

Техническое задание и архитектура

Мониторинг появления доступных слотов в приложении Ozon Job через UI-автоматизацию (POCO X6 Pro 5G / HyperOS).

Дата: 2026-02-07

Коротко: бот каждые N минут открывает Ozon Job, проходит маршрут Склады → поиск «Петровское» → «Записаться» → список работ, находит «Производство непрофиль», определяет статус «слоты появились / нет мест» и отправляет уведомление. Запись на смену не выполняется.

0. Важные оговорки и риски

- Это UI-автоматизация поверх чужого приложения. Любое обновление интерфейса Ozon Job может сломать распознавание экранов и «якоря».
- Проверь правила/условия Ozon Job. Агрессивная автоматизация может привести к ограничениям аккаунта. Рекомендуется держать разумный интервал проверки.
- Если планируется публикация в Google Play: использование Accessibility API требует прозрачного сценария и «детерминированной, правилами заданной» логики; автономное планирование/выполнение действий сверх статического скрипта ограничено политикой Google Play (см. источники в конце).

1. Цель, границы и предусловия

Цель: оперативно узнавать, что для склада «Петровское» и работы «Производство непрофиль» появились доступные слоты.

Не входит: автоматическая запись на смену; обход логина/пароля; перехват сетевого трафика; использование неофициальных API.

Предусловия (обязательно):

- Ozon Job установлен и пользователь уже авторизован (бот не решает капчи/2FA).
- На время проверки приложение должно открываться в foreground. Если экран заблокирован защищенной блокировкой (PIN/пароль), бот не может «взломать» разблокировку - нужно либо держать телефон разблокированным в нужные часы, либо делать уведомление «разблокируй, чтобы продолжить».
- Разрешены уведомления (иначе ты не узнаешь о результате) и выданы разрешения Accessibility (и, при OCR-варианте, MediaProjection).

2. Стратегия реализации

Основной подход - AccessibilityService (чтение дерева UI + нажатия/скролл), так как по твоим скринам элементы и тексты доступны.

Запасной подход - скриншот + OCR (MediaProjection + ML Kit Text Recognition) только если на нужных экранах Accessibility-дерево окажется «пустым» или тексты недоступны.

Важно: для проверки каждые N минут при $N < 15$ минут WorkManager обычно не подходит (в периодических задачах есть минимальные интервалы). Для «частой проверки» практичнее держать foreground service с постоянным уведомлением и внутренним таймером.

3. Пользовательский маршрут по Ozon Job (строго по твоим шагам)

Пункт	Описание	Критерий ОК
Старт	Открыть Ozon Job	Приложение на главном экране
1	Нажать нижнюю вкладку «Склады»	Внизу активна вкладка «Склады»
2	Экран «Выберите склад». Клик по поисковой строке сверху	Виден заголовок «Выберите склад» и поле поиска
3	Ввести текст: «Петровское» (предварительно очистив поле, если нужно)	В поле поиска введено «Петровское»
4	В результатах выбрать карточку склада «Петровское»	Открыта карточка/экран склада «Петровское»
5	Нажать кнопку «Записаться» (это вход в список работ по складу)	Открыт список работ склада «Петровское»
6	Прокрутить список работ до «Производство непрофиль» (обычно внизу)	Карточка «Производство непрофиль» видима
7	Определить статус слотов и при появлении доступности отправить уведомление	Отправлено уведомление и запись события в лог

3.1 Якоря экранов (чтобы бот не «потерялся»)

- Экран «Выберите склад»: заголовок «Выберите склад», наличие поля поиска, список карточек складов.
- Экран «Петровское» (карточка склада): заголовок «Петровское», кнопка «Записаться», блок «Даты: Сегодня / Завтра / +5 дней» (если присутствует).
- Экран «Петровское» (список работ): заголовок «Петровское», карточки работ, иногда секция «НЕТ МЕСТ».

4. Детект доступности «Производство непрофиль»

На твоём скрине «Производство непрофиль» находится под секцией «НЕТ МЕСТ». Это хороший, стабильный маркер.

Рекомендуемый приоритет правил:

- Правило А (главное): если карточка «Производство непрофиль» под заголовком «НЕТ МЕСТ» - считать «нет слотов».
- Правило В (появление доступности): если «Производство непрофиль» появляется вне секции «НЕТ МЕСТ» и/или у него появляются элементы выбора дат («Сегодня», «Завтра», «+N дней») или иной маркер доступности - считать «слоты есть».
- Правило С (не распознано): если карточку не удалось найти/понять секцию - статус «не распознано» (редко уведомлять, с лимитом).

4.1 Анти-спам логика (обязательно)

- Хранить «последнее состояние» цели: {нет / есть / не распознано} + время.
- Уведомлять только при переходе «нет → есть».
- Cooldown: если уже уведомил «есть», не повторять чаще чем раз в X минут (например 20-30), пока не появится новый факт (например «есть → нет → есть»).
- Лимит аварийных уведомлений («не распознано») - например не чаще 1 раза в 60 минут.

5. Планировщик и работа в фоне на Android/HyperOS

Тебе нужна частая проверка (N минут). Это упирается в ограничения фоновой работы Android и особенно Xiaomi/HyperOS.

5.1 Два режима работы

- Режим 1 (рекомендуемый): мониторинг включается пользователем и работает как foreground service с постоянным уведомлением. Внутри сервиса - таймер проверки каждые N минут.
- Режим 2: WorkManager/Alarm + редкие проверки (обычно ≥ 15 минут).

