

Задание за проект за курс по Андроид разработка към академия Net IT

Проектът трябва да включва познания придобити по време на Android и Java курса. Оценяването на проекта ще стане на база брой реализирани функции на апп-а, устойчивост на апп-а, спазване на Андроид правилата за дизайн, архитектура и качество на кода. Проекта се счита за направен на 100%, ако имате 500 точки. Възможно е да имате над 500 точки.

Тема на проекта - Quiz application

Създайте приложение, чрез което да могат да се играят викторини, с брой играчи между 2 и 4 човека. Всяка викторина има два кръга:

1. Въпроси със затворен отговор
2. Въпроси с ВЯРНО/НЕВЯРНО

Пример при игра с 4ма човека:

- За всеки въпрос от първи кръг, за всеки от 4те човека се появява бутон за отговор.
- Въпросът се появява на екрана, но без отговорите.
- Който пръв натисне своя бутон, има правото да отговаря на въпроса.
- След като е избран кой ще отговаря на въпроса, се появяват възможните отговори и таймер.
- Човекът трябва да отговори на въпроса преди да изтече таймера.
- Ако отговори грешно, следващият най-бърз (ако има такъв) отговаря на въпроса. Ако и той не успее, въпроса се счита за неотговорен и се преминава към следващият въпрос.
- При верен отговор, човекът отговорил получава точка, при грешен -1.
- Когато въпросите от кръга свършат, се появяват временните резултати. Само двамата най-добри продължават към кръг 2.
- Втори кръг започва с разпределението на 4 теми. Участникът с по-нисък резултат има правото да разпредели коя тема за кой участник да бъде.
- Всеки участник има време в което трябва да отговори на възможно повече въпроси от дадената тема. Въпросите изискват само ВЯРНО/НЕВЯРНО. Участникът има право и да пасува. При правилен отговор, участникът получава 2 точки, а на противникът му се махат -2 точки. При пасуване, участникът получава -1 точка.
- Когато участниците отговорят на въпросите и от четирите теми, се показва финалното класиране.

Използвайте https://opentdb.com/api_config.php, за да реализирате проекта.

Изисквания

Точките, които ще получите след изпълнение на всяко изискване, са описани след текста на изискването, в скоби.

Функционалности (700)

Общи

- Аппа се компилира и се пуска успешно. (1)
- Аппа покрива правия случай на употреба. (5)
- Аппа покрива поне по един от страничните случаи за всяка функционалност. (10)
- Аппа покрива всички странични случаи. (30)
- Аппа е двуезичен - на български и английски. (5)

Начален екран

- Този екран се отваря при стартиране на аппа (1)
- Екранът има туулбар с името на приложението отгоре (2)
- Екранът има Floating Action Button в долния си десен ъгъл, който води до друг екран (1)
- *Екранът съдържа списък с три вида елементи:*
 - Елемент заглавие на под-списък. Може да е Games или History (1)
 - Елемент активна игра. Съдържа карта, в която има иконката на играта, името на играта, категориите чрез списък от чипове и трудността на играта чрез звездички, които се пълнят (7)
 - Елемент изиграна игра. Съдържа иконката на играта, името и датата, когато играта е била изиграна. (5)
 - Списъкът съдържа и трите вида елементи. (10)
 - Кликане на елемент активна игра води до нов екран. (1)
 - Кликане на елемент изиграна игра води до нов екран. (1)
 - Списъкът зарежда информация динамично, взимайки я от базата данни на аппа. (20)

Екран за създаване на игра

- Този екран се отваря при натискане на плюс бутона от Начален екран (1)
- Екранът има туулбар със заглавие "Create Game" и иконка със стрелка назад (2)
- Натискане на стрелката на туулбара води до предходния екран (1)
- Екранът има поле за въвеждане на името на играта. Полето има подходяща подсказка. (2)
- Екранът има бутон с иконката на играта. При натискане, се появява диалогов прозорец със списък с емоджи иконки. Потребителят може да избере една. Избраната иконка е видима като "име" на бутона за избор на иконките. (10)
- *Екранът съдържа поле с категориите на игра:*
 - Полето има заглавие Categories и плюс бутон. (2)
 - При натискане на плюс бутона се появява диалогов прозорец със списък с все още неизбраните категории. Потребителят може да избере категория. (4)
 - В същият диалогов прозорец потребителят може да избере трудността на категорията. (2)
 - Категориите се зареждат динамично от REST API. (10)
 - В полето има списък с категории. Те имат име и трудност, изразена в запълване на три звездички. Имат и бутон X. (3)
 - При натискане на бутона X на категория, тя изчезва от списъка. (2)
 - Категориите в списъка се появяват след като са избрани от потребителя. (5)
- *Екранът съдържа настройки:*

