API документація для можливості роботи із сайтом за допомогою сторонніх програм.

Зміст

1. Авторизація

2. Вакансії

3. Відгуки

***Авторизація***

1. Авторизація виконується на сторінці API : <https://auth-api.robota.ua>
2. Використовуємо endpoint ***POST/Login***
3. Використовуємо модель для входу, куди вводимо дані про користувача:

{

 "username": "string",

 "password": "string"

 }

* username - логін
* password - пароль

Для виконання запитів на [https://employer-api.robota.ua](https://employer-api.robota.ua/) необхідно в заголовках запиту надсилати також заголовок **Authorization** зі значенням **Bearer** <*token*>

<*token*> - токен, отриманий в результаті **Авторизації**

**Вакансії**

Для створення та редагування вакансії використовуємо: [https://employer-api.robota.ua](https://employer-api.robota.ua/).

* **POST/vacancy/add** – ендпоінт для створення та редагування вакансії. Для створення необхідно передавати ID = 0, для редагування – вказувати явно у полі ID ідентифікатор існуючої вакансії.
* **POST/vacancy/state/{id}** - Використовується для зміни статусу вакансії (Опубліковано, Чернетку, Завершено, Видалено… і тд).
	+ Для зміни статусу вакансії необхідно в урлі після id вакансії передавати статус стрінгою
	+ curl -X POST "<https://employer-api.robota.ua/vacancy/state/> vacancyId?state=string"
		- Deleted – Видалити вакансії
		- Closed – Завершити вакансію
		- Publicated – Опублікувати вакансію
		- NotPublicated – Чернетка.

***Відгуки***

Для роботи з відгуками використовуємо: [https://employer-api.robota.ua](https://employer-api.robota.ua/)

* **POST/apply/list** - отримання списку всіх відгуків
* Використовуємо модель, куди вводимо дані:

{

 "vacancyId": int,

 "folderId": int,

 "page": int,

 "filter": "string"

}

* vacancyId (id вакансии): 0 – відгуки без прив'язки, -1 – всі відгуки, ID вакансії – відгуки щодо конкретної вакансії.
* folderId (папка розміщення відгуків): 0- всі резюме та відгуки («All»), 1- Надіслані відгуки («Sended»), 2- Цікаві («Selected»), 3- Запрошений («Invited») 4- Відмовлено («Refused»), 5- Збережені із бази («SaveCvDb»), 6- Нецікаві («Deleted»), 8- Нові («New»)
* page: номер страницы.
* filter: поиск по откликам.
* **POST/apply/view{id}** - перегляд відгуку
* Вводимо дані:

"id" (path): int.

"resumeType" (query): "string".

* id – id відгуку.
* resumeType (тип резюме): 1 – відгук прикріпленим файлом («AttachedFile»), 2 – резюме створене на сайті («Notepad»), 3 – збережені з бази резюме («Selected»), 4 – відгук без резюме («NoCvApply»).
* **GET /apply/getfile/{id}** - скачування відгуків.
* Вводимо дані:

"id" (path): int.

"resumeType" (query): "string".

* id – id відгуку.
* resumeType (тип резюме): 1 – відгук прикріпленим файлом («AttachedFile»), 2 – резюме створене на сайті («Notepad»), 3 – збережені з бази резюме («Selected»), 4 – відгук без резюме («NoCvApply»).

# **English Documentation**

API documentation for the ability to work with the site through third-party programs.

***Content***

1. *Login*
2. *Vacancy*
3. *Apply*

##

## ***Login***

1. Authorization is performed on the API page:<https://auth-api.robota.ua>
2. Use ***POST/Login***  endpoint
3. Use the model to enter, where enter user information:

{

 "username": "string",

 "password": "string"

 }

You must send token in the **Authorization** header when making requests to [https://employer-api.robota.ua](https://employer-api.robota.ua/) (example: **Authorization: Bearer <token>**)

*<token>* - token received as a result of ***POST/Login***

***Vacancy***

For create and edit a vacancy need use -  [https://employer-api.robota.ua](https://employer-api.robota.ua/).

