



Процесс автоматизации тестирования микрофронтенда

или как сделать все удобно



Tinkoff

Кириенко Анна

Старший инженер по
обеспечению качества

@ a.s.kirienko@tinkoff.ru

TG @an_kirienko



ТИНЬКОФФ

О чем будем говорить?

- Контекст – конструктор форм
- Начало. Назад в прошлое
- Развитие. Первые шаги создания Component Object
- Оптимизация. Улучшение тестирования конструктора форм
- Стало ли лучше?



ТИНЬКОФФ

О чем будем говорить?

- Контекст – конструктор форм
- **Начало. Назад в прошлое**
- Развитие. Первые шаги создания Component Object
- Оптимизация. Улучшение тестирования конструктора форм
- Стало ли лучше?



ТИНЬКОФФ

О чем будем говорить?

- Контекст – конструктор форм
- Начало. Назад в прошлое
- Развитие. Первые шаги создания Component Object
- Оптимизация. Улучшение тестирования конструктора форм
- Стало ли лучше?



ТИНЬКОФФ

О чем будем говорить?

- Контекст – конструктор форм
- Начало. Назад в прошлое
- Развитие. Первые шаги создания Component Object
- Оптимизация. Улучшение тестирования конструктора форм
- Стало ли лучше?



ТИНЬКОФФ

О чем будем говорить?

- Контекст – конструктор форм
- Начало. Назад в прошлое
- Развитие. Первые шаги создания Component Object
- Оптимизация. Улучшение тестирования конструктора форм
- Стало ли лучше?



ТИНЬКОФФ

Кому будет полезно?

- Начинаете внедрять автоматизацию



ТИНЬКОФФ

Кому будет полезно?

- Начинаете внедрять автоматизацию
- Уже автоматизируете

КОНТЕКСТ

Платформа Привлечения

Т ТИНЬКОФФ Частным лицам Бизнесу Премиум Еще Личный кабинет

Кредитные карты

Кредитные карты — это платежные карты, позволяющие оплачивать покупки за счет средств банка. Расплачивайтесь кредиткой и получайте кэшбэк бонусами за любые покупки

Все карты Премиальные Путешественнику Автомобилисту Геймеру Покупки

Выберите карту в лимитированном дизайне

Кредитная карта «Тинькофф Платинум»


Наша лучшая кредитная карта в лимитированном дизайне. Одобрим за 2 минуты, доставим бесплатно

До 1 000 000 Р
Кредитный лимит

До 12 месяцев
Беспроцентная рассрочка

590 Р в год
Стоимость обслуживания

Оформить карту Подробнее



Бесплатно доставим карту уже сегодня

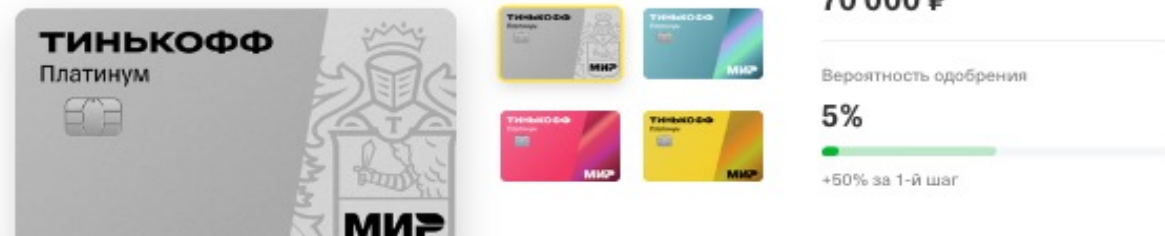
- 1 Заполните онлайн-заявку — вам не нужно посещать банк
- 2 Узнайте решение банка сразу после заполнения заявки
- 3 Банк бесплатно доставит карту, куда удобно. Прямо в руки

Настройте свою карту


Вы выбрали карту с лимитом **70 000 Р**

Вероятность одобрения **5%**

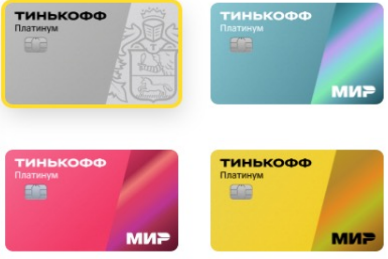

+50% за 1-й шаг



Кредитная форма


Т ТИНЬКОФФ Частным лицам Бизнесу Премиум Еще Личный кабинет 

Настройте свою карту




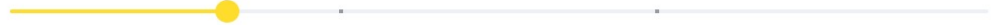
Вы выбрали карту с лимитом **70 000 ₽**

Вероятность одобрения **5%**


 +50% за 1-й шаг

Желаемый кредитный лимит

70 000 ₽ 



15 000 ₽ 100 000 ₽ 500 000 ₽ 1 000 000 ₽

Для чего вам нужна карта 

К какому номеру привязать карту?

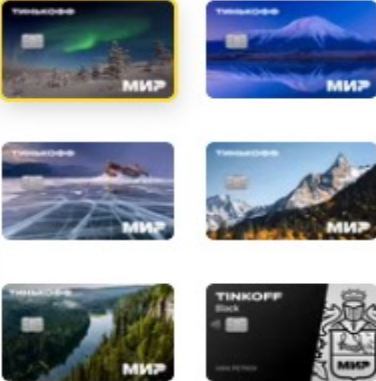

Мобильный телефон*

Заполняя форму, я принимаю [условия передачи информации](#) и даю согласие на получение информации о продуктах и услугах Банка и партнеров

Далее

Дебетовая форма

Выберите дизайн карты



Уже заполнено
5%
+60% за контактную информацию

Выберите повышенный кэшбэк (четыре категории) ▾

Фамилия, имя и отчество* +20%

Мобильный телефон* Электронная почта

Дата рождения*

Гражданство РФ?

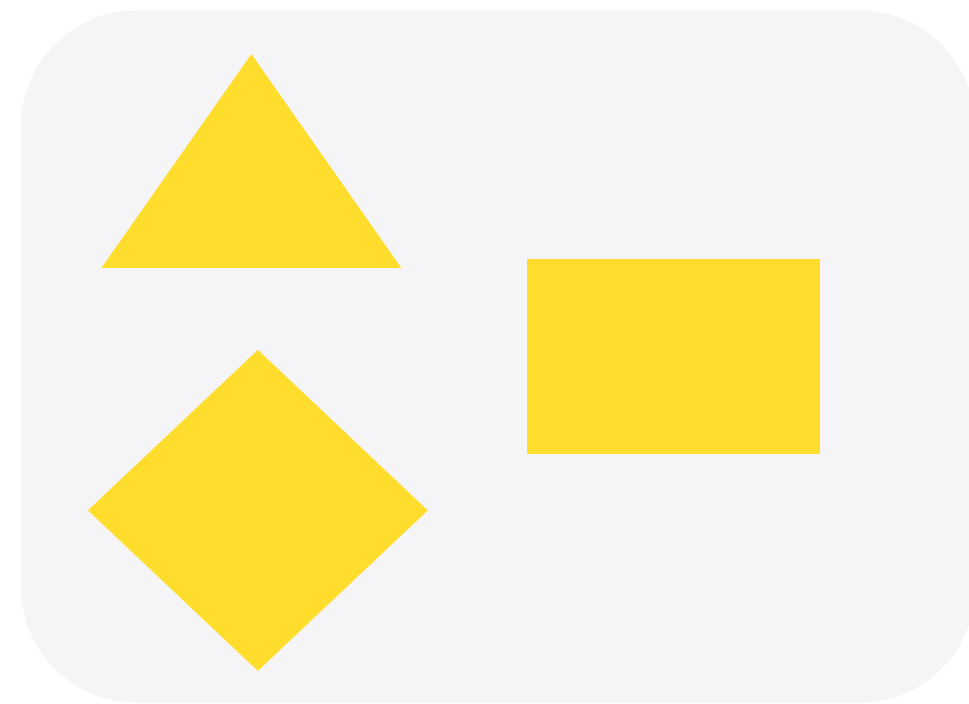
Да Нет

Заполняя форму, я принимаю [условия передачи информации](#) и даю [согласие на получение информации о продуктах и услугах Банка и партнеров](#)

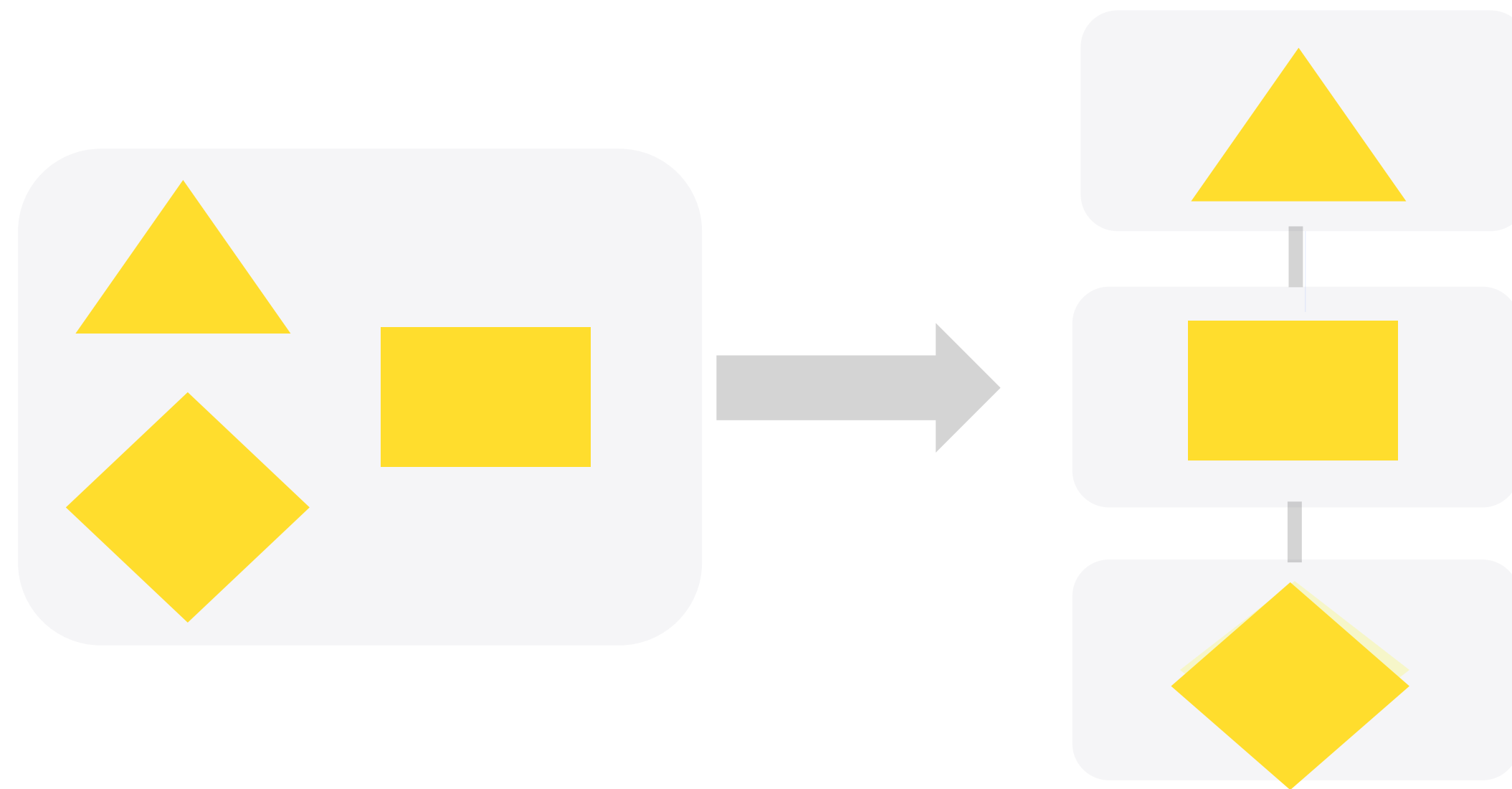
Оформить

Конструктор форм

Монолитная архитектура



Микрофронтонтовая архитектура



Микрофронтенд в Тинькофф

Настройте свою карту

Вы выбрали карту с лимитом **70 000 ₽**

Вероятность одобрения **5%**

+50% за 1-й шаг

Желаемый кредитный лимит

70 000 ₽

15 000 ₽ 100 000 ₽ 500 000 ₽ 1 000 000 ₽

Для чего вам нужна карта

К какому номеру привязать карту?

