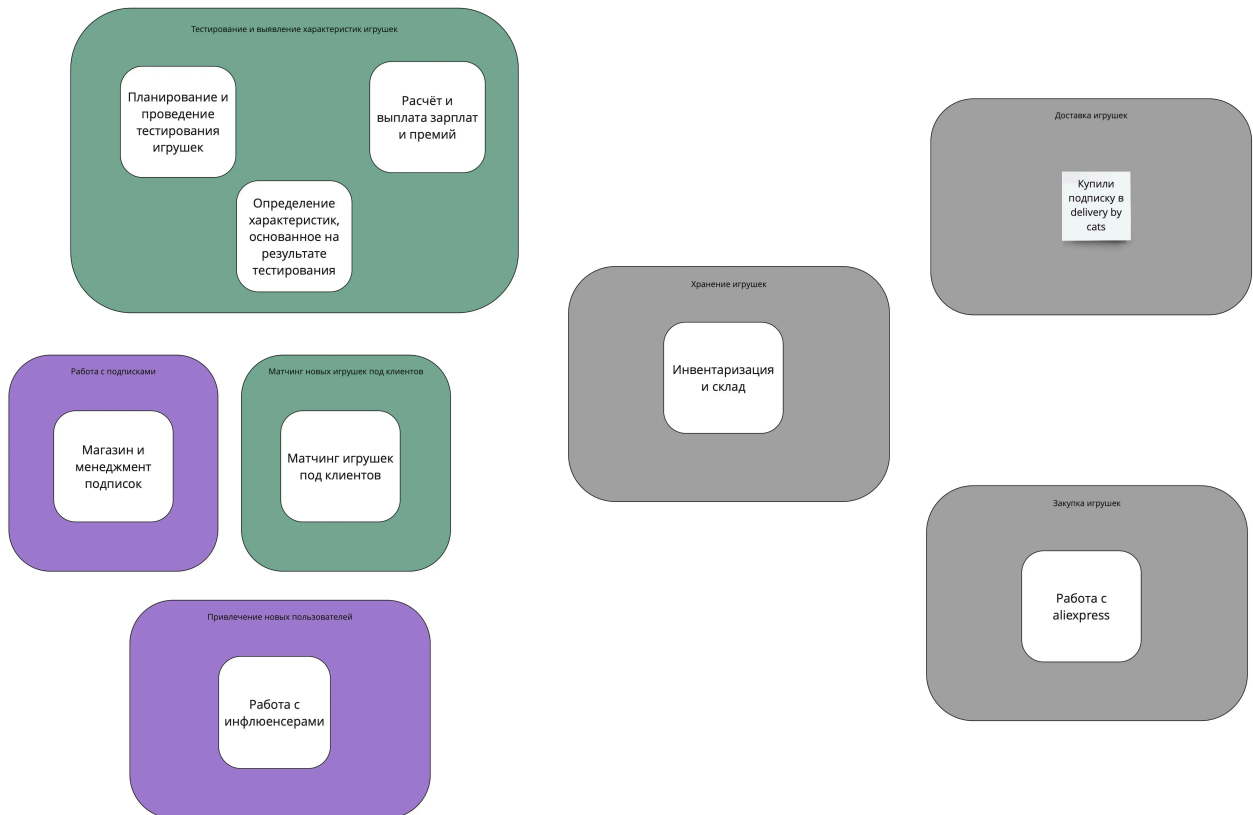


АС - Домашнее задание №2

TODO

- ✓ выпишите все поддомены, которые вы нашли в системе;
- ✓ текстом опишите логику, по которой были выбраны поддомены;
- ✓ определите все типы поддоменов и заполните core domain chart;
 - темплейты [можно найти по ссылке](#) (картинка и draw.io), для [«Миро» темплейт тут](#);
- ✓ опишите логику, по которой был выбран тот или иной тип поддомена (можно повторить таблицу из урока);
- ✓ определите боундед-контексты для каждого из поддоменов, основываясь на требованиях (хватит одной модели, как было в уроке);



Пример модели из урока, которой хватит, чтобы закрыть этот пункт

- ✓ сравните полученные боундед-контексты из поддоменов и боундед-контексты, полученные из ES. Опишите, что разошлось (можно показать на картинке в сравнении) и предположения, почему так получилось;
- ✓ сделайте исправленную версию ES-модели и модели данных, если боундед-контексты разошлись. Если не разошлись — приложите ES и модель данных из прошлого урока;
 - если нашли места, где бизнес-команда разбилась на технические шаги, — отметьте эти места на модели;
- ☐ выпишите характеристики, важные для проекта. В нашем случае мы не можем спросить у бизнеса, что важно, а что нет;
 - для каждой найденной характеристики укажите место, где она была взята;
- ☐ выберите один из четырёх архитектурных стилей, описанных в уроке. Опишите, почему вы сделали такой выбор и по каким характеристикам сравнивали стили (можно использовать картинку из урока со сравнением стилей);
- ☐ сделайте итоговую модель системы, укажите виды коммуникаций между элементами, если выбрали распределённый стиль.

