Archive API

1. Авторизация на стороне API происходит по токену, который выдается ЦАА (Центр Авторизации и Аутентификации).
2. Роли устанавливаются ЦАА:
   1. Гость (*Guest*);
   2. Зарегистрированный пользователь (*RegistratedUser*);
   3. Подтвержденный пользователь (*ConfirmedUser*);
   4. Супер пользователь (*SuperUser*).
3. Методы API:

|  |  |  |
| --- | --- | --- |
| Название метода | | |
| Описание метода | Входные данные | Выходные данные |
| Получить множество задач (*GetProblems*) | | |
| Метод должен возвращать список задач по заданному идентификатору контеста. | Идентификатор контеста  Тип: целое число | Список задач  Тип: список объектов типа Problem |
| Получить ссылку на условие задачи (*GetStatementUrl*) | | |
| Метод должен возвращать ссылку на условие задачи по заданному идентификатору контеста и идентификатору задачи в контесте. | Идентификатор контеста  Тип: целое число | Ссылка на условие задачи  Тип: строковое значение |
| Идентификатор задачи в контесте  Тип: строковое значение |

1. Ограничения по ролям:

|  |  |  |
| --- | --- | --- |
| Метод | Роли | Ограничения |
| Получить множество задач (*GetProblems*) | Гость (*Guest*) | Задачи из открытого контеста |
| Зарегистрированный пользователь (*RegistratedUser*) | Задачи из открытого контеста |
| Подтвержденный пользователь (*ConfirmedUser*) | Задачи из открытого или доступного только ему контеста |
| Супер пользователь (*SuperUser*) | Полный доступ |
| Получить ссылку на условие задачи (*GetStatementUrl*) | Гость (*Guest*) | Задача из открытого контеста |
| Зарегистрированный пользователь (*RegistratedUser*) | Задача из открытого контеста |
| Подтвержденный пользователь (*ConfirmedUser*) | Задача из открытого или доступного только ему контеста |
| Супер пользователь (*SuperUser*) | Полный доступ |

1. Программная реализация продукта должна использовать готовые системы:
   1. Система тестирования (*Executor API*);
   2. Система авторизации и аутентификации (*CAA API*);
   3. Систему ролей *CAA*;
   4. *ORM Entity Framework*.
2. Интерфейс спецификации
   1. Интерфейс спецификации должен генерироваться с помощью готового инструмента Swagger.
   2. Интерфейс для каждого метода должен содержать:
      1. Описание входных данных.
      2. Описание выходных данных.
      3. Пример составления запроса.
      4. Пример составления ответа.
      5. Возможность отправить свой запрос прямо из графического интерфейса.
      6. После отправки своего запроса должен появляться ответ запроса.

Executor API

1. Авторизация на стороне API происходит по токену, который выдается ЦАА (Центр Авторизации и Аутентификации).
2. Роли устанавливаются ЦАА:
   1. Гость (*Guest*);
   2. Зарегистрированный пользователь (*RegistratedUser*);
   3. Подтвержденный пользователь (*ConfirmedUser*);
   4. Супер пользователь (*SuperUser*).
3. Executor API должен содержать методы:

|  |  |  |
| --- | --- | --- |
| Название метода | | |
| Описание метода | Входные данные | Выходные данные |
| Отправить множество решений на проверку (*SendSubmits*) | | |
| Метод должен отсылать на проверку множество решений и выдавать в соответствующем порядке множество результатов. | Список решений задач  Тип: список объектов типа Submit | Список идентификаторов результатов  Тип: список целых чисел |
| Перепроверить множество решений (*RejudgeSubmits*) | | |
| Метод должен переотправлять все заданные решения, которые уже имеются в базе, заново на проверку. | Список идентификатором решений  Тип: список целых чисел | Список идентификаторов результатов  Тип: список целых чисел |
| Получить результат решения (*GetSubmitResult*) | | |
| Метод должен возвращать результат решения по заданному идентификатору. | Идентификатор решения  Тип: целое число | Результат решения  Тип: класс SubmitResult |
| Получить множество результатов (*GetSubmitResults*) | | |
| Метод должен соответствующий возвращать список результатов по заданному списку идентификаторов этих результатов. | Список идентификаторов результатов  Тип: список целых чисел | Список результатов решений  Тип: список объектов типа SubmitResult |
| Получить список поддерживаемых языков программирования (*GetLangs*) | | |
| Метод должен возвращать список языков программирования. | - | Список языков программирования  Тип: список строковых объектов |

