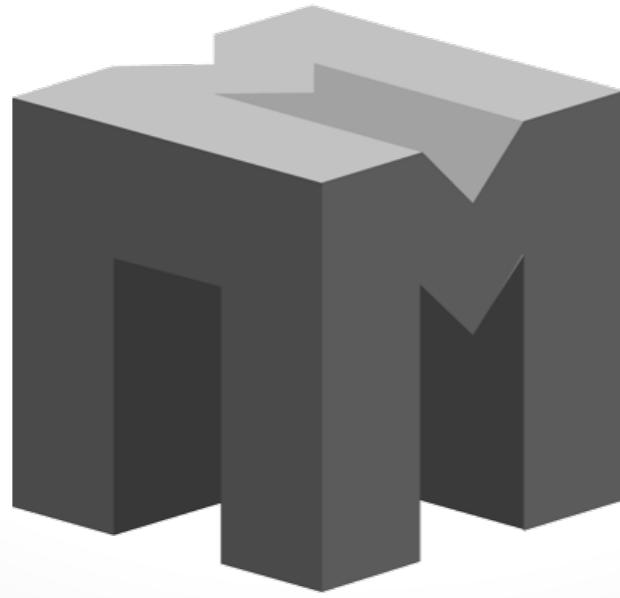


E D U C A T I O N

Surf

iOS Школа

Surf
Code
Now



Стрижов Даниил



iOS
разработчик

Стрижов Даниил



iOS
разработчик

Особенности профессии

Повседневные задачи

Чистый код

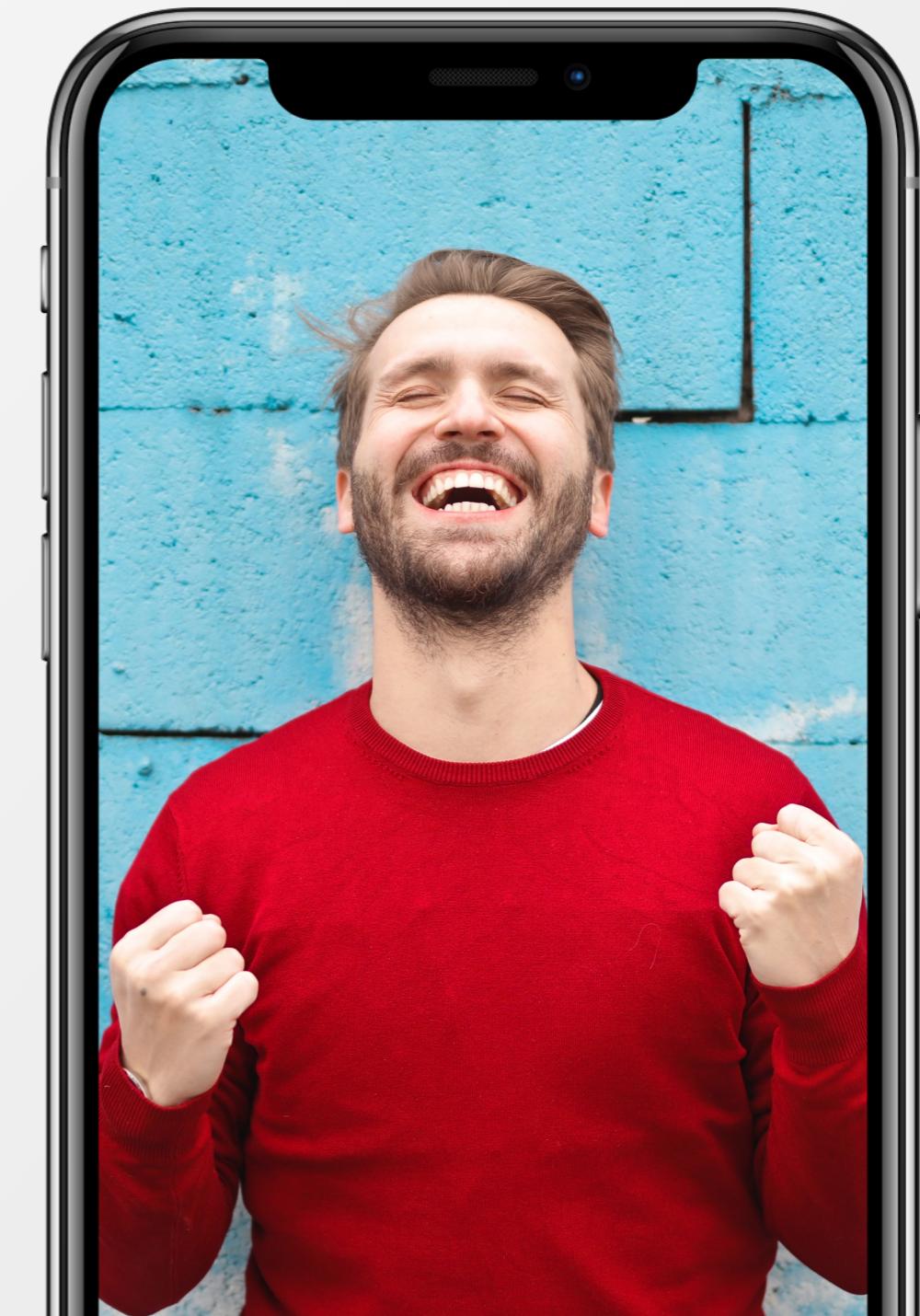
Особенности профессии

Повседневные задачи

Чистый код

По данным CNN мобильный разработчик - лучшая профессия Америки

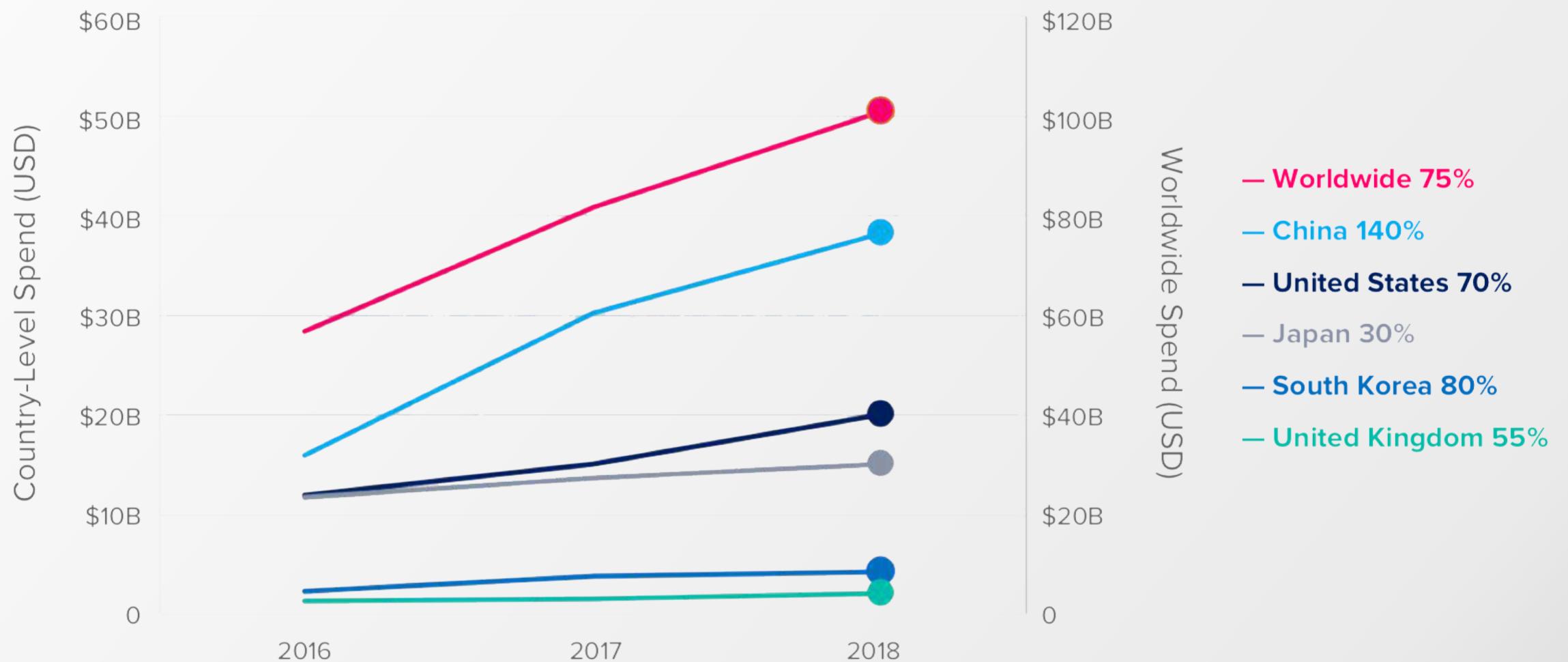
- уровень удовлетворения - А
- польза обществу - С
- уровень стресса - А



Swift входит в топ 6 любимых языков среди программистов



Рынок мобильных приложений вырос до \$101B в 2018 на 75% с 2016



Какие сферы перспективны?

- Банкинг
- E-commerce
- Крупные социальные продукты
- Зарубежные стартапы
- Российские крупные стартапы

Карьера

- Фриланс
- Работа по контракту
- Работа на стартап
- Свои проекты
- Работа в проекте за долю

Независимый разработчик

- 80,2% - Собственные проекты
- 9,3% -Работа по контракту, Фриланс



В чем выгода независимой разработки?