- Екранът има настройка с име Round 1 Number of Questions и дефолтна стойност 10 въпроса. Натискайки върху настройката се появява поле за редактиране на стойността. Настройката указва от колко въпроса се състои първи кръг. (3)
- Екранът има настройка с име After Question Delay и дефолтна стойност 6 секунди. Натискайки върху настройката се появява поле за редактиране на стойността. Настройката указва след отговаряне на въпрос, след колко време да се появи следващият. (3)
- Екранът има настройка с име Round 1 First Try Time и дефолтна стойност 10 секунди. Натискайки върху настройката се появява поле за редактиране на стойността. Настройката указва след като играч е заявил желание да отговори на въпрос, колко време има за да го стори, щом са се появили възможните отговори. (3)
- Екранът има настройка с име Round 1 Second Try Time и дефолтна стойност 5 секунди. Натискайки върху настройката се появява поле за редактиране на стойността. Настройката указва след като първи играч се е провалил да отговори на въпроса, и е дошло времето на втория най-бърз играч да отговори на въпрос, колко време има за да го стори, щом са се появили възможните отговори. (3)
- Екранът има настройка с име Round 2 Time и дефолтна стойност 30 секунди. Натискайки върху настройката се появява поле за редактиране на стойността. Настройката указва каква е продължителността на рунд от втори кръг. (3)
- Екранът има бутон Create. Бутонът създава играта в локалната база данни с подадените ѝ характеристики. Автоматично се пресмята обща трудност за играта като средно аритметично от трудностите на отделните категории. Тя също се записва в базата. (20)
- Всички полета за въвеждане се валидират при създаване на игра. При грешно въведени стойности/празни полета, се появява съответната грешка за всяко такова поле. (10)

Екран с детайлите на игра

- Екранът се отваря при натискането на елемент от списъка с активни игри на Началния екран. (1)
- Екранът зарежда информацията на избрания елемент от Началния екран от локалната база. (5)
- Екранът има туулбар с иконка назад, заглавие с името на играта и иконка боклук. (3)
- При натискане иконката назад, се отваря предходният екран. (1)
- При натискане на иконката боклук от туулбара, играта се изтрива от локалната база данни и се отваря предходния екран. (3)
- Преди да се изтрие играта, се появява диалогов прозорец, който иска съгласието на потребителя за това действие. (3)
- Екранът зарежда информация от локалната база за играта - иконката на играта, трудността изразена със звездички. (2)
- Екранът зарежда информация от локалната база за играта - под под-заглавие Categories се показва списък с категориите в играта и тяхната трудност. (5)
- Поле за последно играна игра - показва кога последно е играна тази игра с дата. Натискайки това поле се отваря нов екран. (2)
- Поле с топ играч за тази игра - показва кой има най-висок резултат за тази игра. Натискайки това поле се отваря нов екран. (2)
- Полетата за последно играна игра и топ играч се зареждат от локалната база данни. (2)
- Екранът съдържа настройка за типа игра - локална или с интернет. Настройката е switch. (2)
- *Екранът съдържа бутон Play. При натискане на бутона:*
 - Ако играчът е избрал игра с интернет, се прави проверка за наличие на връзка. При липса - се изкарва грешка и играчът не продължава напред. В противен случай е пренасочен към следващия екран. (5)
 - Ако играчът е избрал локална игра, се прави проверка дали има записани в локалната база въпроси към тази игра. Ако има, играчът продължава към следващия екран, ако

няма - се прави връзка към REST API и се изтеглят и записват в локалната база. Изтеглят се толкова въпроси със затворен отговор, колкото е броя въпроси в първи кръг по пет. Изтеглят се толкова въпроси с ВЯРНО/НЕВЯРНО, за всяка категория по толкова въпроси, колкото секунди е продължителността на втори кръг за тази игра. При липса на интернет се появява грешка, уведомяваща потребителя, че трябва да има интернет първия път когато играе локална игра. (20)

Екран с последно играни игри

- Екранът има туулбар със заглавие и стрелка назад, която връща до предходния екран. (2)
- *Екранът се отваря след натискане на полето с последно играни игри на екрана с детайлите на конкретна игра.*
 - Заглавието на екрана е името на играта + History. Пример: Math Game History (1)
 - Екранът съдържа списък с история на играните игри за тази игра. (10)
 - Всеки елемент от списъка показва броя играчи играли играта и датата. (3)
- *Екранът се отваря след натискане на елемент от списъка на Начален екран с исторически игри.*
 - Заглавието на екрана е History (1)
 - Екранът съдържа списък с история на изиграните игри за всички игри. (15)
 - Всеки елемент от списъка показва името на играта, броя играчи играли играта и датата. (4)

