



Zajel News

By

Abdelkarim Jasem Alghayyath

Supervisor

DR. Samer Nofal

**Thesis Submitted in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science**

German Jordanian University

School of Electrical Engineering and Information Technology

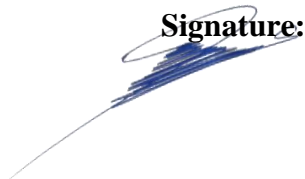
Department of Computer Science

January 2024

Authorization Form

I, Abdelkarim Alghayyath, authorize the German Jordanian University to supply copies of my to libraries, establishments or individuals on request, according to the Regulations of the German Jordanian University

Signature:

A handwritten signature in blue ink, appearing to be 'Abdelkarim Alghayyath', written over a horizontal line.

Date:13/1/2024

Dedication

With profound gratitude, I dedicate this accomplishment to my remarkable family. The unwavering support you've provided and the sacrifices you've made to facilitate my educational pursuit hold a place in my heart that words cannot fully capture. Your belief in my potential and the financial backing for my education have been the bedrock of my success.

This achievement is more than a personal victory; it's a celebration of our collective journey as a family. Your love and encouragement have propelled me forward, acting as a guiding force through every challenge. To my incredible family, you are the pillars of my strength, and this success is as much yours as it is mine. Here's to the shared efforts, the precious moments, and the enduring bonds that have shaped an unforgettable chapter in our lives.

As we raise a toast to this milestone, let it be a reminder that this is not the last accomplishment but the start of many more. Together, we embark on a journey filled with new challenges, victories, and the unwavering support that defines us as a family. Here's to the beginning of a new chapter, where our collective strength propels us towards even greater heights

Abdelkarim Jasem Alghayyath

Acknowledgments

I begin by humbly praising and thanking Allah Almighty, the Most Gracious, the Most Merciful. Throughout this journey, Your divine guidance and unwavering support have been the bedrock of my strength and perseverance. Without Your blessings, achieving this milestone would have been an impossibility. I stand deeply humbled before Your infinite grace, O Allah.

My utmost gratitude extends to my esteemed supervisor, Dr. Samer Nofal. Your invaluable mentorship, unwavering support, and insightful guidance have been instrumental in shaping this work. Your patience, encouragement, and willingness to share your knowledge have helped me refine my research and navigate the challenges I encountered. I am immensely grateful for your presence in my academic journey.

To my beloved parents and family, my heartfelt thanks and love. Your unwavering belief in me, your constant encouragement, and your endless sacrifices have been the wind beneath my wings. I am forever indebted to your love and support, which fueled my determination and kept me going even during the most challenging times.

My sincere gratitude to my friends Osama Alkhatib, Kareem Abuqaoud, Ibrahim Almatar and Mohammad Alsurakhi and many more who have been a constant source of support and encouragement. Your camaraderie and laughter have kept me grounded and energized throughout this process. Thank you for being my sounding board, my cheerleaders, and the anchors of my sanity.

Finally, I would like to thank the examiners and committee members for their time and insightful feedback. Your rigorous examination and valuable suggestions have helped me improve this work significantly. I am honored by your dedication to academic excellence and grateful for the opportunity to learn from your expertise.

Abdekarim Alghayyath

List of Figures

Figure 1 WaterFall Model	5
Figure 2 Sign-Up page.....	7
Figure 3 Login Page	7
Figure 4 News WebView	8
Figure 5 Feed Page-Popularity.....	8
Figure 6 Feed Page -Latest	8
Figure 7 Search Page-By Categorise	9
Figure 8 Search Page - By keyword.....	9
Figure 9 Bookmarks Page.....	10
Figure 10 Settings Page	10
Figure 11 Use Case Diagram	11
Figure 12 StatfulWidget code	12
Figure 13 Featching News getnews().....	12
Figure 14 Dropdown button.....	13
Figure 15 NewsTile widget	14
Figure 16 Simple dart class - NewsView	16
Figure 17 Importing Webview package	17
Figure 18 Using Webviwe Packages.....	17
Figure 19 NewsTile1 code.....	19
Figure 20 NewsTile code.....	19
Figure 21 NewsTile2 code.....	20
Figure 22 Firebase signin with wmeail/password.....	21
Figure 23 Firebase signup with email/password.....	22
Figure 24 Firestore data structure	23
Figure 25 Featching data from NewsApi	24
Figure 26 calling the featching function Getnews().....	25
Figure 27 calling the featching function Getnews().....	25

List of Contents

Authorization Form	i
Dedication.....	i
Acknowledgments.....	ii
List of Figures	iii
List of Contents	iv
Abstract.....	vi
Chapter 1 Introduction	1
1.1 Background:	1
1.2 Problem Statement:	1
1.3 Objectives:	2
1.4 Technologies Employed:	2
1.5 Summary:	2
Chapter 2 Zajel App	3
2.1 Overview:	3
2.2 Objectives:	3
2.2.1 Streamline News Consumption:	3
2.2.2 Enhance User Interaction:	4
2.2.3 Efficient Search Functionality:	4
2.2.4 Personalized Bookmarking:.....	4
2.2.5 Customization through Settings:.....	4
2.3 Workflow:	5
2.3.1 C Requirements Gathering:	5
2.3.2 Design:	6
2.3.3 Implementation	11

2.3.4	Testing	14
	Chapter 3 Technologies	15
3.1	Overview:	15
3.2	Flutter:	15
3.2.1	Dart Coding Language	16
3.2.2	Flutter Packages:	17
3.2.3	Advantages of Using Flutter	18
3.3	Firebase:	21
3.3.1	Firebase Authentication:	21
3.3.2	Cloud Firestore:	23
3.4	NewsApi:	23
3.4.1	Overview of News API Integration:	23
3.4.2	Fetching Data from NewsAPI (News Class):	24
3.4.3	Integration with Zajel News:	25
	Chapter 4 Conclusion	27
4.1	Conclusion:	27
4.1.1	Flutter: A Foundation for Excellence	27
4.1.2	Firebase: A Backend Powerhouse	27
4.1.3	News API: A Gateway to Dynamic Content	27
4.1.4	User-Centric Functionality: Bookmarks, History, and Notifications	28
4.2	The Future of Zajel News:	28
4.2.1	Personalized Feed and Enhanced Interaction	28
4.2.2	User-Generated Content: Empowering the Community	28
4.2.3	Building a News Ecosystem: Zajel News Backend	29
4.2.4	Enhanced User Empowerment and Accessibility	29
	Chapter 5 References	31

Zajel News

By
Abdelkarim Jasem Alghayyath

Supervisor
Dr. Samer Nofal

Abstract

Zajel News, developed using the Flutter framework and integrated with the NEWSAPI, aims to provide users with a seamless and user-friendly experience for accessing news from around the world. Leveraging Firebase, the app introduces personalized features, allowing users to bookmark articles of interest and receive notifications tailored to their preferences.

The primary objective of the app is to streamline the news consumption process, offering a centralized platform for users to stay informed about global events. The integration with NEWSAPI ensures a diverse and comprehensive selection of news articles, spanning various topics and regions.

