# Министерство образования и науки Российской Федерации

# Федеральное государственное бюджетное образовательное учреждение

# высшего профессионального образования

# «Ижевский государственный технический университет имени М. Т. Калашникова»

# Кафедра «Программное обеспечение»

Отчет по лабораторной работе №2

по дисциплине «Конструирование ПО»

тема «Автоматизированная система мониторинга цен конкурентов»

Выполнил

студент гр. Б08-191-2 С. А. Ившин

Принял В.Г. Власов

Ижевск 2016

СОДЕРЖАНИЕ

1.ВВЕДЕНИЕ

1.1 Назначение системы

1.2 Область применения системы

1.3 Определения, акронимы, аббревиатуры

1.4 Ссылки

1.5 Обзор системы

2. ОБЩЕЕ ОПИСАНИЕ СИСТЕМЫ

2.1 Системный контекст

2.2 Режимы и состояния системы

2.3 Основные функциональные возможности системы

2.4 Основные условия системы

2.5 Основные ограничения системы

2.6 Характеристики пользователя

2.7 Допущения и зависимости

2.8 Оперативные сценарии

3. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ, УСЛОВИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ

3.1 Физические

3.1.1 Конструкция

3.1.2 Износостойкость

3.1.3 Адаптируемость

3.1.4 Условия окружающей среды

3.2 Рабочие характеристики системы

3.3 Безопасность системы

3.4 Информационный менеджмент

3.5 Работа системы

3.5.1 Эргономика системы

3.5.2 Ремонтопригодность системы

3.5.3 Надежность системы

3.6 Стратегия и регулирование

3.7 Устойчивость жизненного цикла системы

4. ИНТЕРФЕЙСЫ СИСТЕМЫ

5. КЛАССЫ КОДИРОВАНИЯ

5.1 UML схема.

5.2 Схема соответствий

1. ВВЕДЕНИЕ

1.1 Назначение системы

Основное назначение системы является повышение конкурентоспособности компании за счет автоматического мониторинга интернет магазинов конкурентов.

1.2 Область применения системы

Система: Автоматизация мониторинга продуктов интернет магазинов. Система должна преобразовывать страницы интернет магазинов в базу товаров, обновлять данные по товарам через указанный промежуток времени, а так же предоставлять пользователям возможность, сравнения цен товара в различных магазинах на текущую или выбранную в ручную дату. Преобразование html страниц необходимо выполнить по разделам - каталогам, а так же по товарам в этих каталогах. В системе необходимо предусмотреть такие группы пользователей системы: пользователь, редактор, администратор.

Данная система будет применятся для решения следующих задач:

- повышение конкурентоспособности за счет оперативного изменения цен на товар;

- анализ рентабельности продукции за счет изменения цены и количества товаров в регионе;

- предоставление данных для анализа движения рынка.

Преимущества системы заключаются в следующих пунктах:

- Автоматический сбор данных по товару, не требующий наличия пользователя.

- Актуальность данных в системе.

- Отсутствие ограничения на количество просматриваемых товаров.

1.3 Определения, акронимы, аббревиатуры

Пользователь. Конечный пользователь системы, покупатель.

Редактор. Выполняет привязку товаров и каталогов обработанных магазинов к номинальным товарам. Может привязывать товары вручную, а так же редактировать привязку, созданную автоматически.

Администратор. Имеет права редактирования базы данных, добавления пользователей.

1.4 Ссылки

1.5 Обзор системы

Система делиться на три главных модуля.

Первый модуль отвечает за получение информации из магазинов, и её преобразования в необходимые для системы типы. Обработка каталогов и товаров магазинов происходит автоматизированном режиме, без участия пользователей системы.

Второй модуль анализирует данные базы, после чего создает список соответствия товаров и каталогов из обработанных магазинов к номинальным. На основании этого списка при большой вероятности сходства система автоматически создает привязку товара, иначе сохраняет список, после чего пользователь системы с правами редактора в ручную выбирает номинальный товар из списка наиболее схожих товаров, тем самым создавая привязку.

Третий модуль web приложение, которое позволяет пользователю просматривать цены на товары в различных магазинах на текущий момент или на выбранную дату.

