**Документация**

**1. Регистрация на потребители**

Всеки потребител (учител, ученик) в системата трябва да се регистрира в системата самостоятелно с:

* **Email**
* **Password**

Освен това

* Всеки учител е необходимо да избере училището(**SchoolId**) и да въведе името си (**FullName**).
* Всеки ученик е необходимо да избере училище(**SchoolId**) и да въведе името си (**FullName**).

След регистрацията учителите получават роля от администратора на училището(SchoolAdmin), получил своята роля от администратора на приложението(Admin).

*\* Училищата се създават от Администратора на системата(****Admin****), а класовете във всяко училище се създават от администратора на училището (****SchoolAdmin****).*

**2. Models (Database Requirements)**

**Student**

Всеки ученик се регистрира лично в системата.

* **Id –** string, primary key.
* **FullName** – string between 5 and 50 characters long [required].
* **SchoolId** – string unique identifier, foreign key [required].
* **School** – School.
* **Groups** – collection of groups which student participate in.

Учениците участват в различни групи, добавени в тях от техните създатели – учителите.

*\*Ако ученик не е добавен в група от учител(никога не е бил в група), то той може да бъде премахнат от администратора на училището(SchoolAdmin).*

**Teacher**

Всеки учител се регистрира лично в системата. Профилът му не е активен, докато не бъде потвърден и активиран от администратора на системата. Всеки учител може да създава групи и да добавя ученици в тях.

* **Id** – string, primary key.
* **FullName** - string between 5 and 50 characters long [required].
* **SchoolId** – string, unique identifier, foreign key [required].
* **School** – School.
* **Groups** – collection of groups created by teacher.

**School**

* **Id –** string, primary key.
* **Type** – string between 2 and 7 characters long [required]
* **SchoolAdminId?** – string, refer to teacher with SchoolAdmin role(null – school has no administrator yet).
* **Name** – string between 10 and 100 characters long [required].
* **City** – string with length between 3 and 13 [required].
* **Groups** – collection of Group in School.

**Group**

Група може да бъде създадена само от учител.

* **Id** – string, unique identifier, primary key.
* **Name** – string with length between 2 and 50 characters.
* **ShortName** – string between 2 and 7 characters long[required].
* long[required].
* **Description** – string with length maximum length of 150 characters.
* **IconUrl** – string [required].
* **SchoolId** – string, the school identifier in which the group belongs to[required].
* **TeacherId** – string, creator of the group[required].
* **Teacher** - Teacher
* **School** - School
* **Topics** – collection of group topics.
* **Students** – collection of students(IdentityUser-s) attending the group.

**Topic**

Темите се добавят към групите и съдържат информация и материали изложени от учителя за учениците включени в конкретната група.

* **Id** – string, primary key
* **Name** – string between 2 and 40 characters long [required].
* **Description** - string maximum 150 characters long.
* **Contents** – string, collection of contents items where each item separated by ‘new line’.
* **CreatorId** – string, unique identifier of creator(Teacher) of the topic.
* **Groups** – collection of Group(s)
* **Resources** – collection of topic resources.

**Resource**

* **Id** – integer, primary key.
* **Link** – string, link to resource[required].
* **TextToDisplay** – string, between 3 and 30 characters long[required]
* **IconRef** – string[required].

**TopicResource** – many-to-many mapping table.

* **ResourceId** – string, primary key, foreign key [required]
* **TopicId** – string, primary key, foreign key [required]
* **Resource** – Resource
* **Topic** - Topic

**GroupTopic** – many-to-many mapping table.

* **GroupId** – string, primary key, foreign key [required]
* **TopicId** – string, primary key, foreign key [required]
* **Group** – Group
* **Topic** - Topic

**GroupStudent** – many-to-many mapping table.

* **GroupId** – string, primary key, foreign key [required]
* **StudentId** – string, primary key, foreign key [required]
* **Group** – Group
* **Student** - IdentityUser

**3. Roles**

Всички потребители (без Admin) е нужно сами да се регистрират, като активното им добавяне в системата става от потребител с роля на едно ниво над тяхното.

**Admin** – отговаря за:

* Създаването на училище
* и добавя SchoolAdmin към всяко училище.

**SchoolAdmin** – администраторът на училище се добавя от администратора на системата **Admin**, който в самото начало е **неактивиран** потребител с роля **Teacher**. Администраторът на система му променя ролята от Teacher на SchoolAdmin и го **активира**, като му делегира отговорност за:

* Добавянето на учители към училището, което администрира.

**Teacher** – отговаря за:

* Създаването на своите групи.
* Добавя ученици в своите групи.

Всички останали потребители са ученици.

**4. Data Seed**

При първоначално стартиране на приложението се добавят произволни данни в базата от данни, като:

* се добавят ролите;
* създават се потребители;
* потребителите се добавят в роли – един Admin и няколко в роля Teacher;
* добавят се няколко училища.

Оттук насетне Admin потребителят назначава SchoolAdmin потребител за всяко училище, а те на свой ред активират\* учителите на училищата, които администрират. Също така регистрацията за нови учители и ученици в системата е възможна, като те са неактивни докато по-горе стоящата в йерархията роля не ги активира или премахне от системата.