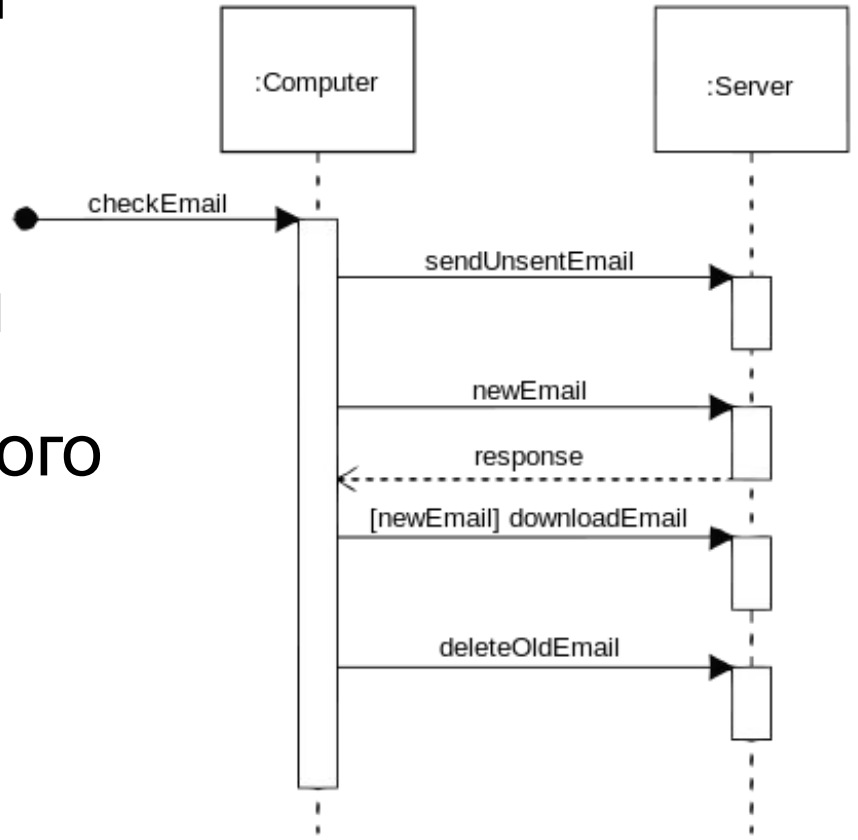
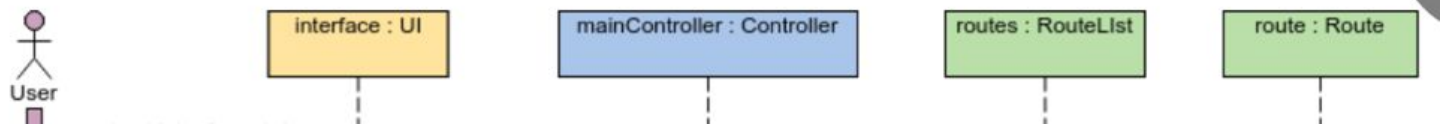
Диаграммы последовательностей

- Диаграммы последовательности следует применять тогда, когда требуется посмотреть на поведение нескольких объектов в рамках одного прецедента.
- Диаграммы последовательности хороши для представления взаимодействия объектов, но не очень подходят для точного определения поведения.
- На диаграмме показаны экземпляры объектов и сообщения, которыми обмениваются объекты в рамках одного прецедента (use case).

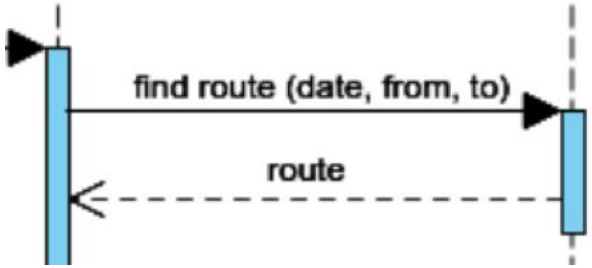


Линия жизни



- Диаграммы последовательности показывают взаимодействие, представляя каждого участника вместе с его линией жизни (lifeline), которая идет вертикально вниз и упорядочивает сообщения на странице.
- Сообщения также следует читать сверху вниз.
- Объекты подписываются в формате «объект:класс» и могут изображаться как в виде обычных прямоугольников, так и с использованием дополнительных обозначений, например, элемента Актер.

