

Розробка мобільних додатків

Лекція 1. Вступ

Викладач

Любченко Денис Валеріанович

Любченко Денис Валеріанович

- 📱 5 років досвіду в мобільній розробці
- 🌟 Мав 2 стартапи
- 😎 Керував найкращим стартап клубом України
- 👨‍💻 Розробляв і впроваджував новий напрямок дизайну Житомирської політехніки
- 🐶 Маю найкращу в світі собаку. Бігль Юпітер (він заставив мене це написати)

Контакти:

kkn_idv@ztu.edu.ua
@kkn_idv - Telegram



Agenda

1. Навіщо нам мобільна розробка?

2. Як ми будемо навчатися?

3. Два підходи до навчання

4. Історія мобільної розробки (Короткий екскурс)

5. Доступні технології для мобільної розробки

6. Чому ми вибираємо Ionic Framework?

7. Старт роботи з Ionic Framework

1. Навіщо нам мобільна розробка?



70%

70% інтернет-трафіку — з мобільних пристроїв

Ericsson Mobility Report: за 10 років мобільний трафік зріс у 300 разів

Available in [English](#) [Français](#) [Español](#) [日本語](#) [한국어](#) [Русский](#) українська

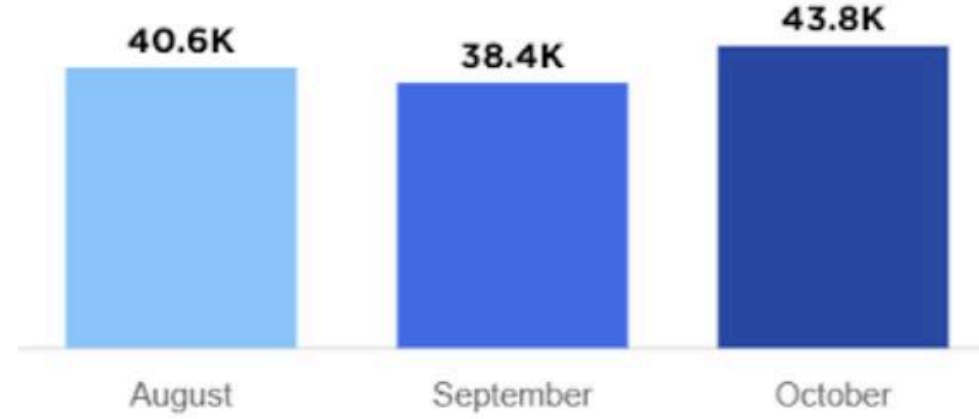
- Згідно з даними нового, 10-го за рахунком, звіту Ericsson Mobility Report, за період від 2011 року до теперішнього часу смартфони з'явилися ще у 5,5 млрд людей.
- Мобільний трафік даних лише в третьому кварталі 2021 року виявився більшим за трафік, згенерований у мобільних мережах за весь період до кінця 2016 року.
- До кінця 2027 року в мережах 5G буде 4,4 млрд підключень — майже половина всіх мобільних підключень у світі.

Джерело: <https://www.ericsson.com/uk/press-releases/2021/11/ericsson-mobility-report-mobile-data-traffic-increased-almost-300-fold-over-10-years>

App Store 

5998344 apps worldwide

New apps worldwide

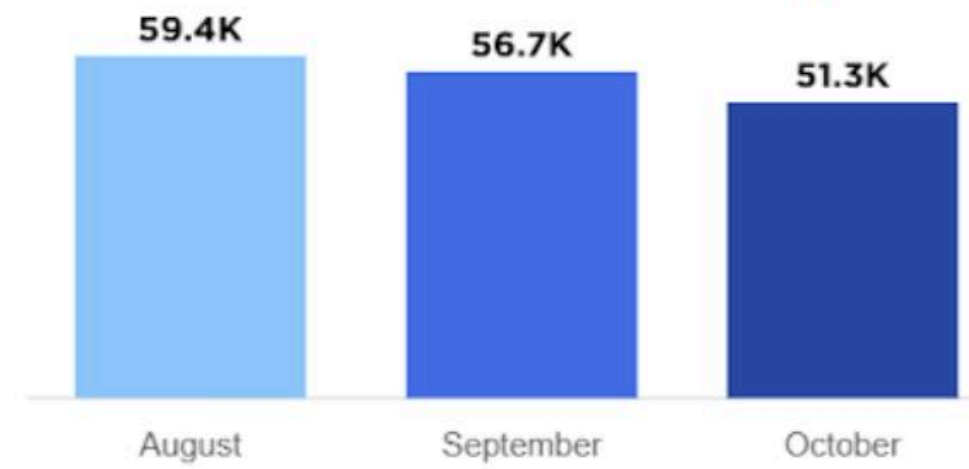


November 2.3K -27.1%
01.11 - 02.11 from last month period

Google Play 

14169172 apps worldwide

New apps worldwide



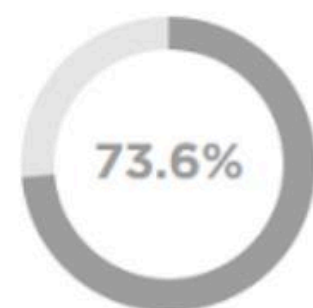
November 3.8K +22.3%
01.11 - 02.11 from last month period

