

‘Book Shelf’: A tool to access Books and minimize search time

Aryan Verma¹, Bhavsar Suraj, Aniket Kumar Singh, Anmol Patel Yadav

¹Students –LNCT (under RGPV University), Bhopal (MP), India

Abstract: Before invent to computer, book mainly referred to hard bound product with details printed on pages. Hence, it was cumbersome for readers to get relevant matter under single point of access. With technological development Books also evolved from contemporary hard print form to Online matter. Globalization is paving way for new dimension from Science to Arts to Medical to Engineering to Computer to Artificial intelligence to Space to Management to Economic, etc. These have in turn given rise various sector/field specific publications leading to increasingly available in electronic format. Its here the role do bookshelves have in the future library.

Summary: For decade’s people have been fond of reading books, major spaces in library had been occupied by open shelves, catering to the information requirements of students, scientists and scholars. But readers interest is impacted by the time expended in searching Books based on Topics, Authors, genres, etc. Hence not many of them are regular in reading new books due to lack of knowledge about the content provided by the author and often feel unwilling to buy them because of their new arrival in the market. New talented authors go unrecognised but this problem can be solved using which clusters the reviews of many people at one place

Of late University and public libraries are slowly transforming into study environments, in which open shelves are being replaced by a variety of other library services. So, the access to the books, journals, magazine, information, etc is increasingly transformed in digital format.

BOOK-SHELF is Mobile application based tool which could be effectively used in University and public libraries, Vendors, stocktist, etc to minimize time and physical efforts in searching hard bound matter.

Key words: Books, Journal, Magazine, Library, book management, Android Smartphone, Mobile Library, Open source, Library

I. INTRODUCTIONS

According to Counterpoint Research, India's smart phone market is set to hit a record high of 173 million units in 2021, growing at 14 per cent annually India's smart phone market is the second biggest market in the world after China. In 2020, the Indian market outperformed the North America, Latin America and Africa markets. The market has been experiencing strong consumer demand after the lifting of COVID-19 restrictions in June, and the strong sales momentum will continue.[1]

This strong mobile use base had given birth of idea of mobile based application of Reading books and reviewing it. ‘Book Shelf’ is an approach to integrate user opinions cum reviews

about various books, magazine, novels, journals and also help readers to locate the availability of Book. We could no longer focus solely on paper publications, but on digital publications too, as most library collections increasingly become hybrid. [2]

II. PRESENT SCENARIO

We know that an information system needs to be accurate and fast. But this information could be sourced from different disciplines. Usually visitors look for references and obtain information accordingly. To upscale the use in digital user can opt for ‘Android’ which is released by ‘Google’ as an open-source mobile phone operating system very much similar to Linux-based platform. In general, it consists of the operating system, middleware, and user interface and application software. The success and development of the current android capable of occupying the highest position gadgets and computer market, it is certainly due to the sophistication of technology systems and applications that are on it are currently a trend among mobile phone users because it can help all areas of work so that it becomes easier . The advantages of android is the open source license so it is possible for anyone with an android programming ability to create or develop applications to run on Android based gadgets. [3]

III. RESEARCH OVERVIEW

Flutter is an free and open-source user interface software development kit made by google which is used to develop cross platform (iOS and Android) apps in fast and expressive way. Flutter enables fast development since it has a rich set of customizable and prebuilt widgets, motion API and wide set of library Flutter apps are written in the Dart language.

The central idea behind Flutter is the use of widgets that combines different widgets that helps developers to build the entire UI. Each of these widgets defines a structural element (like a button or menu), a stylistic element, a layout aspect (like padding), and many others.

Flutter provides developers with its own ready-made widgets that look native to Android or iOS apps. Naturally, developers can create their own widgets as well.

Dart like other systems that use reactive views, Flutter refreshes the view tree for every new frame. To accomplish that, it creates many objects that may live for no more than

one frame. Dart uses generational garbage collection that has proven to be very efficient for this type of systems.

Moreover, Dart has a “tree shaking” compiler that only includes the code that developer needs in coding of application (APP). It is user friendly even if just a widget or two, which can be sourced from its large library of widgets freely.

Finally, Dart comes with a repository of software packages for extending the capabilities of apps. For example, it offers a few packages that help to access Firebase so that developers can build server less apps. Another package allows accessing a Redux data store or makes it easier to access platform services and hardware like the camera.

