

Александр Зубков

Практический опыт интеграции Flutter и Firebird в мобильном приложении

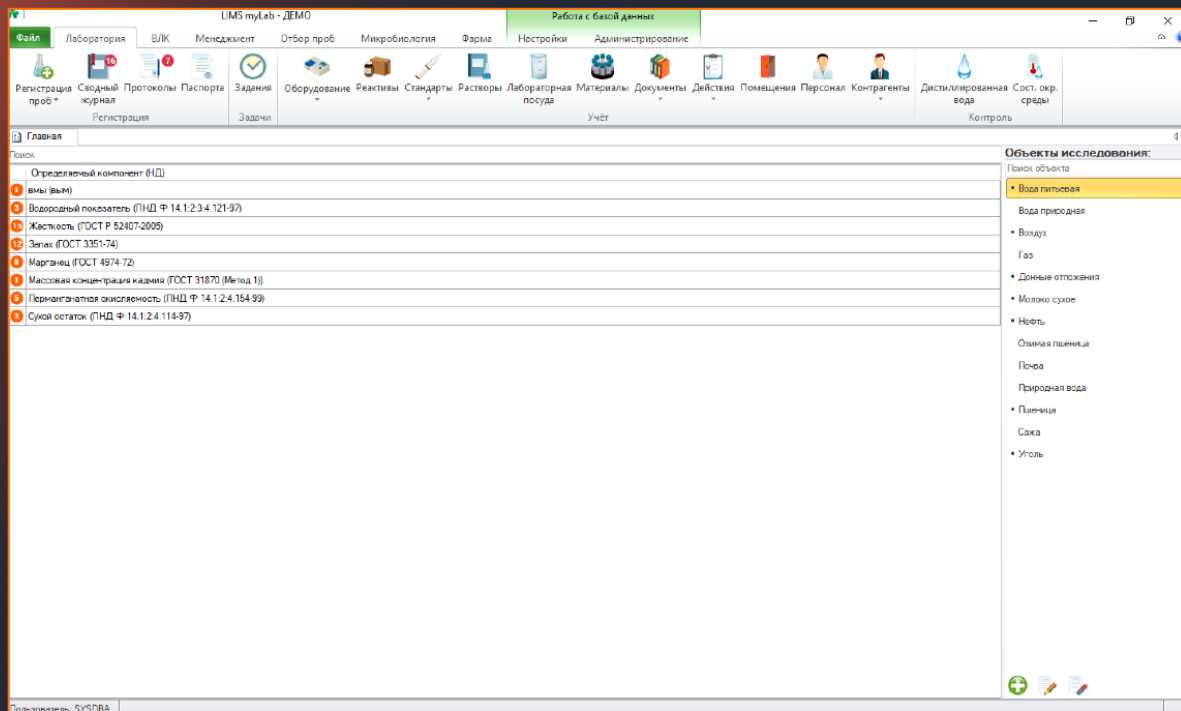
Кто мы и что делаем

ООО «ЛИКОРИС», г. Тюмень

LIMS myLab - лабораторная информационная система для управления процессами и данными

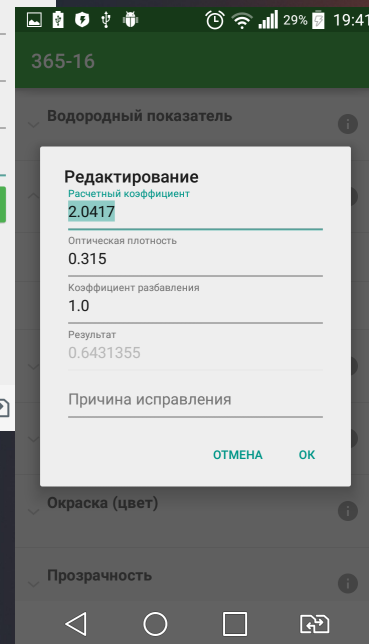
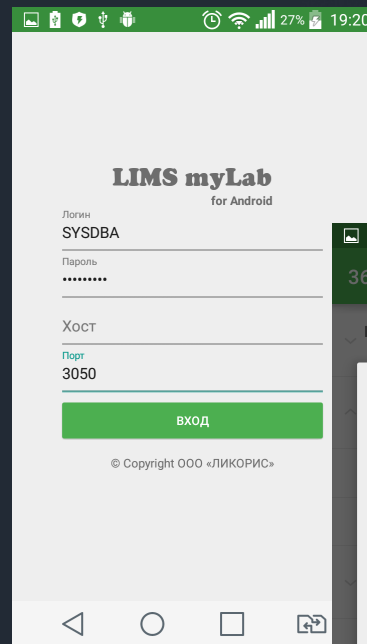
- Сбор и обработка лабораторных данных
- Управление процессами лаборатории
- Автоматизация рутинных задач

