

ЛЕКЦИЯ I © 2021 Нет Ит

# Android: Material Design

Теодор Костадинов



SOFTWARE  
ACADEMY



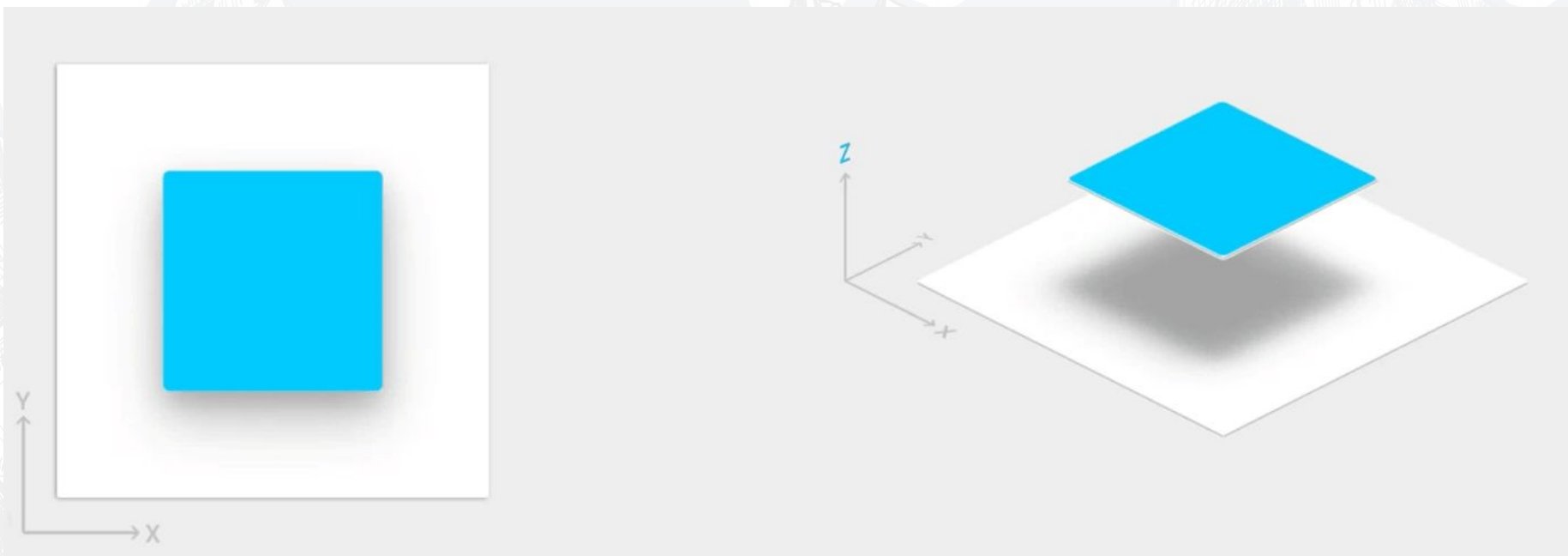
# Съдържание

---

1. Material Design
2. Constraint Layout

# Advanced User Interface

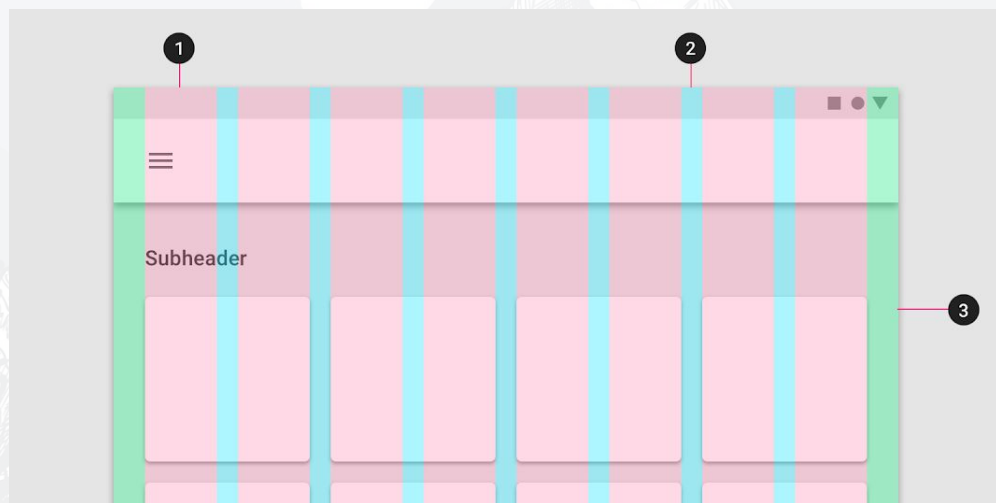
- Material Design е името на поредица от правила, създадени от Гугъл, които показват как да се създава дизайн за Андроид и други дигитални системи
- Първата версия е приложена в Android 5.0 Lollipop
- Идеята е, че копира взаимодействия между материали от реалния свят. (за това се казва “material”)



