

Технічне завдання для програміста Flutter

0) Техстек і стандарти

Target: iOS 14+, Android 8+ (minSdk 26)

Flutter: 3.24+ (Dart 3.x)

Архітектура: **feature-first** + **clean** (domain/data/presentation), **Riverpod 2.x**, **Dio** + **Retrofit**, **Isar** (offline DB), **go_router** (routing + deep links)

i18n: flutter_localizations + intl (ключі EN)

Статичний аналіз: flutter_lints, dart_code_metrics, very_good_analysis (на вибір)

CI: GitHub Actions (build + tests), Fastlane (дистрибуція)

Ідентифікатори: **UUID** скрізь

Рекомендовані пакети (pubspec)

- **network:** dio, retrofit, dio_smart_retry, pretty_dio_logger
- **state/DI:** flutter_riverpod, riverpod_annotation, hooks_riverpod (за бажанням)
- **models:** freezed, json_serializable, build_runner
- **db/offline:** isar, isar_flutter_libs, path_provider
- **secure:** flutter_secure_storage (tokens), shared_preferences
- **routing:** go_router, uni_links (або вбудований deep link з go_router)
- **media:** video_player, chewie, subtitle (SRT), flutter_svg, pdfx, open_filex, share_plus
- **downloads:** background_downloader (фонова викачка), permission_handler
- **qr:** mobile_scanner
- **analytics/logging:** sentry_flutter (або firebase_analytics), logger
- **a11y:** вбудовані Flutter Semantics + масштаб шрифтів

1) Структура проєкту

lib/

 app/

 app.dart

 router.dart

 theme/

 theme.dart

```
tokens.dart
core/
  constants/
  utils/
  error/
  network/
    dio_client.dart
    interceptors/
      auth_interceptor.dart
      retry_interceptor.dart
  storage/
    secure_storage.dart
    prefs.dart
  sync/
    sync_manager.dart
  analytics/
    analytics.dart
features/
  auth/ ...
  onboarding/ ...
  subjects/      # Subject → Unit → Module → Content
  modules/      # екрани модуля (Video/Infographic/Quiz)
  downloads/ ...
  submissions/ ...
  classes/ ...
  certificates/ ...
  search/ ...
  profile/ ...
l10n/
  intl_en.arb
  intl_uk.arb
  intl_pl.arb
```

Моделі/DTO (frozen + json_serializable): SubjectDto, UnitDto, ModuleDto, ContentDto, QuizDto, QuestionDto, CertificateDto, SubmissionDto, ClassDto, UserDto.

Isar: дзеркальні Entity з UUID та індексами по updatedAt, title, subjectId/moduleId.

2) Навігація (Routing & deep links)

- **go_router** зі схемою:
 - /home (Subjects grid)
 - /subject/:id → якщо 1 модуль ⇒ одразу /module/:id, інакше /subject/:id/modules
 - /module/:id/(video|infographic|quiz)
 - /result/:quizResultId
 - /downloads
 - /profile
 - /join/:token (deep link із QR)
- **Deep link**: обробка akad://join/<inviteToken> (Class join). Якщо неавторизований — спершу логін, потім join.

3) Аутентифікація (JWT) і онбординг

- **Auth flow**:
 - Email/Password (Devise) → отримуємо **accessToken** + **refreshToken** (JWT).
 - Зберігаємо в **SecureStorage**; **AuthInterceptor** додає **Authorization: Bearer**.
 - **Refresh** по 401: прозора спроба, fallback → login.
- **Join flow**:
 - Відкриття QR/link (/join/:token) → POST /v1/classes/join з токеном → **status=pending**.
 - UI показує **Pending approval** до підтвердження Teacher/School Admin.
- **Device info**: збережи deviceId (UUID) для телеметрії.

Acceptance:

- Токени зберігаються безпечно; автоматичний refresh; глибокі посилання працюють і після релогу.
-

4) Subjects → Modules → Content (Tile UI + правила)

- **Home: Subjects grid (Tile UI)** з:
 - назвою, іконками вмісту (Video/Infographic/Quiz), **progress ring, badges: Downloaded, Completed.**
- **Single-module Subject:** тап по Tile ⇒ **Video** (skip modules list) ⇒ **Infographic** ⇒ **Quiz.**
- **Multi-module:** Tile ⇒ **Modules list** ⇒ стандартний потік.

Offline UX:

- Кнопка **Download Subject/Module**, банер *Offline*, сторінка **Downloads** з керуванням, **Wi-Fi only** налаштування.