IV. APPLICATION DESIGN

Application design consists of ‘User’ with latest mobile android, login interface of ‘Google’ for sourcing details like book, author, reviews, location of book availability, ‘Digital Search record’ for displaying details of information aggregated from back end.

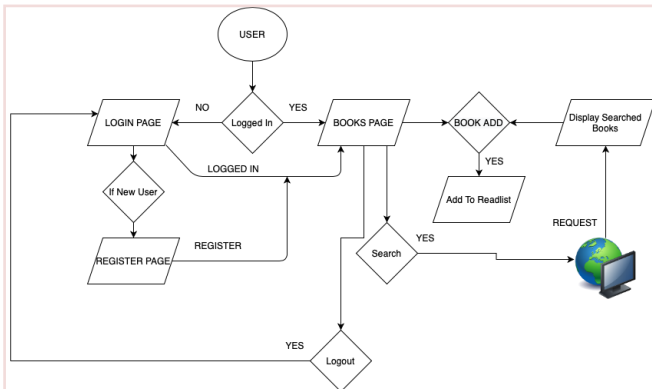


Figure 1: Block diagram of application design.

So for example, if App is installed on ‘Android’ based Mobile than User or Reader could search Book, Magazine, etc and can submit its review. This would be checked by Administrator and only required information in concise form would be processed and uploaded on interface.

This would be useful for various readers residing in other geographically different locations. Thus, increasing the penetration of Books, its content and promoting its sale. This would save time, money in travelling, physical efforts, etc and also beneficial for Author, Publishers, Readers, Book Stockiest/Distributors, Schools, College, University, Libraries, etc.

V. CODING

```
import 'package:book_shelf/pages/home.dart';
import 'package:book_shelf/pages/login_page.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'controllers/auth_controller.dart';
```

```
void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp();
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  final authController = Get.put(AuthController());

  // final _readListController = Get.put(ReadListController());

  @override
  Widget build(BuildContext context) {
    return GetMaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        primaryColor: Colors.black,
        accentColor: Colors.black45,
        appBarTheme: AppBarTheme(
          elevation: 0,
        ),
        scaffoldBackgroundColor: Colors.black54,
        cursorColor: Colors.white,
        textSelectionColor: Colors.yellow.withOpacity(0.3),
        textTheme: TextTheme(
          headline6: TextStyle(fontSize: 16, color:
Colors.white70),
          headline5: TextStyle(fontSize: 18, color:
Colors.white70),
          headline4: TextStyle(fontSize: 24, color:
Colors.white70),
          headline2: TextStyle(fontSize: 28, color:
Colors.white70),
          headline3: TextStyle(fontSize: 30, color:
Colors.white70),
          headline1: TextStyle(fontSize: 32, color:
Colors.white70),
          caption: TextStyle(fontSize: 14, color:
Colors.white70),
        )),
        home: _getHome(),
      );
  }

  Widget _getHome() {
    return authController.user != null ? Home() : LoginPage();
  }
}
```

```
import
'package:book_shelf/controllers/read_list_controller.dart';
import 'package:book_shelf/pages/home_page.dart';
import 'package:book_shelf/pages/profile_page.dart';
import 'package:book_shelf/pages/read_list_page.dart';
import 'package:flutter/material.dart';
```

```

import 'package:get/get.dart';

class Home extends StatelessWidget {
  final readListController = Get.put(ReadListController());
  final _currentIndex = 0.obs;
  final pages = [
    HomePage(),
    ReadListPage(),
    ProfilePage(),
  ];

  @override
  Widget build(BuildContext context) {
    return Obx(() {
      return Scaffold(
        body: buildBody(context),
        bottomNavigationBar:
buildBottomNavigationBar(context),
      );
    });
  }

  Widget buildBody(BuildContext context) {
    return pages[_currentIndex.value];
  }

  Widget buildBottomNavigationBar(BuildContext context) {
    return BottomNavigationBar(
      selectedItemColor: Colors.orangeAccent,
      backgroundColor: Colors.black12,
      currentIndex: _currentIndex.value,
      onTap: (index) {
        _currentIndex(index);
      },
      items: [
        BottomNavigationBarItem(
          icon: Icon(
            Icons.home,
          ),
          label: 'Home',
        ),
        BottomNavigationBarItem(
          icon: Icon(
            Icons.chrome_reader_mode,
          ),
          label: 'Read List',
        ),
        BottomNavigationBarItem(
          icon: Icon(Icons.account_circle),
          label: 'My Profile',
        ),
      ],
    );
  }
}
import 'package:book_shelf/constants/category.dart';

