

ТЗ: Android-прошивка точки мониторинга TazaQ с загрузкой ИИ-модели и автонастройкой

Oct 14, 2026 · @Bekzhigit

1. Общие сведения

Нужно Android-приложение, работающее как прошивка устройства: оно превращает Android-устройство с камерой в точку ИИ-мониторинга TazaQ, само скачивает ИИ-модель с сервера, настраивается по QR-коду площадки и передаёт статусы без участия человека.

Назначение. Определять на контейнерной площадке уровень заполнения (пустой, наполовину, полный, переполнен) и нарушения (мусор рядом с баками, крупногабаритный мусор, бак опрокинут или отсутствует).

Место в системе. Это альтернатива плате Luckfox из варианта 2 «Камеры с ИИ». Приложение работает с тем же сервером, MQTT-брокером и панелью диспетчера TazaQ. Обучение ИИ-модели в объём этого ТЗ не входит: исполнитель получает готовый файл модели от команды ML.

Допущения.

- Устройство — Android-смартфон или Android-камера в уличном корпусе IP66 с подогревом и питанием от 220 В.
- Root-доступ не используется; управление устройством — через режим Device Owner (Android Enterprise).
- Один экземпляр приложения обслуживает одну площадку.

Открытые вопросы.

- ☐ Конкретная модель устройства для пилота
- ☐ Нужна ли сторонняя MDM-система или достаточно собственного Device Owner
- ☐ Кто размечает зону баков при монтаже: монтажник на устройстве или диспетчер на панели

2. Требования к устройству и окружению

Приложение должно работать на Android 9 и новее без root; рекомендуемое устройство — Android 11+ с 3–4 ГБ ОЗУ.

Параметр	Минимум	Рекомендуется
Версия Android	9 (API 28)	11 и новее
Оперативная память	2 ГБ	3–4 ГБ
Встроенная память	16 ГБ	32 ГБ
Камера	Тыловая 8 Мп с автофокусом	Плюс ночной режим или ИК-подсветка в корпусе
Связь	4G LTE, слот SIM	4G LTE и Wi-Fi
Ускорение ИИ	CPU	GPU или NPU устройства
Питание	Постоянное 220 → 5 В	С контролем заряда аккумулятора
Корпус	IP66	IP66 с подогревом и термостатом

Аккумулятор. Литиевая батарея при постоянной зарядке в нагретом корпусе может вздуться. Устройство должно поддерживать ограничение заряда (например, до 80%) или работу от питания в обход батареи; иначе это фиксируется как риск пилота.

Технологический стек. Kotlin; CameraX для съёмки; LiteRT (TensorFlow Lite) для инференса с делегатами GPU, NNAPI (на версиях, где он поддерживается) и CPU (XNNPACK); WorkManager для фоновых задач; Room для очереди; MQTT-клиент с поддержкой TLS (Eclipse Paho или HiveMQ).

3. Автонастройка

Новое устройство должно пройти путь от включения до статуса «точка готова» не дольше чем за 15 минут, без ПК и без ввода текста монтажником.

- Установка.** На экране приветствия монтажник сканирует QR-код провижининга Android Enterprise. Устройство само скачивает APK по ссылке из кода, проверяет его контрольную сумму и назначает приложение владельцем устройства (Device Owner). Для пилота допускается установка через ADB с командой назначения Device Owner.

2. **Идентификация.** При первом запуске приложение создаёт пару ключей в Android Keystore и уникальный device_id.
3. **Привязка к площадке.** Монтажник наводит камеру устройства на QR-табличку площадки — ту же, что сканируют жители. Приложение извлекает токен площадки и вызывает регистрацию на сервере (раздел 7). Сервер возвращает конфигурацию и учётные данные MQTT.
4. **Загрузка модели.** Приложение скачивает актуальную ИИ-модель по правилам раздела 4.
5. **Зона баков.** Приложение показывает кадр, монтажник пальцем обводит зону с баками. Зона сохраняется в конфигурации и отправляется на сервер; диспетчер может поправить её позже с панели.
6. **Самопроверка.** Приложение проверяет камеру, связь, точное время, тестовый прогон модели, температуру и питание, отправляет на сервер отчёт «точка готова» и переходит в режим киоска.

Параметры, которые приходят с сервера и меняются без выезда:

интервал съёмки (по умолчанию 15 минут), порог уверенности модели, зона баков, ночное расписание, канал обновлений модели (stable или beta), адрес MQTT-брокера. Конфигурация запрашивается при каждом запуске и по команде с сервера.

Если на любом шаге произошла ошибка, приложение показывает монтажнику понятный текст причины и кнопку «Повторить»; повторная привязка к другой площадке возможна только по команде с сервера.

4. Загрузка и обновление ИИ-модели

Модель обновляется по воздуху атомарно: новая версия включается только после проверки подписи и успешного тестового прогона, а предыдущая хранится для отката.