Сообщения



- Сообщения отражают взаимодействие (в том числе вызов функций) и изображаются горизонтальными стрелками, концы которых располагаются на линиях жизни.
- Стрелки направлены от отправителя к адресату и упорядочены по времени с помощью линии жизни.
- Также объекту может приходить ответ на сообщение, или Return Message, указывающий на завершение обработки предыдущего сообщения и его результат.

Диаграмма последовательностей: типы сообщений

- **Синхронное сообщение** — актор-отправитель передаёт ход управления актору-получателю, которому необходимо провести в прецеденте некоторое действие. Пока проводимое актором-получателем действие не будет завершено (соответственно, не будет получено ответное сообщение), актор-отправитель теряет возможность производить какие-либо действия. Графически изображается как стрелка с закрашенным треугольником, после которой идёт прямоугольник, отражающий деятельность объекта, в конце которого находится ответное сообщение.

- **Ответное сообщение** — данное сообщение является ответом на синхронное сообщение. Обычно, содержит какое-либо возвращаемое изначальному актору-отправителю значение, также возвращающее ему управление (возможность действовать).

- **Асинхронное сообщение** — актёр-отправитель передаёт ход управления актору-получателю, которому необходимо провести в прецеденте некоторое действие. Основное отличие от синхронного сообщения состоит в том, что актор-отправитель не теряет возможности совершать другие действия.

Диаграммы последовательностей

Диаграммы последовательностей используются для уточнения диаграмм вариантов использования, более детального описания логики сценариев использования.

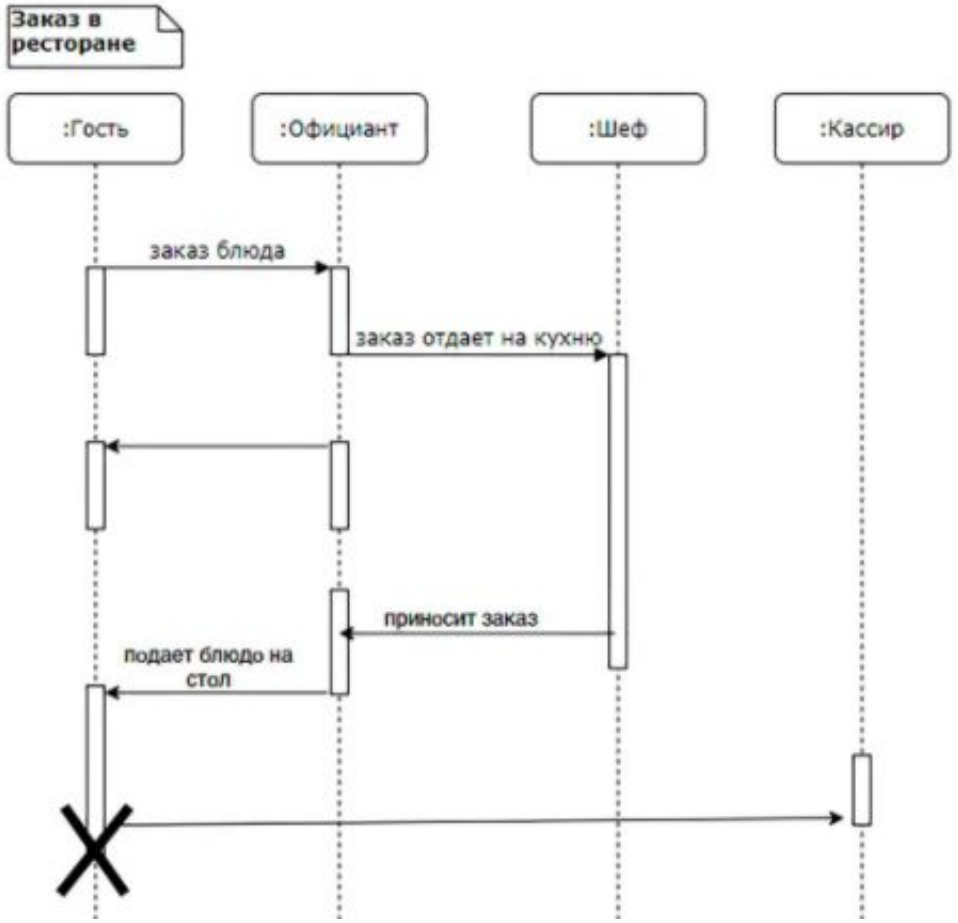
Диаграммы последовательностей обычно содержат **объекты**, которые **взаимодействуют в рамках сценария**, **сообщения**, которыми они обмениваются, и **возвращаемые результаты**, связанные с сообщениями.

