**ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – ВАРНА**

ФАКУЛТЕТ ПО ИЗЧИСЛИТЕЛНА ТЕХНИКА И АВТОМАТИЗАЦИЯ

Катедра**“Компютърни Науки и Технологии”**

Преддипломен проект

Тема:

Проектиране и разработване на..............

**Ръководител: ас. инж. Галина Найденова**

**Изготвил:** Николай Каменов Геров

**Специалност: Софтуерни и интернет технологии**

**Факултетен номер: 61662232**

ТУ- Варна

2020

ЗАДАНИЕТО

Тази страница остава празна!

Тук е мястото да се постави заданието, което ще ви дам след завършване на преддипломният проект.

**СЪДЪРЖАНИЕ**

[1. Увод 4](#_Toc37288742)

[2. Обзор и анализ на проблема 6](#_Toc37288743)

[3. Избрани технологии 7](#_Toc37288744)

[3.1. Android OS 7](#_Toc37288745)

[3.1.1. История 7](#_Toc37288746)

[3.1.2. Архитектура 7](#_Toc37288747)

[3.1.3. Интерфейс 8](#_Toc37288748)

[3.1.4. Защо Android OS? 8](#_Toc37288749)

[3.2. Java 9](#_Toc37288750)

[3.3. SQLite 10](#_Toc37288751)

[3.4. XML 11](#_Toc37288752)

[3.5. Android Studio 11](#_Toc37288753)

[4. Използвана литература 13](#_Toc37288754)

[4.1. Интернет източници: 13](#_Toc37288755)

[4.2. Изпoлзвaни съкращения: 13](#_Toc37288756)

# 1. Увод

След навлизането си, мобилната техногия несъмненно е незаменима част от ежедневието. Самото име подсказва, че става въпрос за преносимост, позволяваща гъвкавост и удобство. Тази технология предоставя възможност да изпълняваме задачи бързо и лесно посредством смартфон, таблет, лаптоп или друго мобилно устройство. Популярността на мобилните устройства може лесно да се обясни с факта, че те са по-лесно преносими, удобни и осигоряват свързаност почти във всяка една ситуация. Ръстът в тази област на IT технологиите отдавна е на лице, затова и много компании се стараят ефективно да използвт този факт.

Според последни проучвания потребителите използват своите мобилни устрийства за повиквания, писане на текстови съобщения, сърфиране в интернет, използване на различни приложения, игри или музика. Разнообразието на мобилните приложения в наши дни е изключително голямо. Развлекателни , информативни, образователни или насочени към хора със специални потребности могат да бъдат целите на дадено приложение, изборът е огромен.

Последни проучвания показват, че продажбите на смартфони в света са три пъти по-високи от тези на компютри. Поради това, всички фирми, вече трябва да се съобразяват с нещо, което беше изключително далечно само преди десет години: **присъствието на смарт мобилни устройства**. В много браншове това вече не е само една от многото опции, а изискване за успех. Повечето мениджъри вече са наясно, че адаптивният уеб сайт е основен начин за създаване на едно качествено мобилно преживяване на потребителите, но едно мобилно приложение може да даде повече.

Едно приложение може да доставя информация директно на смартфоните и

таблетите на потребителите, дори и да не се използва в конкретния момент. То може да се адаптира към различни скорости на преноса на данни. Също така, може да предложи по-вдъхновяващо визуално преживяване. Основната ценност на мобилното приложение е в това да отвори вратата към повече на брой потребители**.** Същевременно, трябва да се отбележи и високата конкуренция - около 90% от всички сваляния се разпределят между само 10% от приложенията на пазара. Поради това, всеки бизнес се нуждае от добре направено [мобилно приложение](https://www.whiz.bg/mobile-app-development) и конкретен план за неговата дистрибуция и реклама.

Множество компании изолзват мобилните приложения, за да популяризират марката си или даден техен продукт. Големият брой инсталации на приложението ви осигуряват широк кръг от лоялни клиенти. Глобалната свързаност на света, в който живеем предоставя възможността за предварително проучване и правилно таргетиране на целевата група. От изключително значение е разбирането на нуждите на аудиторията към която продукта е предназначен.

