Req-001: createUser API - accountManagementServices

As part of the **accountManagementServices**, we need to create an API that will create a new user and store it in db.

- ➔ Method type: **POST.**

- ➔ **API:**
  - name: **createUser**
  - **api/users/createUser**

- ➔ DB Bucket to be used: **userInfo**.

- ➔ Path parameters should be empty.
- ➔ Parameters to be sent as part of the request body: firstName, lastName, userName, contactNumber, emailId, and password.
- ➔ Other parameters: bio, occupation, createdOn, modifiedOn, lastLogin, and isVerified should store default values.
- ➔ “password” needs to be encoded using bcrypt crypto with saltRounds storing it into the db.
- ➔ A random “verificationCode” needs to be generated and sent to the provided emailId via mail for verification of the user.
- ➔ Generated “verificationCode” needs to be stored as a value of the parameter “verificationCode” in the user object.
- ➔ The “verificationLink” which is sent to the user should contain three parameters: userId, time of creation, and verificationCode.
- ➔ The “verificationLink” should be of below form:
  api/users/verify/{userId}/{time}/{verificationCode}

- ➔ **Response messages:**
  - Response message should be “201 Created” in case the operation is successful.
  - Response message should be “400 Bad Request” in case any required parameter is missing.
  - Response message should be “500 Internal Server Error” in case of any error occurred in storing the data in db.

- ➔ **Email Template:**
  - The Email template should contain the following text in the same order as provided with the font size and color as specified.
    “We got a request for a new account creation with this emailId. Please verify your emailId to activate the account.” (font-size: 20px, type: text)
    “Please click on the following link to proceed.” (font-size: 18px, type: text)
    “Verify” (font-size: 16px, color: blue, type: link)
    “Stamp” (type: text) should be present at the end of the email.