Объекты обозначаются прямоугольниками с подчеркнутыми именами

Сообщения (вызовы методов) - линиями со стрелками

Возвращаемые результаты - пунктирными линиями со стрелками

Диаграмма последовательности для варианта использования «Заказ в ресторане»



- **Читаем диаграмму:**

гость заказывает еду у официанта,

- официант его слушает и далее идет на кухню, чтобы передать заказ повару,
- повар начинает готовить,

- параллельно с этим официант приходит к клиенту и наливает ему чай,

- пока клиент пьет чай, повар приносит еду на стойку официанту,

- официант забирает заказ и относит его гостю.

Больше клиент с официантом дел не имеет (официант выпадает из системы)

- гость оплачивает свой заказ у кассира и уходит.

Диаграммы последовательностей: синтаксис

Объявление участников

- actor
- participant
- boundary
- control
- entity
- database
- collections

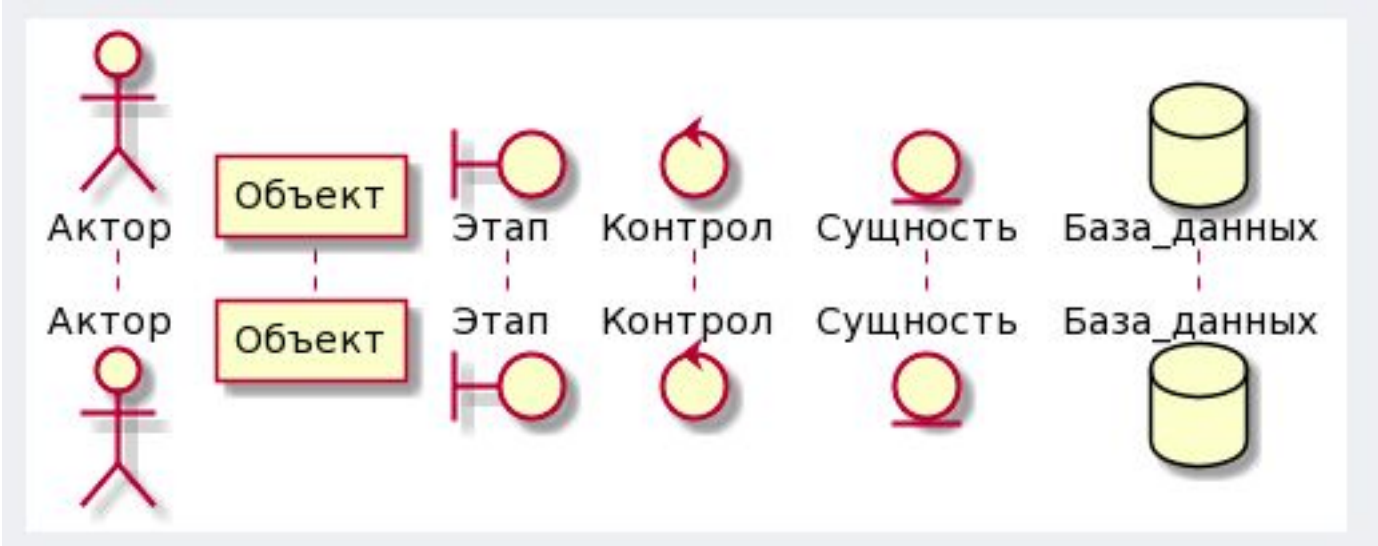
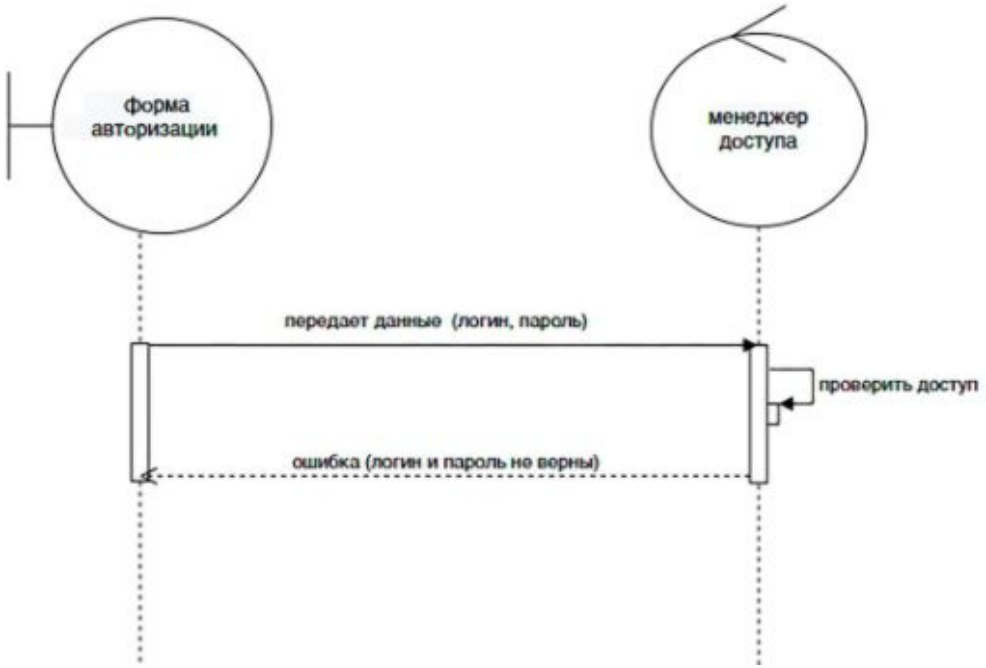


Диаграмма последовательности для варианта использования «Авторизация»



Читаем диаграмму:

Это форма авторизации. Она передает данные в менеджер паролей: логин и пароль. Менеджер паролей проверяет, есть ли такой пользователь в системе. Если нет, то выводит в форме авторизации ошибку. Еще форма авторизации ожидает ответ, пока менеджер пароля проверяет данные, и сообщает форме авторизации, какое сообщение необходимо вывести пользователю.

