<http://programador.ru/simple-jsp-servlet-based-shop/>

[http://mamberry.ru](http://mamberry.ruИстория о том, как я делал интернет-магазин на Java. Linux,Tomcat,MySQL и Java (No EJB,No Spring,No Hibernate).)

[История о том, как я делал интернет-магазин на Java. Linux,Tomcat,MySQL и Java (No EJB,No Spring,No Hibernate).](http://mamberry.ruИстория о том, как я делал интернет-магазин на Java. Linux,Tomcat,MySQL и Java (No EJB,No Spring,No Hibernate).)

В качестве предисловия

Большую часть своей профессиональной карьеры я занимался разработкой бизнес-приложений на *Java*. Программные продукты предназначались, как правило, для различных финансовых организаций  или других сопоставимых с ними по размеру коммерческих структур. Разработка велась для Enterprise Level оборудования и серверов приложения. Стоит отметить, что очень много *java*-программистов работает именно в этом секторе. Возможно Вы представляете, что это такое -- дорогое оборудование, широкий канал, быстрая память, база данных Oracle  и другие атрибуты безбедной IT-жизни...  Но бывает так, что условия меняются с точностью до наоборот.

Что делать, если нужно создать интернет-приложение для маленькой фирмы на хостинге со скромными ресурсами?

Начало

Несколько лет назад моя сестренка открыла [магазин одежды](http://mamberry.ru/) в Нижнем Новгороде. Они с подружкой сами разрабатывали модели, закупали ткани и шили там же, в Нижнем Новгороде.

Я захотел ей как-то помочь. Мы решили, что было бы здорово создать сайт, на котором покупатели смогут просматривать каталог с моделями и размещать заказы на доставку, плюс будет возможность посмотреть схему проезда до магазинчика, информацию для оптовиков, объявления о распродажах и другие полезные мелочи.  По сути нужен был**простой интернет-магазин для маленькой фирмы.**

Конечно, не было речи о больших серверах, промышленных базах данных и других дорогих игрушках. Нужно было сделать что-то простое, шустрое и недорогое.

Поскольку я живу в Москве, а она в Нижнем, нужен был сайт, информацию на котором она могла бы менять без моего участия (или с моим минимальным).

Мне сразу стало видно два пути:  **PHP** или **Java**. Чтобы сделать выбор, решил провести небольшие исследования, так как не хотелось изобретать велосипед. Поскольку этот сегмент разработки ПО был для меня новым, я поставил денвер, посмотрел на PHP-ные решения, потом скачал разные *java-based* CMS-ки, поинтересовался у хостеров, что они могут мне посоветовать, спросил у друзей *PHP*-ников.

В итоге сделал для себя выводы:

**PHP**

плюсы:

* дешевый хостинг (в несколько раз дешевле по сравнению с java, если не брать в рассмотрение Google App Engine...)
* множество готовых решений для этого сегмента, как платных так и бесплатных.

минусы:

* я не знаю PHP и не хочу его изучать

**Java**

минусы:

* хостинг дороже (не учитываем Google App Engine, это отдельная песня)
* большинство решений тяжеловесны, предназначенные для больших и толстых серверов. Даже для Java Pet Store, все равно нужен нормальный Application Server.

плюсы:

* я знаю *Java* достаточно хорошо, все-таки опыт работы больше 8 лет, сертификаты SCJP, SCWCD, SCEApart1. Я **не**считаю себя **Über-программистом,**просто при работе именно с  *java* я чувствую себя комфортно и уверенно.
* желание сделать именно на *java*

В силу своей природной упертости, а возможно из нежелания покидать зону комфорта, я выбрал второй путь. Может это глупо и нерационально, но дико хотелось попробовать. Не нужны мне эти готовые монстрообразного размера CMS-ки, хотелось сделать что-то  совсем простое. Сделать решение на LAMP только  на *Tomcat*'е и *Java*'ой.

Задача, как показал дальнейший опыт, довольно простая, всё сделал сам.

**Выбор хостинга**

При всем богатстве выбора, как оказалось, не так уж много альтернатив.

**Вариант 1.**[**brim.ru**](http://brim.ru/) - самый известный хостинг в рунете, специализирующийся на java, очень даже неплохой.

Потребуется минимальный тариф от 2112 руб/год -  32(!) Мб оперативной памяти.

Регистрацию доменного имени можно выполнить самостоятельно, например через [nic.ru](http://nic.ru/) (+ 620 руб/год), либо через хостера -- тогда бесплатно. Если страшно потерять домен, можете запросить  потом у хостера данные по своей  учетной записи у регистратора и вывести.

