Пловдивски университет „Паисий Хилендарски”

Факултет по математика и информатика

Курсова работа

По дисциплина „Дизайн на мобилни приложения“

На тема: „Мобилно приложение за бордова игра“

| Изготвил: Божидар Ламбов  Специалност: СТД III.  Факултетен номер: 2201681038. | Оценка ( )  Проверил:  / доц. д‐р Н. Касъклиев/ |
| --- | --- |

СЪДЪРЖАНИЕ

**Увод**

1. **Описание на приложението**

**"Bojo's DnD App"** е мобилно приложение, създадено за помощ на играчите на *Dungeons & Dragons (D&D)*. То предоставя удобен начин за дигитално управление на персонаж (герой) в рамките на настолната игра – включително характеристики на героя, инвентар (предмети) и други свързани данни. Целта на приложението е да бъде личен помощник на играча по време на игра: вместо хартиени листове, потребителят има дигитален лист на героя, както и достъп до магазин за предмети и оборудване. Приложението е разработено като самостоятелен проект и е насочено към това да предложи основната функционалност, нужна за един D&D персонаж, в удобно за мобилно устройство оформление.

2. **Използвани технологии и мотивация**

Приложението е изградено за **Android** с помощта на **Android Studio** и езика **Kotlin**. Android Studio е официалната интегрирана среда за разработка (IDE) за Android и предоставя на разработчиците всички необходими инструменти – симулатор, гъвкава система за билдване Gradle, шаблони за код и др. – в една унифицирана работна среда.

Kotlin е съвременен статично типизиран език за JVM, който през 2017 г. беше обявен от Google като официално поддържан език за разработка на Android приложения. Изборът на Kotlin пред Java бе мотивиран от няколко ключови предимства на езика: на първо място, Kotlin позволява писане на по-ясен и кратък код, с по-малко шаблонни конструкции, което повишава продуктивността на разработчика​.

Езикът има вградена null-безопасност (null safety), която помага да се избегнат *NullPointerException* грешки – статистически Android приложенията, написани на Kotlin, крашват ~20% по-рядко именно заради това​.

Kotlin се използва вече от над 60% от професионалните Android разработчици, като подобрява удовлетвореността на програмистите и безопасността на кода​.

Също така Kotlin се интегрира напълно със съществуващия Java код и библиотеки, което улесни използването му в Android Studio без нужда от изцяло нов стек.

Използването на **Android Studio** като среда бе естествен избор, тъй като това е официалният IDE за Android, базиран на IntelliJ IDEA, и предоставя удобни инструменти за разработка, тестване и отстраняване на грешки​.

Android Studio улеснява дизайна на потребителския интерфейс (UI) чрез Layout Editor и предлага емулатор за бързо тестване на приложението върху различни виртуални устройства. Комбинацията Kotlin + Android Studio представлява модерен и продуктивен технологичен стек, който гарантира по-качествен код и по-бърза разработка в сравнение с по-старите подходи (напр. Java + Eclipse).

**Сравнение с D&D Beyond**

За сравнение, **D&D Beyond** е официалната цифрова платформа и помощник за D&D 5-то издание, разработена от *Wizards of the Coast*. Тя предлага уеб сайт и мобилно приложение, които съдържат електронни версии на всички официални книги и правила, генератор на герои, дигитален лист на персонажа, списъци с магии и чудовища, инструмент за създаване на срещи и др.​

D&D Beyond позволява също добавяне на потребителско (homebrew) съдържание към системата​. С други думи, това е изключително богат и всестранен инструментариум, интегриран с официалните данни за играта.

**Предимството** на подхода на D&D Beyond е, че *предоставя “всичко на едно място”* – играчът може да създаде герой с няколко клика, да избира опции директно от официалните източници и да получава автоматично изчисляване на параметри, умения, магии и пр. Синхронизацията в облака позволява достъп до персонажа от различни устройства и споделяне с Dungeon Master (DM) и други играчи.