- Выше зарплата?
- Нет привязки к офису
- Свобода в выборе проектов?
- Нет больничных, отпусков и т.п.
- Нестабильность
- Поддержка инфраструктуры

Компания

- Продуктовая разработка
- Заказная разработка



Продуктовая разработка

- Занимается разработкой своих проектов
- Внешнего заказчика нет
- Занимается не только (в основном) разработкой

Продуктовая разработка

- Продуктовые компании крупнее и стабильнее
- Есть время и средства налаживать нормальные процессы (не всегда)
- Не обязательно будете заниматься продуктом.
- Инфраструктура, инструментарий, поддержка старых продуктов
- Нет разнообразия

Заказная разработка

- Она же аутсорсинговая, консалтинговая, сервисная
- Занимается разработкой на заказ
- У каждого проекта есть заказчик

Заказная разработка

- Проекты с 0 / поддержка проектов
- Небольшое приложение / Интернет-магазин, с оплатой, доставкой, примеркой в AR и т.п.
- Аутстаф (возможен)

Заказная разработка

- Разнообразие проектов
- Быстрое накопление опыта
- Низкий порог входа
- Возможен быстрый рост
- Редко бывают сверхсложные задачи
- Поток проектов не всегда стабилен
- Во многих компаниях нет процесса разработки