One notable feature is the bookmarking system, enabling users to save articles for later reference. This enhances user engagement and facilitates a more personalized news-reading experience. Additionally, the app incorporates a notification system that keeps users updated with the latest articles from their favorite author, enhancing real-time information delivery.

The search functionality further empowers users to explore specific topics or events, providing a more targeted news browsing experience. The use of Flutter ensures a cross-platform application with a native-like feel, contributing to a consistent and polished user interface.

In conclusion, this News app represents a holistic solution for news consumption, combining a diverse news source through NEWSAPI, personalized features via Firebase, and an intuitive user interface using Flutter. The app addresses the need for a centralized, efficient, and user-centric news platform, catering to the preferences and habits of modern users.

Keywords: Flutter , Nativ App, NewsAPI, http, Firebase, UI/UX, Global News, Search, Bookmark, Notification.

Chapter 1

Introduction

1.1 Background:

In the digital age, where information inundates every aspect of our lives, the need for a refined, personalized, and efficient means of consuming news has become increasingly apparent. The fast-paced nature of today's world demands a news platform that not only centralizes diverse sources but also from all around the world, offering a seamless and engaging user experience. This backdrop sets the stage for Zajel News, a cutting-edge application that seeks to revolutionize the way users interact with and absorb news content..

1.2 Problem Statement:

In an era inundated with information, users face the daunting task of navigating a fragmented landscape of news sources, each housed in separate websites and applications. The inefficiency and inconvenience of having to access various platforms for diverse newsletters and news sources contribute to a disjointed and time-consuming user experience. Users are burdened with the challenge of manually aggregating content from multiple sources, hindering their ability to stay informed in a streamlined and personalized manner. Zajel News aims to address this pervasive issue by offering a unified platform that consolidates a diverse array of newsletters and news articles, providing users with a seamless, one-stop solution for their news consumption needs. Through this approach, Zajel News seeks to eliminate the need for users to open numerous websites and apps, offering a centralized hub that optimizes efficiency and enhances the overall user experience.

1.3 Objectives:

Beyond the immediate goal of developing a functional news application, this thesis strives to meet broader objectives. Zajel News aspires to streamline the news consumption process, ensuring that users can effortlessly navigate through a plethora of articles, providing a dynamic and responsive platform. By incorporating cutting-edge technologies, Zajel News aims to set new standards in user engagement and satisfaction within the realm of news consumption.

1.4 Technologies Employed:

Zajel News leverages the powerful Flutter framework, offering users a cross-platform application with a native-like feel. The integration with NEWSAPI is instrumental in providing a diverse and comprehensive selection of news articles, covering an extensive range of topics and regions. The incorporation of Firebase introduces personalized features, including an intuitive article bookmarking system and notifications tailored to individual preferences. This amalgamation of technologies ensures that Zajel News not only meets but exceeds the expectations of modern news consumers, offering a harmonious blend of accessibility, functionality, and personalization.

1.5 Summary:

In essence, Zajel News is not merely a news application; it is a paradigm shift in the way news is consumed. By synergizing the technological prowess of Flutter, the expansive coverage facilitated by NEWSAPI, and the user-centric approach powered by Firebase, this application is poised to redefine the news landscape. This thesis will provide a comprehensive exploration of the development, implementation, and evaluation of Zajel News, shedding light on its potential to emerge as a centralized, efficient, and user-centric news platform that anticipates and caters to the nuanced preferences of the modern news consumer.

Chapter 2

Zajel App

2.1 Overview:

In the landscape of news consumption, Zajel App emerges as a transformative platform designed to simplify and enhance the way users engage with news content. This section provides an exhaustive overview of Zajel App, emphasizing its role as a cutting-edge news application tailored to meet the evolving needs of modern consumers.

Zajel App functions as a centralized hub, seamlessly integrating diverse news sources to offer users a unified and personalized news consumption experience. The application harnesses the power of innovative technologies to prioritize user-centric design, accessibility, and efficiency in navigating a vast array of news articles. By amalgamating the technological prowess of Flutter, the expansive coverage facilitated by NEWSAPI, and the personalized features powered by Firebase, Zajel App transcends the conventional boundaries of news applications.

2.2 Objectives:

2.2.1 Streamline News Consumption:

Zajel App aims to streamline news consumption by offering a feature-rich Feed Page. This page provides users with the latest news updates and the option to explore the most popular news over various timeframes, including the past day, week, and month. The objective is to create an intuitive and dynamic feed that caters to users' preferences and keeps them informed with the most relevant and trending content.

2.2.2 Enhance User Interaction:

The application prioritizes user interaction by allowing seamless engagement with news articles on the Feed Page. Users can view the full articles, bookmark interesting news for future reference, and set notifications for news from their liked author. This objective ensures that users have a comprehensive and interactive experience, empowering them to customize their news consumption according to their interests.

2.2.3 Efficient Search Functionality:

Zajel App facilitates efficient news discovery through a dedicated Search Page. Users can explore news from their favorite categories or conduct specific searches to find articles of interest. This objective enhances the app's user-friendliness by providing a tailored and comprehensive search functionality, ensuring users can access relevant content with ease.

2.2.4 Personalized Bookmarking:

The application caters to users' preferences by offering a Bookmark Page. Here, users can revisit their bookmarked news articles. This objective enhances personalization, allowing users to curate their own collection of noteworthy articles for future reference and a more tailored news consumption experience.

2.2.5 Customization through Settings:

Zajel App provides users with control over their experience through a dedicated Settings Page. This objective allows users to customize their app preferences, such as notification settings, display options, and personalization features. The Settings Page ensures that users have a personalized and adaptable environment, creating a more user-centric and satisfying news consumption experience.

2.3 Workflow:

Zajel News, developed using the Waterfall software engineering model, follows a systematic and linear workflow. The development process unfolds in distinct phases, each contributing to the creation of a feature-rich and user-centric news application.

