# План тестирования IT Start Bot

1. Формирование тестовой документации
   1. Чек-листы
   2. Тест-кейсы
2. Тестирование парсеров  
   *Функциональное тестирование:*

* Запуск парсера по расписанию
* Обработка успешного парсинга
* Обработка ошибок (повторные попытки через 15/45 мин)
* Обнаружение и пропуск дубликатов
* Автоматическое извлечение тегов из текста
* Сохранение данных в БД

*Тестирование производительности:*

* Время обработки одного источника < 5 минут
* Обработка 100+ вакансий ежедневно
* Парсинг 5+ источников одновременно

1. Тестирование планировщика

*Функциональное тестирование:*

* Запуск агентов по расписанию (каждые 6 часов для вакансий)
* Ежедневная рассылка публикаций в 12:00 МСК
* Автоматическое удаление записей старше 90 дней
* Рассылка напоминаний о дедлайнах (за 3 дня)
* Обработка измененных публикаций

1. Тестирование telegram-бота

*Функциональное тестирование:*

* Все команды (/start, /help, /subscribe, /jobs и т.д.)
* Inline-кнопки и навигация
* Персонализированные рассылки
* Поиск с фильтрацией по тегам
* Управление предпочтениями

*Юзабилити-тестирование:*

* Интуитивность команд
* Скорость отклика (< 1 сек)
* Удобство использования на мобильных устройствах
* Понятность сообщений об ошибках

1. Тестирование панели администратора

*Функциональное тестирование:*

* Аутентификация (логин/пароль + 2FA + VPN)
* Управление пользователями (создание, блокировка)
* Модерация публикаций (одобрение/отклонение)
* Просмотр и редактирование публикаций
* Управление тегами и источниками
* Просмотр статистики
* Экспорт данных

*Тестирование безопасности:*

* Защита от SQL-инъекций
* Корректная работа 2FA
* Ограничение доступа по IP
* Логирование действий администраторов

1. Регрессионное тестирование

*Полномасштабная проверка:*

* Все функциональные сценарии после каждого релиза
* Критические пути использования системы
* Интеграция между всеми компонентами
* Основные пользовательские сценарии

1. График и ресурсы

*Этапы тестирования:*

1. Подготовительный (3 дня) - тест-план, чек-листы
2. Функциональное тестирование (2 недели) - все модули
3. Интеграционное тестирование (1 неделя) - взаимодействие компонентов
4. Нагрузочное тестирование (3 дня) - производительность
5. Регрессионное тестирование (после каждого спринта)

*Необходимые ресурсы:*

* Тестовая среда
* Тестовые данные
* Инструменты мониторинга
* Доступ к Telegram Test Environment

# Чек-листы

1. Чек-лист telegram-бота

1.1. Базовые команды

* \*\*/start\*\* - приветственное сообщение, регистрация в БД
* \*\*/help\*\* - отображение списка команд и inline-кнопок
* \*\*/preferences\*\* - просмотр текущих предпочтений
* Ответ бота на неизвестные команды

1.2. Система подписок

* \*\*/subscribe jobs python remote\*\* - подписка на несколько тегов
* \*\*/subscribe\*\* без параметров - подсказка по использованию
* Подписка через inline-кнопки
* Дополнение существующих предпочтений
* \*\*/unsubscribe python\*\* - отписка от конкретного тега
* \*\*/unsubscribe all\*\* - отписка от всех уведомлений
* Автоотписка при блокировке бота

1.3. Поиск публикаций

* \*\*/jobs\*\* - 10 последних вакансий
* \*\*/jobs #python #remote\*\* - поиск по тегам через хештеги
* \*\*/jobs python remote москва\*\* - поиск по тегам без хештегов
* \*\*/internships\*\* - поиск стажировок
* \*\*/conferences\*\* - поиск конференций
* Поиск через inline-кнопки с вводом параметров
* Ограничение вывода (10 записей)

1.4. Рассылки и уведомления

* Получение новых публикаций по подписке
* Формат публикации: заголовок, ссылка, описание, контакты, теги
* Напоминания о дедлайнах конференций (за 3 дня)
* Отсутствие дублирующихся уведомлений