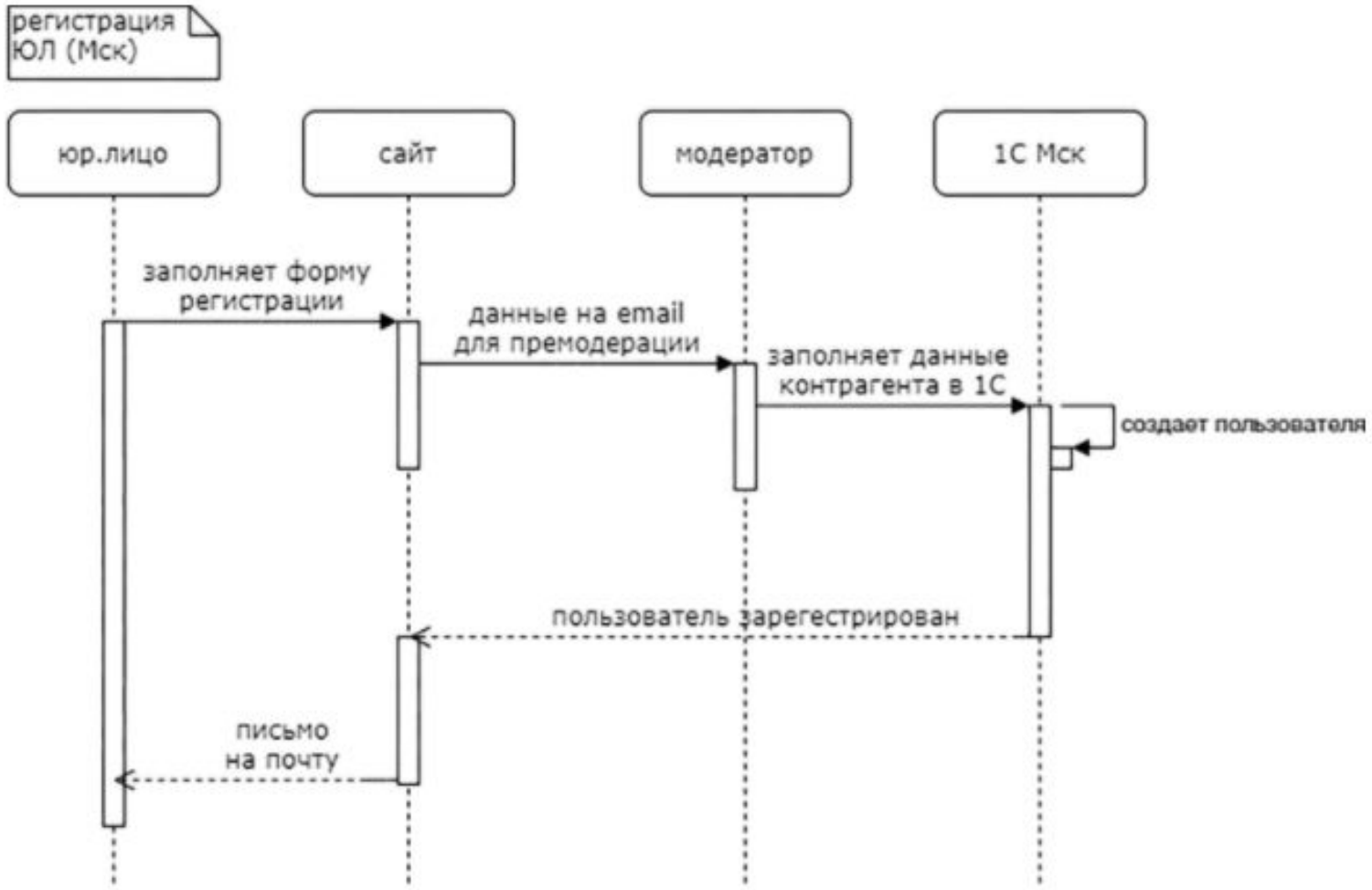
КЕЙС

Идет интеграция сайта с 1С — разрабатывается протокол интеграции. По ходу обсуждения выясняется:

- 1С-ки будет две: Тверская и Московская;
- Основная интеграция — с Тверской;
- Вся товарная база хранится в Тверской 1С;
- Помимо товаров, нужна выгрузка контрагентов;
- Контрагент привязывается при регистрации к Москве или к Твери;
- Контрагенты хранятся частично в Московской, частично — в Тверской базе;
- Не у всех текущих контрагентов заполнены поля емейл и телефон;
- Создание новых контрагентов требует премодерации;
- Кроме того, заказы необходимо передавать в CRM.

