FINAL

This report is confidential and intended solely for the use and information of the organization to which it is addressed.

**IMPLEMENTATION PLAN**

H

i

**Table of Contents**

**Project/System Information 1**

**1.**

1.1

1.2

1.3

1.4

Project Objectives 1

System Description 1

Assumptions 2

Document Organization 3

**2.**

**Implementation Methodology 3**

2.1

2.2

2.3

2.4

2.5

2.6

Implementation Environment 3

Dependencies 4

Configuration Requirements 5

Implementation Procedures 6

Entry and Exit Criteria 8

Change Control Procedure 8

**3.**

**Roles and Responsibilities 8**

**4.**

**Project Schedule 9**

ii

The following Implementation Plan outlines the system implementation environment, the implementation procedures (including change control and test procedures), roles and responsibilities for implementing the system, and the proposed schedule for the implementation.

The System, scheduled for full implementation in September 2003, will track the distribution and use of funds distributed to grantees throughout the fiscal year, beginning with Fiscal Year 2004 (FY04) data. The System has been designed as a dynamic productivity tool so that its modular design can be easily extended to include many of other financial tracking applications. We have applied a structured methodology to the development of the System, offering a final product that meets the needs.

**1.**

**PROJECT/SYSTEM INFORMATION**

**1.1**

**Project Objectives**

**1.2**

**System Description**

The System is a J2EE based web application running inside the Oracle 9i Application Server housed within the client’s environment. The System is built on top of the STRUTS framework [(http://jakarta](http://jakarta.apache.org/struts%29).[apache.org/struts).](http://jakarta.apache.org/struts%29) The STRUTS framework is based on the standard J2EE design pattern that in turn is built on top of the JAVA SERVLET specification. The application has a JAVA Servlet Pages (JSP) component for the front-tier, an ActionServlet component (sub class of the Servlet object – a STRUTS component) for the middle-tier, and a Java Database Connectivity (JDBC) wrapped class component for the backend tier. The application connects both to the other System – a proprietary client database - and to an internal application specific database. The System databases are both Oracle 8.1.7. The other system database is not the responsibility of the system; however, the system has a critical dependency on the other system. The system database is a core component and a deliverable of the system. It is provided as a DUMP file to the Database Administrator (DBA) team. The application (business logic) will be provided to the client engineering team as either a Web Archive (WAR) file or a ZIP (compressed) file which contains the JAVA tree structure with all the programmatic and configuration content. The System Engineer (SE) at the client must build this into the Application Server components (Enterprise Application Archive [EAR] or WAR) for deployment into the Production Oracle 9i Application Server’s J2EE module.

The high level logical architecture of the system production environment is captured in the following diagram.

***Final SYSTEM Implementation Plan***

1

The following assumptions are made in planning the implementation of the System:

The Other System

System Production Architecture (Draft)

High Level - Context

CLIENT-DMZ

Database

Client

Production Server

Physical - Detailed

CLIENT-DMZ

Machine: Sunfire 280Rwith 2x900MHz CPU

OS: Solaris Ram: 4 gigabyte

SSL

Client Workstation

* Ram: 256MB
* OS: Windows

- Internet Explorer IE6.0+ with 128bit SSL

Production Server

* Oracle 9ias with OC4J
* system.war (application)

Backup

Pooled JDBCConnection

Oracle 8.1.7

- System

Database

Oracle 8.1.7

- GATES

Database

Systtem

Sun Machine: OS: Solaris Ram: 1Gig HDD: 80 G

**1.3**

**Assumptions**





The client SE is knowledgeable about using Oracle 9i Application Server.

The client SE in charge of installation must understand the concepts and constructs of building a web application from source code.

The client SE in charge of installation must be a JDeveloper 9.03 expert to develop and deploy web application.

The client SE in charge of installation must understand the issues around connecting to an Oracle database.





***Final SYSTEM Implementation Plan***

2

The major hardware and software components of implementing the system are as follows:

This section describes the information necessary to complete the installation of the system successfully.