Мобильный телефон*

Заполняя форму, я принимаю условия передачи информации и даю согласие на получение информации о продуктах и услугах Банка и партнеров

Далее



Настройте свою карту

Вы выбрали карту с лимитом **70 000 ₽**

Вероятность одобрения **5%**

+50% за 1-й шаг

Желаемый кредитный лимит

70 000 ₽

15 000 ₽ 100 000 ₽ 500 000 ₽ 1 000 000 ₽

Для чего вам нужна карта

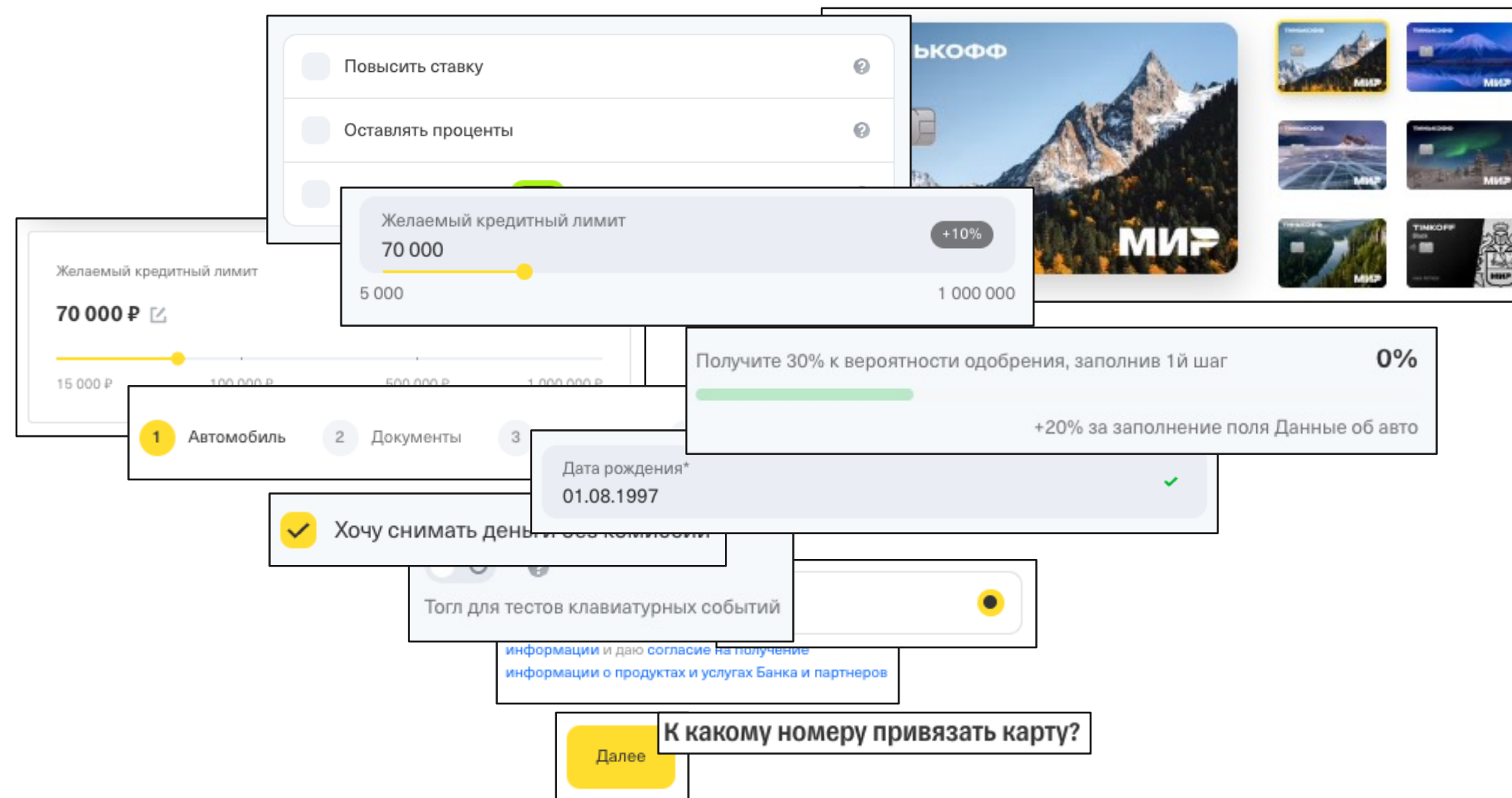
К какому номеру привязать карту?

Мобильный телефон*

Заполняя форму, я принимаю условия передачи информации и даю согласие на получение информации о продуктах и услугах Банка и партнеров

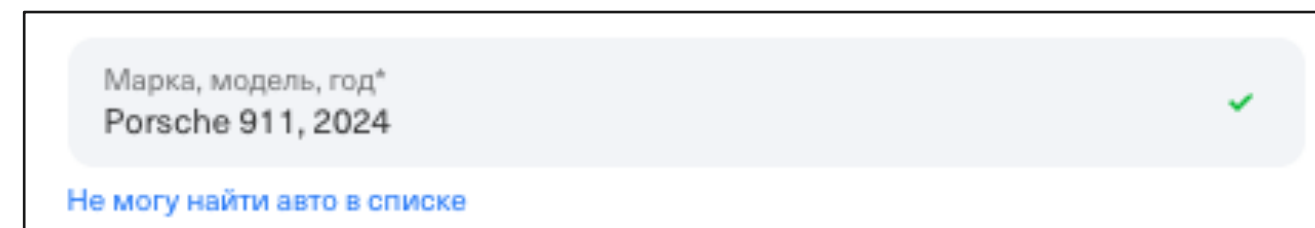
Далее

Конструктор форм FormBuilder



Простые поля

Suggest

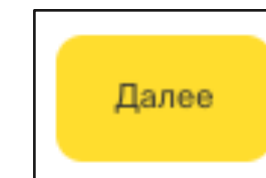


Марка, модель, год*
Porsche 911, 2024 ✓

[Не могу найти авто в списке](#)

The image shows a light blue rounded rectangular input field. Inside, the text 'Марка, модель, год*' is at the top, followed by 'Porsche 911, 2024' and a green checkmark on the right. Below the input field, there is a blue link that says 'Не могу найти авто в списке'.

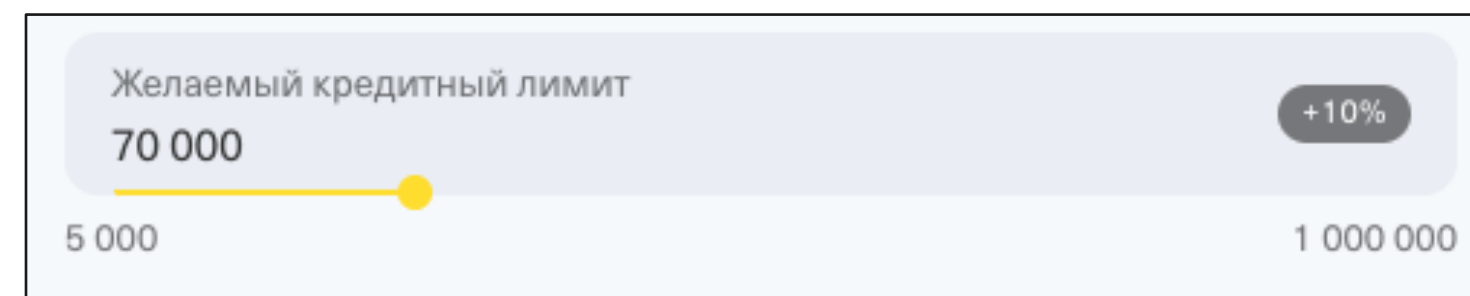
Button



Далее

The image shows a yellow rounded rectangular button with the Russian word 'Далее' (Next) in the center.

Input range

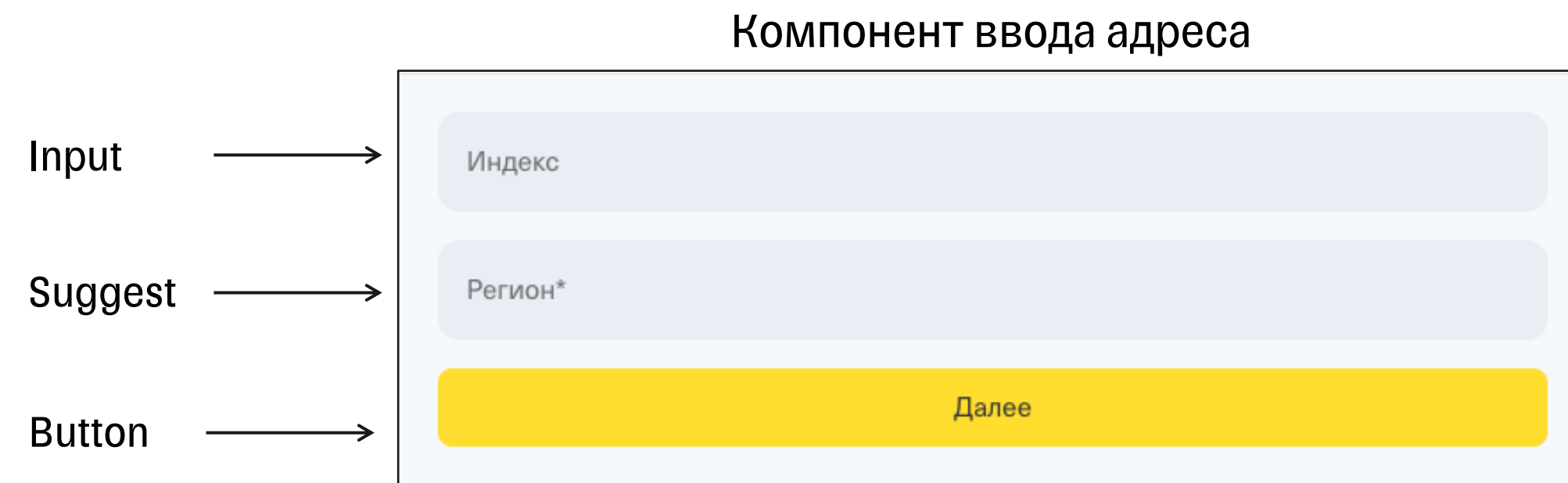


Желаемый кредитный лимит
70 000 +10%

5 000 1 000 000

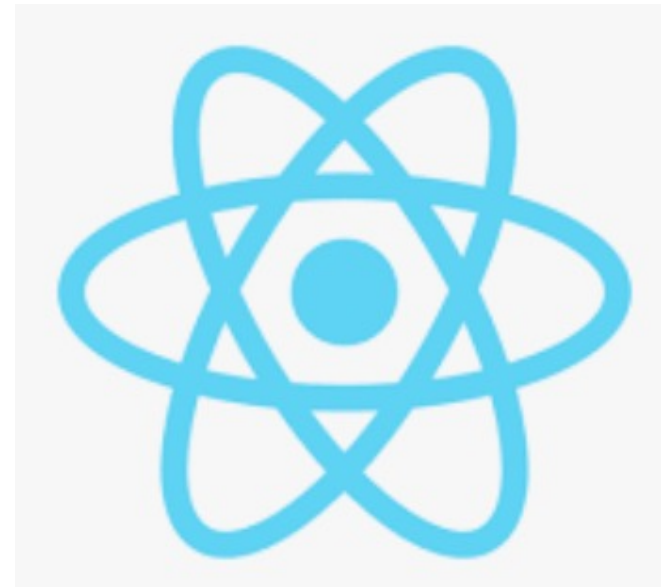
The image shows a light blue rounded rectangular range input field. The title is 'Желаемый кредитный лимит'. The current value is '70 000' and there is a '+10%' button on the right. Below the input field, there is a horizontal range from '5 000' to '1 000 000' with a yellow slider knob positioned at the current value.

Композиторы



Начало. Назад в прошлое

Наш стек 2019 год



React



CodeceptJS



ТИНЬКОФФ

Что было хорошо?

- **Понятный синтаксис методов тестов**
- Пошаговый вывод прохождения тестов
- Асинхронность
- Page objects, helpers



ТИНЬКОФФ

Что было хорошо?

- Понятный синтаксис методов тестов
- Пошаговый вывод прохождения тестов
- Асинхронность
- Page objects, helpers



ТИНЬКОФФ

Что было хорошо?

- Понятный синтаксис методов тестов
- Пошаговый вывод прохождения тестов
- **Асинхронность**
- Page objects, helpers



ТИНЬКОФФ

Что было хорошо?

