

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/267822370>

# QuickASP: PEMBANGIUT KODE PROGRAM ASP UNTUK APLIKASI BASIS DATA BERBASIS WEB

Article in *JUTI Jurnal Ilmiah Teknologi Informasi* - January 2007

DOI: 10.12962/j24068535.v6i1.a183

---

CITATIONS

0

---

READS

321

3 authors, including:



[Siti Rochimah](#)

Institut Teknologi Sepuluh Nopember

95 PUBLICATIONS 242 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Code Smell [View project](#)



Development of Data Modeling Tool [View project](#)

# QuickASP: PEMBANGKIT KODE PROGRAM ASP UNTUK APLIKASI BASIS DATA BERBASIS WEB

Imam Kuswardayan, Rahmi Ika Noviana, Siti Rochimah

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember  
Kampus ITS, Sukolilo Surabaya 60111

Email : [imam@its.ac.id](mailto:imam@its.ac.id), [iik@inf.its-sby.edu](mailto:iik@inf.its-sby.edu), [siti@its-sby.edu](mailto:siti@its-sby.edu)

## ABSTRAK

Dalam pembuatan sistem aplikasi basis data berbasis web, perancangan antarmuka pengguna (*presentation layer*) dan lapisan bisnis (*business layer*) merupakan tahap yang dilalui setelah pemahaman terhadap kebutuhan pengguna sistem. Adanya pola atau keteraturan dalam implementasi tahap ini menyebabkan pengembangan sistem akan lebih efisien jika menggunakan suatu aplikasi yang dapat menghasilkan kerangka dasar aplikasi web dengan cepat untuk kedua lapisan tersebut dan bahkan beserta kode programnya.

Pada penelitian ini telah diimplementasikan suatu perangkat lunak yang selanjutnya disebut *QuickASP*. *QuickASP* membangkitkan kode ASP untuk membangun homepage otomatis. Untuk membangkitkan kode ASP, *QuickASP* membutuhkan komponen berupa basis data dan file *Cascading Style Sheets (CSS)*. Proses awal yang dilakukan *QuickASP* dalam membangkitkan kode program ASP adalah membaca informasi basis data berupa tabel-tabel, nama field dan tipe data. Setelah itu *QuickASP* akan membangkitkan file-file ASP beserta file-file pendukungnya berdasarkan hasil pengaturan tampilan halaman web yang dilakukan oleh pengguna.

Uji coba *QuickASP* dilakukan pada tiga jenis basis data yaitu *Microsoft Access*, *Microsoft SQL Server*, dan *Oracle*. Dari hasil uji coba tersebut, *QuickASP* terbukti dapat membangkitkan homepage otomatis beserta fungsi-fungsi yang disediakan untuk modifikasi record dan fungsi navigasi.

**Kata kunci:** *QuickASP*, *file cascading style sheets*, *kode program ASP*.

## 1. PENDAHULUAN

Saat ini web telah berkembang sebagai salah satu aspek penunjang dalam dunia bisnis yang diwujudkan dalam bentuk *e-commerce*, *e-marketing*, *on-line transaction*, dan lain-lain. Salah satu bahasa pemrograman yang dapat dipakai untuk membuat sebuah web yang dinamis adalah ASP (*Active Server Pages*). Pada umumnya seorang pengembang web harus memahami sintaks ASP dan HTML serta bahasa SQL jika hendak mengembangkan sebuah situs web dengan bahasa pemrograman ASP. Padahal tidak semua orang mempunyai cukup waktu untuk mempelajari lebih dalam tentang pemrograman web (ASP) dan pemrograman basis data (SQL) di depan komputer.

Untuk itu, penelitian ini bertujuan membuat sebuah perangkat lunak agar pembuatan web dinamis berbasis ASP menjadi lebih mudah dan cepat. Perangkat lunak yang dibuat secara otomatis akan menghasilkan kode program ASP kemudian mengirim data ke URL yang diinginkan.

