

НАВЧАЛЬНА СИСТЕМА ДЛЯ ВИВЧЕННЯ І ТЕСТУВАННЯ З ПОЛЬСЬКОЇ МОВИ

Вінницький національний технічний університет

Анотація

Розглянуто особливості та ключові рішення, прийняті при розробці системи для вивчення польської мови «Polaung». Описано головний патерн програмування, який ліг в основу додатку, а також використані технології та сучасні бібліотеки для розробки андроїд додатків: Spring Framework, Spring Boot, Spring Data, Kotlin, Spring MVC, Viewpager2, Rest, Maven, Java Profiler.

Ключові слова: навчальна система, андроїд-додаток, Spring Boot, Android, Client-Server, Java.

Abstract

This article outlines the process of creating the Polaung program, taking into account some of the challenges we faced and the choices we made. The article emphasizes the basic software design template used in the project, and other technologies such as: Spring Framework, Spring Boot, Spring Data, Kotlin, Spring MVC, Viewpager2, Rest, Maven, Java Profiler.

Keywords: training system, android application, Spring Boot, Android, Client-Server, Java.

Вступ

Зараз, як ніколи, люди, які хочуть вивчити нову мову, звертаються за допомогою до своїх мобільних пристроїв, оскільки програми для вивчення мови стають все доступнішими. Такі програми дозволяють користувачам вивчати нову мову з будь-якого місця і в будь-який час, вони також є досить ефективними.

«Незважаючи на те, що мільйони людей у всьому світі вже користуються програмами для вивчення мови, опублікованих досліджень щодо їхнього впливу на розмовні навички немає», – говорить Шон Луевен, професор кафедри лінгвістики германських, слов'янських, азійських та африканських мов Університету штату Мічиган [1].

Тому він провів власні дослідження, в яких його учні вивчали нову мову за допомогою телефонного додатку. «Загалом учні в цьому дослідженні підвищили свій рівень володіння усною мовою, що вимірюється покращенням на добре встановленому та дійсному тесті на мовленнєвому інтерв'ю», – сказав Луевен. Ці результати встановлюють, що використання додатків для вивчення мов може сприяти розвитку навичок усного спілкування, а не лише засвоєння граматики та словникового запасу.

Серед висновків звіту зазначається, що 59% учасників покращили рівень володіння усною мовою принаймні на один підрівень за шкалою рівня володіння мовою, визнаною американською радою з викладання іноземних мов. Частка учнів, які покращили свої знання, зросла вище серед тих, хто більше часу використовував додаток. Для тих, хто навчався щонайменше 6 годин, 69% респондентів підвищили принаймні один підрівень показників, а 75% тих, хто навчався щонайменше 15 годин, покращили свій рівень ще суттєвіше [1].

У наш час вивчення інших мов – це вже є необхідною потребою. Відомим є твердження: «Скільки мов ти знаєш, стільки разів ти людина».

Розробка навчальної системи «Polaung»

Polaung – це набір програм, що включає в себе такі додатки: веб-сервер та Android-програму для вивчення і тестування з польської мови.

Навчальна система складається з наявного функціоналу для вивчення нових слів та фраз польської мови, для знайомства зі стандартними граматичними конструкціями. Для моніторингу прогресу навчального процесу система «Polaung» має тестову частину, яка спрямована на тестування отриманих знань та фіксування поточних досягнень користувача.

Система є багатофункціональною. В її основі лежить Rest сервіс, який видає інформацію, що обробляється та приходить у додаток. Під час побудови плану розробки проєкту виникало питання, які патерни програмування використовувати. Вибір стояв між такими відомими шаблонами: Client-Server, Layered та Pipe-filter і Blackboard, Model-View-ViewModel.

Client-Server – це шаблон, який складається з двох частин: сервера і кількох клієнтів [2-4]. Серверний компонент надає послуги кільком клієнтським компонентам. Крім того, сервер продовжує слухати запити клієнтів під час роботи (див. рис. 1).

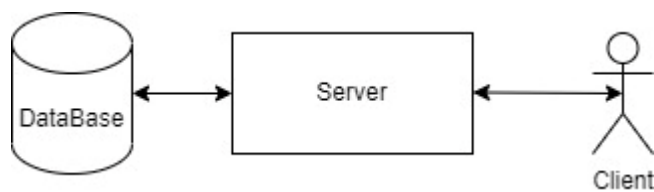


Рис.1. Схема роботи патерну програмування Client-Server [2]

Використання такого принципу дозволяє чітко розділити роботу над поданням (інтерфейсом програми) і роботу над самою програмою. Також частково вирішується проблема взаємодії з різними типами користувачів, які використовують різні види пристроїв. Як наслідок, вся складна робота відбувається на сервері, а на виході маємо перетворені дані. Цей патерн підходить під завдання з написання Rest server, оскільки використовується саме для написання невеликих застосунків за малий проміжок часу.

