**Цель**: разработать простое приложение - листалку, в котором юзер может листать страницы с контентом:

1. Сплеш скрин
2. Главное меню
3. Две дополнительные фичи для приложения, весь список будет в конце ТЗ, ваш проджект менеджер выберет 2 конкретные фичи из этого списка и вы должны будете их реализовать

**ВАЖНО**:

1. minSdk - API 21, targetSdk - самый актуальный на текущий момент
2. В build.gradle уровня приложения использовать

classpath "com.android.tools.build:gradle:7.0.2" (актуально для проектов в Android Studio)

1. Код должен быть написан САМОСТОЯТЕЛЬНО, а не взят из каких-то открытых источников

**Название проекта**: contentlist\_v1

**Название пакета**: com.template именно так. НЕЛЬЗЯ com.template.еще\_что\_то

**Нейминг**: стартовая активити должна называться MainActivity, не важно что в ней будет сплеш скрин / экран загрузки / главное меню или еще что-то с чего может стартовать игра, важно чтобы у активити с

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

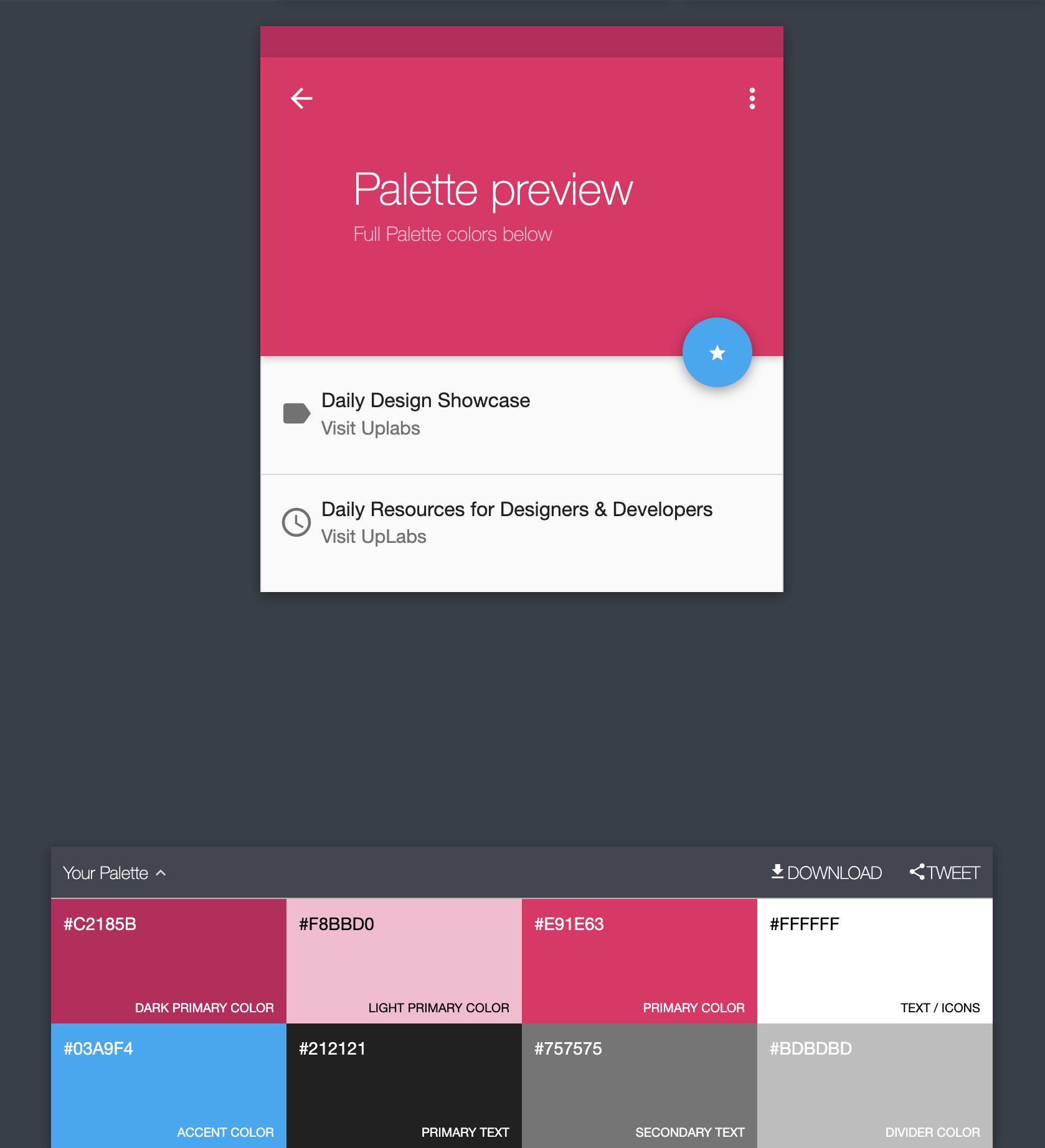
было название MainActivity (в данном приложении в MainActivity будет сплеш скрин)

