



ViewsFlash Application Architecture

Description

Cogix ViewsFlash is a J2EE web application that creates questionnaires for surveys, assessments, data collection, quizzes, tests and polls. The application presents the questionnaires to respondents, tabulates the results in real time, provides analytical reports and charts, and accesses responses.

Questionnaires

Questionnaire forms are created using a what-you-see-is-what-you-get editor, which also includes Wizards for creating a matrix of questions with identical responses and grids of questions of any type applied to different areas. Multi-lingual questionnaires in all languages are supported, and questionnaires comply with web accessibility guidelines required by ADA Section 508 and European WCAG Level 2, including XHTML Strict.

Questionnaires support all common types of data entry including radio buttons, check boxes, drop-down menus, text fields and text areas. Standard Javascript and server-side validations include Must Respond, Numeric value, Email address, Date. Creative question presentations include Elastic Text, where a text box grows and shrink to accommodate varying amounts of text in a comments box, for example.

Questionnaires are personalized by looking up respondent attributes in database tables and LDAP stores, by piping content from responses in earlier pages and looking up appropriate content, by branching to different pages, and by hiding or showing relevant sections depending on previous responses.

A server-side Script facility implements complex validation rules and allows creating measurement and scoring systems. Script is an implementation of JSR 274, the Java-like Beanshell scripting language. It is also useful for arbitrarily complex server-side calculation, scoring, validation, branching and logic. It is also possible to incorporate custom Java code to perform any server-side action necessary.

Questionnaire creation includes a menu to specify font, size, colors, headers, footers, and decorating images. For further customization, the HTML Styles which are used to create questionnaires can be totally customized to adapt look and feel to any web site. Questionnaires can be presented as a self-contained web page, embedded in a web site using iFrames, embedded inside JSP pages, and presented in a Portlet in JSR 168 compliant portals. Questionnaire invitations can be sent by email, from a list stored in a database, LDAP store, or an uploaded list.

Data

Data gathered from responses to a questionnaire is usually stored in its own table in a relational database, created dynamically, in the natural format of one row per response, one column per question. Response data can also be stored in a normalized table, where each row contains the response to one question and includes tags that identify the questionnaire, respondent, and question.

All response data can be retrieved to a browser, where it can be sorted by column, or in Excel. The data can also be accessed directly from the database.

Responses are tallied in real time for live presentation as poll results or in real time displays. On demand analytical reports with customizable visual charts and numerical tables include two and three way cross tabulations and univariate analysis. Reports can be saved and refreshed, in real time, from their URLs.

Production

Once deployed, one-time configuration tasks may include customization of styles to a particular look and feel, and creation of prototypical questionnaires and question sets for storage in question libraries. Questionnaire deployment is completely automatic and requires no IT involvement; authorization rights for publishing and staging can be granted to specific people or roles to control it. All styles, specialized question formats, scripts, questionnaires, and question sets in the library are designed for maximum reuse, while the application is a “set it and forget it” from the system administrator’s point of view.

Security

User authentication to the administrative functions is usually delegated to the J2EE application server and configured as declarative security in web.xml. Different users and roles can be authorized with the right to create questionnaires, analyze data, and retrieve responses.

Respondents who fill out questionnaires can be anonymous and unauthenticated, including a captcha challenge, or can be authenticated with J2EE declarative security, or can use a variety of other authentication methods. Respondents can be restricted to submitting one response, or allowed to enter many responses, or allowed to modify their response.

Extensions are available for implementing custom authentication and authorization methods, for supporting additional databases, and for creating custom server-side behaviors during questionnaire processing. REST Web Services are provided for discovering questionnaire content, retrieving live tallies, and other uses where other applications need to learn about the internal application state.

Deployment

Cogix ViewsFlash is a J2EE Web application, with an optional portlet for deployment in JSR-168 compliant portals as a web module. The application is distributable and can be deployed on clusters or

stand-alone servers. Single server deployment ViewsFlash is deployed as a .war file using the standard deployment mechanisms of the application server. A JNDI data source connects the application to an existing database.

Clustered deployment, caching, and synchronization are identical to single server deployment, but the application server deploys the application to the instances and servers in the cluster. All instances share the same database and use it to synchronize their cached content. For example, when a user modifies a questionnaire on one server, other servers take note of the change by querying the database periodically, and refresh their internal caches accordingly.

For maximum security, the questionnaire management interface on external servers can be disabled, and a server inside the firewall can manage questionnaire creation and analysis.

Staging It is possible to set up staging servers, with automatic and manual methods available for promoting changed elements such as a questionnaire from one server to the next higher level, such as development to test to staging to production. These mechanisms are also used for exporting and importing questionnaires between installations such as a development shop and a client.

Footprint For small and low-volume questionnaires, 128M RAM and 1G database space, in addition to the application server, are sufficient. For high volume and large questionnaires, 1G RAM and 10G disk space are recommended, as well as clustering among multiple servers and CPUs. ViewsFlash has tabulated votes in real time in applications as large as the MTV Video awards, at millions of votes per minute.

Supported platforms Java VM: 1.3 or higher. Hardware: X86, Sparc, pSystem, PA. Operating Systems: Linux, Solaris, HPUX, Windows Server/XP/Vista. Application Servers: Tomcat, WebSphere, Weblogic, Sun, OC4J, JBoss, JRun. Databases: Oracle 8 and above, SQL Server, DB2, MySQL. Portals: WebSphere, Weblogic, Liferay, JBoss, Vignette.

Support

ViewsFlash is supported by telephone and e-mail, with a one hour response time guarantee during business hours PST. Higher SLA agreements are available. Upgrades and patches are provided for versions up to one year old. Cogix provides professional services for simple and complex deployment, planning, constructing complex questionnaires, customization, as well as training.

Hosted Services

Cogix provides hosted deployments of its software as a service, with 99% guaranteed availability. Cogix servers are hosted at above.net with redundant equipment and backbone connections.

For further information, contact Anna.Wiesjahn@cogix.com, 1 831 657 9541.