- Понятный синтаксис методов тестов
- Пошаговый вывод прохождения тестов
- Асинхронность
- Page objects, helpers



ТИНЬКОФФ

Но были и проблемы

- **Читаемость тестов**
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Читаемость

ТЕСТОВ

```
const stories = require('./__stories__/input.stories.js')

Feature('Core/Input/Functional')

Scenario('Поле необязательно и значение не введено', async (I) => {
  await I.amOnPage(stories('inputUnReq'))

  await I.fillField('Nickname', '')
  await I.pressKey('Tab')
  await I.dontSeeElement('Поле обязательно для заполнения')
})
```



ТИНЬКОФФ

Но были и проблемы

- **Читаемость тестов**
- **Helpers**
- XPath и CSS
- Разная верстка
- Дублирование кода

Хелперы конструктора форм

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  
  async seeInputSelectIsFocused(inputName) {  
    await browser.isFocused(this.input(inputName))  
  }  
}
```

Хелперы форм

```
import BaseHelper from '@tinkoff-fb/codeceptjs/helpers/baseHelper'
```

```
class ErrorHandler extends BaseHelper {  
  errorByFieldLocator = (locator, error) =>  
    locate(  
      `//div[contains(@class, "errorMessage)][descendant::${locator}][contains(., "${error}")]`,  
    )  
}
```

```
  async seeErrorByFieldLocator(locator, errorMessage) {  
    await browser.seeElement(this.errorByFieldLocator(locator, errorMessage))  
  }  
}
```

```
module.exports = ErrorHandler
```



ТИНЬКОФФ

Но были и проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

XPath и CSS

```
const stories = require('./__stories__/input.stories.js')
```

```
const locators = {  
  checkMark: '//div[contains(@class, "input-Input__checkIcon")]',  
  error: locate('//div[contains(@class, "errorBlock__showed")]'),  
}
```

```
Feature('Core/Input/Functional')
```

```
Scenario('Поле необязательно и значение не введено', async (I) => {  
  await I.amOnPage(stories('inputUnReq'))  
  await I.fillField(locators.input, '')  
  await I.pressKey('Tab')  
  await I.dontSeeElement(locators.error)  
  await I.dontSeeElement(locators.checkMark)  
}
```

XPath и CSS

```
const OPTION_TITLE_CLASS = 'fb-molecules-SelectItem-SelectItem__title'  
const ERROR_MESSAGE_CLASS = 'fb-molecules-Error-Error__errorMessage'
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```




ТИНЬКОФФ

Но были и проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Разная верстка

```
class="fb-fields-Input-Input__input"
```

```
class="node_modules-@platform-ui-Input-module__root"
```

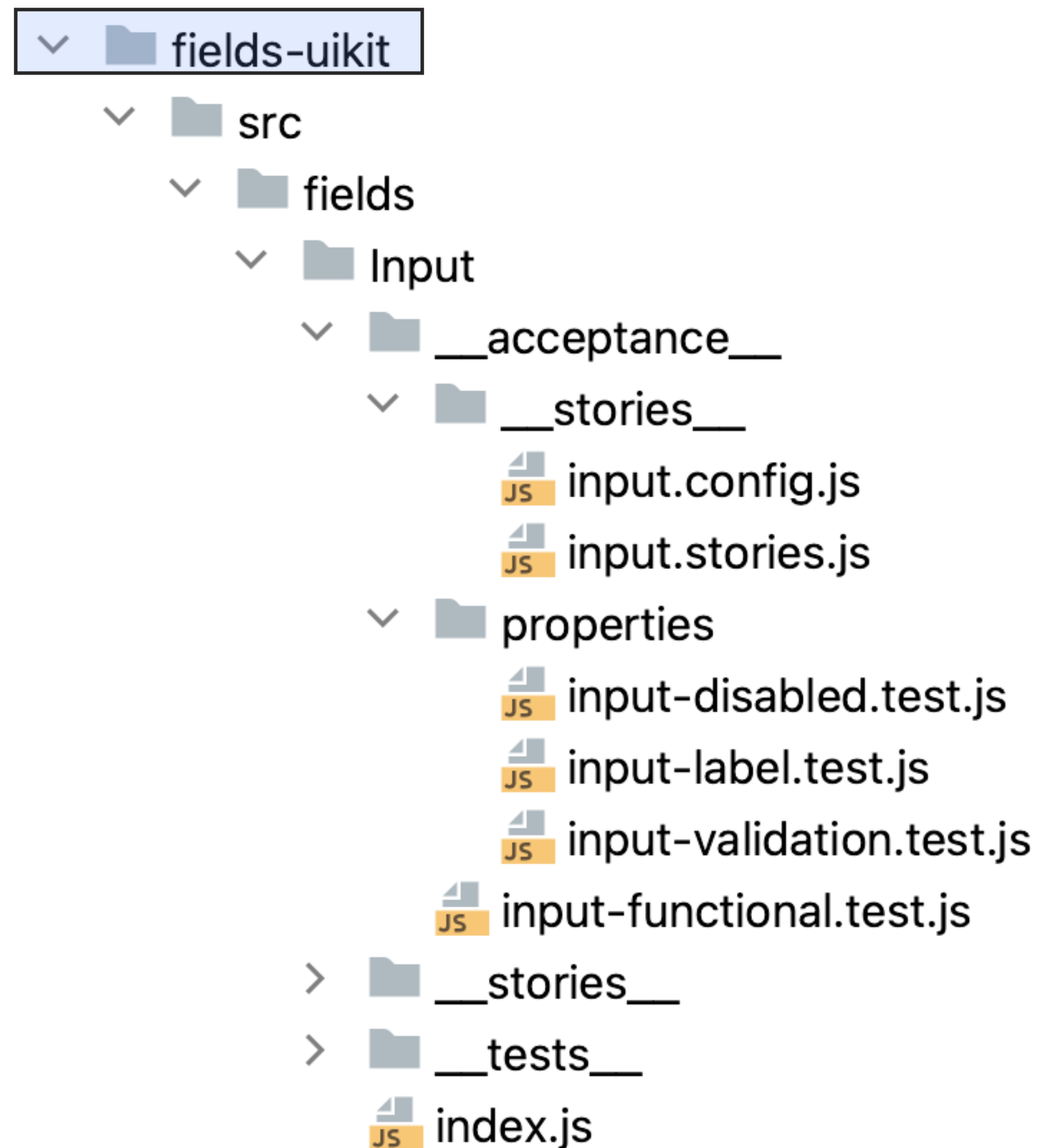
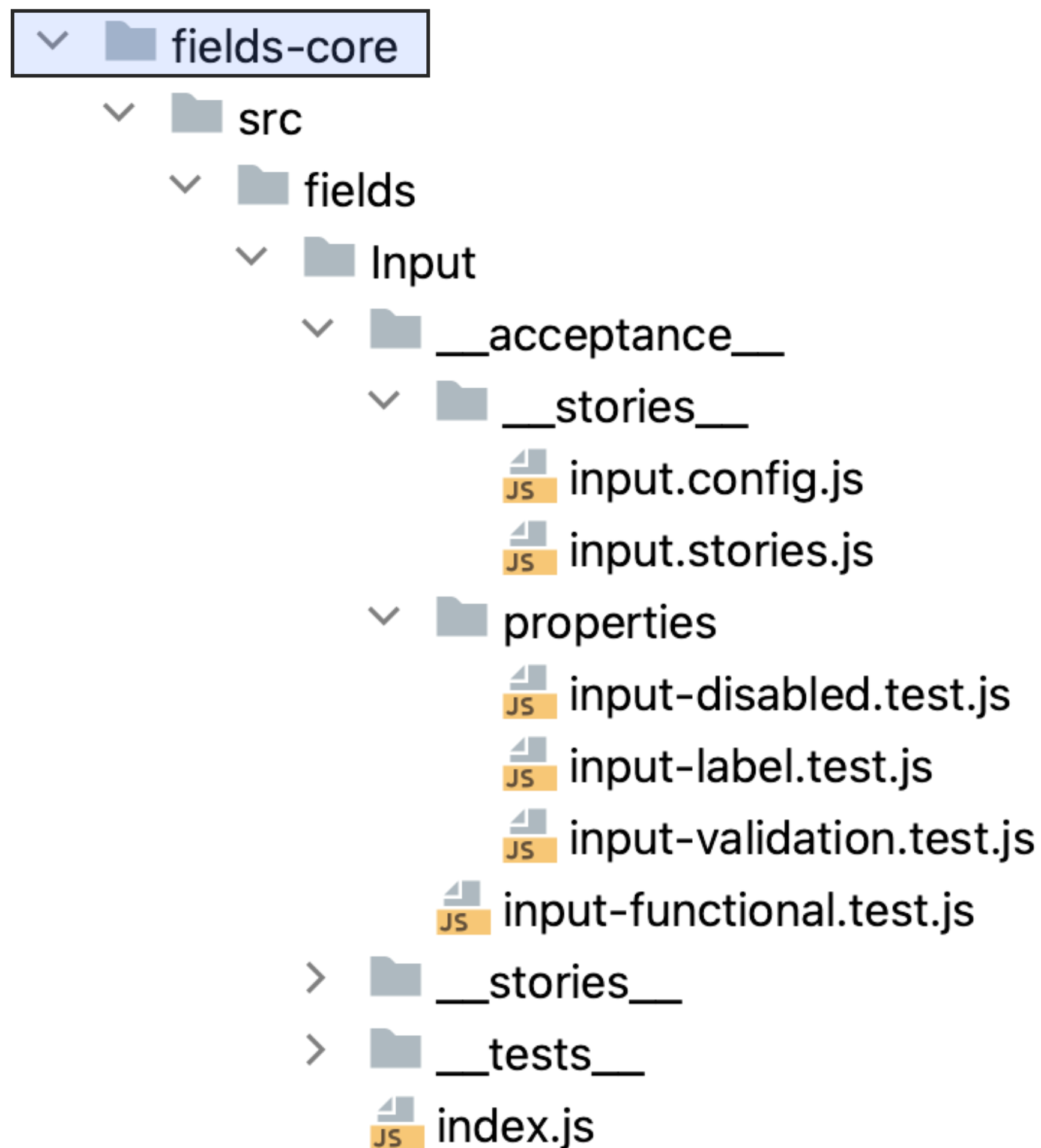


ТИНЬКОФФ

Но были и проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Дублирование



Дублирование

```
const OPTION_TITLE_CLASS = locate('//div[@data-qa-type="fb-core/select.item.title"]')

class InputSelectHelper extends BaseHelper {
  input(inputName) {
    return inputName ? locate(inputName) : ''
  }
  ...
}
```

Дублирование

```
const OPTION_TITLE_CLASS = locate('//div[@data-qa-type="fb-core/select.item.title"]' )

class InputSelectHelper extends BaseHelper {
  input(inputName) {
    return inputName ? locate(inputName) : ''
  }
  ...
}
```

```
const OPTION_TITLE_CLASS = locate('//div[@data-qa-type="uikit/select.item.title"]' )

class InputSelectHelper extends BaseHelper {
  input(inputName) {
    return inputName ? locate(inputName) : ''
  }
  ...
}
```

Дублирование

```
const OPTION_TITLE_CLASS = locate('//div[@data-qa-type="fb-core/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
}
```

```
const OPTION_TITLE_CLASS = locate('//div[@data-qa-type="fb-core/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
}
```

```
co  
  ...
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
}
```

```
...
```

Дублирование

```
const OPTION_TITLE = locate(' //div[@data-qa-type="fb-core/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
const OPTION_TITLE_CLASS = locate(' //div[@data-qa-type="uikit/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
const OPTION_TITLE_CLASS = locate(' //div[@data-qa-type="uikit/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```


Дублирование

```
const OPTION_TITLE_CLASS = locate(' //div[@data-qa-type="fb-core/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
const OPTION_TITLE_CLASS = locate(' //div[@data-qa-type="fb-core/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
const OPTION_TITLE_CLASS = locate(' //div[@data-qa-type="fb-core/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
...
```

Дублирование

```
const OPTION_TITLE
```

```
class
```

```
input
```

```
class
```

```
input
```

```
return
```

```
...
```

```
...
```

```
class
```

```
input
```

```
return
```

```
...
```

```
class
```

```
input
```

```
return
```

```
...
```

```
const OPTION_TITLE = locate(' //div[@data-qa-type="fb
```

```
const OPTION_TITLE = locate(' //div[@data-qa-type="fb
```

```
const OPTION_TITLE_CLASS = locate(' //div[@data-qa-type="uikit/select.item.title"]')
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
class  
input(inputName) {  
  return inputName ? locate(inputName) : ''  
}
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
class InputSelectHelper extends BaseHelper {  
  input(inputName) {  
    return inputName ? locate(inputName) : ''  
  }  
  ...  
}
```