Особенности профессии

Повседневные задачи

Чистый код

Особенности профессии

Повседневные задачи

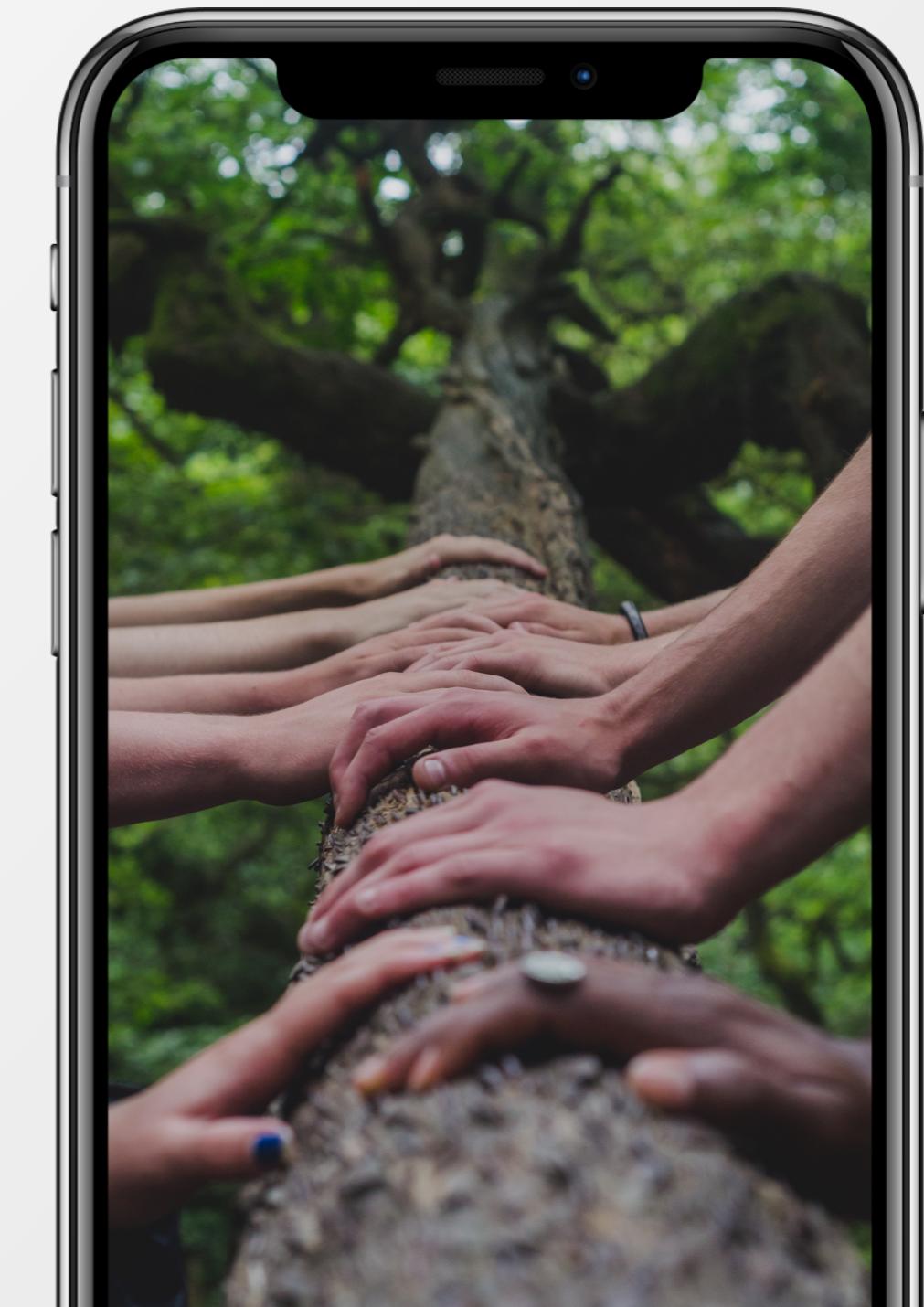
Чистый код

Повседневные задачи

- Взаимодействие в команде
- Планирование и оценка
- Работа с системой версионирования (Git, SVN)
- Code review
- Разработка

Возможный состав команды

- Разработчики
- РМ
- QA
- Дизайнеры
- Backend разработчики
- Аналитики
- Заказчик



Планирование и оценка, можно ли без них?

- Умение точно оценить задачу - важный навык
- Тщательное проектирование
- Представление задачи в виде совокупности подзадач
- Самое главное: привести мысли в своей голове в порядок
- Большая роль опыта

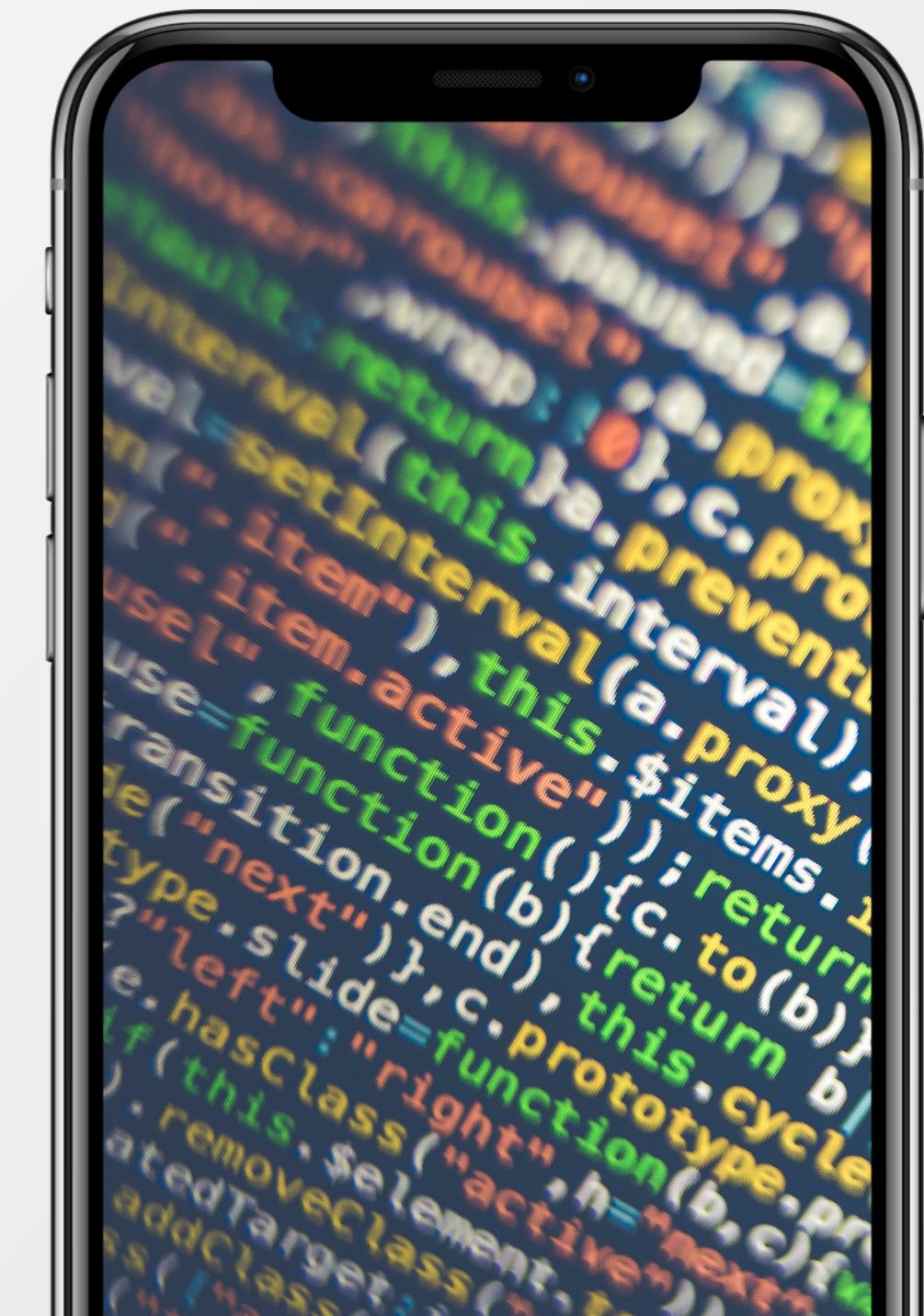
Какая польза от систем версионирования?