Country data

 USA  



Apps



Free



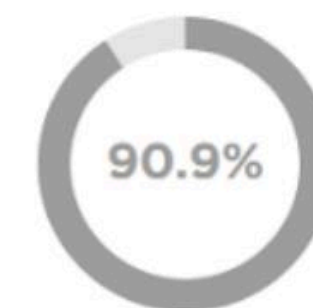
Games



Free



Apps



Free



Games



Free



2. Моє бачення на дисципліну

Моє бачення на цю дисципліну

Підкреслення групової динаміки

Викладач у ролі наставника

У нашому курсі ми детально розглянемо основні методи розробки мобільних додатків (Ionic, Flutter, Android та IOS) з основним акцентом на Ionic Framework.

Ви навчитеся створювати додатки, працювати з базами даних, розробляти дизайни та макети, інтегрувати нативні модулі, проводити тестування та відлагодження.

Мета - створити готовий продукт у кінці курсу.

Шляхи

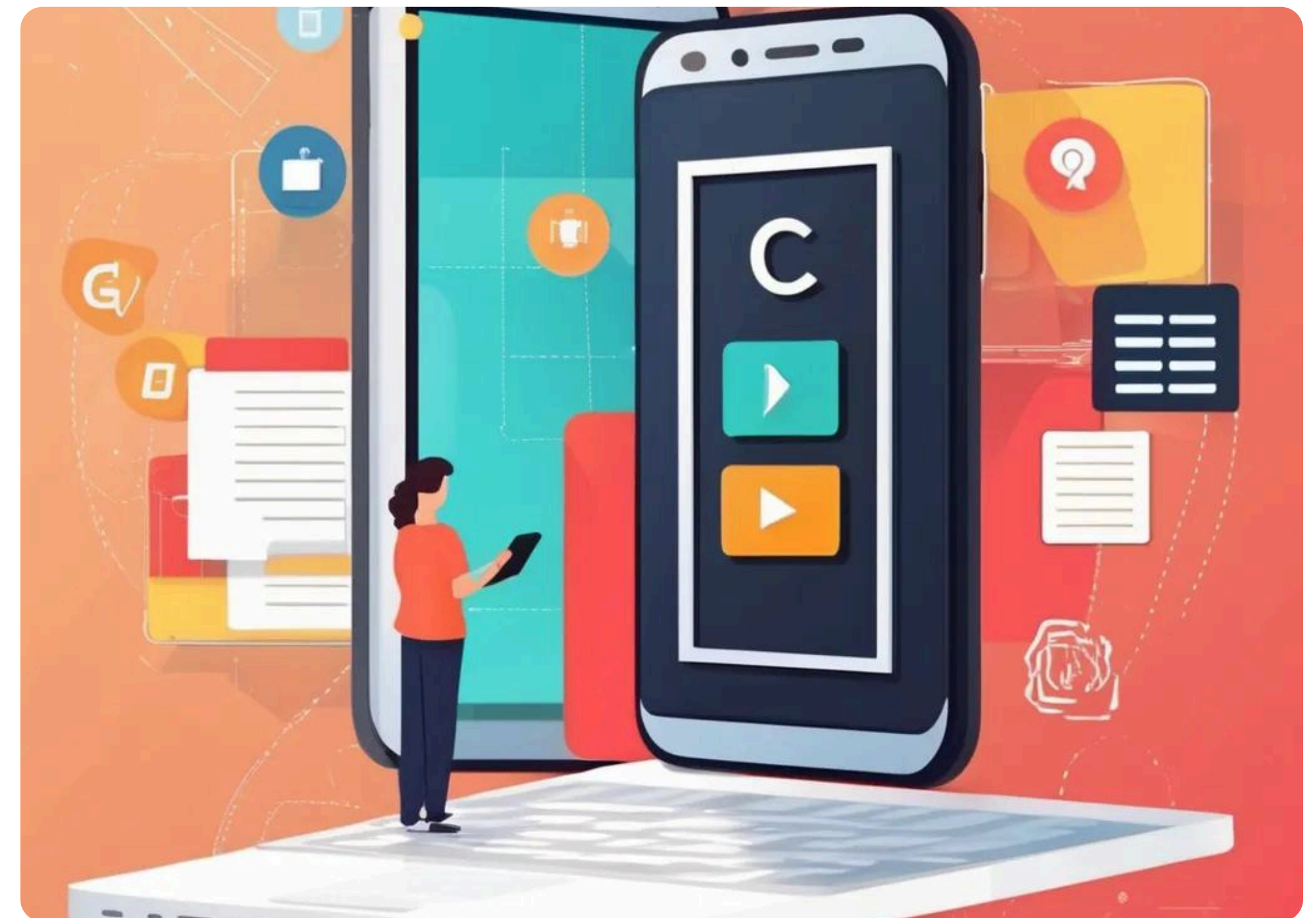
Шлях 1 (на 100)