Почему так:

- У WorkManager у периодических задач есть ограничения по интервалам (PeriodicWorkRequest).
- На Android 12+ ограничен запуск foreground service из фона; мониторинг лучше включать кнопкой и держать сервис живым.
- На Android 14 для foreground service требуется указывать тип(ы) сервиса (fgs types) и разрешения.

5.2 Требования к foreground service

- Постоянное системное уведомление.
- Кнопки: Стоп и Пауза на 1 час.
- Android 14+: корректный foreground service type и соблюдение ограничений старта.
- Экономия батареи: разумный N (например 5-10 минут), backoff при ошибках.

6. POCO X6 Pro / HyperOS: что нужно настроить

На HyperOS фоновые процессы часто убиваются агрессивнее. Чтобы мониторинг не умирал, нужен чеклист:

Пункт	Описание	Критерий ОК
Автозапуск	Разрешить приложению запуск в фоне (Background autostart).	Опция включена
Батарея	Для приложения выбрать «No restrictions / Без ограничений» (или «Unrestricted»).	Нет ограничений

Пауза активности	Отключить «Pause app activity if unused» (если есть).	Отключено
Закрепление	Зафиксировать приложение в недавних/«Lock app».	Закреплено
Уведомления	Разрешить уведомления и критичный канал.	Тестовое уведомление приходит
Сеть в фоне	Разрешить фоновые данные/неограниченный доступ (если режется сеть).	Сеть в фоне работает

Примечание: названия пунктов меню могут отличаться по версии HyperOS/региону, но общий смысл - автозапуск + без ограничений батареи + разрешить работу в фоне.

7. Технологии и компоненты (что именно понадобится реализовать)

7.1 Accessibility слой (основа)

- AccessibilityService: получение событий UI (смена окна, изменение содержимого).
- Чтение дерева: доступ к root и поиск узлов по тексту/классу/контент-описанию.
- Действия: клик, скролл, при необходимости Back/Home.
- Слушать события только Ozon Job (packageName).

7.2 Планировщик/исполнитель проверок

- Foreground service как «контейнер» для регулярных проверок каждые N минут.
- State machine: OPEN_APP → TAB_WAREHOUSES → SEARCH → OPEN_PETROVSKOE → OPEN_WORKS → SCROLL_TO_TARGET → EVALUATE → NOTIFY.
- Backoff при ошибках.
- Остановка/пауза пользователем.

7.3 Распознавание, фильтрация, UX

- Конфиг цели: склад, работа, профиль дат.
- Якоря экранов + правила статуса + анти-спам.
- NotificationChannel «Критичные слоты» и кнопки в уведомлении.
- Логи: история проверок, причины ошибок, экспорт.

8. Ограничения Android, которые важно учитывать

- WorkManager: надежен, но периодические задачи ограничены по интервалам; для частых проверок нужен foreground service.
- Foreground service: требует постоянного уведомления и корректного типа на Android 14+; запуск из фона ограничен на Android 12+.
- Accessibility API: чувствительный доступ. Для Google Play важно детерминированное и прозрачное поведение (без автономного «планирования»).
- HyperOS: агрессивный киллер фона, поэтому критичны автозапуск и «No restrictions» по батарее.

9. Критерии готовности (acceptance)

Пункт	Описание	Критерий ОК
A1	Бот проходит маршрут и находит «Петровское → список работ».	10/10 без ручной помощи
A2	Находит «Производство непрофиль» даже внизу.	Прокрутка стабильна
A3	Корректно распознает «НЕТ МЕСТ».	Нет ложных тревог за 24 ч
A4	При появлении доступности - уведомление.	В течение 1 цикла
A5	Анти-спам.	Нет повторов чаще cooldown
A6	После блокировки экрана (HyperOS) живёт.	Жив 6-12 часов

10. План тестирования

- 20 циклов подряд с разным состоянием поиска (пустая строка/уже введено/клавиатура открыта).
- Сценарий «потерялся»: увести на другой экран - бот возвращается к «Склады».
- Плохая сеть/офлайн: backoff и отсутствие спама.
- HyperOS: перезагрузка, автозапуск, батарея «No restrictions», тест уведомлений.
- После обновления Ozon Job - прогон и сверка лога якорей.

11. Запасной план (если Accessibility не хватает)

Если тексты/элементы недоступны через Accessibility, применить OCR-вариант:

- MediaProjection для скриншота (нужно явное согласие на захват экрана).
- ML Kit Text Recognition для извлечения текста со списка работ и секции «НЕТ МЕСТ».
- Дальше правила статуса те же.

12. Источники и ссылки (для разработчика)

Android Developers - Create your own accessibility service

<https://developer.android.com/guide/topics/ui/accessibility/service>

Android Developers - Foreground service types required (Android 14)

<https://developer.android.com/about/versions/14/changes/fgs-types-required>

Android Developers - Launch a foreground service / restrictions from background

<https://developer.android.com/develop/background-work/services/fgs/launch>

Google Play Console Help - Use of the AccessibilityService API

<https://support.google.com/googleplay/android-developer/answer/10964491>

Don't kill my app - Xiaomi

<https://dontkillmyapp.com/xiaomi>

AndroidX WorkManager source: MIN_PERIODIC_INTERVAL_MILLIS = 15 minutes

<https://android.googlesource.com/platform/frameworks/support/+androidx-master-dev/work/workmanager/src/main/java/androidx/work/PeriodicWorkRequest.java>

Android Developers - Media projection

<https://developer.android.com/media/grow/media-projection>

Google ML Kit - Text recognition v2

<https://developers.google.com/ml-kit/vision/text-recognition/v2/android>