```

```

import 'package:book_shelf/constants/dimensions.dart';
import
'package:book_shelf/controllers/books_controller.dart';
import 'package:book_shelf/enums/genre.dart';
import 'package:book_shelf/pages/search_page.dart';
import 'package:book_shelf/widgets/book_tile.dart';
import 'package:book_shelf/widgets/pagination_view.dart';
import 'package:flutter/material.dart';
import 'package:get/get.dart';

class HomePage extends StatelessWidget {
  final _booksController = Get.put(BooksController());
  final _categories = Category.categories.obs;
  final scrollController = ScrollController();
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Book Shelf'),
        actions: [
          IconButton(
            icon: Icon(Icons.search), onPressed: () =>
Get.to(SearchPage()),
          ),
        ],
      body: Obx(() {
        return buildBody(context);
      }),
    );
  }

  Widget buildBody(BuildContext context) {
    return SafeArea(
      child: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Container(
              height: 140,
              child: ListView.builder(
                scrollDirection: Axis.horizontal,
                itemCount: _categories.length,
                itemBuilder: (context, index) {
                  return Obx(
                    () {
                      return Center(
                        child: buildGenreCard(context,
_categories[index]);
                    },
                  );
                },
              ),
            ),
            SizedBox(height: 8),
            Padding(
              padding: const EdgeInsets.all(4.0),

```

```

    child: Text(
      _booksController.selectedGenre.value.getName,
      style: Theme.of(context).textTheme.headline4,
    ),
  ),
  SizedBox(height: 12),
  buildBookList(context),
],
),
);
}

Widget buildGenreCard(BuildContext context, Category
category) {
  return GestureDetector(
    onTap: () {
      if (_booksController.selectedGenre.value !=
category.title) {
        _booksController.selectedGenre(category.title);
      }
    },
    child: Container(
      margin: EdgeInsets.symmetric(horizontal: 6, vertical: 6),
      child: Stack(
        children: [
          Container(
            clipBehavior: Clip.antiAlias,
            decoration:
              BoxDecoration(borderRadius:
BorderRadius.circular(12)),
            child: (_booksController.selectedGenre.value ==
category.title)
              ? Image.asset(
                  category.imagePath,
                  width: 150,
                  height: 120,
                  fit: BoxFit.cover,
                )
              : Image.asset(
                  category.imagePath,
                  width: 120,
                  height: 90,
                  fit: BoxFit.cover,
                ),
          ),
          Positioned(
            bottom: 20,
            left: 8,
            right: 8,
            child: Text(
              category.title.getName,
              style: Theme.of(context)
                .textTheme
                .headline6
                .apply(fontSizeDelta: 1, backgroundColor:

```

```

Colors.black38),
          ),
        ],
      ),
    );
}

Widget buildBookList(BuildContext context) {
  return Expanded(
    child: PaginationView(
      child: ListView.builder(
        shrinkWrap: true,
        physics: NeverScrollableScrollPhysics(),
        itemCount: _booksController.books.length,
        itemBuilder: (context, index) {
          return BookTile(
            book: _booksController.books[index],
            imageHeight: Dimensions.mediumImageHeight,
            imageWidth: Dimensions.mediumImageWidth,
            ratingIconSize: 24,
          );
        },
      ),
      isLoading: _booksController.isLoading.value,
      total: _booksController.total.value,
      onNext: () {
        _booksController.getBooksByGenre(_booksController.books.
length);
      },
      count: _booksController.books.length,
    ),
  );
}

import 'package:book_shelf/constants/dimensions.dart';
import 'package:book_shelf/constants/language.dart';
import
'package:book_shelf/controllers/read_list_controller.dart';
import 'package:book_shelf/models/book.dart';
import 'package:book_shelf/widgets/book_image.dart';
import 'package:flutter/material.dart';
import 'package:flutter_rating_bar/flutter_rating_bar.dart';
import 'package:get/get.dart';

class BookDetailsPage extends StatelessWidget {
  final Book book;
  BookDetailsPage({Key key, @required this.book}) :
super(key: key);
  final ReadListController _readListController = Get.find();

  @override
  Widget build(BuildContext context) {
    return Scaffold(