```
...
```

```
locate"]')
```




Развитие. Первые шаги создания Component Object



ТИНЬКОФФ

Проблемы

- Читаемость тестов
 - Helpers
 - XPath и CSS
 - Разная верстка
 - Дублирование кода
-



ТИНЬКОФФ

Проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Решение

- Component Object-подход



ТИНЬКОФФ

Требования

- Единый способ построения локаторов



ТИНЬКОФФ

Требования

- Единый способ построения локаторов
- Единый интерфейс методов



ТИНЬКОФФ

Что сделали?

- Разработали Component Object-подход
- Добавили data-qa-атрибуты
- Реализовали Component Objects на JS

Page Object

Т ТИНЬКОФФ Частным лицам Бизнесу Премиум Еще Личный кабинет

Кредитные карты

Кредитные карты — это платежные карты, позволяющие оплачивать покупки за счет средств банка. Расплачивайтесь кредиткой и получайте кэшбэк бонусами за любые покупки

[Все карты](#) [Премиальные](#) [Путешественнику](#) [Автомобилисту](#) [Геймеру](#) [Покупки](#)


Выберите карту в лимитированном дизайне

Кредитная карта «Тинькофф Платинум»

Наша лучшая кредитная карта в лимитированном дизайне. Одобрим за 2 минуты, доставим бесплатно

До 1 000 000 Р Кредитный лимит	До 12 месяцев Беспроцентная рассрочка	590 Р в год Стоимость обслуживания
--	---	--


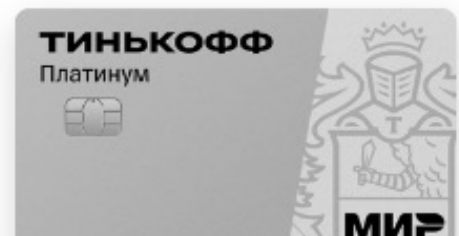
[Оформить карту](#) [Подробнее](#)



Бесплатно доставим карту уже сегодня

- 1** Заполните онлайн-заявку — вам не нужно посещать банк
- 2** Узнайте решение банка сразу после заполнения заявки
- 3** Банк бесплатно доставит карту, куда удобно. Прямо в руки

Настройте свою карту

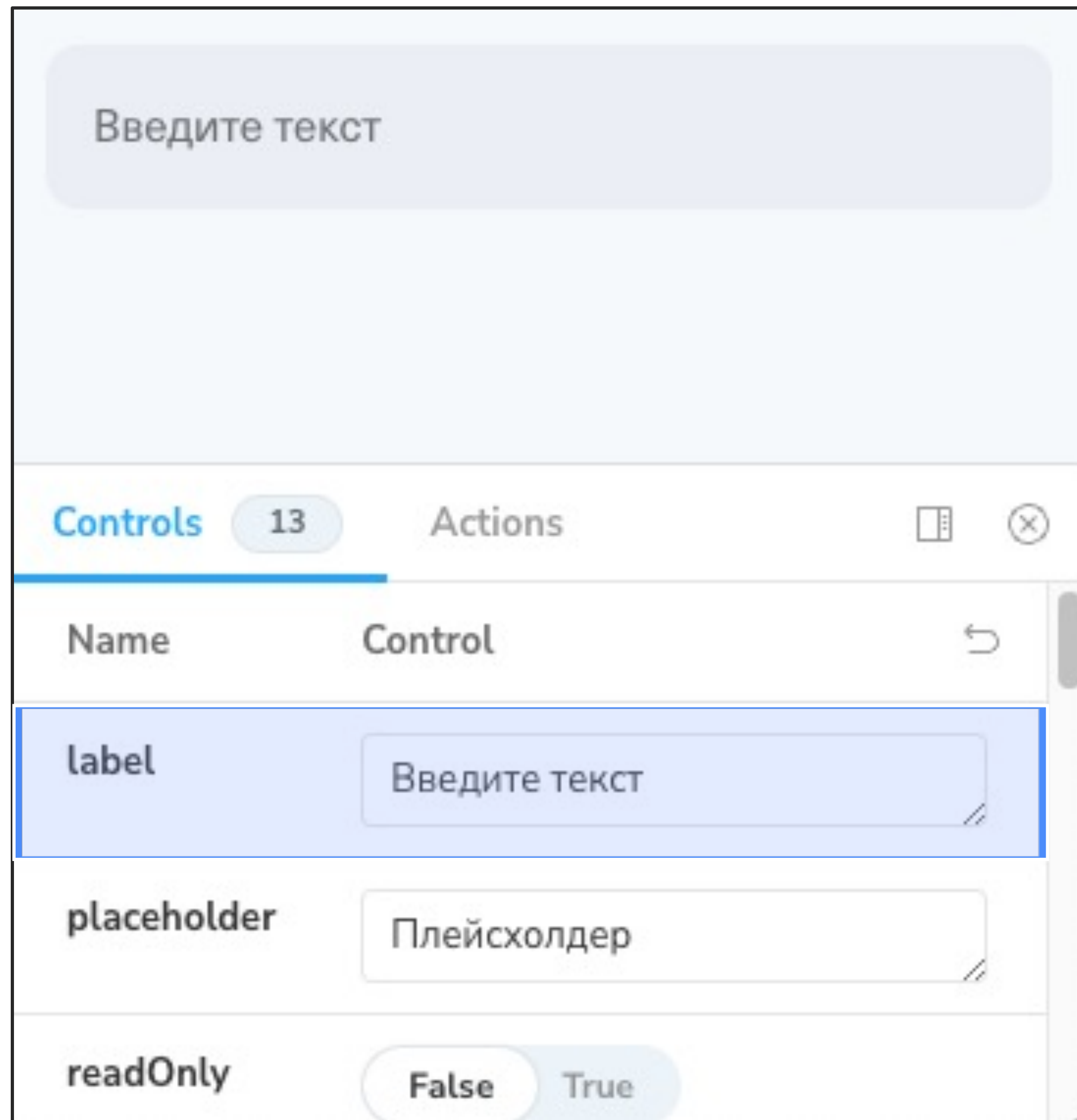


Вы выбрали карту с лимитом **70 000 Р**

Вероятность одобрения **5%**

+50% за 1-й шаг

Компонент конструктора форм



Компонент конструктора форм

The image shows a screenshot of a form builder component editor. At the top, there is a preview of a form with a text input field containing the placeholder text "Введите текст". Below the preview is a toolbar with two tabs: "Controls" (selected) and "Actions". The "Controls" tab shows a list of 13 controls. The list has two columns: "Name" and "Control". The first row shows "label" with a preview of a text input field containing "Введите текст". The second row shows "placeholder" with a preview of a text input field containing "Плейсхолдер". At the bottom of the interface, there is a "readOnly" property with a toggle switch set to "False".

Name	Control
label	<input type="text" value="Введите текст"/>
placeholder	<input type="text" value="Плейсхолдер"/>

readOnly False True

Компонент конструктора форм

The image shows a screenshot of a form builder component editor. At the top, there is a light blue rounded rectangle containing the text "Введите текст". Below this is a control palette with two tabs: "Controls" (selected) and "Actions". The "Controls" tab shows a list of 13 items. The first three items are:

Name	Control
label	<input type="text" value="Введите текст"/>
placeholder	<input type="text" value="Плейсхолдер"/>
readOnly	<input type="checkbox" value="False"/> <input checked="" type="checkbox" value="True"/>

Методы компонента

```
input.withLabel()
```

Методы компонента

```
input.withLabel()
```

```
input.withPlaceholder()
```

Методы компонента

```
input.withLabel()
```

```
input.withPlaceholder()
```

```
input.withReadOnly()
```


Методы компонента

```
await I.waitForElement(input.withLabel())
```

Методы компонента

```
await I.waitForElement(input.withLabel().withPlaceholder().withReadOnly())
```

Компонент конструктора форм

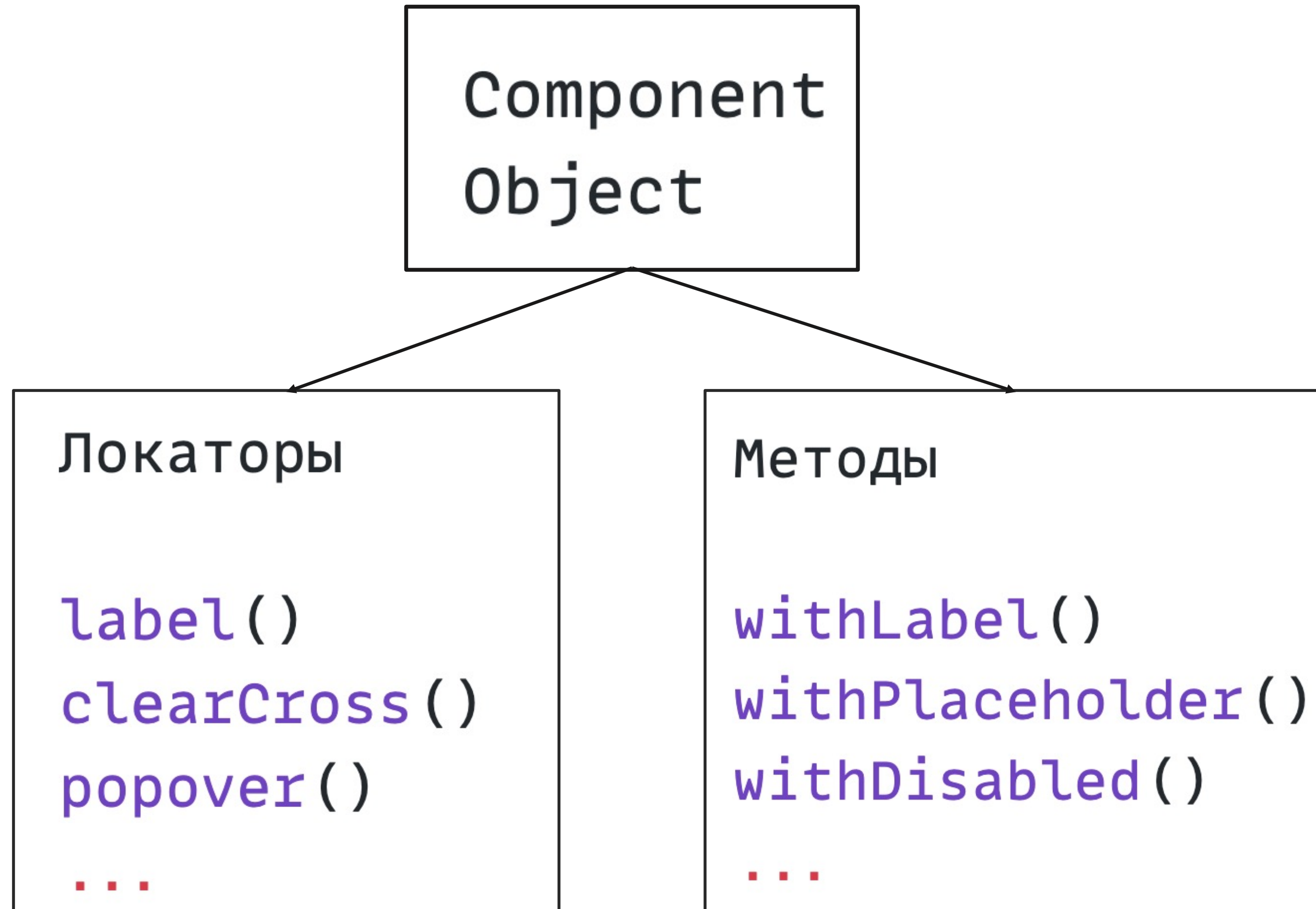
The image shows a component editor interface. At the top, a light blue rounded rectangle contains a text input field with the placeholder text "Введите текст". Below this is a control panel with two tabs: "Controls" (active, showing 13 items) and "Actions". The "Controls" tab is expanded to show a list of controls:

Name	Control
label	<input type="text" value="Введите текст"/>
placeholder	<input type="text" value="Плейсхолдер"/>
readOnly	<input type="checkbox"/> False <input checked="" type="checkbox"/> True

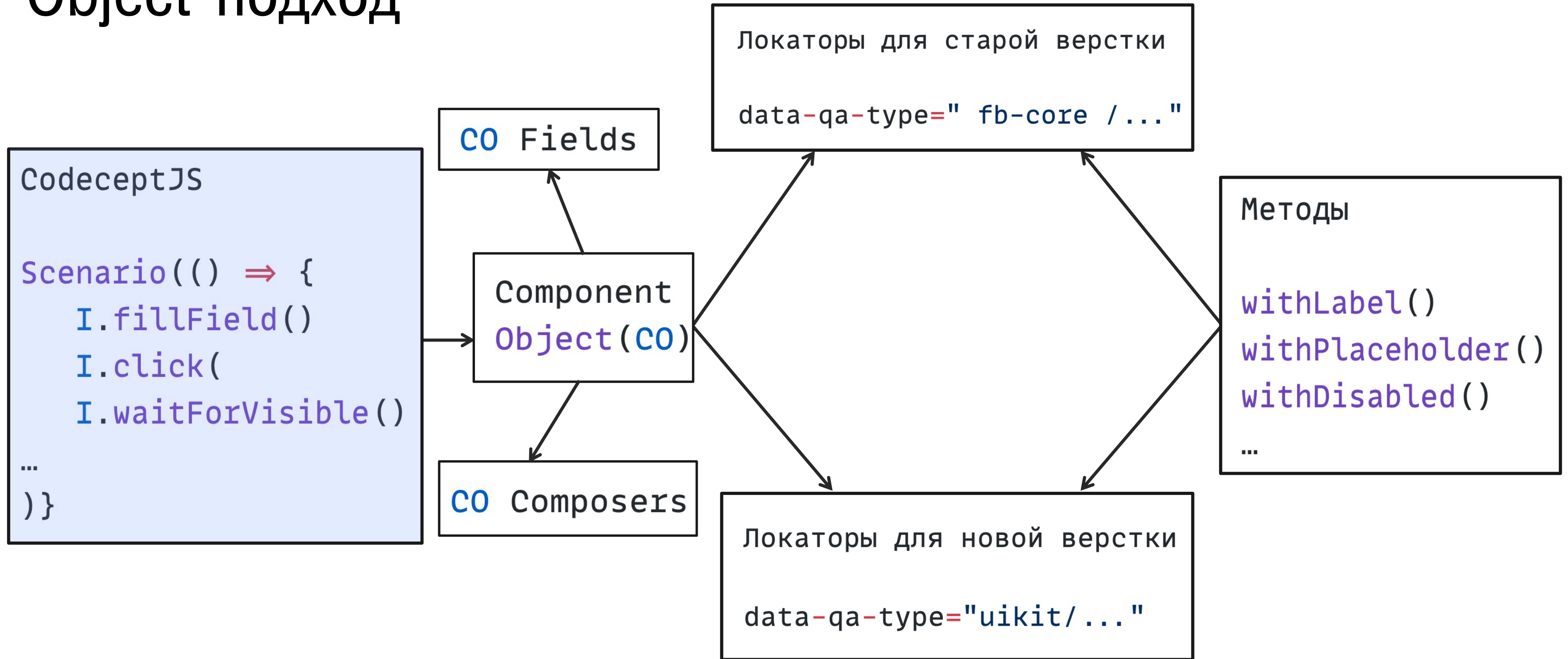
Локаторы компонента