1. Ограничение по ролям:

|  |  |  |
| --- | --- | --- |
| Метод | Роли | Ограничения |
| Получить множество задач (*GetProblems*) | Гость (*Guest*) | Задачи из открытого контеста |
| Зарегистрированный пользователь (*RegistratedUser*) | Задачи из открытого контеста |
| Подтвержденный пользователь (*ConfirmedUser*) | Задачи из открытого или доступного только ему контеста |
| Супер пользователь (*SuperUser*) | Полный доступ |
| Отправить множество решений на проверку (*SendSubmits*) | Гость (*Guest*) | Нет прав |
| Зарегистрированный пользователь (*RegistratedUser*) | Нет прав |
| Подтвержденный пользователь (*ConfirmedUser*) | Проверка решений задач из открытого или доступного только ему контеста |
| Супер пользователь (*SuperUser*) | Полный доступ |
| Перепроверить множество решений (*RejudgeSubmits*) | Гость (*Guest*) | Нет прав |
| Зарегистрированный пользователь (*RegistratedUser*) | Нет прав |
| Подтвержденный пользователь (*ConfirmedUser*) | Проверка решений задач из открытого или доступного только ему контеста |
| Супер пользователь (*SuperUser*) | Полный доступ |
| Получить результат решения (*GetSubmitResult*) | Гость (*Guest*) | Только открытая информация результата (*SubmitResult*) по задачам из открытых контестов |
| Зарегистрированный пользователь (*RegistratedUser*) | Только открытая информация результата (*SubmitResult*) по задачам из открытых контестов |
| Подтвержденный пользователь (*ConfirmedUser*) | Открытая информация результата (*SubmitResult*) по задачам из открытых и доступных ему контестов, либо доступная информация результата по своим посылкам (*Submit*) |
| Супер пользователь (*SuperUser*) | Полный доступ |
| Получить множество результатов (*GetSubmitResults*) | Гость (*Guest*) | Только открытая информация результатов (*SubmitResult*) по задачам из открытых контестов |
| Зарегистрированный пользователь (*RegistratedUser*) | Только открытая информация результатов (*SubmitResult*) по задачам из открытых контестов |
| Подтвержденный пользователь (*ConfirmedUser*) | Открытая информация результатов (*SubmitResult*) по задачам из открытых и доступных ему контестов, либо доступная информация результатов по своим посылкам (*Submit*) |
| Супер пользователь (*SuperUser*) | Полный доступ |
| Получить ссылку на условие задачи (*GetStatementUrl*) | Гость (*Guest*) | Задача из открытого контеста |
| Зарегистрированный пользователь (*RegistratedUser*) | Задача из открытого контеста |
| Подтвержденный пользователь (*ConfirmedUser*) | Задача из открытого или доступного только ему контеста |
| Супер пользователь (*SuperUser*) | Полный доступ |
| Получить список поддерживаемых языков программирования (*GetLangs*) | Гость (*Guest*) | Полный доступ |
| Зарегистрированный пользователь (*RegistratedUser*) | Полный доступ |
| Подтвержденный пользователь (*ConfirmedUser*) | Полный доступ |
| Супер пользователь (*SuperUser*) | Полный доступ |

1. Программная реализация продукта должна использовать готовые системы:
   1. Система тестирования (*Bunsan API*);
   2. Система авторизации и аутентификации (*CAA API*);
   3. Систему ролей *CAA*;
   4. Система управления архивом (*Archive API*);
   5. *ORM Entity Framework*.
2. Интерфейс спецификации
   1. Интерфейс спецификации должен генерироваться с помощью готового инструмента Swagger.
   2. Интерфейс для каждого метода должен содержать:
      1. Описание входных данных.
      2. Описание выходных данных.
      3. Пример составления запроса.
      4. Пример составления ответа.
      5. Возможность отправить свой запрос прямо из графического интерфейса.
      6. После отправки своего запроса должен появляться ответ запроса.