# Введение

В жизни каждого рядового гражданина любой страны бывают ситуации, когда нужно взаимодействовать с государственными, муниципальными учреждениями, различными бюрократическими организациями. Такие контакты редко проходят без проблем, проволочек, задержек и недопонимания.

В этой работе дана возможность формализовать процесс взаимодействия гражданина с некоторой условной организацией при помощи созданного мною программного обеспечения.

## Используемые сокращения

* Созданное программное обеспечение, описываемое в данной работе – **Система**.
* Организация, использующая Систему – **Офис**.
* Гражданин, обращающийся в Офис с целью обработки некоторого запроса – **Заявитель**.
* Сам запрос Заявителя, подлежащий рассмотрению Офисом – **Запрос**.
* Сотрудник Офиса, задействованный в текущий момент в обработке Запроса – **Сотрудник**.
* Пользователь Системы (Сотрудник, Заявитель и др.) – **Пользователь**.

## Цель разработки

Целью Системы является полная фиксация действий по заполнению Запроса Заявителем, его обработке Сотрудниками, фактов общения между ними и Заявителем. Также в Систему заложены механизмы стимуляции своевременного выполнения требуемых действий обоими сторонами (Сотрудниками и Заявителем).

## Платформа, используемая в разработке

Для достижения оптимального соотношения между переносимостью, расширяемостью, удобством разработки и поддержкой современных стандартов было принято решение использовать современные открытые платформы и технологии. Обратной стороной такого отказа от коммерческих закрытых продуктов является необходимость отдельной установки, настройки различных современных платформ и технологий, их адаптация под разработчика и друг под друга.

Было найдено решение, соединяющее наиболее популярные компоненты разработки в одной платформе. Это платформа CUBA, распространяющаяся по свободной лицензии Apache 2.0 (https://www.cuba-platform.com/). Ее разработчики являются активными участниками международных Java-форумов и конференций. Ключевые компоненты и свойства платформы:

* Фреймворк **Vaadin**. Используется для формирования пользовательского веб-интерфейса Системы.
* Фреймворк **Spring**. Для реализации бизнес-логики среднего слоя Системы, для связи между ее основными блоками.
* Фреймворк **EclipseLink**. Для реализации ORM взаимодействия между средним слоем Системы и слоем хранения данных. ORM (англ. Object-Relational Mapping, рус. объектно-реляционное отображение).
* Платформа поддерживает популярные системы управления базами данных: PostgreSQL, MySQL, Oracle Database, Microsoft SQL Server. Также в платформу встроена HSQLDB (реляционная СУБД с открытым исходным кодом, полностью написанная на Java).

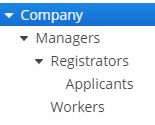
Платформа CUBA объединяет эти технологии под одной оболочкой, используя систему сборки проектов **Gradle**, так что программисту нет необходимости тратить время на настройку среды разработки. В пользовательском интерфейсе повсеместно используется концепция MVC (Model-View-Controller). Веб-интерфейс описывается декларативно с помощью xml-файлов, разработка контроллеров экранов и реализация бизнес-логики среднего слоя ведется на Java Standard Edition.

# Основные объекты Системы

## Пользователи

Поскольку платформа CUBA уже содержит развитые средства администрирования, протоколирования и обеспечения безопасности, было принято решение связать с Пользователями всех участников обработки Запросов с помощью композиции объектов. Различаться участники обработки Запросов должны по группам доступа и тесно связанным с ними ролям Пользователей.

## Группы доступа



Права и возможности доступа к объектам Системы, особенности интерфейса определяются принадлежностью Пользователя к одной из заданных групп доступа с назначенными ролями:

* Company – администраторы Системы с полными правами. Право на удаление Запросов принадлежит только администраторам.
* Managers – менеджеры Системы. Могут настраивать поведение Системы при обработке Запросов (см. далее раздел "[Объект Позиция (Position)](#_Объект_Позиция_(Position))"). Могут изменять состояние Запросов, но не создавать или редактировать их.
* Registrators – Сотрудники, отвечающие за создание Запросов, внесение и корректировку личных данных Заявителей. Могут редактировать только ту часть Запроса, где содержится информация о личных данных Заявителя.
* Applicants – Заявители. Пользователи с минимальными правами. Могут видеть только свой Запрос и редактировать его только в части подачи запрошенной информации и общения с Сотрудниками.
* Workers – Сотрудники с ограниченными правами. Могут видеть только назначенные им в текущий момент Запросы и редактировать их только в части одобрения (или отказа в одобрении) информации от Заявителя и общения с Заявителями.