Desktop приложение



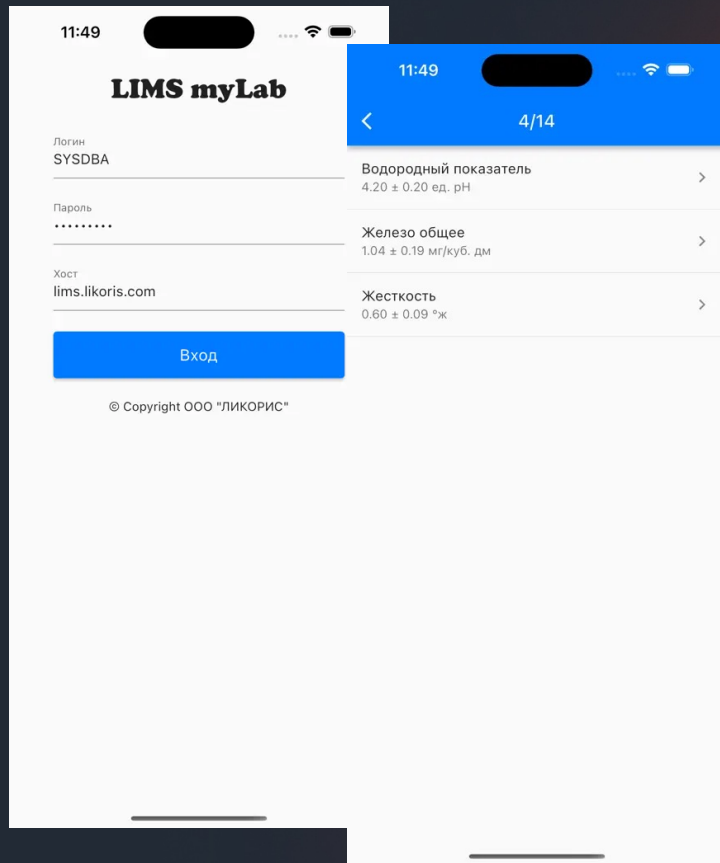
Android приложение

- Приложение написано на Java
- Поддержка только Firebird 2.5
- Прямое подключение к базе данных по Wi-Fi



iOS приложение

- Приложение написано на Swift
- Работа только через API



Выбор кроссплатформенного решения



Dart

```
GigaCode: explain | explain step by step
1  int sum(int n) {
2    int result = 0;
3
4    for (int i = 1; i <= n; i++) {
5      result += i;
6    }
7
8    return result;
9  }
10
```