Permasalahan yang diangkat dalam penelitian ini antara lain bagaimana agar perangkat lunak yang dibuat mampu membaca struktur basis data, menyediakan fasilitas yang memungkinkan pengguna untuk melakukan modifikasi desain antar muka pada situs web hasil pembangkitan, serta membangkitkan kode program ASP untuk pembuatan web yang dinamis dengan berbagai pilihan DBMS.

Basis data sumber yang dapat dibaca oleh perangkat lunak ini dibatasi pada basis data *Microsoft Access*, *SQL Server*, dan *Oracle*. Sedangkan tipe data yang dikenali oleh perangkat lunak terbatas pada tipe data yang berupa teks, numerik, *boolean*, dan gambar (.gif, .jpg, .bmp).

## 2. ACTIVE DATA OBJECT (ADO)

ADO memiliki kemampuan untuk mengakses beragam sumber data, termasuk di dalamnya adalah basis data dalam format *Microsoft Access* dan *Microsoft SQL Server*, maupun basis data yang bukan dari *Microsoft* seperti basis data *Oracle* dan *Sybase*. Dan yang lebih penting lagi yaitu bahwa ADO pada akhirnya akan menggantikan DAO (*Data Access Objects*) dan RDO (*Remote Data Objects*), yang merupakan teknologi pengaksesan data cara lama. *Microsoft* meyakinkan bahwa ADO akan terus dikembangkan dan mendukung kompatibilitas dengan sistem operasi dan basis data masa depan.

ADO memiliki model objek yang dapat dilihat pada Gambar 1.

## 3. PROPERTI BASIS DATA SQL SERVER

Pada *SQL Server*, informasi mengenai struktur basis data *SQL Server* seperti nama basis data, nama-nama tabel, struktur tabel, nama-nama view, nama-nama kolom untuk tiap tabel, tipe data masing-masing kolom, dan informasi lainnya, tersimpan

dalam sebuah view yang bernama INFORMATION\_SCHEMA.

Berikut beberapa sintaks SQL yang digunakan untuk menampilkan beberapa properti basis data.

1. Untuk menampilkan daftar nama-nama basis data dimana pengguna tertentu memiliki hak akses tertentu  

```
select * from information_
schema.schemata
```
2. Untuk menampilkan daftar nama-nama tabel  

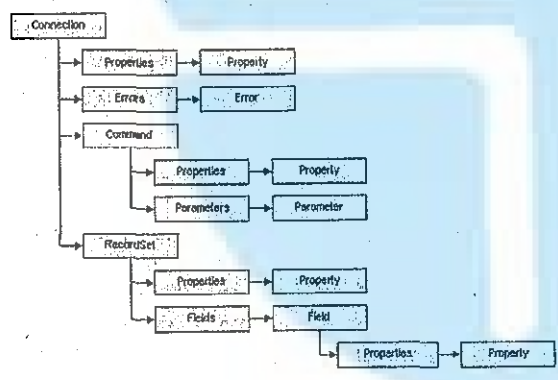
```
select * from information_
schema.tables
where table_type = 'base table'
```
3. Untuk menampilkan daftar nama kolom yang merupakan *primary key*  

```
select * from Information
Schema.Key_Column_Usage where
TABLE_NAME=' ' & MyTable & ' ' and
constraint_name like '%PK%'
```

Selain dengan cara mengakses view INFORMATION\_SCHEMA, SQL Server juga menyediakan fungsi COLUMNPROPERTY yang dapat digunakan untuk mendapatkan properti basis data yang lain.

Berikut sintaks yang digunakan untuk mengidentifikasi apakah suatu kolom *identity* atau bukan:

```
SELECT COLUMNPROPERTY(OBJECT_ID
('TableName'),'ColumnName',
'IsIdentity') as IdentityValue
```



Gambar 1 Model Objek ADO

#### 4. PROPERTI BASIS DATA MS-ACCESS

Informasi mengenai struktur basis data pada Microsoft Access dapat diambil dengan menggunakan fungsi OpenSchema() pada Microsoft Visual Basic 6. Berikut beberapa parameter dari fungsi OpenSchema().

