Strona: <http://projektnet.mini.pw.edu.pl/NTime>

Administrator:

Login/Email: admin@ntime.pl

Hasło: admin1

## Założenia:

* Są 4 role:
  + Player - P
  + Organizer - O
  + Moderator - M
  + BustModerator – Moderator z odebranymi uprawnieniami
  + Administrator - A
* Organizator ma dostęp do jakiegoś zasobu związanego z zawodami tylko wtedy, gdy jest powiązany z danymi zawodami i może edytować je jak flaga w zawodach "ModeratorEditLock" jest na false
* Player może zapisywać się na zawody i modyfikowac owy wpis o ile operacje te nie są wykonywane po "SignUpEndDate" w obiekcie zawodów
* Player może mieć pełny wgląd tylko na własne wyniki
* Jeśli w ścieżce jest napisane "1" to oznacza, że trzeba tam podać id do danego zasobu (liczba dodatnia, całkowita), chyba że jest poprzedzony przez "/FromCompetition/", "/IntoCompetition/", "/ByCompetition/", "/SetCompetition/" lub "/UnsetCompetition/" wtedy należy podać id do zawodów, a jeśli "/FromPlayerAccount/" to podajemy id do konta zawodnika.
* W przypadku kontrolerów "AdministratorAccount", "ModeratorAccount" id to ciąg znaków
* Api obsługuje Json o ile Content-Type jest nie wskazuje inaczej
* Jeśli adres kończy się na "?ItemsOnPage=10&PageNumber=0" to oznacza, że wynik jest stronicowany, w odpowiedzi dostanie się całkowitą liczbę obiektów i tablicę przeznaczoną do wyświetlenia na wskazanej stronie (Uwaga: strony liczymy od zera). Zwracana jest struktura z polem TotalCount – int i Items – tablica.
* Hasła powinny zawierać co najmniej 8 znaków w tym co najmniej jedną cyfrę i mały znak
* Można ignorować pole id przy tworzeniu nowych zasobów, ale nie można go pominąć, może być dowolne a najlepiej –1
* Jeśli przy polu w wypisanej strukturze danych jest "\*" to pole nie może przyjąć null-a, więc nie może być pominięte
* 404 jeśli id-ki się nie zgadzają
* Czas liczymy w sekundach jako zmienna typu decimal (oznaczone przy niektórych zmiennych)

## Logowanie:

### POST Token

Mogą: wszyscy

Content-Type: application/x-www-form-urlencoded

Body: grant\_type=password&username=admin&password=admin1

Przykładowa odpowiedź:

{

"access\_token": "BMxniHiUjhexqWm1RwexXkAK9Pq70n4-QZNdjRgle1gL15a1mrGLM3lHhjq0DOCaWsGh0T2Nr9KWqStfIQuPxD8iYDDYDKzRbU2Rv\_sXPeEbBDPpvgAVsOnL9aiWZsG2ve811uGBiH0l0Nfko50uQDQcMAuHAbBQjXuwLChO-up5DTLgrHozIlc8han4U-t1nDWeiVd1vPqPcaJi6TvFm6BfNGjZhrm8G2xAoWF2C60894Qjs-VqWJlER9eSBLk6B6TfPFHrEpwHixf5olC2DQjpv1Jd-3PFNu2RQTnlbqMCl5pjJXasYGuzPGWW5n8J58D-tC7lxOYXqIKZTHfGKzAE2oYsN8kMxmC0QCc0tqgnBJvbGiJ-w0t8JktmicFKpJR0zUlC3YLQryPx-HvcvLnaIwXpenVATYosSDoCmvfT6kpy4Sg34qQYYw7dKf8RvJraxbnlqYmJqrZ6L9uD9oQ6CHHAj7Khsfah6KXSSexWZvL4S3vE20-TEIfMiC09UaSzANhDaYXIAMB7\_gJo0g",

"token\_type": "bearer",

"expires\_in": 1209599,

"userName": "admin",

".issued": "Thu, 28 Dec 2017 20:42:00 GMT",

".expires": "Thu, 11 Jan 2018 20:42:00 GMT"

}

W każdym następnym żądaniu należy umieścić w nagłówku:

Authorization: Bearer BMxniHiUjhexqWm1Rwe....

