

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
ІНСТИТУТ МЕХАНІКИ ТА АВТОМАТИКИ АПВ НААН
ДЕРЖАВНИЙ БІОТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ



***ЗБІРНИК
ТЕЗ ДОПОВІДЕЙ***

*XI Міжнародної науково-технічної конференції з нагоди
117-ї річниці від дня народження
доктора технічних наук, професора,
віцепрезидента УАСГН
КРАМАРОВА
Володимира Савовича
(1906-1987)*

«КРАМАРОВСЬКІ ЧИТАННЯ»

*22-23 лютого 2024 року
м. Київ*

УДК 631.17+62-52-631.3

Збірник тез доповідей XI Міжнародної науково-технічної конференції «Крамаровські читання» з нагоди 117-ї річниці від дня народження доктора технічних наук, професора, віцепрезидента УАСГН Крамарова Володимира Савовича (1906-1987) 22-23 лют. 2024 р., м. Київ / МОН України, Національний університет біоресурсів і природокористування України. К.: Видавничий центр НУБіП України, 2024. 505 с.

Proceedings of the XI International Scientific and Technical Conference dedicated to the 117th anniversary of the birth of Doctor of Technical Sciences, Professor, Vice President of the UAAS Kramarov Volodymyr Savovych (1906–1987), February 22–23, 2024, Kyiv / MES of Ukraine, National University of Life And Environmental Sciences of Ukraine. Kyiv: Publishing center of NULES of Ukraine, 2024. 505 p.

В збірнику представлені тези доповідей науково-педагогічних працівників, наукових співробітників, аспірантів та студентів НУБіП України, провідних вітчизняних і закордонних вищих навчальних закладів та наукових установ, в яких розглядаються завершені етапи розробок.

The Proceedings presents abstracts of reports of scientific and pedagogical workers, research staff, graduate students and students of the NULES of Ukraine, leading domestic and foreign higher educational institutions and scientific institutions, in which completed stages of development are considered.

УДК 004.5

**МОДЕЛЮВАННЯ ТА ТЕХНІЧНЕ ПРОЄКТУВАННЯ
ІНФОРМАЦІЙНОЇ СИСТЕМИ**

П. О. МАЛЬЧЕНКО, магістр

Г. О. ІВАНОВ, канд. техн., наук, доцент,

П. М. ПОЛЯНСЬКИЙ, канд. екон, наук, доцент,

Миколаївський національний аграрний університет. м. Миколаїв, Україна

E-mail: twink1337zhaba@gmail.com, ivanovgo0708@gmail.com,

polyansky.pasha@gmail.com

Метою роботи є інформація про зовнішній вигляд застосунку, його основний функціонал, розподіл функцій по активностям та вигляд представлень для роботи з таблицями на мобільному пристрої.

The purpose of the work is information about the appearance of the application, its main functionality, the distribution of functions by activity and the type of views for working with tables on a mobile device.

Ключові слова: застосунок, основний функціонал, активності, інтерфейс, авторизація, шаблони, наповнення.

Keywords: application, basic functionality, activities, interface, authorization, templates, content.

Підвищення зручності та спрощення процесу отримання інформації щодо відвідувань та успішності студентів шляхом створення за стосунку, електронного журналу із підтримкою офлайн режиму роботи, розглянуто в роботах [1] і [2]..

Розробка інтерфейсу застосунку. При розробці зовнішнього вигляду застосунку перш за все потрібно розібратися з яких основних частин він складатиметься, визначити активності та їх наповнення. Застосунком будуть користуватися як викладачі, так і студенти, а, отже, найбільш логічним буде розподіл функціоналу на дві частини: викладацька та студентська. Функціонал також буде відрізнятися: у студентів доступ до інформації лише у форматі читання, тому інтерфейс обох частин відрізнятиметься і кольоровою палітрою.

Зустрічати будь-якого користувача буде форма авторизація для отримання інформації з бази даних. Кожен користувач має вводити логін та пароль, а також обрати групу, до якої відноситься. Виходячи з цих простих умов, активність авторизації виглядатиме наступним чином.

На сторінку авторизації також потрібно додати логотип або назву застосунку, а також його автора. Після авторизації починається робота безносе-редньо з таблицями. Залежно від користувача, наповнення активностей виглядають по-різному. Почнемо з боку викладача. У кожного викладача є групи та дисципліни, які він/вона викладають у цих групах. Обравши групу та дисципліну, переходимо до занять, що проводяться у відповідний день, місяць та рік.

Звичайно, тут же має бути можливість додати нове заняття, а також переглянути результати роботи усіх студентів групи з вибраної дисципліни за семестр. В кожному занятті є студенти, яким потрібно виставити оцінки та присутність (рис. 5). Обравши потрібного студента, викладач має можливість виставити оцінку за обране у попередніх активностях заняття, а також, за необхідності, додати коментар до роботи.

Як показує практика, такі коментарі є дуже важливими, особливо, якщо використовувати умовні скорочення. Також, у випадку проведення декількох занять з однієї і тієї ж дисципліни в один день, викладач має можливість додати додаткову інформацію про розподіл балів за кожне з занять. Такий функціонал повністю задовольняє потреби викладача у виставленні оцінок, оскільки є можливість виставляти оцінки та відвідування студентів, як і у паперовій версії журналу, а також залишати нагадування про проблемні моменти вибраного студента і надалі проводити аналіз таких моментів, аби покращувати перебіг вивчення дисципліни. По закінченню семестру, постає питання яку кількість балів набрав кожен студент, а також, як багато занять було пропущено. У відповідній активності знаходиться інформація про всіх студентів та їх результати, де, у певних випадках, є можливість безпосередньо перейти до вибраного заняття, та змінити оцінку, якщо це потрібно.

