# Email-service documentation

Для запуска почтового клиента

<!-- java e-mail starter-->  
<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-mail</artifactId>  
</dependency>  
<!-- end java e-mail starter-->

public class MailConfig

хранит всю конфигурацию для отправки электронного письма из google-mail

public class EmailVerificationDto {  
 String tokenValue; - тот токен, который будет отправлен пользователю для того, что бы он мог подтвердить свой e-mail  
 String userName;  
 String userLastName;  
 String email;  
}

application.properties + данные того электронного ящика, с которого будет отправлено письмо  
email.password=TestingPrograms!  
email.address=testingProgramsJava@gmail.com

тот на который будет отправляться запрос при нажатии на линк верификации в письме, которое получит пользователь  
verification.link = <http://localhost:8082/users/v1/email-verification>

Логика работы сервиса

1. Сервис слушает сообщения RabbitMQ для отправки писем о получении разрешения (плюс скрин шот)

email.key=email  
email.queue=EMAIL  
email.exchange=Email

В эту очередь приходит сообщение из сервисов School-service, Tsofim-service, Gymnast-service. Очереди приходят когда необходимо отправить письмо пользователю о том, что ацарат бриют получен.

В очереди приходит объект

public class EmailDto {  
 String userName;  
 String userLastName;  
 String childFirstName;  
 String childSecondName;  
 String email;  
 String picture;  
 Services service;  
}

В зависимости от того, какой сервис приходит в обьекте – отправляется то или другое письмо. Методы –

public String sendGymnastEmail(EmailDto mail)

public String sendSchoolEmail(EmailDto mail)

public String sendTsofimEmail(EmailDto mail)

Письмо отправляются пользователю, чей электронный адрес приходит в EmailDto(по сути – это адрес ответственного лица или родителя ребёнка, на которого получено разрешение)

1. Сервис слушает сообщения о подтверждении e-mail

сonf.email.key=confemail  
conf.email.queue=CONFEMAIL  
conf.email.exchange=Email

данное сообщение приходит от user-service в момент регистрации нового пользователя. При получении запроса в очереди, сервис отправляет пользователю письмо с token для подтверждения электронного адреса.

В очереди приходит объект –

public class EmailVerificationDto {  
 String tokenValue;  
 String userName;  
 String userLastName;  
 String email;  
}

Письмо отправляется на электронный адрес пользователя, который нужно подтвердить.

public String sendRegistrationEmail(EmailVerificationDto mail)

1. Ошибка при отправлении письма

Если что то «пошло не так» и отправка письма по какой то причине не прошла, администратору отправляется письмо сообщением об этом.

Письмо отправляется на адрес – указанный в настройках

email.address=testingprogramsjava@gmail.com

Отправляет метод public void emailError(String email, String service, String userFirstName, String userLastName)