GitHub Username: https://github.com/tanya-yu/

Lifestyle News

Description

The app provides articles from various sources related to your specific interest.

Intended User

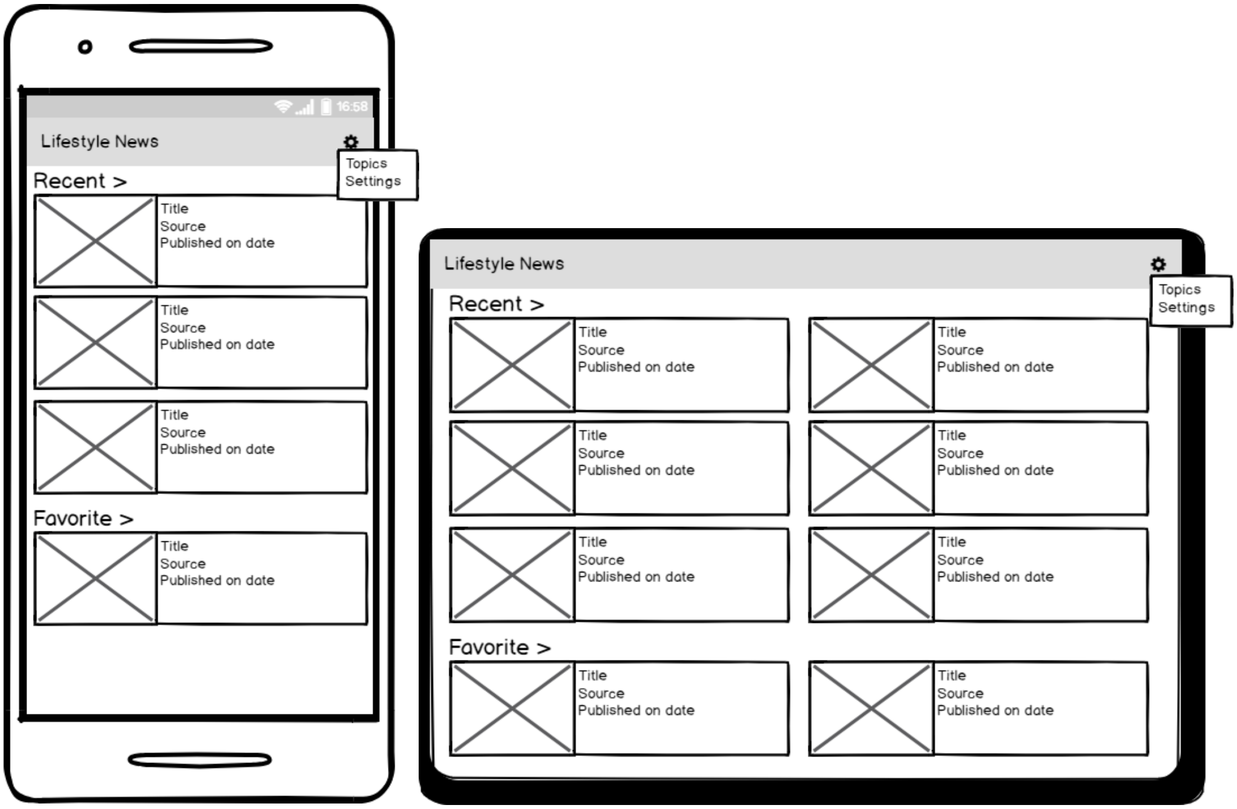
Everyone who has interest in specific topics.

Features

- Regularly loads articles from the Internet
- Saves new articles in the local database
- Deletes old articles from the local database
- Sends notifications when new articles are loaded
- Provides opportunity to choose topics of interest
- Provides opportunity to mark an article as “Favorite”
- Provides opportunity to share an article

User Interface Mocks

Screen 1 Main screen



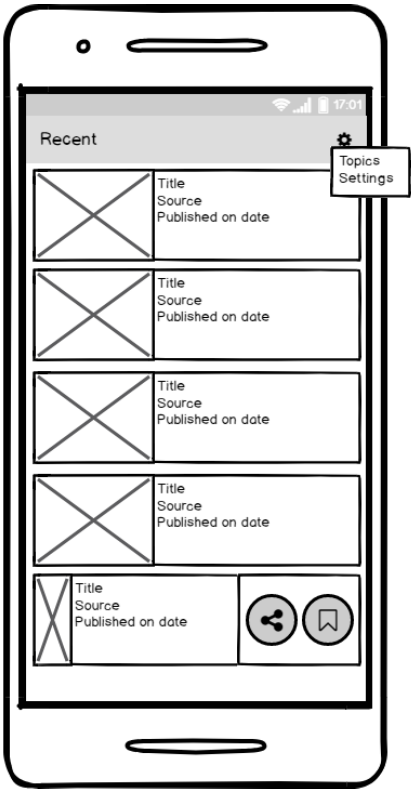
3 most recent articles and 3 latest added favorite articles are displayed on the main screen. For bigger devices this value equals 6.