- Переключение между задачами
- История
- Безопасность
- Работа в команде



Зачем нужен code review?

- Контроль качества
- Обмен опытом
- Генерация идей



Разработка

- Создание продукта
- Креативность
- Общение

Особенности профессии

Повседневные задачи

Чистый код

Особенности профессии

Повседневные задачи

Чистый код

“Programs must be written for people to read, and only incidentally for machines to execute ”

–Х. Абельсон, Д.Д. Сассман “Структура и интерпретация компьютерных программ”

Чем пахнет
твой code?



Чистый код

- Написание кода – не цель
- Лучший код – не написанный
- Читаемый код важнее быстрого
- Шаблоны проектирования – не принципы программирования

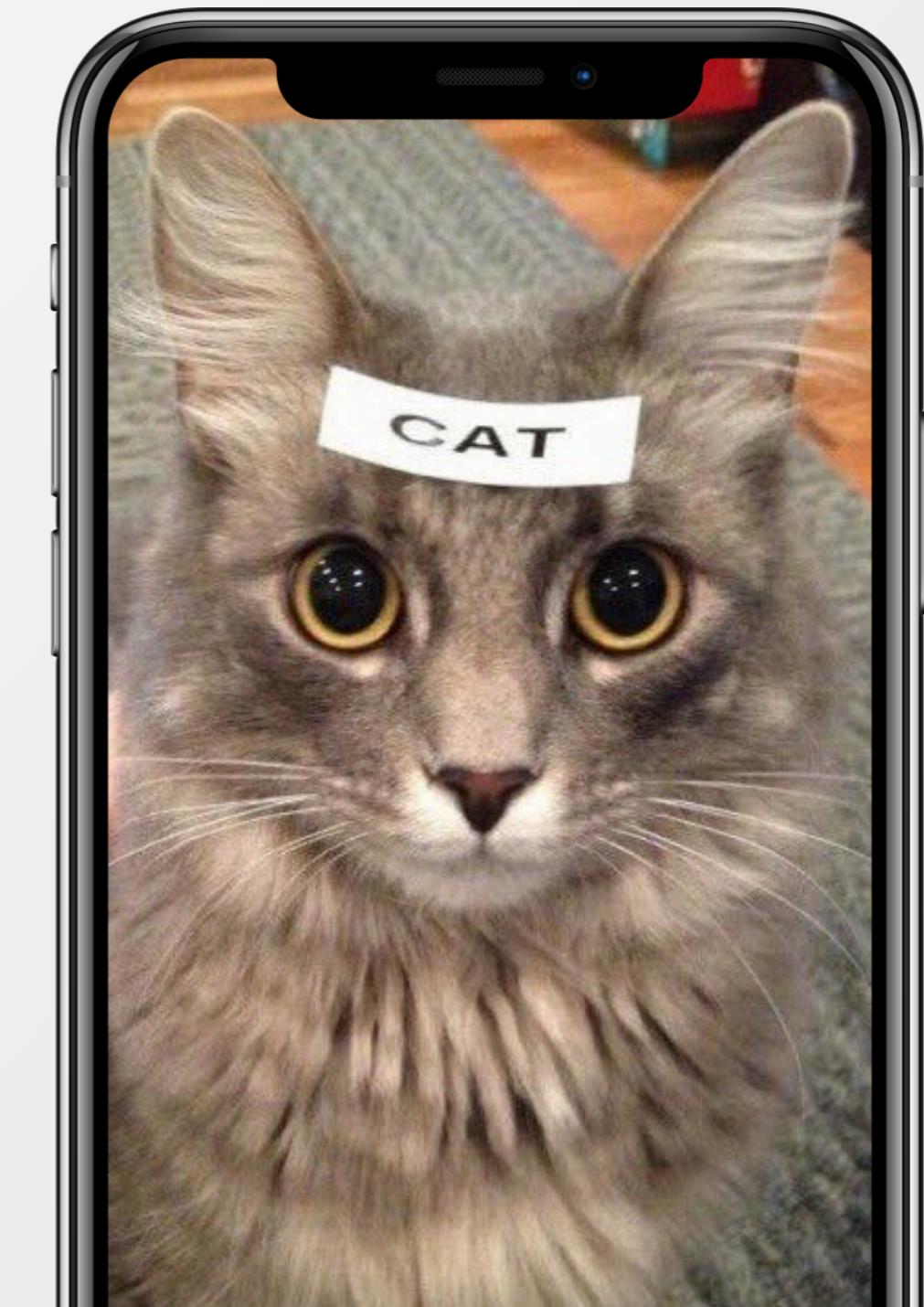
Что объединяет все дизайн принципы?