Пройти курс зі мною, отримувати менторську допомогу, сапорт на кожному етапі до розробки продукту.



Шлях 2 (на 60)

Пройти онлайн курс (з сертифікатом), написати тести, зробити інд. завдання



Мобільна історія



1973

Перший мобільний телефон було створено компанією Motorola в 1973 році. Він називався DynaTAC 8000X і був досить громіздким та важким. 800 грамів. На ринок він вийшов аж в 1983 році за ціною 3995 доларів США. З нього можна лише дзвонити.



Початок 90их

Персональні цифрові асистенти (КПК), такі як Psion EPOC, набули популярності. Вони пропонували можливості роботи з електронними таблицями, базами даних, текстовими редакторами та календарями.



1993

IBM представила свій перший "смартфон" – IBM Simon. Цей сенсорний телефон, випущений у 1994 році, включав календар, годинник та калькулятор, які можна вважати першими "додатками".



1996

Palm створила операційну систему Garnet OS (Palm OS), що дозволило запускати на КПК сторонні додатки.



1998

Nokia, лідер ринку мобільних телефонів, представила першу всесвітньо відому мобільну гру – "Змійка". Це стало початком ери мобільних розваг.



2002

Смартфон BlackBerry став знаковим завдяки своїй функції бездротової електронної пошти, що було революційним для того часу.



2007

Вихід iPhone та операційної системи iOS ознаменував початок ери сучасних смартфонів із сенсорними екранами та магазинами додатків.