Екран с класация на игра

- Екранът се отваря от екрана с детайлите на игра. (1)
- Екрана има туулбар. Заглавието на екрана е името на играта + Leaderboard (1)
- Екранът съдържа списък, който се зарежда от локалната база. (5)
- Всеки елемент от списъка има името на играч, броя играчи в съответната игра, и постигнатият резултат. (5)
- Списъкът е подреден спрямо резултата. (2)

Екран с избор на играчи

- Екранът се отваря при натискането на Play бутона на екрана с детайлите за игра. (1)
- Екранът има туулбар със заглавие Select Players и стрелка назад. (2)
- При натискане иконката назад, се отваря предходният екран. (1)
- Екранът има четири квадратни плюс бутона разположени на целия екран. (4)
- *При натискане на плюс бутон се появява диалогов прозорец.*
 - Диалоговият прозорец има заглавие Add Player (2)
 - В диалоговият прозорец може да се въведе името на играч. (2)
 - Диалоговият прозорец има дефолтна картинка за аватар на потребителя. (2)
 - Диалоговият прозорец има два бутона - за избор на снимка от галерията и от камерата. (2)
 - Потребителят може да избере снимка от галерията и тя се показва на мястото на дефолтния аватар в диалоговият прозорец. (10)
 - Потребителят може да избере снимка от камерата и тя се показва на мястото на дефолтния аватар в диалоговият прозорец. (6)
 - Потребителят може да натисне бутона Add на диалоговият прозорец и въведената информация ще се появи на мястото на съответния плюс бутон. (5)
 - Потребителят може да натисне бутона Cancel или извън диалоговият прозорец, за да го затвори. (2)
- При попълване на информация за потребител, тя се появява на мястото на плюс бутона - със снимката и името на потребителя. (3)
- Натискане на вече въведена информация за потребител отваря диалоговият прозорец с попълнената му информация за редакция. (5)
- *Екранът има Start бутон*

- При натискане, ако има въведени по-малко от два играчи, се появява грешка. (2)
- При натискане на бутона, се появява нов екран. (1)

Екран с рунд 1

- Екранът се отваря при натискане на Start бутона на екрана с избор на играчи. (1)
- Екранът се отваря в landscape. (2)
- Екранът се отваря на цял екран. (3)
- Екранът показва толкова на брой бутони, колкото са избраните играчи от предходния екран. Бутоните са по тглите на екрана и са с текст - имената на играчите и различни цветове. (10)
- Екранът зарежда въпросите за рунда. Ако е избрана игра с интернет, те се зареждат от REST API спрямо категориите и трудността на играта и настройката за брой въпроси. Ако е избрана локална игра - се зареждат от локалната база на произволен принцип от възможните за играта. (20)
- *Екранът стартира рунда.*
 - Екранът показва текст "Prepare to answer in X", където X е секундите от настройката After Question Delay. X намалява с 1 всяка секунда. (10)
 - Когато мине определения брой секунди, екранът показва въпрос. (2)
 - Въпросът стои на екрана 10 секунди. Ако никой не отговори, се започва от начало със следващия въпрос. (5)
 - Ако някой натисне бутона си, се изчаква още 2 секунди за да се види дали още някой няма да натисне бутона си. (20)
 - Екранът показва възможните отговори. Първият натиснал бутона има X секунди да отговори. X се определя от настройките на играта. (10)
 - На екрана има таймер отброяващ оставащото му време. (15)
 - При маркиране на отговор, екранът показва дали той е верен или грешен като го оцветява. Ако е верен, се започва от начало със следващия въпрос. (10)
 - Ако е грешен, втория по бързина участник, ако има такъв, може да отговаря. (10)
 - Таймера се рестартира с време X, където X се определя от настройките на играта. (5)
 - В горният край на екрана има текст, който подсказва кой играч отговаря в момента. (2)
 - Верен отговор носи 1 точка, грешен отговор взима 1 точка. (8)
- След като са минали всички въпроси от рунда, се преминава автоматично на следващият екран. (2)