- SOLID
- KISS
- DRY
- YAGNI

Антипринцип STUPID

- Singleton (Синглтон)
- Tight Coupling (Сильная связанность)
- Untestability (Не тестируемость)
- Premature optimization (Преждевременная оптимизация)
- Indescriptive Naming (Ущербные наименования)
- Duplication (Дублирование)

Сколько комментариев содержит хороший код?



Можно ли писать код не зная паттернов проектирования?

- Пораждающие
- Структурные
- Поведенческие

Сколько нужно опыта, чтобы писать хороший код сразу?

- Рефакторинг здесь и сейчас
- Цикличность



Разбор кода

```
override func setup() {
    super.setup()
    imageView.accessibilityIgnoresInvertColors = true
    imageView.contentMode = .scaleAspectFill
    addSubview(imageView)
    gradientView.setStart(.clear, end: UIColor(white: 0, alpha: 0.8))
    addSubview(gradientView)
    captionLabel.numberOfLines = 3
    addSubview(captionLabel)
}
```

Разбор кода

```
override func setup() {
    super.setup()
    appendSubviews()
    setInitialState()
}

private func appendSubviews() {
    addSubview(imageView)
    addSubview(gradientView)
    addSubview(captionLabel)
}

private func setInitialState() {
    imageView.accessibilityIgnoresInvertColors = true
    imageView.contentMode = .scaleAspectFill
    gradientView.setStart(.clear, end: Constants.gradientEndColor)
    captionLabel.numberOfLines = Constants.captionLabelNumberOfLines
}
```

Разбор кода

```
override func reset() {
    super.reset()
    imageView.wmf_reset()
    captionLabel.text = nil
}

let ratio: CGFloat = 1.02

override func sizeThatFits(_ size: CGSize, apply: Bool) -> CGSize {
    var size = super.sizeThatFits(size, apply: apply)
    if size.width != UIView.noIntrinsicMetric {
        size.height = round(ratio * size.width)
    }
}
```

```

override func sizeThatFits(_ size: CGSize, apply: Bool) -> CGSize {
    var size = super.sizeThatFits(size, apply: apply)
    if size.width != UIView.noIntrinsicMetric {
        size.height = round(ratio * size.width)
    }else if size.height != UIView.noIntrinsicMetric {
        size.width = round(size.height / ratio)
    }
    if apply {
        let boundsInsetByMargins = CGRect(origin: .zero, size: size).inset(by:
layoutMargins)
        imageView.frame = CGRect(origin: .zero, size: size)
        if captionLabel.wmf_hasAnyNonWhitespaceText {
            captionLabel.isHidden = false
            gradientView.isHidden = false
            var labelFrame = captionLabel.wmf_preferredFrame(at:
boundsInsetByMargins.origin, maximumSize: boundsInsetByMargins.size, minimumSize:
CGSize(width: boundsInsetByMargins.size.width, height: UIView.noIntrinsicMetric),
alignedBy: semanticContentAttribute, apply: false)
            let extraBottomPadding: CGFloat = 5.0
            labelFrame.origin = CGPoint(x: labelFrame.origin.x, y: size.height -
labelFrame.height - layoutMargins.bottom - extraBottomPadding)
            captionLabel.frame = labelFrame
            let gradientOriginY = labelFrame.minY - layoutMargins.bottom
            gradientView.frame = CGRect(x: 0, y: gradientOriginY, width:
size.width, height: size.height - gradientOriginY + extraBottomPadding)
        } else {
            captionLabel.isHidden = true
            gradientView.isHidden = true
        }
    }
    return size
}

```

Разбор кода

```
override func sizeThatFits(_ size: CGSize, apply: Bool) -> CGSize {  
    var size = super.sizeThatFits(size, apply: apply)  
    size = getSize(for: size, ratio: Constants.sizeRatio)  
  
    if apply {  
        setViewsVisibility()  
        setViewsFrame(size)  
    }  
  
    return size  
}
```

Разбор кода

```
private func getSize(for size: CGSize, ratio: CGFloat) -> CGSize {  
    var size = size  
  
    if size.width != UIView.noIntrinsicMetric {  
        size.height = round(ratio * size.width)  
    } else if size.height != UIView.noIntrinsicMetric {  
        size.width = round(size.height / ratio)  
    }  
  
    return size  
}
```

