Member Management

**Low Level Design Document**

Contents

[1. ModuleName:Member management 4](#_Toc150771634)

[1.1 Member Registration 4](#_Toc150771635)

[1.1.1 Responsibilities 4](#_Toc150771636)

[ Verify Email 4](#_Toc150771637)

[ Generate Token 4](#_Toc150771638)

[ Validate Token 4](#_Toc150771639)

[ Create Member 4](#_Toc150771640)

[1.1.2 Class Design 4](#_Toc150771641)

[1.1.3 Flow Diagram 4](#_Toc150771642)

[1.1.4 Algorithm Design 5](#_Toc150771643)

[Email Verification 5](#_Toc150771644)

[ Accept Registrant Email Id 5](#_Toc150771645)

[ Verify provided Email id is correct or not already registered 5](#_Toc150771646)

[ After successful verification move to OTP generate page 5](#_Toc150771647)

[Generate OTP 5](#_Toc150771648)

[ Generate OTP 5](#_Toc150771649)

[ Set the OTP Expiry time 5](#_Toc150771650)

[ Send the OTP to the given email Id 6](#_Toc150771651)

[Validate OTP 6](#_Toc150771652)

[ Registrant has to provide OTP sent to his email account 6](#_Toc150771653)

[ Verify that the Registrant provided OTP and system sent OTP are same 6](#_Toc150771654)

[ Verify the expired time 6](#_Toc150771655)

[ If both conditions are met registrant will be registered in the system 6](#_Toc150771656)

[1.1.5 Error Handling 6](#_Toc150771657)

[ Invalid Email Id return an error message 6](#_Toc150771658)

[ OTP validation failure return to OTP generation Page 6](#_Toc150771659)

[1.1.6 Interface Design 6](#_Toc150771660)

[ Method: Post 6](#_Toc150771661)

[ End Point: /api/register 6](#_Toc150771662)

[1.1.7 Security Considerations 6](#_Toc150771663)

[ Use bcrypt or another secure hashing algorithm for storing email 6](#_Toc150771664)

[ Implement secure OTP generation 6](#_Toc150771665)

[1.1.8 Database Schema 6](#_Toc150771666)

[Member Table: 6](#_Toc150771667)

[Columns: 6](#_Toc150771668)

[ MemberId(primary key) 6](#_Toc150771669)

[ EmailId(unique) 6](#_Toc150771670)

[OTP Table: 6](#_Toc150771671)

[Columns: 6](#_Toc150771672)

[ EmailId(Fk) 6](#_Toc150771673)