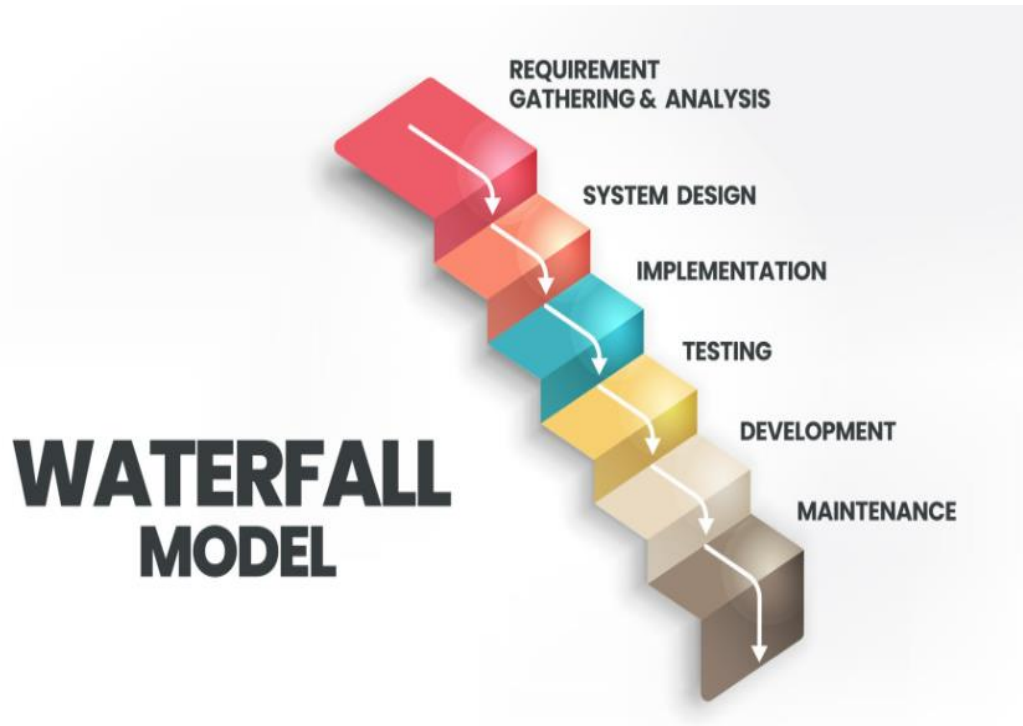


Figure 1 WaterFall Model

2.3.1 C Requirements Gathering:

The development journey begins with a thorough analysis of requirements. As the sole developer, you initiated this phase by identifying the key functionalities desired in Zajel News. This involved defining user stories, determining system requirements, and outlining the specific features such as the Feed Page, Search Page, Bookmarks Page, and Settings Page and their helping files.


2.3.2 Design:

In the design phase of the Zajel News application, meticulous specifications were established to define the architecture, user interface, and system components. A pivotal component of this phase was the creation of a use case diagram, offering a visual representation of high-level interactions between users and the Zajel News system. This diagram acted as a guiding blueprint, illustrating how users interact with pivotal features such as viewing news, bookmarking articles, and adjusting settings.

Crucially, Zajel News chose to implement its user interface using the Flutter framework. Flutter provides an extensive collection of widgets and tools, offering a powerful solution for developing cross-platform applications with a uniform and visually appealing design. Leveraging Flutter ensures that Zajel News delivers a seamless and responsive user experience across a diverse range of devices and platforms.

The decision to adopt Flutter reflects a deliberate choice to create a modern and dynamic application. Flutter's flexibility and efficiency in crafting expressive user interfaces align with Zajel News' commitment to providing a polished and engaging news consumption experience for its users. As we delve into the implementation section, we will explore how these design principles materialized in the form of tangible features within the Zajel News application.

User Interface:



Zajel News

Email Address


Password

Sign In

Or continue with

Don't have an account? [Register now](#)

Figure 3 Login Page



Welcome To ZAJEL

Email Address

Password

Confirm Password

Sign Up

Or continue with

Already have an account? [Login now](#)

Figure 2 Sign-Up page

Login/Sign up Pages:

Feed Page:

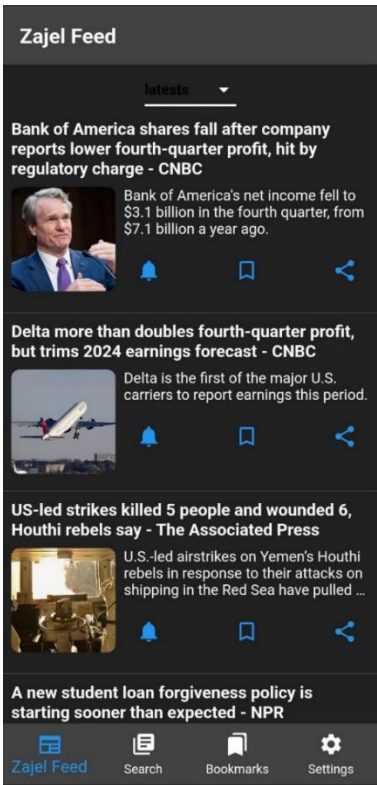


Figure 6 Feed Page -Latest

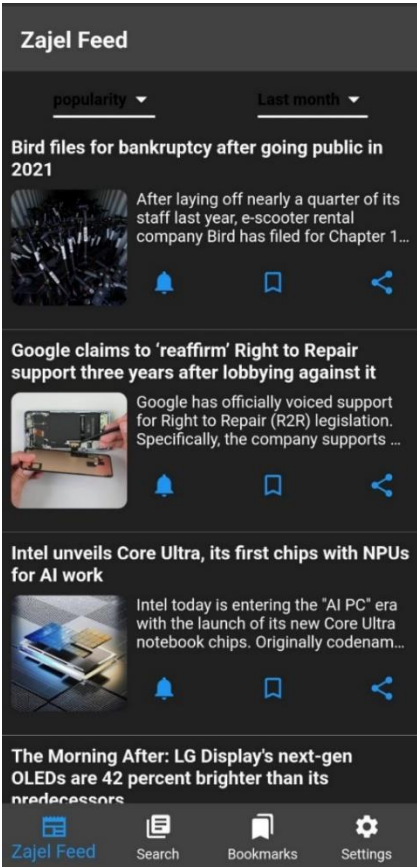


Figure 5 Feed Page-Popularity

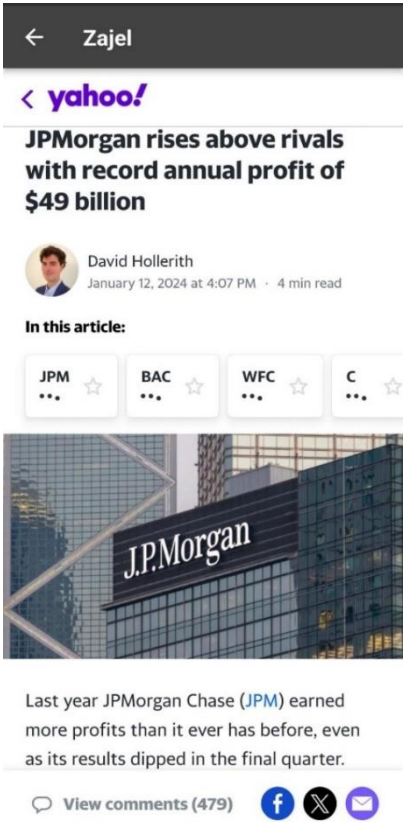


Figure 4 News WebView

Search Page:

