A diagram of a computer program

Description automatically generated

**User Registration**

To utilize the service, users are required to complete a registration process using the **POST /api/auth/register** endpoint. Registration necessitates the provision of specific user information in the form of an Authentication request.

**Example of Authentication Request**

{

"email": "example@email.com",

"firstName": "John",

"lastName": "Doe",

"password": "Str0ngP@ssw0rd"

}

Upon successful completion of the registration process, the system will validate whether the provided password adheres to the defined security requirements, and if the email address is unique and not already registered. If successful, the user will receive a confirmation email at the specified email address to confirm their registration and activate their account.

**Email Confirmation**

The email confirmation process is initiated by dispatching an EmailRequest to RabbitMQ. As part of this process, the user receives a partial response containing their user ID. The Email service is responsible for consuming the EmailRequest and executing the necessarcy actions to send the confirmation email.

**Example of EmailRequest**

{

"senderEmail" : "example@email.com",

"subject" : "emailSubject",

"email" : "EmailHtmlBody"

}

The sender's email address, subject, and email body (in HTML format) are included in the EmailRequest. This ensures that users receive a confirmation email with the appropriate content.

By following these steps, users can successfully register and confirm their accounts within the microservice application.

User sends request through api gateway. The user doesn’t have direct access to microservices.

When the user sends the request the jwt token received during authentication is included. This token is validated on api-gateway, and then depending on the result, the request is either forwarder to specified service or the exception is thrown  
  
It is done by implementing AuthenticationFilter

**User Access via API Gateway**

Users interact with the system by sending requests through the API Gateway. The microservices are not directly accessible to users, providing an additional layer of security and control.

**JWT Token Validation**

When a user sends a request through the API Gateway, the request includes a JWT (JSON Web Token) received during the authentication process. This token serves as proof of the user's identity and authorization.

**Token Validation Process**

The API Gateway employs an AuthenticationFilter to validate the JWT token. The filter performs the following steps:

Token Validation: The JWT token included in the request is validated to ensure its authenticity, integrity, and expiration.

Result-based Routing: Depending on the validation result:

If the token is valid, the request is forwarded to the specified microservice responsible for handling the request.

If the token is invalid or has expired, an exception is raised, signaling an authentication failure.

By implementing the **AuthenticationFilter**, the API Gateway ensures that only authenticated and authorized requests are forwarded to the appropriate microservices, enhancing the overall security of the system.