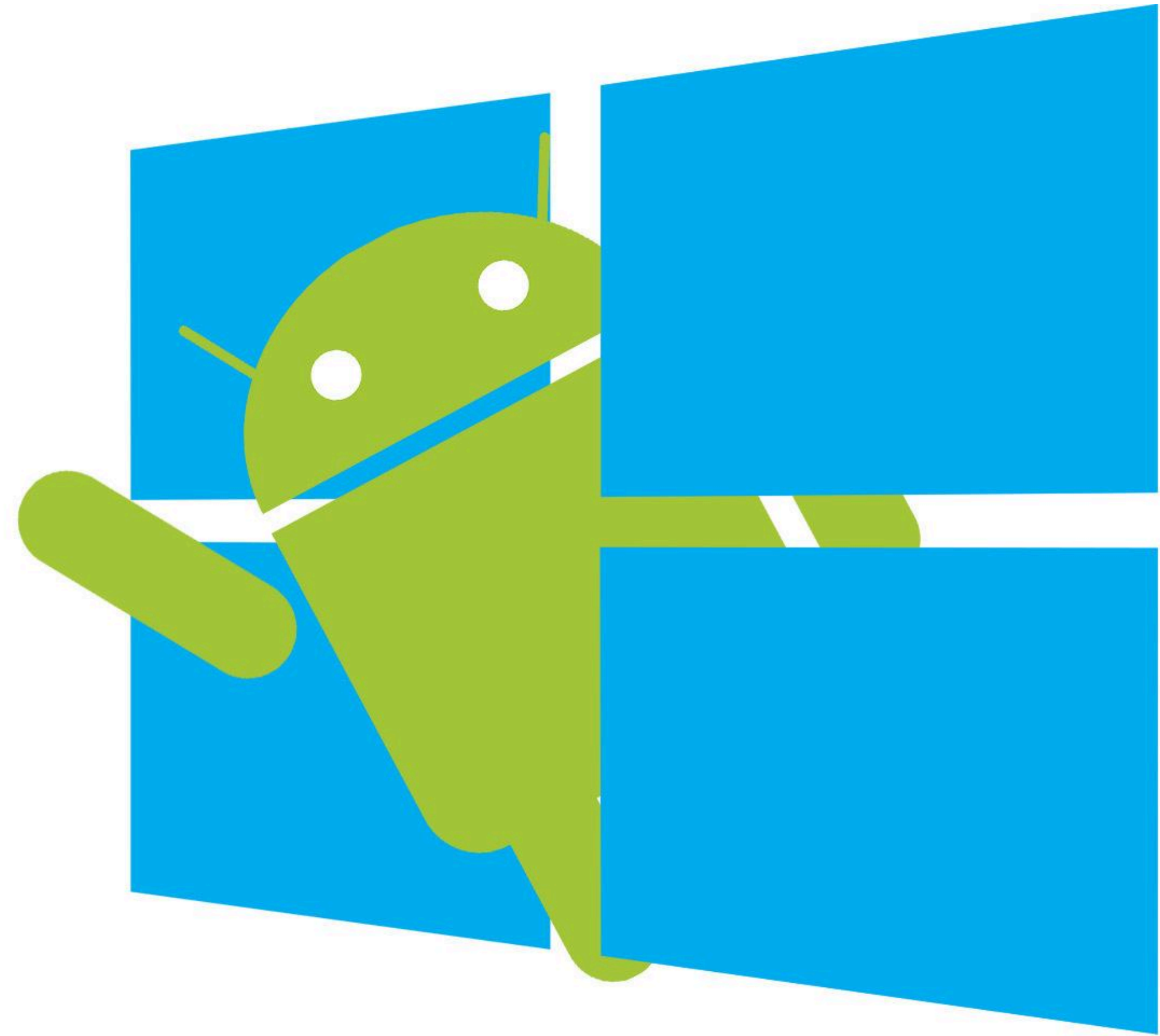
2008

Apple запустила App Store, де користувачі iPhone могли завантажувати мобільні додатки.



2008-2010

Google (Android OS) та Microsoft (Windows Phone) представили власні мобільні операційні системи, що сприяло подальшому розвитку ринку смартфонів.



2010-2011

Ринок мобільних додатків почав стрімко зростати.



2011

Кількість завантажень додатків з App Store перевищила 10 мільярдів, що вдесятеро більше, ніж у 2009 році.



2014

Кількість завантажень додатків у світі досягла 138 мільйонів, а кількість розробників програмного забезпечення – 19 мільйонів.



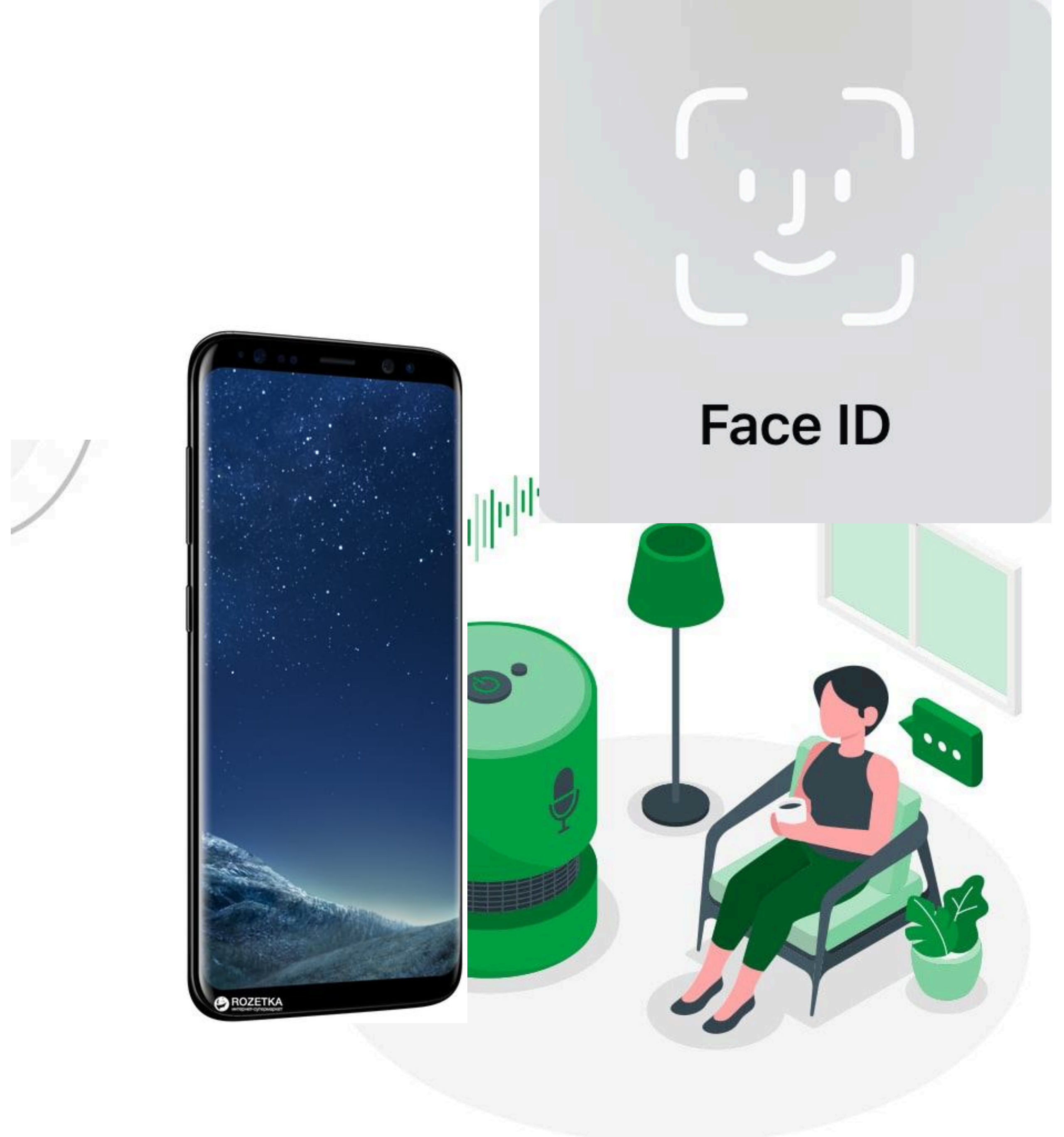
2015-2016:

Відбувається широке поширення 4G LTE, що забезпечує швидший мобільний інтернет. Розвиваються технології віртуальної (VR) та доповненої (AR) реальностей, з'являються перші VR-гарнітури та AR-додатки для смартфонів. Значно покращуються камери смартфонів, з'являються нові функції, такі як подвійні об'єктиви та оптична стабілізація зображення.



2017-2018:

Смартфони отримують безрамкові дисплеї, що робить їх більш компактними. Apple представляє Face ID та інші технології біометричної аутентифікації. Штучний інтелект (ШІ) починає використовуватися для покращення роботи камери, розпізнавання голосу та інших функцій.



2019-2020

Розгортаються мережі 5G, що забезпечують ще швидший інтернет. З'являються перші складні смартфони зі збільшеним екраном. Виробники працюють над збільшенням автономності пристроїв.



2023-2024

Штучний інтелект та машинне навчання стають ключовими технологіями, забезпечуючи персоналізований досвід. Розширюються можливості камер, з'являються нові функції, такі як покращена нічна зйомка. Розвиваються гнучкі дисплеї та нові форм-фактори. Смартфони стають центральним елементом екосистеми розумних пристроїв.



2025...

Технології мобільної розробки



Нативна

Swift, Kotlin



Кросплатформенна/Гібридна

Ionic, Flutter, Xamarin, React Native



PWA

HTML, CSS, JS



Нативна розробка

Нативні додатки створюються спеціально для певної платформи (iOS або Android) із використанням офіційних мов програмування.

- Swift (iOS) – офіційна мова для розробки додатків під iOS. Вона швидка, безпечна та добре інтегрована з екосистемою Apple.
- Kotlin (Android) – мова програмування, рекомендована Google для Android-розробки. Вона сучасна, лаконічна і має потужні засоби безпеки.

+ Плюси

- ✓ Максимальна продуктивність
- ✓ Доступ до всіх можливостей пристрою
- ✓ Висока стабільність та безпека

– Мінуси

- ✗ Дорожче у розробці (потрібно писати код окремо для iOS та Android)
- ✗ Довший цикл розробки

Кросплатформні технології

Кросплатформні фреймворки дозволяють створювати один код для обох платформ (Android і iOS), що значно зменшує витрати на розробку.

- Flutter (Google) – потужний фреймворк для створення швидких та красивих UI. Використовує мову Dart.
- React Native (Meta) – дозволяє писати мобільні додатки на JavaScript із використанням React.
- Ionic Framework (Ionic team) – так само як і реакт, але дозволяє писати на Angular, Vue та React.
- Xamarin (Microsoft) – використовує мову C# і дозволяє створювати додатки для Android, iOS та Windows.

+ Плюси

- ✓ Економія часу та ресурсів (один код для всіх платформ)
- ✓ Швидший вихід продукту на ринок
- ✓ Спрощена підтримка

– Мінуси

- ✗ Менша продуктивність у порівнянні з нативними додатками
- ✗ Обмежений доступ до деяких специфічних функцій пристрою

PWA (Progressive Web Apps)

PWA – це веб-додатки, які виглядають і працюють як нативні мобільні додатки, але відкриваються в браузері.