Формат пакета модели. Файл .tflite (детектор класса YOLO-nano, квантование INT8) и файл метаданных JSON: версия, список классов, размер входа, пороги, SHA-256, подпись сервера, минимальная версия приложения, эталонное изображение и ожидаемый на нём результат.

Когда проверять обновления: при запуске, раз в сутки и по команде с сервера.

Порядок обновления.

1. Запросить манифест последней версии для своего канала (stable или beta).

2. Если версия новее установленной — скачать файл во временную папку с докачкой при обрыве (HTTP Range), только при наличии сети.
3. Проверить SHA-256 и подпись сервера; при несовпадении удалить файл и сообщить об ошибке.
4. Прогнать модель на эталонном изображении и сравнить с ожидаемым результатом.
5. Переключить активную модель; предыдущую версию сохранить.
6. Отправить на сервер подтверждение с новой версией и временем инференса.

Выбор ускорителя. После установки каждой модели приложение замеряет скорость и корректность на GPU, NNAPI (где доступен) и CPU и выбирает самый быстрый вариант, который даёт правильный результат на эталоне. Выбор попадает в отчёт на сервер.

Автооткат. Если после обновления ошибки инференса повторяются 3 раза подряд или сервер прислал команду отката — приложение возвращает предыдущую версию модели и сообщает об этом.

5. Работа на точке

Каждые 15 минут (интервал задаётся с сервера) приложение делает снимок, определяет статус площадки на устройстве и отправляет на сервер только результат; видео по сети не передаётся.

Цикл работы.

1. Снимок через CameraX из фонового сервиса с типом camera, без вывода изображения на экран.
2. Обрезка кадра по зоне баков и инференс модели.
3. Сведение результатов по бакам в статус площадки: уровень заполнения (по самому полному баку) и список нарушений.
4. Постановка сообщения в локальную очередь.

Что и когда отправляется.

- Статус — только при его изменении, плюс контрольное сообщение раз в час.
- Нарушение — сразу, с фото площадки JPEG до 200 КБ, обрезанным по зоне баков.
- Heartbeat — раз в 5 минут: версия приложения и модели, заряд, температура, уровень сигнала, свободная память, время последнего инференса.

- Фото для дообучения модели — по команде сервера или до 10 случайных снимков в сутки.

Очередь и потеря связи. Сообщения хранятся в локальной базе и уходят при появлении связи (store-and-forward). Порядок отправки: нарушения, затем текущий статус, затем история. Очередь хранит данные минимум 7 суток или до 500 МБ, после чего удаляются самые старые записи истории, но не нарушения.

Ночной режим. По расписанию с сервера или по низкой освещённости приложение переключается на ночные параметры камеры; если кадр слишком тёмный для надёжного распознавания, статус помечается как «нет данных», а не «пустой».

Пример сообщения о статусе:

```
{"site":"k7Qm2x","device":"a1f3","ts":"2026-10-14T14:32:00+05:00","fill":"overflow","violations":["trash_nearby"],"conf":0.87,"model":"1.4.0"}
```

6. Режим киоска, автозапуск и самовосстановление

Точка должна работать неделями без людей: после перезагрузки или отключения питания приложение само возобновляет работу не позже чем через 2 минуты.

Режим киоска. Приложение работает как Device Owner в режиме Lock Task: выйти из него, открыть настройки, отладку по USB или установить другое приложение нельзя. Служебный экран открывается только по команде с сервера.

Автозапуск. Запуск по событию загрузки системы, работа в постоянном фоновом сервисе (foreground service с типом camera), исключение из оптимизации батареи.

Самовосстановление.

- Если цикл съёмки не выполнялся 2 интервала подряд — перезапуск сервиса.
- Если перезапуск не помог — перезагрузка устройства средствами Device Owner.
- Плановая перезагрузка раз в сутки ночью (включается с сервера).
- При температуре устройства выше 45 °C интервал съёмки удваивается, выше 55 °C съёмка приостанавливается до охлаждения; оба события уходят на сервер.

Обновление приложения по воздуху. Новая версия APK устанавливается без участия человека через PackageInstaller. Перед установкой проверяется подпись; если после обновления приложение падает 3 раза подряд, сервер получает отчёт, а монтажнику назначается выезд.

Удалённые команды с сервера: перезагрузка, перечитать конфигурацию, сделать снимок сейчас, выгрузить логи, откатить модель, сбросить привязку к площадке, открыть служебный экран.

7. Протокол и API обмена с сервером

Настройка, модели и фото идут по HTTPS, а статусы, события и команды — по MQTT с QoS 1 и постоянной сессией.

