**Причины использования SSO СПбГУ**

1. **Улучшение надёжности сервиса**. - пользователи, которые будут использовать сервис, скорее всего часто **будут использовать тот же пароль**, что и для **учётной записи** СПбГУ, чтобы не приходилось запоминать новый пароль. Если база данных будет скомпрометирована, то опасности может подвергнуться все сервисы пользователя, использующие учётную запись СПбГУ.

2. **Фильтрация нежелательных пользователей**. - на данном этапе сервис рассчитан на использование **именно студентами** СПбГУ.

У пользователя есть учётная запись СПбГУ <=> ему допускается пользоваться этим сервисом.

3. **Более качественный user experience**. - нет необходимости в создании аккаунта, подтверждения почты и т.д.

Иначе придётся придумывать другие способы аутентификации, более **сложные** в реализации для разработчика и для использования пользователя

4. **Расширяемость**. - в процессе расширения функционала сервиса может быть **полезна информация** о студенте, которая может быть предоставлена SSO СПбГУ (например, курс или форма обучения)

**Предполагаемая реализация**

На данный момент в сервисе настроен SSO от Яндекса по протоколу OAuth 2.0. Насколько я помню с прошлого семестра, SSO СПбГУ этот протокол поддерживает.

Принципиальных отличий в реализации в этом случае не будет. Нужно будет только изменить:

* **authorizationURL**
* **tokenURL**
* **clientID**
* **clientSecret**
* callbackURL / redirectURL
* **scope** и **url для получения данных о пользователе**. Для сервиса на данном этапе нужны только ФИО и корпоративная почта

Из этого списка я **управляю только callbackURL**. Соответственно, для возможности внедрения SSO СПбГУ, мне нужно предоставить **все остальные элементы списка**.

Дополнительно была бы полезна (но необязательна) **документация о формате возвращаемых данных о пользователе**. Это немного ускорит процесс настройки адаптера, но в худшем случае я сам смогу просмотреть возвращаемые данные и понять, как их стоит обработать.

Сервис будет **обращаться** к SSO **не слишком часто**: при успешной авторизации сервис создаёт свои access- и refresh- JWT-токены, которые хранят всю необходимую информацию о пользователе.