+ Плюси

- ✓ Не потребує встановлення
- ✓ Працює на будь-якому пристрої
- ✓ Легке оновлення

– Мінуси

- ✗ Обмежений доступ до апаратних функцій пристрою
- ✗ Працює повільніше, ніж нативні додатки чи кросплатформенні

Нативна

Swift, Kotlin



Кросплатформенна/Гібридна

Ionic, Flutter, Xamarin, React Native



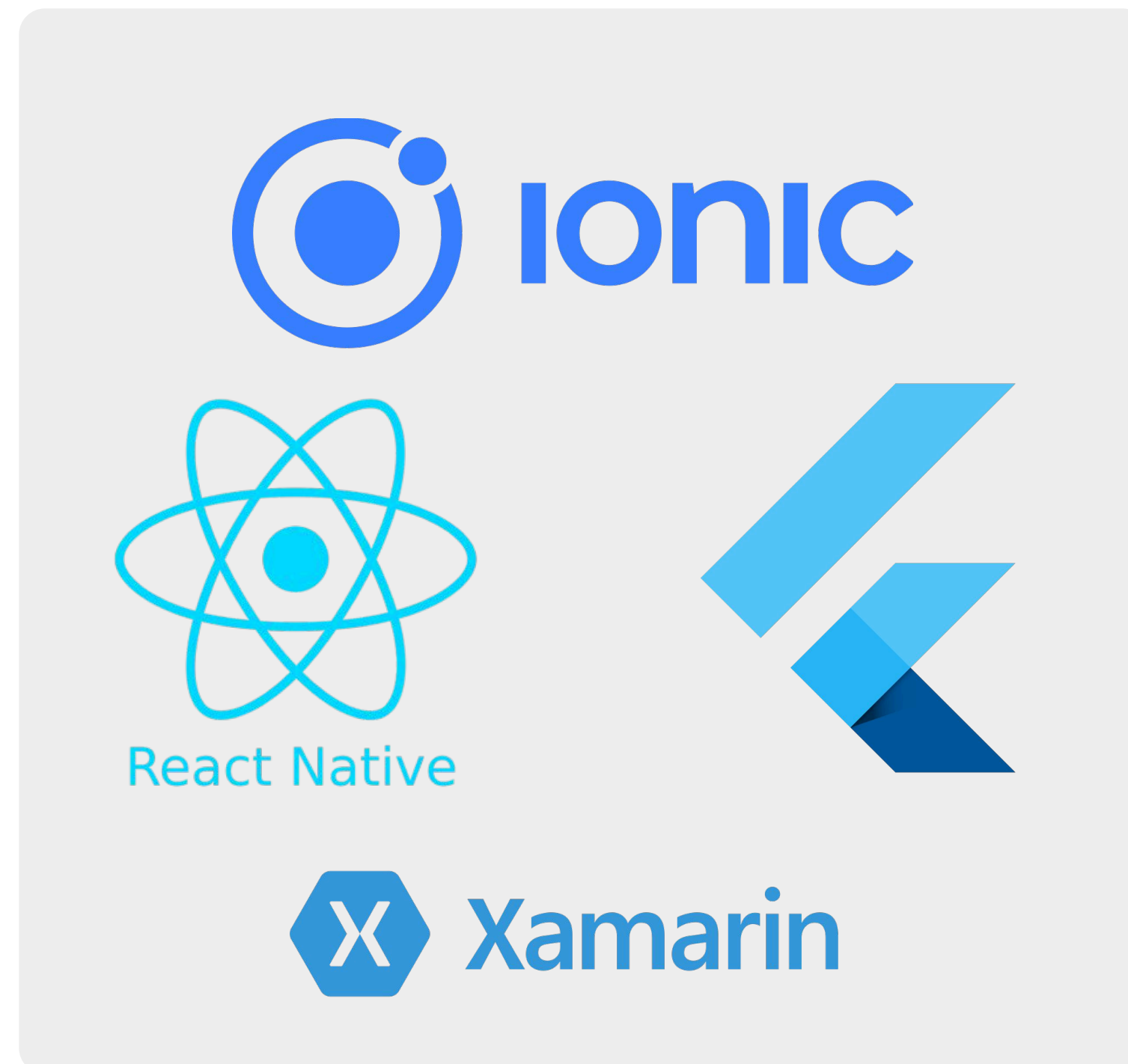
PWA

HTML, CSS, JS



Кросплатформенна/Гібридна

Ionic, Flutter, Xamarin, React Native





Чому я обрав для вас Ionic Framework?



Чому я обрав для вас Ionic Framework?