<https://dartpad.dev>

Dart

2011 год

Google создаёт Dart
как замену
JavaScript

2017 год

Выход Flutter
Dart оживает

Сегодня

Постоянное
развитие и
синтаксический
сахар

Flutter. Один код – все платформы

Mobile

Desktop

Web

Windows
macOS
Linux

iOS
Android

Отдельная
компиляция

Свой движок отрисовки

Не зависит от нативных UI-компонентов, имитирует нативное поведение

Быстрая разработка

Hot reload, горячая перезагрузка кода без перезапуска приложения

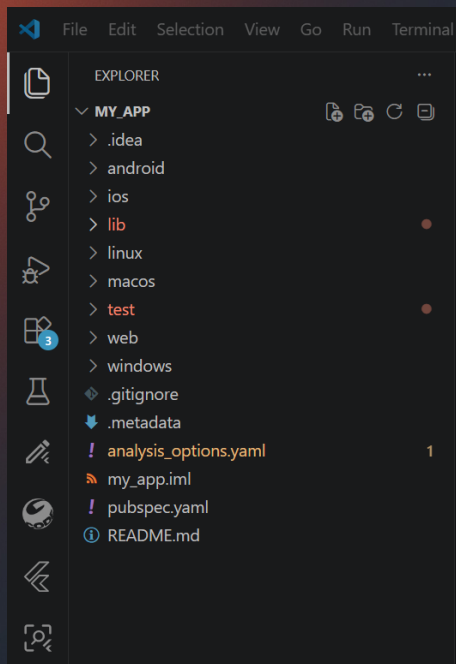
Flutter. Установка

- Добавить расширения **Flutter** для **VS Code**
- В палитре команд **VS Code** ввести **flutter**
- Выбрать **Flutter: New Project**
- **Download SDK**
- В диалоге **Select Folder for Flutter SDK** указать путь для установки **Flutter**
- **Clone Flutter**
- **Add SDK to PATH**
- Перезапустить **VS Code**

> flutter create my_app



Flutter. Первое приложение



lib

это основная папка, в которой располагается весь исходный код приложения. В ней находятся все Dart-файлы проекта, определяющие логику и интерфейс приложения. В этой папке лежит файл `main.dart`.

main.dart

Точка входа в приложение, где находится главный метод `main()`. Этот файл содержит основной код приложения и выступает отправной точкой для его запуска.

pubspec.yaml

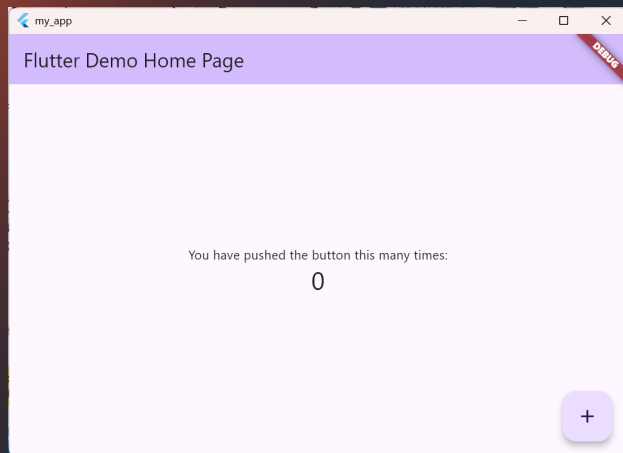
Файл относится к важным компонентам проекта. Он определяет конфигурацию проекта и используется, чтобы определить зависимости, версии библиотек, настройки сборки и другие свойства.