```
await I.waitForElement(input.label())
```

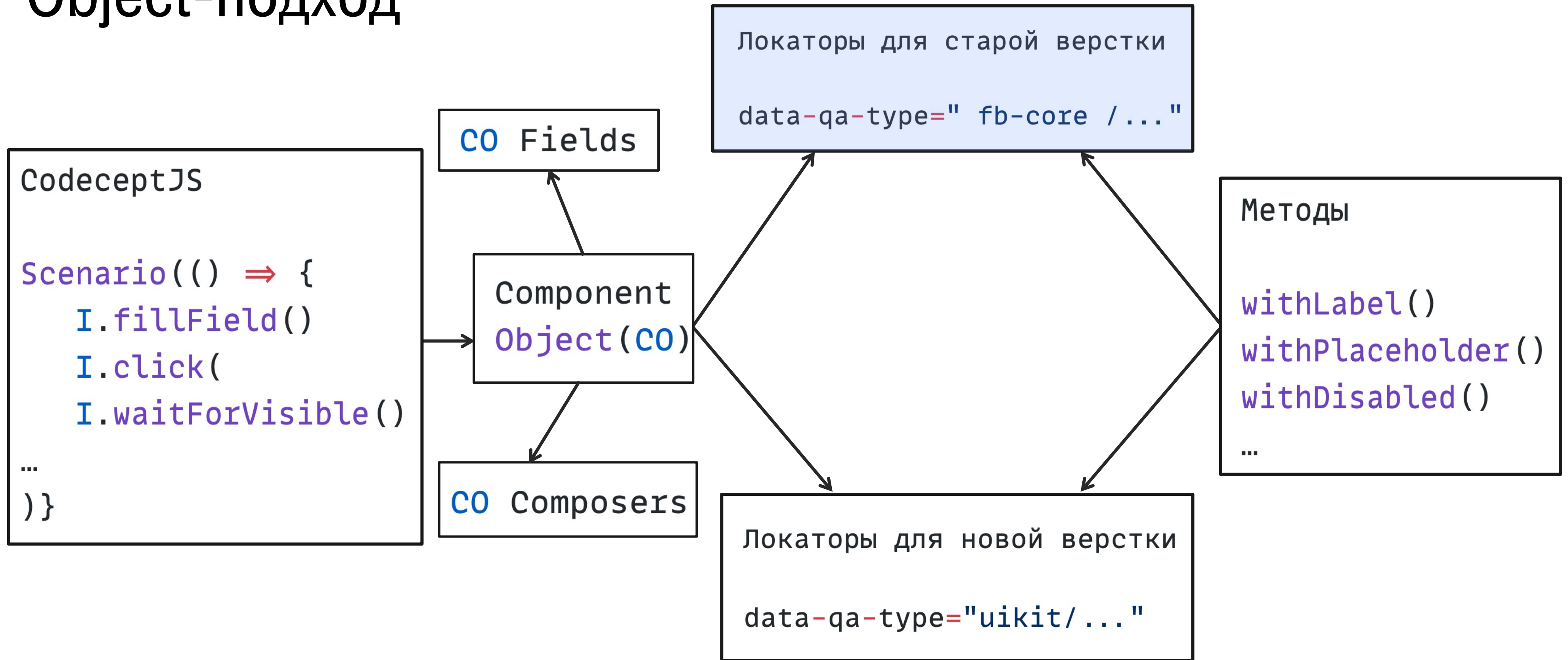
Component Object



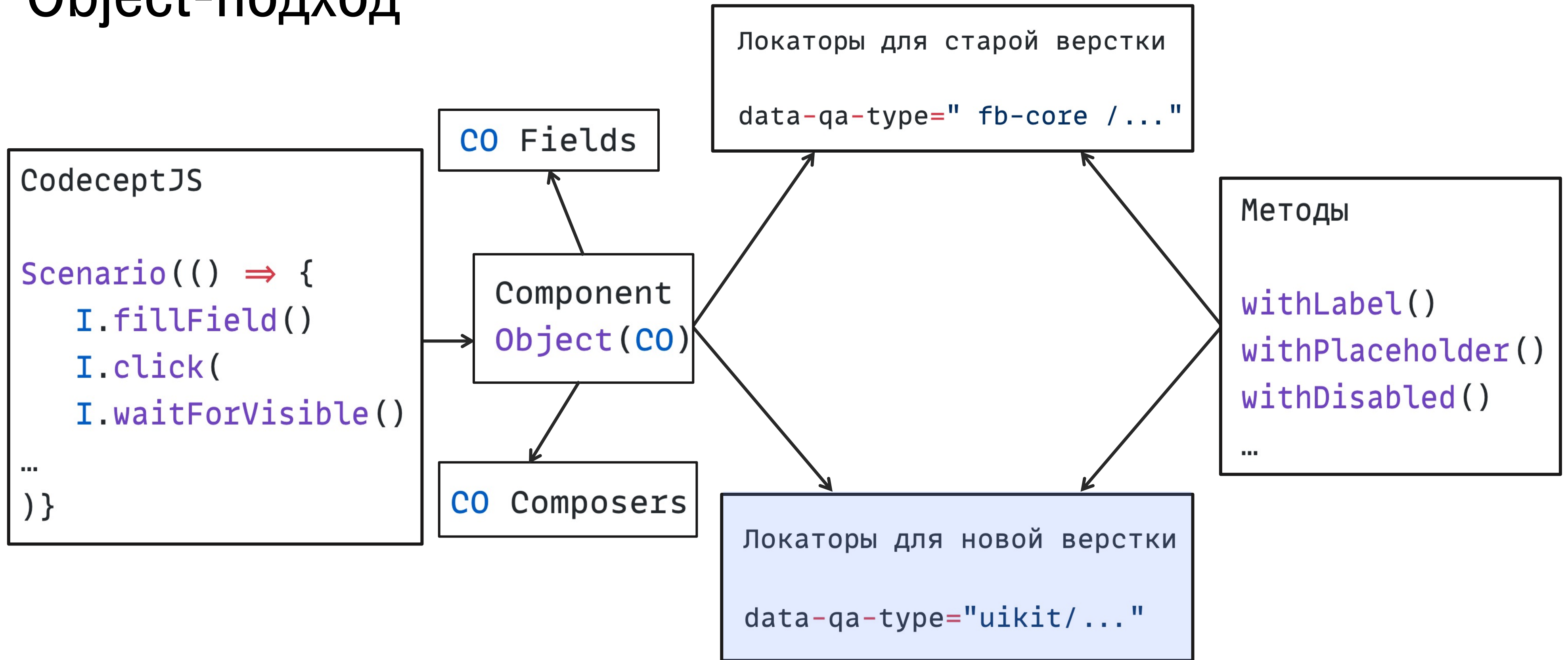
Component Object-подход



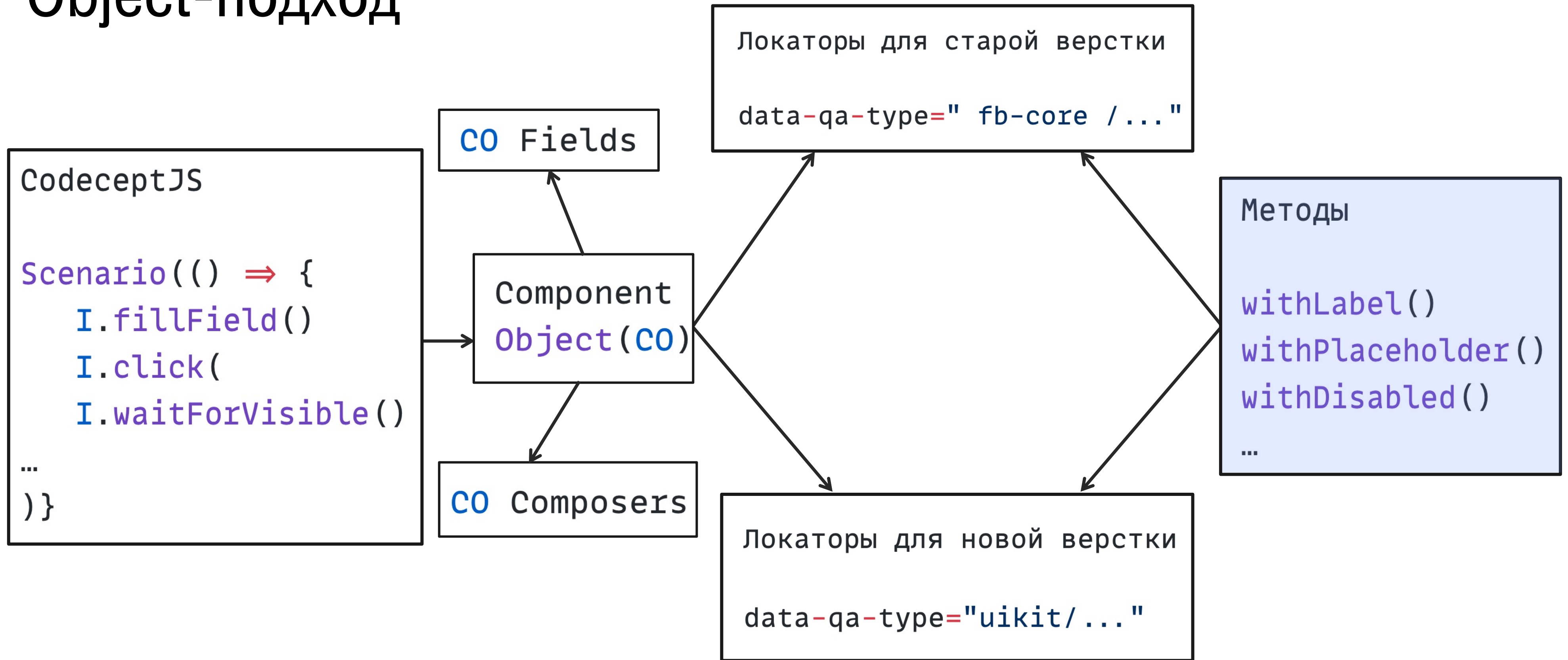
Component Object-подход



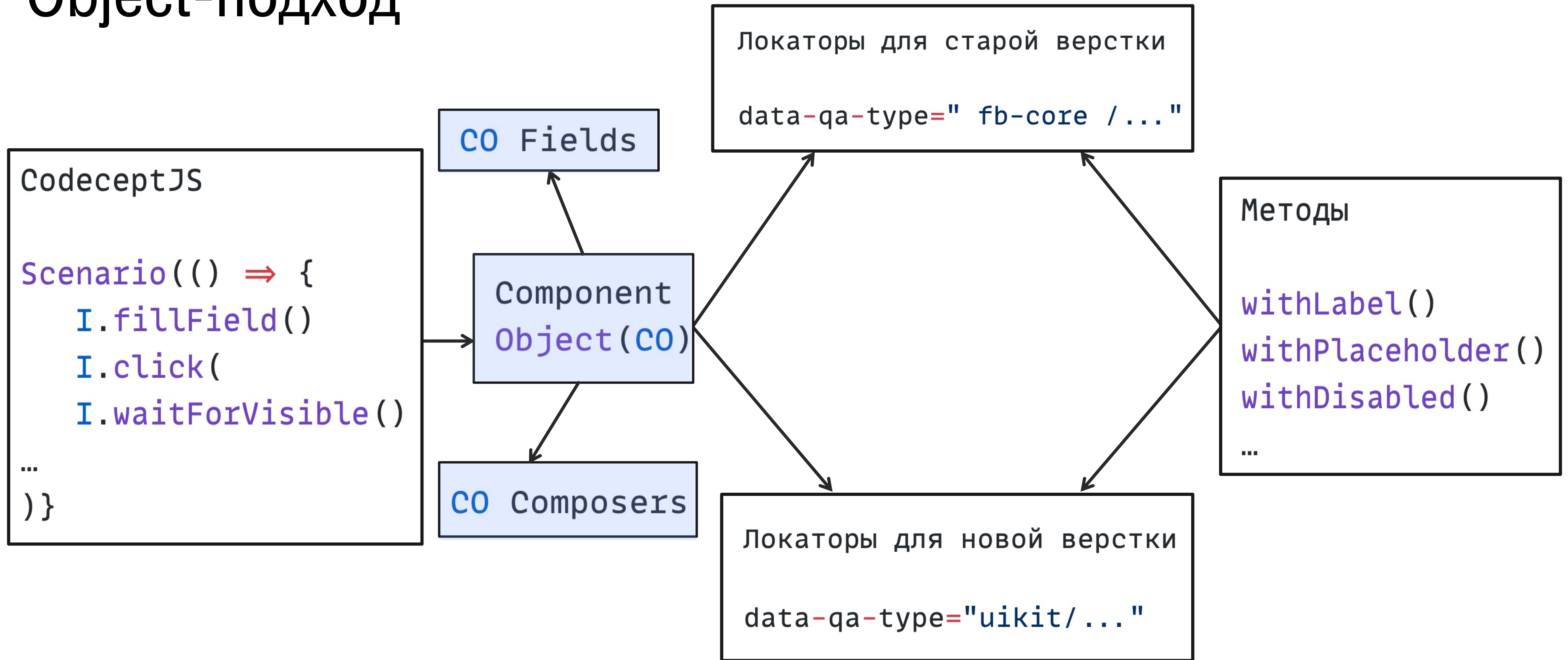
Component Object-подход



Component Object-подход



Component Object-подход





ТИНЬКОФФ

Что сделали?

- Разработали Component Object-подход
- Добавили data-qa-атрибуты
- Реализовали Component Objects на JS

Атрибуты