- 🚀 Один код для iOS, Android та Web
- 🎨 Готові UI-компоненти для створення сучасних інтерфейсів
- 🔗 Легка інтеграція з нативними API через Capacitor та Cordova
- 💡 Велика спільнота та багато навчальних матеріалів
- 🔧 Гнучкий і універсальний інструмент

+ Плюси

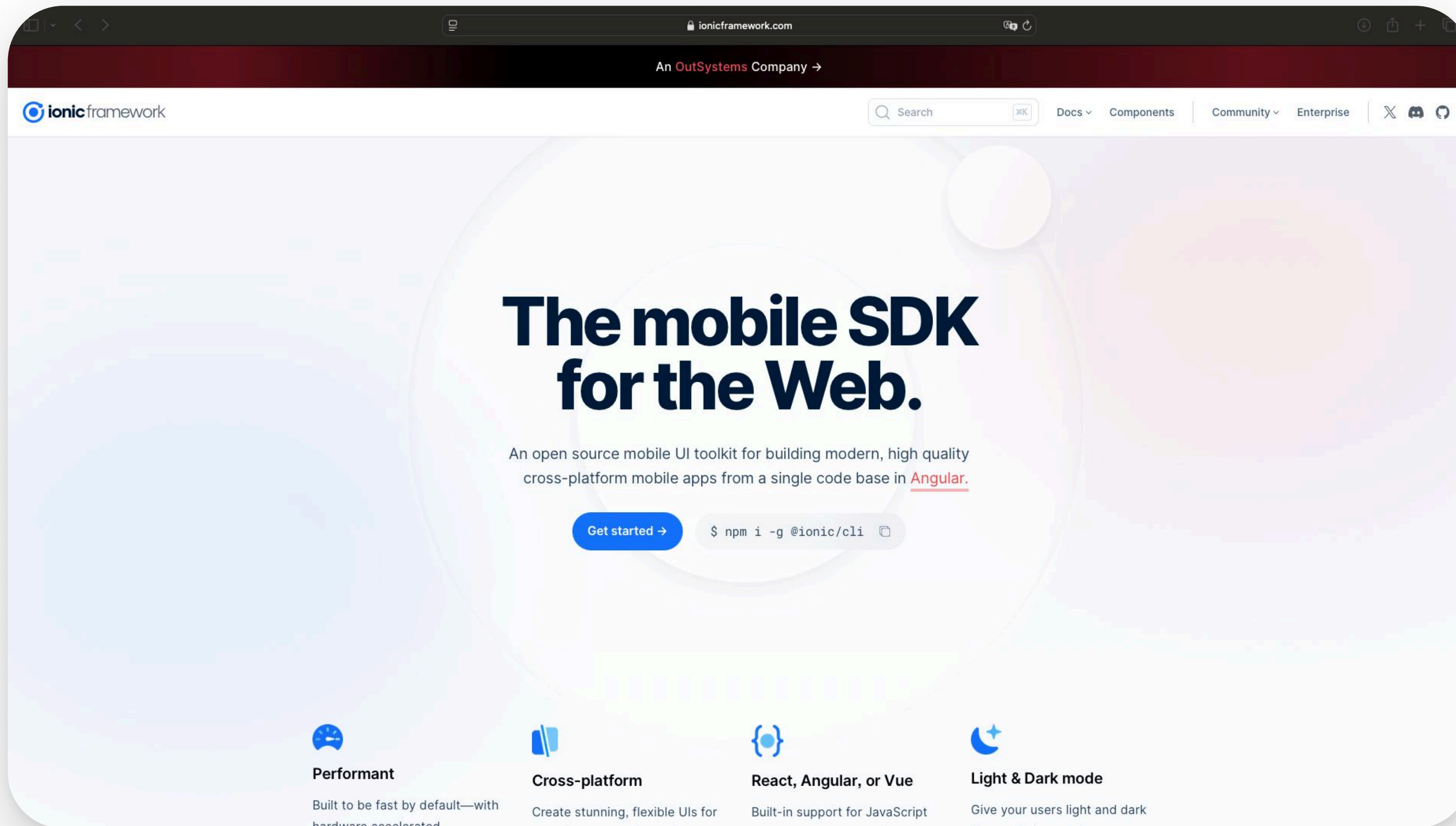
- ✓ Швидка розробка
- ✓ Простота у вивченні (підійде для веб-розробників)
- ✓ Гнучкість та підтримка багатьох платформ

— Мінуси

- ✗ Менша продуктивність порівняно з нативними рішеннями
- ✗ Деякі складні анімації можуть працювати повільніше
- 🙋 За ним не стоїть жодна велика компанія



7. Старт роботи з Ionic Framework



<https://ionicframework.com>

Технічні вимоги

IDE

IDE (VS Code чи інші)