## Kontroler Account:

### POST api/Account/Logout

Mogą: wszyscy zalogowani

### POST api/Account/ChangePassword

Mogą: wszyscy zalogowani

Body:

* \*OldPassword
* \*NewPassword
* \*ConfirmPassword

### POST api/Account/Register

Mogą: wszyscy

Body:

* \*Email
* \*Password
* \*ConfirmPassword

### GET api/Account/Role

Mogą: wszyscy zalogowani

Zwraca id konta (string), email i przypisaną role

## Kontroler Competition

### Model danych

* \*Id
* \*Name
* \*EventDate
* SignUpEndDate - jeśli null to nie ma deadline-u na zapisy
* Description
* Link
* Organizer - string
* City
* \*OrganizerEditLock – nie da się zmodyfikować to poprzez PUT i POST (przy tworzeniu), po stworzeniu nowych zawodów ustawiony jest na false

### GET /api/Competition?ItemsOnPage=10&PageNumber=0

Mogą: wszyscy

Pobiera stronicowaną listę zawodów sortowaną po czasie rozpoczęcia

### GET /api/Competition/FromPlayerAccount/1?ItemsOnPage=10&PageNumber=0

Mogą: A, M, P

Pobiera stronicowaną listę zawodów, na które dany użytkownik jest zapisany.

### GET /api/Competition/1

Mogą: wszyscy

Pobiera dane jednych zawodów

### PUT /api/Competition/1

Mogą: A, O

Modyfikuje zawody

### POST /api/Competition

Mogą: A

Tworzy nowe zawody

### POST /api/Competition/1/OrganizerLock/true

### POST /api/Competition/1/OrganizerLock/false

Mogą: A

Nic nie trzeba podawać nic w body

Ustawia flagę OrganizerEditLock

## Kontrolery AgeCategory, Distance, ExtraPlayerInfo

W miejsce "<attr>" wstaw AgeCategory, Distance lub ExtraPlayerInfo

### Model danych AgeCategory

* \*Id
* \*Name
* \*YearFrom
* \*YearTo

### Model danych Distance

* \*Id
* \*Name
* \*Length
* \*DistanceTypeId – int: może przyjąć
  + 0 – na dystans
  + 1 – na okrążenia - wtedy LapsCunt jest istotne
  + 2 – na czas w pętli - wtedy TimeLimit jest istotne
* \*LapsCount
* \*TimeLimit - decimal

### Model danych ExtraPlayerInfo

* \*Id
* \*Name
* \*ShortName

### GET api/<attr>/FromCompetition/1

Mogą: wszyscy

Wylistuje dane z danych zawodów

### GET api/<attr>/1

Mogą: wszyscy

Pobiera obiekt o podanym id

### PUT api/<attr>/1

Mogą: A, O

Modyfikuje obiekt o podanym id

### POST api/<attr>/IntoCompetition/1

Mogą: A, O

Tworzy nowy obiekt w danych zawodach

### DELETE api/<attr>/1

Usuwa obiekt o podanym id

Mogą: A

## Kontroler Player

Dotyczy "wpisu" na konkretne zawody i ich ewentualne wyniki, a nie konta zawodników

### Model danych filtru

Jeśli jest nullem to filtr nie jest brany pod uwagę

* \*PlayerSort - int
  + 0 - imię
  + 1 – nazwisko
  + 2 – klub
  + 3 – numer startowy
  + 4 – czas startu
  + 5 – kategoria
  + 6 – data urodzenia
* \*DescendingSort - bool
* Query – string, zaawansowane wyszukiwanie po napisie
  + "30 - 50" wybierze numery startowe z przedziału
  + "30, 40, 45, 50" wybierze podane numery startowe
  + "jan kow MINI" wybierze wpisy, w których zawierają się podane po spacji selektory np. Jan Kowalski z wystansu MINI i Jan Kowalik z wystansu MINI+
* Men – bool
* WithoutStartTime - bool
* Invalid - bool
* CompleatedCompetition - bool
* HasVoid - bool
* Distance – id
* AgeCategory – id
* ExtraPlayerInfo – id

### Model danych do publicznej listy (tylko do wglądu)

* \*Id
* \*FirstName
* \*LastName
* \*IsMale - bool
* Team - string
* \*StartNumber - int
* StartTime – decimal (liczone od godziny 00:00)
* FullCategory - string

### Model danych do zapisów

* \*Id
* \*FirstName
* \*LastName
* \*BirthDate
* \*IsMale
* Team
* \*ExtraPlayerInfoId - id
* \*DistanceId - id
* \*CompetitionId -id

### Model danych pełny

* \*Id
* \*FirstName
* \*LastName
* \*BirthDate
* \*IsMale
* Team
* \*StartNumber
* StartTime – decimal (liczone od godziny 00:00)
* \*IsStartTimeFromReader - być może zostanie usunięte
* FullCategory – aktualizuje się automatycznie
* \*LapsCount
* \*Time - decimal
* \*DistancePlaceNumber
* \*CategoryPlaceNumber
* \*CompetitionCompleted - bool
* ExtraPlayerInfoId - id
* DistanceId - id
* CompetitionId -i d
* PlayerAccountId - id

