**Проект 3-го спринта: задание**

Тебе предстоит протестировать мобильное приложение и API. Как и в прошлых спринтах, результаты заданий нужно поместить в гугл-док и прикрепить ссылку в тренажёр. Для обоих заданий пригодится [шаблон](https://docs.google.com/spreadsheets/d/1my1Bq2H6Kz-gAXvfLXJQ8Nx5TXrcFXu6zRPbME3UND0/edit#gid=2006427015).

**Мобильное приложение**

Команда сделала рефакторинг приложения под Android. Чтобы выпустить новую версию, нужно протестировать те части продукта, которых коснулись изменения. Тестовой документации ещё нет, поэтому её нужно написать.

**Постановка задачи**

1. Проанализируй требования к мобильному приложению Яндекс.Метро из предыдущего урока.
2. Напиши чек-лист для тестирования мобильного приложения на часть требований, **выделенных жирным шрифтом**. При необходимости визуализируй требования через mindmap или блок-схемы. Визуализацию сдавать не нужно.
3. Протестируй мобильное приложение в эмуляторе с помощью Android Studio или на своём Android-устройстве и заведи баг-репорты в Яндекс.Трекере: очередь «BUG: Баг-репорты». Скачай готовящуюся к релизу версию приложения [тут](https://code.s3.yandex.net/qa/files/yandexmetro-android-v3.6.apk).

Чтобы проверить, что обновление происходит корректно, скачай предыдущую версию [тут](https://code.s3.yandex.net/qa/files/yandexmetro-android-v2.1.apk). Вспомни, как правильно это сделать: нужно установить предыдущую версию приложения, а затем обновить её на новую.

В процессе тестирования отмечай результаты выполнения теста: PASSED или FAILED. Если тест со статусом FAILED, заведи баг-репорт в Яндекс.Трекере в [очереди BUG](https://tracker.yandex.ru/bug) и вписывай ID в соответствующую таблицу результатов.

1. Напиши отчёт о тестировании (\*). Что можешь рассказать команде о статусе протестированной части продукта?

**(\*) — задача со звёздочкой** — задание, которое можно не сдавать. Оно не повлияет на результат итогового проекта.

https://code.s3.yandex.net/qa/schemes/project-roz.png

**API**

Разработчики сделали новую функциональность в API Яндекс.Прилавка. Новую версию API передали тебе на тестирование.

Запустить сервер

Изучи новую функциональность.

**Работа с наборами:** возможность добавлять продукты в набор — ручка POST /api/v1/kits/{id}/products.

**Работа с курьерами:** возможность проверить, есть ли доставка курьерской службой «Привезём быстро» и сколько она стоит. Ручка POST /fast-delivery/v3.1.1/calculate-delivery.xml.

**Работа с корзиной:**

* возможность получить список продуктов, которые добавили в корзину. Ручка GET /api/v1/orders/:id;
* возможность добавлять продукты в корзину. Ручка PUT /api/v1/orders/:id;
* возможность удалять корзину. Ручка DELETE /api/v1/orders/:id.

**Постановка задачи**

1. Проанализируй требования к новой функциональности бэкенда Яндекс.Прилавка. Изучи документацию к API в Apidoc. Требования к бэкенду находятся [здесь](https://code.s3.yandex.net/qa/files/backend_requirements.pdf).
2. Спроектируй тесты в виде чек-листа, чтобы покрыть функциональность, которую тебе передали на тестирование: она описана выше. Авторизацию проверять не нужно.
3. Чек-лист помести в гугл-таблицу. Создай копию [шаблона](https://docs.google.com/spreadsheets/d/1my1Bq2H6Kz-gAXvfLXJQ8Nx5TXrcFXu6zRPbME3UND0/edit#gid=2006427015) и открой доступ на комментирование по ссылке.
4. Протестируй API через Postman и заведи баг-репорты в Трекере, если это понадобится.
5. Напиши отчёт о тестировании (\*). Что ты можешь рассказать команде о статусе протестированной части продукта?

**(\*) — задача со звёздочкой** — задание, которое можно не сдавать. Оно не повлияет на результат итогового проекта.

https://code.s3.yandex.net/qa/schemes/project-roz.png

**Как оформить работу**

Создай копию [шаблона](https://docs.google.com/document/d/16mMZpxpV4RyQTOMRxfMgBdfGdD9cKFYSJg-xIAKMrY8/edit?usp=sharing) и помести туда результаты заданий. Открой доступ на комментирование и прикрепи ссылку в тренажёр — во вкладку «Ревью». Вкладка появится в следующем уроке.

**Дождись проверки и комментариев**

Ревьюер проверит задания в течение 96 часов. Если будут замечания, проект вернётся тебе на доработку. Доработать задания и прислать их на проверку ещё раз можно в течение 48 часов.

**Как будут проверять твой проект:**

* Насколько хорошо оформлены чек-листы и насколько полные проверки.
* Насколько качественно проведено тестирование: все ли спроектированные проверки действительно выполнены.
* Какие найденные баги заведены и насколько корректно они оформлены.