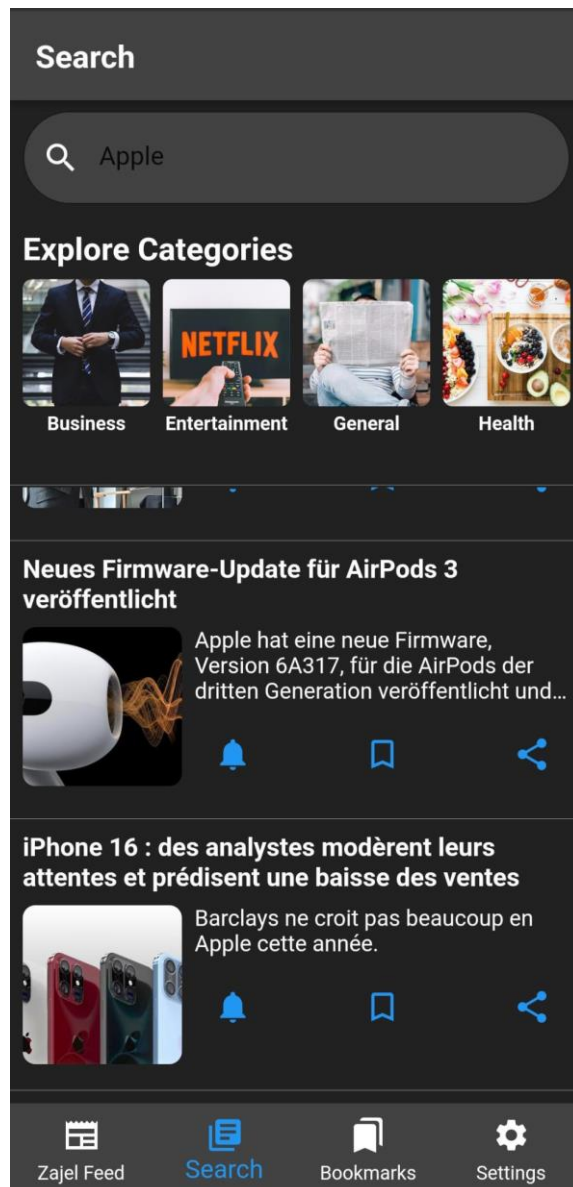


Figure 8 Search Page - By keyword

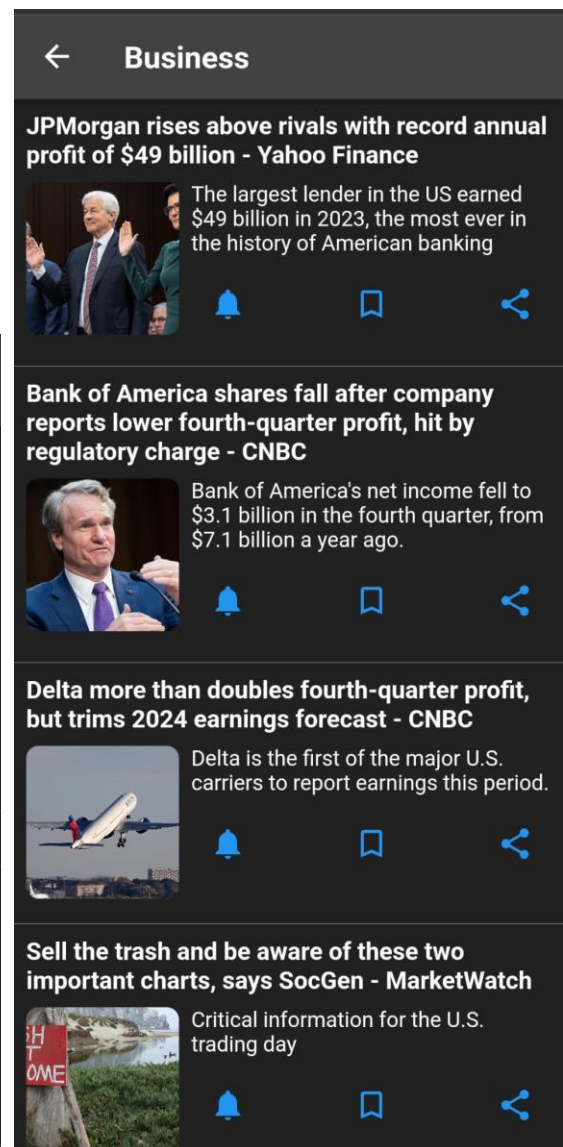


Figure 7 Search Page-By Categorise

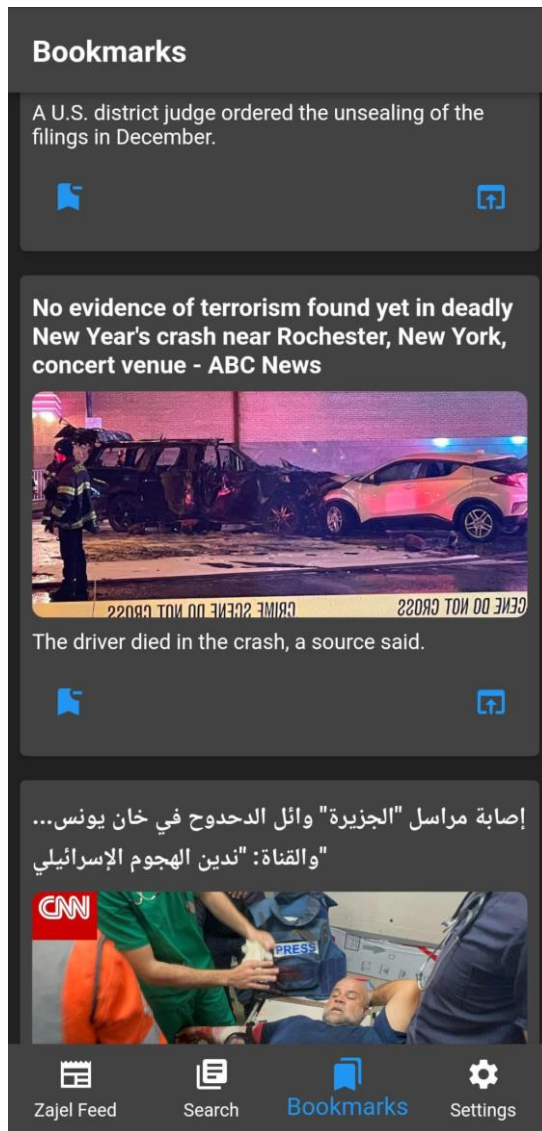


Figure 9 Bookmarks Page

Bookmark Page:

