GitHub Username: ioannisa

Neakriti News Application

Description

An application to provide articles from the greek news agency neakriti.gr. I have already written in the very old days (2012) this application which is already outdated, and I try to update every now and then, but it lacks proper writing, design (material design), widgets, fragments, TV Stream, storing articles, and so on.

We have concluded with my employer that this application needs to be rewritten (the right way).

The application currently manages day and night theme, firebase remote configuration, firebase push messages (notification), Live Radio Streaming.

By rewriting this application as a capstone project will excel my skills and allow me to remove some of the burden of “time”, as I will be able to code the project at my office, as my employer is aware and has agreed to complete the capstone project during my working hours, since it will serve as a real project for the company neakriti.gr.

The current android version of the app (written solely by me) can be found in the following URL:
https://play.google.com/store/apps/details?id=com.ioannisa.rssreadernk

The iOS version (written solely by me using objective-c) can be found in the following URL:
https://itunes.apple.com/us/app/id486803221

Intended User

The app is intended for the average greek news reader, giving him the ability to explore local news from the neakriti.gr news agency.

Features

- Displays news Articles (html formatted) in a WebView via downloaded Feeds
- Allows Users to save favorite news articles for later reading
- Firebase Remote Configuration for various changes without the need to reinstall the app
- Firebase push Messaging will enable notifications for
    - Important Articles opened directly on Application
    - Things other than articles that need to open on Browser instead
    - Important update notifications targeting specific problematic builds
- Text to Speech on articles content to aid visually impaired people
- Live Radio Stream of the “Radio984” radio station (part of the same enterprise as neakriti.gr)
- Live TV Stream of CreteTV station (part of the same enterprise as neakriti.gr)
- Day and Night themes
- Preferences regarding text size, theme, article categories, etc
- Targeting Phone and Tablets via fragments
- Material Design principles
- Home Screen Widget showing the top articles
- At the end of each article there will be an AdMob horizontal banner
- Support article Text to Speach (TTS) via the eSpeak app (for support on TTS for Greek language)
    - eSpeak app can be found here https://play.google.com/store/apps/details?id=com.marvin.espeak

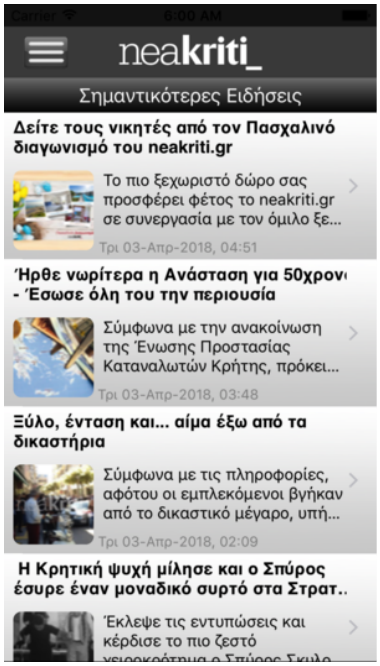
Accessibility

There are some accessibility features (included in the above features list) that will help certain user groups.

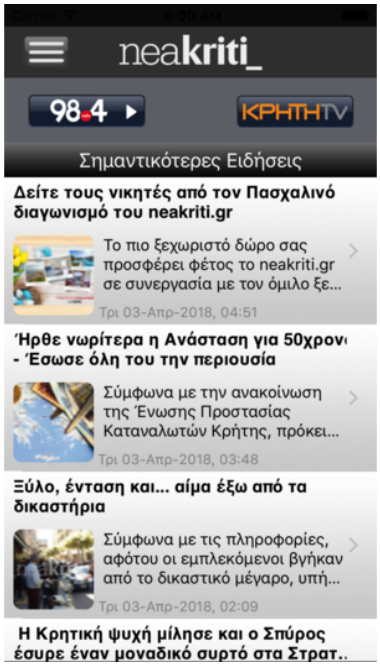
- Text-to-Speech on articles vis eSpeak (where available) will help the visually impaired on article content reading.
- Variable Font sizes (specified inside the preferences fragment) will allow users to change the font size for the content of the articles to allow more comfort while reading.
- The availability of day and night themes will also provide some extra comfort to the users and the dark theme may be more relaxing for people sensitive to bright lights.

