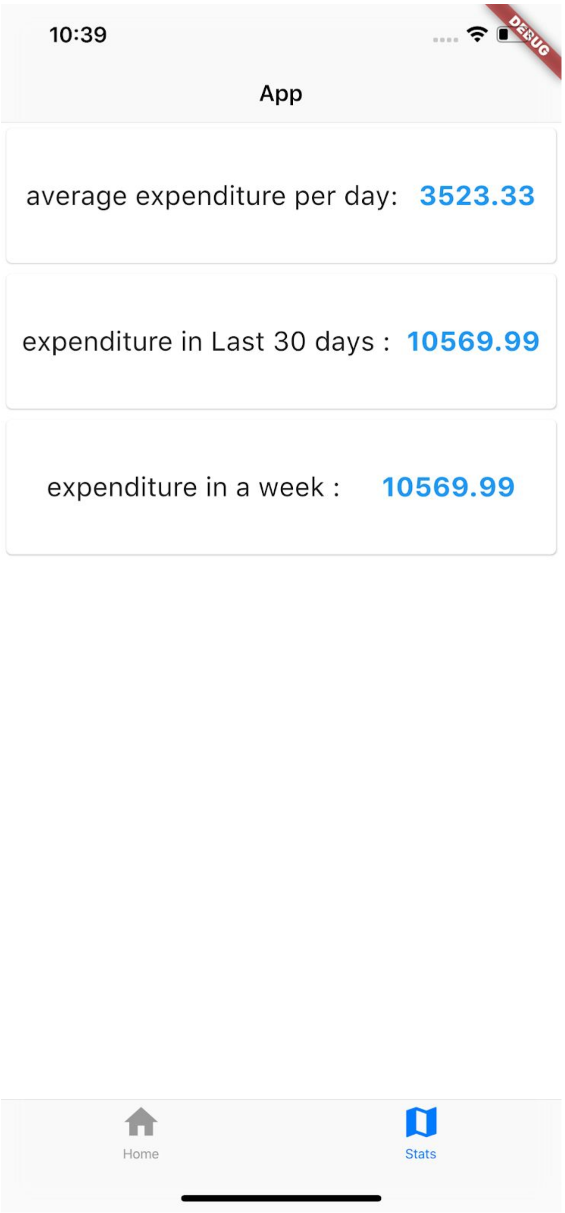
average expenditure per day: 1416.67

expenditure in Last 30 days : 8500.00

expenditure in a week : 8500.00

This is the second tabBar Page. TabBar is also customized according to the Operating system. For Android, we have used BottomNavigationBar.

For IOS, ‘CupertinoDialogAction’ widget is used which provides a really good IOS theme.