```
▼ <div class="node_modules-@platform-ui-Input-module__root"  
  data-qa-type="uikit/input">
```

Атрибуты

▼ `<div class="node_modules-@platform-ui-Input-module__root" data-qa-type="uikit/input">`

▼ `<div class="fb-fields-Input-Input__input" data-qa-type="fb-core/input">`



ТИНЬКОФФ

Что сделали?

- Добавили data-qa-атрибуты
- Разработали Component Object-подход
- Реализовали Component Objects на JS

Применение Component Object формами

```
import { FbUikitFixtures } from '@tinkoff-fb/fields-uikit-co'

export interface FormUikit {
  input: FbUikitFixtures['input']
  multiSelect: FbUikitFixtures['multiSelect']
  ...
}

...

async checkCashbackValue(...options: string[]) {
  const { browserInterface, multiSelect } = this.args

  await test.step(`Проверяем значение инпута кэшбэка. Ожидаемое значение: ${options}`, async () => {
    await browserInterface.waitForElement(multiSelect('За что вы хотели бы получать кэшбэк').withValue(options.join(', ')))
  })
}
```


Применение Component Object формами

```
test('Кредитный лимит больше лимита дефолтной формы', async ({ I, story, input, inputLimitSlider }) => {  
  await story.amOnStory(stories('uikitCreditAllAirlinesBE'))  
  
  await I.waitForVisible(inputFio('Фамилия, имя и отчество'))  
  await I.clearTitle(inputLimitSlider.title)  
  await I.fillField(inputLimitSlider.title, '1200000', true)  
  await I.waitForVisible(inputLimitSlider.titleWithStrictText('1\u00a0200\u00a0000\u00a0\u20a0'))  
  await I.waitForVisible(input.withError('Подтвердите мобильный телефон'))  
})  
})
```




ТИНЬКОФФ

Проблемы

- **Читаемость тестов**
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Решение

- **Название компонента, методы и локаторы Component Object в тесте**
- Не используем кастомные хелперы в конструкторе форм
- Меньше кастомных хелперов у форм
- Нет XPath и CSS
- Единый способ построения локаторов

Читаемость тестов

```
Feature('Core/Input/Functional')

Scenario('Поле необязательно и значение не введено', async (I) => {
  await I.amOnPage(stories('inputUnReq'))

  await I.fillField('Nickname', '')
  await I.pressKey('Tab')
  await I.dontSeeElement('Поле обязательно для заполнения')
})
```

Читаемость тестов

```
Feature('Core/Input/Functional')

Scenario('Поле необязательно и значение не введено', async (I) => {
  await I.amOnPage(stories('inputUnReq'))

  await I.fillField('Nickname', '')
  await I.pressKey('Tab')
  await I.dontSeeElement('Поле обязательно для заполнения')
})
```

```
Feature('Core/Input/Functional')

Scenario('Поле необязательно и значение не введено', async ({ I, input }) => {
  await story.amOnStory(stories('inputUnReq'))