Как поступать, зависит от обстоятельств.  В любом случае, нужно будет отправить сканы документа, удостоверяющего личность (т.е. паспорта), так что будьте к этому готовы.

Забугорный хостинг получится дороже, либо сильно урезанный по возможностям. Можно арендовать VPS и играться с ним сколько душе будет угодно, но это будет в разы дороже и совсем другая история.

**Вариант 2. Google App Engine  (GAE)** - Бесплатно! + регистрация доменного **GoDaddy.com (10$)**

Функциональные возможности такие же. В добавок получите различные плюшки от google, но почти везде имеются свои отличия, особенности.

Например:

* Хотите JSTL + EL?  Нужно добавлять <%@ page isELIgnored="false" %>
* Своя база данных
* Свое API для работы с картинками
* Свое API для аутентификации интегрированное с Google-службами

Если перечислять всё, получим целый букет различных фишек-граблей на которые придется наступать, если Вы до этого имели опыт написания веб-приложений под Java. С другой стороны,  Google радует программистов множеством интересного API (например работа с XMMP, т.е. чат). Думаю, в дальнейшем сделаю отдельный пост о  Google App Engine, в котором расскажу о своем реальном опыте работы с этой платформой.

Что касается меня, то у меня выбора не было, поскольку Google App Engine для Java выпустили только летом 2009 г., а я к этому времени уже всё сделал.

Сейчас сайт работает уже больше двух лет на бримовском хостинге.  Я их сервисом вполне **доволен**.

Идем дальше...

**Архитектура**

Архитектура простая, как молоток. Есть три компонента, назовем их условно:  *Фронт*, *Админка* и *Хранилище*.

*Хранилище* нужно для того, чтобы получать, сохранять и удалять продукцию по каталогам. Другими словами это jar-ик в котором находится набор DAO (Data Access Object) для работы с БД и набор POJO объектов представляющих сущности (продукт, каталог и т.д.) .

*Фронт* - сам сайт. Тут все просто. Для представления используем JSP в связке с JSTL и EL. В паре мест, где удобней Model 2, используем обычные сервлеты с форвардом на JSP-ку. Struts не нужен, а тем более JSF.  Позади JSP-ок с сервлетами находятся классики с бизнес логикой, которые лезут в *Хранилище.*

*Админка* позволяет добавлять,удалять и редактировать информацию о продуктах и каталогах. Админка закрыта от публичного доступа. Предназначена для обычного пользователя. Так же, как и фронт использует классы из *Хранилище*.

Да, конечно, можно было своими шаловливыми ручонками лезть прямо в базу данных из JSP-ек или использовать для этих целей JSTL SQL тэги. К сожалению, такой подход вызывает у меня рвотный рефлекс и я не смог себя пересилить. Пришлось соблюсти минимальные рамки приличия.

**Хранение данных**

Сразу стал вопрос ORM или не ORM?

Конечно можно использовать:

**Нibernate**

плюсы**:**

* у меня есть опыт работы с этой платформой (знаю, какие есть подводные камни и т.д.)
* отлично документированный фреймворк

минусы:

* будет жрать ресурсы, а у меня всего 32 Mb, нет уж...

Вообще хотел сразу сказать относительно Hibernate -- он конечно упрощает жизнь программиста, но многие не соображают, что делают, когда его используют. Например, вот мои любимые грабли, на которые многие постоянно наступают - [популярные грабли](http://programador.ru/hibernate-update/).

**DataNucleus**

Минусы:

* Менее популярный по сравнению с Hibernate, как следствие -- информации меньше.

Плюсы:

* Пошустрее будет, но все равно у меня ведь 32(!) Mb.
* Используется в Google App Engine. Если захочется потом соскочить на GAE хостинг, то это будет полегче, чем спрыгивать вместе с Hibernate .

**iBatis** -- можно, но зачем?

В общем, чтобы закрыть этот вопрос, мое мнение:  ORM-ы удобны, когда  у вас больше десятка разных классов, объекты которых нужно хранить в базе данных  и с каждой рабочей неделей начальство или заказчик подкидывает Вам как минимум по парочке новых, меняя при этом старые..  Плюс Вы не экономите на hardware, а это не мой случай. И если это не так, то ORM не нужен. Если хотите получить действительно быстрое и робастное приложение, то лучше использовать JDBC(!), и не верьте если Вам говорят, что JDBC -- это архаизм.

Лично меня не особо утомляет написать несколько select, insert, update,delete, раньше всё так делал, и ничего, не поплохело.

Вот небольшая памятка, что не нужно забывать (для тех кто не знает):

1. Использовать preparedstatement
2. Закрывать connection, statement и result set
3. Использовать connection pool (dpcp или c3p0)
4. Не забыть про индексы

Представление

Обычный JSP + JSTL + EL. Вот всё, что нужно на самом деле. JavaServerFaces - не нужен.