**Недостатъците** обаче произлизат от модела на разпространение: макар базовият генератор и някои съдържания да са безплатни, повечето разширени опции и пълните правила изискват закупуване на дигитални книги или абонамент. Много потребители се оплакват от този *paywall* модел – например, дори да притежавате физическите книги, в D&D Beyond трябва да платите повторно за дигиталната им версия, иначе нямате достъп до голяма част от съдържанието​.

В интернет общността често се изтъква, че всичко, което D&D Beyond прави, може да се постигне и с други безплатни програми или приложения, без допълнително заплащане​.

Освен това някои функционалности на официалното приложение са ограничени – потребители посочват случаи, в които определени особености на класове или предмети не се отчитат автоматично, или че дадени действия (например нулиране на заряди на предмети след почивка) трябва да се правят ръчно​.

Мобилното приложение на D&D Beyond също получава смесени отзиви – интерфейсът му не е особено интуитивен за всички, а навигацията между разделите на героя може да забави геймплея​.

В обобщение, D&D Beyond е мощен инструмент с богати данни, но изисква интернет достъп и често – допълнителни покупки, което го прави по-малко достъпен за някои играчи.

В контраст, **„Bojo's DnD App“** следва *по-опростен и самостоятелен подход*. Приложението е безплатно и работи локално на телефона, без да налага покупка на съдържание – потребителят сам въвежда нужните данни за героя си. Това означава, че няма ограничение от paywall за използване на определени раси, класове или предмети – всичко може да бъде добавено ръчно от играча според нуждите. Естествено, подходът има и **недостатъци**: приложението не предоставя автоматично всички правила и опции (както D&D Beyond), а разчита на потребителя да познава и въвежда характеристиките си. Също така „Bojo's DnD App“ е фокусирано само върху основните нужди (персонаж и инвентар) и не обхваща цялата екосистема на D&D (напр. няма вграден списък с всички чудовища или магии по подразбиране). Въпреки това, за един самостоятелен играч или малка група, желаещи да имат дигитален лист на героя без допълнителни такси, подобно приложение е привлекателна алтернатива.

**Основни изисквания към приложението**

При планирането на проекта **„Bojo's DnD App“** бяха дефинирани няколко основни изисквания (функционални и нефункционални):

* Приложението да поддържа **регистрация и вход на потребители**. Всеки потребител има свой профил (акаунт) с автентикация чрез потребителско име и парола. Това гарантира, че данните за героя са лични и защитени.
* Всеки потребител да може да **създаде и съхрани персонаж (герой)** със следните основни характеристики: име, раса, клас, ниво, базови атрибути (Сила, Ловкост, Издръжливост, Интелект, Мъдрост, Харизма), точки живот (HP) и др. Приложението трябва да осигурява екран, наподобяващ листа на героя, където тези параметри могат да се преглеждат и редактират.
* Приложението да предоставя **„магазин“ за предмети**, откъдето потребителят да може да добавя оборудване и инвентар към своя герой. Това включва списък от стандартни D&D предмети (оръжия, брони, пособия, отвари и пр.) с описание и цена, които могат да бъдат „закупени“ (добавени) към инвентара на героя.
* **Навигацията** в приложението трябва да бъде интуитивна – след успешен вход потребителят лесно да достига до основните секции (персонаж и магазин). Сменянето на екраните (напр. от персонаж към магазин и обратно) да става бързо и ясно, без загуба на данни.
* Приложението трябва да работи **офлайн** и да съхранява локално данните (на устройството) – т.е. избраната технология за база данни следва да е локална (напр. SQLite), за да не се изисква постоянна интернет връзка.
* Да се обърне внимание на **стабилността и удобството**: интерфейсът да следва добрите практики за дизайн (Material Design за Android), текстовете да са четими, а действията – с подходяща обратна връзка (напр. съобщения при грешка в логин, потвърждение при успешно добавяне на предмет и т.н.). Приложението трябва да бъде тествано за сценарии с неправилни входни данни (напр. невалиден вход, изключване на приложението внезапно и др.) и да се справя с тях без сривове.
* Кодът на приложението трябва да бъде четим и поддържим, с ясна структура. Използването на Kotlin следва да допринесе за по-малко бъгове и по-лесна поддръжка, а при работа с база данни да се предпочитат абстракции (като Room библиотеката), за да се намали писането на суров SQL на ръка.