Тепер потрібно показати роботу застосунку зі сторони студента. Обравши відповідну роль, та пройшовши авторизацію, студент отримує список дисциплін, що вивчаються в цьому семестрі. Також, одразу можна переглянути

статистику за семестр та отримати рекомендації щодо пріоритетності дисциплін до найшвидшого успішного завершення. Перейшовши до активності обраної дисципліни, у студента є інформація про загальну кількість балів за семестр, а також, інформація по кожному заняттю, що було проведено з цієї дисципліни. Переглянувши активності студентів, можна побачити, що вони відображають усі основні пункти, що наявні у паперовій версії журналу, а також, присутня додаткова активність з рекомендаціями та статистичними даними, що дозволяють контролювати успішність особисто студентів та визначати сильні і слабкі сторони, що дозволить ефективніше визначати можливі сторони розвитку вузько направлених вмінь.

Розробка зовнішнього вигляду списків та табличних представлень. Завдяки можливостям баз даних, є можливість створювати вибірки даних, що будуть розподілені між декількома активностями, в той час, як на комп'ютері це було б в

табличному представленні. Питанню створення баз даних та їх під'єднання до застосунку буде приділено увагу у наступних розділах, а зараз потрібно обмовити створення шаблонів для виведення даних з баз даних. Створення шаблонів дозволяє працювати з різними наборами даних однаково, що значно полегшує процес роботи із застосунком взагалі. Звичайно, дані мають обиратися одного типу. Тому, для кожного представлення треба створити шаблони, які будуть наповнюватися даними з бази. Для зручного користування застосунком було обрано елемент `Recycler View`, що призначений покращити продуктивність, у порівнянні зі стандартним `List View`. `RecyclerView` дозволяє легко ефективно відображати великі набори даних. Ви надаєте дані та визначаєте, як виглядає кожен елемент, а бібліотека `RecyclerView` динамічно створює елементи, коли вони потрібні [3]. Як впливає з назви, `Recycler View` переробляє ці окремі елементи. Коли елемент прокручується з екрана, `Recycler View` не руйнує його вигляд.

Натомість `Recycler View` повторно використовує подання для нових елементів, які прокручуються на екрані. Це повторне використання значно покращує продуктивність, покращуючи швидкість реагування програми та зменшуючи споживання енергії. Визначивши елемент у коді, його обробкою займатиметься окремий адаптер, що визначатиме усі елементи, які входять до обраного списку і дозволить взаємодіяти з кожним окремо. Так, наприклад, ПБ студента має строковий вигляд, а присутність студента на занятті визначається `CheckBox`. В самій базі даних ці поля мають різні типи, а тому і представлені у списку незалежно один від одного. Таке використання шаблонів дозволяє звернутися до усього списку, надавши йому інформацію з бази даних, а саму інформацію розподілити по простим елементам окремо і налаштувати вигляд таким чином, щоб максимальна кількість інформації була надана, в той же час її об'єм значно менший, якщо порівнювати з табличним представленням. Тобто, отримуючи повний обсяг інформації, кожна активність дозволяє працювати саме з певними елементами таблиці, що полегшує навігацію на порівняно маленьких екранах смартфонів.

Висновки. 1. При розробці зовнішнього вигляду застосунку перш за все потрібно розібратися, з яких основних частин він складатиметься, визначити активності та їх наповнення.

2. Застосунком будуть користуватися як викладачі, так і студенти, а, отже, найбільш логічним буде розподіл функціоналу на дві частини: викладацька та студентська. Функціонал також буде відрізнятися: у студентів доступ до інформації лише у форматі читання, тому інтерфейс обох частин відрізнятиметься кольоровою палітрою. Після авторизації починається робота безпосередньо з таблицями. Залежно від користувача, наповнення активностей виглядають по-різному.

3. Завдяки можливостям баз даних, є можливість створювати вибірки даних, що будуть розподілені між декількома активностям, в той час, як на комп'ютері це було б в табличному представленні.

4. Створення шаблонів дозволяє працювати з різними наборами даних однаково, що значно полегшує процес роботи із застосунком взагалі. Звичайно, дані мають обиратися одного типу. Тому, для кожного представлення треба створити шаблони, які будуть наповнюватися даними з бази. Для зручного користування застосунком було обрано елемент *Recycler View*, що призначений покращити продуктивність, у порівнянні зі стандартним *List View*.

Recycler View дозволяє легко ефективно відображати великі набори даних. Ви надаєте дані та визначаєте, як виглядає кожен елемент, а бібліотека *Recycler View* динамічно створює елементи, коли вони потрібні.

Список використаних джерел

1. Мальченко П.О. Створення зручного застосунку для проведення контролю успішності учнів для шкіл, та вирішення деяких проблем з уже існуючими системами для університетів / П.О. Мальченко, М.Л. Дворецький, Г.О. Іванов // Перспективна техніка і технології – 2021 : матеріали XVII-ї Міжнародної науково-практичної конференції молодих учених, аспірантів і студентів, 21-23 вересня 2021 р., м. Миколаїв, Миколаївський національний аграрний університет. – Миколаїв : МНАУ, Том 1. 2021. – С. 5–10.

2. Мальченко П.О. Застосування інформаційних технологій в освіті / П.О. Мальченко, М.Л. Дворецький, Г.О. Іванов // Перспективна техніка і технології – 2021 : матеріали XVII-ї Міжнародної науково-практичної конференції молодих учених, аспірантів і студентів, 21-23 вересня 2021 р., м. Миколаїв, Миколаївський національний аграрний університет. – Миколаїв : МНАУ, Том 1. 2021. – С. 10–12.

3. *RecyclerView*. URL: <https://metanit.com/java/android/5.11.php> (дата звернення 27.05.21).