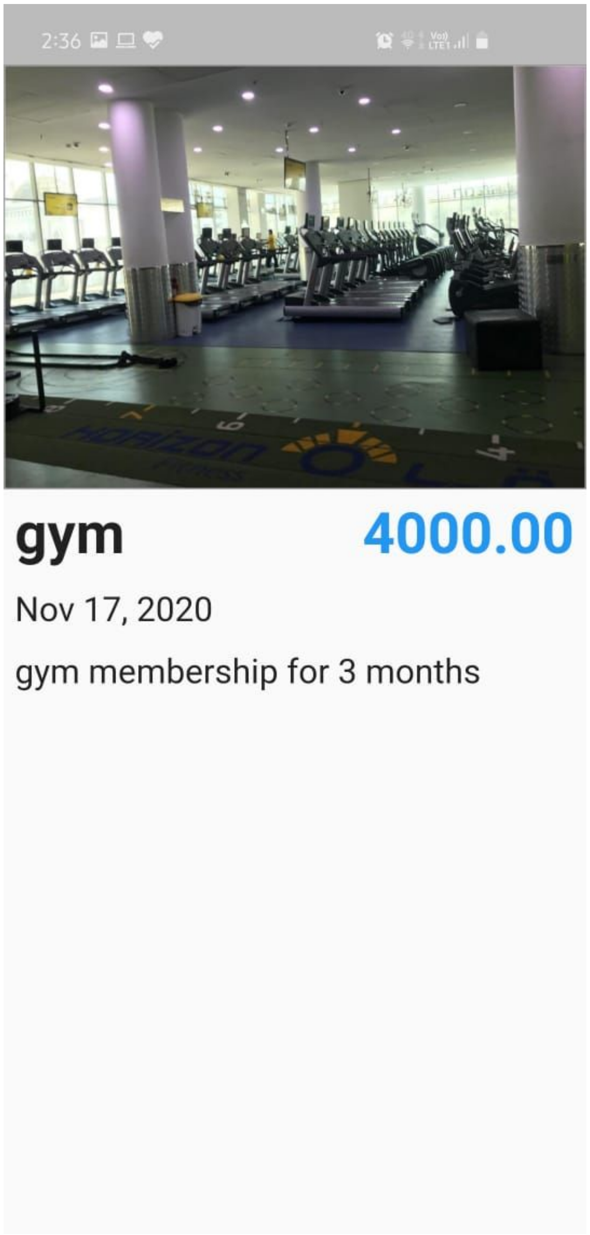
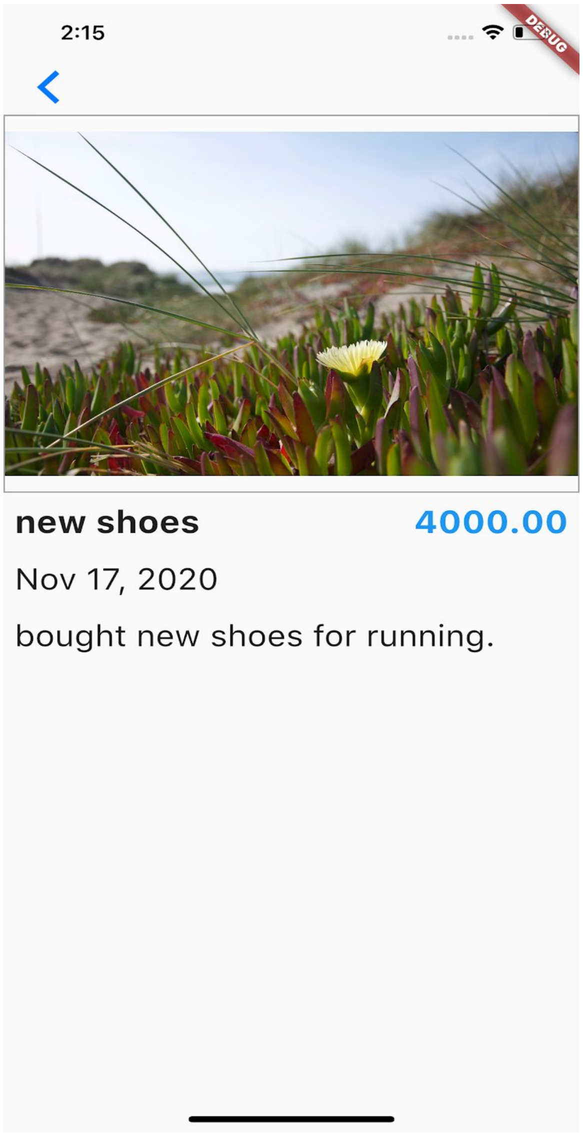
App

