TASK LIST :

1. Set up the project :
   - Create new android studio project
   - Configure the project with Kotlin as the programming language
   - Enable view binding in the project
2. Design a splash screen with a 2000ms delay and then navigate to Login Activity. Use any royalty free graphics of your choice in the splash screen.
3. Design the Login Activity :
   - The login page should contain username(email), password, login button and a textfield/button to navigate to Register Activity
   - Please be free to use any design for the login page while making sure that the design is uniform throughout the application.
   - User proper validation and input type for username(email), password.
4. Design the Register Activity :
   - The register page should have EditTexts for first-name, last-name, email, password and confirm password and a button to register.
   - The register page will also have a register button and a textField to allow navigating back to Login button. Eg : "Already have an account ? **Login**"
   - Please make sure to not make duplicate instances of Login and Register Activity while you are navigating into these activities.
5. Implement registration logic :
   - Handle registration button logic
   - Validate user input(non-empty fields, matching password etc)
   - If registration is successful, navigate to Login Activity
6. Implement Login Logic :
   - Handle login button login
   - Validate user credentials(username and password)
   - If successful , navigate to Main Activity
7. Design the Main Activity :
   - Load user data in the Main Activity
   - Use a dummy circular imageview at the top indicating it as the user, followed by First Name, Last name, Email
   - Implement logout functionality.
   - Design this Main Activity as per your creativity.
8. Utilize view binding across all your views in the activities.
9. Test the application to ensure that login, registration and main activities function as expected and handle edge cases gracefully.
10. Provide inline comments explaining the code and submit the codebase for review.

ADDITIONAL NOTES : Use best practices for code reorganization and readability and emphasize error handling and user friendly messages. Use SQLite