```

```

appBar: AppBar(
  title: Text('Book Details'),
),
body: buildBody(context),
floatingActionButton: Obx() {
  return _buildFloatingAction(context);
});
}

Widget buildBody(BuildContext context) {
return ListView(
  children: [
    Padding(
      padding: EdgeInsets.all(24),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          buildImage(context),
          SizedBox(height: 24),
          buildGeneralDetails(context),
          SizedBox(height: 24),
          buildDescriptionTitle(context),
          SizedBox(height: 24),
        ],
      ),
    ),
  ],
);
}

Widget _buildFloatingAction(BuildContext context) {
return _readListController.isAvailable(book)
? FloatingActionButton(
  backgroundColor:
Colors.yellowAccent.withOpacity(0.5),
  onPressed: () {},
  child: Icon(
    Icons.done,
    color: Colors.black87,
  ),
  tooltip: 'Already Added',
)
: Builder(
  builder: (context) {
return FloatingActionButton(
  backgroundColor:
Colors.yellowAccent.withOpacity(0.5),
  tooltip: 'Add to readlist',
  child: Icon(
    Icons.add,
    color: Colors.black87,
  ),
  onPressed: () => _didTapAddButton(context),
  elevation: 6,
);
},
}

```

```

);
}
Widget buildImage(BuildContext context) {
return Container(
  child: Center(
    child: BookImage(book: book, height:
Dimensions.largeImageHeight),
  ),
);
}

Widget buildGeneralDetails(BuildContext context) {
return Container(
  width: double.maxFinite,
  decoration: BoxDecoration(
    color: Colors.grey[900],
    borderRadius: BorderRadius.circular(12),
    boxShadow: [
      BoxShadow(
        blurRadius: 8, color: Colors.grey[700], offset:
Offset(0, 0)
      ),
    ],
  child: Padding(
    padding: const EdgeInsets.all(16.0),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      mainAxisAlignment: MainAxisAlignment.start,
      children: [
        Text(book.title, style:
Theme.of(context).textTheme.headline4),
        SizedBox(height: 16),
        if (book.authors.isNotEmpty)
          Text(
            'Author(s): ${book.authors.join(', ')}',
            style: Theme.of(context).textTheme.headline6,
          ),
        SizedBox(height: 12),
        if (book.publisher.isNotEmpty)
          Text('Published by: ${book.publisher}',
            style: Theme.of(context).textTheme.headline6),
        SizedBox(height: 12),
        if (book.publishedDate.isNotEmpty)
          Text('Published Date: ${book.publishedDate}',
            style: Theme.of(context).textTheme.headline6),
        SizedBox(height: 12),
        if (book.pageCount > 0)
          Text('Total Pages: ${book.pageCount}',
            style: Theme.of(context).textTheme.headline6),
        SizedBox(height: 12),
        if (book.genres.isNotEmpty)
          Text(
            'Genre(s): ${book.genres.join(', ')}',
            style: Theme.of(context).textTheme.headline6,
          ),
        SizedBox(height: 12),
        if (book.language.isNotEmpty)
          Text('Language:

```



```

    ${Language.language(book.language)}',
    style: Theme.of(context).textTheme.headline6),
    SizedBox(height: 16),
    Row(
      children: [
        RatingBarIndicator(
          itemPadding: EdgeInsets.symmetric(horizontal: 4),
          itemCount: 5,
          itemSize: 40,
          unratedColor: Colors.yellow.withOpacity(0.5),
          rating: double.parse(book.rating.isEmpty ? '3' :
book.rating),
          itemBuilder: (context, index) {
            return Icon(Icons.star, color:
Colors.yellowAccent);
          },
        ),
        SizedBox(width: 12),
        if (book.ratingsCount > 0)
          Expanded(
            child: Text(
              '${book.ratingsCount.toString()} ratings',
              style: Theme.of(context).textTheme.headline5,
            ),
          ),
      ],
    ),
  ],
),
);
}

Widget buildDescriptionTitle(BuildContext context) {
  return Text(
    'Description',
    style: Theme.of(context).textTheme.headline3,
  );
}

Widget buildDescriptionWidget(BuildContext context) {
  return Container(
    height: 250,
    child: Column(
      children: [
        Flexible(
          child: Container(
            width: double.maxFinite,
            decoration: BoxDecoration(
              color: Colors.grey[900],
              borderRadius: BorderRadius.circular(12),
              boxShadow: [
                BoxShadow(
                  blurRadius: 8,
                  color: Colors.grey[700],
                  offset: Offset(0, 0)
                ),
              ],
          ),
        ),