2. ОБЩЕЕ ОПИСАНИЕ СИСТЕМЫ

2.1 Системный контекст

2.2 Режимы и состояния системы

Система может находиться в двух состояниях, которые соответствуют одному из состоянию в различных группах:

группа состояний 1:

Система может находиться в следующих состояниях:

- состояние системы в ожидании начала автоматического парсинга;

- состояние системы в автоматическом парсинге страниц интернет магазинов;

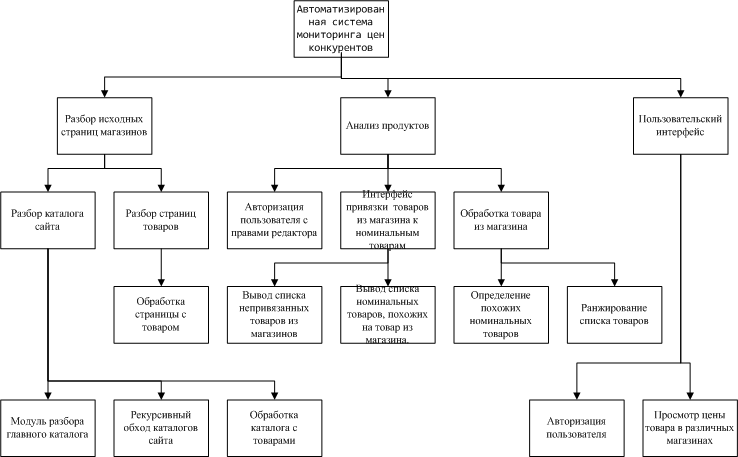
группа состоянии 2:

- состояние системы при авторизации пользователя;

- состояние системы при авторизации продавца;

- состояние системы при авторизации администратора;

2.3 Основные функциональные возможности системы



2.4 Основные условия системы

2.5 Основные ограничения системы

Начальная версия системы будет работать только на территории Удмуртской республики город Ижевск. Пользователю системы будет предоставлен вывод цены по дате, без выбора диапазона и графика изменения цены.

2.6 Характеристики пользователя

Пользователь. Конечный пользователь системы, заказчик. Количество пользователей этого типа не ограничено. Использует систему для отслеживания изменения цен товаров.

Редактор. использует систему для привязки обработанных товаров к номинальным. Пользователи этого типа создаются администратором.

Администратор. Имеет возможности назначать редакторов. Пользователь этого типа только один. Создается при создании системы.

2.7 Допущения и зависимости

2.8 Оперативные сценарии

Данная система подразумевает под собой следующий сценарий: автоматический мониторинг запускается через указанные промежутки времени и выполняет обновление данных по товарам и каталогам магазинов.

Для доступу к ресурсам сайта, пользователь должен пройти авторизацию, логин и пароль пользователю выдается после подключения к системе. Пользователь выбирает товар из списка товаров и дату для просмотра, после чего получает результат - цены на товар в различных магазинах.

Редактор системы производит авторизацию на ресурсе, после чего он выбирает магазин, каталог и продукт, а так же список номинальных товаров, похожих на этот продукт. Редактор может привязать продукт из магазина к номинальному, а так же редактировать связь продуктов, созданную автоматически.

Администратор системы создает пользователей других групп, а так же имеет доступ к исходной базе данных для редактирования.

ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ, УСЛОВИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ

3.1 Физические

3.1.1 Конструкция

3.1.2 Износостойкость

3.1.3 Адаптируемость

В случае увеличения нагрузки на сервер, количество арендованных серверов будет увеличено, а нагрузка равномерно распределена между ними.

3.1.4 Условия окружающей среды

Серверы не зависят от погодных условий.

3.2 Рабочие характеристики системы

3.3 Защита системы

Система должна быть защищена от sql - инъекций и xss - атак. При авторизации необходимо пройти проверку через систему reCapcha.