Clicking on the article navigates to Article details screen. Clicking on Recent navigates to Articles list for recent articles, and clicking on Favorite navigates to Articles list for favorite articles.

User can refresh list by using swipe-to-refresh action.

Also, the main screen has menu with options: Topics (Screen 4 Topics settings) and Settings (Screen 5 Other settings).

Screen 2 Articles list



This screen can be used for displaying recent articles or favorite articles. The difference is in title in tool bar.

User can refresh list by using swipe-to-refresh action.

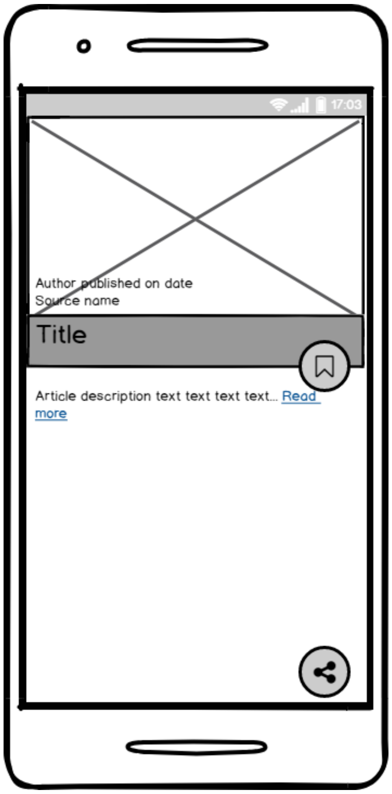
Sorting: recent articles are sorted by published date, favorite by added in favorite date.

This screen has the same menu as Screen 1 Main screen.

Each article has two swipe actions: Add to favorite and Share.

Also, the list includes native adds (free version only).

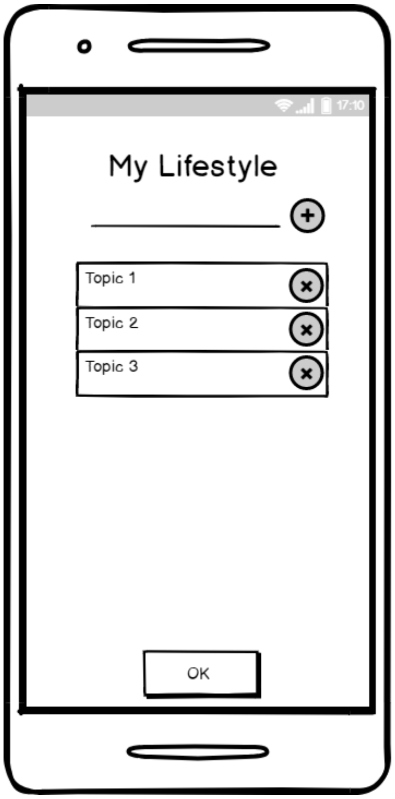
Screen 3 Article details



This screen displays the article content, its image, title, author, source name and published date. User can add article to favorite or to share it. The link “Read more” navigates user to the article website.

Sliding left navigated to the next article, sliding right navigates to previous article.

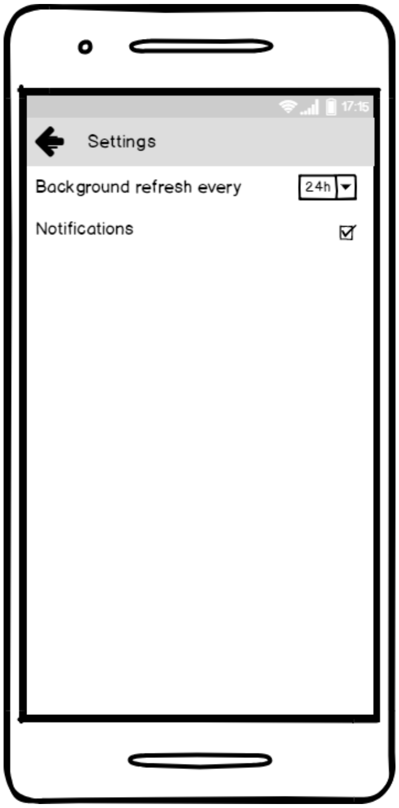
Screen 4 Topics settings



This screen is designed to manage interesting topics.

Button + adds topic in list, button – removes topic from list.