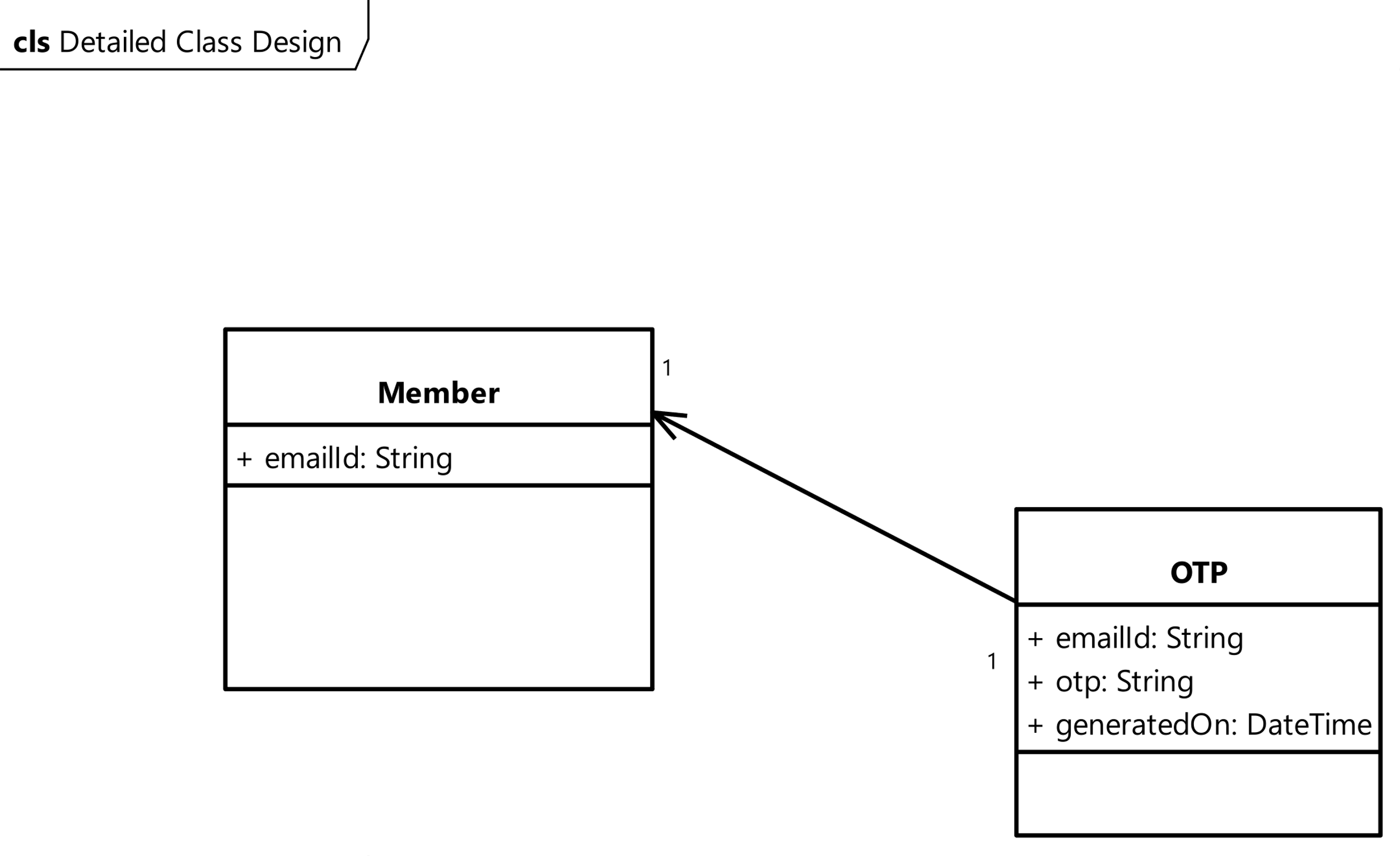
[ OTP(Unique) 7](#_Toc150771674)

[ generatedOn 7](#_Toc150771675)

[1.2 Profile Management 7](#_Toc150771676)

1. ModuleName:Member management
   1. Member Registration
      1. Responsibilities

* Verify Email
* Generate Token
* Validate Token
* Create Member
  + 1. Class Design



* + 1. Flow Diagram
    2. Algorithm Design

Email Verification

* Accept Registrant Email Id
* Verify provided Email id is correct or not already registered
* After successful verification move to OTP generate page

Generate OTP

* Generate OTP
* Set the OTP Expiry time
* Send the OTP to the given email Id

Validate OTP

* Registrant has to provide OTP sent to his email account
* Verify that the Registrant provided OTP and system sent OTP are same
* Verify the expired time
* If both conditions are met registrant will be registered in the system
  + 1. Error Handling
* Invalid Email Id return an error message
* OTP validation failure return to OTP generation Page
  + 1. Interface Design
* Method: Post
* End Point: /api/register
  + 1. Security Considerations
* Use bcrypt or another secure hashing algorithm for storing email
* Implement secure OTP generation
  + 1. Database Schema

Member Table:

Columns:

* MemberId(primary key)
* EmailId(unique)

OTP Table:

Columns:

* EmailId(Fk)
* OTP(Unique)
* generated On
  1. Profile Management