- a. adSchemaTables  
Menampilkan daftar tabel pada basis data
- b. adSchemaColumns  
Menampilkan daftar kolom suatu tabel
- c. adSchemaPrimaryKeys  
Menampilkan *primary key* suatu tabel.
- d. adSchemaForeignKeys  
Menampilkan *foreign key* suatu tabel.

#### 5. PROPERTI BASIS DATA ORACLE

Informasi struktur basis data Oracle tersimpan dalam beberapa tabel. Berikut beberapa sintaks SQL yang digunakan untuk menampilkan beberapa properti basis data Oracle.

- a. Untuk menampilkan daftar nama-nama tabel  

```
select table_name from user_tables
```
- b. Untuk menampilkan daftar nama-nama view  

```
select view_name from user_views
```
- c. Untuk menampilkan daftar nama-nama *field* dari sebuah tabel  

```
select column_name from user_tab_
columns where table_name =
'table_name'
```
- d. Untuk menampilkan daftar properti dari sebuah *field*  

```
select data_default, data_type,
data_length, nullable from
user_tab_columns where table_name =
'table_name' and column_name =
'column_name'
```
- e. Untuk menampilkan daftar *field* yang merupakan *primary key* dari sebuah tabel  

```
select cols.table_name,
cols.column_name, cols.position,
cons.status, cons.owner
from all_constraints cons,
all_cons_columns cols
where cols.table_name = 'table_name'
and cons.constraint_type = 'p' and
cons.constraint_name
cols.constraint_name
and cons.owner = cols.owner
order by cols.table_name,
cols.position
```
- f. Untuk menampilkan daftar *field* yang merupakan *foreign key* dari sebuah tabel  

```
select a.constraint_name,
a.constraint_type,
b.table_name mytable,
b.column_name mycolumn
from all_constraints a,
all_cons_columns b
where a.owner='owner_name'
and a.constraint_type='r'
and b.constraint_name =
a.r_constraint_name
```

#### 6. ACTIVE SERVER PAGES

Pada dasarnya bahasa-bahasa yang dapat digunakan untuk membangun suatu web dapat dibedakan menjadi dua, yaitu *Client-side* dan *Server-side*. *Client-side* memiliki arti bahwa informasi yang disampaikan akan dieksekusi di *client* atau *browser*. Adapun lawannya adalah *Server-side*, yaitu proses pengerjaan informasi akan dikirim dan diproses di server web tersebut.

Active Sever Pages (ASP) adalah suatu skrip yang bersifat *server-side* yang memiliki kemampuan untuk dikombinasikan dengan teks, HTML, dan komponen-komponen lain untuk membuat halaman

web yang lebih menarik, dinamis, dan interaktif. Komponen adalah objek yang sudah terkompilasi dengan *native code* di masing-masing platform, baik itu platform Windows maupun platform lainnya. Dalam ASP dapat digunakan sintaks pemrograman VBCode program ataupun Jcode program sehingga *web programmer* dapat secara bebas memilih salah satu bahasa pemrograman yang dikuasai atau disenanginya.

Dalam pembuatan aplikasi yang interaktif antara pemilik web dan pengunjung dengan menggunakan ASP diperlukan dukungan bahasa lain. Sebagai contoh, apabila ingin membuat aplikasi yang melibatkan penggunaan basis data maka perlu juga digunakan bahasa SQL.

ASP dimaksudkan untuk menggantikan teknologi lama yang bersifat *server-side*, seperti CGI (Common Gateway Interface) yang memiliki beberapa kelemahan dan berjalan di lingkungan UNIX. Ciri yang dimiliki oleh aplikasi yang menggunakan ASP salah satunya adalah bahwa aplikasi tersebut memiliki ekstensi *.asp*.

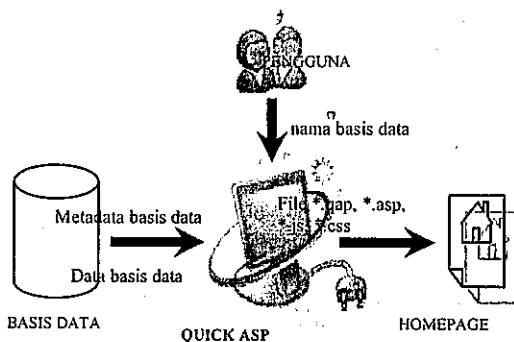
Kelebihan-kelebihan ASP dalam pembuatan aplikasi dinamis dibandingkan dengan yang lain yaitu sebagai berikut.

