

# Veeb

Jaagup Kippar

# Tööpõhimõte

- Arvutid ühendatud võrguks
- Igal arvutil oma nimi / number
- Arvutite vahel suudavad liikuda tekstid/baidid
- Kasutaja saadab serverarvutile päringu, millist lehte soovitakse vaadata.
- Kui leht olemas ja vaatamine lubatud, siis saadetakse vastu soovitud leht, muul juhul veateade.

# Kasutus

- Tehniliselt poolt suhteliselt lihtne protokoll.
- Kõik oleneb sellest, mida näidata ja kuidas näidata.
- 1995nda aasta algul näidati Küberneetika Instituudis oleval konverentsil imelehte, kus tekst tuli kõrvalruumist, pilt sinna sisse Austraaliast.
- Tänapäeva teate isegi :-)

# Tehnoloogiad

- Lehestike loomiseks ja mugavamaks vaatamiseks on leiutatud hulk tehnoloogiaid.
- "Traditsiooniliseks" võib ehk lugeda serveris asuvaid staatilisi lehti ning kliendi pool neid püüdnult näitavat veebilehitsejat.
- Nii serveri kui ka kliendi poole on aegade jooksul hulk abivahendeid loodud.

# Serveripoolsed tehnoloogiad

- CGI - Common Gateway Interface võimaldab juba veebi algusest peale igasugustel käivitatavatel programmidel veebikliendilt andmeid saada ning väljund sinna tagasi kuvada.
- Levinud keeleks oli PERL, kuid võib kasutada omale sobivat keelt/rakendust. Näiteks Matlabi nimeline rakendus lubab nõnda kasutaja soovitud graafikuid koostada ja esitada.

# Serveripoolsed tehnoloogiad

- Võimaldavad "tavaliste" programmeerimis-keeltega võrreldes tekstilist väljundit HTML-kujunduse jaoks mugavamalt luua.
- Aitavad eraldada kujundust, andmete töötlemist ning hoidmist
- Võimaldavad väljundit puhverdada, et iga lehe andmeid ei peaks alati uuesti arvutama.
- Aitavad kasutajaid tuvastada ning nendele õigusi jagada.

# Serveripoolised tehnoloogiad

- PHP
- ASP.NET
- Java Servlet / JSP
- Python + Zope
- Ruby on Rails
- Oracle JDeveloper
- ...

# Serveripoolsed valmiskeskonnad

- Lasevad lihtsalt ja mugavalt teha seda, milleks nad mõeldud on.
- Võivad üllatavalt paindumatud olla soovivate muudatuste suhtes.
- Wordpress, Joomla, Drupal, phpBB, Edicy, Plone, SharePoint ...



# Kliendipoolsed tehnoloogiad

- Võimaldavad kiiresti reageerida kasutaja tegevusele masina taga
- Suudavad reaajas ning kasutajale nähtamatult vahetada serveriga andmeid ning selle põhjal muuta lehe sisu.
- Suudavad ekraanile manada juhitava animatsiooni

# Kliendipoolsed tehnoloogiad

- JavaScript - veebilehel "kohe olemas". Viimastel aastatel taas tõsiselt au sisse tõusnud.
- Java rakendid (appletid). Võimaldavad veebilehitsejas kasutada pea kogu Java programmeerimiskeele võimalusi (~100000 käsklust)
- Flashi/Flexi lehed. Mugav kujundamiseks ja andmete esitamiseks. Silverlight
- Mitmesugused eraldi lisatavad pluginad

# Laiendusraamistikud

- Toetuvad olemasolevatele keeltele.
- Võimaldavad levinumaid toimetusi mõnikord mugavamalt ette võtta.
- Serveri pool: Smarty lehemallid, Code Igniter, ...
- Kliendi pool: Prototype, JQuery, Yahoo User Interface ...

# Vidinaid

- Lehele lisatavad ning kusagil mujal töötavad lahendused
- YouTube videoklipid
- Google kaardid ja muud teenused (tõlkimine, diagrammid, Gadgetid)
- Keskkondade pakutavad tükid (nt. Facebooki kommuuni andmete näitamine)

# Reaalajas muutuvad lehed

- Pidev ühendus - nt. jututoad, rallimängud. Eelistavad eraldi kanalit, kustkaudu serverist tulevaid andmeid püüda.
- Perioodiline laadimine taustal - AJAX - Active JavaScript and XML. Näiteks uudiste ja ilmamateate uuendamine ilma kogu lehte uuesti lugemata.
- WebSocket-ühendus

# Ohud

- Terviklus - kas näen ja kasutan seda, millega tegelikult tegemist on.
- Käideldavus - kas saan vajalikud asjad õigel ajal kätte ning suudan neid talletada
- Konfidentsiaalsus - kas andmeid näevad ning rakendusi saavad kasutada vaid need, kes selleks volitatud on.
- 100% turvaline lahendus kipub olema ka 100% kasutu - tuleb leida sobiv tasakaal.

