
GitHub Username: alexakasanjeev

Žodis

Description

Žodis app can be very helpful in learning English Root Words.This application enables us to learn a list of commonly used root words, their meanings and some examples of words that was formed from the root words.

Instead of memorizing word meaning, it is lot easier to learn root word. This App has Root Words with their explanations and this can be very helpful for improving English Vocabulary.
This Žodis app has following features

* Offline Application
* Add important Root Word to Favorite list
* Quiz to practise what you learnt
* Pronunciation of word

Intended User

This app is targeted towards student community. It has been shown students learn better by gamification rather than mugging words. This app is perfect mixture of knowledge and fun quizzes to improve your vocabulary..

Features

- Practice 5 minutes a day
- Fun quiz to learn efficiently
- Unlocked root words will be displayed separately
- A widget showing all the root words learnt

User Interface Mocks

LauncherActivity

Launcher Activity shows a RecyclerView List containing all the levels user can play and learn.

Initially there will be 10 to 12 level, where each level will contain 5-6 root words along with its meaning and use cases. After the user has seen all root words, user can take a small quiz containing 8 to 10 questions related to what just learnt.

Insight Activity

In this tab it will show all the roots word user has learnt, and also their will be a filter button to show favourite words.

Level Activity (Detail Activity)

Each level will contain 5 to 6 root word, in this each page will show 1 root word with its meaning and use case. User can click next to show more root word.

After seeing all root word user can hit click quiz to practice all the root words.

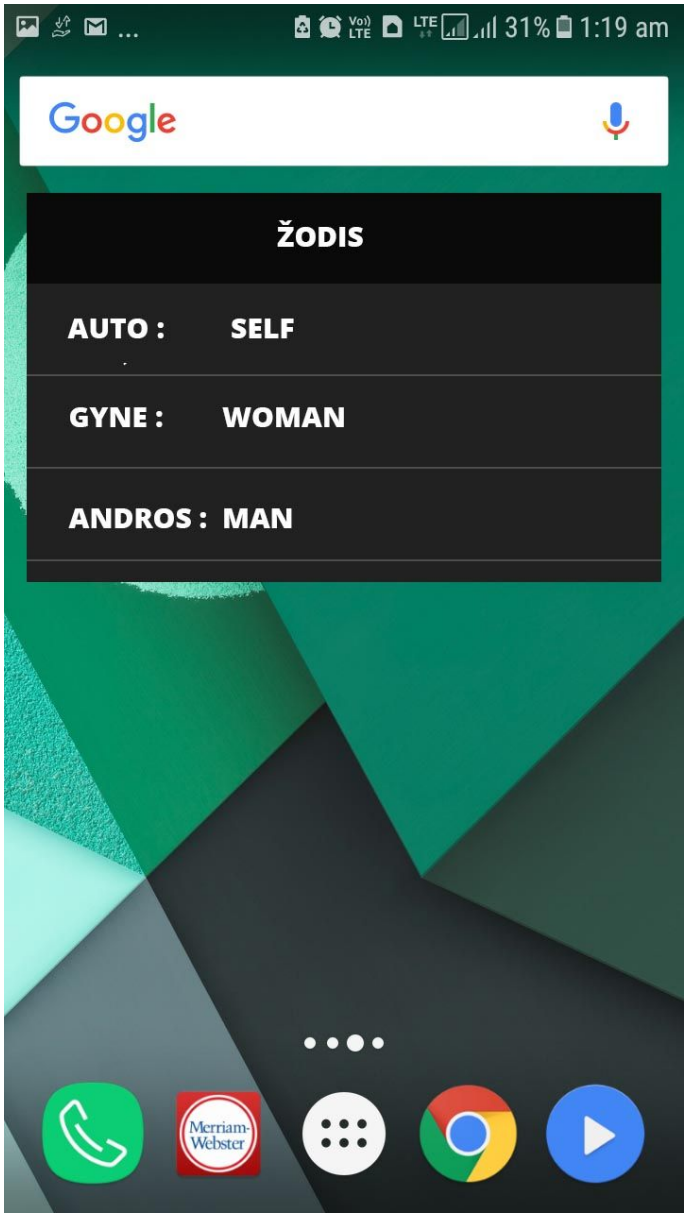
Quiz Activity

Quiz activity will ask 8 to 10 questions containing questions related to root word of the level selected.

Type of question :-

1. Select the correct meaning
2. Guess the root word
3. Word will be shown and tell it root word

After user has successfully completed quiz. All the root words of this level will be shown on Insight Activity as mention above.



Widget Ui

Widget will be shown on screen which will show all the root words user has learnt.

Key Considerations

How will your app handle data persistence?

There will be a local sqlite database which will store all the root words related information on table. Content Providers will be used to populate the RecyclerView in the launcher activity. All the data will be stored locally.

Activities and Fragments will fetch data by using **AsyncTask** to avoid blocking UI thread.

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

ButterKnife : It will help to avoid writing unnecessary code and help me focus on the app's core features.

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

1. Google Mobile Ads : Dummy ads will be shown below the screen.
2. Google Account Login : User has to login with their google account. This will be used to share quiz score on their Google+ profile.

Next Steps: Required Tasks

Task 1: Project Setup

In this step I will create new project and define all the dependencies needed during production of the app..

Configure dependencies :

- Define ButterKnife dependency in Gradle
- Define Google Account Login and Google Admob dependency in Gradle
- Sync project

Task 2: Implement Content Provide

Create local database and create necessary classes for Content Provider.

Implement the following classes:

- Define all the Contracts and Uris in a ZodisContract
- Define a Helper class for SQLiteOpenHelper named ZodisDbHelper
- Define ZodisContentProvider and define all CURD operations

Since data is cached locally, Activities and fragment will use **AsyncTask** to load data in background and update the UI when data is loaded with the help of ContentResolver.

Task 3: Implement Google Account login service and create Login Activity

Subtask

- Use google account login to help user login with their google account
- Define Network permission in AndroidManifest.xml
- Create NetworkUtil class that will help check Internet connection
- Save login state if user successfully logged in

Task 4: Implement UI for Each Activity and Fragment

Subtask:

- Build UI for MainActivity
    - Contains a RecyclerView
    - Use Loader to fill RecyclerView
- Build UI for LevelActivity
    - Contains a ViewPager to swipe
    - Display all root word for that level
- Build UI for QuizActivity
- Build UI for InsightTab
    - Contains a RecyclerView to show all the root words learnt
- Configure **Material Design** Theming for the application in styles.xml
- Define transitions between activities

Task 5: Create a Widget

Widget will be used to show all the root words user has learnt. Clicking on widget will launch the App. Widget will use ContentProvider to get data.

Task 6: Create Notification

Create a helper class to schedule a notification to remind user to practise each day. User will be able to disable notification in settings.

Task 7: Create Signing Configuration

App is equipped with a signing configuration, and the keystore and passwords are included in the repository. Keystore is referred to by a relative path.

Task 8: Implement optional components for Rubric

Implement the following optional components :-

- Share button to share quiz score on Google+

Thank You