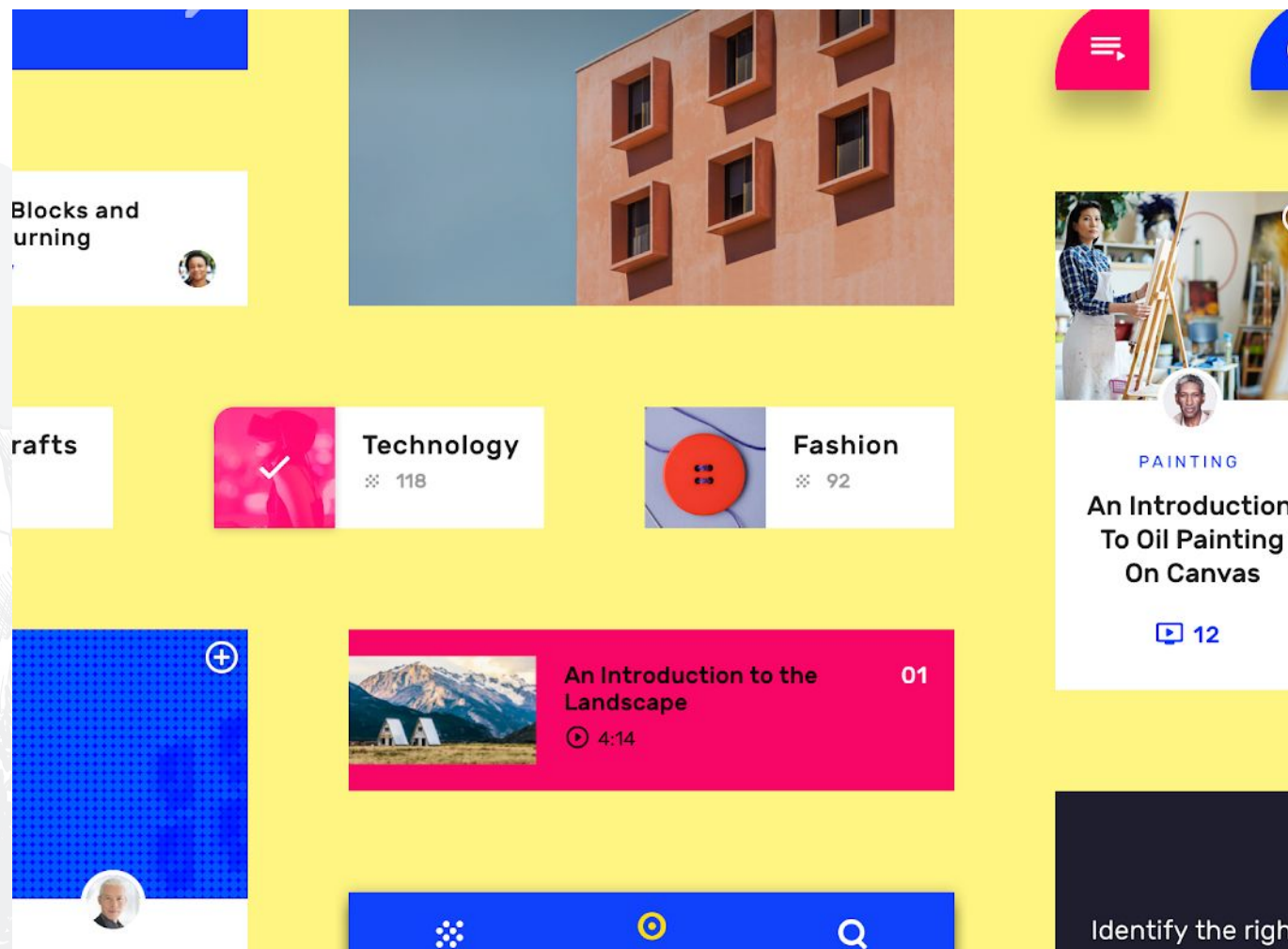


# Основни концепции

- Ярки и смели цветове
- Ограничаване до 2-3 основни цвята
- Ясни и изчистени движения
- Подредба по колони



