**Construirea unui site web cu vanzari on-line**

## Prezentarea proiectului:

Proiectul consta intr-o aplicatie web prin care utilizatorul poate accesa un catalog de produse ale unui magazin virtual si care poate prelua comenzi de la acestia prin intermediul internetului.

## Detalii tehnice:

Mediul de dezvoltare: vim (editor text)

Limbaje folosite: Perl, Bash

Serverul web: Apache (2.2) cu mod\_perl, mod\_svn

Baza de date: MySQL (5.0)

SVN (depozit de fisiere de orice tip cu capacitati de versionare)

## Modul de concepere si construire a aplicatiei si a interfetei:

**-aplicatia**

**-interfata**

Partea de sus: web art, panou autentificare

Partea stanga: panou de navigatie prin categoriile principale de produse

Partea dreapta: cautare rapida + detalii despre produsele selectate sau cautate

## Resurse software necesare:

Sistemul de operare Linux (Ubuntu Server 6.06 LTS)

Serverul web Apache 2.2 cu mod\_perl, mod\_auth, mod\_rewrite, mod\_svn

Serverul de baze de date MySQL 5.0

SVN modificat

## Resurse hardware necesare:

Serverul are urmatoarele componente relevante: Procesor P4 Intel 3Ghz, Memorie 4GB RAM, 3x250GB 7200 rpm HDD in modul software RAID. (mirroring & striping).

**Estimari asupra performantelor:**

Efectiv, site-ul ocupa 2GB pe HDD cu tot cu continut, iar load average este ~ 10% CPU, 2GB de memorie fiind ocupati.

In full load, ~50%-70% CPU folosit cu 4GB RAM. (majoritatea ocupata de cache). Sistemul poate face fata deoarece mai exista si SWAP files care pot prelua activitatile de memorie in cazul in care cei 4GB de memorie fizica sunt ocupati. Activitatea cea mai intensiva este introducerea de noi produse si poze (necesita autentificare, upload, introducere date si salvare). Numarul de cereri concurente nu este anticipat a fi peste 100 (limitare de latime de banda). Latime de banda alocata: 10Mbit garantat.

## Perspective de dezvoltare ulterioara a aplicatiei:

Aplicatia a fost creata in Perl si Bash si documentata, astfel dezvoltarea ulterioara a acestui produs este usor de realizat. Datorita sistemului SVN sub care a fost contruita aplicatia si stocarea datelor, dezvoltarea se poate face direct pe fisierele respective, apoi commit catre repository.

Testarea s-a realizat cu ajutorul unor unelte specifice (Mozilla addon: Web Developer si validatoarele de la www.w3.org) pentru a verifica integritatea paginii si a previzualiza anumite efecte in timpul dezvoltarii.

# Diagrama:

/Nova  
|--nova   
|--svn-update  
|--svn-commit  
|--backups (Backup intern)  
| |-- backup scriptul de backup  
| \-- arhive  
|-- bin  
| |-- eke Cookies ( Equal Keys Expiration times )  
| \-- eke.pl  
| |-- eke.start  
| \-- magick Codificatorul grafic (CAPTCHA)  
| |-- magic   
| \-- tmp  
|-- DOCUMENTATIE  
| |-- FAQ ( Ce sa ? Cum sa fac ? )  
| |-- Harta Aceasta schema  
| |-- Utilizare si limitari Manual de utilizare  
|-- LEGAL  
|-- man DOCUMENTATIE TEHNICA  
| |-- log sistemul de logging (script shell)  
| |-- manual descrie cateva functii principale si apelarea manuala  
| |-- perms permisiuni si proprietari (script shell)  
| \-- svn documentatie Subversion  
|-- apache Scripturi de pornire/oprire si configurarea serverului apache2  
| |-- apache2-ctl script de control apache2  
| |-- apache2-nova }   
| |-- apache2-admin }   
| |-- nova Configurarea site-ului extern  
| | |-- modules.conf  
| | |-- modules.debian.conf  
| | |-- modules link catre modules.conf sau modules.debian.conf ( in funtie de modul in care a fost instalat apache2 )  
| | |-- nova.conf configurari globale  
| | |-- nova.host configurari VirtualHost  
| | \-- ports.conf porturi de Listen  
| \-- admin Configurarea site-ului intern  
|-- cron   
| |-- clean curata cheile si imaginile invechite generate de procesul de inregistrare  
| |-- clean.cron procedura de curatare in format crontab  
| |-- clean.rc instaleaza continutul fisierului `pustiu.cron` in /etc/crontab  
| \-- logrotate goleste logurile si salveaza o copie de backup

|-- statistics  
|-- temp\_store  
|-- core

| \--db sql users  
|-- images  
|-- www