

REST API.

Get players list

URL	/rest/players
Method	GET
URL Params	Optional: String name, String title, Race race, Profession profession, Long after, Long before, Boolean banned, Integer minExperience, Integer maxExperience, Integer minLevel, Integer maxLevel, PlayerOrder order, Integer pageNumber, Integer pageSize
Data Params	None
Success Response	Code: 200 OK Content: [{ “id”:[Long], “name”:[String], “title”:[String], “race”:[Race], “profession”:[Profession], “birthday”:[Long], “banned”:[Boolean], “experience”:[Integer], “level”:[Integer], “untilNextLevel”:[Integer] }, ...]
Notes	<p>Поиск по полям name и title происходит по частичному соответствию. Например, если в БД есть игрок с именем «Камираж», а параметр name задан как «ир» - такой игрок должен отображаться в результатах (Камираж).</p> <p>pageNumber – параметр, который отвечает за номер отображаемой страницы при использовании пейджинга. Нумерация начинается с нуля</p> <p>pageSize – параметр, который отвечает за количество результатов на одной странице при пейджинге</p>

Get players count

URL	/rest/players/count
Method	GET
URL Params	Optional: String name, String title, Race race, Profession profession, Long after, Long before, Boolean banned, Integer minExperience, Integer maxExperience, Integer minLevel, Integer maxLevel
Data Params	None
Success Response	Code: 200 OK Content: Integer
Notes	

Create player

URL	/rest/players
Method	POST
URL Params	None
Data Params	<pre>{ "name": [String], "title": [String], "race": [Race], "profession": [Profession], "birthday": [Long], "banned": [Boolean], --optional, default=false "experience": [Integer] }</pre>
Success Response	<p>Code: 200 OK</p> <p>Content: {</p> <pre> "id": [Long], "name": [String], "title": [String], "race": [Race], "profession": [Profession], "birthday": [Long], "banned": [Boolean], "experience": [Integer], "level": [Integer], "untilNextLevel": [Integer] }</pre>
Notes	<p>Мы не можем создать игрока, если:</p> <ul style="list-style-type: none">- указаны не все параметры из Data Params (кроме banned);- длина значения параметра "name" или "title" превышает размер соответствующего поля в БД (12 и 30 символов);- значение параметра "name" пустая строка;- опыт находится вне заданных пределов;- "birthday": [Long] < 0;- дата регистрации находится вне заданных пределов. <p>В случае всего вышеперечисленного необходимо ответить ошибкой с кодом 400.</p>

Get player

URL	/rest/players/{id}
Method	GET
URL Params	id
Data Params	None
Success Response	Code: 200 OK Content: { “id”:[Long], “name”:[String], “title”:[String], “race”:[Race], “profession”:[Profession], “birthday”:[Long], “banned”:[Boolean], “experience”:[Integer], “level”:[Integer], “untilNextLevel”:[Integer] }
Notes	Если игрок не найден в БД, необходимо ответить ошибкой с кодом 404. Если значение id не валидное, необходимо ответить ошибкой с кодом 400.

Update player

URL	/rest/players/{id}
Method	POST
URL Params	id
Data Params	{ "name":[String], --optional "title":[String], --optional "race":[Race], --optional "profession":[Profession], --optional "birthday":[Long], --optional "banned":[Boolean], --optional "experience":[Integer] --optional }
Success Response	Code: 200 OK Content: { "id":[Long], "name":[String], "title":[String], "race":[Race], "profession":[Profession], "birthday":[Long], "banned":[Boolean], "experience":[Integer], "level":[Integer], "untilNextLevel":[Integer] }
Notes	Обновлять нужно только те поля, которые не null. Если игрок не найден в БД, необходимо ответить ошибкой с кодом 404. Если значение id не валидное, необходимо ответить ошибкой с кодом 400.

Delete player

URL	/rest/players/{id}
Method	DELETE
URL Params	id
Data Params	
Success Response	Code: 200 OK
Notes	Если игрок не найден в БД, необходимо ответить ошибкой с кодом 404. Если значение id не валидное, необходимо ответить ошибкой с кодом 400.