Flutter. Первое приложение

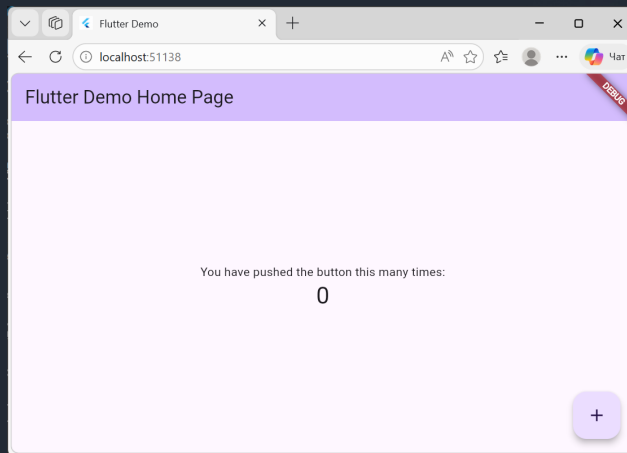
```
1 import 'package:flutter/material.dart';
2
3 Run | Debug | Profile | GigaCode: explain | explain step by step
4 void main() {
5   runApp(const MyApp());
6 }
7
8 GigaCode: explain | explain step by step
9 class MyApp extends StatelessWidget {
10   const MyApp({super.key});
11
12   @override
13   Widget build(BuildContext context) {
14     return MaterialApp(
15       title: 'Flutter Demo',
16       theme: ThemeData(
17         colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
18       ), // ThemeData
19       home: const MyHomePage(title: 'Flutter Demo Home Page'),
20     ); // MaterialApp
21   }
22 }
23
24 GigaCode: explain | explain step by step
25 class MyHomePage extends StatefulWidget {
26   const MyHomePage({super.key, required this.title});
27
28   final String title;
29
30   @override
31   State<MyHomePage> createState() => _MyHomePageState();
32 }
33
```

```
31 class _MyHomePageState extends State<MyHomePage> {
32   int _counter = 0;
33
34   void _incrementCounter() {
35     setState(() {
36       _counter++;
37     });
38   }
39
40   @override
41   Widget build(BuildContext context) {
42     return Scaffold(
43       appBar: AppBar(
44         backgroundColor: Theme.of(context).colorScheme.inversePrimary,
45         title: Text(widget.title),
46       ), // AppBar
47       body: Center(
48         child: Column(
49           mainAxisAlignment: MainAxisAlignment.center,
50           children: <Widget>[
51             const Text('You have pushed the button this many times:'),
52             Text(
53               '$_counter',
54               style: Theme.of(context).textTheme.headlineMedium,
55             ), // Text
56           ], // <Widget>[]
57         ), // Column
58       ), // Center
59       floatingActionButton: FloatingActionButton(
60         onPressed: _incrementCounter,
61         tooltip: 'Increment',
62         child: const Icon(Icons.add),
63       ), // FloatingActionButton
64     ); // Scaffold
65   }
66 }
67
```

