

## Лабораторная работа 6

# Отчёты

Требуется реализовать механизм автоматизации создания отчётов о проделанной командой работе за определённый период разработки (далее - спринт), интегрируя это с системой управления задачами, используемой командой. В команде один из сотрудников - тимлид, он составляет итоговый отчёт о проделанной работе команды в конце каждого спринта.

### Ход событий:

1. Сотрудник может добавлять новые задачи, вносить изменения в существующие и, наконец, выполнять их.
2. Сотрудник должен писать отчёт о проделанной работе за каждый рабочий день. Чтобы это сделать он использует список всех изменений, произведённых с момента создания предыдущего отчёта. На основе ежедневных отчётов и будет составляться отчёт за спринт - итог проделанной каждым сотрудником работы. В течение дня сотрудник делает отчёт, прикрепляя к нему выполненные за этот день задачи.
3. В конце спринта сотрудник должен написать отчёт за спринт. Для этого у него имеется доступ к выполненным за время спринта задачам и к ежедневным отчётам.
4. Тимлид в конце спринта пишет отчёт за всю команду, просматривая список выполненных задач и отчётов.

### Сущности

- Сотрудник
- Задача, существующая в системе управления задачами
- Отчёт (за один день и за весь спринт)

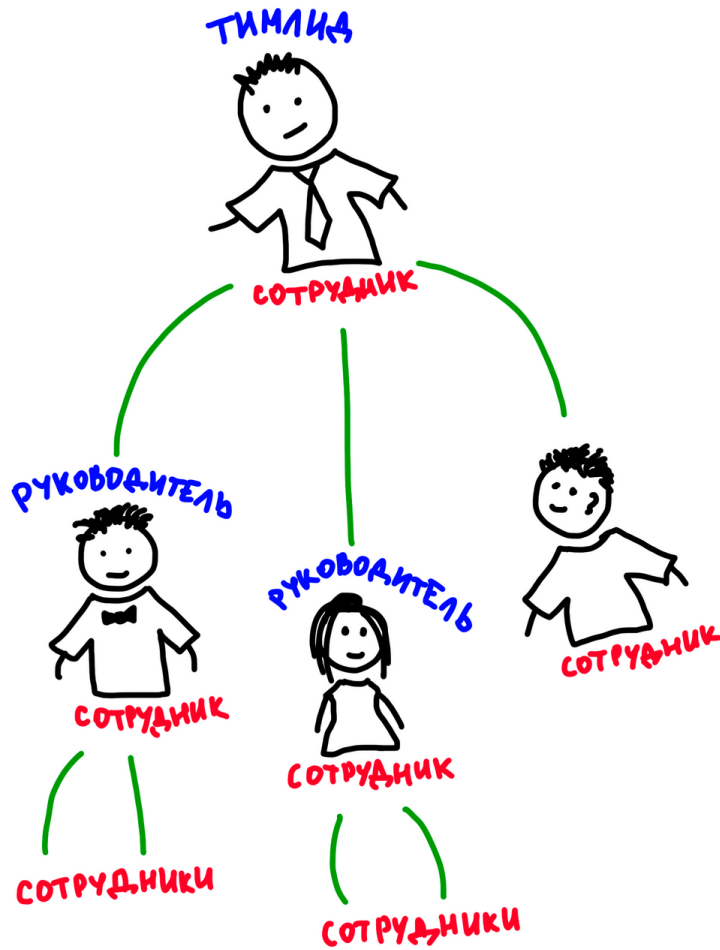
### Сотрудник

Сотрудник представлен следующим набором атрибутов:

- имя
- руководитель (необязательный)
- список подчиненных (необязательный)

В реализованной системе должна быть возможность добавлять новых сотрудников, изменять руководителя сотрудника и получать иерархию всех сотрудников.

Иерархия сотрудников представляется следующим образом: тимлид - сотрудник, руководитель - корень дерева, которому подчиняются другие сотрудники, которые могут иметь или не иметь подчинённых.



## Задача

Задача представлена следующим набором атрибутов:

- название
- описание
- назначенный за ней сотрудник (только один на задачу)

Задача может находиться в одном из трёх состояний:

1. Open - задача создана, но к её выполнению ещё не приступили.
2. Active - задача находится в процессе выполнения.
3. Resolved - задача выполнена.

Задача находится в системе управления задачами, которую также необходимо реализовать как отдельную сущность. Сотрудник должен иметь возможность добавить комментарий к задаче.

Все изменения (состояние/назначенный сотрудник/комментарий) должны фиксироваться во времени. При этом в системе должна быть возможность просмотреть, когда было сделано определённое изменение.

**Система управления задачами** должна поддерживать следующий функционал:

- поиск задач по ID

- поиск задач по времени создания/последнего изменения
- поиск задач, закреплённых за определённым пользователем (сотрудником команды)
- поиск задач, в которые пользователь вносил изменения
- создание задачи, изменение её состояния, добавления к ней комментария, изменение назначенного за ней сотрудника
- получение списка задач, которые назначены подчинённым определённого сотрудника

## **Отчёт**

### **За день**

Необходимо реализовать возможность создания отчёта за текущий день и его редактирования в течение этого дня. Также у сотрудника должна быть возможность просмотреть задачи, которые были выполнены за этот период.

### **За спринт**

Необходимо реализовать возможность написания отчётов за весь спринт. В системе на каждый спринт создается драфт отчёта (то есть черновик, начальный проект отчёта за спринт, который в дальнейшем будет корректироваться), который заполняет каждый сотрудник. Для написания отчёта пользователю системы должна предоставляться возможность получить список всех своих задач за спринт, а также список отчётов своих подчинённых за спринт.

После окончания написания отчёта за спринт сотрудник должен сохранить его в системе. После этого отчёт считается завершённым, доступ на его редактирование закрывается. Тимлид должен иметь возможность видеть статус отчёта. Если отчёт завершённый, то он агрегирует в командный отчёт за спринт.

## **Детали реализации**

В рамках лабораторной предполагается реализация многослойной архитектуры.

На усмотрение студента можно добавить:

- хранение в базе данных, использованием ORM