1. Sintaks-sintaksnya mudah dipelajari karena tidak mengenal pendeklarasian variabel dan akses tingkat rendah lainnya
2. Kode program terintegrasi dengan *file* HTML sehingga memudahkan pembuatan desain tampilan dan dapat berjalan lebih cepat dibandingkan dengan CGI
3. ASP merupakan kode program yang berorientasi pada objek dan dapat dikembangkan lebih jauh dengan menggunakan komponen-komponen *ActiveX server* atau ADO
4. Tidak ada proses *compiling* atau *linking*
5. ASP memiliki fleksibilitas, dapat berinteraksi dengan aplikasi-aplikasi web yang dibuat dengan bahasa CGI, ISAP, dan kode program lainnya.

Untuk mencoba, kode program ASP dapat dijalankan pada sebuah PC berbasis Windows tanpa terhubung ke internet dengan terlebih dahulu menginstall *Personal Web Server (PWS)* atau *Internet Information Server (IIS)*.

Pemrograman ASP merupakan program kode program sebagai alat bantu pembuatan *homepage*. Halaman-halaman web akan bergerak secara dinamis dengan adanya program kode program, sedangkan form-form HTML cenderung statis. Program ASP selalu ditandai blok program `<%` dan akhir blok `%>`. Tanda tersebut bisa disisipkan di sembarang tempat kode HTML dan file ASP harus disimpan dengan akhiran *.asp*, sedangkan tempat file harus di direktori yang dieksekusi oleh web server. *Default* tempat biasanya adalah `c:\webshare` atau `c:\inetpub\wwwroot`.

## 7. DESKRIPSI UMUM SISTEM QuickASP



Gambar 2 Gambaran Umum Sistem

Secara garis besar, perangkat lunak yang hendak dibuat berfungsi sebagai pembangkit kode program ASP beserta kode program lain sebagai pendukung untuk menghasilkan halaman web dinamis. Disebut dinamis karena pengunjung web dapat melakukan proses penambahan, pengubahan serta penghapusan data suatu tabel pada basis data tertentu. Dengan demikian, perlu dilakukan pembukaan dan pembacaan isi maupun struktur basis data. Dari gambar di atas terlihat bahwa pengguna cukup memasukkan nama basis data yang hendak dibangkitkan ke dalam kode program ASP serta memilih *stylesheet* yang telah disediakan oleh aplikasi. Selanjutnya aplikasi akan membaca isi dan struktur basis data tersebut guna pengaturan tampilan masing-masing halaman web. Kemudian pada akhirnya aplikasi akan membangkitkan kode program ASP dan beberapa kode program pendukung lainnya.

Secara detail proses-proses yang dilakukan pengguna yaitu:

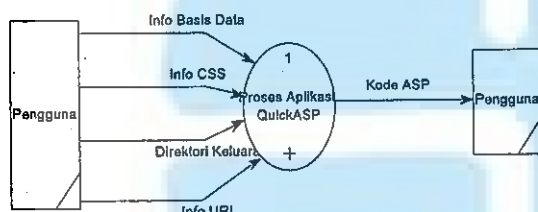
- a. Memilih antara membuat proyek baru atau membuka proyek yang sudah ada
- b. Memilih direktori untuk menyimpan file proyek yang berekstensi *\*.qap*. File yang berekstensi *\*.qap* (*QuickASP Project*) ini terbentuk setiap saat aplikasi QuickASP selesai melakukan pengaturan halaman web yang akan dihasilkan. File ini berisi kode-kode hasil pengaturan halaman web yang diperlukan oleh QuickASP guna pembangkitan kode ASP.
- c. Memasukkan nama basis data beserta propertinya jika diperlukan (*username* dan *password*)
- d. Memilih tabel yang hendak dibangkitkan ke dalam kode ASP
- e. Memilih antara melakukan pembangkitan kode program secara manual (tahap demi tahap) atau melakukan pembangkitan kode program secara otomatis oleh perangkat lunak
- f. Melakukan modifikasi format tampilan masing-masing halaman web sesuai keinginan pengguna yang meliputi halaman daftar (*List Page*), halaman pengubahan data (*Edit Page*), halaman

pencarian data (*Search Page*), dan halaman login (*Login Page*). Proses ini bersifat opsional karena telah disediakan format tampilan *default* untuk masing-masing halaman web.

