

Полезности для БА

- Полезная инфа
- Стек технологий:
- Куда нужны доступы: - за всеми доступами к Виталику
- Необходимой софт или понимание технологий :
- Рекомендации по заведению задачи:
- Жизненный цикл задачи
- Команды с которыми взаимодействуем:
- Основные Бизнес Заказчики
- Отпуска
- Выход нового сотрудника в команду:

Полезная инфа

1. <https://api.rozetka.com.ua/v1.1/help/> - Апи овокс.
2. <http://wiki.sd.local/pages/viewpage.action?pageId=26445033> схема Гомер и другие команды
3. it.po@rozetka.com.ua - куда писать на счет монолита и спорных вопросов если не знаешь куда обратиться))
4. <https://sites.google.com/a/owox.com/rozetka/> - доки овокс
5. Работа с Календарем - [Работа с календарями в почтовых клиентах](#)
6. http://splitter.staging.com.ua/gomer/rabbitmqmanagement/queues/index?sort=-messages_ready - очередь - Cron - их описание
7. [SCRUM_Schedule_Go](#) - Календарь скрам встреч.
8. Ретроспективы - [SCRUM_Ретроспективы_Go](#)
9. Список приоритетов по Гомеру - https://docs.google.com/spreadsheets/d/1f3wDpXJcFy_Q9htRDBVnPPtXFuGsNI_uREIq6i_mpVU/edit#gid=1494078221
10. [admin.rozetka.com.ua](#) - Админка овокса, на прод заходим под любым браузером. Для теста нужно использовать **Waterfox** - по должны прописаны тестовые тачки в хостах
11. Статистика созданных заявок на СД - <http://sd.local/projects/SUPPORT/reports/custom/266/timescale/2>
12. Основной канал коммуникации с командой **Телеграмм**, многие заказчики используют **Скайп**, Также проверяйте регулярно **Джабер** могут приходить обращения. Также после каждой встрече с Бизнесом необходимо **отправлять отчеты по встречам** всем участникам встречи.

Стек технологий:

- git version - ~2.14.1 - система контроля версий
- mySql - ~5.7.22 - сплиттера (хранятся роли пользователей)
- postgresql - >=9.6.9 - Основная БД Гомера
- php и производные либы - 7.1 - Основной язык программирования
- Redis server v=4.0.1 - Кеширование
- rabbit,"RabbitMQ", "3.6.15" - очередь
- nginx/1.12.1 - сервер
- **Docker** - контейнеры для развертывания проектов, тоже будет использоваться для CI/CD
- Jenkins - CI/CD

Куда нужны доступы: - за всеми доступами к Виталику

- Тестовая База Gomer
- Доступ на тестовый Овокс
- Для Сплиттера доступ тестовая база Сплиттера
- Доступ к БД тестового стора 13 тачка.

Необходимой софт или понимание технологий :

- **ДО:** Jira
- **Документация:** Confluence
- Для наглядности можно использовать стандарты, которые используются для описания бизнес-процессов: **BPMN, UML**
- **Построение схем:** <https://www.draw.io/>
- **Для работы с базой данных:** Navicat
- **Для работы с АПИ:** Postman
- **Просмотр кода:** Notepad++
- **Справочник по SQL:** <http://2sql.ru/> <https://postgrespro.ru/>
- **PHP, JSON, XML, REST API, GraphQL, Типы запросов API, CI/CD, Unit tests, Функциональные тесты, Smoke tests.**
- **Zabbix3.local** - для просмотра нагрузки на сервер.
- **Swagger** - API Документатор

- Также стремимся к **микросервисной** архитектуре. Почитать для что это такое.

Рекомендации по заведению задачи:

- **Заголовок** - Должен быть понятен и отражать суть задачи
- Каждая задача должна содержать **Цель** задачи или **Проблему** которая она решает.
- Далее указываем решение задачи (**описание**) - по описанию важно описать понятно для того человека который в первый раз видит систему.
- Также в конце пишем **DoD** или **ожидаемый результат**. В данной части мы должны будет описать кейсы, ожидаемое поведение системы после выполнения задачи.
- Желательно в задачи добавлять **Макеты интерфейсов или схемы бизнес процессов (UML)** - (это любят разработчики и это полезно для своего понимания)
- В каждую задачу необходимо добавить **Epic**. Если его нет создать новый.
- Оценки в **Dev story points** - Необходимо добавлять оценки разрабов, в **QA story points** - оценки тестировщиков, после чего в **Estimate** добавляем общую оценку. - **У всех плановых задача должны быть оценки.**
- Обязательно менять статусы задач, Задачи когда уходят на разрабов должны быть в статусе **Selected for development**
- Для пожарных задач указываем **критический** приоритет
- Также в заявках можно использовать метки. Чтобы можно было легко найти задачу.
- Обязательно указывать комменты к задачам, если что-то меняется в процессе оценок или выполнения. - Все больше задача должны быть в конfluence.
- Если задача пришла с **тех саппорта**, как доработка. Необходимо в задаче тех. саппорта, нажать "**Создать связанную задачу**". После чего в тело основной задачи, добавить наше описание. Также текст обращения оставить в конце задача в формате **Из обращения:** далее текст обращения
- По **спринтам** желательно сразу раскидывать задачи на несколько спринтов вперед, чтобы вся команда понимала куда мы движемся и какие задачи будут.
- В спринте должен быть хотя бы один **пожарник** - у него нет спринтовых задач. Задачи попадают в работу только запланированные **Баги или критические баги**.