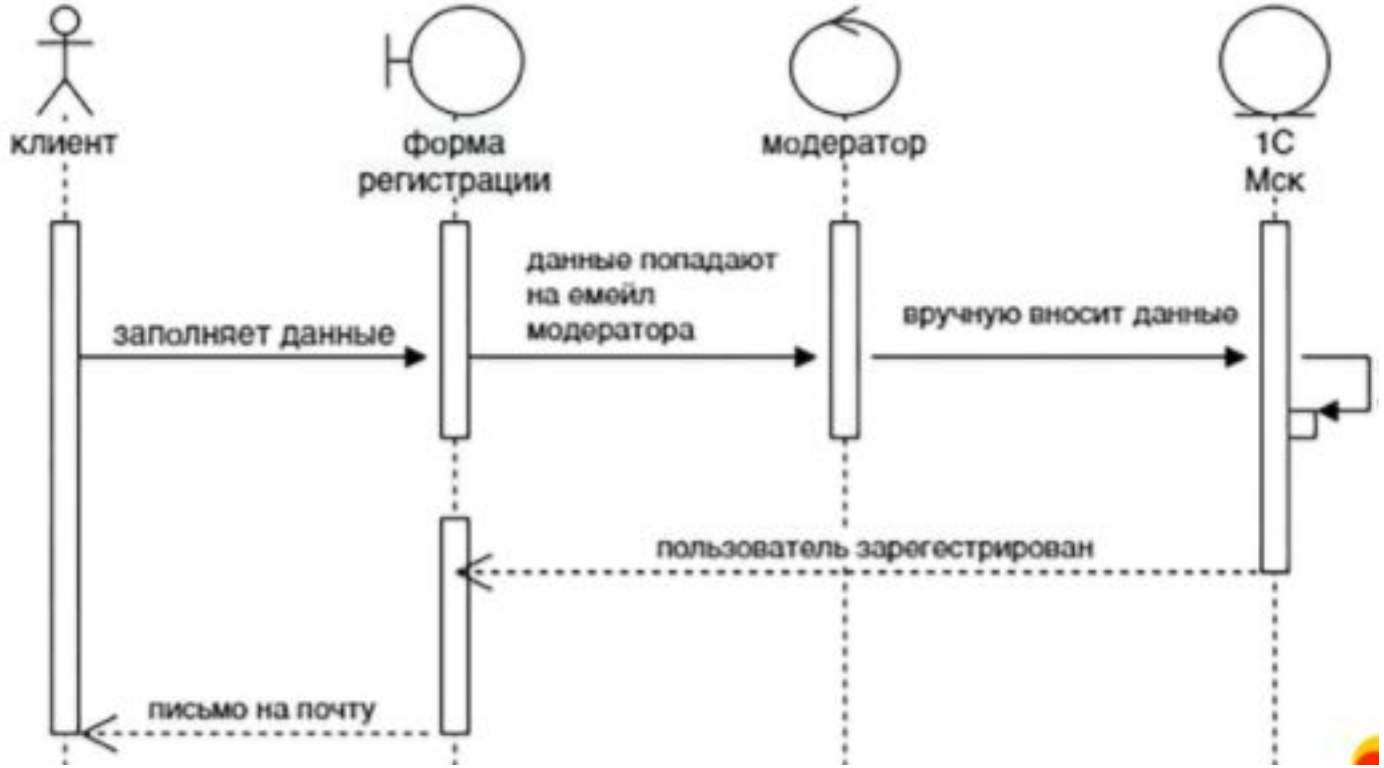
Клиент ожидает, что ему предложат решение, как будет работать схема с CRM, сайтом и 1С-ками.