User Interface Mocks

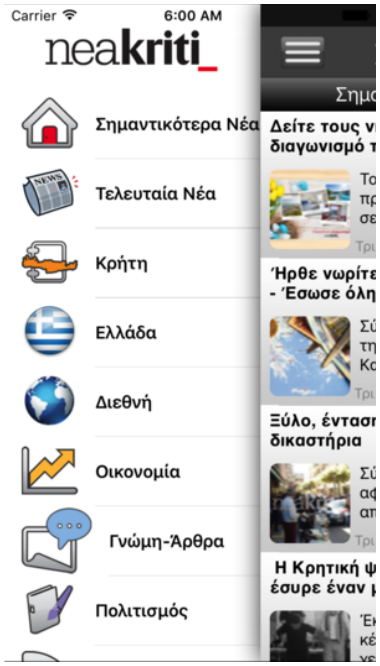
Main Screen



Main Screen



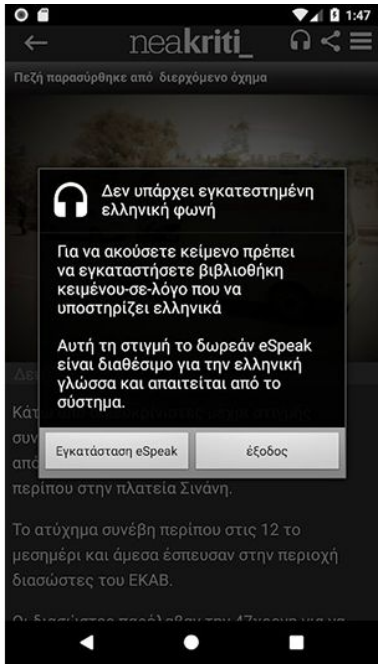
Extending Streams



App Drawer

Detail Screen

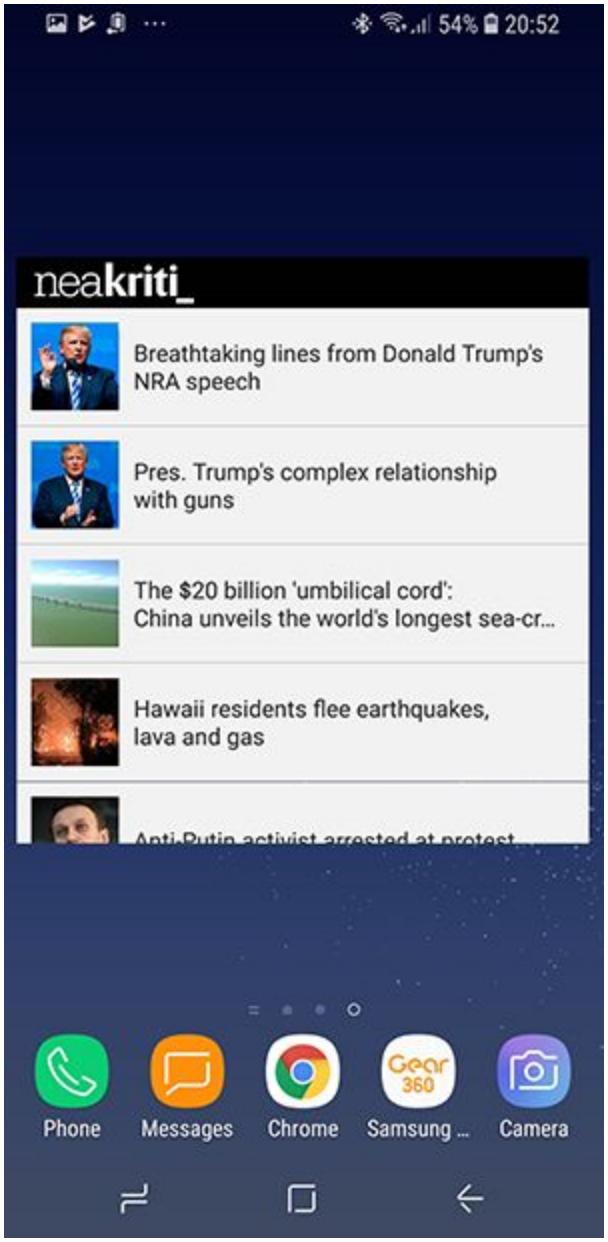
Night Reading



TTS via eSpeak

Detail Screen

Widget



A widget containing a listview of the top articles will be available *(the above mock is a photoshop altered result from the original widget of CNN news application)*.

Key Considerations

How will your app handle data persistence?

1. Application will store favorite articles via a Content Provider in SQLite
2. Shared Preferences for standard preferences (like text size preferences)

Describe any edge or corner cases in the UX.

