

Протокол тестирования API авторизации с токеном облачной системы.

№	Вызываемый метод	Ожидаемый результат	Результат
1.	AccessManager		
1.1.	login(String[] data)	Происходит авторизация пользователя по паролю и логину	
1.2.	logout()	Происходит выход из личного кабинета пользователя и сброс токена авторизации	
1.3.	getHashToken(String token)	Происходит получение нового токена авторизации	
2.	AccessController		
2.1.	setIView(iView iView)	Метод устанавливает класс, реализующий интерфейс iView для работы с пользовательским интерфейсом	
2.2.	getIView: iView	Метод получает класс, реализующий интерфейс iView для работы с пользовательским интерфейсом	
2.3.	setIRepo(iRepo iRepo)	Метод устанавливает класс, реализующий интерфейс iRepo для работы с базой данных	
2.4.	getIRepo: iRepo	Метод получает класс, реализующий интерфейс iRepo для работы с базой данных	
2.5.	checkToken(Token Token, User User): bool	Происходит проверка полученного токена авторизации. Метод возвращает булево значение	
2.6.	generateToken(User User): Token	Происходит генерация нового токена авторизации. Метод возвращает новый экземпляр класса Token	
2.7.	delToken(User User)	Происходит удаление токена авторизации	
3.	Token		
3.1.	setToken(String[] token)	Метод устанавливает токен авторизации на основе входящего массива данных, хешируя его.	
3.2.	getToken(): String	Метод возвращает токен авторизации в виде значения типа String	
4.	AccessRepo		
4.1.	createUser(String[] data)	Метод создает нового пользователя на основе входящего массива данных, сохраняя его в базе данных	

4.2.	updateUser(String[] data)	Метод редактирует данные пользователя на основе входящего массива данных, сохраняя изменения в базе данных	
4.3.	deleteUser(int idUser)	Метод удаляет пользователя из базы данных по его id	
4.4.	readUser(int idUser)	Метод читает данные пользователя из базы данных по его id	
5.	User		
5.1	setLogin(String login)	Метод устанавливает параметр Login для пользователя, сохраняя изменения в базе данных	
5.2.	getLogin(): String	Метод получает параметр Login для пользователя из базы данных	
5.3.	setEmail(String email)	Метод устанавливает параметр Email для пользователя, сохраняя изменения в базе данных	
5.4.	getEmail(): String	Метод получает параметр Email для пользователя из базы данных	
5.5.	setPassword(int pass)	Метод устанавливает параметр Password для пользователя, сохраняя изменения в базе данных. Пароль хранится в виде хеша	
5.6.	getPassword(): int	Метод получает хешированный параметр Password для пользователя из базы данных	