Важен порядок подчиненности групп доступа. Пользователь может видеть только участников своей группы и подчиненных ей. Поэтому при внесении Запросов в Систему пользователи группы Registrators могут оперировать только пользователями группы Applicants, а пользователи группы Managers – создавать и изменять данные пользователей групп Registrators, Applicants и Workers.

## Объект Позиция (Position)

Набор составных объектов Позиция описывает всю логику работы Системы по обработке Запроса. В общем смысле перед сотрудниками Офиса и Заявителем стоит задача провести Запрос от начальной Позиции, на которую он ставится при создании, через все промежуточные объекты до конечной Позиции, на которой обработка Запроса считается завершенной.

Каждая позиция характеризуется своим уникальным идентификатором (по которому проводится сортировка Позиций), описанием, количеством дней для подачи информации Заявителем и о количеством дней для обработки предоставленных данных Сотрудником.

Объекту Позиция подчинены два набора объектов:

* набор Действий Позиции (PositionAction). Содержит список действий, выполнение которых необходимо для перехода на новую Позицию.
* набор Сотрудников, закрепленных за Позицией (PositionUser).

Набор Позиций и его отдельные элементы могут редактироваться только администраторами или менеджерами Системы.

### Объект Действие Позиции (PositionAction)

Содержит тип требуемого от Заявителя Действия (элементы перечисления ActionType: "отправить файл", "написать сообщение", "позвонить", "приехать в Офис" и т.д.) и описание этого действия.

### Объект Сотрудник Позиции (PositionUser)

Содержит композицию с пользователем Системы, максимальное количество назначаемых Запросов, и текущее количество обрабатываемых запросов (меняется Системой)

## Объект Запрос (Request)

Основным объектом, над которым ведется работа в Системе, является Запрос. Это – составной объект, содержащий композицию с Заявителем, его персонализированные данные и данные самого Запроса (серия, номер, описание и т.д.).

Объекту Запрос подчинены два набора объектов:

* набор Записей Протокола Запроса (RequestLog), фиксирующих все изменения Запроса Пользователями.
* набор Шагов Запроса (RequestStep) – составных объектов, хранящих текущее состояние Запроса (см. далее раздел "[Объект Шаг Запроса (RequestStep)](#_Объект_Шаг_Запроса)"). Для ускорения выборки текущего Шага Запроса используется денормализация – добавлено отдельное хранение в Запросе ссылки на последний элемент этого набора.

### Объект Запись Протокола Запроса (RequestLog)

Создается при любом изменении Запроса. Доступен только для чтения. Содержит информацию о проведенном изменении, ссылки на инициатора изменения и адресата записи (Пользователи Системы). Также содержит дополнительную информацию об объекте, затронутым этим изменением. На основе набора Записей Протокола Запроса построена система оповещений Пользователей (при создании – адресату отправляется электронная почта, также любой Пользователь может просмотреть личный ящик сообщений, содержащий все Записи Протокола Запроса, адресованный лично ему или инициированные им).

## Объект Шаг Запроса (RequestStep)

Составной объект, содержащий информацию о текущем шаге Запроса. Создается автоматически, удаление недоступно (только вместе с Запросом).

Он характеризуется тремя ключевыми параметрами:

* Позицией, на которой находится Запрос.
* Состоянием Запроса (элементы перечисления State: "приостановлен", "вносится информация", "проверяется", "закрыт", "архивирован", "отменен").
* Сотрудником, проводящим проверку (только для состояний "вносится информация", "проверяется"). Данный параметр может быть неопределенным, если на выбранной позиции нет свободного Сотрудника Позиции.

Также содержит граничные (для текущего шага) сроки полного внесения информации Заявителем (заполняется автоматически если назначен Сотрудник) и одобрения Запроса Сотрудником (заполняется автоматически после отправки Запроса на одобрение) и даты реального внесения и одобрения.

Поле штрафа заполняется автоматически (через задачу планировщика Системы) и используется для стимуляции как своевременного внесения требуемой информации Заявителем, так обработки внесенной информации Сотрудником.

Объекту Шаг Запроса подчинены два набора объектов:

* набор Действий Шага Запроса (RequestStepAction). Формируется автоматически из набора Действий Позиции при создании Шага Запроса (если назначен Сотрудник). Содержит список действий, выполнение которых необходимо для перевода Запроса на новую Позицию (после одобрения Сотрудником).
* набор Общений Шага Запроса (RequestStepCommunication). Для общения между Пользователями, задействованными в обработке Запроса.

