

Citizen Services Portal¹

The Citizen Services Portal (CSP) was conceptualised jointly by the Panchmahals District Administration, Gujarat (PDA), and the Centre for Electronic Governance at the Indian Institute of Management, Ahmedabad (CEG). Using the power of Information and Communication Technology (ICT), the CSP is designed to reach out to the rural masses by facilitating access to information and services of the government through an Internet portal and PC based information kiosks. The CEG has designed and developed the pilot version of the CSP with the enthusiastic participation of the PDA and other government departments of the Panchmahals district. The pilot version of CSP was launched on 4th October, 2001, at 14 locations. The pilot version of CSP is hosted on the IIMA web server: <http://202.41.76.161:8080/godhra/>, which can be accessed from any Internet connected PC using a simple browser.

The overall objective of the project is to offer the rural citizens an improved access to information on government schemes and services and also facilitating the process of applying for the same from their villages. The CSP is expected to enhance transparency in government functioning and offer scope for improved service delivery.

CEG-IIMA has chosen Panchmahals as the pilot district because it is predominantly a tribal district and most importantly due to the enterprising efforts of the district collector in involving the various departments at the collectorate and the concerned agencies in the state capital of Gandhinagar.

In the pilot district, the locations like STD/PCO booths, milk collection centers of dairy cooperative societies, cooperative bank counters, etc., which have PCs, are identified to be the potential information kiosk agencies. These are the locations, which have telephone connectivity and are also the places that the rural citizens visit on day-to-day basis. These agencies are being facilitated to build the necessary infrastructure through PMRY (Prime Minister's Rojgar Yojana) and Internet Dhaba schemes. In the pilot phase around 20 such agencies are involved in offering the information services to the people of Panchmahals district through the portal.

Since the connectivity and internet response times are major concerns for the success of such projects, a large portion of the portal is copied onto CDs (Compact Disks) so that they can be operated on standalone PCs to offer the services. Internet connectivity will however be required for on-line applications, to obtain the status of submitted applications and to access information from related sites. It is planned to replace the CDs from time to time with updated contents. This distributed organization of the portal is expected to reduce the network traffic significantly and more importantly reduce the dependence on telephone connectivity and the internet service providers.

¹ Case Prepared by Prof. T.P. Rama Rao, Mr. Dhawal Bhatia, Mr. Kalpesh Mehta, and Ms. Monica Raina of Center for Electronic Governance, Indian Institute of Management, Ahmedabad, (2002).

The Citizen Services Portal is a gateway to information on citizen services from twenty government departments, which route some or all of their services through the district collectorate. These include the Collectorate, Prant office, Police, Public Distribution System (PDS), District Planning Board (DPB), District Industries Centre (DIC), District Rural Development Agency (DRDA), Tribal Area Sub-Plan (TASP), Zilla Panchayat (ZP), State Transport (ST), Road Transport Office (RTO), Department of Telecom (DoT), Sales Tax (ST), Employment Exchange, and Forestry. In addition, the portal also provides information on livelihood related activities like agriculture, animal husbandry, employment avenues etc. A help-line offering expert advice on matters related to law, medicine, career, agriculture etc. is also provided on the portal.

Information relevant to citizens about these departments was obtained, translated into easily understandable Gujarati language, and created as web pages using the Gujarati font software. The information is largely on the organization of the department, services and schemes offered, a checklist of documents, down-loadable forms required to apply for schemes / services and the citizen's charter. The citizen's charter gives the information on the services offered by the department, the people / departmental heads to be contacted, forms and documents to be submitted, and the maximum duration for the processing of the service.

Hyperlinks were created on the portal to offer the information services from other related government sites. Some of these links are: Gujarat Informatics Limited (GIL)'s application for retrieval of government resolutions (GR Book); Remote Sensing and Communications Unit (RESECO)'s applications using Gujarat Geographical Information System (GGIS); and National Informatics Centre (NIC)'s applications like Public Grievance Redress And Monitoring System (PGRAMS) etc.