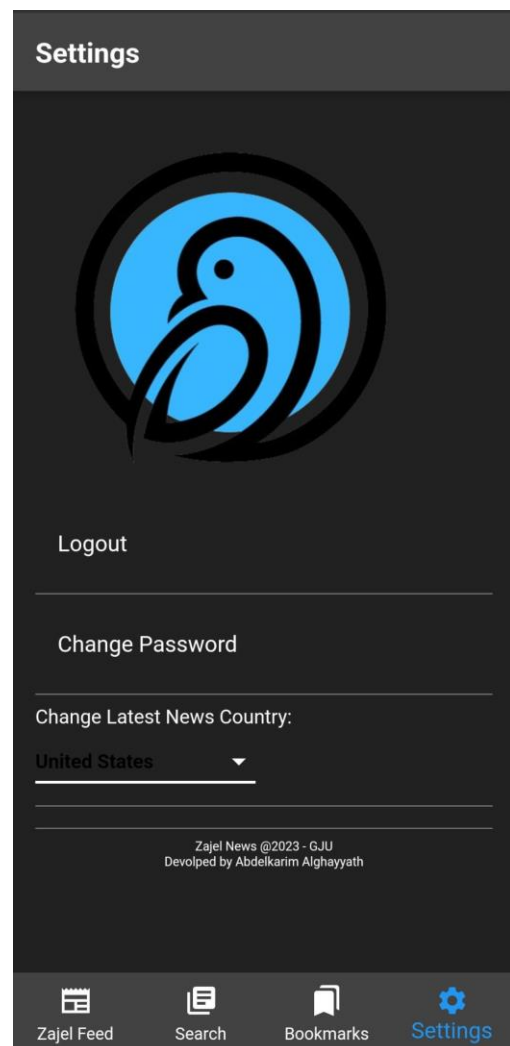


Figure 10 Settings Page

Settings Page:

Use Case Diagram:

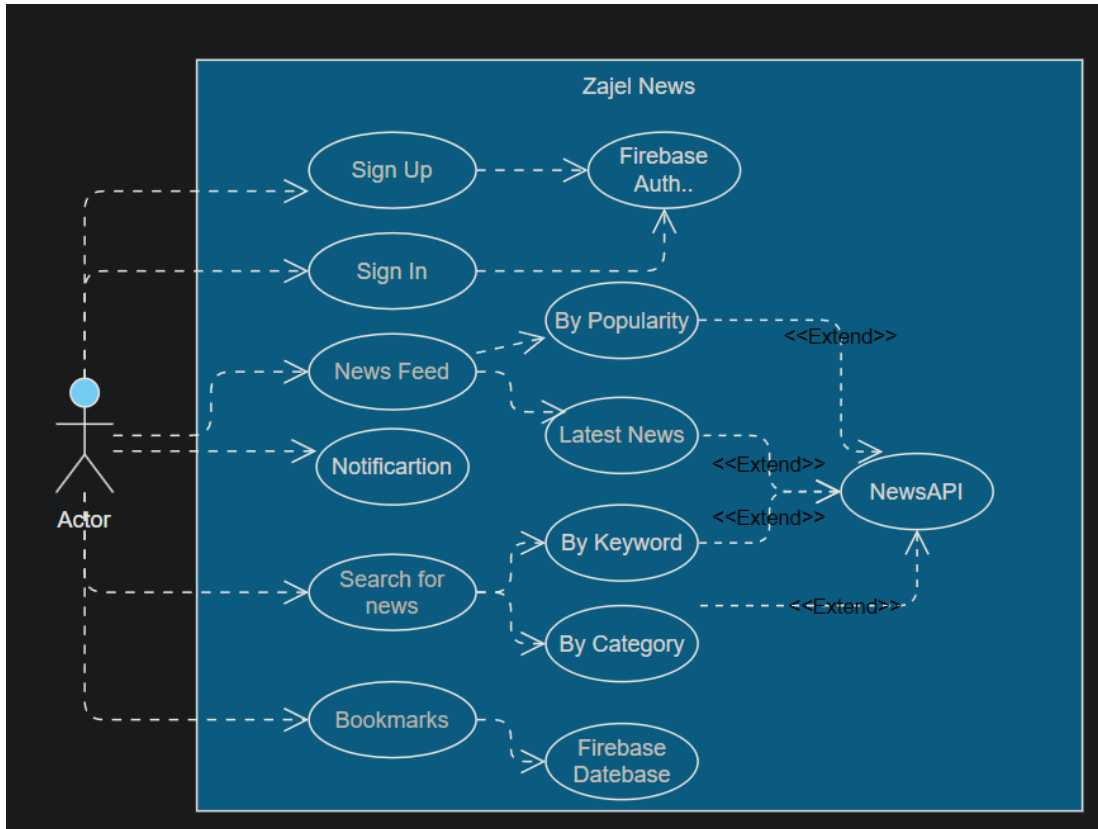


Figure 11 Use Case Diagram

2.3.3 Implementation

The implementation phase involved translating the design specifications into actual code. As the sole coder, I brought Zajeel News to life by writing and testing the software. The features outlined in the use case diagram, including the Feed Page, Search Page, Bookmarks Page, and Settings Page, were developed with attention to detail and adherence to the established design.

Feed Page Implementation:

The Feed Page was implemented as a stateful widget to manage the dynamic nature of the news feed, allowing for updates and changes in real-time.

```
class FeedPage extends StatefulWidget {  
  const FeedPage({Key? key}) : super(key: key);  
  
  @override  
  State<FeedPage> createState() => _FeedPageState();  
}  
  
class _FeedPageState extends State<FeedPage> {  
  bool _loading = true;
```

Figure 12 StatfulWidget code

Asynchronous Data Fetching:

Utilizing asynchronous programming, I implemented functions to fetch news data based on user preferences, such as sorting by 'latest' or 'popularity' and specifying time frames.

```
Future<void> getNews() async {  
  newslst.clear();  
  News newsclass = News();  
  await newsclass.getNews();  
  setState(() {  
    newslst = newsclass.news;  
    _loading = false;  
  });  
}
```

Figure 13 Featching News getnews()

Dropdown Menu Controls:

Implemented dropdown menus for sorting and filtering options, allowing users to choose between 'latest' and 'popularity' and select time frames for popularity-based sorting.

```
Row(
  mainAxisAlignment: MainAxisAlignment.spaceAround,
  children: [
    DropdownButton<String>({
      value: sortByValue,
      onChanged: (String? newValue) {
        setState(() {
          sortByValue = newValue!;
        });
        if (sortByValue == 'latests') {
          getNews();
        } else if (sortByValue == 'popularity') {
          timeFrameValue = 'Last day';
          getNewsByPopularity();
        }
      },
      style: TextStyle(color: Colors.white),
      icon: Icon(Icons.arrow_drop_down, color: Colors.white),
      underline: Container(
        height: 2,
        color: Colors.white,
      ), // Container
      items: <String>['latests', 'popularity']
        .map<DropdownMenuItem<String>>((String value) {
          return DropdownMenuItem<String>({
            value: value,
            child: Text(
              value,
              style: TextStyle(
                color: Colors.black,
                fontWeight: FontWeight.w700,
              ), // TextStyle
            ), // Text
          }); // DropdownMenuItem
        }).toList(),
    ), // DropdownButton
    if (sortByValue == 'popularity')
      DropdownButton<String>({
        value: timeFrameValue,
        onChanged: (String? newValue) {
          setState(() {
            timeFrameValue = newValue!;
          });
          getNewsByPopularity();
        },
        style: TextStyle(color: Colors.white),
        icon: Icon(Icons.arrow_drop_down, color: Colors.white),
        underline: Container(
          height: 2,
          color: Colors.white,
        ), // Container
        items: <String>['Last day', 'Last week', 'Last month']
          .map<DropdownMenuItem<String>>((String value) {
            return DropdownMenuItem<String>({
              value: value,
              child: Text(
                value,
                style: TextStyle(
                  color: Colors.black,
                  fontWeight: FontWeight.w700,
                ), // TextStyle
              ), // Text
            }); // DropdownMenuItem
          }).toList(),
      ), // DropdownButton
  ],
), // Row
```