2. Чек-лист парсеров

2.1. Базовый функционал

* Запуск исполняемого файла агента
* Парсинг данных из источников
* Сохранение данных в БД
* Логирование в Sentry

2.2. Обработка ошибок

* Повторные попытки при ошибках (15 мин, 45 мин)
* Логирование ошибок парсинга
* Продолжение работы при частичных ошибках
* Уведомление о критических сбоях

2.3. Обработка данных

* Автоматическое определение дубликатов
* Извлечение тегов из текста публикаций
* Корректное присвоение тегов по категориям
* Сохранение контактной информации

2.4. Производительность

* Время парсинга одного источника < 5 минут
* Обработка 100+ вакансий ежедневно
* Парсинг 5+ источников одновременно
* Отсутствие утечек памяти

3. Чек-лист планировщика

3.1. Расписания

* Запуск парсеров вакансий каждые 6 часов (12:00, 18:00, 00:00, 06:00 МСК)
* Ежедневный парсинг в 12:00 МСК для конференций/стажировок
* Рассылка публикаций по расписанию
* Настройка расписания через админку

3.2. Автоматические задачи

* Рассылка неотклоненных публикаций подписчикам
* Публикация в Telegram-канал
* Удаление записей старше 90 дней
* Отправка напоминаний о дедлайнах
* Обработка обновленных публикаций

3.3. Управление

* Чтение настроек из БД при инициализации
* Обновление триггеров при изменении расписания
* Обработка исключений при выполнении задач
* Логирование выполнения задач

4. Чек-лист административной панели

4.1. Аутентификация и безопасность

* Вход по логину/паролю
* Двухфакторная аутентификация
* Доступ только через VPN/Whitelist IP
* Смена пароля текущего пользователя
* Выход из системы
* Блокировка при множественных неудачных попытках входа

4.2. Управление пользователями

* Создание новых пользователей (администраторы/модераторы)
* Редактирование существующих пользователей
* Блокирование/разблокирование пользователей
* Назначение ролей (администратор/модератор)
* Генерация временных паролей

4.3. Управление публикациями

* Просмотр списка всех публикаций
* Фильтрация по дате, типу и тегам
* Пагинация при большом количестве записей
* Редактирование всех полей публикации
* Отклонение публикаций с указанием причины
* Ручная рассылка публикации

4.4. Управление источниками

* Просмотр списка источников парсинга
* Добавление новых источников
* Редактирование параметров источников
* Включение/отключение источников

4.5. Управление тегами

* Просмотр тегов по категориям
* Добавление новых значений тегов
* Редактирование тегов
* Удаление тегов

4.6. Статистика и аналитика

* Статистика пользователей за период (подписки/отписки)
* Топ-5 запрашиваемых навыков
* Количество активных пользователей
* Процент ошибок парсеров за период
* Выбор произвольного периода для анализа

4.7. Экспорт данных

* Экспорт публикаций в CSV
* Экспорт публикаций в Excel
* Выбор периода для экспорта
* Включение всех полей публикации в экспорт

5. Чек-лист базы данных

5.1. Целостность данных

* Связи между таблицами
* Уникальность записей
* Корректность типов данных
* Ограничения

5.2. Производительность БД

* Индексы на часто используемых полях
* Время выполнения запросов
* Блокировки и deadlock'и
* Использование кеша Redis

5.3. Резервное копирование

* Ежедневное автоматическое бэкапирование
* Восстановление из бэкапа
* Хранение бэкапов 3 месяца

6. Чек-лист безопасности

6.1. Защита данных

* Защита от SQL-инъекций
* Валидация пользовательского ввода
* Шифрование контактной информации (PGP)
* Безопасное хранение паролей

6.2. Контроль доступа

* Разделение ролей (администратор/модератор)
* Проверка прав доступа для каждого действия
* Ограничение доступа к админке по IP
* Сессионное управление

6.3. Мониторинг безопасности

* Логирование действий администраторов
* Логирование ошибок в Sentry
* Мониторинг подозрительной активности
* Регулярные проверки безопасности

7. Чек-лист производительности

7.1. Отзывчивость системы

* Время отклика бота на команды < 1 секунды
* Время загрузки страниц админки < 2 секунды
* Время парсинга одного источника < 5 минут
* Время выполнения запросов к БД