Пройдитесь по чек-листу и проверьте, что не наступили на грабли:

- ✓ Найдённые поддомены описывают проблемы, решаемые бизнесом, а боундед-контексты — решения этих проблем. Это должно выражаться в названии: «магазин» не может быть поддоменом, а «продажа продукции» — может.
- ✓ Все названия поддоменов и боундед-контекстов описывают бизнес-проблемы и решения, а не их техническую реализацию.

Поддомены

Перечисление в порядке тайм-лайна по ES

- **Hiring** - найм котов-воркеров: обработка заявок кандидатов, проведение тестирования кандидатов, управление наборами тестов, анкетирование успешного кандидата на предмет сбора его характеристик
- **Task Tracking** - заведение заявок на услугу в системе и отслеживание её выполнения в системе. Также в нагрузку сюда решил добавить экспорт сотрудников из Happy Cat Box - слишком скучная для полноценного поддомена
- **Task Matching** - всё, что касается инновационной системы распределения котов-воркеров по заявкам на основе характеристик воркеров, а также оценки стоимости заявки на основе распределения.
- **Supplies** - снабжение расходниками и взаимодействие с пекарней печенек (частный вид расходника)
- **Betting** - ведение менеджерского тотализатора на основе задач и их статусов, распределение выигрышей по участникам и разработчикам MCF
- **Customer Accounting** - расчёты выплат котам-тестерам на основе стоимости услуг и скидок. Выставление счетов на оплату
- **Worker Accounting** - расчёты выплат котам-воркерам на основе проделанной работы с учетом штрафов и стоимости услуг. Проведение выплат (через сервис Golden Hat).
- **Quality Control** - исследования отменённых заявок (звонки, анкетирования), формирование гипотез по улучшению, проведение экспериментов
- **Notifications** - рассылка оповещений по эл. почте

Примечания

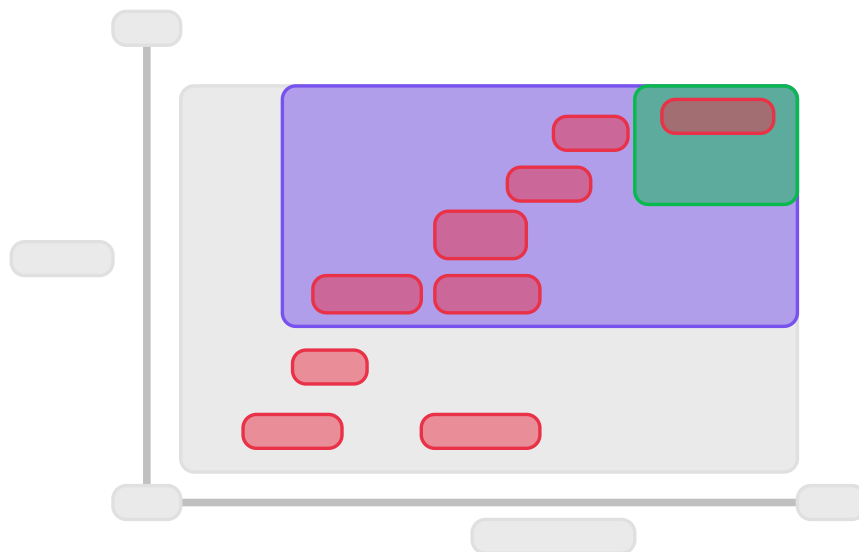
- Сбор характеристик при найме - это хитрое место, по факту здесь заведуют только транспортной частью - подготовить, послать анкету, за сам набор интересующих характеристик отвечают в Task Matching - им виднее, как длина хвоста влияет на выполнение тех или иных работ :)
- Поборол соблазн объединить в один поддомен все расчёт денег. Уж больно сильно разнятся эти два процесса в деталях: и отчетные периоды разные (неделя - месяц) и логика начислений\списаний разнится (штрафы против скидок). В одном случае нужно ещё и денежные переводы проводить, в другом наоборот - выставлять счета на оплату и т.д.
- Сайт найма должен уметь держать большую нагрузку и гибко реагировать на резкие скачки. Что предъявляет повышенные требования к команде работающей над Hiring (вряд ли outsource). Покупке готового решения препятствует завязка на данные для Task Matching - вряд коробочное решение будет уметь с ними правильно работать. Кастомизация готового продукта тоже менее предпочтительна - опасно раскрывать детали реализации инновационной части системы.

