Лабораторная № 6 Navigation. Method Chanel

1. Добавить к предыдущей лабораторной работе логику переходов между экранами используя Navigator, также создать переходы с передачей данных между экранами.

2. Добавить на один из экранов widget PageViewer продемонстрировать его работу

3. Создать Platform Method Channel в зависимости от вашего физического устройства или эмулятора (Android или iOS), метод возвращать информацию Flutter (обычно недоступную как пример: уровень заряда батареи, открытие набора телефонного номера, проверка разрешений и т.д.) Выполняется по вариантам.

|  |  |
| --- | --- |
| Вариант | Задание |
| 1 | Получить текущий ориентацию устройства |
| 2 | Получить тип сети к которую использует устройство (Wifi/Мобильный интернет) |
| 3 | Получить информацию заражается ли устройство |
| 4 | Получить название производителя устройства |
| 5 | Получить ip адрес |
| 6 | Вывести показания акселерометра |
| 7 | Получить текущую геопозицию. |
| 8 | Получить текущую локализацию устройства |
| 9 | Получить часовой пояс устройства |
| 10 | Получить статус Bluetooth |
| 11 | Получить SSID (имя) подключенной сети wifi |
| 12 | Получить информацию включен ли GPS |
| 13 | Получение текущей загрузки процессора |
| 14 | Получить информацию с датчика освещённости |
| 15 | Получить данные с гироскопа |

3. Создать Platform Method Channel передающий данные для запуска встроенного приложения. Все данные передаются из Flutter

|  |  |
| --- | --- |
| Вариант | Задание |
| 1, 7, 14 | Запуск браузера с переданным URL |
| 2, 8, 15 | Запуск приложения карты с переданными координатами |
| 3, 6, 12 | Запуск отправки сообщения с заполненным телом сообщения |
| 4, 10, 9 | Запуск приложение почты с заполнением темы сообщения |
| 5, 11, 13 | Запуск приложения для установки будильника с заданным временем |

4 На основе документации также добавить два Platform Method Channel для IOS и Android для получения заряда батареи. Показать разницу.

Вопросы по лабораторной работе:

1. Компонент Navigator, какие основные методы взаимодействия?

2. Виды route?

3. Что такое Platform Method Channel?

4. Как производится валидация вызываемого method channel?

**1. Компонент Navigator: основные методы взаимодействия**

В Flutter Navigator — это компонент, который управляет навигацией между экранами (страницами) приложения. Основные методы Navigator для взаимодействия:

* **push**: добавляет новый маршрут на верх стека маршрутов, то есть открывает новый экран. Например:

dart

Копировать код

Navigator.push(context, MaterialPageRoute(builder: (context) => NewScreen()));

* **pop**: удаляет текущий маршрут с верхушки стека, возвращаясь на предыдущий экран. Например:

dart

Копировать код

Navigator.pop(context);

* **pushReplacement**: заменяет текущий маршрут новым, без сохранения текущего экрана в стеке. Например:

dart

Копировать код

Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) => NewScreen()));

* **pushAndRemoveUntil**: добавляет новый маршрут и удаляет все предыдущие маршруты до указанного условия. Полезно, когда нужно очистить историю навигации. Например:

dart

Копировать код

Navigator.pushAndRemoveUntil(context, MaterialPageRoute(builder: (context) => NewScreen()), (route) => false);

**2. Виды Route**

В Flutter Route — это абстракция маршрута, представляющая экран или страницу в приложении. Основные виды маршрутов:

* **MaterialPageRoute**: использует стандартные переходы для платформы Android, обычно это слайд-анимация справа налево. Пример:

dart

Копировать код

Navigator.push(context, MaterialPageRoute(builder: (context) => NewScreen()));

* **CupertinoPageRoute**: создает переход в стиле iOS, где переход экрана происходит с анимацией слева направо, типичной для iOS.
* **PageRouteBuilder**: позволяет создавать кастомные анимации для переходов, определяя анимацию входа и выхода.
* **NamedRoute**: маршруты, определенные по именам, полезны для создания централизованной навигации по экрану. Определяются в корневом MaterialApp или CupertinoApp.
* **ModalRoute**: используется для отображения маршрутов, которые временно перекрывают другой контент, например, модальные окна.

**3. Что такое Platform Method Channel?**

Platform Method Channel — это способ взаимодействия между кодом на Flutter (Dart) и нативным кодом на Android (Kotlin/Java) и iOS (Swift/Objective-C). MethodChannel обеспечивает вызовы методов с параметрами из Flutter, передавая их в платформенно-специфический код, а также возвращая результаты обратно. Это позволяет Flutter приложению использовать платформенные API, например, доступ к файловой системе, камере или другим нативным функциям.

Пример создания канала в Flutter:

dart

Копировать код

static const platform = MethodChannel('com.example.channel');

**4. Как производится валидация вызываемого method channel?**

Валидация вызываемого MethodChannel включает несколько этапов:

* **Проверка наличия метода**: на стороне нативного кода проверяется, существует ли запрашиваемый метод. Обычно это делается в setMethodCallHandler. Если метод не распознан, отправляется ошибка notImplemented, например:

kotlin

Копировать код

if (call.method == "desiredMethod") {

// Выполнить код

} else {

result.notImplemented()

}

* **Проверка и валидация аргументов**: перед обработкой метода проверяются переданные аргументы. Например, если требуется строковый аргумент, проверяется его тип и содержание. Если аргумент отсутствует или неверный, возвращается ошибка:

kotlin

Копировать код

val param = call.argument<String>("paramName")

if (param == null || param.isEmpty()) {

result.error("INVALID\_ARGUMENT", "Параметр не указан или пустой.", null)

}

* **Обработка исключений**: на стороне платформы любой возможный код ошибки отправляется обратно в Flutter с помощью result.error, чтобы приложение могло правильно обработать и отобразить ошибку:

kotlin

Копировать код

try {

// выполнение метода

result.success(data)

} catch (e: Exception) {

result.error("ERROR\_CODE", "Сообщение об ошибке", e.localizedMessage)

}

4o

**Вы сказали:**

я нажимаю на кнопку, но окно отправки почты не открывается static const platform = MethodChannel('com.example.lab6/channel');