Model-View-ViewModel – це патерн, який часто використовується при розробці андроїд додатків. Він є досить зручним для розуміння програмістам, оскільки всі його компоненти логічно розділені. Основними компонентами шаблону MVVM є:

- Model – абстрагує джерело даних, ViewModel працює з DataModel, щоб отримати та зберегти дані;
- View – інформує ViewModel про дії користувача;
- ViewModel – надає потоки даних, що мають стосунок до View.

Однією з ключових переваг є те, що класи нащадки не мають прямого посилання на класи наслідування, вони мають лише посилання за допомогою «observables». Автоматично оновлюється інтерфейс користувача, коли змінюється модель перегляду, потрібно лише оголосити властивості моделі як observables, оскільки це спеціальні об'єкти JavaScript, які можуть сповіщати про зміни та можуть автоматично виявляти залежності.

Сучасні технології, що використовувалися під час розробки системи

Для розробки серверної частини використано Spring Framework останньої стабільної версії, а також його допоміжні компоненти Spring Boot та Spring Data для реалізації REST API та для управління даними.

Використано систему управління реляційними базами даних SQLite [5]. На етапі розробки всі дані зберігаються на локальному сервері бази даних. Дані зберігаються у вигляді таблиць.

Мобільна програма створена середовищем побудови та тестування мобільних застосунків Android Studio [6] на мові програмування Java та Kotlin. Це дозволило розробку програмного забезпечення для інтенсивного вивчення іноземних слів, їх фільтрацію та відображення у зручній для мобільного користувача формі. Під час розробки використано сучасні бібліотеки:

- Java Util, що містить основні рішення примітивних завдань;

- Spring-android-rest-template – для роботи з Rest в Android розробці;
- Data Binding API – для перетворення JSON у POJO;
- Appcompat – для використання спеціальних реалізацій ActionBar та Material Design.

Активно використовується система управління версій Git [7]. З її допомогою відбувається написання коду, його порівняння, злиття та рефакторинг. За допомогою віддаленого репозиторія на GitHub відбувається запис повної історії розробки веб-серверу та андроїд додатку.

Висновок

Розроблена навчальна система «Polaung» орієнтована на швидке та ефективне вивчення польської мови. Розробка велася з метою полегшити потенційним користувачам доступ до інформації у вигляді зручного інтерфейсу з багатим комплектом допоміжних слів та фраз. Окрім того, було досліджено декілька потужних сучасних технологій у розробці андроїд-додатків.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Shawn Loewen, Daniel R. Isbell, Zachary Sporn, The effectiveness of app-based language instruction for developing receptive linguistic knowledge and oral communicative ability 1 [Електронний ресурс] Режим доступу до ресурсу: <https://onlinelibrary.wiley.com/doi/10.1111/flan.12454>.
2. Клієнт-сервер архітектура [Електронний ресурс] Режим доступу до ресурсу: <https://buklib.net/books/23148/>.
3. Найважливіші архітектурні шаблони, які необхідно знати [Електронний ресурс] Режим доступу до ресурсу: <https://devzone.org.ua/post/nayvazhlivishi-arkhitekturni-shablони-yaki-neobkhdno-znati>.
4. Клієнт-серверна архітектура та ролі серверів. [Електронний ресурс] Режим доступу до ресурсу: <https://medium.com/@IvanZmerzlyi/клієнт-серверна-архітектура-та-ролі-серверів-9893d8048229>.
5. SQLite [Електронний ресурс] Режим доступу до ресурсу: <https://www.sqlite.org/>.
6. Android Studio [Електронний ресурс] Режим доступу до ресурсу: <https://developer.android.com/studio>.
7. Git [Електронний ресурс] Режим доступу до ресурсу: <https://git-scm.com/>.

Войтко Вікторія Володимирівна – кандидат технічних наук, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: dekanfki@i.ua.

Ракитянська Ганна Борисівна – кандидат технічних наук, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: rakit@vntu.edu.ua.

Бевз Світлана Володимирівна – кандидат технічних наук, доцент кафедри електричних станцій і систем, Вінницький національний технічний університет, м. Вінниця, e-mail: svbevz@i.ua.

Бурбело Сергій Михайлович – кандидат технічних наук, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: smburbelo@gmail.com.

Маніч Андрій Михайлович – студент групи ЗПІ-18б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: anderss1918@gmail.com.

Viktoriia Voitko – Ph.D., Associate Professor of Software Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: dekanfki@i.ua.

Anna Rakytyanska – Ph.D., Associate Professor of Software Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: rakit@vntu.edu.ua.

Svitlana Bevz – Ph.D., Associate Professor, Department of Power Plants and Systems, Vinnitsa National Technical University, Vinnitsa, e-mail: svbevz@i.ua.

Sergii Burbelo – Ph.D., Associate Professor of Software Engineering, Vinnitsa National Technical University, Vinnitsa, e-mail: smburbelo@gmail.com.

Andrii Manych – student of group ЗПІ-18b, Faculty for Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnitsa, e-mail: anderss1918@gmail.com.