Node.js

v22+



Simulator

Android studio(Genymotion)
чи Xcode



Завантажити Ionic CLI з npm

```
npm install -g @ionic/cli
```


Встановіть Angular

```
npm install @ionic/angular@latest --save
```



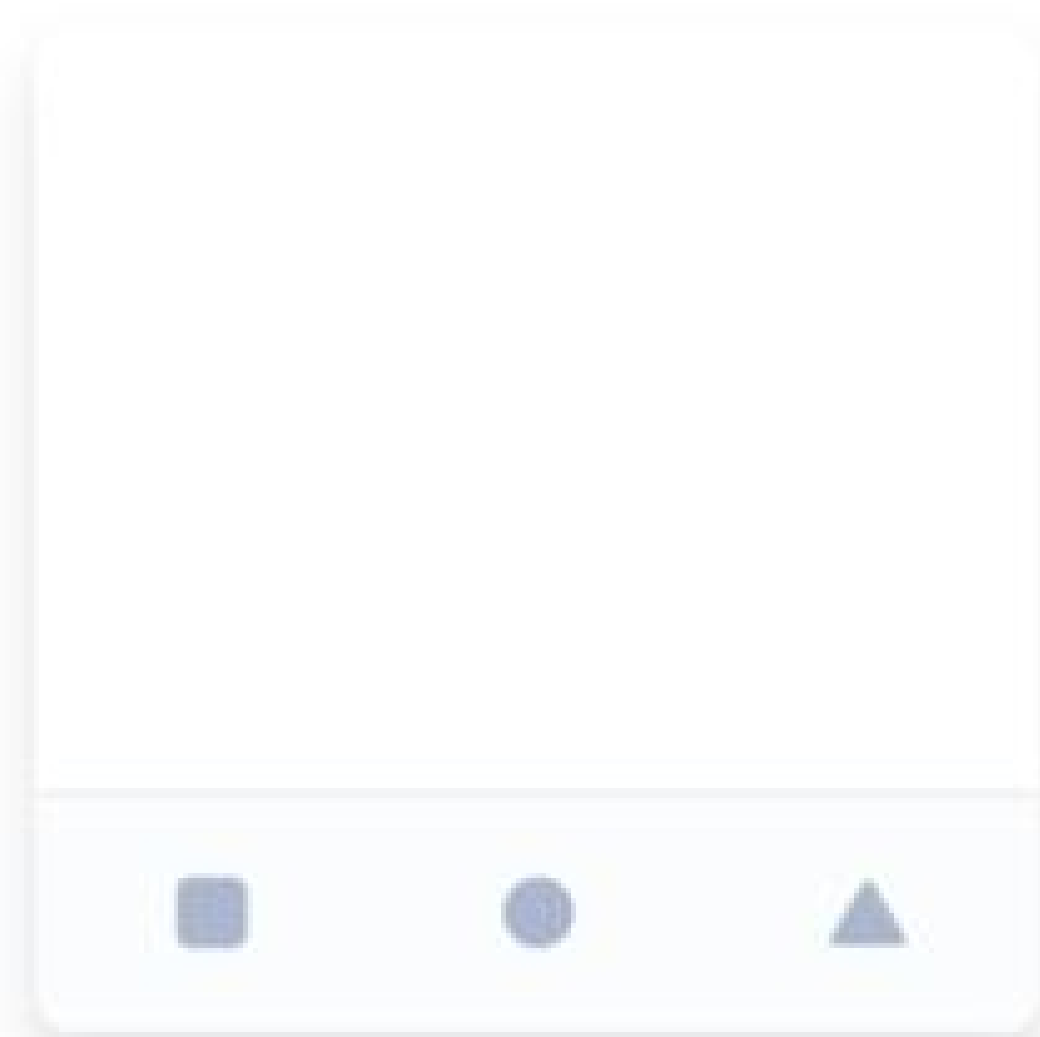

Далі я буду показувати приклади саме з використанням Angular. Але, по вашому бажанню, ви можете використовувати Vue або React

Створіть проект

ionic start



BLANK



TABS



SIDE MENU

Запустіть проект

```
cd myApp  
ionic serve
```


Симуляція додатку у браузері

<http://localhost:4200>

Структура проекту

```
src/  
├── app/  
├── assets/  
├── environments/  
├── theme/  
├── global.scss  
├── index.html  
├── main.ts  
├── polyfills.ts  
├── test.ts  
└── zone-flags.ts
```


Де кодить?

```
src/  
└─ app/  
    ├── app-routing.module.ts  
    ├── app.component.html  
    ├── app.component.spec.ts  
    ├── app.component.ts  
    └─ app.module.ts
```


Генерація пустих функцій

```
$ ionic generate [schematic] [name]
```

```
? What would you like to generate?  
> page  
  component  
  service  
  module  
  class  
  directive  
  guard
```




Q&A