1. Moving back from the detail activity will return back to the main activity.
2. Moving back from the Video Stream activity will terminate the video stream
3. Moving back from the main activity will return back to the home screen.
    a. If the radio stream is playing, it will continue to play, but an ongoing notification will allow easy access back to the radio.

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

1. Google's Compatibility Library will be used to ensure compatibility with older android versions
2. Picasso (and/or Glide) will be used for caching and displaying the application images.
3. Retrofit (may possibly be used) for downloading the news feeds.
4. Data Binding will be used to eliminate any findViewById calls in adapters and activities.
5. ExoPlayer will be used for the live streaming of TV and Radio
6. OkHttp will be used for some network connections
7. Firebase Messaging library will be used for implementation of push notifications
8. Firebase Remote Config will be used to allow updating parts of the application without the need to download updates from the Google Play.

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

1. Firebase Cloud Messaging will be used for implementing Push Notifications
2. Firebase Remote Configuration will enable updating several app settings remotely
3. AdMob banners will appear at the end of each article

Next Steps: Required Tasks

Task 1: Project Setup

Create parsing mechanism for the Feeds provided

Example Feed articles from category with id 1500 (default is 20 articles per category)
https://www.neakriti.gr/webServices/MobileFeedAndroid_v2.aspx?srvid=1500

Example Feed for the 30 most recent items from the category with id 1500
https://www.neakriti.gr/webServices/MobileFeedAndroid_v2.aspx?srvid=1500&items=30

Example Feed for fetching just the article with id 1460380
https://www.neakriti.gr/webServices/MobileFeedAndroid_v2.aspx?docid=1460380

The same article on the neakriti.gr website:
https://www.neakriti.gr/article/eidiseis/1460380/sto-irakleio-to-google-devfest-greece-2017/

Task 2: Implement UI for Each Activity and Fragment

1. Create Master Activity and an equivalent Fragment
2. Create layout for displaying each item in the list of articles of Main Activity
3. Create a Detail Activity and an equivalent Fragment
4. Implement Shared Element Transitions between Master and Detail activity main image
5. Create Phone/Tablet arrangements between Master/Detail fragments
6. Create Menu items and Preferences Fragment
7. Build night theme on top of the standard (day) theme
8. Add Article Sharing on Detail Activity
9. Enable Text-to-Speech (TTS) on detail activity (using eSpeak app for Greek TTS support)
10. Create Preferences Fragment that will allow the user to
    a. Define preview lines in articles list
    b. Define available categories
    c. Define font sizes
    d. Enable/Disable night theme
    e. Enable/Disable notifications topics for Firebase Cloud Messaging.

Task 3: Populate RecyclerView and Store articles for offline reading

Activities will be populated with data. Article featured images will be cached for 2 days to save data during frequent usages of the app or when revisiting categories.

The articles content is in HTML so it can be parsed by a WebView in the Detail. A stripped version (text only) will be created and saved in SQLite database via Content Providers for each of the first 20 articles of each visited category being loaded (again as long as the latest load is not before the previous 2 days).

Thus the articles available for offline reading will not contain any internal images or iframes or any other kind of tags referencing external information. They will contain only the textual representation of the article and the featured article icon.

Finally, there will be implemented an option to “save article for later reading”. This final option will allow to store the article so it can be viewed later, either as offline (textual only) article, or with full content (internal icons, iframes etc) whenever the device is connected at the internet.

Task 4: Handle Live Video and Audio Streams

There will be two different media types supported. One for the Live streaming of the TV Channel (Crete-TV) and the other for the Live Radio Streaming (Radio 984).

Opening the TV Stream will occupy the entire screen (will be displayed in landscape mode only) on an activity and hitting back, will close the activity, thus terminating the TV Streaming.

Opening the Radio Stream will produce an ongoing notification (notification that cannot be dismissed). Even when the app is closed clicking this notification will open the player activity.

We will need an additional permission to check the call status, so if there is an incoming phone call the live streams will automatically stop playing.

Task 5: Add Firebase support to application

Will add Firebase push notifications
Will add Remote Configuration

*** Statement:**
App will be written solely in the Java Programming Language
App will utilize stable release version of all libraries, Gradle, and Android Studio.

*** Statement**
All the libraries being used, together with Gradle and Android Studio versions will belong to stable versions.

*** Statement**
There will be no hardcoded strings. All strings, colours etc will be stored in their equivalent resource xml files (e.g. strings.xml)