- g. Memilih *stylesheet* (\*.css) yang telah disediakan untuk mengatur tampilan *homepage* ketika dilihat dari *web browser*
- h. Menentukan direktori tempat penyimpanan file-file hasil pembangkitan yang berisi kode ASP serta kode program pendukung lainnya (*Java Script*)
- i. Menentukan URL untuk menampilkan halaman *default* (*default page*) dalam *web browser*.

File-file yang dibangkitkan merupakan kode ASP untuk membangun *homepage*. Komponen basis data pada sistem *homepage* merupakan sumber data bagi *homepage* tersebut. Pada saat melakukan proses pembangkitan kode ASP, basis data akan memberikan informasi berupa struktur tabel-tabel, nama *field* dan tipe *field* yang ada didalamnya. Struktur tabel akan digunakan untuk menentukan nama direktori penyimpanan file-file ASP, navigasi pada *homepage*, menentukan nama dan tipe masukan pada *homepage*.

### 8. PEMODELAN PROSES



Gambar 3 DFD Level 0

Pada DFD Level 0 terdapat sebuah proses utama yakni proses aplikasi QuickASP. Dimulai dengan pengguna memasukkan info basis data, memilih CSS yang telah disediakan, menentukan direktori keluaran bagi file-file ASP dan file-file pendukung, serta menentukan URL untuk melihat hasil pembangkitan pada *web browser*. Sehingga pada akhirnya pengguna akan menerima file-file yang berisi kode-kode ASP serta kode program pendukung lainnya.

Kemudian sebagaimana yang terlihat pada Gambar 4, proses 1 pada level 0 dijabarkan menjadi lima proses pada level 1 yaitu :

- a. Proses Pemetaan Isi dan Struktur Basis Data

Berdasarkan info basis data yang dimasukkan pengguna meliputi nama server, nama basis data, serta info lain yang berhubungan dengan sekuritas basis data, perangkat lunak membuka basis data tersebut. Selanjutnya dilakukan pembacaan struktur basis data meliputi nama tabel, nama *field*, tipe data, panjang maksimum karakter, *primary key*, dan *foreign key*.

- b. Proses Pengaturan Format Tampilan Halaman Web

Setelah dilakukan pembacaan struktur basis data, kemudian aplikasi menampilkan format *default* tampilan untuk tiap-tiap halaman web. Format tampilan ini meliputi navigasi *record* pada halaman daftar (*List Page*), tipe masukan pada halaman perubahan data (*Edit Page*) yang telah disesuaikan dengan tipe data masing-masing *field*, tampilan halaman pencarian (*Search Page*), serta tampilan halaman login (*Login Page*).

- c. Proses Pemilihan Direktori Keluaran

Proses ini digunakan untuk memilih sebuah direktori di *server* yang akan digunakan sebagai tempat menyimpan file-file kode ASP yang dibangkitkan.