```

```

        child: Padding(
          padding: const EdgeInsets.all(16.0),
          child: SingleChildScrollView(
            child: Text(book.description,
              textAlign: TextAlign.start,
              style: TextStyle(
                fontSize: 16,
                color: Colors.white54,
                letterSpacing: 1,
                wordSpacing: 4)),
          ),
        ),
      ],
    ),
);
}

void _didTapAddButton(BuildContext context) {
  _readListController.add(book);
  final snackBar = SnackBar(
    content: Text('Added to Readlist!'),
    action: SnackBarAction(
      label: 'Undo',
      textColor: Colors.white,
      onPressed: () {
        _readListController.remove(book);
      },
    ),
  );
  Scaffold.of(context).showSnackBar(snackBar);
}

import 'package:book_shelf/constants/dimensions.dart';
import
'package:book_shelf/controllers/read_list_controller.dart';
import 'package:book_shelf/models/book.dart';
import 'package:book_shelf/widgets/book_tile.dart';
import 'package:flutter/material.dart';
import 'package:get/get.dart';

class ReadListPage extends StatelessWidget {
  final ReadListController _readListController = Get.find();
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('My ReadList'),
      ),
      body: Obx(() {
        return _buildBody(context);
      }));
  }

  Widget _buildBody(BuildContext context) {
    return Padding(

```

```

padding: const EdgeInsets.symmetric(horizontal: 20,
vertical: 16),
child: ListView.builder(
  itemCount: _readListController.books.length,
  itemBuilder: (context, index) {
    return Dismissible(
      direction: DismissDirection.endToStart,
      background: Container(
        color: Colors.red.withOpacity(0.7),
        padding: EdgeInsets.symmetric(horizontal: 20),
        alignment: AlignmentDirectional.centerEnd,
        child: Icon(
          Icons.delete,
          color: Colors.white,
          size: 36,
        ),
      ),
      key: Key(_readListController.books[index].id),
      onDismissed: (direction) {
        removeBook(context,
_readListController.books[index]);
      },
      child: BookTile(
        book: _readListController.books[index],
        imageHeight: Dimensions.smallImageHeight,
        imageWidth: Dimensions.smallImageWidth,
      ),
    );
  },
);
}

void removeBook(BuildContext context, Book book) {
_readListController.remove(book);
final snackBar = SnackBar(
  content: Text('Removed Successfully'),
  action: SnackBarAction(
    label: 'Undo',
    textColor: Colors.white,
    onPressed: () {
_readListController.add(book);
  },
);
 Scaffold.of(context).showSnackBar(snackBar);
}

import 'package:book_shelf/controllers/auth_controller.dart';
import
'package:book_shelf/widgets/primary_raised_button.dart';
import 'package:book_shelf/widgets/primary_text_field.dart';
import 'package:flutter/material.dart';
import 'package:get/get.dart';

class RegisterPage extends StatelessWidget {