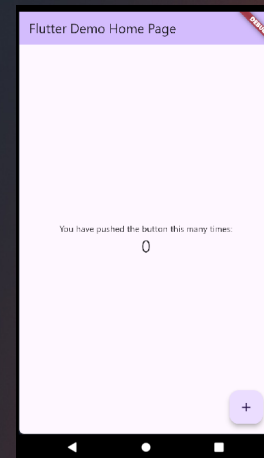
Flutter. Первое приложение



> flutter build windows

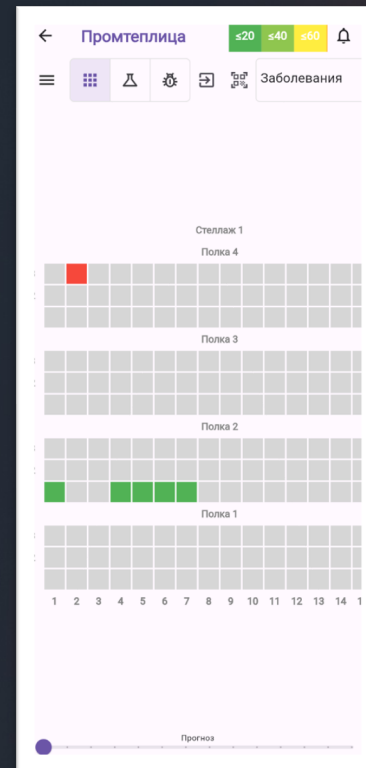
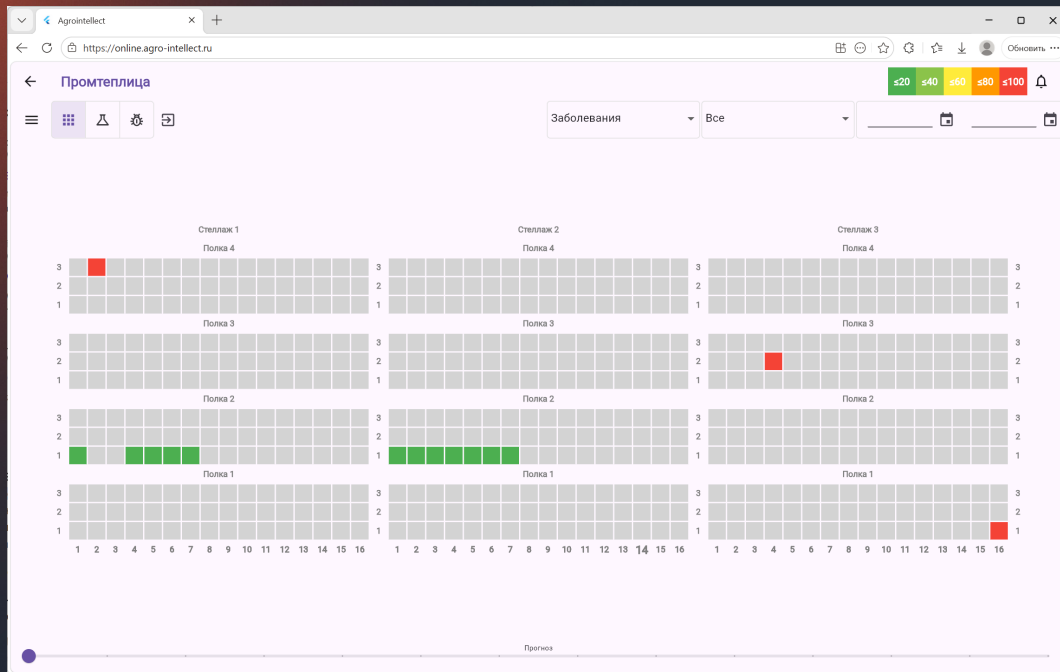


> flutter build web

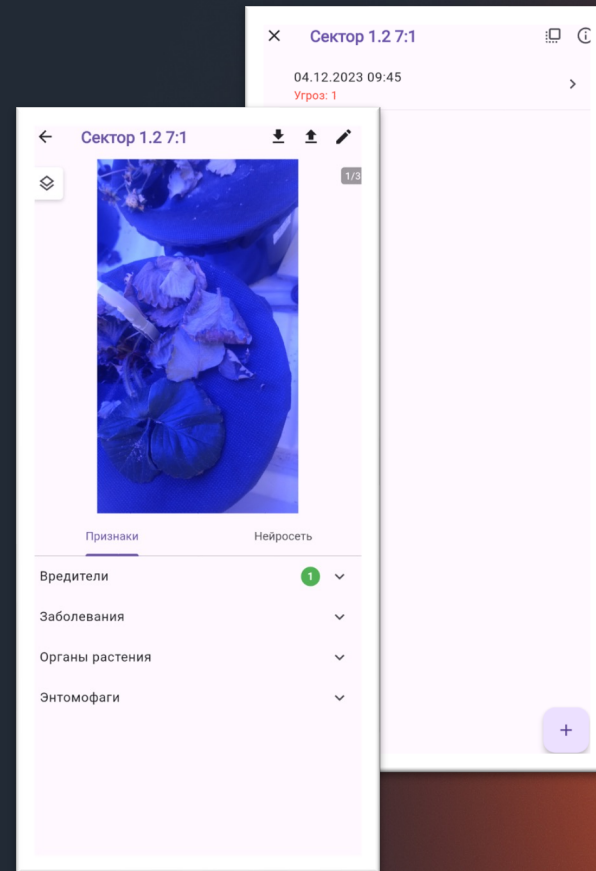
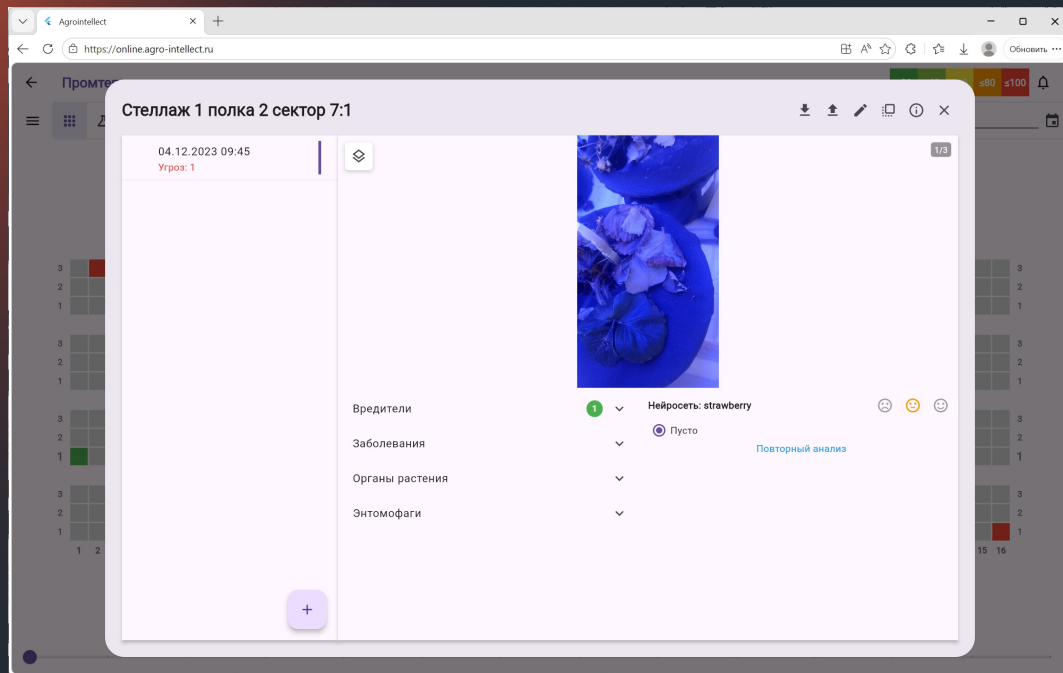