  await I.fillField(input('Nickname'), '')
  await I.pressKey('Tab')
  await I.dontSeeElement(input('Nickname').withError('Поле обязательно для заполнения'))
})
```



ТИНЬКОФФ

Проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Решение

- Название компонента, методы и локаторы Component Object в тесте
- Не используем кастомные хелперы в конструкторе форм
- Меньше кастомных хелперов у форм
- Нет XPath и CSS
- Единый способ построения локаторов



ТИНЬКОФФ

Проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Решение

- Название компонента, методы и локаторы Component Object в тесте
- Не используем кастомные хелперы в конструкторе форм
- Меньше кастомных хелперов у форм
- Нет XPath и CSS
- Единый способ построения локаторов



ТИНЬКОФФ

Проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Решение

- Название компонента, методы и локаторы Component Object в тесте
- Не используем кастомные хелперы в конструкторе форм
- Меньше кастомных хелперов у форм
- Нет XPath и CSS
- Единый способ построения локаторов



ТИНЬКОФФ

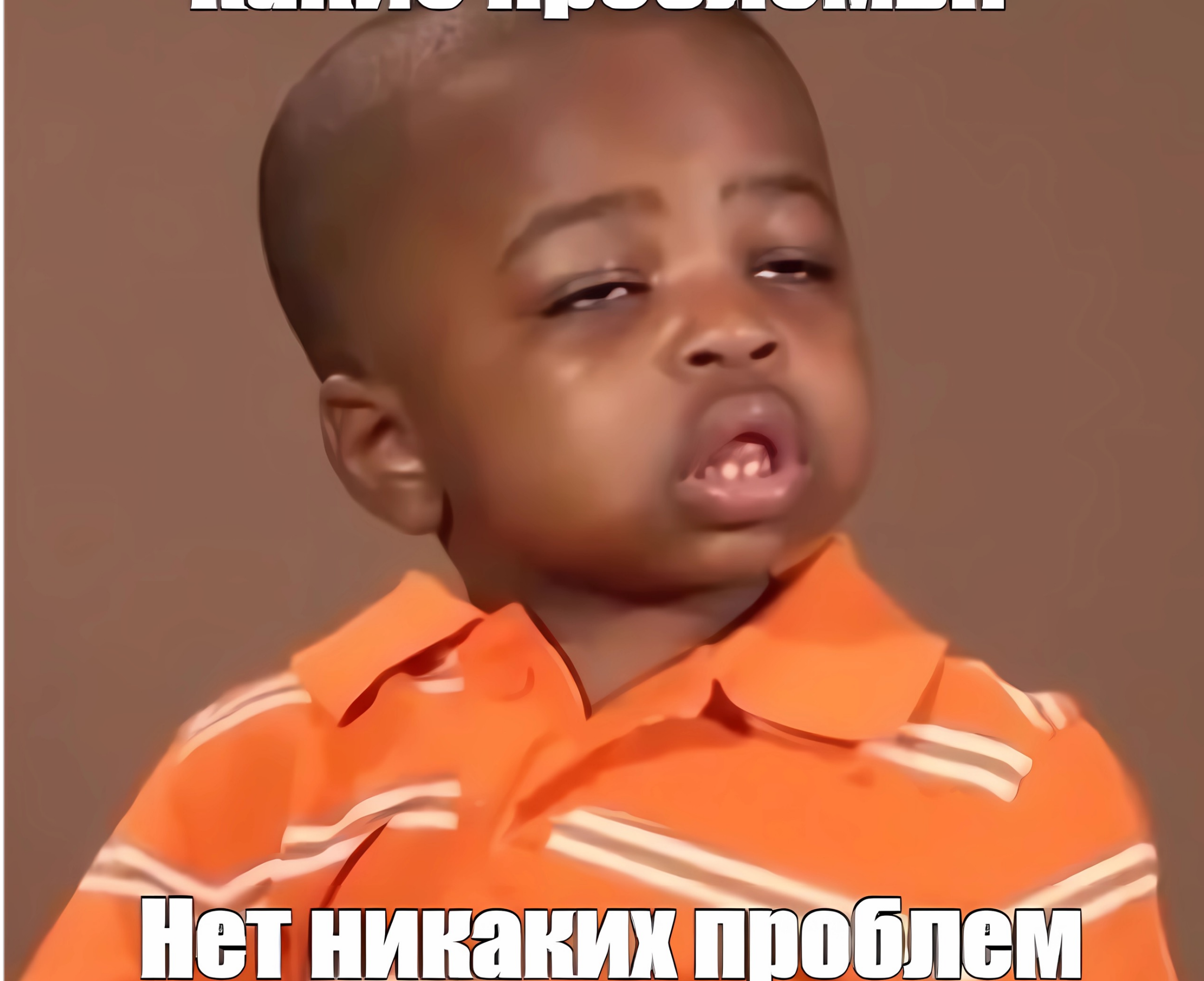
Проблемы

- Читаемость тестов
- Helpers
- XPath и CSS
- Разная верстка
- Дублирование кода

Решение

- Название компонента, методы и локаторы Component Object в тесте
- Не используем кастомные хелперы в конструкторе форм
- Меньше кастомных хелперов у форм
- Нет XPath и CSS
- Единый способ построения локаторов

Какие проблемы?



Нет никаких проблем



ТИНЬКОФФ

Оставшиеся проблемы

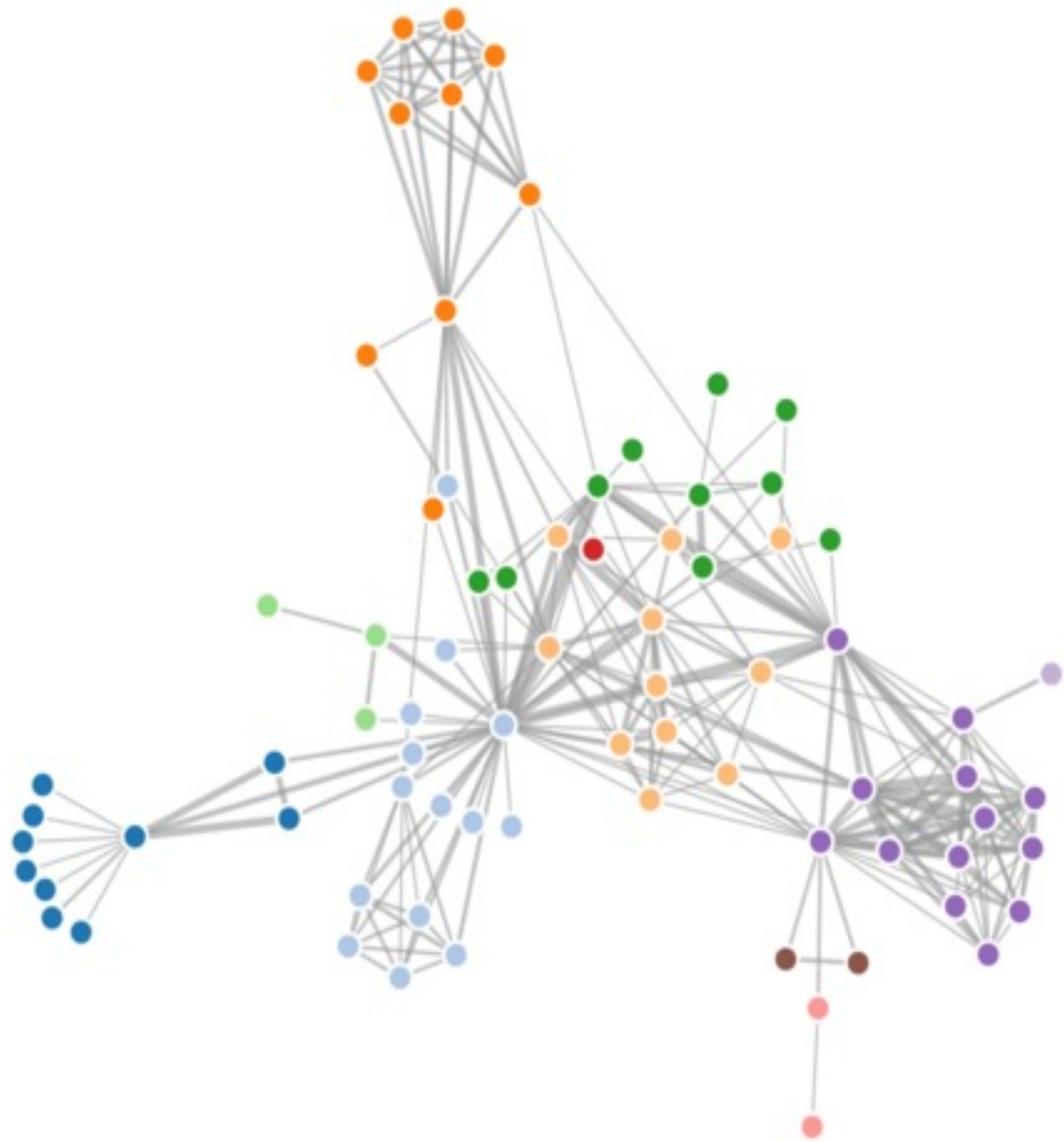
- Дублирование кода тестов
- Подключение лишних зависимостей
- Подключение Component Object к формам на TS



ТИНЬКОФФ

Оставшиеся проблемы

- Дублирование кода тестов
- Подключение лишних зависимостей
- Подключение Component Object к формам на TS





ТИНЬКОФФ

Оставшиеся проблемы

- Дублирование кода тестов
- Подключение лишних зависимостей
- Подключение Component Object к формам на TS



**Оптимизация.
Улучшение тестирования
конструктора форм**



ТИНЬКОФФ

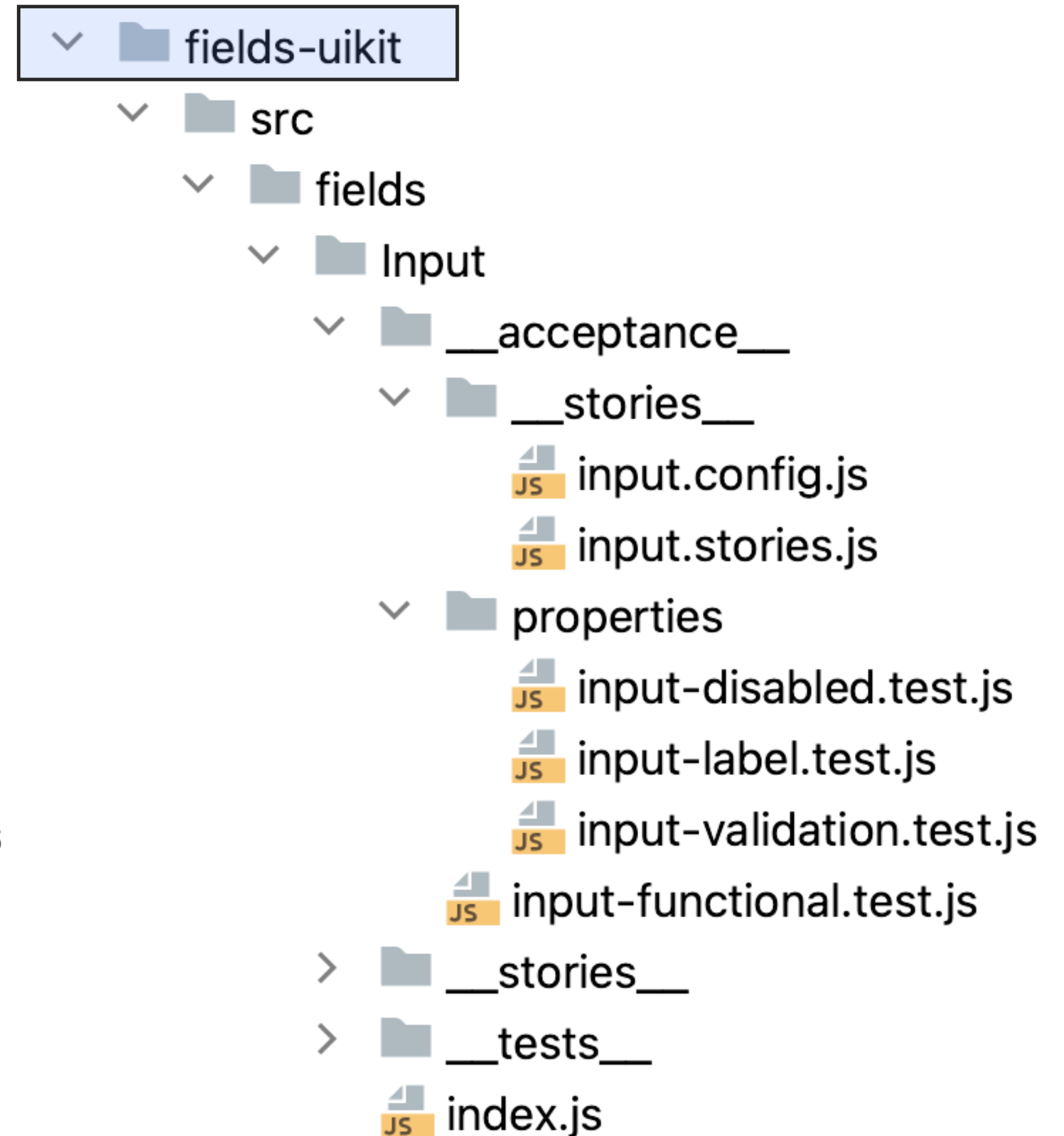
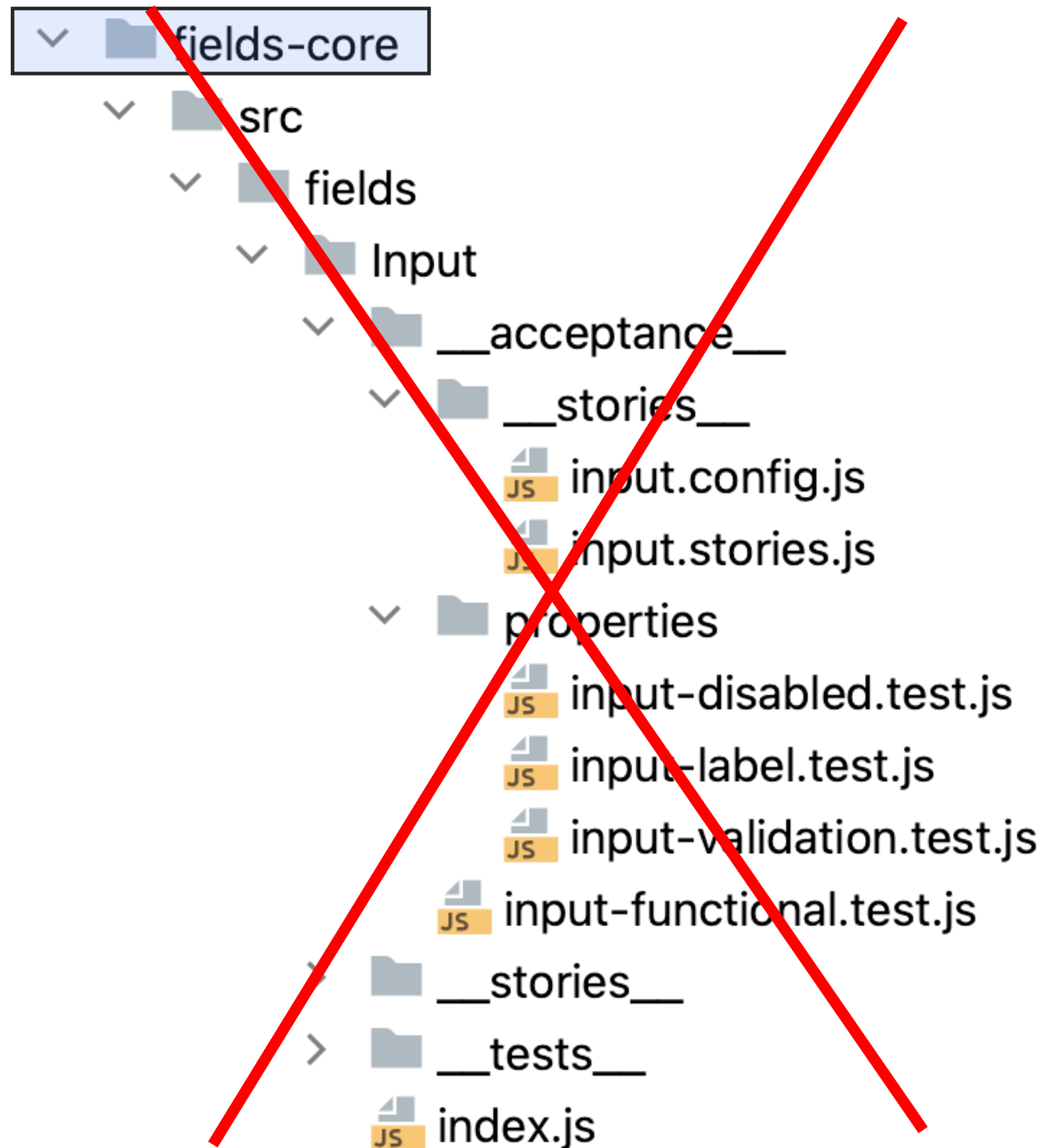
Проблемы

- Дублирование кода тестов
- Подключение лишних зависимостей
- Подключение Component Object к формам на TS

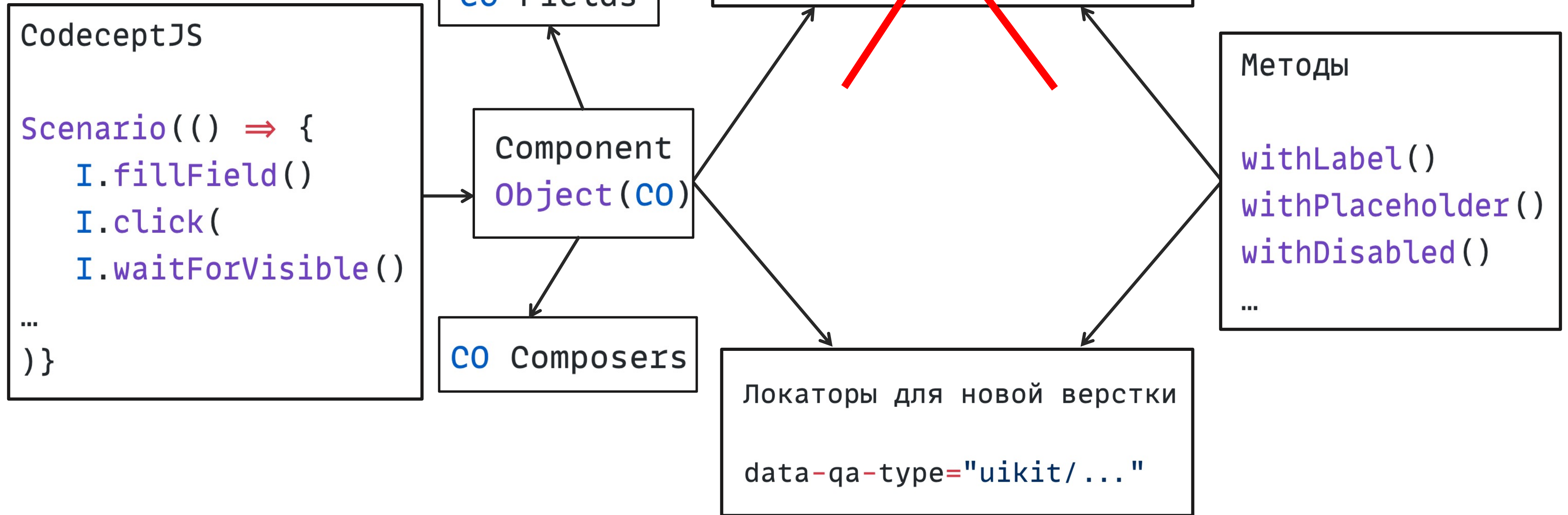
Решение

- Удалили пакет со старыми компонентами
- Мигрировали на TS
- Отдельный пакет для Component Objects

Удалили пакет



Component Object-подход





ТИНЬКОФФ

Проблемы

- Дублирование кода тестов
- Подключение лишних зависимостей
- Подключение Component Object к формам на TS

Решение

- Удалили пакет со старыми компонентами
- Мигрировали на TS
- Отдельный пакет для Component Objects

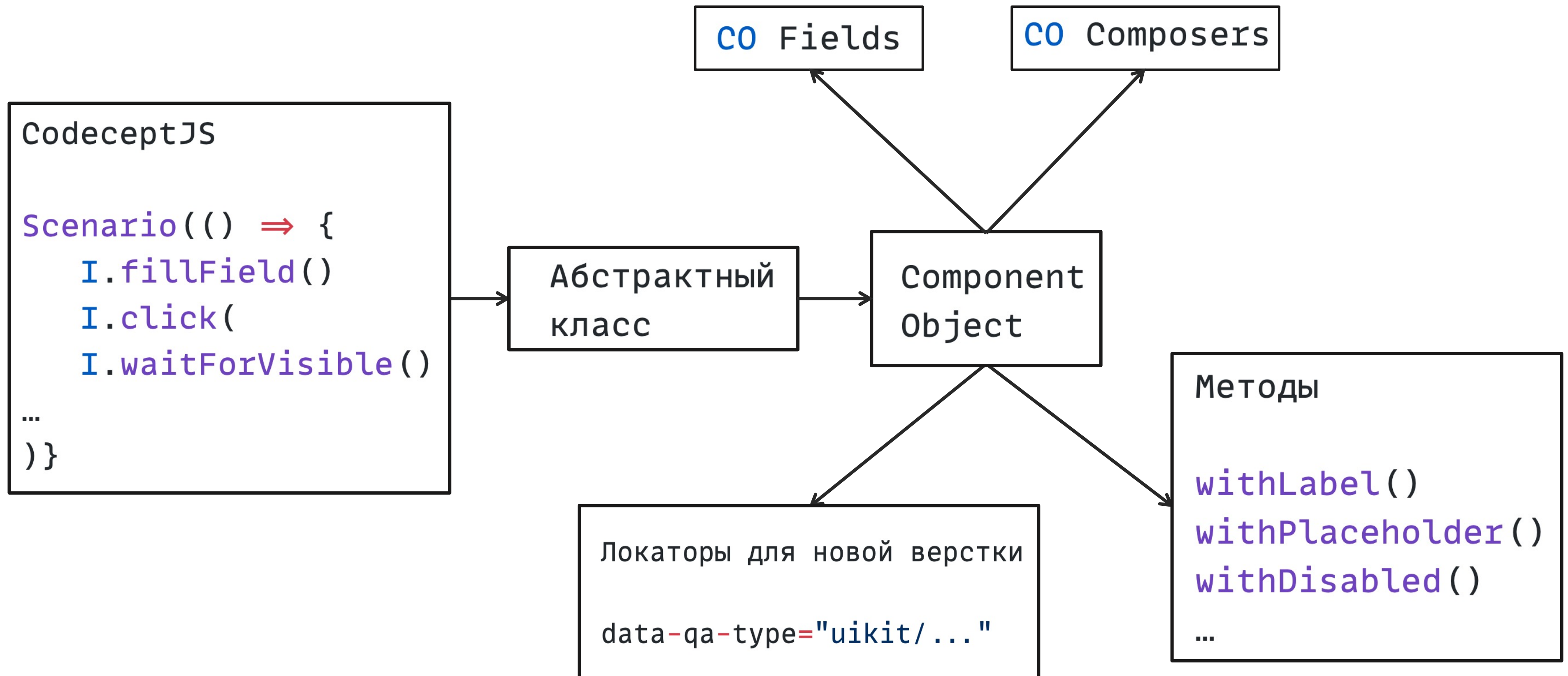


ТИНЬКОФФ

Миграция на TS

- Сделали абстрактный класс
- Component Object на TS
- Подсказки в тестах на TS

Component Object-подход



Абстрактный класс