Разбор кода

```
private func setViewsVisibility() {
    let isHidden = !captionLabel.wmf_hasAnyNonWhitespaceText
    captionLabel.isHidden = isHidden
    gradientView.isHidden = isHidden
}
```

Разбор кода

```
private func setViewsFrame(_ size: CGSize) {
    setImageViewFrame(size)

    guard captionLabel.wmf_hasAnyNonWhitespaceText else {
        return
    }

    let captionLabelFrame = setCaptionLabelFrame(size)
    setGradientViewFrame(size: size,
                         captionLabelFrame: captionLabelFrame)
}
```

Разбор кода

```
private func setImageViewFrame(_ size: CGSize) {  
    imageView.frame = .init(origin: .zero, size: size)  
}
```

Разбор кода

```
@discardableResult
private func setCaptionLabelFrame(_ size: CGSize) -> CGRect {
    let boundsInsetByMargins = CGRect(origin: .zero, size: size)
        .inset(by: layoutMargins)

    let minimumSize = CGSize(width: boundsInsetByMargins.size.width,
                             height: UIView.noIntrinsicMetric)

    var labelFrame = captionLabel
        .wmf_preferredFrame(at: boundsInsetByMargins.origin,
                            maximumSize: boundsInsetByMargins.size,
                            minimumSize: minimumSize,
                            alignedBy: semanticContentAttribute,
                            apply: false)

    let originY = size.height
        - labelFrame.height
        - layoutMargins.bottom
        - Constants.extraBottomPadding

    labelFrame.origin = .init(x: labelFrame.origin.x, y: originY)
    captionLabel.frame = labelFrame

    return labelFrame
}
```

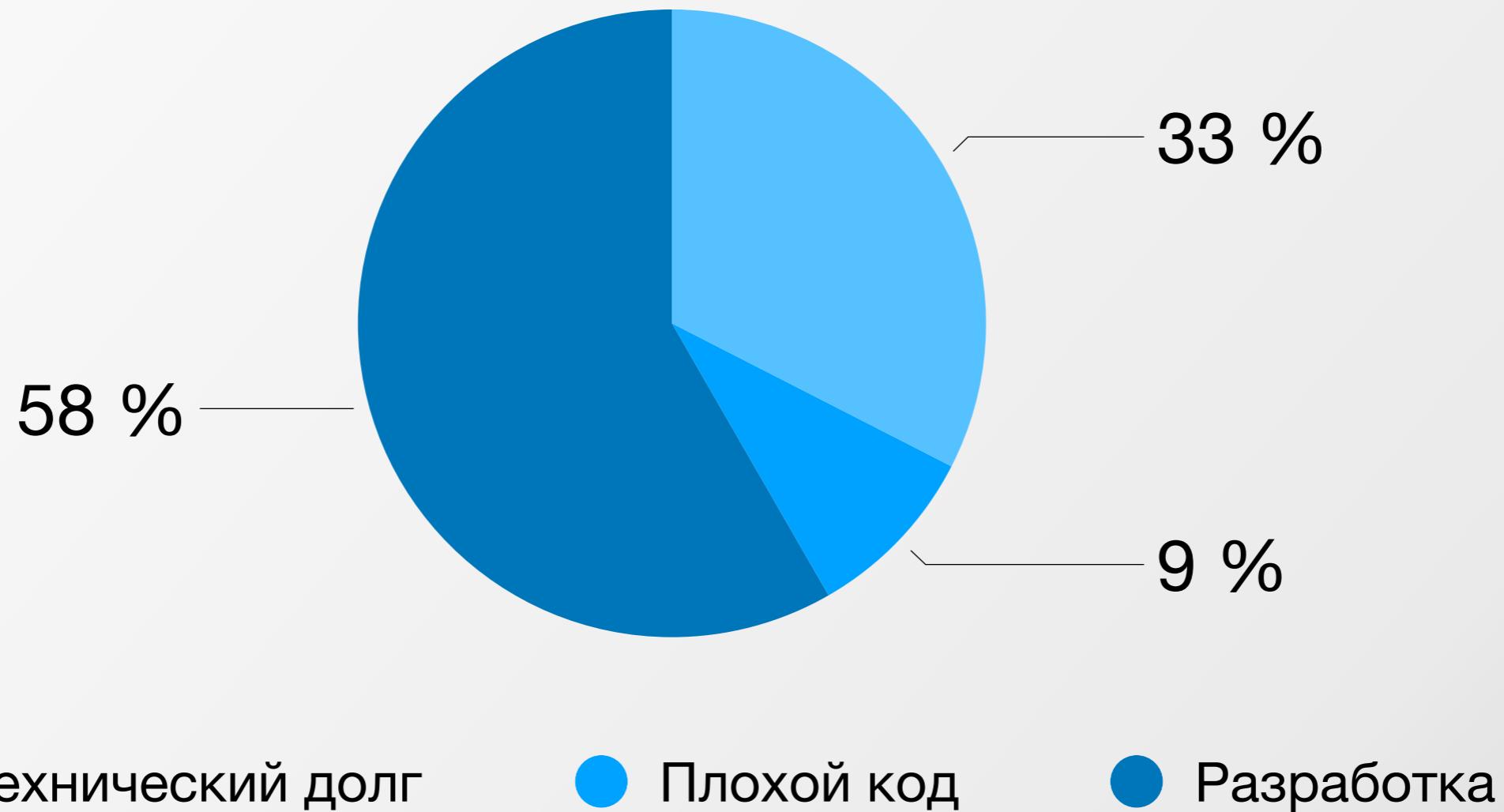
Разбор кода

```
private func setGradientViewFrame(size: CGSize,  
                                 captionLabelFrame: CGRect) {  
  
    let gradientOriginY = captionLabelFrame.minY  
        - layoutMargins.bottom  
  
    let height = size.height  
        + gradientOriginY  
        + Constants.extraBottomPadding  
  
    gradientView.frame = .init(x: .zero,  
                               y: gradientOriginY,  
                               width: size.width,  
                               height: height)  
}
```

