**Требования к системе**

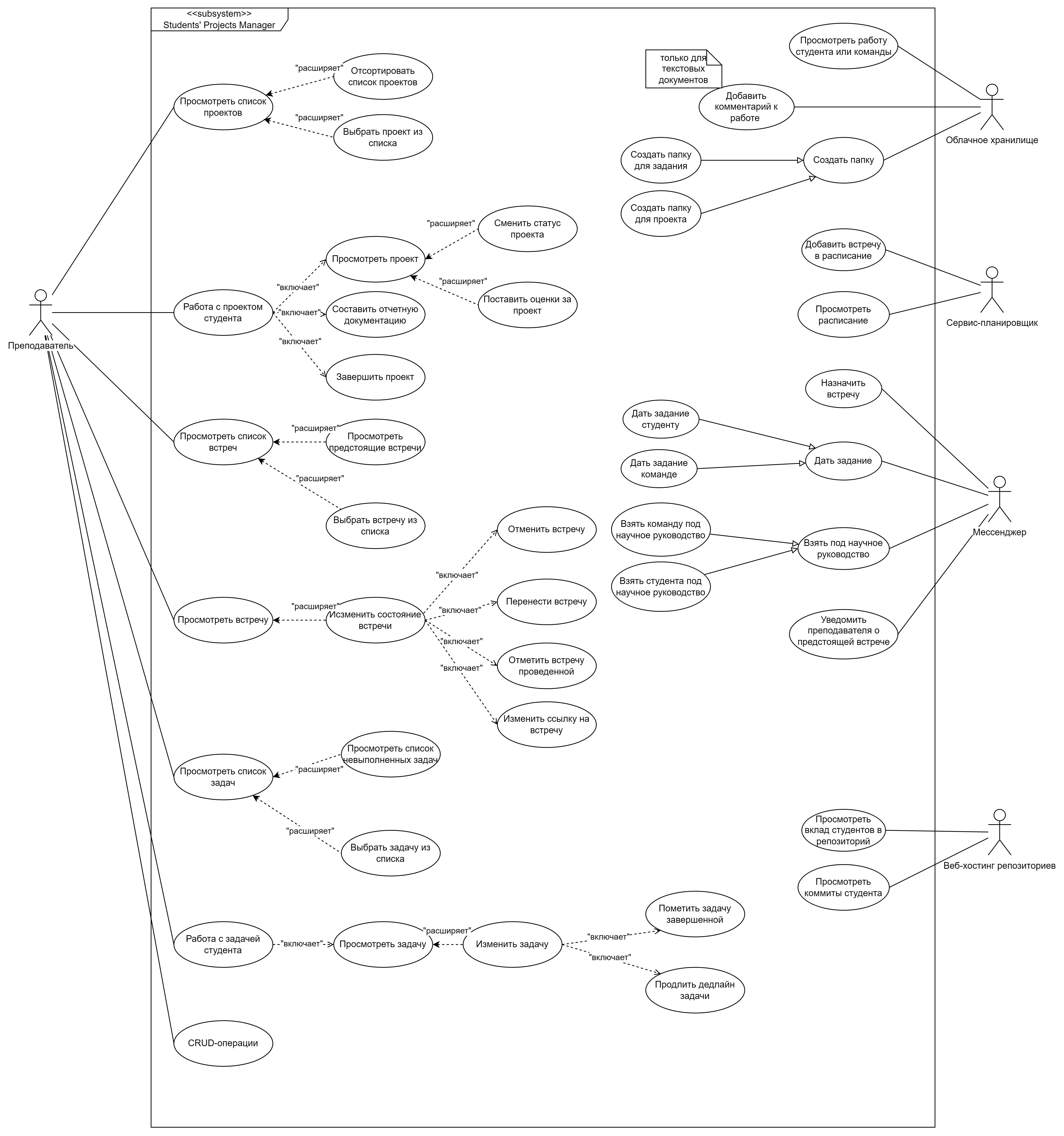
**Стейкхолдеры**

1. Преподаватель (доступность, надежность, безопасность, конфиденциальность, аутентифицируемость, целостность) - преподаватель может заниматься курированием работ в любое время, поэтому система должна быть доступна в любое время. Ему важно, чтобы данные о проектах всегда были актуальными, не терялись и не портились. Так как интеграции используют личные данные преподавателя со сторониих сервисов, то они должны быть защищены от утечки.
2. Студент (конфиденциальность, целостность) - студент не пользуется заявленной системой, однако его работы доступны преподавателю и системе, и ему важно, чтобы эта информация была защищена от распространения, и система не изменяла его файлы.
3. Разработчики (расширяемость, масштабируемость, сопровождаемость, адаптируемость) - разработчикам важно, чтобы в систему было легко вносить изменения при расширении функционала, масштаба, платформ.

