Титульник

Задание на курсовую работу

Система складского учета — профаммная система, затрагивающая все аспекты, связанные с движением товара на склад и со склада. По результатам анализа можно выделить семь основных функций системы.

Основные функции системы:

1. Учет заказов. Прием заказов от клиентов и ответы на запросы клиентов о состоянии заказов
2. Ведение счетов. Направление счетов клиентам и отслеживание платежей. Прием счетов от поставщиков и отслеживание платежей, направляемых поставщикам
3. Отгрузка со склада. Составление спецификаций на комплектацию товаров, отправляемых со склада клиентам
4. Складской учет. Постановка прибывающих товаров на учет и снятие товаров с учета при отправке заказов
5. Закупки. Заказ товаров поставщикам и отслеживание поставок
6. Получение. Принятие на склад товаров от поставщиков
7. Планирование. Выпуск отчетов, в том числе отражающих тенденции спроса на отдельные виды товаров и активность поставщиков.

В качестве части стратегии компании, занимающейся торговлей по каталогам, по проникновению на новые участки рынка было решено создать ряд относительно автономных региональных складов продукции. Каждый такой склад несет ответственность за учет товаров и выполнение заказов. В целях повышения эффективности своей работы склад обязан сам поддерживать ту номенклатуру товаров, которая в наилучшей степени соответствует потребностям местного рынка. Номенклатура может быть разной для каждого региона и должна оперативно меняться в соответствии с потребностями клиентов. Головная компания хотела бы иметь на всех складах одинаковую систему учета.

Основными функциями системы являются:

* учет товаров, приходящих от поставщиков, при приеме их на склад;
* учет заказов по мере поступления их из центральной удаленной организации; заказы также могут приниматься по почте. Их обработка ведется на местах;
* генерация указаний персоналу, в частности, об упаковке товаров;
* генерация счетов и отслеживание оплат;
* генерация запросов о поставке и отслеживание платежей поставщикам.

Кроме автоматизации стандартных складских операций система также должна предоставлять богатые возможности по генерации различных форм отчетности, в том числе отражающих тенденции развития рынка, списков наиболее надежных и ненадежных поставщиков и клиентов, материалов для рекламных кампаний.

Введение

1. ОО Анализ и проектирование

Этап анализа (analysis) состоит в исследовании системных требований и пробле¬ мы, а не в поисках путей ее решения. Например, при разработке новой информаци¬ онной системы для автоматизации торговли необходимо описать способы ее исполь¬ зования и основные функции.Анализ — это достаточно широкое понятие. Его содержание более точно отража¬ ют термины анализ требований (requirements analysis) (т.е. исследование требований к системе) и объектно-ориентированный анализ (object-oriented analysis) (исследование объектов предметной области).В процессе проектирования (design) основное внимание уделяется концептуальному решению (в виде программного обеспечения или аппаратных средств), обеспечиваю¬ щему выполнение основных требований, но не вопросам его реализации. Например, на этапе проектирования описываются программные объекты или схема базы дан¬ ных. Идеи проектирования зачастую исключают низкоуровневые или “очевидные” детали — очевидные с точки зрения потенциального пользователя. Конечно же, про¬ ектные решения могут быть реализованы в программном коде, и эта реализация должна быть точной и полной.Как и понятие анализа, этот термин тоже стоит уточнить и говорить об объектно- ориентированном проектировании (object-oriented design) или проектировании базы дан¬ ных (database design).Подытоживая вышесказанное, ООА/П можно определить как “выполнение пра¬ вильных действий (анализ) и обеспечение их правильности (проектирование)”.

В процессе объектно-ориентированного анализа основное внимание уделяется опре¬ делению и описанию объектов (или понятий) в терминах предметной области. Например, в случае информационной системы аэропорта среди понятий должны присутствовать Plane (самолет), Flight (рейс) и Pilot (пилот).В процессе объектно-ориентированного проектирования (или просто объектного про¬ ектирования) определяются программные объекты и способы их взаимодействия с целью выполнения системных требований. Например, в системе аэропорта про¬ граммный объект Plane может содержать атрибут tailNumber (бортовой номер) и метод getFlightHistory (получить историю полетов) (рис. 1.2).

“Унифицированный язык моделирования UML — это визуальный язык для определения, конструирования и документирования артефактов систем”

1. Итеративный эволюционный и гибкий процесс

Итеративная разработка — это основа подхода к созданию программных систем, положенного в основу описания объектно-ориентированного анализа и проектирования в данной книге. Такие адаптивные приемы, как быстрое моделирование, позволяют применять язык моделирования UML наиболее эффективно. Унифицированный процесс — это пример одного из самых популярных итеративных методов разработки на базе ООА/П. В данной главе рассматривается именно этот процесс, а также основы быстрого моделирования с использованием языка UML.

В основу унифицированного процесса и многих других современных методов положена важная идея — это итеративная разработка (iterative development). В рамках этого подхода разработка выполняется в виде нескольких краткосрочных мини-проектов фиксированной длительности (например, по три недели), называемых итерациями (iteration). Каждая итерация включает свои собственные этапы анализа требований, проектирования, реализации и завершается тестированием, интеграцией и созданием работающей части системы. Итеративный жизненный цикл основывается на постоянном расширении и дополнении системы в процессе нескольких итераций с периодической обратной связью и адаптацией добавляемых модулей к существующему ядру. Система постепенно разрастается шаг за шагом, поэтому такой подход иногда называют итеративной и инкрементальной разработкой (iterative and incremental development) (рис. 2.1). Поскольку обратная связь и адаптация приводят к развитию полученных ранее спецификаций и проектных решений, этот подход иногда называют также итеративной и эволюционной разработкой (iterative and evolutionary development).

1. Цели Задачи (общие)

1 Глава. Начальная фаза.

- Описание прецедентов

- Другие требования

2 Глава. Первая итерация фазы развития.

- Итерация 1 – основы

- Модели предметной области

- Системные диаграммы последовательностей

- Описание операций

- Итеративный переход от анализа требований к проектированию

- Логическая архитектура и диаграммы пакетов UML

- Переход к объектному проектированию

- Диаграммы взаимодействия на UML

- Диаграммы классов UML

- GRASP: проектирование объектов на основе распределения обязанностей

- Объектное проектирования на основе шаблонов GRASP

- Области видимости

- Преобразование проектного решения в программный код

3 Глава. Вторая итерация фазы развития.

- Средства построения диаграмм UML

- Быстрый дополнительный анализ

- Вторая итерация и дополнительные шаблоны

- Дополнительные шаблоны GRASP для распределения обязанностей

- Применение шаблонов проектирования GoF

4 Глава. Третья итерация фазы развития.

- Третья итерация — вспомогательные вопросы

- Диаграммы видов деятельности UML

- Диаграммы состояний UML и моделирование

- Взаимосвязь прецедентов

- Дополнительные диаграммы последовательностей и описание операций

- Уточнение модели предметной области

- Архитектурный анализ

- Уточнение логической архитектуры

- Новые проектные решения на основе шаблонов GoF

- Проектирование на основе пакетов

- Диаграммы развертывания и компонентов UML

- Проектирование каркаса взаимодействия с базой данных на основе шаблонов

- Документирование архитектуры с помощью N+1 представления

Заключение

Библиографический список