Epic - Большая задача которая включает в себя группу задач.

Жизненный цикл задачи

- Получение требований или идеи от бизнеса (может быть собственное предложение по улучшению)
- Согласование требований с бизнесом - Описание задачи в Конфлюенсе.
- Разбитие описание на задачи по возможности (создавать в джире)
- После того как задача готова она приоритизируется
- Вычитка задачи с командой и оценка (если задача достаточно подробно описана)
- Добавляем задачу в спринт в статусе готова к разработки после чего у задачи следующие статусы
 - 1.1 **В работе**
 - 1.2 **Ждет ревью** - 2 разработчика с команды обязаны проверить код задачи.
 - 1.3 **Проходит ревью** - в данном статусе другие разработчики делают замечания по реализации задачи, замечания исправляются
 - 1.4 **Ожидает тестирования**
 - 1.5 **В тестирование** - Один из тестировщиков тестирует задачу
 - 1.6 **В Деве** - задача попадает после тестирования - тут она ждет тестирования в сборке
 - 1.7 **Тест релиза** - Тестирование всех задач которые были сделаны в спринте, также тест всего продукта. Регрессия
 - 1.8 **В мастере** - Задача готова к выливке
 - 1.9 **Выполнено** - задача вылита на прод
- Разраб должен **сам следить за статусом задачи**. Назначать в джире **ревьюеров**, также тестировщик который взял задачу в тест должен **отметится в задаче как тестер**.
- **Пожарные задачи** - Разработчики откалываются от **мастера**, также задача пропускает такие этапы как: В деве, Тест Релиза. - **Может быть вылита вне очереди**

Команды с которыми взаимодействуем:

- **Market Core/Front** - ЛК продавца, заказы, продавцы, АПИ для обновления инфы по товарам. **РО** - Ухалков Вова
- **Splitter Core** - Гомер как модуль сплиттера, + **Splitter** content выливается только через сплиттер (единая БД) - **ПО Качалина Женья**
- **Store** - реплика Овокса, используем как основную БД для **XL**, Все справочники синх со стора для Сплиттера и Гомера. Нужно переводить вывод всей инфы и выгрузок на стор, он работает быстрее чем монолит. - **ПО Гаврилов Саша**
- **Монолит (Овокс)** - Обращаемся по всем проблемам на почту it.ro или обсуждаем с **Максом Хуснутдиновым или Тарабан Игорь**
- **Goods** - Команда разрабатывает АПИ для взаимодействия сущностями товаров (Товары, опции, категории, производители, значения опций, Серии, Мерчанты, СЛА) После ухода от монолита будем использовать их **АПИ**. **ПО** - **Цыток Илья**
- **CRM/Аналитика** - Внутренняя аналитика, Аналитика продавца, DATA Science. **ПО** - **Наташа Кожанова**

Основные Бизнес Заказчики

- Каракулин Артем - Директор Маркет плейса

- Сарапин Артем - Операционный директор (Зам. Каракулина)
- Анна Бевз - Руководитель контент отдела (главный бизнес заказчик направления)
- Кулеш Татьяна - Руководитель контента МП.
- Соболева Дарья/Маевская Марина - Руководители поддержки МП
- Проценко Людмила - Руководитель отдела продаж
- Хроменко Максим - Руководитель отдела развития (развивают продажи текущих продавцов)
- Чистяков Антон/Лаврик Александр - Менеджеры МДМ - будут ставить задачи интерфейсной части, + коммуникация с Гудс сервисом
- Комышан Павел/Брыленко Николай - Коммерческие директора Розетки. (больше для СПЛ контента)
- Столяр Антон - Аналитик коммерческого отдела. (больше для СПЛ контента)
- Другие команды с которыми идет взаимодействие. (описаны выше)
- Расторопова Ира - CEO отдел

Отпуска

Продолжительность отпуска 21 день (включая выходные) + все гос. праздники также выходные в Розетке. На счет отпусков 1 неделю - предупреждать за 1-2 недели Виталика (отмечать в календаре). За отпуска 2 недели - предупреждать не позднее двух недель.

Отдела разработки - <https://docs.google.com/spreadsheets/d/1sfjYQb-RyPy7oOCSGktgUtqx68Ey1eIXbUw6Zc0V76Q/edit#gid=0>

Свои отпуска отмечать в календаре - [Перейти](#)

Выход нового сотрудника в команду:

- Необходимо убедиться, что создана заявка на оборудование или на пропуск. Уточнять у **Сергея Козко**
- Если учетка создана необходимо сотрудника добавить в Джиру и Конфлюенс. <http://sd.local/plugins/servlet/project-config/MC/roles> - где добавлять (с соответствующей ролью)
- Все заявки на оборудование или доступы пишет тим-лид и руководитель отдела. Человеку нужна флешка с **Убунтой** (операционная система), также доступы к **гит-лабу** и **апсурсу** (пишет **Данила**)
- Необходимо провести вводную инструкцию по процессу работы (скрам, встречи и так далее) правила в команде.
- Также потратить время чуток рассказать о системе вместе с тим-лидом. Пройтись по основному функционалу