**Графика:**

1. Цветовую схему вам выдаст проджект менеджер при помощи сайта <https://www.materialpalette.com>

Например, ПМ выдал вам ссылку на вот такую цв. схему:

<https://www.materialpalette.com/pink/light-blue>



Все оттенки подписаны, вам нужно лишь вставить все styles/colors

Accent Color - будет использоваться для всех кнопок

Для фона используется или Light Primary Color или Primary Color

Также можно использовать какие-то дополнительные цвета по вашему усмотрению, креатив приветствуется :)

1. Иконки, если будут нужны какие-то иконки, то можете брать любые (на ваш вкус) вот с этих ресурсов:

<https://materialdesignicons.com>

<https://www.flaticon.com>

Повторюсь, креатив приветствуется :)

Единственный момент, если вы уже использовали какие-то иконки в других наших с вами проектах, то новые иконки должны отличаться от предыдущих

1. Шрифты: либо стандартные, либо можете добавить свои

Единственный момент, если вы уже использовали какие-то шрифты в других наших с вами проектах, то новые шрифты должны отличаться от предыдущих

**Ориентация экрана:** либо портрет, либо альбом, какая именно вам скажет ПМ

Верстка: требований к UI / UX нет, ниже будет сказано лишь какие элементы должны быть на экране, а как они будут расположены / их размеры и прочее - на ваше усмотрение. **Главное, чтобы все работало, все было кликабельно и удобно юзеру**.

**Логика приложения:**

1. Сплеш скрин:

* по центру иконка и название приложения (иконка стандартная ic\_launcher, название app\_name)
* реализовать какую-то анимацию для иконки (например иконка слегка увеличивается и уменьшается и так постоянно, пока юзер на данном экране)
* длина сплеш скрина 1.5 секунды

1. Главное меню

* в главном меню будут 2 кнопки вперед и назад
* контейнер с контентом, контент будет отображаться в виде html / pdf / doc (docx) / txt / страниц какой именно формат вам скажет проджект менеджер
* нажимая на кнопку вперед - назад юзер может переключаться между этими страницами

**Контент:**

наполнением контента будет заниматься наш копирайтер, поэтому в разрабатываемом приложении нужно соблюсти следующие требования:

1. Весь контент должен храниться в одном месте, чтобы контентщик мог легко найти куда добавить готовые страницы со статьями / книгами и прочим.
2. Также при сдаче проекта указать путь до файла, где хранится контент, как его заменить и какой нейминг используется для страниц
3. Кол-во контента не должно регулироваться в коде, сколько страниц контентщик добавит, столько и должно отображаться в приложении

Для того, чтобы мы могли протестировать апк, добавьте 3 - 4 страницы с любым текстом в нужном формате.

**Дополнительный функционал:**

ниже представлен список из различных фич, вы должны реализовать **2 фичи** из этого списк, номера необходимого фунционала вам выдаст проджект менеджер

1. Кастомный тулбар, на котором посередине будет отображаться прогресс чтения (типа 2/10 и тд)
2. Кнопка About Us которая ведет на отдельный экран. На данном экране: Иконка и название приложения, версия приложения и его вес.
3. Кнопка About Us которая открывает на кастомный диалог. В данном диалоге: Иконка и название приложения, версия приложения и кнопка закрывающая диалог.
4. Для переключения можно использовать не только кнопки вперед-назад, но и свайп влево вправо по экрану
5. Кнопка “игры” (либо текст games, либо иконка) открывает игровой портал в Chrome Custom Tabs

* <https://static.gamezop.com/docs/guides/CCT-Implementation.html>

Сама ссылка на игровой портал: <https://www.gamezop.com/?id=xyz>

1. Тулбар с Navigation Drawer, в котором будет: иконка, название, версия и кнопки: поделиться и оценить приложение (делимся пакетом приложения, оценить - открываем страницу приложения в гугл плей)
2. Кнопка добавить в избранное и соответственно экран, где можно посмотреть сохраненные избранные страницы (например, в виде списка, кликаешь по элементу и открывается нужная страница)
3. Юзер нажимает на кнопку назад на телефоне (находясь в главном меню) и ему показывается Toast “Press the back button twice to exit” => чтобы юзер мог выйти из приложения нужно быстро нажать дважды кнопку назад
4. В конце всех страниц мы открываем отдельный экран (фрагмент / активити / диалог, не важно), где говорится, что мы все прочитали и есть кнопка “start over” - открывает первую страницу
5. Анимация переходов между экранами страницами с контентом (на ваш вкус, но должна быть как вперед, так и назад. Если вы уже реализовывали анимации перехода в других наших проектах, то нужно использовать другие)
6. Сохранение прогресса чтения при закрытии приложения. В следующий раз открываем на том месте, где остановились
7. Кнопка Rate the article. Нажимаем и появляется диалог, где предлагается оценить конкретную статью (страницу с контентом). Оценка по пятибалльной шкале. Оценку сохраняем. В следующий раз при нажатии на кнопку на этой странице показывается диалог с той оценкой, которую юзер дал в прошлый раз.