```
export interface IInputLocatorsFindOptions {  
    labelText: string  
    placeholderText: string  
    disabledFlag: boolean  
    ...
```


Абстрактный класс

```
export interface InputLocatorsInterface {  
  input(): Locator  
  withDisabled(disabledFlag: boolean): this  
  withLabel(labelText?: string): this  
  withPlaceholder(placeholderText?: string): this  
  ...
```

Абстрактный класс

```
export abstract class InputLocatorsAbstractClass<InputFindOptions>
  implements InputLocatorsInterface {
  /**
   * Метод для проверки того, что поле неактивно
   * @public
   *
   * @param {Boolean} disabledFlag - Флаг проверки того, что поле неактивно, по умолчанию true
   */
  public withDisabled(disabledFlag: boolean = true) {
    return this.copyLocatorsInstance({ disabledFlag } as Partial<InputFindOptions>)
  }
  ...
}
```



ТИНЬКОФФ

Миграция на TS

- Сделали абстрактный класс
- Component Object на TS
- Подсказки в тестах на TS

Component Object на TS

```
/**
 * @internal
 */
export class Input<InputFindOptions> extends
Fields.InputLocatorsAbstractClass<IInputLocatorConstructorParams,
InputFindOptions> {
    protected inputLabelLocator: Locator
    ...
}
...
this.inputLabelLocator = locate({
    namespace: 'uikit/',
    componentName: 'input.inputBox',
    attrValue: 'label',
})
...

```

`input.withLabel(labelText)`

`this.inputLabelLocator = locate({
 namespace: 'uikit/',
 componentName: 'input.inputBox',
 attrValue: 'label',
})`

Component Object на TS

```
this.inputLabelLocator = locate({  
  namespace: 'uikit/',  
  componentName: 'input.inputBox',  
  attrValue: 'label',  
})  
  
...
```



```
./ /*[@data-qa-type="uikit/input.inputBox.label"]
```

Component Object на TS

```
/**
 * Локатор поля инпута
 * @public
 */
public input = (): Locator => {
  const {
    labelText,
    ...
  } = this.findOptions

  let locator = this.inputLabelLocator

  if (typeof labelText === 'string') {
    locator = locator.withText(labelText)
  }

  return locator
}
```

```
input.withLabel(labelText)
```



ТИНЬКОФФ

Миграция на TS

- Сделали абстрактный класс
- Component Object на TS
- Подсказки в тестах на TS

Подсказки в тестах

```
test( title: 'Поле необязательно и значение введено', testFunction: async ({ browserInterface, story : InstanceType<any>, input : ... }) => {  
  await story.amOnStory(stories('inputUnReq'))  
  
  await browserInterface.fillField(input( defaultFilter: 'Nickname'), 'test')  
  await browserInterface.waitForVisible(input( defaultFilter: 'Nickname').withValue( value: 'test'))  
  await browserInterface.pressKey('Tab')  
  await browserInterface.waitUnfocused(input( defaultFilter: 'Nickname'))  
  await browserInterface.dontSeeElement(input( defaultFilter: 'Nickname').withError())  
  await browserInterface.seeElement(input.)  
})
```

- m withClearCross InputLocatorsAbstractClass<IInputLocatorConstructorParams, Fields...
- m withCustomIcon InputLocatorsAbstractClass<IInputLocatorConstructorParams, Fields...
- m withFocused InputLocatorsAbstractClass<IInputLocatorConstructorParams, Fields.II...
- m withLabel InputLocatorsAbstractClass<IInputLocatorConstructorParams, Fields.IInp...
- m withLoader InputLocatorsAbstractClass<IInputLocatorConstructorParams, Fields.IIn...
- m withMessage(messageContent?: string) (Input, src/input.ts) this
- m withName InputLocatorsAbstractClass<IInputLocatorConstructorParams, Fields.IInpu...

Подсказки в тестах

```
test( title: 'Поле необязательно и значение введено', testFunction: async ({ browserInterface, story : InstanceType<any> , input :... }) => {  
  await story.amOnStory(stories('inputUnReq'))  
  
  await browserInterface.fillField(input( defaultFilter: 'Nickname'), 'test')  
  await browserInterface.waitForVisible(input( defaultFilter: 'Nickname').withValue( value: 'test'))  
  await browserInterfac  
  await browserInterfac  
  await browserInterfac  
  await browserInterfac  
})
```

```
public  
public InputLocatorsAbstractClass<IInputLocatorConstructorParams, IInputLocator  
  value?: string | undefined): Input
```

Метод для проверки того, что значение выбрано

Params: value – Текстовое значение в поле, опциональный параметр

packages-tests/.../input-abstract.ts



ТИНЬКОФФ

Проблемы

- Дублирование кода тестов
- Подключение лишних зависимостей
- Подключение Component Object к формам на TS

Решение

- Удалили пакет со старыми компонентами
- Мигрировали на TS
- Отдельный пакет для Component Objects

Отдельный пакет

- packages-tests
 - address-dadata-manual-co
 - appointment-meeteor-co
 - co-abstract-classes
 - composer-accordion-co
 - cross-sale-fields-co
 - field-input-date-uikit-co
 - field-map-point-selector-co
 - fields-uikit-co
 - playwright-fixtures
 - popup-uikit-co
 - progressbar-uikit-co
 - pw-codecept-base-helper
 - screens-uikit-co
 - sms-confirm-co
 - stepper-uikit-co
 - test-base

Отдельный пакет

- ▼ packages-tests
 - > address-dadata-manual-co
 - > appointment-meeteor-co
 - > co-abstract-classes
 - > composer-accordion-co
 - > cross-sale-fields-co
 - > field-input-date-uikit-co
 - > field-map-point-selector-co
 - > fields-uikit-co
 - > playwright-fixtures
 - > popup-uikit-co
 - > progressbar-uikit-co
 - > pw-codecept-base-helper
 - > screens-uikit-co
 - > sms-confirm-co
 - > stepper-uikit-co
 - > test-base



ТИНЬКОФФ

ИТОГИ

- Единый подход к тестированию для всех команд
Платформы привлечения



ТИНЬКОФФ

ИТОГИ

- Единый подход к тестированию для всех команд
Платформы привлечения
- Независимость кода тестов от изменений верстки



ТИНЬКОФФ

ИТОГИ

- Единый подход к тестированию для всех команд
Платформы привлечения
- Независимость кода тестов от изменений верстки
- Простое построение и расширение локаторов



ТИНЬКОФФ

ИТОГИ

- Единый подход к тестированию для всех команд
Платформы привлечения
- Независимость кода тестов от изменений верстки
- Простое построение и расширение локаторов
- Единый интерфейс методов

Стало ли лучше?



**Тестирование микрофронтендов
можно сделать удобным**

СПАСИБО!



Больше про
FormBuilder



Скачать
презентацию