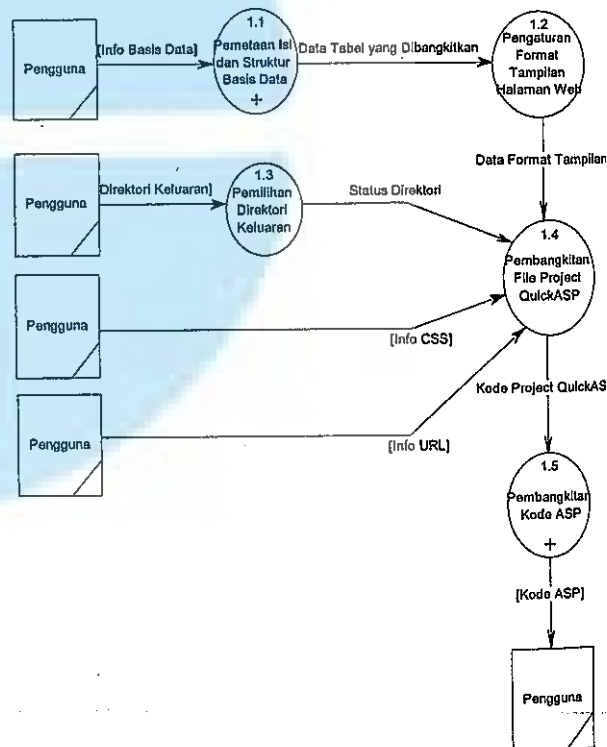
- d. Proses Pembangkitan File Proyek QuickASP

Proses ini dilakukan untuk membangkitkan file proyek berekstensi \*.qap yang berisi seluruh pengaturan halaman web termasuk data URL dan direktori keluaran.

- e. Proses Pembangkitan Kode ASP.

Proses ini dilakukan untuk membangkitkan file-file ASP beserta file-file pendukungnya (*Java Script*, *CSS*).

Gambar 4 DFD Level 1



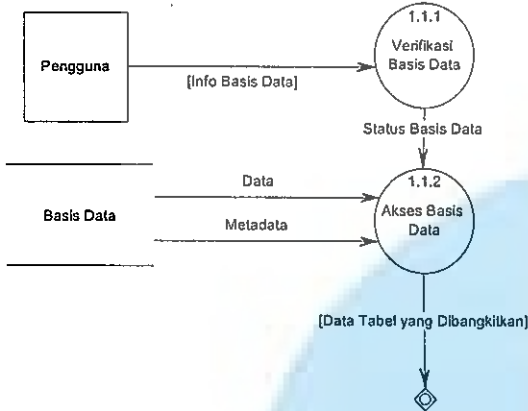
Pada Gambar 5 berikut digambarkan DFD level 2 yang merupakan penjelasan dari proses 1.1 (Proses Pemetaan Isi dan Struktur Basis Data), yang terbagi menjadi dua sub proses, yaitu :

- a. Proses Verifikasi Basis Data

Proses ini dilakukan untuk memeriksa apakah info basis data yang dimasukkan pengguna valid dan apakah basis data sudah ada di direktori server.

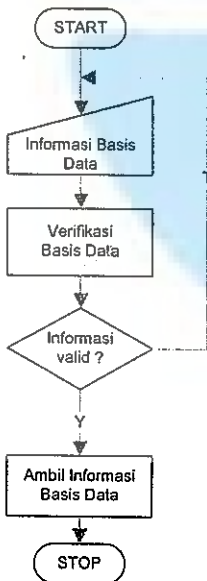
b. Proses Akses Basis Data

Jika status basis data valid, maka proses selanjutnya adalah mengakses basis data dan mendapatkan informasi mengenai isi basis data serta metadata basis data yang meliputi nama tabel, nama *field*, tipe data, panjang maksimum karakter, *primary key*, *foreign key*, *nullable field*, serta *identity field*.



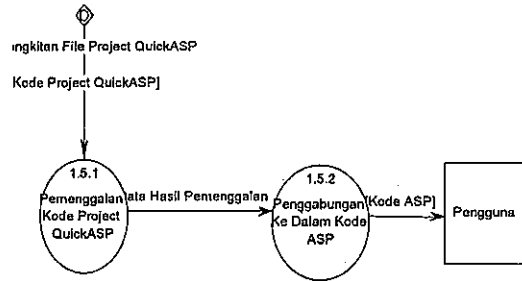
Gambar 5 DFD Level 2 Proses 1.1

Berikut ini adalah diagram alir untuk kedua proses tersebut:



Gambar 6 Diagram alir Proses 1.1

Selanjutnya pada Gambar 6 di bawah terlihat gambar DFD level 2 yang merupakan penjelasan dari proses 1.5 (Pembangkitan Kode ASP).