Изработката на мобилно приложение може да е сериозна инвестиция, но и да се изплати в дългосрочен план с положителен ръст в обема на продажбите. За се случи това, продуктът трябва да бъде успешен. Има различни ефективни формули за успех, базирани на съвкупност от много фактори.

Най-успешните приложения помагат на потребителите да разрешат конкретен проблем. Някои от най-силно представящите се апликации могат да правят само едно нещо, но го правят перфектно. Визията на приложението е много важна. Тя трябва да грабва вниманието на потребителите. Функционалностите трябва да правят приложението максимално полезно.

Страхотните приложения имат механизъм за споделяне в социалните мрежи и по имейл. Това практически осигурява безплатен и ефективен канал за реклама. Когато едно приложение е популярно в социалния кръг на един потребител, е много по-вероятно той също да го свали и използва.

Докато само преди няколко години платформата iOS бе водеща, днес в глобален мащаб доминира Android с дял надхвърлящ 50% на повечето национални пазари. Най-ефективната стратегия е да се обхванат и двете платформи, за да се да достигне до възможно най-голям брой потребители.

Невъзможно е да се преувеличи важността на анализа на потребителското поведение на мобилното приложение. Успешните приложения имат специално интегрирани инструменти за да следят кое съдържание събира най-много кликове и кои функции се използват най много. Това позволява навременна оптимизация както на съдържанието, така и на функционалностите за да се постигне максимална потребителска удовлетвореност.

Имайки предвид високата конкуренция на пазара, дори и приложенията с много ниска цена трудно достигат до потребителите тъй като има много, които са напълно безплатни. Освен това, за една фирма, която цели да използва своето приложение за маркетингови цели или за да предлага услуги на потребителите, не би било изгодно да поставя каквато и да е било бариера за достъпа до него.

Дори и най-доброто мобилно приложение на света не би било успешно, ако не се рекламира по правилния начин. Един добър план обхваща както атрактивно представяне с оптимизирано съдържание в Google Play и App Store, така и банер реклами, постове в социалните мрежи и други основни инструменти на дигиталния маркетинг.

.

# 2. Обзор и анализ на проблема

В днешния забързан свят всеки е зает и бърза за някъде. Всекиму, ежедневно има парични разходи под една или друга форма. Храна и напитки, транспортни разходи, развлечение, са само няколко от многото сфери, в които харчим средства всеки ден. При цялото напрежение съпътстващо ежедневито ни, може да се окаже трудно да следим колко средства точно сме изхарчили днес, а камо ли преди няколко дена или седмица. Това може и да се окаже предпоставка за големи разходи.

Тук възниква въпросът как може да лесно и удобно да оказваме контрол над средствата си. Воденето на отчет на хартия е един начин, но това означава, че трябва да носим и средство за писане у себе си. За максимално удобсто, решението на проблема трябва да е лесно преносим и ненатрапчив.

Тук идва и отговора – мобилните технологии. Голяма част от активното население вече притежава смартфон, те са неразделна част от ежедневието ни. Именно чрез тях проблема със следенето на финансите може да бъде решен. Смартфоните са един от най-използваните предмети, с които боравим в ежедневито си. Приложение позволяващо на потребителя си да прави преглед на разходите и приходите си е основната цел на разработвания продукт. Той трябва да бъде лесен за употреба от широка аудитория, от хора с технически познания, го такива без. Необходимата функционалност трябва да придружена от практичност и лекота при работа с продукта.

С гъвкавостта и бързите темпове, с които се развиват мобилните технологии, това ще бъде лесно постигнато.