# Serveripoolsed ohud

- SQL Injection - andmebaasikäskude saatmine veebilehelt asjaliku sisu asemel. Tehnoloogia või programmeerija apsakate tõttu võib juhtuda, et need lähevad ootamatul juhul käima. Sama võib juhtuda failisüsteemi käskudega.
- Ohjeldamatu koguse segavate andmete saatmine, samuti ohjeldamatu päringute saatmine
- Alistatud serveri kasutamine edasi uuteks rünnakuteks.

# Kliendipoolsed ohud

- Veebilehitseja ei peaks laskma veebist saabuvatel andmetel mõjutada masina sisu ega muid veebisaite, kuid see "liivakast" kipub ikka ja jälle rebenema.
- Pildikodeeringutesse peidetud andmed, programmeerimiskeelte käsklused, mida pole suudetud ohjata.



# Serveripoolsed turvalahendused

- Saabuvate andmete kontroll ning vaid lubatud sümbolite/järgnevuste läbilaskmine.
- PHP Safe mode - pääseb ligi ainult oma andmebaasile. Ei saa luua väliseid võrguühendusi, ei saa saata kirju.
- Veebitehnoloogiate juures olev "liivakast" (sandbox). Saab teha vaid piiratud toiminguid.
- Serveripargi jaotamine ning tulemüüridega kaitsmine.

# Kliendipoolsed turvalahendused

- Lynx :-) (tekstipõhine veebilehitseja)
- Ohutuim arvuti on kinnine arvuti
- Lehitseja ja kasutaja võimalikult madalad õigused.
- Väljalülitatud Javaskript ja muud programmid - mõned keskkonnad siis jälle ei tööta.
- Püüa eelnevalt hinnata veebilehtede usaldusväarsust.
- :-(

# Veebilehe ülespanek

- Serveri nimi
- Kasutajanimi
- Parool
- Asukoht serveris
- Asukoht veebis
- Kopeeriv programm (SCP / FTP)
- Veebipõhine haldus (CPanel)

# Veebilehtede majutus

- Oma server - kõigi oma murede ja rõõmudega
- Tasuta ülesriputamiskohad - staatilise lihtsa tutvustava lehe jaoks piisav - hot.ee
- Renditud veebiring - 5-20 eurot kuus.
- Tasub vaadata, mis parameetrid parajasti tarvilikud - kettamaht, üleslaadimine, alamdomeenide arv, andmebaasimaht, tarkvara ja versioonid.

# Veebilehtede koostamine

- Veebis loetav võib olla pea iga tavaline dokument
- Save as HTML / PDF
- Keskkonnaga kaasas olev redaktor
- Dreamviewer, Visual Web Developer Express, Artisteer, ...
- Lihtne tekstiredaktor
- Vistutamine

# Lehestiku struktuur

- Kuni viie lehekülje puhul võivad kõik lehed igalt poolt otse kättesaadavad olla.
- Suurema lehestiku puhul kasulik struktuuripuu.
- Valik klõpsude arvu ja lehe sügavuse vahel.
- 5 - 25 - 125 - 625
- 50 - 2 500 - 125 000 - 6 250 000
- Puu loomise raskused - ebasümmeetrilisus, kuulumine mitmesse valdkonda.
- Võrk ja otsing.

# Veebiotsing

- Otsingud jõuavad vaid lehtedeni, millele kusagilt viidatakse, või mis eraldi registreeritakse.
- Oma ja tuttavate tarbeks vajalikud lehed ei peagi avalikult leitavad olema - iseasi kas neid on samas mõtet parooliga kaitsta.
- Tasulise reklaami puhul tasub kalkuleerida, kas see ikka kasulik on.

# SEO - leitavus otsimootorites

- Lehe sisesed tegevused - asjalikud pealkirjad, märksõnad, standardile vastav kirjapanek.
- Lehe välised tegevused - viited autoriteetsetelt lehtedelt, registreerimine otsimootorite ja kataloogide juures.
- Otsimootorite mitmesugused, kuid salajased ja muutuvad algoritmid.



# Kodutöö

Olemasoleva veebisaidi analüüs.

- Sihtgrupid, struktuur.
- Kasutatavad serveri- ja kliendipoolsed tehnoloogiad.
- Ühendused teiste lehestikega, vistutatud komponendid
- Turvaanalüüs - võimalikud murekohad
- SEO - leitavus ja sobivus otsimootorites.

# Essee

- Ideaalne E-riik.

(Milliseid veebirakendusi luua ja kuidas täiendada, et riigi andmestikega suhtlemine toimuks võimalikult mugaval viisil)