### Объект Действие Шага Запроса (RequestStepAction)

Создается автоматически, доступен только для изменения.

Заявитель может только внести информацию, соответствующую указанному типу действия (отправить файл, написать сообщение). После чего становится доступной возможность отправки (submit) действия. После отправки всех действий шага можно отправить Запрос на одобрение Сотруднику.

Сотрудник может только просматривать внесенную в действие Пользователем информацию, одобрять (approve) ее, если она помечена как отправленная (submitted). После одобрения всех действий можно провести одобрение всего Запроса, что вызовет перевод Запроса на новый шаг.

### Объект Общение Шага Запроса (RequestStepCommunication)

Создается Заявителем, Сотрудником или членами вышестоящих групп пользователей. Содержит поля текста вопроса (обязательное для создателя), файла вопроса, текста ответа (обязательное для отвечающего), файла ответа. Автор Общения может редактировать текст и файл вопроса, а также удалять Общение пока поля ответа пустые. Получатель общения может редактировать текст и файл ответа. Если автор Общения принял ответ Получателя, редактирование Общения становится недоступным.

# Работа Системы

## Начальная настройка

Перед началом работы системы администратором должны быть проведены некоторые настройки:

* Внесены группы в соответствии с изложенным в разделе "[Группы доступа](#_Группы_доступа)".
* Добавлены пользователи в группы "Managers", "Registrators" и "Workers".
* Заполнен набор Позиций.
* Установлены системные свойства: указаны группы доступа, начальная и конечная позиции. Это удобнее делать через экран "Settings", но можно и вручную через "Administratrion" – "Application Properties" – раздел "office".

|  |  |
| --- | --- |
|  |  |

## Работа с Запросом

Запрос вводится членом группы "Registrators". При этом заполняются необходимые поля для Запроса (серия, номер, описание) и для Заявителя. При записи автоматически проводится контроль уникальности требуемых полей. После записи Запрос переводится на начальную позицию по алгоритму, приведенному ниже.

### Перевод Запроса на указанную Позицию

Создается новый Шаг Запроса: Позиция – указанная, Состояние – "приостановлен", Сотрудник – "null".

Далее проводится поиск свободных Сотрудников Позиции:

* Если такой Сотрудник найден, то создается новый Шаг Запроса: Позиция – указанная, Состояние – "вносится информация", Сотрудник – найденный свободный Сотрудник Позиции. На основании набора Действий Позиции создается набор Действий Шага Запроса. Начинается ожидание внесения Заявителем информации, запрошенной в Действиях Шага Запроса, и их одобрения Сотрудником.
* Если поиск свободных Сотрудников Позиции не дал результата, Запрос остается в состоянии "приостановлен", поиск будет повторен через задачу планировщика Системы.

### Поиск свободных Сотрудников Позиции

Перебираются все Сотрудники Позиции, у которых число назначенных Запросов (A) меньше максимального количества назначаемых Запросов (B). Выбирается тот, у которого величина загруженности A/B минимальна.

### Работа над Запросом Заявителя и Сотрудника

При получении сообщения о создании Шага Запроса из Состоянием "вносится информация" Заявитель должен внести запрошенные данные в Действия Шага Запроса. Далее отправить (submit) каждое Действие. После отправки всех действий Заявитель может отправить **весь** Запрос (для текущего Шага). Желательно это сделать до истечения срока полного внесения информации. В противном случае к Заявителю будут применяться санкции вплоть до отмены Запроса.

После отправки Запроса Заявителем устанавливается срок одобрения Запроса Сотрудником. Сотрудник должен проверить и одобрить (или отклонить) информацию от Заявителя для каждого Действия Шага Запроса. После одобрения всех действий Сотрудник может одобрить **весь** Запрос (для текущего Шага). Желательно это сделать до истечения срока одобрения. В противном случае к Сотруднику будут применяться санкции вплоть до блокировки входа в Систему.

Одобренный запрос переводится на следующую позицию по правилам, описанным в разделе: "[Перевод Запроса на указанную Позицию](#_Перевод_Запроса_на)".

Напомню, что все действия над Запросом и его подчиненными объектами фиксируются в объектах Запись Протокола Запроса (RequestLog) с отправкой сообщений задействованным Пользователям.