Figure 14 Dropdown button

Dynamic News Tile Widgets:

Employed dynamic widgets, such as the **NewsTile** widget, to display individual news articles in the feed. Each tile contains interactive elements like buttons for notifications, bookmarking, and sharing.

```
// ...
Expanded(
  child: ListView.builder(
    itemCount: newslst.length,
    itemBuilder: (context, index) {
      return Column(
        children: <Widget>[
          NewsTile(
            newsimgurl: newslst[index].urlToImage,
            newstitle: newslst[index].title,
            newsdescription: newslst[index].description,
            newsurl: newslst[index].articleUrl,
            newsauthor: newslst[index].newsauthor.toString(),
          ), // NewsTile
          Divider(height: 1, color: Colors.grey),
        ], // <Widget>[]
      ); // Column
    },
  ), // ListView.builder
), // Expanded
```

Figure 15NewsTile widget

2.3.4 Testing

Following the implementation, thorough testing was conducted to ensure the functionality and reliability of Zajel News. This involved unit testing, integration testing, and system testing to identify and rectify any bugs or issues. User interaction scenarios depicted in the use case diagram were rigorously tested to guarantee a seamless experience.

Chapter 3

Technologies

3.1 Overview:

The technological foundation of Zajel News encompasses a synergy of powerful tools and frameworks. This chapter provides an in-depth exploration of each technology employed, with a special focus on Flutter as the primary cross-platform framework. This knowledge base is essential for comprehending the coding methodologies discussed in subsequent chapters.

3.2 Flutter:

Flutter, a powerful cross-platform framework developed by Google, serves as the foundation for Zajel News. This high-performance framework is built on the Dart programming language, offering a plethora of features for creating visually appealing and responsive applications.

In the Zajel News app, everything revolves around widgets. These customizable building blocks form the user interface and contribute to the app's dynamic and interactive design. Flutter's composability allows developers to create complex interfaces efficiently.

3.2.1 Dart Coding Language

Dart, the programming language behind Flutter, is an object-oriented language that supports key concepts such as classes and interfaces. This snippet illustrates a Dart class From Zajeł News for the `NewsView` class which will show the news in a webview:

```
1  import 'dart:async';
2  import 'package:flutter/material.dart';
3  import 'package:webview_flutter/webview_flutter.dart';
4
5  class NewsView extends StatefulWidget {
6    final String newsurl;
7    const NewsView({Key? key, required this.newsurl}) : super(key: key);
8
9    @override
10   State<NewsView> createState() => _NewsViewState();
11 }
12
13 class _NewsViewState extends State<NewsView> {
14   final Completer<WebViewController> _completer =
15     Completer<WebViewController>();
16
17   @override
18   Widget build(BuildContext context) {
19     return Scaffold(
20       appBar: AppBar(
21         title: const Row(
22           mainAxisAlignment: MainAxisAlignment.start,
23           children: <Widget>[
24             Text(
25               "Zajeł",
26               style: TextStyle(
27                 color: Colors.white,
28                 fontWeight: FontWeight.w700,
29               ), // TextStyle
30             ), // Text
31           ], // <Widget>[]
32         ), // Row
33         backgroundColor: Colors.grey[800],
34       ), // AppBar
35       body: Container(
36         child: WebView(
37           initialUrl: widget.newsurl,
38           onWebViewCreated: (WebViewController webViewController) {
39             _completer.complete(webViewController);
40           },
41         ), // WebView
42       ), // Container
43     ); // Scaffold
44 }
```

Figure 16 Simple dart class - `NewsView`

3.2.2 Flutter Packages:

Zajel News utilizes several Flutter packages to enhance functionality and streamline development. And as we saw in the previous Figure we will notice that we used the `webview_flutter` package:

```
lib > pages > hom
1  import library package:webview_flutter/webview_flutter.dart
2  import package:webview_flutter/webview_flutter.dart
3  import 'package:webview_flutter/webview_flutter.dart';
```

Figure 17 Importing Webview package

```
body: Container(
  child: WebView(
    initialUrl: widget.newsurl,
    onWebViewCreated: (WebViewController webViewController) {
      _completer.complete(webViewController);
    },
  ), // WebView
), // Container
```

Figure 18 Using Webviwe Packages

And the following packages have been integrated into the app:

- **firebase_core (v2.22.0):** Initialization and configuration of Firebase services.
- **firebase_auth (v4.14.1):** Secure user authentication for seamless sign-up and sign-in processes.
- **http (v1.1.0):** Smooth communication with external APIs, such as NewsApi, for real-time news content retrieval.
- **cloud_firestore (v4.13.6):** Integration of Cloud Firestore for efficient storage and management of user-specific data.
- **webview_flutter (v2.0.13):** Embedding of web content for a seamless user experience.
- **share (v2.0.4):** Implementation of sharing functionalities for easy news article sharing.

- **flutter_local_notifications (v16.3.0):** Implementation of local notifications for enhanced user engagement.

Flutter's vast collection of ready-made packages and its supportive, collaborative community make app development faster, easier, and more enjoyable. You can find solutions for almost any feature you need, learn from experienced developers, and contribute your own ideas to help others, all while staying at the forefront of mobile technology.

3.2.3 Advantages of Using Flutter

The implementation of Zajel News using Flutter brings forth several advantages:

- **Widget-based Structure:** Flutter's widget-based structure allows for easy customization and efficient UI creation.
- **Composability:** Composable widgets enable the development of an aesthetically pleasing interface within a reasonable timeframe.
- **Cross-Platform Compatibility:** Flutter's cross-platform compatibility ensures a consistent user experience across various devices.
- **Hot Reload:** Flutter's hot-reload functionality accelerates development by providing immediate code changes.