Source: <https://material.io/design>



# Material Design в Андроид

- Стилизирането на апп се случва основно от styles.xml файла
- Там се дефинират както глобални стилове за всички компоненти (вюта), така и могат да се дефинират индивидуални стилове, които да бъдат приложени на определено вю
- Чрез styles.xml може да бъде сменен всеки един аспект от външния вид на едно вю
- Анимациите са много важни за доброто преживяване на потребителя

Задача: Имплементирайте Shared Element Transition:

<https://material.io/develop/android/theming/motion>

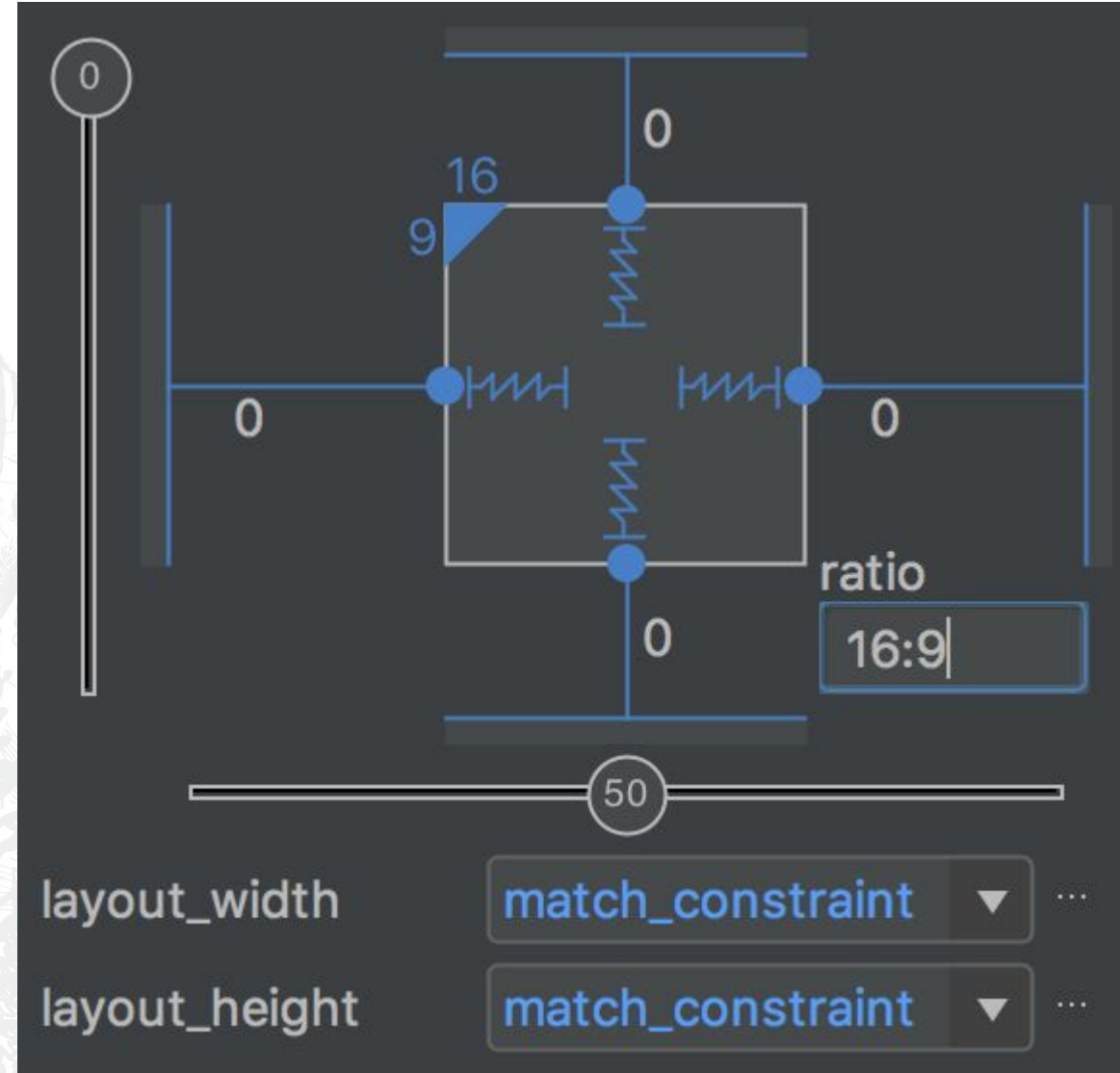


Почивка  
до 20:20



# Constraint Layout

- Подобрение над Relative Layout
- Помага йерархията на вютата да е по-плоска
- Може изцяло да се настройва от Дизайн таба