Gambar 6 DFD Level 2 Proses 1.5

Proses pembangkitan kode ASP terbagi menjadi dua sub proses yaitu:

- a. Proses Pemenggalan Kode Proyek QuickASP  
Setiap kali pengguna selesai melakukan pengaturan halaman web yang hendak dibangkitkan, aplikasi melakukan pembangkitan file proyek yang berekstensi \*.qap untuk menyimpan hasil pengaturan tersebut. Selanjutnya file tersebut dibaca dan dilakukan pemenggalan (*parse*) isi file.
- b. Proses Penggabungan ke dalam Kode ASP  
Hasil pemenggalan isi file proyek tersebut selanjutnya dimasukkan ke dalam struktur data array guna mempermudah pada saat penggabungan dengan kode program ASP yang hendak dibangkitkan.

9. HASIL UJI COBA

Uji coba dilakukan dengan tujuan:

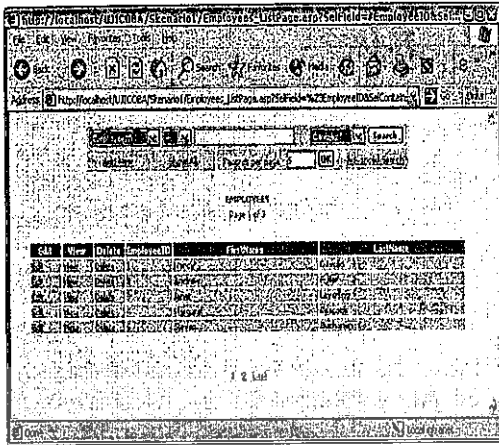
1. Mengetahui apakah QuickASP dapat membangkitkan file-file ASP beserta file-file pendukungnya; dan
2. Apakah halaman web hasil pembangkitan QuickASP dapat melakukan proses menambah, menghapus, mengubah data serta proses menu navigasi.

▪ Skenario pertama

Tujuan skenario yang pertama adalah untuk menguji QuickASP dalam proses pembangkitan kode ASP dengan masukan sebagai berikut:

- Tipe basis data : SQL Server
- Nama basis data : Northwind
- Nama tabel : Employees
- File CSS : Ocean.css
- Metode : Manual

Pada uji coba pertama ini, server yang digunakan sebagai tempat menyimpan file-file ASP hasil pembangkitan kode QuickASP merupakan komputer lokal.



Gambar 7 Halaman Web Hasil Pembangunan Skenario Pertama

■ Skenario Kedua

Tujuan skenario yang kedua adalah untuk menguji QuickASP dalam proses pembangunan kode ASP dengan metode pembangunan otomatis beberapa tabel sekaligus.

Pada skenario yang ke-dua, data masukannya sebagai berikut:

- Tipe basis data: SQL Server
- Nama basis data: pubs
- Nama tabel: authors, discount, employee, jobs, pubinfo, publishers, roysched, sales, stores, titleauthor, titles
- File CSS: Nature.css
- Metode: Automatically



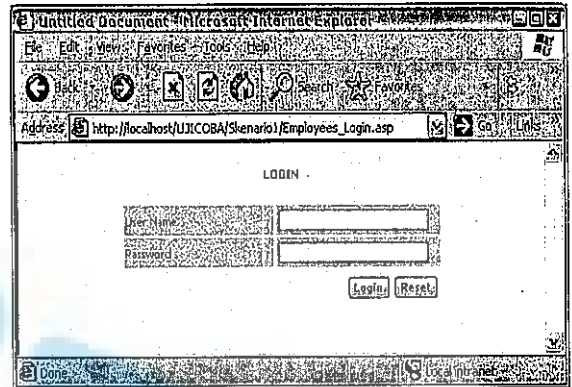
Gambar 8 File-File Hasil Pembangunan Skenario Kedua

■ Skenario Ketiga

Pada skenario ketiga, aplikasi akan membuka file proyek yang sudah dibuat pada skenario 1 lalu

melakukan modifikasi yaitu menambahkan halaman login.

Setiap kali pengguna membuka form pengaturan tampilan, maka secara otomatis aplikasi QuickASP akan menampilkan hasil pengaturan pada proyek yang dibuka.