> flutter build android

Agrointellect



Agrointellect



Flutter + API

Прототип на Flutter

Переписали приложение
Работа через API

Клиенты против

У нас всё работает –
не трогайте

Режим работы

- API

Flutter + прямое подключение к Firebird

Вопрос

Можно ли Flutter подключить напрямую к Firebird?

Ответ

fbdb - написанный драйвер для Dart

Результат

- Работает
- Стабильна
- Закрывает задачи

fbdb

```
import "package:fbdb/fbdb.dart";

void main() async {
  FbDb db = await FbDb.attach(
    host: "localhost",
    database: "employee",
    user: "SYSDBA",
    password: "masterkey",
  );
  final q = db.query();
  await q.openCursor(
    sql: "select FIRST_NAME, LAST_NAME "
        "from EMPLOYEE "
        "order by LAST_NAME "
        "rows 10 ";
  );
  await for (var r in q.rows()) {
    print("${r['LAST_NAME']}, ${r['FIRST_NAME']}");
  }
  await q.close();
  await db.detach();
}
```



ABI, УПАКОВКА, ЗАВИСИМОСТИ

```
APK / AAB
└─ lib/
    ├── arm64-v8a/
    │   └─ libfbclient.so
    ├── armeabi-v7a/
    │   └─ libfbclient.so
    ├── x86_64/
    │   └─ libfbclient.so
    └─ x86/
        └─ libfbclient.so
```

API vs Direct

| Критерий | API | Прямое подключение |
|-------------------------|---------------|---------------------|
| Прототип | Медленнее | Быстрее |
| Безопасность | Выше | Зависит от защиты |
| Работа через интернет | Надёжнее | Рискованнее |
| Работа в локальной сети | Хорошо | Часто оправдано |
| Масштабирование | Лучше | Сложнее |
| Изменение схемы БД | Гибче | Может ломать клиент |
| Бизнес-логика | Сервер | Клиент + БД |
| Контроль | Центролизован | Распределён |

Google Play

Google Play Console

Тестирование и выпуск

разные функции, создайте несколько каналов закрытого тестирования.