```

```

final AuthController _authController = Get.find();
final _emailController = TextEditingController();
final _nameController = TextEditingController();
final _passwordController = TextEditingController();

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Register'),
    ),
    body: _buildBody(context),
  );
}

Widget _buildBody(BuildContext context) {
  return Padding(
    padding: const EdgeInsets.all(24.0),
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      crossAxisAlignment: CrossAxisAlignment.stretch,
      children: [
        Obx(() {
          return PrimaryTextField(
            hint: 'Full Name',
            controller: _nameController,
            errorText: _authController.nameError.value,
            prefixIcon: Icon(Icons.person),
            keyboardType: TextInputType.name,
          );
        })),
        SizedBox(height: 16),
        Obx(() {
          return PrimaryTextField(
            hint: 'Email',
            controller: _emailController,
            errorText: _authController.registerEmailError.value,
            prefixIcon: Icon(Icons.email),
            keyboardType: TextInputType.emailAddress,
          );
        })),
        SizedBox(height: 16),
        Obx(() {
          return PrimaryTextField(
            hint: 'Password',
            controller: _passwordController,
            errorText:
_authController.registerPasswordError.value,
            prefixIcon: Icon(Icons.lock),
            suffix: _buildSuffixButton(),
            keyboardType: TextInputType.name,
            isObscure: _authController.registerIsObscure.value,
          );
        })),
        SizedBox(height: 32),
        PrimaryRaisedButton(onPressed: _didTapRegister, text:

```

```

'REGISTER'),
  ],
),
);
}
Widget _buildSuffixButton() {
  return IconButton(
    icon: Icon((_authController.registerIsObscure.value
      ? Icons.remove_red_eye
      : Icons.remove_red_eye_outlined),
    onPressed: _didTapSuffixButton);
}

void _didTapSuffixButton() {
  _authController.registerIsObscure.value =
    !(_authController.registerIsObscure.value);
}

void _didTapRegister() {
  _authController.register(
    email: _emailController.text,
    password: _passwordController.text,
    name: _nameController.text);
}

import 'package:book_shelf/constants/dimensions.dart';
import 'package:book_shelf/widgets/book_tile.dart';
import 'package:book_shelf/widgets/pagination_view.dart';
import 'package:flutter/material.dart';
import
'package:book_shelf/controllers/search_books_controller.dart'
;
import 'package:get/get.dart';

class SearchPage extends StatelessWidget {
  final SearchBooksController _searchBooksController =
    Get.put(SearchBooksController());
  final TextEditingController _textEditingController =
    TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: TextField(
          cursorColor: Colors.white,
          style: TextStyle(
            color: Colors.white,
          ),
          onChanged: (value) =>
            _searchBooksController.query(value),
          controller: _textEditingController,
          autofocus: true,
          decoration: InputDecoration(
            hoverColor: Colors.white,
            hintText: 'Search by title, author, publisher...',
            hintStyle: Theme.of(context).textTheme.headline6,

```

```

),
),
actions: [
  IconButton(
    icon: Icon(Icons.clear),
    onPressed: () { _textEditingController.clear();
  },
),
],
),
body: Obx(() {
  return _buildBody();
}),
);
}

Widget _buildBody() {
  return Padding(
    padding: const EdgeInsets.symmetric(horizontal: 20,
      vertical: 12),
    child: PaginationView(
      child: ListView.builder(
        shrinkWrap: true,
        physics: NeverScrollableScrollPhysics(),
        itemCount: _searchBooksController.books.length,
        itemBuilder: (context, index) {
          return BookTile(
            book: _searchBooksController.books[index],
            imageHeight: Dimensions.smallImageHeight,
            imageWidth: Dimensions.smallImageWidth);
        },
      ),
      isLoading: _searchBooksController.isLoading.value,
      total: _searchBooksController.total.value,
      onNext: () {
        _searchBooksController.getBooks(_searchBooksController.b
          ooks.length);
      },
      count: _searchBooksController.books.length,
    ),
  );
}

import 'package:book_shelf/controllers/auth_controller.dart';
import 'package:book_shelf/pages/register_page.dart';
import
'package:book_shelf/widgets/primary_raised_button.dart';
import 'package:book_shelf/widgets/primary_text_field.dart';
import 'package:flutter/material.dart';
import 'package:get/get.dart';

class LoginPage extends StatelessWidget {
  final AuthController _authController = Get.find();
  final TextEditingController _emailController =