CEG has interacted with several people in designing the framework of the portal. These include the rural citizens of Panchmahals, local elected representatives, and development workers. Through these interactions, it was identified that the rural citizens of Panchmahals need information on the processes to be followed to start a community project or a new venture, in addition to the information on the schemes they can avail of and the progress of their applications. Several experts have provided inputs to the design of the portal. Based on all these suggestions, the portal was enhanced to cover the relevant information to the citizens and to create a sustainable information management system. The Collector, Panchmahals district had got a Participatory Rural Appraisal (PRA) exercise conducted, to prioritise the information services to be offered to the citizens through the portal. The PRA exercise revealed that the citizens need information on livelihood opportunities, best practices in agriculture, animal husbandry, drought management etc. The collector also provided the necessary state government interface to ensure that the concerned departments provide the latest information about their services to citizens from time to time and also facilitate the acceptance of forms, which are submitted online. The Chief Secretary, Government of Gujarat, Principal Secretary, Revenue, Secretary, Higher & Technical Education and the Development Commissioner were also consulted during the development of the portal to obtain feedback and to initiate processes for the state government's participation in this proof-of-concept project.

During the course of the development, CEG held several meetings with the various department officials, their heads, community leaders, elected representatives and the common man. The voluminous data received from the departments was categorized and converted to easily understandable local language format. Services of a retired state government official were used for two months for this language conversion task.

At the pilot stage, the portal has around 700 web pages (all of them in Gujarati), 130 downloadable forms, information on 45 schemes and 25 citizen charters, covering almost 20 departments. Citizens can access the relevant information or forms by navigating through the pages of the concerned department or by giving their profile to the search-engine, which retrieves the details of the schemes that match the submitted profile. In addition, delivery of some components of citizen services will be handled in electronic mode through this portal. For example, the form to avail the National Old Age Pension Scheme (NOAPS) can be submitted online and the applicant can view its processing status by giving the computer-generated registration number. Similarly, in the future, applications for ration cards can be submitted online and the citizen only needs to make one trip to the office to pick up the ration card.

CEG organized on-site workshops to familiarize the department officials with CSP, and to create a sense of ownership in them. Workshops were also held to train and acclimatize the CSP to the operators. This was a hands-on training on the navigational and other key aspects of the portal. They were also given an orientation on adopting a service approach towards the citizens.

CEG as well as the PDA will collect a detailed feedback during the pilot run. Based on the experience of the kiosk agencies, users and the collectorate, the web pages and features will be modified and enhanced, before replicating in other districts.

Technologies used for Application Development:

The software tools used for the development of the CSP are presented in Exhibit –I. The system was developed using the Win 98 clients with MS FrontPage, MS Office and Adobe Acrobat (for static page development), Java Development Kit 1.2 (for active document creation) and Java Servlets Development Kit (for dynamic content generation).

The software on the host server is Unix OS with Oracle 8i Application Server and Apache Web Server.

CSP contains about 130 forms, 45 schemes and 700 pages, cutting across 20 departments. Interactive user interfaces were built for NOAPS, Grievance Redressal, Search Engine and Feedback sections. About 15 Java Servlets, using the JSDK 2.0 protocol and Java Development Kit (JDK) 1.2 have been developed. The Oracle 8i Application Server and Apache Web Server are the software servers used for running these servlets. The approximate space occupied by different components of CSP is given in the Exhibit-I

Currently CSP is hosted on the CEG server (Compaq Proliant 1600), with Unixware 7 as the OS and Oracle 8i as the database software. The configuration of the CEG server is presented in the Exhibit-II.

Exhibit –I

Development Environment (at CEG clients):

Function	Software	Disk Space	Memory Required
Operating System	Windows 98	400 MB	32 MB
Content Generation (HTML)	Microsoft FrontPage 2000 V 4.0.2	167 MB	8 MB
	MS Office 97	100 MB	32 MB for Office 8 MB for Word
	Adobe Acrobat (open source - ADOBE website)	20 MB; (60 MB Temp- during installation)	8 MB
Dynamic Content Generation	Java Servlets Development Kit 2.0 (open source - SUN website)	70 MB (software) 120 MB (Documntn)	8 MB - 32 MB