7.2. Нагрузочное тестирование

* Поддержка 5000+ одновременных пользователей
* Обработка 100+ новых вакансий ежедневно
* Работа при пиковых нагрузках
* Стабильность при длительной работе

7.3. Масштабируемость

* Добавление новых источников без переписывания кода
* Увеличение количества пользователей
* Рост объема данных
* Горизонтальное масштабирование

8. Чек-лист интеграции

8.1. Взаимодействие компонентов

* Парсер → БД (сохранение данных)
* Планировщик → Парсер (запуск по расписанию)
* Бот → БД (поиск и подписки)
* Административная панель → Бот (ручная рассылка)
* Бот → Telegram API (отправка сообщений)

8.2. Внешние интеграции

* Telegram Bot API
* Sentry (логирование ошибок)
* Grafana (метрики и мониторинг)
* Внешние API источников

9. Чек-лист юзабилити

9.1. Telegram-бот

* Понятность сообщений и команд
* Удобство inline-кнопок
* Ясность инструкций для новых пользователей
* Отсутствие технического жаргона
* Единообразие интерфейса

9.2. Административная панель

* Интуитивная навигация
* Понятные labels и подписи
* Удобство фильтрации и поиска
* Отзывчивость интерфейса
* Единый стиль всех элементов

10. ЧЕК-ЛИСТ РЕГРЕССИОННОГО ТЕСТИРОВАНИЯ

10.1. Критические сценарии

* Регистрация и аутентификация пользователей
* Подписка на уведомления
* Поиск публикаций
* Модерация контента
* Парсинг основных источников
* Рассылка уведомлений

10.2. Интеграционные тесты

* Полный цикл: парсинг → модерация → рассылка
* Обновление публикаций и уведомления
* Управление пользователями и правами доступа
* Экспорт и импорт данных

# Тест-кейсы

* 1. Тест-кейсы telegram-бота

| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат | Фактический результат | Статус |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TC-BOT-001 | Регистрация нового пользователя | Высокий | Пользователь ранее не взаимодействовал с ботом | 1. Отправить команду /start 2. Проверить ответ бота | - Приветственное сообщение с описанием возможностей | Ответ бота:  👋 Привет! Я помогу найти вакансии, стажировки, конференции и хакатоны.  📝 Подписка: /subscribe (или кнопка «Подписаться»)  🚫 Отписка: /unsubscribe  📋 Предпочтения: /preferences  🔎 Поиск: /jobs /internships /conferences /contests  Нажмите нужную кнопку ниже или введите команду. | Успешно |
| TC-BOT-002 | Просмотр справки по командам | Высокий | Пользователь зарегистрирован через /start | 1. Отправить команду /help 2. Проверить отображение inline-кнопок | - Отображается список всех доступных команд - Присутствуют inline-кнопки основных команд - Описание каждой команды понятное и на русском языке | Ответ бота:  ℹ️ Справка  • /subscribe [теги] — подписка (без аргументов запустит мастер)  • /unsubscribe [теги] — отписка (без аргументов запустит мастер)  • /preferences — показать сохранённые теги  • /jobs /internships /conferences /contests [теги] — быстрый поиск  📌 Обязательные поля: типы публикаций (jobs / internships / conferences / contests).  👉 Для jobs/internships добавьте:  • Сфера (occupation): разработчик, тестировщик, аналитик…  • Платформа или язык (platform/language): ios, android, python, csharp…  ➕ Опционально: формат (remote/office/hybrid), график (part-time/full-time), город. | Успешно |
| TC-BOT-003 | Подписка на теги через команду | Высокий | Пользователь зарегистрирован | 1. Отправить /subscribe jobs python remote москва 2. Проверить ответ бота 3. Выполнить /preferences | - Бот подтверждает подписку - Теги сохранены в БД в таблице предпочтений - /preferences показывает добавленные теги с группировкой по категориям |  |  |
| TC-BOT-004 | Подписка через inline-кнопки | Средний | Пользователь зарегистрирован | 1. Нажать inline-кнопку "Подписаться" 2. Выбирать категории тегов пошагово 3. Завершить настройку | - Пошаговый интерфейс выбора тегов по категориям - Предпочтения сохраняются корректно - Существующие настройки дополняются, а не заменяются | Кнопка «Подписаться» в меню работает. Бот последовательно предлагал выбрать категории (Языки, Локация, Тип работы). | Успешно |
| TC-BOT-005 | Поиск вакансий без параметров | Высокий | В БД есть минимум 10 вакансий | 1. Отправить команду /jobs 2. Проверить результат | - Возвращается 10 последних вакансий из БД - Каждая вакансия содержит: заголовок, ссылку, описание, теги - Форматирование сообщения читаемое | Бот прислал 1 сообщение с 10 вакансиями.  Каждая вакансия содержит:  1. Заголовок  2. Ссылка  3. Наименование работодателя  Время ответа: ~0.8 сек | Успешно |
| TC-BOT-006 | Поиск с фильтрацией по тегам | Высокий | В БД есть вакансии с тегами "python" и "remote" | 1. Отправить /jobs #python #remote 2. Отправить /jobs python remote 3. Сравнить результаты | - Оба формата возвращают одинаковые результаты - Возвращаются только вакансии, содержащие указанные теги - Количество результатов может быть меньше 10 |  |  |
| TC-BOT-007 | Отписка от конкретных тегов | Средний | Пользователь подписан на теги "python", "remote", "москва" | 1. Выполнить /unsubscribe python 2. Проверить /preferences 3. Убедиться в получении уведомлений | - Тег "python" удален из предпочтений - Уведомления приходят только по оставшимся тегам - Бот подтверждает отписку |  |  |
| TC-BOT-008 | Получение уведомления о новой вакансии | Высокий | Пользователь подписан на тег "python", в системе появилась новая вакансия | 1. Дождаться времени рассылки 2. Проверить входящие сообщения | - Приходит уведомление о новой вакансии - Сообщение содержит все обязательные поля - Форматирование корректное |  |  |
| TC-BOT-009 | Напоминание о дедлайне конференции | Средний | В БД есть конференция с дедлайном через 3 дня, пользователь подписан | 1. Дождаться времени отправки напоминаний 2. Проверить получение сообщения | - Приходит напоминание за 3 дня до дедлайна - Сообщение содержит пометку о дедлайне - Указана точная дата окончания подачи заявок |  |  |

