**Тестовое задание на позицию ASP.NET разработчика**

Реализовать ASP.NET приложение, позволяющее осуществлять мониторинг загрузки файликов в БД.

Приложение состоит из 2 частей:

* Сервис мониторинга файлов;
* Пользовательский интерфейс, позволяющий просматривать результаты мониторинга.

*Сервис мониторинга файлов*

Реализовать сервис, сохраняющий осуществляющий мониторинг заданного каталога.

Сервис запускается и начинает с заданной периодичностью «мониторить» заданный каталог на наличие файлов в следующем формате:

-----------------------------

<Entry>

<!--уникальный цифровой идентификатор записи-->

<id>1</id>

<!--дата создания записи-->

<date>01:01:2010 18:30:52</date>

<!--содержание записи (строка длиной до 1024 символа)-->

<content>Моя заметка</content>

</Entry>

-----------------------------

При обнаружении файла, сервис сохраняет всю необходимую информацию в БД MS SQL и перемещает в заданный каталог обработанных файлов.

ПО должно:

* быть отказоустойчивым;
* журналировать действия;
* быстро работать (файлов в каталоге может быть много);
* иметь скрипты  сборки, собирающие готовый к использованию дистрибутив;
* файл readme.txt, содержащий инструкцию по сборке, настройке и использованию.

*Пользовательский интерфейс*

Необходимо реализовать 2 страницы, через которые можно увидеть результаты работы приложения.

1. Первая страница отображает информацию обо всех файлах, которые попали в папку – ID записи, имя файла, размер, и статус. Статус варьируется в зависимости от результата парсинга – удачно, неудачно, не получилось считать и т.д. На страничке должна быть возможность фильтрации данных по различным полям – имя, ID, статус.
2. На вторую страничку можно перейти, кликнув по ID записи или имени файла. На данной страничке отображается информация о записи – ID, дата создания и содержание записи или же сообщение о некорректном формате и т.д. Должна присутствовать кнопка “Back”, по нажатию на которую, происходит возврат на предыдущую страницу и восстанавливается состояние данной страницы – номер страницы в гриде, порядок сортировки, фильтры и т.д.

Веб-приложение должно быть реализовано при помощи технологии ASP.NET Web API. Запрещается использование сторонних готовых решений, которые полностью или частично реализуют один или несколько описанных выше пунктов. Допускается использование вспомогательных сторонних библиотек – jQuery, Nlog, knockout, log4net и т.д. Для отображения информации с использованием grid’ов также можно использовать сторонние контролы – InGrid, jQueryGrid, Flexgrid и др.