CSP Software size

2 MB tables in database,

HTML data - 50 MB,

Java Servlets executables (class files) - 300 KB,

Fonts exe - 550 KB for local language fonts (Gujarati)

Estimated Traffic on the server:

14 Kiosks, 2000 - 3000 hits per month

Runtime Environment (for remote users):

Kiosks:

- Any PC with Windows OS
- Telephone lines
- Modem for Internet connectivity (minimum speed - 38400 BPS)
- Printer and
- CD-ROM drive

Users:

Browser on Windows OS

Internet Explorer 4.0 or above

Netscape Navigator 4.0 or above

Exhibit – II

Server Hardware (CEG server):

- Compaq Server PROLIANT 1600
- 256MB SDRAM
- Three 9GB Hard Disk Drives
- Network Controller: COMPAQ 10/100 TX PCI UTP controller (embedded) on the PCI local bus

Server	Compaq ProLiant 6/450
Processor	450 MHz Intel Pentium II
Bus Speed	100 MHz
Architecture	PCI/ISA
RAM Std/Max	256MB SDRAM / 1GB
Drive Controller	Dual Channel Wide-Ultra SCSI-3
Slots	6 Total =2 PCI, 4 PCI/ISA

Chipset:

The Compaq ProLiant 1600 6/450 is using the Intel 440BX chipset. This chipset supports the new 100MHz GTL bus and the 100MHz SDRAM memory. This new technology provides a significant performance increase over the 66MHz bus and EDO DRAM memory design of the first Pentium II-based systems.

Server Software:

Function	Software	Disk Space	Memory Required
Operating System	SCO UnixWare 7.0	1 GB	32 MB
Database	Oracle 8i Enterprise version	811 MB	128 MB
Webserver	Apache HTTP Server Version 1.3	160 MB	840 KB / daemon process (250 KB disk)

Access of CEG Sever through Internet:

The CEG server is accessible through its IP address 202.41.76.161. Port 8080 is assigned to the Apache web server and all web sites hosted on the CEG server are accessible through this port. Port 80 is the default port for Apache web servers. Each web site is assigned a root directory under the apache web server. The root directory for CSP portal is "godhra".

Hence the complete URL is <http://202.41.76.161:8080/godhra/> where

HTTP	Hyper Text Transport Protocol used to access html pages.
202.41.76.161	Internet Protocol Address (IP) of the server on which the web site or the portal is hosted
8080	port number on which the web server will entertain requests form remote clients
godhra	root directory of the CSP portal where all the html pages, images, pdf files will be available.

The Apache HTTP Server

▪ Introduction

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows NT. It is a collaborative software development effort aimed at creating a robust, commercial-grade, featureful, and freely-available source code implementation of an HTTP (Web) server that provides HTTP services in sync with the current HTTP standards. Apache² has been the most popular web server on the Internet since April of 1996. The August 2002 [Netcraft Web Server Survey](#) found that 63% of the web sites on the Internet are using Apache, thus making it more widely used than all other web servers combined. The Apache HTTP Server is a project of the [Apache Software Foundation](#).

▪ What is Apache?

The Apache httpd server

- is a powerful, flexible, HTTP/1.1 compliant web server
- implements the latest protocols, including HTTP/1.1 (RFC2616)
- is highly configurable and extensible with third-party modules
- can be customised by writing 'modules' using the Apache module API
- provides full source code and comes with an unrestrictive license runs on Windows NT/9x, Netware 5.x and above, OS/2, and most versions of Unix, as well as several other operating systems
- is actively being developed
- encourages user feedback through new ideas, bug reports and patches

▪ Implemented Features

DBM databases for authentication:

Allows you to easily set up password-protected pages with enormous numbers of authorized users, without bogging down the server.

Customized responses to errors and problems:

Allows you to set up files, or even CGI scripts, which are returned by the server in response to errors and problems, e.g. setup a script to intercept **500 Server Errors** and perform on-the-fly diagnostics for both users and yourself.

Multiple DirectoryIndex directives:

Allows you to say DirectoryIndex index.html index.cgi, which instructs the server to either send back index.html or run index.cgi when a directory URL is requested, whichever it finds in the directory.