**Реализация на приложението**

Приложението включва няколко основни екрана, всеки от които покрива отделна функционалност:

* **Екран за вход (Login)** – това е началният екран на приложението. Той съдържа форма за въвеждане на потребителско име и парола. След валидиране на данните (проверка спрямо записаните в базата данни потребители) потребителят преминава към вътрешността на приложението. Екранът за вход показва и бутон/линк към регистрация за нови потребители. За подобряване на удобството, са имплементирани основни проверки: например, ако задължителните полета са оставени празни, се визуализира съобщение за грешка и не се продължава. При грешно потребителско име или парола, приложението уведомява, че логинът е неуспешен. Успешният вход зарежда данните на съответния потребител (например неговия герой) и насочва към следващия екран.
* **Екран за регистрация** – достъпен чрез бутона на логин екрана, този екран позволява създаване на нов акаунт. Формата за регистрация обикновено съдържа полета за потребителско име, парола и евентуално имейл.

В нашата реализация е достатъчно потребителят да избере уникално потребителско име и парола. При натискане на "Регистрация", приложението проверява дали потребителското име вече съществува в базата данни. Ако е заето, се показва грешка и потребителят трябва да избере друго.

Ако данните са валидни, в базата се създава нов запис за потребителя. След успешна регистрация потребителят може директно да бъде пренасочен към екрана за вход (съобщавайки му се, че регистрацията е успешна), за да влезе с новосъздадените си данни.

* **Екран на персонажа** – това е основният екран, който показва информацията за текущия D&D герой на потребителя. Той представлява дигитален еквивалент на **character sheet** и съдържа текстови полета/етикети за името на героя, неговата раса, клас, ниво, както и числови полета за всеки от шестте основни атрибута (STR, DEX, CON, INT, WIS, CHA – съответно Сила, Ловкост, Издръжливост, Интелект, Мъдрост, Харизма).

Също така се показват текущите точки живот (HP) и други показатели като клас броня (AC), скорост (Speed) и пр., ако са релевантни. Екранът на персонажа обединява и **инвентара** на героя – обикновено като списък от предмети, които героят притежава.

В най-простия случай, това може да е скролируем списък от имената на предметите или текстово поле, в което предметите са изброени. В нашето приложение, всеки добавен през магазина предмет се изписва в секция "**Инвентар**" на екрана на героя.  
  
 Освен статичните данни, екранът на персонажа предлага и някои **интерактивни елементи**.

Например, може да има бутон "Редактирай", чрез който потребителят да коригира определени полета (напр. ако героят вдигне ниво и някой атрибут се променя, потребителят може да актуализира стойността). Друг важен бутон е "Към магазина" – той позволява на потребителя да навигира към екрана на магазина (описан по-долу), за да добави нови предмети.

Ако са нужни допълнителни действия (напр. хвърляне на зарове или управление на магии), те биха могли да се достъпят чрез допълнителни бутони или меню, но в текущата версия тези функции не са имплементирани, за да се фокусираме върху основното.

* **Екран "Магазин"** – този екран служи за преглед и избор на предмети, които потребителят може да добави към героя си. Представлява списък от налични предмети за закупуване.

Всеки елемент в списъка показва името на предмета, кратко описание и евентуално цена или категория (например оръжие, броня, зелье и т.н.).

Реализацията може да използва предварително дефиниран списък от базови D&D предмети, който е или записаn в базата данни, или е харкоднат в ресурсите на приложението (в зависимост от имплементацията – за прототип може списъкът да е статично зададен).