Регистрация ЮЛ (Москва)



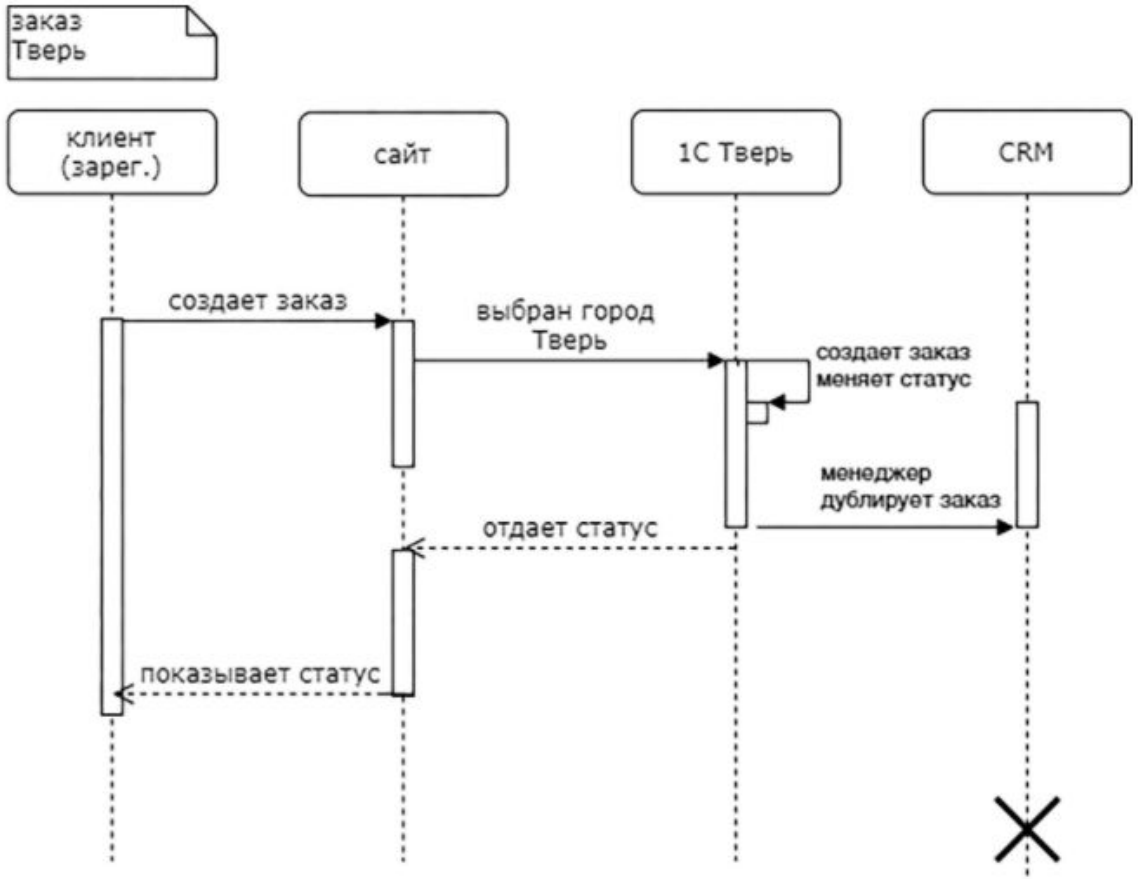
Юридическое лицо заходит на сайт и заполняет форму регистрации, указывая город Москва. Сайт передает данные на е-мейл модератору для премодерации. Модератор вручную заполняет данные контрагента в московскую 1С, создавая нового пользователя. Сайт получает сообщение, что регистрация завершена и отправляет письмо пользователю.

Регистрация пользователя



Физическое лицо заходит на сайт и заполняет форму регистрации, указывая город. Сайт передает данные в тверскую 1С, создавая нового пользователя. Сайт получает сообщение, что регистрация завершена и отправляет письмо пользователю.

Оформление заказа, Тверь



Зарегистрированный клиент создает заказ на сайте, выбирая город Тверь. В тверской 1С создается заказ (и менеджер в ручную дублирует заказ в 1С). После создания заказа в 1С у заказа меняется статус, который передается сайту и показывается клиенту.

Диаграмма последовательности для варианта использования «Обеспечить покупателя информацией»

Найдите ошибки!