Unlimited flexible URL rewriting and aliasing:

Apache has no fixed limit on the numbers of Aliases and Redirects which may be declared in the config files. In addition, a powerful rewriting engine can be used to solve most URL manipulation problems.

² <http://httpd.apache.org/>

Content negotiation

i.e. the ability to automatically serve clients of varying sophistication and HTML level compliance, with documents which offer the best representation of information that the client is capable of accepting.

Virtual Hosts

A much requested feature, sometimes known as multi-homed servers. This allows the server to distinguish between requests made to different IP addresses or names (mapped to the same machine). Apache also offers dynamically configurable mass-virtual hosting.

Configurable Reliable Piped Logs

You can configure Apache to generate logs in the format that you want. In addition, on most Unix architectures, Apache can send log files to a pipe, allowing for log rotation, hit filtering, real-time splitting of multiple vhosts into separate logs, and asynchronous DNS resolving on the fly.

Apache FAQ:

Does or will Apache act as a Proxy server?

Apache version 1.1 and above comes with a [proxy module](#). If compiled in, this will make Apache act as a caching-proxy server.

What are "multiviews"?

"Multiviews" is the general name given to the Apache server's ability to provide language-specific document variants in response to a request. This is documented quite thoroughly in the [content negotiation](#) description page. In addition, *Apache Week* carried an article on this subject entitled "[Content Negotiation Explained](#)".

Why doesn't Apache include SSL?

SSL (Secure Socket Layer) data transport requires encryption, and many governments have restrictions upon the import, export, and use of encryption technology. If Apache included SSL in the base package, its distribution would involve all sorts of legal and bureaucratic issues, and it would no longer be freely available. Also, some of the technology required to talk to current clients using SSL is patented by [RSA Data Security](#), who restricts its use without a license.

Does Apache include a search engine?

Apache does not include a search engine, but there are many good commercial and free search engines which can be used easily with Apache. Some of them are listed on the [Web Site Search Tools](#) page. Open source search engines that are often used with Apache include [ht://Dig](#) and [SWISH-E](#).

How can I rotate my log files?

1. by piping the transfer log into an appropriate log file rotation utility.
2. In the src/support/ directory, you will find a utility called [rotatelogs](#) which can be used like this: `TransferLog "|/path/to/rotatelogs /path/to/logs/access_log 86400"` to enable daily rotation of the log files.
3. Another logfile rotation utility "*cronolog*" is available from <http://www.cronolog.org/>. It can automatically create logfile subdirectories based on time and date, and can have a constant symlink point to the rotating logfiles.

How do I keep certain requests from appearing in my logs?

The maximum flexibility for removing unwanted information from log files is obtained by post-processing the logs, or using piped-logs to feed the logs through a program which does whatever you want. However, Apache does offer the ability to prevent requests from ever appearing in the log files. You can do this by using the [SetEnvIf](#) directive to set an environment variable for certain requests and then using the conditional [CustomLog](#) syntax to prevent logging when the environment variable is set.

Does Apache support any sort of database integration?

No. Apache is a Web (HTTP) server, not an application server. The base package does not include any such functionality. See the [PHP project](#) and the [mod_perl project](#) for examples of modules that allow you to work with databases from within the Apache environment.

Can I use Active Server Pages (ASP) with Apache?

The base Apache Web server package does not include ASP support. However, there are a couple of after-market solutions that let you add this functionality; see the [related projects](#) page to find out more.

Does Apache come with Java support?

The base Apache Web server package does not include support for Java, Java Server Pages, Enterprise Java Beans, or Java servlets. Those features are available as add-ons from the Apache/Java project site, <URL:<http://jakarta.apache.org/>>.

A few important parameters in the configuration file of the Apache Server...

1. StartServers directive ** Value on CSP Server = 5 (default)**

Syntax: StartServers number

Default: StartServers 5

Context: server config

Status: core

The StartServers directive sets the number of child server processes created on startup. As the number of processes is dynamically controlled depending on the load, there is usually little reason to adjust this parameter.

