Thank you for taking the time to complete this Odoo developer test for i**lexius**. This test is designed to assess your skills in Odoo module development, business logic implementation, debugging, and API integration.

The goal of this test is to give you a glimpse of the kind of work you will be doing daily at ilexius. Since **Odoo is the core platform we use**, this will also help you get familiar with it if you haven't worked with it extensively before.

Please focus on **clean, maintainable code** and follow **best practices**. If you encounter any challenges, explain your approach and assumptions in a README file.

Task 1: Create a Simple Odoo Module

- Create a new custom Odoo module named test_applicant.
- The module should define a new model test.model with the following fields:
    - name (Char, required), description (Text), active (Boolean, default=True)

      reference_code (Char, computed, unique, format: TEST-0001, TEST-0002, etc.)
    - state (Selection: draft, confirmed, done, default=draft)
- Add a menu item under Settings > Test Applicant, with a form and tree view for this model.
- Create a custom security group (test_applicant.group_manager).
    - Only users in this group should see the menu item and access the model.
    - The group should not be automatically assigned to any user.
- Implement a function that auto-generates reference_code upon record creation.
- Add a button "Confirm", which changes the state to confirm.
- Ensure the reference code is unique for each record.

Task 2: Create an Automated Action and API Endpoint

- Implement a cron job that runs every 5 minutes to automatically mark records as done if they have been in a confirmed state for more than 30 minutes.

- The reference_code should be reset daily.
    - reference_code follows the format: TEST-0001, TEST-0002, etc.
    - The sequence should restart from TEST-0001 every day.
- Create a REST API endpoint (/api/test-model) that:
    - Supports GET in listing all records.
    - Supports POST to create a new record (requires authentication).
    - Uses JSON response format.
    - cover it with tests

Task 3: Add "Login As" Button on User List (Advanced)

- Extend the res.users model to add a new button "Login As" in the user list view.
- When clicked, the system should:
    - Authenticate as the selected user.
    - Redirect to the home dashboard as that user.
- Ensure only admin users (with base.group_system) can use this feature.
- Prevent logging in as superuser (ID=2) for security reasons.

Once you have completed the test, please submit your solution in a **Git repository** (GitHub, GitLab, Bitbucket, or another version control platform).

Your submission should include:

✅ A structured Odoo module with all required files.
✅ A **README** file explaining your implementation, any challenges faced, and how to install/test the module.
✅ A commit history showing your development process (avoid a single large commit).
✅ If possible (not required), a short video demo or screenshots showcasing the working solution.

Thank you again for your time and effort! We look forward to reviewing your work.