Acceptance:

- Single vs Multi rule дотримано; прогрес синхронний з локальною БД; офлайн статус видно на Tile.

5) Відео, інфографіка, PDF

- **Video:** video_player + chewie (контроли), **SRT** через subtitle (парсер) + overlay. Кеш файлів — **background_downloader** у appDir/media/...
- **Infographic:** flutter_svg (SVG), fallback PNG; pinch-to-zoom (власно або через interactive_viewer).
- **PDF** (матеріали/сертифікат): перегляд через pdfx, **Share** через share_plus, **Open** через open_filex.

Acceptance:

- Відео з субтитрами (toggle), збереження позиції; зум для інфографік; відкриття/шарінг PDF.

6) Quiz → Result → Certificate

- **Quiz:** single (Radio), multiple (Checkbox). Локальна валідація, submit: POST /v1/quiz_results.
- **Score** на бекенді; якщо $\geq 80\%$ — **certificate_ready=true.**
- **Certificate:**
 - GET /v1/certificates/:id.pdf → зберегти локально (імена по UUID), показати **Result & Certificate:** Download, Share.

- **Telemetry:** `quiz_submitted`, `certificate_downloaded`.

Acceptance:

- Валідація станів; недоступність **Download** до `pass`; файл зберігається й відкривається офлайн.
-

7) Offline-first & Sync

- **DB:** Isar схеми з індексами:
 - `SubjectEntity(id PK uuid, title, order, updatedAt)`,
 - `ModuleEntity(id, subjectId, order, singleFlow, updatedAt)`,
 - `ContentEntity(id, moduleId, type, title, durationSec, updatedAt)`.
- **SyncManager:**
 - стартовий **bootstrap**: завантажити **Subjects/Units/Modules/Contents** (пагінація, `updated_since` якщо є).
 - **Delta sync** при запуску/появі мережі (Rx із Riverpod).
 - **Downloads**: черга в `background_downloader` (великі медіа), `retry`, прогрес у UI.
- **Conflict policy**: дані `read-only` для контенту; `submissions` — версіонування, див. розд. 9.

Acceptance:

- App повністю працює офлайн після завантаження; повторні старту не дублюють записи; прогрес/синхронізація стабільні.
-

8) Downloads (менеджер завантажень)

- **API**: отримати маніфест файлів (`media urls + hashes`); порівнювати **etag/md5** для перевантаження.
- Фонові завантаження в iOS/Android через `background_downloader` з **persistent notifications** на Android (політика ОС).
- **Settings**: “Wi-Fi only”, “Remove download”, розмір і зайняте місце.

Acceptance:

- Керування станами: `queued/running/paused/completed/failed`; коректне відновлення після рестарту ОС.
-

9) Submissions (учнівські матеріали)

- **Enable per Module** (флаг із бекенду).
- **Create submission:** вибір файлів (фото/відео/pdf), опис; POST /v1/submissions.
- **Queue:** у офлайн — лягає в **outbox** (Isar). При з'явленні мережі — автосенд (retry з backoff).
- **Statuses:** submitted → under_review → accepted/rejected (polling або push-сигнали — якщо будуть).
- **List & search:** локальна пагінація + серверні фільтри (subject/module/type/status/query, date range).

Acceptance:

- Надсилання в офлайн з подальшим аплодом; перегляд статусів; масштабований пошук.
-

10) Пошук

- **Local:** по cached Subjects/Modules/Contents (title, tags), submissions (title, filename, type, status).
- **Remote:** фільтри як у бекенді; UI збережених пошуків (опц.).
- **Index:** Isar індекси по titleLower, updatedAt, type, status.

Acceptance:

- 10k+ елементів шукаються локально < 100 мс (на середньому девайсі) по title/tags.
-

11) Аналітика (Telemetry)

Події:

subject_opened, module_opened, video_started/completed, infographic_opened, quiz_started/submitted, certificate_downloaded, download_started/completed/failed, submission_created/accepted/rejected, offline_entered/left.

Буфер у офлайн з подальшою відправкою. Додавати **userId, schoolId, classId, subjectId, moduleId, client=mobile, deviceId**.

Acceptance:

- Події акумулюються офлайн і коректно флешаться онлайн; відсутні дублікати.
-

12) Доступність (WCAG 2.1 AA)

- **Dynamic Type:** мінімум 3 кроки масштабування без зламу верстки.
- **Semantics:** ролі, labels для контролів, **focus order** логічний.
- **Captions/Subtitles:** перемикач; контраст $\geq 4.5:1$; **hit area** $\geq 44 \times 44$ dp.

