Отрисуем процесс **“Выдача справки сотруднику”** в нотации UML, а именно - **диаграмму деятельности (Activity Diagram)**.

**Процесс: “Выдача справки сотруднику”**

[Initial Node] -- "Запрос на справку" (черный круг)

|

v

[Action] -- "Сотрудник подает заявку" (прямоугольник)

|

v

[Decision Node] -- "Заявка подана онлайн?" (ромб)

/ \

Да Нет

| |

v v

[Action] -- "Система проверяет данные" (прямоугольник) [Action] -- "Сотрудник заполняет заявление" (прямоугольник)

| |

v v

[Action] -- "Система формирует справку" (прямоугольник) [Action] -- "Сотрудник передает заявление" (прямоугольник)

| |

v v

[Merge Node] -- (жирная черта)

|

v

[Decision Node] -- "Справка срочная?" (ромб)

/ \

Да Нет

| |

v v

[Action] -- "Справка формируется немедленно" (прямоугольник) [Action] -- "Справка формируется в течение X дней" (прямоугольник)

| |

v v

[Merge Node] -- (жирная черта)

|

v

[Action] -- "Сотрудник получает справку" (прямоугольник)

|

v

[Activity Final Node] -- (круг с точкой)

**Расшифровка элементов UML Activity Diagram:**

* **[Initial Node] (черный круг):** Начальная точка процесса. В данном случае — “Запрос на справку”.
* **[Action] (прямоугольник):** Задача или действие, которое выполняется. Например, “Сотрудник подает заявку”, “Система формирует справку”, “Сотрудник получает справку”.
* **[Decision Node] (ромб):** Точка принятия решения, где процесс разветвляется на один из нескольких возможных путей. Например, “Заявка подана онлайн?”, “Справка срочная?”.
* **[Merge Node] (жирная черта):** Точка слияния, где потоки процесса объединяются после принятия решений.
* **[Activity Final Node] (круг с точкой):** Конечная точка процесса.
* **[Arrows] (стрелки):** Указывают направление потока управления.

**Описание процесса:**

1. **Запрос на справку:** Процесс начинается с запроса сотрудника на получение справки.
2. **Подача заявки:** Сотрудник подает заявку на получение справки.
3. **Онлайн или нет:** Проверяется, подана ли заявка онлайн.
   * **Да:** Система автоматически проверяет данные и формирует справку.
   * **Нет:** Сотрудник заполняет заявление и передает его.
4. **Слияние потоков:** После заполнения или автоматической обработки, поток объединяется.
5. **Срочность справки:** Проверяется, является ли справка срочной.
   * **Да:** Справка формируется немедленно.
   * **Нет:** Справка формируется в течение X дней.
6. **Получение справки:** Сотрудник получает готовую справку.
7. **Завершение процесса:** Процесс завершается.

**Дополнительные замечания:**

* В реальной диаграмме деятельности UML, каждый элемент может иметь дополнительные свойства и атрибуты.
* Этот процесс может быть расширен, добавлением, например, проверок наличия необходимых данных, проверки правильности заявления, и т.д.
* UML Activity Diagram позволяет наглядно представить последовательность действий, принятия решений и возможные пути выполнения процесса.

**Преимущества Activity Diagram:**

* **Наглядность:** Легко увидеть последовательность действий.
* **Простота:** Диаграмма легко читается и понимается.
* **Универсальность:** Подходит для описания разных процессов.
* **Подходит для описания процессов, в которых важно показать последовательность действий и разветвления процесса.**

Этот пример демонстрирует, как можно использовать UML Activity Diagram для описания процесса выдачи справки сотруднику.