Отчет по UML-диаграмме.

Базовые интерфейсы:

1) **IHTTPConnector** – предоставляет интерфейс для взаимодействия с сетью посредством HTTP-запросов.
    1. Вспомогательная шаблонная структура **HTTPResponse** для возврата результата запроса. Содержит тело, заголовки и статус ответа.
2) **ILocalStorage** – предоставляет интерфейс для централизованного хранения и получения данных между объектами в формате ключ-значение в виде строк.

Реализации базовых интерфейсов:

1) **HTTPConnector** - реализация интерфейса IHTTPConnector с использованием библиотеки boot::beast.

Вспомогательный класс:

1. **Deferrer** – класс, который выполняет в своем деструкторе функции, которые в него положили. Например, функция закрытия tcp-соединения.

2) **LocalStorageInMemory** – реализация интерфейса ILocalStorage, хранящая все данные в памяти программы в map<string, string>.

Пользовательские интерфейсы:

1) **IAuthConnector** – интерфейс, позволяющий пользователю зарегистрироваться, войти и выйти из системы.
2) **IEventsConnector** – интерфейс, позволяющий пользователю посмотреть события, отметиться для участия в одном из них или создать событие.
3) **IUsersConnector** – интерфейс, позволяющий пользователю посмотреть свой профиль и редактировать его.

Общие структуры интерфейсов:

1. Шаблонная структура ответа на пользовательский запрос Response<T, boost::error_code>.
2. Структура User.
3. Структура Token.
4. Структура Address.
5. Структура Event.

Конечный продукт:

1) PartyTimeConnector – класс, который хранит реализации все трех пользовательских интерфейсов в виде shared_ptr, которые принимает в конструкторе.