Acceptance:

- Перевірка через Accessibility Scanner (Android), VoiceOver (iOS) — критичні сценарії прохідні.
-

13) Тести

- **Unit:** репозиторії, мапінги DTO \leftrightarrow Entity, синх-логіка.
- **Widget:** Tiles (варіанти станів), Video/Infographic/Quiz/Result.
- **Golden:** SubjectTile, Badges, Buttons (світла/темна тема, збільшений шрифт).
- **Integration:** e2e — Login \rightarrow Join via deep link \rightarrow Download \rightarrow Offline playback \rightarrow Quiz \rightarrow Certificate \rightarrow Submission.

Acceptance:

- Покриття: **unit 70%+** критичних шарів; **e2e** проходить на обох платформах у CI.
-

14) CI/CD

- **GitHub Actions:**
 - `flutter analyze + format --set-exit-if-changed`
 - `flutter test (+ golden)`
 - build **Android AAB**; iOS archive через **Fastlane** (manual signing на початку)
- **Env:** `.env` (dart-define) для `API_BASE_URL`, `SENTRY_DSN`, `FEATURE_FLAGS`.

Acceptance:

- PR тригерить лінти/тести; main — білди артефактів.
-

15) Етапи та deliverables

M1 — Каркас & базова навігація

- Ініціалізований проєкт, теми (light/dark), i18n scaffold.

- **go_router** з основними маршрутами, deep link /join/:token.
- **Auth** (login, токени в SecureStorage, refresh).
- **Subjects grid (Tile UI)** + заглушки екранів Module/Video/Infographic/Quiz.
- **Isar** підключено, базові entity, початковий **SyncManager** (метадані без великих медіа).
- Логи/аналітика-шар (по-ор адаптер).

Acceptance: логін, відображення предметів зі сторони API, відкриття single/multi subject за правилом, deep link join відкривається.

M2 — Функціонал 1.0 + офлайн + сертифікат

- Повний контент-флоу **Video** → **Infographic** → **Quiz** → **Result/Certificate**.
- **Subtitles (SRT), PDF view/share, Downloads** (subject/module, менеджер черги).
- **Offline-first:** апка працює без мережі після завантаження; retry політики.
- **Search (local)** по кешу; телеметрія ключових подій; **Settings** (Wi-Fi only).
- Оптимізація продуктивності (перше відображення ≤ 3 с на 4G).

Acceptance: пілот у 4 школах прохідний (включно з офлайном), сертифікат генерується/скачується, експорт CSV відображається у вчителя (web).

M3 — Після тестів + Submissions

- **Submissions:** створення/черга в офлайні/відправка/перегляд статусів, **filters & search**.
- Полірування UX (empty/error/offline states), accessibility audit.
- Телеметрія розширена; стабілізація та багфікс; фінальна документація.

Acceptance: повний lifecycle submissions з перевіркою вчителем (web), офлайн-черга працює, метрики знімаються.

16) API інтеграція (скорочено, для контрактів)

- **Auth:** /v1/auth/login, /v1/auth/refresh, /v1/auth/logout
- **Subjects/Modules/Contents:** /v1/subjects?updated_since=..., /v1/modules, /v1/contents
- **Quiz:** POST /v1/quiz_results → { score, passed, certificate_id? }
- **Certificates:** GET /v1/certificates/:id.pdf
- **Classes:** /v1/classes, POST /v1/classes/join (token)

- **Submissions:** GET/POST /v1/submissions, GET /v1/submissions? filters...
- **Search:** /v1/search?scope=content|submissions&...
- **Telemetry:** POST /v1/events

Всі ID — **UUID**, відповіді — **JSON:API** або узгоджений JSON, часові поля — ISO8601 UTC.

17) Нотатки по перформансу

- Кешуйте списки та деталі окремо; використовуйте **KeepAlive** для списків.
 - Ледачі викачування великих медіа (після підтвердження користувача).
 - Мемоізуйте парсинг **SRT**; не блокуйте UI на heavy I/O (Isolate при потребі).
 - Обов'язково профілювання на “середньому” Android (3–4 ГБ RAM).
-

18) Handoff UX/UI

- Беріть **tokens** з Figma (Tokens Studio), мапте на ThemeData.
- Компоненти: **SubjectTile, ModuleTile, Buttons, Badges, Progress, Quiz controls** — 1:1 з Figma.
- Доступність: не вимикайте **Semantics**, враховуйте великі шрифти та reduced motion.