- **Customization:** The flexibility to create custom widgets, such as **NewsTile**, enhances the app's visual appeal and user interaction.

```

110  @override
111  Widget build(BuildContext context) {
112    return GestureDetector(
113      onTap: () {
114        savetoHistory();
115        Navigator.push(
116          context,
117          MaterialPageRoute(
118            builder: (context) => NewsView(
119              newsurl: newsurl,
120            ), // NewsView
121          ), // MaterialPageRoute
122        );
123      },

```

Figure 20 NewsTile code

```

127  crossAxisAlignment: CrossAxisAlignment.start,
128  children: <Widget>[
129    Text(
130      newstitle,
131      style: const TextStyle(
132        fontWeight: FontWeight.bold,
133        fontSize: 16,
134        color: Colors.white,
135      ), // TextStyle
136    ), // Text
137    SizedBox(height: 8),
138    Row(
139      crossAxisAlignment: CrossAxisAlignment.start,
140      children: <Widget>[
141        ClipRRect(
142          borderRadius: BorderRadius.circular(8),
143          child: Image.network(
144            newsingurl,
145            width: 100,
146            height: 100,
147            fit: BoxFit.cover,
148          ), // Image.network
149        ), // ClipRRect
150        SizedBox(width: 8),
151        Expanded(
152          child: Column(
153            crossAxisAlignment: CrossAxisAlignment.start,
154            children: <Widget>[
155              Text(
156                newsdescription,
157                maxLines: 3,
158                overflow: TextOverflow.ellipsis,
159                style: TextStyle(color: Colors.white),
160              ), // Text
161              SizedBox(height: 8),
162              // Icon buttons
163              Row(
164                mainAxisAlignment: MainAxisAlignment.spaceBetween,
165                children: <Widget>[
166                  IconButton(
167                    icon: Icon(
168                      Icons.notifications,
169                      color: Colors.blue,
170                    ), // Icon
171                    onPressed: () {
172                      saveNotificationToFirestore();
173                      ScaffoldMessenger.of(context).showSnackBar(
174                        SnackBar(
175                          content: Text('You will get Notifeid!'),
176                        ), // SnackBar
177                      );
178                    },
179                  ), // IconButton

```

Figure 19 NewsTile1 code

```

181         IconButton(
182             icon: Icon(
183                 Icons.bookmark_border,
184                 color: Colors.blue,
185             ), // Icon
186             onPressed: () {
187                 // Call the function to save news to Firestore
188                 saveNewsToFirestore();
189                 ScaffoldMessenger.of(context).showSnackBar(
190                     SnackBar(
191                         content: Text('News saved to Bookmarks'),
192                     ), // SnackBar
193                 );
194             },
195         ), // IconButton
196         IconButton(
197             icon: Icon(Icons.share, color: Colors.blue),
198             onPressed: () {
199                 Share.share(newsur1);
200             },
201         ), // IconButton
202     ], // <Widget>[]
203 ], // Row
204 ], // <Widget>[]
205 ), // Column
206 ), // Expanded
207 ], // <Widget>[]
208 ), // Row
209 SizedBox(height: 8),
210 ], // <Widget>[]
211 ), // Column
212 ), // Container
213 ); // GestureDetector
214 }
215 }
216

```

Figure 21 NewsTile2 code

3.3 Firebase:

Firebase, a comprehensive mobile and web application development platform by Google, plays a pivotal role in shaping the backend infrastructure of Zajel News. The integration of Firebase brings forth a range of services, offering solutions for authentication, real-time data storage, cloud functions, and more.

3.3.1 Firebase Authentication:

Firebase Authentication is utilized in Zajel News to ensure secure and seamless user experiences during the sign-up and sign-in processes. By leveraging Firebase Authentication, users can create accounts, log in securely, and access personalized features within the app.

The authentication process helps Zajel News establish a reliable and trusted user base. Firebase Authentication supports various sign-in methods, including email/password, Google Sign-In, and more. This flexibility enhances user accessibility and ensures a smooth onboarding experience.

```
13 class _LoginPageState extends State<LoginPage> {
14   final emailController = TextEditingController();
15   final passwordController = TextEditingController();
16
17   void userSignIn() async {
18     showDialog(
19       context: context,
20       builder: (context) {
21         return Center(
22           child: CircularProgressIndicator(),
23         ); // Center
24       },
25     );
26
27     try {
28       await FirebaseAuth.instance.signInWithEmailAndPassword(
29         email: emailController.text,
30         password: passwordController.text,
31       );
32       Navigator.pop(context);
33     } on FirebaseAuthException {
34       Navigator.pop(context);
35       showMessage('Email/Password are WRONG!');
36     }
37   }
38 }
```

Figure 22 Firebase signin with wmeail/password

```

14   final emailController = TextEditingController();
15   final confirmPasswordController = TextEditingController();
16   final passwordController = TextEditingController();
17
18   void userSignUp() async {
19     showDialog(
20       context: context,
21       builder: (context) {
22         return Center(
23           child: CircularProgressIndicator(),
24         ); // Center
25       },
26     );
27
28     try {
29       if (confirmPasswordController.text == passwordController.text) {
30         await FirebaseAuth.instance.createUserWithEmailAndPassword(
31           email: emailController.text,
32           password: passwordController.text,
33         );
34
35         // user creation successful
36         Navigator.pop(context);
37         showSuccessMessage('Account created successfully!');
38       } else {
39         Navigator.pop(context);
40         showErrorMessage('Passwords don\'t match!');
41       }
42     } on FirebaseAuthException catch (e) {
43       Navigator.pop(context);
44
45       //error handling show to user
46       switch (e.code) {
47         case 'weak-password':
48           showErrorMessage('The password provided is too weak.');
```

Figure 23 Firebase signup with email/password

3.3.2 Cloud Firestore:

Cloud Firestore serves as the primary NoSQL database solution for Zajel News. This scalable and flexible database allows the app to store user-related data, including bookmarks, user history, and preferences.

Zajel News leverages Cloud Firestore to create collections and documents that organize data efficiently. For instance, user bookmarks and history are stored in separate collections, ensuring a structured and easily accessible database.

```
// Example Firestore Document Structure
/user_bookmarks
  /{userId}
    /bookmarks
      /{articleId}
        title: "News Title"
        description: "News Description"
        imageUrl: "https://example.com/image.jpg"
        articleUrl: "https://example.com/article"
        author: "Author Name"
```

Figure 24 Firestore data structure

3.4 NewsApi:

3.4.1 Overview of News API Integration:

News API integration is a key element of Zajel News, providing a dynamic and constantly updated source of news articles from various publishers and sources. By leveraging the News API, Zajel News ensures that users have access to a diverse range of current news topics and headlines.

The News API serves as a bridge between Zajel News and multiple news sources, allowing the app to aggregate and display relevant articles based on user preferences and search criteria.

3.4.2 Fetching Data from NewsAPI (News Class):

The **News** class within Zajeel News is responsible for interfacing with the News API. This class employs the Dart programming language and utilizes the `http` package to make HTTP requests to the API endpoint. The retrieved data is then parsed and converted into **NewsModel** objects, which are used to populate the app's news feed.