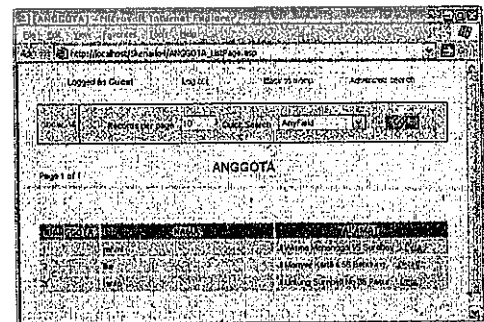
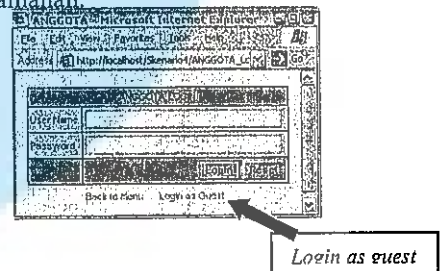
Gambar 9 Tampilan Halaman Login Hasil Pembangunan Skenario 3

■ Skenario Keempat

Pada skenario keempat ini, aplikasi melakukan pembangunan kode program ASP dengan masukan sebagai berikut :

- Tipe basis data : Oracle
- Nama basis data : RAHMI
- Nama tabel : Anggota
- File CSS : Rose.css
- Metode pembangunan : Manual

Selain itu pada skenario ini akan dilakukan pembangunan halaman login dengan berbagai pilihan tingkat keamanan.



Gambar 10 Tampilan Halaman Daftar untuk Login sebagai Tamu

## 10. KESIMPULAN

Beberapa kesimpulan yang dapat diambil dari proses pembuatan QuickASP pada penelitian ini adalah:

1. QuickASP dapat melakukan proses pembangkitan kode ASP dari basis data Microsoft Access, Microsoft SQL Server, dan Oracle.
2. QuickASP dapat menghasilkan homepage dinamis. Homepage yang dihasilkan dilengkapi dengan fungsi menambah data, mengubah data, menghapus data dan menu navigasi.
3. QuickASP dapat menampilkan pada halaman web data-data dengan tipe data yaitu *text*, *numeric*, *date*, *boolean* dan *image* (.gif, .jpg, .bmp).
4. QuickASP merupakan aplikasi desktop, bukan aplikasi berbasis web. Sehingga untuk menggunakan QuickASP, pengguna perlu melakukan instalasi QuickASP di komputer lokal.

## 11. KEMUNGKINAN PENGEMBANGAN

Beberapa saran yang dapat penulis berikan untuk penelitian ini adalah:

1. QuickASP saat ini berupa aplikasi desktop, sehingga bisa dikembangkan menjadi aplikasi QuickASP berbasis web.

2. Halaman yang dihasilkan bisa dikembangkan ke bahasa pemrograman web lainnya seperti JSP dan PHP sehingga memberikan banyak pilihan bagi pihak pengguna aplikasi.
3. Adanya fitur pemilihan *template* sehingga pengguna dapat memiliki banyak pilihan desain web yang beragam dan menarik.
4. Penambahan jenis DBMS yang dapat diakses oleh QuickASP dan homepage yang dibangkitkan, seperti MySQL.

## 12. DAFTAR PUSTAKA

1. ASP tutorial at <http://w3schools.com/asp>, 1999.
2. Castagnetto, Jesus, and Rawat, Harish, and Schumann, Sascha, "Professional ASP Programming". Chris Scollo, Deepak Veliath: 1999.
3. Isharyono, Andrie. "P3L Database Management Berbasis Web untuk Database MySQL dengan Menggunakan PHP". 2001.
4. Kurniawan, Yahya, S.T, "Aplikasi Web Database dengan ASP", PT. Elex Media Komputindo, Jakarta: 2002.
5. Microsoft Corporation, Microsoft Developer Network Library, <http://msdn.microsoft.com>. Juli 2001.
6. ANDI, "Panduan Aplikatif Pengembangan Web Berbasis ASP", Yogyakarta: 2002.