Для следующего выпуска

Отображение от края до края может работать не у всех пользователей

Начиная с Android 15, показ от края до края будет включен по умолчанию для всех приложений, использующих SDK 35. Чтобы контент приложений выглядел корректно в Android 15 и более новых версиях, разработчики должны будут самостоятельно регулировать отступы. Изучите этот вопрос, протестируйте показ от края до края и внесите необходимые изменения. Чтобы обеспечить обратную совместимость, вы также можете вызвать метод `enableEdgeToEdge()` в Kotlin или `EdgeToEdge.enable()` в Java.

Удобство использования | Название выпуска: 13 (2.1.0)

В вашем приложении используются неподдерживаемые API или параметры отображения от края до края

Один или несколько используемых вами API или параметров показа от края до края и показа окон не поддерживаются в Android 15. Вам нужно отказаться от них, чтобы исправить проблему.

Удобство использования | Название выпуска: 13 (2.1.0)

Перекompилитруйте приложение, чтобы нативные библиотеки поддерживали страницы памяти размером 16 КБ

В приложении есть нативные библиотеки, несовместимые с устройствами, которые используют страницы памяти размером 16 КБ. На таких устройствах приложение может не устанавливаться, не запускаться или работать со сбоями.

Техническое качество | Название выпуска: 13 (2.1.0)

Информация об обновлениях | Панель мониторинга | Справка | © Google, 2025 | Конфиденциальность | Соглашение о распространении ПО | Условия использования

GitHub Actions

```
name: 663 lines (554 loc) - 21.2 KB
jobs:
  CI:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout
        uses: actions/checkout@v4
      - name: Setup build environment
        uses: actions/setup-ubuntu@v2
        with:
          version: 24.04
      - name: Cache dependencies
        uses: actions/cache@v4
        with:
          path: |
            .github
            actions/x64
            ...github/ctions/adjust-snapshot-buildno
      - name: Install dependencies
        run: |
          sudo apt-get update
          sudo apt-get install libtool-bin libtomcrypt libtommath1 libtommath1 libcu-dev 2周假.z/libig-dev cmake
      - name: Build and test
        run: |
          CK=clang CXX=clang++ ./autogen.sh --enable-binreloc --prefix=/opt/firebird
          make -j4
          make tests -j4
          make run_tests
          make dist
      - name: Upload artifacts
        run: |
          tar zxvf 发布.Fire.2B0*.linux-x64.tar.gz
          (cd Firebird-*.linux-x64; sudo ./install.sh --silent)
```

GitHub Actions

```
Code Blame 663 lines (554 loc) · 21.2 KB
1 name: CI
2
3 on: [push, pull_request]
4
5 jobs:
6
7   build-linux-ubuntu-x64-clang:
8     if: github.ref_type != 'tag'
9     runs-on: ubuntu-24.04
10
11    steps:
12      - name: Checkout
13        uses: actions/checkout@v4
14        with:
15          fetch-depth: 10
16
17      - name: Adjust snapshot build number
18        uses: ./github/actions/adjust-snapshot-buildno
19
20      - name: Prepare
21        run: |
22          sudo apt-get install libtool-bin libtomcrypt1 libtomcrypt-dev libtommath1 libtommath-dev libicu-dev zlib1g-dev cmake
23
24      - name: Build
25        run: |
26          CC=clang CXX=clang++ ./autogen.sh --enable-binreloc --prefix=/opt/firebird
27          make -j4
28          make tests -j4
29          make run_tests
30          make dist
31          tar xzvf gen/Firebird-*-linux-x64.tar.gz
32          (cd Firebird-*-linux-x64; sudo ./install.sh -silent)
33
34      - name: Upload installer
```

GitHub Actions

```
export LDFLAGS="-Wl,-z,max-page-size=16384"
```

Будущее. Что на счёт iOS?

- собрать нативные библиотеки
- ждать wire protocol в fbdb

Спасибо за внимание!