2. Тест-кейсы парсеров

| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат |
| --- | --- | --- | --- | --- | --- |
| TC-PARSER-001 | Успешный парсинг источника | Высокий | Источник доступен, агент активен | 1. Запустить парсер вручную 2. Проверить логи выполнения 3. Проверить данные в БД | - Парсинг завершен без ошибок - Данные сохранены в таблицу публикаций |
| TC-PARSER-002 | Обработка ошибки парсинга | Высокий | Источник недоступен | 1. Запустить парсер 2. Проверить логи через 15 минут 3. Проверить логи через 45 минут | - Ошибка залогирована в Sentry - Повторная попытка через 15 минут - Третья попытка через 45 минут - В БД записан неуспешный результат |
| TC-PARSER-003 | Обнаружение дубликатов | Высокий | В БД уже есть вакансия с таким же названием, компанией и датой | 1. Запустить парсер с данными, содержащими дубликат 2. Проверить записи в БД | - Дубликат не сохранен в БД - Количество новых записей = общее количество - дубликаты - В логах указано о пропуске дубликатов |
| TC-PARSER-004 | Автоматическое извлечение тегов | Средний | В БД есть предопределенные теги, в тексте вакансии есть ключевые слова | 1. Запустить парсер с вакансией, содержащей "Python", "Remote", "Moscow" 2. Проверить связи публикация-тег в БД | - Теги "python", "remote", "moscow" автоматически присвоены публикации - Теги взяты из предустановленных значений БД - Поиск тегов выполняется по подстроке |
| TC-PARSER-005 | Производительность парсинга | Средний | Источник содержит 50+ вакансий | 1. Засечь время начала парсинга 2. Запустить парсер 3. Засечь время завершения | - Время выполнения < 5 минут - Все данные сохранены корректно - Потребление памяти в пределах нормы |