# 3. Избрани технологии

## 3.1. Android OS

Android е [операционна система](https://bg.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) на [Google Inc.](https://bg.wikipedia.org/wiki/Google_Inc.) за [мобилни устройства](https://bg.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B1%D0%B8%D0%BB%D0%BD%D0%BE_%D1%83%D1%81%D1%82%D1%80%D0%BE%D0%B9%D1%81%D1%82%D0%B2%D0%BE), базирана на [ядрото на Linux](https://bg.wikipedia.org/wiki/Linux_(%D1%8F%D0%B4%D1%80%D0%BE)). Тя е създадена, поддържана и развивана първоначално от Android Inc., която е купена от Google Inc. през 2005.

Пускането ѝ на пазара на 5 ноември 2007 г. е съпроводено с основаването на [Open Handset Alliance](https://bg.wikipedia.org/wiki/Open_Handset_Alliance) – консорциум от хардуерни, софтуерни и телекомуникационни компании за развиването на отворени стандарти при мобилните устройства. Google пуска фронтенд кода под свободен лиценз.

За развитието на Android се грижат голям брой софтуерни разработчици, които създават така наречените „apps“ (*Applications*) – малки [приложения](https://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD_%D1%81%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80), които разширяват функционалността на системата. Приложенията могат да бъдат сваляни от различни сайтове в интернет или от големи онлайн магазини, като [Android Market](https://bg.wikipedia.org/wiki/Android_Market) (впоследствие преименуван на Google Play) – магазинът на [Google](https://bg.wikipedia.org/wiki/Google). По данни към юни 2018 г. за Android има над 3 300 000 приложения. Приложенията се пишат предимно на [Java](https://bg.wikipedia.org/wiki/Java), [Python](https://bg.wikipedia.org/wiki/Python) или [Ruby](https://bg.wikipedia.org/wiki/Ruby).

За последното тримесечие на 2010 г. Android вече е определена като най-продаваната платформа за смартфони в света.[[3]](https://bg.wikipedia.org/wiki/Android#cite_note-3) Популярността на системата непрекъснато расте и през второто тримесечие на 2014 г. 86% от продадените смартфони са с инсталирана операционната система Android. Към началото на 2017 г. Android работи на по-голямата част от смартфоните в почти всички страни в света, с изключение на САЩ, Канада, Австралия, Япония.

### 3.1.1. История

Компанията Android Inc. е основана в Пало Алто, Калифорния, през октомври 2003 г. от Анди Рубин (съосновател на компанията Danger), Рич Майнър, (съосновател на Wildfire Communications, Inc.), Ник Сиърс (бивш вицепрезидент в T-Mobile) и Крис Уайт (начело на дизайна и разработката на интерфейса на WebTV ), за да развиват, по думите на Рубин, „по-умни мобилни устройства, които са по-наясно с местоположението и предпочитанията от притежателя си“.

### 3.1.2. Архитектура

Основата на Android е [ядрото на Linux](https://bg.wikipedia.org/wiki/Linux_(%D1%8F%D0%B4%D1%80%D0%BE)) (версия 2.6). То е отговорно за управлението на паметта и [процесите](https://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)), както и за [мрежовите връзки](https://bg.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8E%D1%82%D1%8A%D1%80%D0%BD%D0%B0_%D0%BC%D1%80%D0%B5%D0%B6%D0%B0). Тук са разположени и [драйвърите](https://bg.wikipedia.org/wiki/%D0%94%D1%80%D0%B0%D0%B9%D0%B2%D1%8A%D1%80).

Директно над ядрото се намира т.нар. Runtime Environment. Тя съдържа най-важните библиотеки по време на изпълнение и най-важната функционалност на езика [Java](https://bg.wikipedia.org/wiki/Java). Тук се съдържа и виртуалната машина Dalvik Virtual Machine (DVM). Тя се различава от класическите виртуални машини на Java (Java Virtual Machine, JVM) по това, че е оптимизирана за мобилни уреди с малко памет. Оптимизацията позволява и едновременното изпълнение на няколко виртуални машини на същия уред.