Screen 5 Other settings

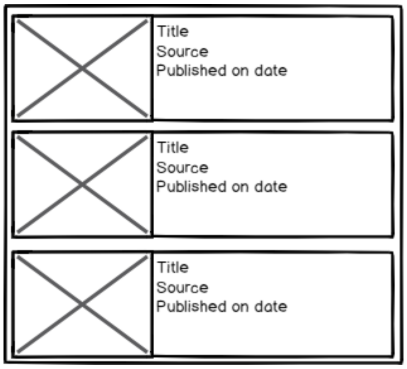


Background refresh every 24h

Notifications ☑

This screen allows user to choose background refresh period (2, 4,8,12, 24 or 48 hours) and switch notifications.

Screen 6 Recent or Favorite articles widget



Recent and Favorite articles widgets are different widgets, but they look the same. They displays 3 (for bigger devises 6) articles as on Main screen. Clicking on article navigates to Article details screen.

Key Considerations

How will your app handle data persistence?

The app will load articles from News API (https://newsapi.org/) as scheduled using a JobDispatcher and save them in the SQLite database. The local database stores data only for new articles (maximum 100) and for favorite ones, so the old articles will be deleted. For this purpose, I will build a Content Provider to insert, update and delete data from the database.

Describe any edge or corner cases in the UX.

When a user launches the app and there is no specific topics in the preferences (for example, the app is launching for the first time), at first the TopicsActivity should appear to let user to specify his interests, and only after that MainActivity is shown.

Describe any libraries you'll be using and share your reasoning for including them.

Library	**Reasons for including**
Picasso	To handle the loading and caching of images
Firebase JobDispatcher	To schedule background job to load articles
Android Support Library	To support additional important material design components and patterns

Describe how you will implement Google Play Services or other external services.

Service	**Reasons for including**
Google Mobile Ads	To handle Content Native Ads in articles list for free version
Android Framework JobScheduler	To implement Job-Scheduling to load articles in the background

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

- Create project
- Configure libraries
- Create paid and free flavors

Task 2: Implement UI for Each Activity, Fragment and Widget

Create layouts and navigation for each Activity, Fragment and Widget

Subtasks:

- Build UI for MainActivity
- Build UI for ArticlesListActivity
- Build UI for ArticleDetailsActivity
- Build UI for ArticleDetailsFragment
- Build UI for TopicsActivity
- Build IU for SettingsActivity
- Build UI for SettingsFragment
- Build UI for RecentArticlesWidget
- Build UI for FavoriteArticlesWidget

Task 3: Implement Content Provider

The app stores recent articles and favorite articles in SQLite database. They can be stored in one table ArticleEntry:

Name	**Description**	**Type**	**Null/Not null**	**Key**
_id	Id	integer	Not null	PK
source_name	Name of the article’s source	varchar(200)	Null	
author	Author’s name	varchar(500)	Null	
title	Title of the article	varchar(500)	Null	
description	Article content	text	Null	
url	Url to article	varchar(500)	Null	
url_to_image	Url to article image	varchar(500)	Null	
published_at	Date, when article was published	datetime	Not null	
favorite	Favorite mark	boolean	Not null	
favorite_date	Date when article was marked as “Favorite”	datetime	Null	

Subtasks:

- Create ArticleEntry
- Create ArticlesDbHelper
- Create ArticlesProvider which will insert, delete and update articles

Task 4: Implement FirebaseJobService

The app regularly loads new articles in the background (refresh period is set up by user in Settings screen). The app stores only 100 recent articles, so the oldest ones must be deleted. It does not affect Favorite articles, they would not be deleted. Also, user can refresh articles list manually using SwipeToRefresh action.

Subtasks:

- Create SyncArticlesFirebaseJobService
- Create SyncArticlesTask
- Create SyncArticlesIntentService

Task 5: Implement MainActivity and ArticlesListActivity

Implement functional for displaying loaded articles in RecyclerView in MainActivity and ArticlesListActivity.

Subtasks:

- Create ArticlesAdapter

Task 6: Implement ArticleDetailsActivity

Implement functional for displaying an article and actions.

Subtasks:

- Implement “Add to Favorite” action
- Implement “Share” action
- Implement “Read more” action

Task 7: Implement TopicsActivity

Implement functional allowing user to specify his interest.

Subtasks:

- Implement “Add Topic” function
- Implement “Remove Topic” function

Task 8: Implement SettingsActivity

Implement functional allowing user to choose Background refresh period and switch Notifications.

Task 9: Include Google AdMob

Google AdMob native ads should be included in ArticlesListActivity and only in free version.

Task 10: Implement Notifications

Notifications appears after background refreshing is done. Notification shows how many new articles is loaded.

Task 11: Implement RecentArticlesWidget and FavoriteArticlesWidget

Recent and Favorite widgets look the same, but they show different data. RecentArticlesWidget displays 3 recent articles, FavoriteArticlesWidget displays 3 favorite articles.