```



```

TextEditingController();
final TextEditingController _passwordController =
TextEditingController();

@override
Widget build(BuildContext context) {
return Scaffold(
  appBar: AppBar(
    title: Text('Login'),
  ),
  body: _buildBody(context),
);
}
Widget _buildBody(BuildContext context) {
return Padding(
  padding: const EdgeInsets.all(24.0),
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    crossAxisAlignment: CrossAxisAlignment.stretch,
    children: [
      Obx() {
return PrimaryTextField(
  hint: 'Email',
  controller: _emailController,
  errorText: _authController.signInEmailError.value,
  prefixIcon: Icon(Icons.email),
  textInputType: TextInputType.emailAddress,
);
}),
      SizedBox(height: 16),
      Obx() {
return PrimaryTextField(
  hint: 'Password',
  controller: _passwordController,
  errorText:
_authController.signInPasswordError.value,
  prefixIcon: Icon(Icons.lock),
  textInputType: TextInputType.name,
  suffix: _buildSuffixButton(),
  isObscure: _authController.signInIsObscure.value,
);
}),
      SizedBox(height: 32),
      PrimaryRaisedButton(onPressed: _didTapLogin, text:
'LOGIN'),
      SizedBox(height: 24),
      PrimaryRaisedButton(onPressed: _didTapRegister, text:
'REGISTER'),
    ],
  ),
);
}

Widget _buildSuffixButton() {
return IconButton(
  icon: Icon(_authController.signInIsObscure.value

```

```

? Icons.remove_red_eye
: Icons.remove_red_eye_outlined),
onPressed: _didTapSuffixButton);
}

void _didTapSuffixButton() {
_authController.signInIsObscure.value =
!(_authController.signInIsObscure.value);
}

void _didTapLogin() {
_authController.signIn(
  email: _emailController.text, password:
_passwordController.text);
}

void _didTapRegister() {
Get.to(RegisterPage());
}
}
}

```

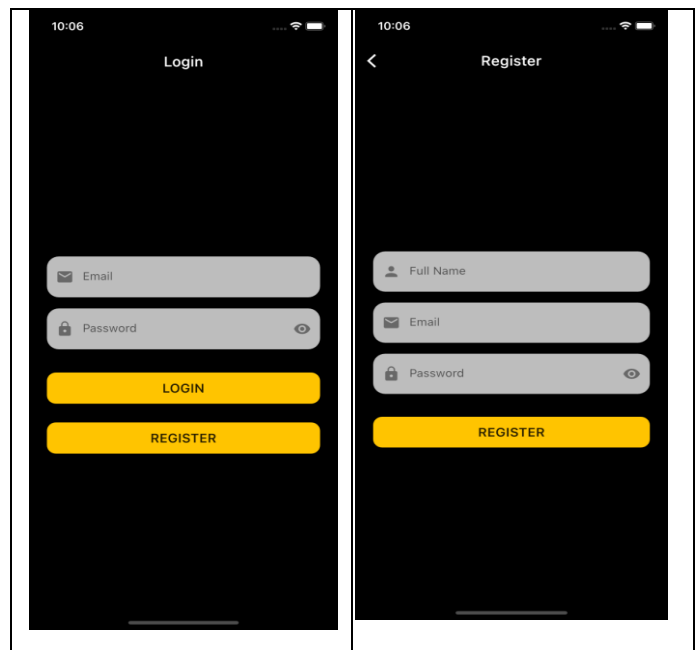
[4]