Разбор кода

- До 27 строк
- После 71

\$85B / год - стоимость плохого кода



Особенности профессии

Повседневные задачи

Чистый код

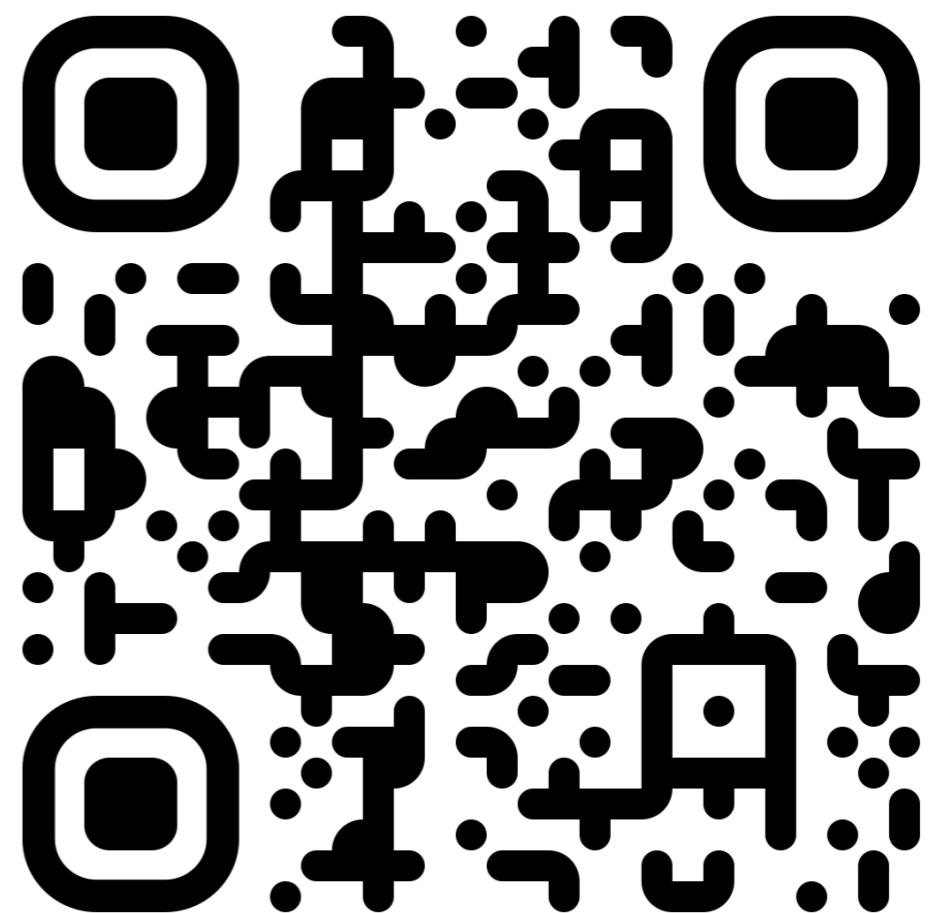
Особенности профессии

Повседневные задачи

Чистый код

Материалы

[http://bit.ly/
iossurf19lecture9](http://bit.ly/iossurf19lecture9)





E D U C A T I O N

Surf