3.4 Информационный менеджмент

3.5 Работа системы

3.5.1 Эргономика системы

3.5.2 Эксплуатационная технологичность

3.5.3 Надежность системы

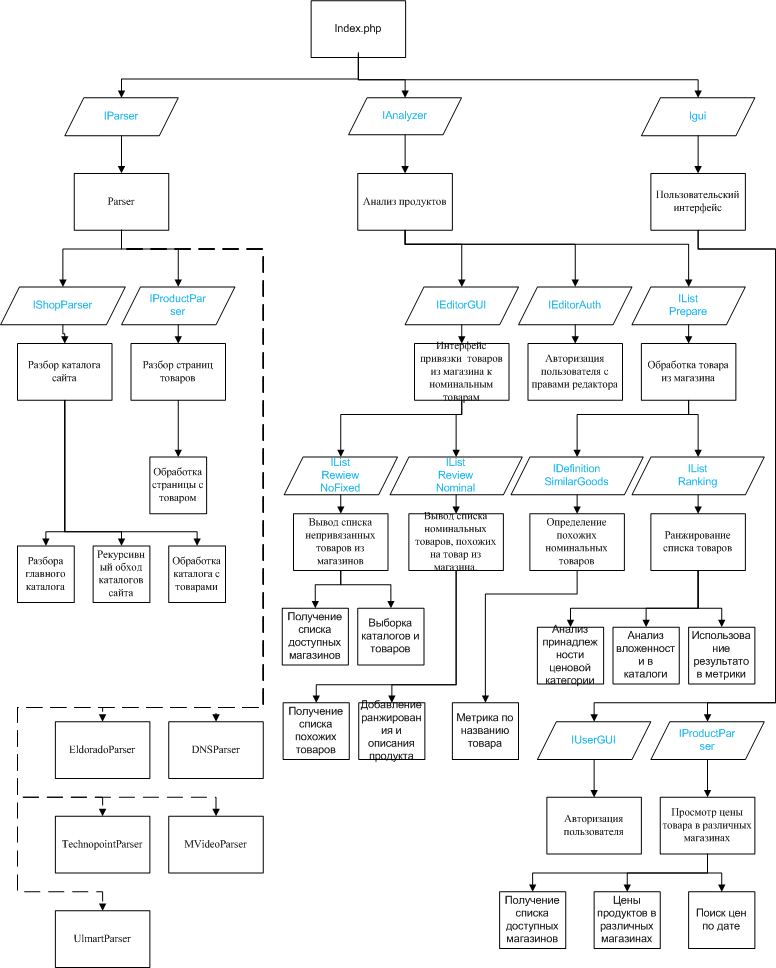
Система защищена от sql - инъекций и xss - атак. Предусмотрена защита со стороны получения страниц с интернет магазинов.

3.6 Стратегия и регулирование

3.7 Устойчивость жизненного цикла системы

Для устойчивости жизненного цикла системы используется обратная связь с пользователем.

4. СИСТЕМНЫЕ ИНТЕРФЕЙСЫ



Интерфейсы:

1. Интерфейс модуля парсинга html страниц

interface IParser

{

void StartParsing();

void UpdateNominal();

void InsertNominalCatalog(List<Catalog> catalogs);

void InsertNominalProducts(List<Product> products);

}

2. Парсинг страниц магазина, получение списка каталогов и товаров

interface IShopParser

{

void StartCatalogParsing();

void InsertOrUpdateCatalog(Catalog catalogs);

void InsertOrUpdateProduct(Product products);

string RequestInvoker(Uri url, Headers header, Cookei cookei);

}

3. Парсинг страниц магазина получение товара со страницы товара

interface IProductParser

{

void StartProductParsing();

void InsertOrUpdateProduct(Product products);

void ProductRequestInvoker(Uri url, Headers header, Cookei cookei);

}

4. Модуль анализа полученной информации

interface IAnalyzer

{

void AnalyzerInit();

void AnalyzerHtmlPreparation();

}

5. Интерфейс привязки товаров из магазина к номинальным товарам

interface IEditorGUI

{

void EditorGUIInit();

void EditorGUIHtmlPreparation();

}

6. Авторизация пользователя с правами редактора

interface IEditorAuth

{

void ShowAuthForm();

bool CheckAuth();

}

7. Обработка товара из магазина

interface IListPrepare

{

void StartProductProcessing();

}

8. Вывод списка непривязанных товаров из магазинов

interface IListRewiewNoFixed

{

List<Shop> GetAvailableShops();

List<Catalog> GetCatalogs(List<Shop> avalibleShop);

List<Product> GetProducts(List<Shop> avalibleShop);

}

9. Вывод списка номинальных товаров, похожих на товар из магазина.

interface IListReviewNominal

{

List<Product> GetSimilarProducts();

List<Product> RankingProducts(List<Product> products);

}

10. Определение похожих номинальных товаров

interface IDefinitionSimilarGoods

{

int StringComparison(string nominalitem, string shopitem);

void InitComparisonMethod();

}

11. Ранжирование списка товаров

interface IListRanking

{

int GetPriceProductSimilarPoints();

int GetCatalogProductSimilarPoints();

List<Product> GetSimilarWithMetrics(List<Product> products);

}

12. Авторизация пользователя

interface IUserGUI

{

void ShowAuthForm();

bool CheckAuth();

}

13. Просмотр цены товара в различных магазинах

interface IProductMonitoring

{

List<Shop> GetAvalibleShopList();