### POST api/Player/TakeSimpleList/FromCompetition/1?ItemsOnPage=10&PageNumber=0

Mogą: wszyscy

W body podać obiekt filtru

Zwraca dane publicznej listy

### POST api/Player/TakeFullList/FromCompetition/1?ItemsOnPage=10&PageNumber=0

Mogą: A, O, M

W body podać obiekt filtru

Zwraca pełne dane

### GET api/Player/TakeFullList/FromPlayerAccount/1?ItemsOnPage=10&PageNumber=0

Mogą: A, M, P

Zwraca zapisy (pełne dane) z odbytych zawodów i zawodów na które się zapisało podanego zawodnika

### GET api/Player/FromPlayerAccount/1/FromCompetition/1

Mogą: A, M, P

Zwraca pełne dane zarejestrowane użytkownika dla wybranych zawodów

### GET api/Player/1

Mogą: A, O, M, P

Zwraca pełne dane o podanym id

### PUT api/Player/1

Mogą: A, O, M

Modyfikuje pełne dane

### DELETE api/Player/1

Mogą: A, O, M, P

Kasuje zapis

### GET api/Player/Register/1

Mogą: A, O, M, P

Pobiera zapis w o podanym id

### PUT api/Player/Register/1

Mogą: P

Modyfikuje zapis

### POST api/Player/Register/IntoCompetition/1

Mogą: wszyscy

Zapisuje się na dane zawody

## Kontroler PlayerAccount

### Model Danych

* \*Id
* \*FirstName
* \*LastName
* \*BirthDate
* \*IsMale
* Team
* PhoneNumber
* \*EMail

### GET api/PlayerAccount/Search/Filtr?ItemsOnPage=10&PageNumber=0

### GET api/PlayerAccount/Search?ItemsOnPage=10&PageNumber=0

Może: A

Zwraca stronicowaną listę kont zawodników z lub bez filtru ("Filtr") który jest stringiem

### GET api/PlayerAccount

Może: P

Zwraca dane aktualnie zalogowanego zawodnika

### GET api/PlayerAccount/1

Może: A, P

Zwraca dane zawodnika o podanym id

### GET api/PlayerAccount/FromCompetition/1?ItemsOnPage=10&PageNumber=0

Może: A

Zwraca listę kont zawodników z których zapisano się na konkretne zawody

### PUT api/PlayerAccount/1

Może: A, P

Edycja

## Kontroler OrganizerAccount

Wszystkie operacje może tylko administrator

### Model Danych

* \*Id
* \*FirstName
* \*LastName
* \*PhoneNumber - string
* \*EMail
* CompetitionDtos – tablica zawodów do których jest przypisanych (model danych zawodów), przy wylistowaniu null, pole pomijane przy dodawaniu i modyfikacjach

### GET api/OrganizerAccount/Search/Filtr?ItemsOnPage=10&PageNumber=0

### GET api/OrganizerAccount/Search?ItemsOnPage=10&PageNumber=0

Zwraca stronicowaną listę organizatorów z lub bez filtru ("Filtr") który jest stringiem

### GET api/OrganizerAccount/ByCompetition/1

Zwraca organizatorów przypisanych do danych zawodów

### GET api/OrganizerAccount/1

Pojedynczy organizator o podanym id, z polem CompetitionDtos

### PUT api/OrganizerAccount/1

Edycja

### POST api/OrganizerAccount

Stworzenie nowego organizatora z losowym hasłem. Hasło nie jest zwracane.

### DELETE api/OrganizerAccount/1

Usunięcie organizatora

### POST api/OrganizerAccount/1/PasswordReset

Reset hasła. Nowe, losowe hasło jest zwracane.

### POST api/OrganizerAccount/1/SetCompetition/1

Ustawia organizatorowi o podanym id dostęp do zawodów o podanym id

### POST api/OrganizerAccount/1/UnsetCompetition/1

Analogicznie

## Kontroler ModeratorAccount

Wszystkie operacje może tylko administrator

Model danych taki jak w akcji "Role"

Id to string, (w prototypach oznaczony jako "aaa")

### GET api/ModeratorAccount

Pobiera listę

### POST api/ModeratorAccount

Tworzy nowe kont,

W body to samo jak przy rejestracji

### DELETE api/ModeratorAccount/aaa

Usuwa

### POST api/ModeratorAccount/aaa/PasswordReset

Zmienia hasło na losowe i je zwraca

### POST api/ModeratorAccount/aaa/Bust

Zmienia role na "BustModerator"

### POST api/ModeratorAccount/aaa/Unbust

Analogicznie

## Kontroler AdministratorAccount

Wszystkie operacje może tylko administrator

Analogicznie jak w Moderatorze

### GET api/AdministratorAccount

### POST api/AdministratorAccount

### DELETE api/AdministratorAccount/aaa

### POST api/AdministratorAccount/aaa/PasswordReset