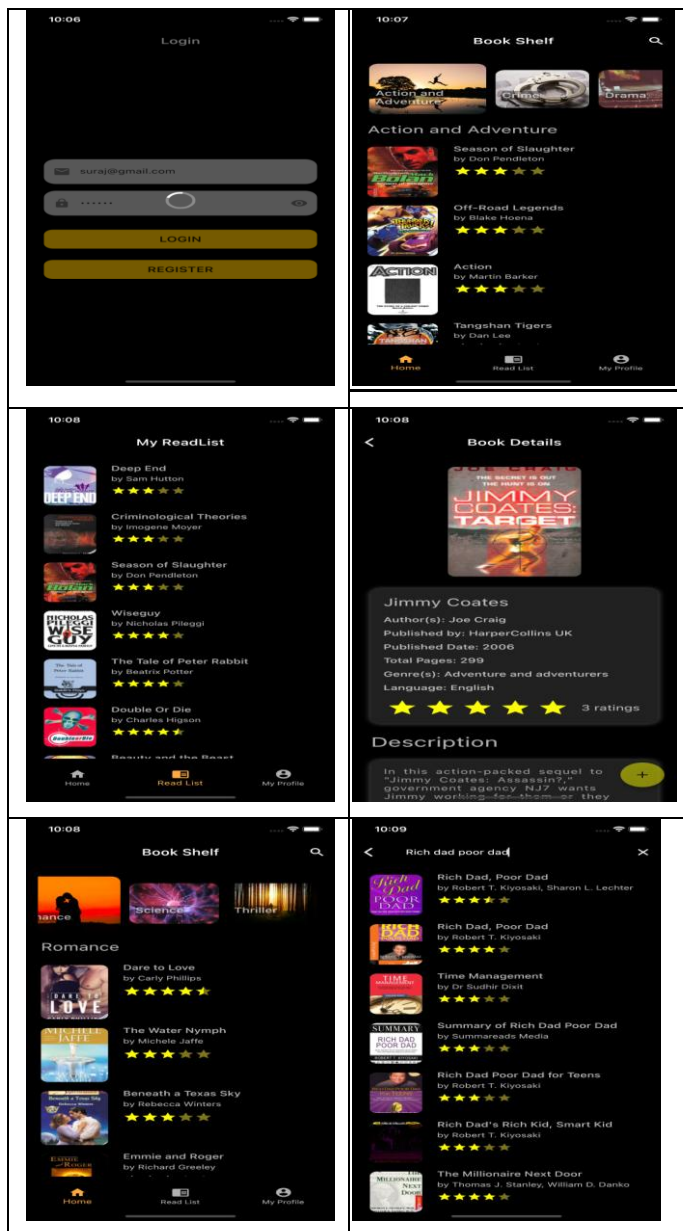
VI. FUTURE DEVELOPMENT AND UP GRADATION

This application is presently in an initial budding stage and could be further developed to integrate other features like Book availability, stock expiry/shortage, publishers access, region or area wise book sale, etc.[5]

As ‘Book-Shelf’ applications becomes “popular” along friends/family it could attract advertisement from various sources. Thus, increasing income for app or website at minimal intellectual cost (or sometimes for free) rather than using heavily priced advertising agencies.

The present user interface of Application ‘Book Shelf’ has been detailed below:-





The above interface has been designed with help of Flutter[4], Firebase [6], Dart[7] and sources review from Google API Books [8].

VII. CONCLUSION

Based on the simulation of application in mobile following are experimental results are concluded:-

- 1) Reader/user with the help of Android smartphone can access libraries or any book Online.
- 2) Applications enable multiple search for the information that available in and enter reviews of books.
- 3) This Application helps readers at multiple locations read and other services like borrow books online, deadline of return, stock end details, etc

- 4) Application helps to monitor the status of the books rating along with Sale and purchase
- 5) minimized as a result of delays in returning books forgot the date of return of the book

REFERENCES

- [1] Livemint, “India smartphone market to reach record 173 mn units in 2021: Report.” <https://www.livemint.com/technology/gadgets/india-smartphone-market-to-reach-record-173-mn-units-in-2021-report-11629464307978.html> (accessed Nov. 01, 2021).
- [2] C. Wilders, “Predicting the Role of Library Bookshelves in 2025,” *J. Acad. Librariansh.*, vol. 43, no. 5, pp. 384–391, 2017, doi: 10.1016/j.acalib.2017.06.019.
- [3] Y. Hendriana, “Development of Mobile Library Application Based on Android in Universitas Ahmad Dahlan,” *Int. J. Innov. Res. Sci. Eng. Technol. (An ISO)*, vol. 3297, 2007, doi: 10.15680/IJRSET.2015.0403055.
- [4] C. Software, “What is Flutter? Here is everything you should know | by Concise Software | Medium.” <https://medium.com/@concisesoftware/what-is-flutter-here-is-everything-you-should-know-faed3836253f> (accessed Nov. 02, 2021).
- [5] Y. Hendriana, “Development of Mobile Library Application Based on Android in Universitas Ahmad Dahlan,” *Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 4, no. 3, pp. 1064–1071, 2015, doi: 10.15680/IJRSET.2015.0403055.
- [6] Firebase, “Firebase.” <https://firebase.google.com/> (accessed Nov. 02, 2021).
- [7] D. Platform, “Community and support | Dart.” <https://dart.dev/community> (accessed Nov. 02, 2021).
- [8] G. B. APIs, “Google Books APIs | Google Developers.” <https://developers.google.com/books> (accessed Nov. 02, 2021).