Екран с резултатите от рунд 1

- Екранът се отваря след приключване на рунд 1. (1)
- Екранът има заглавие "Results after Round 1". (1)
- Екранът показва играчите, подредени от най-много до най-малко точки. (5)
- За всеки играч пише името му и броя точки. (5)
- Топ 2мата играчи са оцветени, останалите са бледи. (2)
- Има бутон за продължаване напред. (1)

Екран за избор на категория

- Екранът се появява след натискане на бутона за продължаване напред на екрана с резултатите от рунд 1. (1)
- Екранът показва 4 категории спрямо категориите на играта. (20)
- Ако играта има само 3 категории, една се показва два пъти. Ако играта има само 2 категории, те се показват по два пъти. Ако играта има само 1 категория, този екран се пропуска напълно. (10)
- Играчът с по-малък резултат от миналият рунд има правото да разпредели кой на каква категория ще играе. Това става ясно от текст, който изписва името на този играч отгоре. (2)

- Разпределянето се случва като играчът натиска върху категориите. Те се оцветяват в различни цветове спрямо цветовете на играчите. Едно натискане я оцветява в един цвят, второ - в друг. (20)
- Играчът трябва да разпредели 4те категории между себе си и опонента по равно. Ако не го направи, се появява грешка. (5)
- Когато играчът е готов, той натиска бутон продължи. (1)

Екран с рунд 2

- Екранът се появява след екрана за избор на категория. (1)
- Зареждат се въпросите за текущата категория - или локално или с REST API. (30)
- Играчите отговарят на въпроси в 2те им избрани категории. (10)
- Всеки играч има X секунди (спрямо настройките на играта) за дадена категория. (10)
- На екрана има таймер отброяващ секундите. (15)
- На екрана отгоре е изписано името на играча, който отговаря в момента. (2)
- На екрана е изписан текущият въпрос. Въпросите са от тип вярно/невярно. (5)
- На екрана са изписани опциите TRUE/FALSE/PASS. (5)
- При избор на опция се оцветява в цвят в зависимост дали е вярна, и се отваря нов въпрос, до изтичане на времето. (20)
- След изтичане на времето се появява бутон Continue, който стартира следващата категория. (1)
- При правилен отговор, участникът получава 2 точки, а на противникът му се махат -2 точки. При грешен отговор, на участникът се махат 2 точки. При пасуване, участникът получава -1 точка. (30)

Екран с резултатите от играта

- Екранът се отваря след приключване на рунд 2. (1)
- Екранът има заглавие "Results". (1)
- Екранът показва играчите, подредени от най-много до най-малко точки. (5)
- За всеки играч пише името му и броя точки. (5)
- Има бутон за затваряне на играта. Той води до Начален екран. (5)

Качество на кода (100)

- Правилно и логично именувани променливи, методи и класове (30)
- Малки и компактни методи, без странични действия (25)
- Малки класове с единствена отговорност (25)
- Прилагане на архитектура и логично разделен код по пакети (20)