HTTPS API

Метод	Назначение
POST /api/v1/devices/register	Регистрация устройства: device_id, токен площадки, открытый ключ, модель устройства; в ответ — конфигурация и учётные данные MQTT
GET /api/v1/devices/{id}/config	Текущая конфигурация точки
GET /api/v1/models/latest?channel=stable	Манифест последней модели: версия, ссылка, размер, SHA-256, подпись
GET (временная ссылка из манифеста)	Файл модели с поддержкой докачки
POST /api/v1/devices/{id}/photos	Фото нарушения или фото для дообучения
POST /api/v1/devices/{id}/logs	Выгрузка логов по команде

MQTT-топики

Топик	Направление	Содержимое
tazaq/{site}/{device}/status	Устройство → сервер	Статус площадки
tazaq/{site}/{device}/event	Устройство → сервер	Нарушения, ошибки, смена модели

Топик	Направление	Содержимое
tazaq/{site}/{device}/heartbeat	Устройство → сервер	Состояние устройства раз в 5 минут
tazaq/{site}/{device}/cmd	Сервер → устройство	Удалённые команды
tazaq/{site}/{device}/cmd/ack	Устройство → сервер	Подтверждение и результат команды

Все сообщения — JSON с полем ts в часовом поясе Астаны (UTC+5) и уникальным id для защиты от дублей на сервере.

8. Безопасность

Устройство на улице считается доступным посторонним, поэтому всё, что приходит на него и уходит с него, подписано, зашифровано и привязано к конкретной точке.

- **Шифрование.** Весь обмен — TLS 1.2 и выше с закреплением сертификата сервера (certificate pinning).
- **Ключи.** Ключи устройства хранятся в Android Keystore, на аппаратном уровне, где это поддерживается. У каждой точки свои учётные данные MQTT; сервер может отозвать их одной командой.
- **Подпись.** Файл модели и APK проверяются по подписи перед установкой; неподписанное не запускается.
- **Локальный доступ.** Отладка по USB, режим разработчика, установка из неизвестных источников и сброс к заводским настройкам заблокированы политиками Device Owner.
- **Персональные данные.** Камера снимает двор, поэтому на сервер уходят только кадры, обрезанные по зоне баков. Лица людей и госномера машин на фото размываются на устройстве до отправки. Хранение и передача фото должны соответствовать закону РК «О персональных данных и их защите»; порядок согласовывается с юристом заказчика.

9. Нефункциональные требования, логи и мониторинг

Целевые показатели совпадают с критериями пилота варианта 2: точка работает не менее 95% времени и доставляет не менее 98% сообщений.

Показатель	Требование
Время инференса одного снимка	Не более 1 с на устройстве с 3 ГБ ОЗУ
Время работы точки	Не менее 95% за месяц
Доставка сообщений	Не менее 98%
Работа без связи	До 7 суток без потери нарушений
Мобильный трафик	Не более 100 МБ в месяц на точку, без учёта обновлений приложения
Восстановление после сбоя питания	Не более 2 минут
Температура окружающей среды	–40...+40 °С в корпусе с подогревом

Логи. Локальные логи с ротацией, не более 50 МБ; выгружаются на сервер по команде. В логи не пишутся ключи и учётные данные.

Мониторинг. На панели диспетчера по каждой точке видны: онлайн или офлайн, время последнего снимка, версия приложения и модели, температура, заряд и уровень сигнала. Если heartbeat не приходил 15 минут, точка автоматически помечается как «нет связи».

10. Состав работ, этапы и приёмка

Разработка занимает около 10 недель и укладывается в 12-недельный план пилота варианта 2; обучение модели идёт параллельно силами команды ML.

Этап	Результат	Срок
1. Каркас	Device Owner, режим киоска, автозапуск, фоновый сервис камеры	2 недели
2. Автонастройка	Провижининг по QR, привязка к площадке, разметка зоны баков, самопроверка	1 неделя
3. Модель	Загрузка, проверка подписи, тестовый прогон, выбор ускорителя, откат	2 недели
4. Работа на точке	Цикл съёмки, инференс, очередь, MQTT, ночной режим	2 недели
5. Эксплуатация	OTA приложения, удалённые команды, watchdog, логи, размытие лиц и номеров	1 неделя

Этап	Результат	Срок
6. Испытания	Работа на 2–3 реальных площадках, исправление ошибок	2 недели

Критерии приёмки.

- ☐ Новое устройство проходит путь от включения до статуса «точка готова» не дольше 15 минут, без ПК
- ☐ После отключения питания приложение само возобновляет работу не позже чем через 2 минуты
- ☐ Модель, опубликованная на сервере, установлена на всех точках на связи в течение 24 часов
- ☐ Повреждённый или неподписанный файл модели отклоняется, работа продолжается на прежней версии
- ☐ После 72 часов без связи все накопленные нарушения доставлены на сервер
- ☐ Пользователь на месте не может выйти из приложения или открыть настройки
- ☐ 14 суток непрерывной работы на площадке без ручного вмешательства

Что передаёт исполнитель: исходный код в репозитории заказчика, подписанный релизный APK, QR-код провижининга, описание API и инструкцию монтажника на 1–2 страницы.