

Спроектируйте по ООП хранилище лог-файлов

Спроектируйте сохранение лог-файлов в хранилище для их последующего полнотекстового поиска.

Задачей приложения является чтение лог-файлов из различных источников, приведение их к некоторому общему виду и сохранение в заданном хранилище (например, в MS Sql Server).

Процесс чтения лога может состоять из нескольких этапов:

1. Прочитать новые записи с места последнего чтения.
2. Разобрать их и вернуть вызывающему коду.

Классы чтения и сохранения логов являются независимыми шагами импорта лог-файлов и не должны знать друг о друге.

Данные лог файлов читаются из некоторого источника.

Приложение может следить за набором файлов и загружать их содержимое по мере появления в логах новых записей.

Разные типы записей могут обрабатываться по-разному. Например, сообщения с исключениями могут сохраняться в другую таблицу базы данных, разные типы сообщений могут иметь разную логику группировки для пакетного сохранения и т. п.

Хранилища данных могут быть различными и иметь различные интерфейсы.

Может быть введено ограничение на количество записей, которые будут добавляться в хранилище за одну секунду (throttling).

Может быть использован сторонний сервис для сохранения логов.

Требования

Должно быть использовано не менее восьми паттернов проектирования из GoF. Использование паттерна должно быть очевидно либо из названия класса либо на основании соответствующих комментариев в коде.

Должен быть реализован прототип программы, где большая часть функциональности закрыта «заглушками»

(Опционально) Можете подготовить диаграмму классов

* Вы можете использовать книгу (Тепляков С. Паттерны проектирования на платформе .NET. — СПб.: Питер, 2015), где использован похожий проект в качестве сквозного примера по использованию паттернов.

** Вы можете сами придумать свое приложение с любой функциональностью как на основе логгера, так и вообще никак не связанное с ним, однако удовлетворяющее основным требованиям задания.