Потребителят може да избере даден предмет и да го "купи" – при тази операция, предметът се добавя към инвентара на героя в базата данни.

В UI това може да е реализирано чрез бутон "Добави" до всеки предмет. След добавяне, приложението може автоматично да се върне към екрана на персонажа или да покаже уведомление, че предметът е успешно добавен.

Екранът "**Магазин**" помага да се симулира аспект на играта, при който героите придобиват ново оборудване между приключенията.

Тъй като тук не се използва истинска валута, може да се опрости логиката – например всеки предмет просто се добавя без проверка за злато. (При желание, може да се добави поле за злато на героя и да се изисква то да е достатъчно, преди добавяне на предмет – но в началната версия това не е включено).

* **Други екрани (по желание)** – освен горните, в приложението могат да присъстват и помощни екрани или диалози. Например, ако потребителят желае да излезе от профила си, може да има опция "Изход", която да го връща към екрана за вход (и евентуално изчиства запазените в паметта данни за текущия потребител). Подобна функционалност може да се реализира с бутон в екрана на героя или в меню.

В нашия случай, основните сцени са изброените по-горе, като навигацията между тях покрива нуждите на потребителя: регистрация -> вход -> (герой <-> магазин).

### Примерен UI код (activity\_character.xml)

За илюстрация на изгледа на екрана на персонажа, по-долу е даден откъс от XML layout файла activity\_character.xml.

<LinearLayout

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="16dp">

<!-- Hero Name -->

<TextView

android:id="@+id/tvLabelName"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Name:"

android:textSize="18sp"

android:textStyle="bold"/>

<EditText

android:id="@+id/etName"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Character Name"

android:textSize="18sp"/>

Горният код описва вертикално скролируемо поле, съдържащо текстови полета за име, раса, клас, атрибути и инвентар, както и бутон за отваряне на магазина.

Например, TextView с ID tvLabelName и текст "Name:" е етикет за полето за име, а непосредстващият се EditText etName позволява въвеждането/редакцията на името на героя.

**Заключение**

В процеса на разработка на **„Bojo's DnD App“** бяха реализирани основните функционалности, заложени в проекта: система за потребители (регистрация/вход), създаване и управление на герой с неговите атрибути, както и модул "магазин" за добавяне на предмети.

Приложението използва съвременни технологии (Kotlin, Android Studio), което спомогна за по-бързото и надеждно изграждане на функциите. Kotlin допринесе за по-стегнат код и по-малко грешки при изпълнение, благодарение на безопасността при типовете и проверките на време на компилация. В сравнение с алтернативни решения (като използване на готови платформи или по-стария Java), собственото приложение позволява на разработчика пълен контрол върху функционалността и данните.

Основните **предимства** на полученото приложение са:

* **Опростеност и насоченост:** Интерфейсът е фокусиран само върху необходимото за играча – няма излишно сложни менюта или разсейващи елементи. Това го прави лесен за използване дори от хора, които не са техничари.
* **Автономност:** Приложението работи без нужда от интернет връзка или външни услуги. Всички данни са на устройството на потребителя, което означава, че няма риск от загуба на достъп поради сървърни проблеми или изискване за онлайн профил.
* **Персонализация:** Потребителят може да въвежда свободно стойности за своя герой, което означава, че няма ограничения в съдържанието. Например, дори homebrew (домашно измислени) раси или предмети могат да се добавят, тъй като приложението не ги ограничава до предварително зададени списъци (за разлика от някои официални инструменти).
* **Бързина:** Локалното съхранение и простата логика правят действието бързо – няма забавяне от мрежови заявки. Навигацията между екраните е гладка, а самото приложение има малък размер и стартира бързо.
* **Разширяемост:** Кодът, написан на Kotlin, е модулен и може да бъде разширяван. Ако в бъдеще се реши да се добавят нови функции (например тракване на магии, хвърляне на зарове, или дори синхронизация в облак), това може да стане върху основата, изградена до момента.