Классификация поддоменов

Поддомен	Преимущество	Изменчивость	Варианты реализации	Интерес решения проблемы	Тип
Hiring	нет	часто	inhouse, high-level outsource	средний	supporting
Task Tracking	нет	редко	in-house, покупка + кастомизация, outsource	редкая	supporting
Task Matching	да	часто	in-house only	высокий	core

Поддомен	Преимущество	Изменчивость	Варианты реализации	Интерес решения проблемы	Тип
Supplies	нет	редко	outsource, in-house	средний	supporting
Customers Accounting	нет	часто	покупка, outsource, in-house	низкий	supporting
Workers Accounting	нет	часто	покупка, outsource, in-house	низкий	supporting
Betting	нет	редко	покупка, outsource, in-house	низкий	generic
Quality Control	нет	редко	покупка + интеграция, outsource, in-house	низкий	generic
Notifications	нет	редко	покупка, outsource, in-house	низкий	generic

MCF - Domain Chart



Соображения

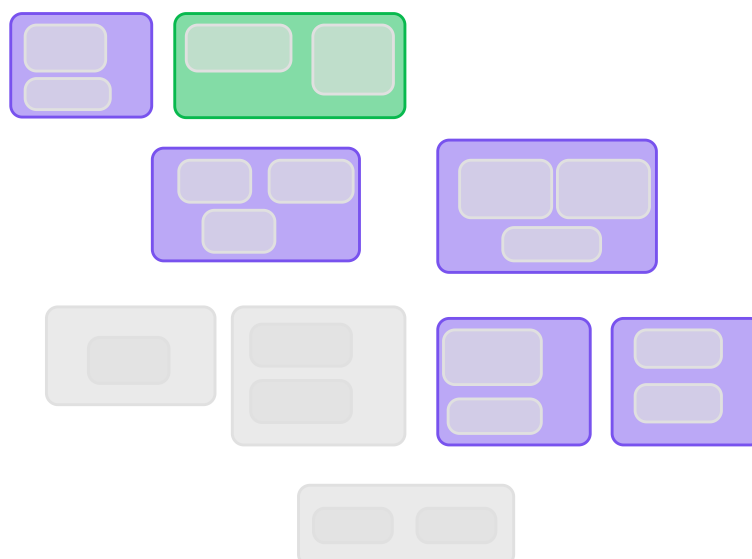
- В **Hiring** могут меняться тесты, описания вакансий, формы анкетирования кандидатов. Если вопрос контента можно было бы закрыть CMS или админкой, то новые типы тестов и анкет могут повлечь изменения в коде (добавление нового поведения). Особо интересных и сложных задач тут не предвидится, но и нетривиальные могут возникать изредка. Снабжает бизнес новым воркерами - supporting тип.
- **Task Tracking** - относительно не высокая сложность, каталог услуг и сайт для сотрудников и менеджеров видится мне основными местами для изменений, интеграция с Task Matching'ом не должна вызвать трудностей. Обеспечивает ход основного бизнеса
- **Task Matching** - постоянная калибровка точности распределения воркеров по задачам, уточнение характеристик, их состава, их качества и влияния на выполнение заявок, нетривиальность алгоритмов (рандом - это для MVP, на практике тут будет сложная система ранжирования, возможно даже на основе ML), это основное конкуретное преимущество MCF, должно разрабатываться строго в недрах компании - core
- **Supplies** - непонятно насколько часто меняется набор расходников, необходимый для услуги. Кардинально вряд ли будет меняться, но могут заменяться на лучшие аналоги, по мере появления на рынке новых продуктов и совершенствования научно-технического прогресса в целом. Кроме того,

добавление новой услуги в каталог Task Tracking будет влечь необходимость продумывать её снабжение расходниками. Дополнительную сложность создаёт необходимость интеграции с Пекарней печенек.