On Windows:

When running under Microsoft Windows, this directive has no effect. There is always one child which handles all requests. Within the child requests are handled by separate threads. The ThreadsPerChild directive controls the maximum number of child threads handling requests, which will have a similar effect to the setting of StartServers on Unix.

2. MinSpareServers directive ** Value on CSP Server = 5 (default)**

Syntax: MinSpareServers number

Default: MinSpareServers 5

Context: server config

Status: core

The MinSpareServers directive sets the desired minimum number of idle child server processes. An idle process is one which is not handling a request. If there are fewer than MinSpareServers idle, then the parent process creates new children at a maximum rate of 1 per second. Tuning of this parameter should only be necessary on very busy sites. Setting this parameter to a large number is almost always a bad idea.

3. MaxSpareServers directive ** Value on CSP Server = 10 (default)**

Syntax: MaxSpareServers number

Default: MaxSpareServers 10

Context: server config

Status: core

The MaxSpareServers directive sets the desired maximum number of idle child server processes. An idle process is one which is not handling a request. If there are more than MaxSpareServers idle, then the parent process will kill off the excess processes. Tuning of this parameter should only be necessary on very busy sites. Setting this parameter to a large number is almost always a bad idea.

This directive has no effect when used with the Apache Web server on a Microsoft Windows platform.

4. MaxRequestsPerChild directive ** Value on CSP Server = 30 **

Syntax: MaxRequestsPerChild number

Default: MaxRequestsPerChild 0

Context: server config

Status: core

The MaxRequestsPerChild directive sets the limit on the number of requests that an individual child server process will handle. After MaxRequestsPerChild requests, the child process will die. If MaxRequestsPerChild is 0, then the process will never expire.

5. MaxClients directive ** Value on CSP Server = 150 **

Syntax: MaxClients number

Default: MaxClients 256

Context: server config

Status: core

The MaxClients directive sets the limit on the number of simultaneous requests that can be supported; not more than this number of child server processes will be created. To configure more than 256

clients, you must edit the `HARD_SERVER_LIMIT` entry in `httpd.h` and recompile. Any connection attempts over the `MaxClients` limit will normally be queued, up to a number based on the `ListenBacklog` directive. Once a child process is freed at the end of a different request, the connection will then be serviced.

6. **ThreadsPerChild** directive (for Windows OS only)

Syntax: `ThreadsPerChild` number

Default: `ThreadsPerChild 50`

Context: server config

Status: core (Windows, NetWare)

Compatibility: Available only with Apache 1.3 and later with Windows This directive tells the server how many threads it should use. This is the maximum number of connections the server can handle at once; be sure and set this number high enough for your site if you get a lot of hits.

This directive has no effect on Unix systems. Unix users should look at `StartServers` and `MaxRequestsPerChild`.

7. **Port** directive

**** Value on CSP Server = 8080 ****

Syntax: `Port` number

Default: `Port 80`

Context: server config

Status: core

Number is a number from 0 to 65535; some port numbers (especially below 1024) are reserved for particular protocols. The standard port for the `http` protocol is 80. In the absence of any `Listen` or `BindAddress` directives specifying a port number, a `Port` directive given in the "main server" (i.e., outside any `<VirtualHost>` section) sets the network port on which the server listens. If there are any `Listen` or `BindAddress` directives specifying `:number` then `Port` has no effect on what address the server listens at. Port 80 is one of Unix's special ports. All ports numbered below 1024 are reserved for system use, i.e., regular (non-root) users cannot make use of them; instead they can only use higher port numbers. To use port 80, you must start the server from the root account. After binding to the port and before accepting requests, Apache will change to a low privileged user as set by the `User` directive. If you cannot use port 80, choose any other unused port. Non-root users will have to choose a port number higher than 1023, such as 8000.

SECURITY: if you do start the server as root, be sure not to set `User` to root. If you run the server as root whilst handling connections, your site may be open to a major security attack.

CSP Server Capability:

Max Child Server Processes = 10

Max Requests per Child Server Processes = 30

Max Simultaneous Clients requests to the Server = 150

Load Balancing of all the connection(requests) is done by the Apache Web Server itself (internally).