### 3.1.3. Интерфейс

[Интерфейсът](https://bg.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5)) е базиран на директна манипулация – директно се управлява чрез докосване на [екрана](https://bg.wikipedia.org/wiki/%D0%95%D0%BA%D1%80%D0%B0%D0%BD_%D0%B7%D0%B0_%D0%B4%D0%BE%D0%BA%D0%BE%D1%81%D0%B2%D0%B0%D0%BD%D0%B5_(Touchscreen)). Самите докосвания отговарят на действия от реалния живот, като плъзгане и почукване, чрез тях се манипулират обектите на екрана и виртуалната клавиатура. Интерфейсът е проектиран да отговаря моментално на зададените от потребителя задачи. Наличният хардуер, като например акселерометър, жироскоп и сензор за близост, се използва от някои приложения в отговор на допълнителни действия на потребителя. Например променяне на [ориентацията](https://bg.wikipedia.org/w/index.php?title=%D0%9E%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F_%D0%BD%D0%B0_%D0%95%D0%BA%D1%80%D0%B0%D0%BD%D0%B0&action=edit&redlink=1) на екрана от портретен в пейзажен режим в зависимост от това как е ориентирано [устройството](https://bg.wikipedia.org/wiki/%D0%A3%D1%81%D1%82%D1%80%D0%BE%D0%B9%D1%81%D1%82%D0%B2%D0%BE).

При включване устройствата зареждат начален екран, по подобие на [работния плот](https://bg.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B1%D0%BE%D1%82%D0%B5%D0%BD_%D0%BF%D0%BB%D0%BE%D1%82_(%D0%BA%D0%BE%D0%BC%D0%BF%D1%8E%D1%82%D1%80%D0%B8)) при персоналните компютри. Началният екран е съставен от икони и джаджи. Иконите отварят свързаните с тях приложения, а [джаджите](https://bg.wikipedia.org/w/index.php?title=%D0%94%D0%B6%D0%B0%D0%B4%D0%B6%D0%B8&action=edit&redlink=1) са по скоро за показване на [информация](https://bg.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D1%8F) в реално време, например прогноза за времето. Началният екран може да бъде съставен от няколко страници, между които потребителят може да преминава чрез плъзгане. Повечето производители персонализират начина, по който изглеждат техните разновидности на Android, за да се разграничат от конкурентите си.

В горната част на екрана се намира лентата за състоянието, показваща информация относно устройството и неговата свързаност, например наличието на обхват или мобилни данни. Лентата за състоянието може да се плъзне надолу, като така се показва лентата с известията, където приложенията показват важна информация или известяват за обновяване, например получаване на [SMS](https://bg.wikipedia.org/wiki/SMS) или [имейл](https://bg.wikipedia.org/wiki/%D0%95%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0_%D0%BF%D0%BE%D1%89%D0%B0). Известията се показват, докато не ги прочетете, като почукате два пъти върху тях или ги отхвърлите, като ги плъзнете настрани извън екрана.

### 3.1.4. Защо Android OS?

Едно от безпорните предимства на операционната система Android за Вашият телефон или таблет е огромният набор от приложения (програми) за нея, голяма част от които безплатни или на символична цена.

Пълната функционалност на ОС Android изисква регистрация, но тя е безплатна и е свързана единствено с Google регистрация.

Част от другите предимства на операционна система Android са:

* Всички браузъри в Андроид могат да свалят файлове, след които да отваряте през файл менаджер
* За пълна функционалност се изисква регистрация, но опира до бързо създаване на Google акаунт, с който после ползвате всички услуги на Google: Gmail, YouTube, Google +, Android Market
* Android е лесен и доста user friendly операционна система. Постоянно се обновява и винаги раполагате с последната и версия.
* Ползва се в много модели таблети и смартфони- можете да изберете производител.