Here is a snippet illustrating how the `getNews` function within the **News** class fetches data from the News API:

```
1  import 'dart:convert';
2  import 'package:zajel/models/news_model.dart';
3  import 'package:http/http.dart' as http;
4  import 'package:zajel/global/preferences.dart';
5
6  class News {
7    List<NewsModel> news = [];
8    String API = "ac0da06a6031439db63818970f0d7316";
9    Future<void> getNews() async {
10      // Clear the list before fetching new articles
11      news.clear();
12
13      String url =
14        "https://newsapi.org/v2/top-headlines?country=$Country&pageSize=100&apiKey=$API";
15      try {
16        var response = await http.get(Uri.parse(url));
17        if (response.statusCode == 200) {
18          var jsonData = jsonDecode(response.body);
19          if (jsonData['status'] == "ok") {
20            jsonData["articles"].forEach((element) {
21              if (element['urlToImage'] != null &&
22                element['description'] != null) {
23                NewsModel article = NewsModel(
24                  newsauthor: element["author"],
25                  title: element['title'],
26                  description: element['description'],
27                  urlToImage: element['urlToImage'],
28                  articleUrl: element["url"],
29                );
30                news.add(article);
31              }
32            });
33          }
34        } else {
35          throw Exception('Failed to load news');
36        }
37      } catch (e) {
38        print('Error fetching news: $e');
39      }
40    }
41  }
```

Figure 25 Featching data from NewsApi

3.4.3 Integration with Zajel News:

The `getNews` function seamlessly integrates into the **FeedPage** widget of Zajel News, ensuring that the app's news feed is consistently populated with the latest articles from the News API. The integration enhances the content diversity and timeliness of Zajel News, providing users with a rich and up-to-date news experience.

```
1  import 'dart:async';
2  import 'package:flutter/material.dart';
3  import 'package:zajel/global/newstitle.dart';
4  import 'package:zajel/global/news.dart';
5  import 'package:zajel/global/preferences.dart';
6  import 'package:intl/intl.dart';
7
8  class FeedPage extends StatefulWidget {
9    const FeedPage({Key? key}) : super(key: key);
10
11    @override
12    State<FeedPage> createState() => _FeedPageState();
13  }
14
15  class _FeedPageState extends State<FeedPage> {
16    bool _loading = true;
17    String sortByValue = 'latests';
18    String timeFrameValue = 'Last day';
19
20    @override
21    void initState() {
22      super.initState();
23      getNews();
24    }
```

Figure 26 calling the featching function Getnews()

```
63
64  Future<void> getNews() async {
65    newlist.clear();
66    News newsclass = News();
67    await newsclass.getNews();
68    setState(() {
69      newlist = newsclass.news;
70      _loading = false;
71    });
72  }
```

Figure 27 calling the featching function Getnews()

The integration of the News API ensures that Zajel News remains a reliable and comprehensive source of news information for users. By dynamically fetching articles from various sources, Zajel News maintains its relevance and keeps users engaged with fresh and diverse content.

Chapter 4

Conclusion

4.1 Conclusion:

In the culmination of this thesis, we have explored the development and architecture of the Zajel News application, a feature-rich news platform designed to provide users with a seamless news consumption experience. The integration of cutting-edge technologies, including Flutter, Firebase, and the News API, has been instrumental in crafting a robust and dynamic news application.

4.1.1 Flutter: A Foundation for Excellence

The choice of Flutter as the cross-platform framework has proven to be pivotal in achieving a high-performance and visually appealing user interface. Flutter's widget-based architecture, composability, and hot-reload functionality have expedited the development process, allowing for the creation of a feature-rich application with a single codebase. The Dart programming language, powering Flutter, has facilitated the implementation of customizable and responsive widgets, contributing to the overall fluidity of the Zajel News app.

4.1.2 Firebase: A Backend Powerhouse

Firebase has served as the backbone of Zajel News, providing a secure and scalable backend infrastructure. Firebase Authentication ensures user data security during sign-up and sign-in processes, while Cloud Firestore acts as a flexible and real-time NoSQL database for storing user-specific information such as bookmarks, user history, and notification preferences.

4.1.3 News API: A Gateway to Dynamic Content

The integration with the News API has elevated Zajel News to a dynamic and constantly evolving news platform. By fetching real-time news articles from diverse sources, the app ensures that users are presented with an extensive array of news content. The **News** class seamlessly interacts with the News API, providing a smooth and efficient process for retrieving and displaying the latest news articles.

4.1.4 User-Centric Functionality: Bookmarks, History, and Notifications

Zajel News goes beyond providing a simple news feed. The implementation of features such as bookmarks and user history, stored and managed through Firebase Cloud Firestore, enhances user customization and personalization. Users can save their favorite articles, revisit their reading history, and receive notifications tailored to their preferences, creating a tailored news experience.

4.2 The Future of Zajel News:

As Zajel News evolves, our vision extends beyond being a mere news consumption platform. The next phase of development is centered around providing users with an even more personalized and interactive experience, enhancing their engagement and allowing them to actively contribute to the platform.

4.2.1 Personalized Feed and Enhanced Interaction

The future of Zajel News envisions a highly personalized feed, tailoring news content to individual user preferences. Through advanced algorithms and user behavior analysis, the "For You" feed will curate news articles that align with a user's interests, ensuring a truly customized news consumption experience. Users will have the ability to interact with the content more actively, providing feedback through likes, comments, and shares.

4.2.2 User-Generated Content: Empowering the Community

In line with fostering a sense of community, Zajel News aims to empower users by allowing them to generate and share their own content. Users will be granted the capability to write and publish articles directly within the application. This user-generated content will form a dynamic part of the news ecosystem, enabling users to share their perspectives, insights, and unique stories. The interactive features of liking and commenting on user-generated articles will foster a sense of community engagement.

4.2.3 Building a News Ecosystem: Zajel News Backend

The ultimate future objective for Zajel News is to establish its own backend infrastructure, essentially owning and operating its News API. This backend system will employ advanced web scraping techniques to gather news from a diverse range of internet sources and newsletters. By curating content from various reputable outlets, Zajel News aims to provide users with a comprehensive and reliable news source, eliminating the need for external APIs and ensuring the platform's autonomy.

4.2.4 Enhanced User Empowerment and Accessibility

The future enhancements will not only revolve around content but also focus on empowering users with more control over their experience. Features such as personalized notification settings, advanced bookmarking options, and intuitive navigation will be introduced to enhance user accessibility. Zajel News aims to be more than a news aggregator; it aspires to be a platform where users actively shape and contribute to the news ecosystem.

4.3 Comparison

Zajel News drew inspiration from several news aggregator applications, each offering unique solutions to similar challenges. Among these, one of the most significant inspirations for Zajel News was Google News. In this feature comparison, we examine key aspects of both Zajel News and Google News to provide users with insights into the anticipated user experience and functionalities of Zajel News

Features	Zajel news	GoogleNews
Up to date News	Yes	Yes
Personalized Feed (For you page)	No	Yes
Customizable Feed page	Yes	No
Following Trends	No	Yes
Bookmarking	Yes	Yes
Sharing news	Yes	Yes
Push Notification	Yes	Yes
Searching by keyword	Yes	Yes
Searching by Categories	Yes	Yes
Extended categories	No	Yes

Chapter 5

References

<https://newsapi.org/docs>

<https://docs.flutter.dev/cookbook/networking/fetch-data>

<https://docs.flutter.dev/>

<https://dart.dev/guides>

<https://api.dart.dev/stable/3.2.3/dart-core/dart-core-library.html>

<https://pub.dev/packages/http>

https://pub.dev/packages/webview_flutter

https://pub.dev/packages/flutter_local_notifications

<https://github.com/flutter/flutter>

<https://www.reddit.com/r/FlutterDev/>

<https://firebase.google.com/docs/auth>

<https://firebase.google.com/docs/database>