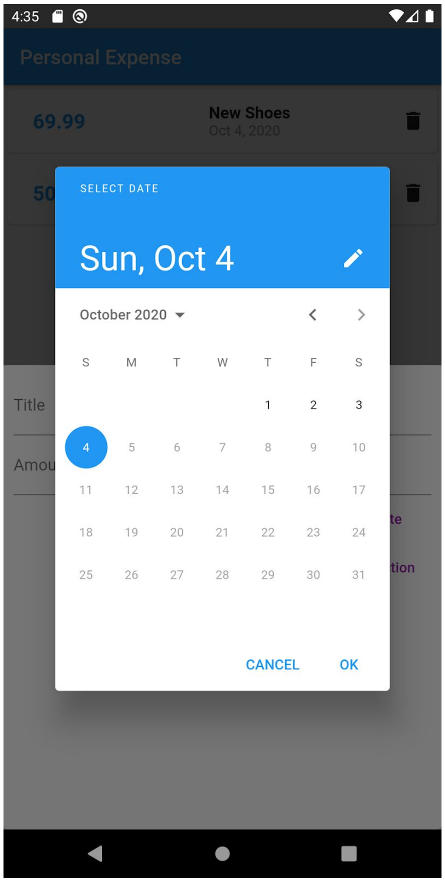
average expenditure per day: 3523.33

expenditure in Last 30 days : 10569.99

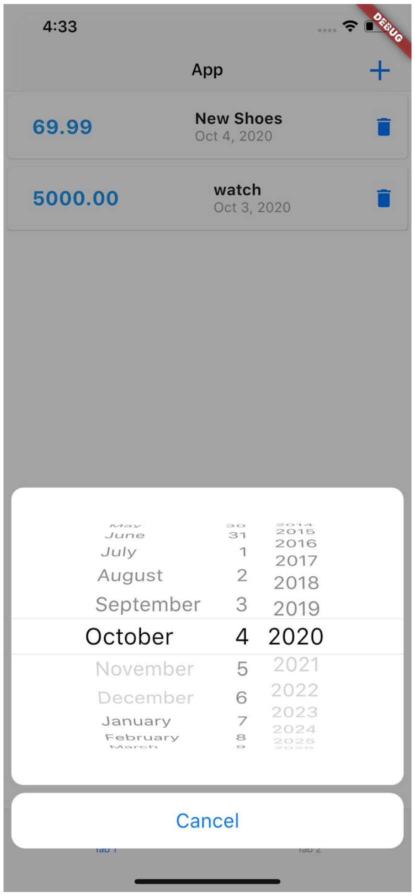
expenditure in a week : 10569.99

This is the second tabBar Page. TabBar is also customized according to the Operating system. For IOS, we have used cupertinoTabBar.

 gym 4000.00 Nov 17, 2020 gym membership for 3 months This Page is pushed when we Tap a transaction. It shows the picture added(if Any). And the comment on the transaction in a very simple and systematic layout. To go back we can simply press back button.	 new shoes 4000.00 Nov 17, 2020 bought new shoes for running. This Page is pushed when we Tap a transaction. It shows the picture added(if Any). And the comment on the transaction in a very simple and systematic layout. To go back to the homepage we can press the native back button of IOS which is shown on the app bar.



We have also kept in mind that the native google calendar is shown to pick the date in android.



While in IOS, the standard date picker of IOS is shown which adds up to the overall theme of an IOS application.

RESULTS AND DISCUSSION

In the above screenshots we saw how the application interacts differently with IOS and Android. Using flutter we were able to get that native “feel” on iPhone and iPad as well. While the basic functionality of the application remained the same, there is a huge difference in the UI of android and IOS applications.

We have used almost all of the Cupertino Widgets(IOS) available on flutter.

Outcomes-

While one can surely say that Flutter has done some great work to create hybrid applications, it still lacks in the number of Cupertino Widgets and the functionality of IOS applications. However as it is a relatively new language one can expect a lot of improvements in the SDK and it may rise as one of the top language for mobile development.

Future of hybrid languages-

Hybrid languages are already on the rise and many developers and companies are switching to these languages for its functionality, reusability, and most importantly **cost effectiveness.**
These languages have proved to be really effective and they have a great future ahead.

CONCLUSION

Through this project, we learned the fundamentals of mobile development using
the flutter framework.

We also saw how an application interacts with different operating systems i.e IOS and Android.

We conclude that flutter is an excellent and very useful framework for making hybrid applications.Through this we didn’t have to write separate codes for Android and IOS as Flutter is a hybrid framework and can have a single code basis which in turn can be customized according to the Operating System.

Also,using flutter we could make the interface similar to native applications.
We could use a single language for both front-end and back-end.

REFERENCES

Flutter https://flutter.dev/docs
Flutter and dart https://pub.dev