## 3.2. Java

Java е [обектно-ориентиран](https://bg.wikipedia.org/wiki/%D0%9E%D0%B1%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%B0%D0%BD%D0%BE_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5) [език за програмиране](https://bg.wikipedia.org/wiki/%D0%95%D0%B7%D0%B8%D0%BA_%D0%B7%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5), разработен от [Sun Microsystems](https://bg.wikipedia.org/wiki/%D0%A1%D1%8A%D0%BD_%D0%9C%D0%B0%D0%B9%D0%BA%D1%80%D0%BE%D1%81%D0%B8%D1%81%D1%82%D1%8A%D0%BC%D1%81) и пуснат в употреба през 1995 година, като част от Java платформата. Впоследствие се появяват множество други реализации включително от [GNU](https://bg.wikipedia.org/wiki/%D0%93%D0%9D%D0%A3), [Microsoft](https://bg.wikipedia.org/wiki/%D0%9C%D0%B0%D0%B9%D0%BA%D1%80%D0%BE%D1%81%D0%BE%D1%84%D1%82), [IBM](https://bg.wikipedia.org/wiki/IBM), [Oracle](https://bg.wikipedia.org/wiki/%D0%9E%D1%80%D0%B0%D0%BA%D1%8A%D0%BB_(%D0%BA%D0%BE%D1%80%D0%BF%D0%BE%D1%80%D0%B0%D1%86%D0%B8%D1%8F)) и други технологични доставчици. Изходният код, написан на Java, не се компилира до машинен код за определен [микропроцесор](https://bg.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BA%D1%80%D0%BE%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D0%BE%D1%80), а се компилира до междинен език - така нареченият [байткод](https://bg.wikipedia.org/w/index.php?title=%D0%91%D0%B0%D0%B9%D1%82%D0%BA%D0%BE%D0%B4&action=edit&redlink=1). Байт кодът не се пуска за директно изпълнение от микропроцесора, а се изпълнява от негов аналог – [виртуален процесор](https://bg.wikipedia.org/wiki/%D0%92%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D0%BD%D0%B0_%D0%BC%D0%B0%D1%88%D0%B8%D0%BD%D0%B0), наречен [Java Virtual Machine](https://bg.wikipedia.org/wiki/Java_Virtual_Machine) (JVM).

Подобна реализация има своите предимства и недостатъци. Сред главните предимства са:

* лесната преносимост между различните платформи ([софтуерни](https://bg.wikipedia.org/wiki/%D0%A1%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80) или [хардуерни](https://bg.wikipedia.org/wiki/%D0%A5%D0%B0%D1%80%D0%B4%D1%83%D0%B5%D1%80)) – веднъж написана и компилирана, една Java-програма може да бъде стартирана на компютри независимо от архитектурата или от операционната им система. За целта е необходимо само да се инсталира виртуална машина за съответната платформа;
* допълнителните действия, извършвани от виртуалната машина, като освобождаване на паметта от обекти (класове), които не се използват (Garbage collector), проверка за размерността на масивите;
* възможността за контрол на правата на потребителя на ниво виртуална машина
* първоначално заделяна на heap, част от паметта резервирана за джава
* висока степен на сигурност поради факта, че програмистите не работят директно с паметта и др.

Главен недостатък е необходимостта от допълнителни ресурси (под формата на процесорно време и памет) за изпълнението на самата виртуална машина.

Съществуват и компилатори, които превеждат байт кода до машинен код. Недостатъка при използването им е, че тогава програмата не може да се стартира на различни платформи.

Ранните версии на виртуалните машини страдат от слаба производителност поради необходимостта от компилиране на байт кода до машинен код. Този проблем е решен в значителна степен след въвеждането на [JIT](https://bg.wikipedia.org/w/index.php?title=JIT&action=edit&redlink=1) компилатора във виртуалната машина.