**Функциональные требования**

1. Преподаватель регистрирует свои данные (ФИО, почта, телефон)
2. Преподаватель дает системе права доступа на использование личных данных интеграционных сервисов.
3. Преподаватель берет под научное руковоство студентов при помощи интеграции с мессенджером.
4. Преподаватель добавляет взятых под научное руководство проекты и студентов в систему.
5. Преподаватель управляет проектами студентов.
6. Преподаватель дает задания студентам со сроком при помощи интеграции с мессенджером.
7. Преподаватель просматривает задания студентов при помощи интеграции с облачным хранилищем.
8. Преподаватель просматривает исходный код студентов при помощи интеграции с менеджером репозиториев.
9. Преподаватель оценивает задания студентов.
10. Преподаватель дает комментарии к заданиям студентов при помощи интеграции с облачным хранилищем
11. Преподаватель назначает встречи студентам при помощи интеграции с мессенджером.
12. Система автоматически формирует отчетные документы преподавателя по проектным работам.
13. Система формирует расписание преподавателя при помощи интеграции с планировщиком событий.
14. Система оповещает преподавателя о предстоящих встречах за день до встречи в 12:00.

**Диаграмма вариантов использования:**



Перечень предположений:

* Одна проектная работа может выполняться несколькими студентами.
* Рассматриваются проектные работы, выполняемые с 1 по 4 курс бакалавриата (в частности КР, ВКР)
* Встречи преподавателя и студентов проходят на стороннем сервисе.
* Система предназначена исключительно для преподавателя, ВУЗ не имеет к ней никакого отношения.

**Нефункциональные требования:**

**Явные требования**

Рассмотрим требования заказчика:

* Преподаватель размещает проект студента и получает информацию о нем.

*Т.к. проект размещает пользователь, то требуется иметь защиту от ошибок пользователя при вводе данных (Usability)*

* Преподаватель дает задания студентам со сроком выполнения.

*Т.к. задания размещает пользователь, то требуется иметь защиту от ошибок пользователя при вводе данных (Usability)*

* Преподаватель назначает встречи студентам.
* Преподаватель видит прогресс проектных работ студентов.

*Т.к. преподавателю важно, чтобы студенты вовремя и корректно выполняли этапы проектных работ, то информация по ним должна быть актуальной и корректной (корректность, целостность).*

* Преподаватель видит прогресс выполнения заданий студентов.

*Т.к. преподавателю важно, чтобы студенты вовремя и корректно выполняли задания, то информация по ним должна быть актуальной и корректной (корректность, целостность).*

* Интеграция со сторонними сервисами, содержащих информацию, связанных с проектной деятельностью студентов (мессенджер, менеджер репозиториев, облачное хранилище, планировщик событий).

*При интеграции со сторонними сервисами может пострадать надежность системы, т.к. основная задача системы - получение информации из сервисов и управление ею в системе. Если обращение к сервису терпит неудачу, то это оказывает влияние на надежность вызывающей системы. Помимо всего прочего пользователь предоставляет личные данные со сторониих сервисов, и они должны быть защищены от утечки (надежность и безопасность, конфиденциальность).*

* Расписание встреч.

*Преподавателю важно видеть актуальное и верное расписание во избежания накладывания встреч со студентами (корректность, целостность).*

* Отчетная документация.

*Отчетные документы должны быть оформлены по стандартам университета, поэтому важно корректно сформировывать их (корректность, целостность, правильность).*

* Оценки за работу/проект.
* Преподаватель общается со студентом через мессенджер.
* Студент загружает задания/проект на сторонний сервис (облачное хранилище, менеджер репозиториев).

Помимо перечисленных выше архитектурных свойств можно также выделить такие свойства как аутентифицируемость и авторизуемость, т.к. преподавателю может понадобится доступ с другого устройства, а другие люди, заходящие в систему не могли видеть чужую информацию.

Данная система вряд ли будет иметь широкий круг пользователей, т.к. в основном ориентирована на проекты и преподавателей НИУ ВШЭ, поэтому такое свойство как масштабируемость не является ключевым в данной архитектуре.

Преподаватели вряд ли готовы ждать несколько минут, чтобы узнать собственное расписание или какую-либо другую информацию, поэтому важно учесть такое свойство как производительность. Т.к. предполагается, что системой будет пользоваться относительно малое количество человек, то данное свойство не будет основным фокусом критериев качетсва.

**Неявные требования**

Данная система имеет потенциал к расширению своей целевой аудитории (например, отслеживание дргуих видов проектов, создание модуля для студентов) и функционалу (больше способов интеграции), поэтому для успешного продолжения развития проекта требуется чтобы он был сопровождаемый, расширяемый.

Нельзя сказать, когда именно преподаватели будут пользоваться системой, поэтому требуется, чтобы система была доступной в любое время.