Software Project
Assignment 4: Quality Assurance & Automation

Project: TourGuide Manager

Customers: **Karina Shavalieva** (@K_Arbyzova),
Hannanov Rishat (@RishatHannanov)

Team 59 (Members and Their Contribution):

- **Alexey Chegaev** (a.chegaev@innopolis.university) - Team lead, configure database and programmed authorization.
- **Aleksandr Medvedev** (a.medvedev@innopolis.university) - flutter developer, programmed calendar for guides.
- **Nikita Shankin** (n.shankin@innopolis.university) - flutter developer, programmed profile.
- **Georgii Beliaev** (g.beliaev@innopolis.university) - web-developer, writing web-site for admins.
- **Alexander Simonov** (a.simonov@innopolis.university) - web-developer, writing blacklist of users and guides.

Date: 30.06.2025

QA tools & CI pipeline

GitHub Actions CI for Flutter

- **File:** `.github/workflows/flutter_ci.yml`

- **Link**

What it does

- **Runs tests**
- **Pull requests targeting** `main`

Steps included

- **Checkout the repository**
  - **Uses** `actions/checkout@v4`
- **Install Flutter**
  - **Uses** `subosito/flutter-action@v2` **with:**
    - **Flutter version:** `3.32.0`
    - **Channel:** `stable`
    - **Caching enabled**

Steps included

- **Verify Flutter version**
  - **Runs** `flutter --version`
- **Install project dependencies**
  - **Runs** `flutter pub get`
- **Analyze the code (linting)**
  - **Runs** `flutter analyze`
- **Run unit tests**
  - **Runs** `flutter test`
- **Run integration tests (optional)**
  - **Runs:**
    ```
    set +e
    flutter test integration_test || true
    ```

Purpose

- **This pipeline ensures code quality and stability by automatically analyzing the code and running tests whenever changes are pushed or merged into the main branch.**

Justified List of QA Tools Used

- **`flutter analyze`**
  - Used to perform static code analysis. It helps identify syntax errors, unused imports, and code that doesn't follow Dart or Flutter best practices. This ensures cleaner, more maintainable code across the project.

- **`flutter test`**
  - Used to run unit tests on business logic and smaller isolated components of the application. It helps verify that individual functions or classes work correctly.

- **`integration_test`**
  - Used to perform integration testing by simulating real user interactions with the app (e.g., tapping buttons, filling forms). This validates that different components work together as expected.

Justified List of QA Tools Used

- **`flutter_lints`**
  - A predefined set of linting rules recommended by the Flutter team. It enforces consistent coding style and best practices across the team, reducing bugs caused by inconsistent code.
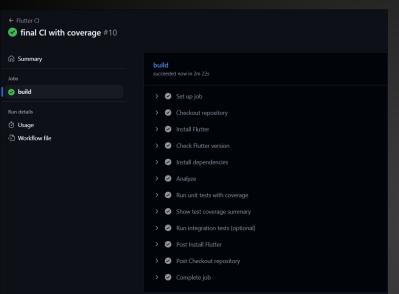
- **`GitHub Actions`**
  - Used to automate CI/CD. It runs all the checks (analyze, test) automatically on each push or pull request to the main branch. This ensures code quality is continuously maintained and regressions are caught early.

- **`flutter test --coverage`**
  - Generates test coverage reports to track how much of the code is exercised by tests. This helps identify untested code areas and improve test completeness.

Screenshots of testing reports

← Flutter CI
✅ **final CI with coverage** #10

🏠 Summary

Jobs
✅ build

Run details
⏱ Usage
📄 Workflow file

build
succeeded now in 2m 22s

> ✅ Set up job
> ✅ Checkout repository
> ✅ Install Flutter
> ✅ Check Flutter version
> ✅ Install dependencies
> ✅ Analyze
> ✅ Run unit tests with coverage
> ✅ Show test coverage summary
> ✅ Run integration tests (optional)
> ✅ Post Install Flutter
> ✅ Post Checkout repository
> ✅ Complete job

∨ ✅ Analyze

```
1  ▶ Run flutter analyze
7    Analyzing tour_guide_manager...
8    No issues found! (ran in 12.0s)
```

✅ Run unit tests with coverage

```
1   ▶ Run flutter test --coverage
7
8   ✅ /home/runner/work/SWD_Project/SWD_Project/tour_guide_manager/test/email_input_test.dart: Email controller holds input value
9   ✅ /home/runner/work/SWD_Project/SWD_Project/tour_guide_manager/test/excursion_model_test.dart: ExcursionModel initializes correctly
10  ✅ /home/runner/work/SWD_Project/SWD_Project/tour_guide_manager/test/password_test.dart: Rejects short passwords
11  ✅ /home/runner/work/SWD_Project/SWD_Project/tour_guide_manager/test/password_test.dart: Accepts long passwords
12  ✅ /home/runner/work/SWD_Project/SWD_Project/tour_guide_manager/test/format_manual_test.dart: Correctly formats date (manual)
13  ✅ /home/runner/work/SWD_Project/SWD_Project/tour_guide_manager/test/date_format_test.dart: Formats date as yyyy-MM-dd
14
15  🎉 6 tests passed.
```

All Links There