Java e [език за програмиране](https://bg.wikipedia.org/wiki/%D0%95%D0%B7%D0%B8%D0%BA_%D0%B7%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5) от високо ниво с общо предназначение. Синтаксисът му е подобен на [C](https://bg.wikipedia.org/wiki/C_(%D0%B5%D0%B7%D0%B8%D0%BA_%D0%B7%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5)) и [C++](https://bg.wikipedia.org/wiki/C%2B%2B), но не поддържа много от неговите възможности с цел опростяване на езика, улесняване на [програмирането](https://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5) и повишаване на сигурността. [Програмите](https://bg.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8E%D1%82%D1%8A%D1%80%D0%BD%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B0), написани на Java, представляват един или няколко файла с разширение .java. Тези фай­лове се компилират от [компилатора](https://bg.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D0%B0%D1%82%D0%BE%D1%80) на Java – javac до изпълним код и се записват във файлове със същото име, но различно разширение .class. Клас-файловете съдържат Java [байткод](https://bg.wikipedia.org/w/index.php?title=%D0%91%D0%B0%D0%B9%D1%82%D0%BA%D0%BE%D0%B4&action=edit&redlink=1) инструкции, изпълним от [вирту­алната машина](https://bg.wikipedia.org/wiki/%D0%92%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D0%BD%D0%B0_%D0%BC%D0%B0%D1%88%D0%B8%D0%BD%D0%B0).

В днешно време Java е един от най-популярните езици за програмиране. На него пишат милиони разработчици по цял свят. Най-големите световни софтуерни корпорации като IBM, Oracle, Google и SAP базират своите решения на Java платформата и използват Java като основен език за разработка на своите продукти. Сред „големите“, Java не се ползва единствено от Microsoft, тъй като те разработват и поддържат собствена платформа, подобна на Java платформата, наречена [.NET](https://bg.wikipedia.org/wiki/.NET) Framework и език, подобен на Java, наречен C#.

Езикът Java е първоначално разработен и поддържан от Sun Microsystems, но през 2006 г. Java платформата става проект с [отворен код](https://bg.wikipedia.org/wiki/%D0%9E%D1%82%D0%B2%D0%BE%D1%80%D0%B5%D0%BD_%D0%BA%D0%BE%D0%B4) и се поддържа и развива от световната Java общност съвместно със Sun. Благодарение на отворения код популярността на Java постоянно се увеличава и броят Java разработчици непрекъснато расте.

Езикът Java се разпространява заедно със специална среда, върху която се изпълнява, наречена Java Runtime Environment (JRE). Тя включва т. нар. Java [виртуална машина](https://bg.wikipedia.org/wiki/%D0%92%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D0%BD%D0%B0_%D0%BC%D0%B0%D1%88%D0%B8%D0%BD%D0%B0) (JVM) и пакет стандартни [библиотеки](https://bg.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5)), предоставящи базова функционалност. Благодарение на нея Java програмите са преносими и след като веднъж бъдат написани, след това почти без промени могат да работят на стотици хардуерни платформи и операционни системи.

## 3.3. SQLite

SQLite е [релационна база данни](https://bg.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D0%B8) с [отворен код](https://bg.wikipedia.org/wiki/%D0%9E%D1%82%D0%B2%D0%BE%D1%80%D0%B5%D0%BD_%D0%BA%D0%BE%D0%B4) поддържаща стандарта [SQL](https://bg.wikipedia.org/wiki/SQL). Реализирана е като библиотека към приложенията, а не като самостоятелно работеща програма. Използва се в продукти, разработвани от [Adobe](https://bg.wikipedia.org/wiki/Adobe), [Apple](https://bg.wikipedia.org/wiki/Apple), [Mozilla](https://bg.wikipedia.org/wiki/Mozilla), [Google](https://bg.wikipedia.org/wiki/Google) и др.

Концепцията на SQLite е цялата база да бъде съсредоточена в един-единствен файл. Това я прави база данни без сървърен процес, особено подходяща за използване в мобилни устройства, таблети и софтуер, където е невъзможно поддържането на сървърен процес.

SQLite е бърза и надеждна база данни, затова е избрана за база данни по подразбиране в OpenOffice Base, LibreOffice Base, както и в езиците за програмиране PHP, Ruby on Rails и други. CMS системата Drupal също работи с SQLite.