* **POST/vacancy/add** – this endpoint for creating and editing vacancy. For creation, it is necessary to transfer ID = 0, for editing, specify explicitly the identifier of an existing vacancy in the ID field.
* **POST/vacancy/state/{id}** - use to change vacancy status (Publicated, Draft, Closed, Deleted … etc).
	+ For change the status of a vacancy, it is necessary to post the string status in the url after the id of the vacancy
	+ curl -X POST "<https://employer-api.robota.ua/vacancy/state/> vacancyId?state=string"
		- Deleted – Delete a vacancy
		- Closed – Close a vacancy
		- Publicated – Publicate a vacancy
		- NotPublicated – Draft

***Apply***

For work with applies need use -<https://employer-api.robota.ua>

* **POST/apply/list** - getting a list of applies on vacancy
* Use the model where enter the data:

{

 "vacancyId": int,

 "folderId": int,

 "page": int,

 "filter": "string"

}

* vacancyId (id vacancy): 0 – applies not tied to vacancies, -1 – all applies, ID vacancy – getting applies on vacancy with ID XXXXXX.
* folderId (folder for posting applies): 0- all resumes and applies («All»), 1- sent applies («Sended»), 2- interesting («Selected»), 3- invited («Invited») 4- refused («Refused»), 5- saved from the resume database («SaveCvDb»), 6- deleted («Deleted»), 8- new («New»), 9- black list («BlackList»).
* page: page number.
* filter: search on applies.
* **POST/apply/view{id}** - view apply
* Enter data:

"id" (path): int.

"resumeType" (query): "string".

* id – id of the application.
* resumeType (type of a cv): 1 – application via file («AttachedFile»), 2 – application via Profile («Notepad»), 3 – saved cv from CVDB («Selected»), 4 – application without CV(by phone) («NoCvApply»).
* **GET /apply/getfile/{id}** - application download
* Enter data:

"id" (path): int.

"resumeType" (query): "string".

* id – id of the application.
* resumeType (type of a cv): 1 – application via file («AttachedFile»), 2 – application via Profile («Notepad»), 3 – saved cv from CVDB («Selected»), 4 – application without CV(by phone) («NoCvApply»).

#  **Документация на русском языке**

API документация для возможности работы с сайтом посредством сторонних программ.

***Содержание***

1. *Авторизация*
2. *Вакансии*
3. *Отклики*

***Авторизация***

1. Авторизация выполняется на странице API :  <https://auth-api.robota.ua>
2. Используем endpoint ***POST/Login***
3. Используем модель для входа, куда вводим данные о пользователе:
{

               "username": "string",

               "password": "string"

            }

* username - логин
* password - пароль

Для выполнения запросов на [https://employer-api.robota.ua](https://employer-api.robota.ua/) необходимо в заголовках запроса отправлять также заголовок **Authorization** со значением **Bearer** <*token*>

<*token*> - токен, полученный в результате ***Авторизации***

***Вакансии***

Для создания и редактирования вакансии  используем: [https://employer-api.robota.ua](https://employer-api.robota.ua/).

* **POST/vacancy/add** – эндпоинт для создания и редактирования вакансии. Для создания необходимо передавать ID = 0, для редактирования - указывать явно в поле ID идентификатор существующей  вакансии.
* **POST/vacancy/state/{id}** - Используется для изменения статуса вакансии (Опубликована, Черновик, Завершена, Удалена … и тд).
	+ Для изменения статуса вакансии необходимо в урле после id вакансии передавать статус стрингой
	+ curl -X POST "<https://employer-api.robota.ua/vacancy/state/> vacancyId?state=string"
		- Deleted – Удалить вакансию
		- Closed – Закрыть вакансию
		- Publicated – Опубликовать вакансию
		- NotPublicated – Черновик.

***Отклики***

Для работы с откликами  используем: [https://employer-api.robota.ua](https://employer-api.robota.ua/)

* **POST/apply/list** - получение списка всех откликов
* Используем модель, куда вводим данные:

{

  "vacancyId": int,

  "folderId": int,

  "page": int,

  "filter": "string"

}

* vacancyId (id вакансии): 0 – отклики без привязки, -1 – все отклики, ID вакансии – отклики по конкретной вакансии.
* folderId (папка размещения откликов): 0- все резюме и отклики («All»), 1- Присланные отклики («Sended»), 2- Интересные («Selected»), 3- Приглашен («Invited»)  4- Отказан («Refused»), 5- Сохраненные из базы («SaveCvDb»), 6- Неинтересные («Deleted»), 8- Новые («New»)
* page: номер страницы.
* filter: поиск по откликам.
* **POST/apply/view{id}** - просмотр отклика
* Вводим данные:

"id" (path): int.

"resumeType" (query): "string".

* id – id отклика.
* resumeType (тип резюме): 1 – отклик прикрепленным файлом («AttachedFile»), 2 –резюме созданное на сайте («Notepad»), 3 – сохраненные из базы резюме («Selected»), 4 – отклик без резюме («NoCvApply»).
* **GET /apply/getfile/{id}** - скачивание откликов.
* Вводим данные:

"id" (path): int.

"resumeType" (query): "string".

* id – id отклика.
* resumeType (тип резюме): 1 – отклик прикрепленным файлом («AttachedFile»), 2 –резюме созданное на сайте («Notepad»), 3 – сохраненные из базы резюме («Selected»), 4 – отклик без резюме («NoCvApply»).