# Constraint Layout Usage

- Each view has to have at least one vertical and one horizontal constraint
- For view sizes we have: fixed size, wrap\_parent and match\_constraint
- The Layout Editor allows us:
  - To convert a relative layout into a constraint layout
  - To automatically set constraints depending where the view is
  - To easily align multiple views together
  - To set default margins



# build.gradle

Добавяме следния ред:

```
dependencies {  
    implementation "androidx.constraintlayout:constraintlayout:2.0.0"  
}
```

И се уверяваме, че:

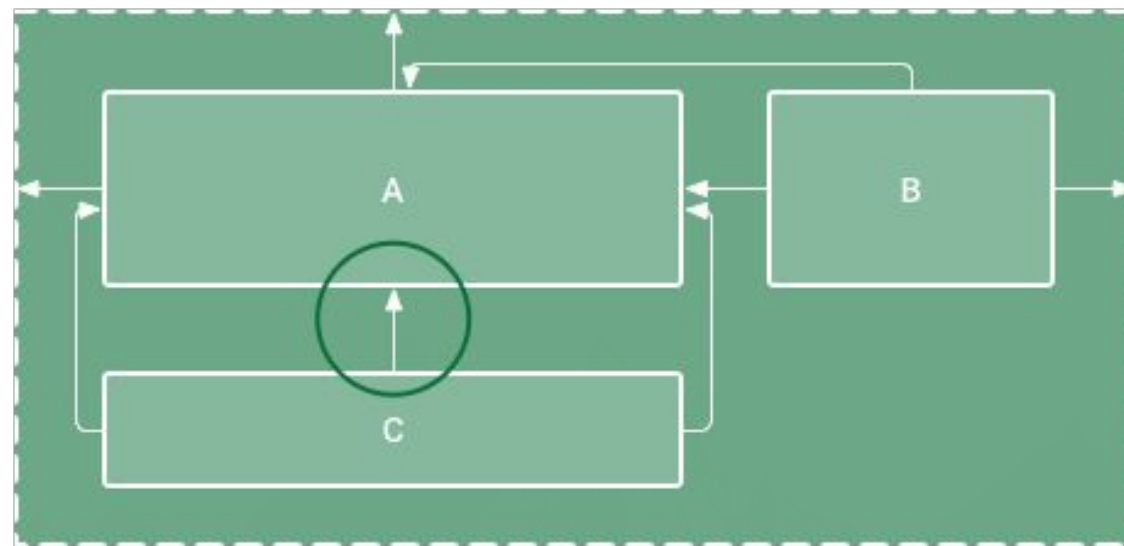
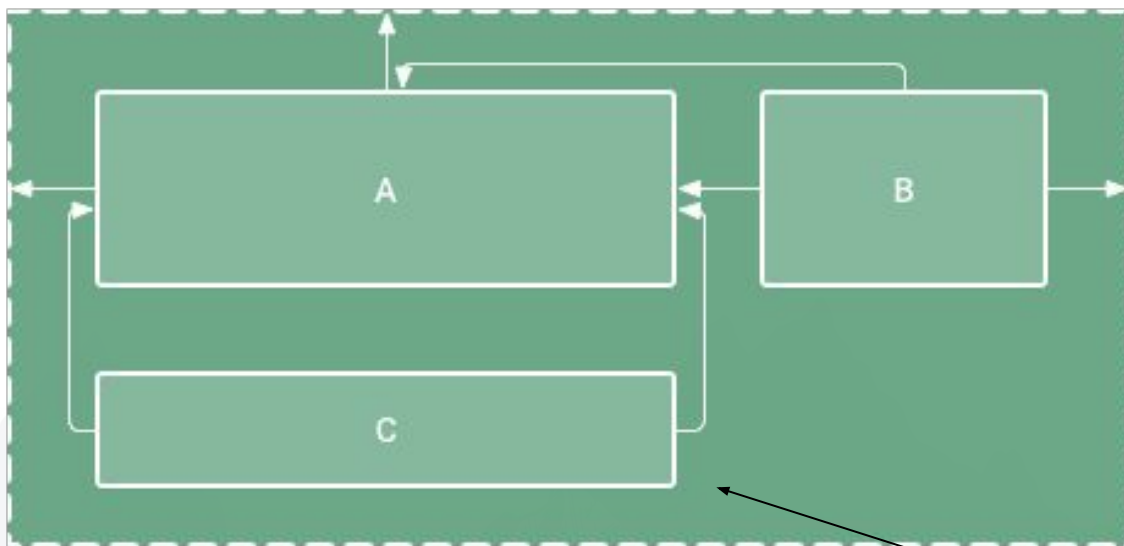
```
repositories {  
    google()  
}
```