- **Customers Accounting, Workers Accounting** - раздумывал куда их всё-таки отнести. С одной стороны, расчёт и выставление инвойсов на основе правил (скидки\штрафы) - это распространённая задача, решаемая как готовыми продуктами\сервисами, так и легко пишущаяся на заказ на стороне, каких-то нетривиальных проблем на первый взгляд не прослеживается. Однако, есть ряд соображений, склоняющий меня к типу "supporting":
 - необходимость тесной интеграции с Task Tracking (отображение воркерам и клиентам их инвойсов, скидок, текущих балансов, штрафов на сайте трекера)
 - само управление скидками и штрафами тесно завязано на услуги - ещё одна связь с ТТ
 - необходимость интеграции с конкретной системой переводов платежей (из требований непонятно насколько Golden Hat распространена - возможно не каждый готовый биллинговый сервис может с ней интегрироваться) Именно, из-за этого WA на доменном чарте выше по шкале сложности реализации, чем CA.
- **Betting** - generic, почему: тотализатор не обеспечивает ход основного бизнеса. Вспомогательная активность, чтобы расшевелить менеджеров. Логика меняется редко. Не самая часто встречающаяся функция в системах, но готовые решения имеются:
 - <https://www.thebetengine.co.uk/>
 - <https://www.t2ge.eu/betting-engine/>
 - <https://www.openbet.com/products/bet/betting-engine>
 - в том числе и open source:
 - <https://pypi.org/project/sports-betting/>
 - <https://github.com/openbookie>
 - <https://github.com/topics/betting>
- **Quality Control** - как будто можно решить через интеграцию с готовой CRM-системой, звонки, история общения, экспорт тасок туда для обработки, опросы клиентов осуществлять чуть ли не Google Forms на первое время. Трекинг предложений по улучшению тоже можно через CRM вести (до тех пор пока дальше текстовых заметок дело не идёт). Пока склоняюсь к generic
- **Notifications** - тривиальнейшая задача рассылки сообщений по e-mail. SendGrid, MailChimp и т.п. Generic sub-domain.
-

Bounded Contexts

☐ MCF - Bounded Contexts



Что разошлось

- **Hiring flow** из ES разбилось на два контекста:
 - **Candidates testing** - всё, что связано с приёмом и обработкой заявок на устройство, здесь же и управление тестами (добавление новых, удаление неактуальных, готовые пресеты тестов)
 - **Newcomers export** - сбор характеристик, необходимых для Task Matching и передача в пулл воркеров в Task Tracking
- Самые драматичные изменения произошли в **Task lifecycle**:
 - был разбит на 4 поддомена - Task Matching (core), Task Tracking (supporting), Supplies (supporting), Quality Control (generic) - из-за разного характера и критичности влияния на бизнес
 - **Task Matching** обособлен ещё и из-за разницы в терминологии - "regular sample" вместо "cat-worker" и reference sample вместо "customer" ([US-290]). От термина "cat-tester" вообще пришлось отказаться, т.к. в данной системе ему нет применения. Само знание, что клиенты - это коты-тестеры ценности не добавляет и смысла не несёт.
 - Внутри **Task Matching** выделил
 - **Task Matching & Cost Estimation** - контекст для матчинга воркеров на задачи и оценку стоимости задачи на его основе. Будет скорее всего делать одна команда ML-специалистов
 - **Regular Sample Attributes discovery** - сбор и хранение характеристик Обычных Образцов + передача их в Hiring для анкетирования при приеме на работу
 - **Task Tracking** содержит контексты:
 - управления каталогом услуг
 - импорта клиентов из Happy Cat Box
 - отслеживания статусов задач
 - **Supplies** содержит контексты:
 - Закупки и хранения расходников на складе
 - Сборки расходников по конкретному заказу
 - Закупки печенек в Пекарне
 - **Quality Control** содержит контексты:
 - Коммуникации с клиентами (история звонков, содержимое разговоров)
 - Предложений по улучшению (учёт и результаты экспериментов по внедрению)
- **Billing flow** разделился на Workers Accountancy и Customers Accountancy (по уже описанным выше причинам)
 - **Workers Accountancy** содержит контексты:
 - выставления инвойсов и денежных переводов
 - учёта и настройки штрафов воркеров
 - **Customers Accountancy** содержит контексты:
 - выставления инвойсов клиентам
 - настройки системы скидок
- **Bets flow** остался без изменений, только переименовался в Bets processing
- **Notifications** - новый поддомен - на ES не был отражен. Здесь обнаружил два контекста в себе:
 - Непосредственная отправка и вся операционка по ней - очереди на отправку, ретраи, архивация, обход спам-фильтров
 - Branding - дизайн оформлений писем в едином стиле MCF, шаблоны писем для разных событий
 - Всё это имеет смысл, если будет выбран вариант in-house разработки вместо внедрения готового решения