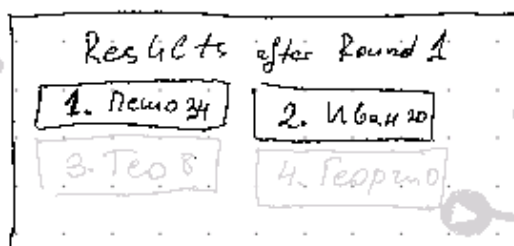
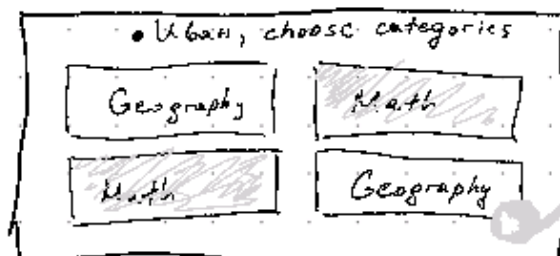
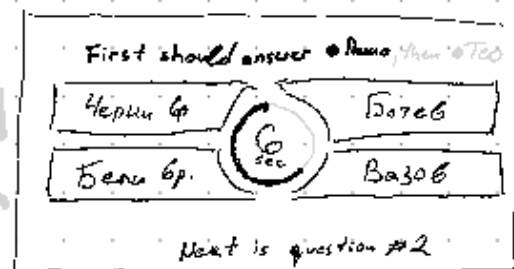
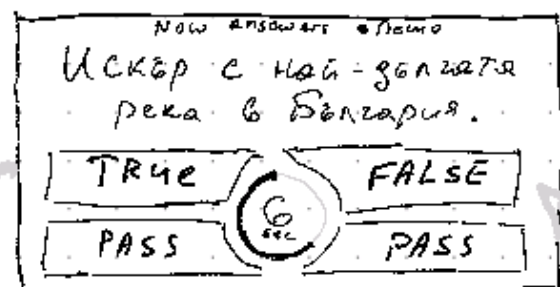
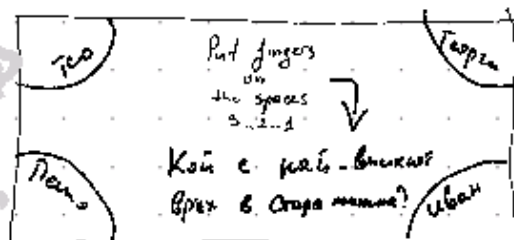
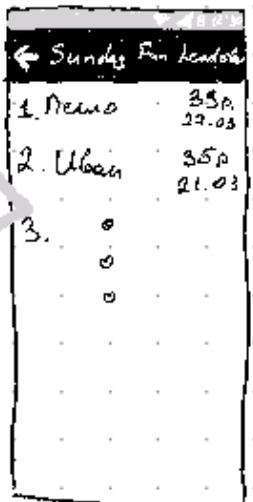
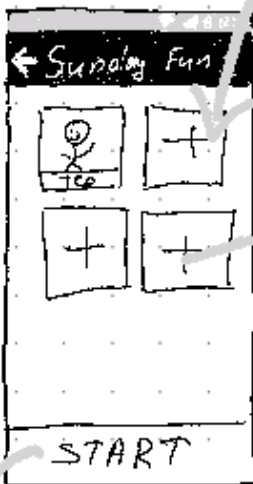
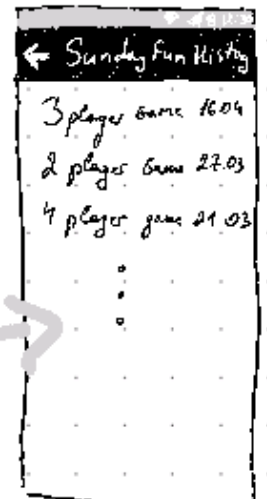
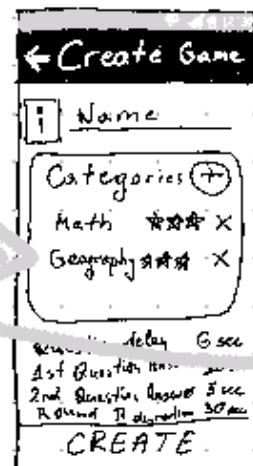
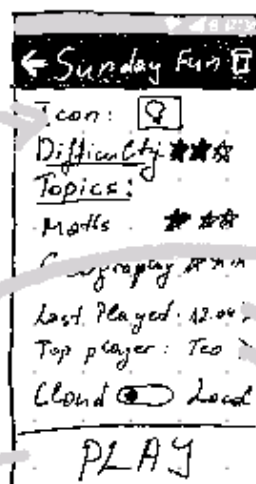
Описание на компонентите

Компонентите, които задължително трябва да присъстват в апа ви са изброени долу. Дори да сте покрили изискванията по брой точки, ако някой от тези компоненти липсва в апа ви, проекта ви не може да бъде приет.

- Activity
- Fragment
- RecyclerView
- ToolBar
- Navigation Component
- Връзка с интернет и парсване на данни от там
- Локална база данни

Външен вид

Приложението трябва да изглежда интуитивно и да спазва Material Design guidelines. Примерни екрани и връзка между екраните може да намерите тук:



API

Част от заявките, които ще ви трябват са:

- За 10 произволни въпроса с тема “География” от тип multiple choice:
<https://opentdb.com/api.php?amount=10&category=22&type=multiple>

Параметрите в заявките са:

- amount (задължителен) - колко въпроса
- category (опционален) - темата на въпросите (ако липсва ще има от всички теми)
- type (опционален) - какъв тип да са въпросите (вярно/грешно или с четири избора) (ако липсва ще има и от двата типа)

Качване

Проектът трябва да е качен в GitHub в публично репозитори. Проекта трябва да е създаден още в самото начало, за да може да се проследи активността при обновяване на кода.

Презентиране

Проектът трябва да има документация. Примерна документация би включвала:

- Описание на проекта, архитектурата и главните пакети и модули
- Описание на реализираните ф-сти
- Използвани външни библиотеки
- Използвани ресурси

Когато се представя проекта, трябва да го показвате на компютър, за да може да виждат всички. За тази цел трябва да имате инсталирано приложение, например Vysor (има и алтернативи).