Сейчас 21 век на дворе, так что если хотите, чтобы было красиво, то используйте jQuery или другие javascript-овые библиотеки.

Дизайн

Если Вы не имеете художественных навыков, лучше не пытаться делать его самостоятельно. Дизайн стоит брать профессиональный, т.е. придется **покупать**. Я купил на стоке, рублей за 500. Ну не Артемий Лебедев, согласен, но за 500 рублей пойдет.  Всё относительно ведь, например, перекусить в "Граблях" или "Му-Му" будет стоить приблизительно таких же денег.

Лучше потратиться на дизайн, чем сделать самому, а в итоге будет какашка (извиняюсь за выражение). Даже если друзья и родственники будут говорить "ну вроде ничего так, для первого раза пойдет", они скорее всего не хотят Вас обидеть...

Если все-таки решили делать сами, воспользуйтесь хотя бы готовыми палитрами цветов, их в интернете полно.

**Всё дело в деталях**

Детали, в которых бывает зарыта собака:

* Структура и сборка проекта :  [maven](http://maven.apache.org/) (это скорее профессиональная привычка)
* Редактор текстовых полей в админке:   [jwysiwick](http://code.google.com/p/jwysiwyg/)
* Оформление заказа. Нотификация по эл. почте. Обычный [JavaMail](http://java.sun.com/products/javamail/)
* Ведение логов. **Log4j**через **Slf4j** (тоже профессиональная привычка)
* Информация о расположении магазина на карте. Использую **Яндекс.Карты**. Нужно зарегистрировать и получить ключи в Яндексе.
* Счетчик посещений. Использую **Яндекс.Метрика**. Делаете JSP-ку, которую инклудите в свой хедер или футер (ну или как душа пожелает).
* Реклама. Так как сервис предназначен для русскоязычной аудитории, пробовал рекламу от **Яндекс.Директа**. Где-то по 1000 руб на кампанию.  
  Первая рекламная кампания прошла  неплохо. Сестра звонила, сообщала, что в магазин стали постоянно приходить покупатели, которые нашли магазин через интернет.  
  Вторую кампанию провел летом, на фоне сильного сезонного спада, так что количество покупателей было не много.  
  Думаю следующую кампанию провести ближе к осени.
* Почта.  Как правило, каждая компания хочет, чтобы у ее сотрудников был электронный адреспохожий на*ivan@company.ru ,* а не *vanya\_1977@yandex.ru.*Как вариант, стоит попробовать сделать интеграцию с **Google Apps**. Это бесплатно. Получаете нормальный доступ к своей фирменной почте и с веб-интерфейсом похожим на GMail. Очень удобно. Нужно зарегистрироваться на Google Apps. Затем нужно написать письмо с просьбой админу в brim.ru, он подправляет нужные записи в DNS-серваке.

Итак, сайт (интернет-магазин одежды для беременных) : [mamberry.ru](http://mamberry.ru/)

**Итог**

Сайт работает больше двух лет. Время от времени, по мере появления свободного времени добавляю всякие мелочи.  
На SEO и раскрутку деньги не тратим, все-таки это маленький магазинчик, масштаб не тот, да и платить деньги оптимизаторам не охота из принципа.  Думаю, пройдет немного времени и свой покупатель нас найдет.  Например, недавно неожиданно для себя обнаружил, что по запросу в Яндексе "одежда для беременных Нижний Новгород" сайт вылез в первые строчки.

В сумме:

**Минусы**

* за первый год потрачено: хостинг 2112 руб/год + дизайн 500 руб + яндекс.директ 1000 руб =3612 руб.
* Потраченное время: в сумме  выходные  и вечера, потраченные в течение года, дадут где-то месяц чистой работы.

**Плюсы**

* Приобретенный опыт - бесценен.
* Счастливая сестра, для которой брат сделал интернет-магазин.

P.S. Кому нужен совет или кто столкнулся с похожей задачей -- пишите, может что подскажу

■

**Опубликовано:** 19 июля 2010

**Написано в рубрике:** [3. Инструментарий](http://programador.ru/category/toolkit/),[4. Полезняшки](http://programador.ru/category/hints/),[Java](http://programador.ru/category/lang/java/).

**Теги:**[Java](http://programador.ru/tag/java/), [программирование](http://programador.ru/tag/%d0%bf%d1%80%d0%be%d0%b3%d1%80%d0%b0%d0%bc%d0%bc%d0%b8%d1%80%d0%be%d0%b2%d0%b0%d0%bd%d0%b8%d0%b5/)