3. Тест-кейсы планировщика

| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат |
| --- | --- | --- | --- | --- | --- |
| TC-SCHED-001 | Запуск парсеров по расписанию | Высокий | Настроено расписание парсинга вакансий на 12:00 МСК | 1. Дождаться 12:00 МСК 2. Проверить логи планировщика 3. Проверить логи парсеров | - Планировщик запустил все активные парсеры вакансий - Парсеры выполнили свою работу - Данные обновлены в БД |
| TC-SCHED-002 | Автоматическая рассылка публикаций | Высокий | Есть неотклоненные публикации, настроено время рассылки | 1. Дождаться времени рассылки 2. Проверить отправленные сообщения в боте  3. Проверить публикации в Telegram-канале | - Все неотклоненные публикации отправлены подписчикам - Публикации отправлены в Telegram-канал - Отклоненные публикации не рассылаются |
| TC-SCHED-003 | Удаление старых записей | Средний | В БД есть записи старше 90 дней | 1. Запустить задание очистки вручную 2. Проверить записи в БД | - Записи старше 90 дней удалены  - Записи младше 90 дней остались в БД - Связанные данные также очищены |
| TC-SCHED-004 | Рассылка уведомлений [UPD] | Средний | Публикация отредактирована через админку | 1. Отредактировать публикацию в админке 2. Дождаться следующего цикла рассылки 3. Проверить уведомления | - Отправлено уведомление с пометкой - Содержание соответствует отредактированной версии - Подписчики получили обновление |

4. Тест-кейсы административной панели

| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат |
| --- | --- | --- | --- | --- | --- |
| TC-ADMIN-001 | Успешная аутентификация | Высокий | Существует учетная запись администратора | 1. Перейти на страницу входа 2. Ввести корректные логин/пароль 3. Ввести код 2FA 4. Проверить доступ к панели | - Успешный вход в систему - Отображение главной панели администратора - Создана защищенная сессия |
| TC-ADMIN-002 | Создание пользователя модератора | Высокий | Администратор авторизован в системе | 1. Перейти в раздел "Пользователи" 2. Нажать "Создать пользователя" 3. Заполнить данные, выбрать роль "Модератор" 4. Сохранить | - Новый пользователь создан - Сгенерирован временный пароль - Роль "Модератор" назначена корректно - Пользователь может войти с указанными данными |
| TC-ADMIN-003 | Просмотр и фильтрация публикаций | Средний | В БД есть публикации разных типов и дат | 1. Перейти в раздел "Публикации" 2. Применить фильтр по типу "вакансии" 3. Применить фильтр по дате 4. Применить фильтр по тегам | - Отображается список публикаций - Фильтрация работает корректно - Пагинация отображается при большом количестве записей - Основная информация по каждой публикации видна |
| TC-ADMIN-004 | Модерация публикации | Высокий | Есть непромодерированные публикации, пользователь с ролью модератора | 1. Войти как модератор 2. Открыть публикацию для модерации 3. Отклонить публикацию с указанием причины 4. Проверить статус в БД | - Публикация помечена как отклоненная - Причина отклонения сохранена - Отклоненная публикация не участвует в рассылках |
| TC-ADMIN-005 | Редактирование публикации | Средний | Есть существующая публикация | 1. Открыть публикацию для редактирования 2. Изменить заголовок и текст 3. Сохранить изменения 4. Проверить рассылку уведомления [UPD] | - Изменения сохранены в БД - При следующей рассылке отправлено уведомление |
| TC-ADMIN-006 | Ручная рассылка публикации | Средний | Есть публикация, не отправленная в рассылке | 1. Открыть публикацию 2. Нажать "Разослать вручную" 3. Проверить получение сообщения подписчиками | - Публикация отправлена всем подписчикам - Сообщение появилось в Telegram-канале - В логах зафиксирована ручная рассылка |
| TC-ADMIN-007 | Просмотр статистики пользователей | Низкий | В БД есть данные о пользователях за разные периоды | 1. Перейти в раздел "Статистика" 2. Выбрать период (например, последние 30 дней) 3. Проверить отображение метрик | - Отображается количество подписавшихся/отписавшихся - Показывается разница (прирост/убыль) - Отображается общее количество активных пользователей - Данные соответствуют реальным данным в БД |
| TC-ADMIN-008 | Экспорт данных в CSV | Средний | В БД есть публикации за выбранный период | 1. Перейти в раздел "Экспорт" 2. Выбрать период 3. Выбрать формат CSV 4. Скачать файл | - Файл скачивается корректно - Содержит все поля публикаций - Данные в файле соответствуют данным в БД - Форматирование CSV корректное |
| TC-ADMIN-009 | Управление тегами | Средний | Администратор авторизован | 1. Перейти в раздел "Теги" 2. Добавить новый тег в категорию "technology" 3. Редактировать существующий тег 4. Удалить неиспользуемый тег | - Новый тег сохранен в БД - Изменения применены корректно - Удаление тега не ломает существующие связи |
| TC-ADMIN-010 | Настройка расписания парсинга | Низкий | Администратор авторизован | 1. Перейти в раздел "Источники" 2. Изменить расписание парсинга для конкретного источника 3. Сохранить изменения 4. Проверить выполнение по новому расписанию | - Расписание сохранено в БД - Планировщик использует новое расписание - Парсинг запускается в указанное время |