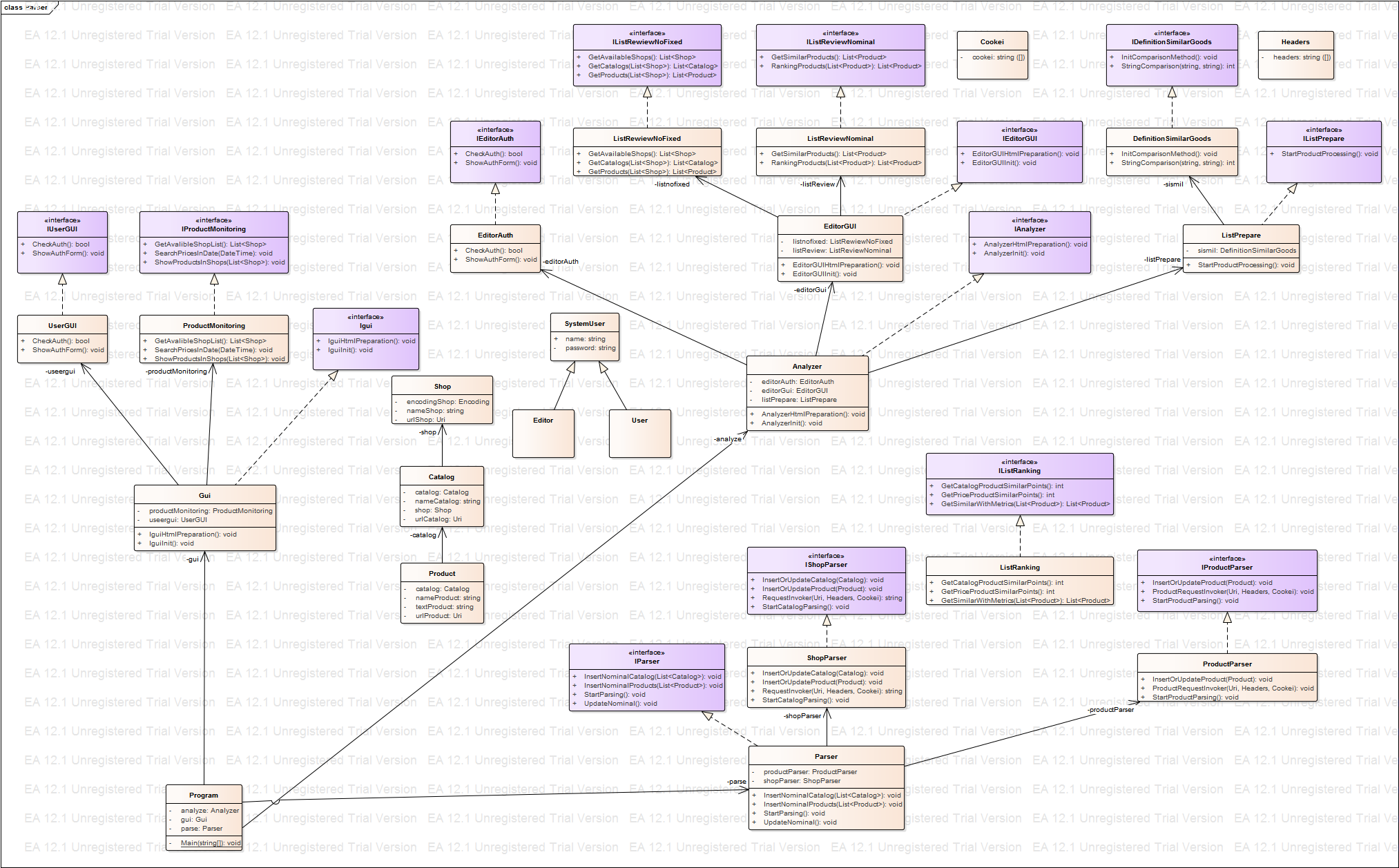
void ShowProductsInShops(List<Shop> shops);

void SearchPricesInDate(DateTime date);

}

5. КЛАССЫ КОДИРОВАНИЯ

5.1 UML схема программы



Классы кодирования

5.2 Классы кодирования программы соответствуют по функциональному назначению интерфейсам, по которым построены классы кодирования (см. п.4 Системные интерфейсы). Для системы так же разработаны следующие классы представления данных:

Класс каталог:

class Catalog

{

Uri urlCatalog;

Shop shop;

string nameCatalog;

Catalog catalog;

}

Класс продукт

class Product

{

Uri urlProduct;

string nameProduct;

string textProduct;

Catalog catalog;

}

Класс магазин:

class Shop

{

string nameShop;

Uri urlShop;

Encoding encodingShop;

}

Класс заголовки:

class Headers

{

string[] headers;

}

Класс куки

class Cookei

{

string[] cookei;

}