app:layout\_constraint



- **За да може да работи, всяко `view` трябва да има поне едно хоризонтално и едно вертикално ограничение:**
  - `app:layout_constraintStart/constraintEnd`
  - `app:layout_constraintTop/constraintBottom`
- **Всяко едно органичение представлява връзка с друго `view`, `parent layout`-а или невидим `guideline`-и**



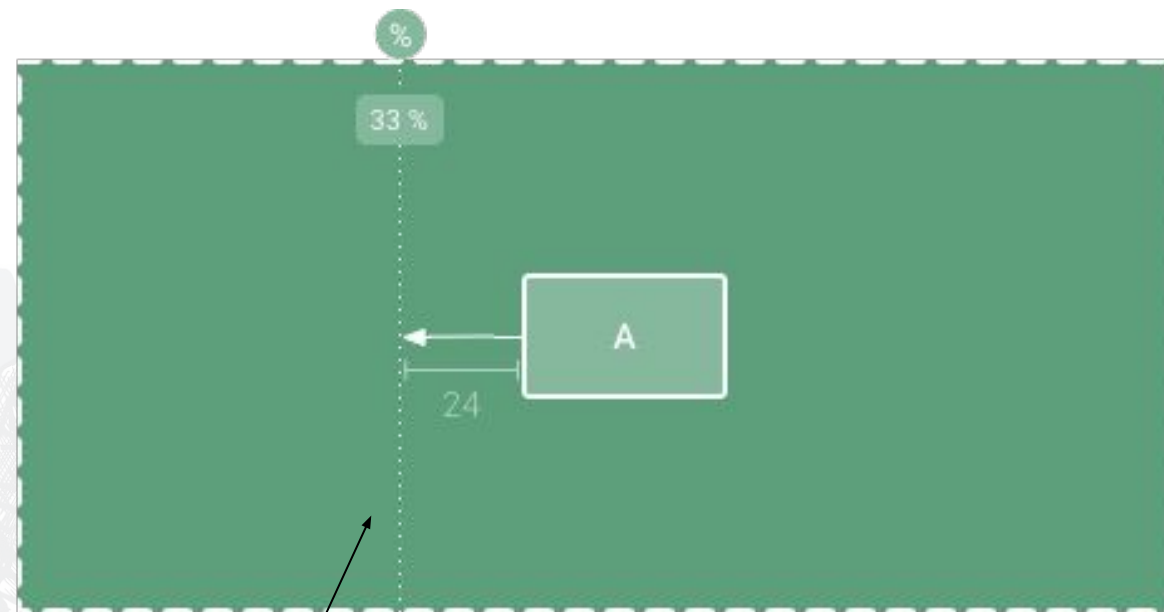
Липсва вертикално  
ограничение

Макар да изглежда привидно на място, при стартиране на приложението, ще бъде изчертано най-отгоре. Задължително трябва да имаме поне по едно ограничение хоризонтално и вертикално.



# Guidelines

Не винаги искаме да подравняваме нашите view-та едно спрямо друго или на база на границите на ViewGroup-ата. За тази цел можем да дефинираме своя собствена невидима по време на изпълнение граница, спрямо която нашите view-та да се организират.



Натискаме Guideline в toolbar-а и после **Add Vertical Guideline** или **Add Horizontal Guideline**.

ВЪПРОСИ?







**HOMEWORK**

© 2020 Нет Ит

# БЛАГОДАРЯ ЗА ВНИМАНИЕТО!



SOFTWARE  
ACADEMY