5. Тест-кейсы безопасности

| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат |
| --- | --- | --- | --- | --- | --- |
| TC-SEC-001 | Защита от SQL-инъекций | Высокий | Система развернута в тестовой среде | 1. В поле поиска ввести: ' OR '1'='1 2. В поле логина ввести SQL-  инъекцию 3. Проверить ответ системы | - Запрос отклонен или экранирован - В логах нет ошибок БД - Система не возвращает лишних данных |
| TC-SEC-002 | Проверка 2FA | Высокий | Для пользователя включена 2FA | 1. Войти с корректным логином/паролем 2. Ввести неверный код 2FA 3. Ввести корректный код 2FA | - С неверным кодом вход запрещен - С корректным кодом вход разрешен - После нескольких неудачных попыток возможна блокировка |
| TC-SEC-003 | Проверка прав доступа | Высокий | Существуют пользователи с ролями администратор и модератор | 1. Войти как модератор 2. Попытаться получить доступ к разделам "Пользователи", "Статистика" 3. Проверить доступ к API endpoints | - Модератор не имеет доступа к разделам не своей компетенции - API возвращает 403 Forbidden для запрещенных операций - В логах фиксируются попытки несанкционированного доступа |

6. Тест-кейсы производительности

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат |
| TC-PERF-001 | Время отклика бота | Высокий | Система под нагрузкой | 1. Отправить команду /start 2. Замерить время ответа 3. Повторить для основных команд | - Время отклика ≤ 1 секунды для всех команд - Отсутствие таймаутов - Стабильная работа при последовательных запросах |
| TC-PERF-002 | Нагрузка 100 одновременных пользователей | Средний | Настроена нагрузочная среда | 1. Запустить 100 параллельных запросов к боту 2. Мониторить метрики производительности 3. Проверить логи ошибок | - Система обрабатывает все запросы - Время отклика в пределах нормы - Отсутствие ошибок 5xx |

7. Тест-кейсы интеграции

| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат |
| --- | --- | --- | --- | --- | --- |
| TC-INT-001 | Полный цикл публикации | Высокий | Все компоненты системы работают | 1. Парсер получает новую вакансию 2. Модератор проверяет и одобряет публикацию 3. Планировщик рассылает публикацию 4. Пользователь получает уведомление | - Все этапы выполняются корректно - Данные передаются между компонентами без потерь - Пользователь получает актуальную информацию |
| TC-INT-002 | Интеграция с Sentry | Средний | Настроено подключение к Sentry | 1. Сгенерировать ошибку в парсере 2. Проверить поступление ошибки в Sentry 3. Проверить корректность метаданных | - Ошибка залогирована в Sentry - Стек трейс корректен - Присутствует контекст выполнения |

8. Тест-кейсы юзабилити

| ID | Название | Приоритет | Предусловия | Шаги | Ожидаемый результат |
| --- | --- | --- | --- | --- | --- |
| TC-UX-001 | Удобство интерфейса админки | Низкий | Административная панель доступна | 1. Проверить навигацию между разделами 2. Проверить понятность labels и подсказок 3. Оценить удобство фильтрации | - Интерфейс интуитивно понятен - Навигация логичная - Подписи и сообщения понятны на русском языке |