**Литература**

**D&D Beyond – Wikipedia (EN):** *Official digital toolset and game companion for Dungeons & Dragons 5th edition*. Описание на платформата D&D Beyond и нейните функционалности​  
[en.wikipedia.org](https://en.wikipedia.org/wiki/D%26D_Beyond#:~:text=D%26D%20Beyond%20,and%20add%20custom%20homebrew%20content).

**Reddit - r/DnD Discussion:** *"Anyone else think the D&D Beyond mobile app kinda sucks?"* – Обсъждане от D&D общността относно недостатъците на D&D Beyond приложението, включително мнения за платеното съдържание и ограничените функционалности​  
[reddit.com](https://www.reddit.com/r/DnD/comments/1cb074p/anyone_else_think_the_dnd_beyond_mobile_app_kinda/#:~:text=Is%20D%26D%20Beyond%20even%20worth,am%20often%20met%20with%20paywalls).

**Android Developers – Kotlin and Android:** Официална документация на Google за предимствата на Kotlin при Android разработка – висока продуктивност, по-малко грешки, използван от 60%+ от разработчиците​  
[developer.android.com](https://developer.android.com/kotlin#:~:text=Write%20better%20Android%20apps%20faster,developer%20satisfaction%2C%20and%20code%20safety).

**Android Developers – Meet Android Studio:** Официално въведение към Android Studio, описващо го като официален IDE за Android с богати функционалности (Gradle, емулатор, и др.)​  
[developer.android.com](https://developer.android.com/studio/intro#:~:text=Android%20Studio%20is%20the%20official,building%20Android%20apps%2C%20such%20as).

**Android Developers – Save data using SQLite:** Ръководство за съхранение на данни в Android с SQLite и препоръка за използване на Room като абстракция над базата данни​  
[developer.android.com](https://developer.android.com/training/data-storage/sqlite#:~:text=%2A%20There%20is%20no%20compile,SQL%20queries%20and%20data%20objects).

**Netguru Blog – Pros and Cons of Kotlin:** Статия, описваща плюсовете на Kotlin спрямо Java, напр. по-ясен и стегнат код и откриване на бъгове при компилация​  
[netguru.com](https://www.netguru.com/blog/kotlin-pros-and-cons#:~:text=Kotlin%20code%20is%20clearer%20and,can%20fix%20errors%20before%20runtime).

**Увод**

В увода се съдържа:

* кратко описание на софтуерния проект(какво представлява мобилното приложение);
* използваните технологии, средства и др. и мотивите за избора им;
* проучване, ако е правено, на приложения от същия тип с техните характеристики;
* изисквания към приложението.

**1. ……..**

В другите точки по преценка на изготвящия колко да бъдат и как да се именуват се описва реализацията на приложението. Включват се:

* Текстово описание на реализацията – всичко най-важно, което читателят трябва да знае, за да добие ясна представа, както за самото приложение, така и за това как е разработено, вкл. интерфейс, функционалност, алгоритми, БД;
* Скрийншоти с кратко описание;
* Извадки от код (XML, Java) с пояснения, **не се иска да включите целия код**;
* Друго - скици, диаграми, таблици и др.

**Заключение**

С няколко параграфа обобщете направеното от Вас - дайте информация за тестване, проблеми, предимства на Вашето приложение.

**Литература**

1.

2.

3.

**! Внимание:** цитирайте коректно всички източници, които сте използвали в курсовата работа **!**

**Общи:** формат - PDF или WORD**,** обем – около 10 стр.,моля следвайте шаблона и не използвайте твърде голям шрифт и големи изображения, не се препоръчва оцветяване на кода, кодът да е текст, а не картинка от IDE, обърнете внимание и на правописа.

Курсовата работа може да се предаде както на хартия, така и в електронна форма.