## 3.4. XML

XML (eXtensible Markup Language) е стандарт (метаезик), дефиниращ правила за създаване на специализирани маркиращи езици, както и синтаксисът, на който тези езици трябва да се подчиняват. Сам по себе си той е безполезен, защото указва само как да бъде структуриран един документ (чрез маркиране с етикети), но не и какво означават отделните маркери (етикети). Ето защо е по-правилно да се нарече метаезик отколкото език – обикновените езици имат [семантика](https://bg.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BC%D0%B0%D0%BD%D1%82%D0%B8%D0%BA%D0%B0), т.е. предават някаква информация, докато XML указва само граматиката (по-точно [синтаксиса](https://bg.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81)) на езиците, базирани на него.

Ползата от XML се състои в това, че синтаксисът (структурирането) на документите се измисля само веднъж, а специализираните маркиращи езици само дефинират [семантиката](https://bg.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BC%D0%B0%D0%BD%D1%82%D0%B8%D0%BA%D0%B0) т.е. набора от маркиращи етикети и тяхното значение. Другата изгода е, че XML прави възможно разработването на семантично агностични програмни средства които могат да обработват всички XML базирани езици, дори и такива които не са съществували по време на съставянето на софтуера. Тези програмни средства могат след това да се използват и в изработването на софтуер който е написан за конкретен XML език. Или накратко – разделянето на синтаксиса от семантиката е поредното въплъщение на принципа „Разделяй и владей“.

XML е създаден като олекотена версия на [SGML](https://bg.wikipedia.org/wiki/SGML), целящ по-лесното му внедряване и налагане сред разработчиците. Създаден е от Уеб консорциума ([World Wide Web Consortium](https://bg.wikipedia.org/wiki/World_Wide_Web_Consortium)) и съответно е добре пригоден за употреба в Интернет пространството.

## 3.5. Android Studio

Android Studio е официалната среда за разработка (IDE - [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment)) на приложения работещи на операционната система на Google – Android. Представен на 16 Май 2013г, на конференцията  Google I/O, Android Studio е безплатен софтуер под лиценза Apache License 2.0.

Базиран на софтуера JetBrains'IntelliJ IDEA, Android Studio е създаден специално за Android разработчиците. Той е достъпен за сваляне за Windows, Mac OS X и Linux, като замества  Eclipse Android Development Tools (ADT) като основен IDE за разработка на приложения за Android.

Някои от основните функционалности и опции на разположение на разработчиците са:

* създаване на приложения с помощта на Gradle. Gradle е приложение за автоматизиране на проектите, чиято цел е да сглоби(build) даден проект, използвайки концепциите на [Apache Ant](https://bg.wikipedia.org/w/index.php?title=Apache_Ant&action=edit&redlink=1) и [Apache Maven](https://bg.wikipedia.org/w/index.php?title=Apache_Maven&action=edit&redlink=1), като също така добавя и базиран на езика [Groovy](https://bg.wikipedia.org/w/index.php?title=Groovy&action=edit&redlink=1) – [DSL](https://bg.wikipedia.org/w/index.php?title=Domain-specific_language&action=edit&redlink=1) вместо традиционния и широко използван [XML](https://bg.wikipedia.org/wiki/XML) формат за деклариране на конфигурациите към проекта.
* Refactoring и quick fixes специфични за Android
* Вграден емулатор за тестване и дебъгване на създаваните приложения. Този емулатор позволява приложението да се инсталира на широка гама от устройста с различна разделителна способнос на екрана.
* Drag-and-drop система за създаване на потребителсия интерфейс.
* Поддържа се създаване на Android Wear апликации.
* Опция за вградено използване на различни системи за контрол на версията (GitHub, SVN и др.).
* Версиите след Android Studio 3.0 поддържат използването на Kotlin.

# 4. Използвана литература

## 4.1. Интернет източници:

1. Wikipedia, the free encyclopedia - https://en.wikipedia.org/wiki/Main\_Page
2. BGpoll.net - https://www.bgpoll.net
3. Whiz.bg - https://www.whiz.bg

## 4.2. Изпoлзвaни съкращения:

JRE - Java Runtime Environment

JVM - Java Virtual Machine

XML - Extensible Markup Language