Project Description: Rental Management Application

Project Overview:

A Rental Management Application is a comprehensive software solution developed to facilitate the management of rental properties, ensuring that property owners, property managers, and tenants can effectively and efficiently handle all aspects of the rental process. The application serves as a centralized platform for various tasks, including property listing, tenant onboarding, lease management, maintenance requests, and financial transactions.

Key Features and Functionalities:

1. Property Listings:
    - Property owners can create detailed listings of their rental properties, including property descriptions, rental rates, and high-quality photos.
    - Property managers can easily organize and promote available properties for rent.
2. Tenant Management:
    - Property managers can manage tenant information, including tenant applications, references, and lease agreements.
    - Tenants can submit applications, review lease terms, and communicate with property managers.
3. Lease Tracking:
    - The application helps property managers and property owners keep track of lease agreements, including lease terms, renewal dates, and rent payments.
4. Maintenance Requests:
    - Tenants can submit maintenance requests and track their status.
    - Property managers can assign and manage maintenance tasks efficiently.
5. Financial Management:
    - The application enables property owners and managers to handle financial transactions, including rent collection.
6. Communication and Messaging:
    - The application includes communication tools for property managers, property owners, and tenants to exchange messages and notifications.
7. Document Management:
    - Users can store and manage essential documents such as lease agreements, rental applications, and property inspection reports.
8. Reporting and Analytics:
    - The application provides insights into property performance, financial data, occupancy rates, and maintenance history through reports and analytics.
9. Security and Data Protection:
    - Robust security measures ensure the safety of sensitive data, including personal and financial information.

Practice Assignment 6: Design Pattern

I. Factory Method Pattern (40%)

As an extension of Rental Management Application, look at "Maintenance Requests" requirement then complete the following tasks:

Low-Priority Requests:

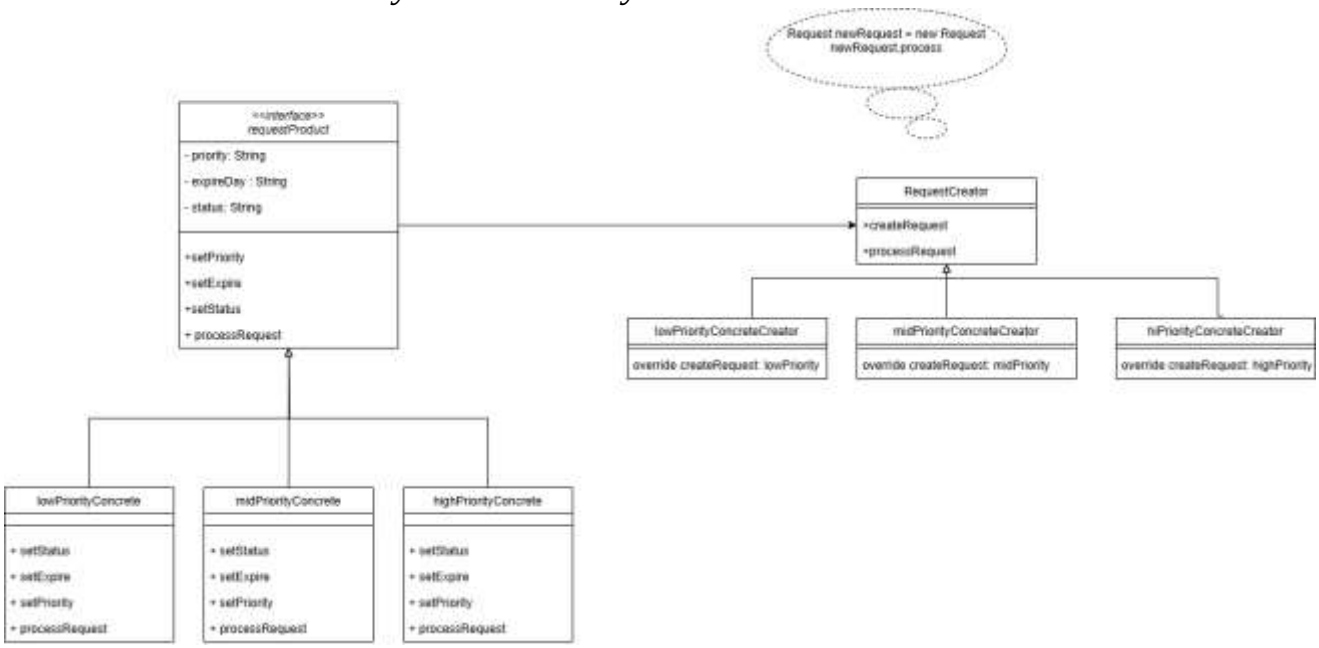
- *setPriority: Priority is set to "Ignore".*
- *setStatus: Status is set to "Done".*
- *setExpire: is set to current day.*
- *processRequest: Print out/Sent the following message "Request denied".*

Medium-Priority Requests:

- *setPriority: Priority is set to "Medium".*
- *setStatus: Status is set to "Accepted".*
- *setExpire: is set to one month from current day.*
- *processRequest: Print out/Sent the following message "Request accept, estimated completion date is [expireDay]".*

High-Priority Requests:

- *setPriority: Priority is set to "Emergency".*
- *setStatus: Status is set to "Accept".*
- *setExpire: is set to current day.*
- *processRequest: Print out/Sent the following message "Emergency request, our Administer will contact you immediately !".*



1. Add one "**User story**" to your Trello which labeled "**Maintenance Requests**" with 2 technical tasks: "**Factory Method Pattern**"; "**Update GitHub**".
2. Set yourself as the sole "**Member**".
3. Set "**Maintenance Requests"** due day to 27/02/2025. It can have any Prioritize.

4. Implement Factory Method Pattern according to Class Diagram.

II. Code Smells (20%)

1. Make sure that your code has meaningful names.

2. Make sure that your methods do not multi task" (More than three should be avoided).

III. Make sure that your board mirrors your current progress. (20%)

~The end~