**Section 2: Implementation Methodology.** Identifies the environment, dependencies, configuration requirements, and the necessary steps and entry / exit criteria for the implementation.

**Section 3: Roles and Responsibilities**. Describes the main roles and responsibilities for both the CLIENT and consultant implementation personnel.

**Section 4. Implementation Schedule**. Contains the schedule for implementing the system.



The CLIENT SE in charge of installation must have a knowledgeable DBA available who can set up the primary connections to GATES and the SYSTEM database.

The CLIENT SE in charge of installation must have all the privileges/username and passwords needed to install and configure the Application Server and the Oracle database.



**1.4**

**Document Organization**

This remainder of this Implementation Plan is organized as follows:







**2.**

**IMPLEMENTATION METHODOLOGY**

**2.1**

**Implementation Environment**

This section outlines the proposed hardware and software for implementing the system. Implementation will take place at the client Central Office in Washington, DC. Client will provide and set up, with assistance from consultant personnel, all hardware and software required for the implementation activities.

Hardware requirements

These are the provided client specifications for the Production machine.

Server Specifications (Production box)









Sun Fire 280R

2x900 MHz Sparc CPU 4GB RAM

100MB Network

Server Software requirements

These items are the core Server software required by SYSTEM.





Oracle 9i Application Server for Solaris Oracle 8.1.7 Database

***Final SYSTEM Implementation Plan***

3

Build Software

The following software must be used to build to application for the Oracle 9i Application Server.



Jdeveloper 9.3.1 (available from http://otn.oracle.com) or ANT (available from http://ant.apache.org)

JAVA/ JDK/ J2SE version 1.3.1 (available from http://java.sun.com)



Consultant will provide an ANT script to build the EAR file required for the Oracle 9i Application Server. This EAR file contains all the necessary application code (JAVA binaries/ class files) for the system to execute on the application server. The client SE can choose to install the provided source code (ZIP file) into his JDeveloper IDE and using that tool to deploy the application onto the production box.

**2.2**

**Dependencies**

This section lists the dependencies related to the correct operating and use of the system. The SE must be aware of these dependencies and components of the application to ensure correct installation.

Login Dependencies

As stated before, the system has a critical dependency on the other database. The system is an extension of the other system, providing another view into the other services and data. One of the dependencies of the system is the user login, since only valid other system users can use the system. The System Administrator (SA) for the system and other system must make sure that the system can connect to the other system. The SA must make sure that the DataSource property value found in the src/org/system/resources/SystemApp.properties file must point to the correct value. The property files are further discussed in the next section.

User Roles

Though the users are authenticated against the other system, their ROLES are kept within the system databases (specifically, the GEN\_HSUSER table). The SA must make sure that users accessing the system must also exist in this table. NOTE: The key into the GEN\_HSUSER table is the username used within the other system.

Audit Trail

Every CRUD (Create, Update and Delete) actions on the database are recorded in an equivalent AUDIT\_<name of the funding table> table. The SA can review the contents of this table to create an audit trail of a user within the system.

Logging

Log4J is used for logging purposes within the system. Log4J is initialized via the web.xml properties file for the web application (the System) found in the WEB-INF folder. The web.xml refers to a log4j.properties file; this file is also stored in the WEB-INF directory.

***Final SYSTEM Implementation Plan***

4

The following configurations are required for the system to work within the client’s environment.

**2.3**

**Configuration Requirements**

Log4J.properties

The following are the properties that can be adjusted by the SA.

***log4j.rootCategory=DEBUG, R***

The SA can change this setting by replacing the DEBUG to INFO, DEBUG, WARN, ERROR and FATAL. This limits the insertion of the log from the code into the file on the server. As the default setting provided in this release, the DEBUG filter allows any logs that have been categorized as DEBUG or greater. In other words, the system will log any item that has been categorized as DEBUG, WARN, ERROR and FATAL and not INFO. Changing this value to ERROR can have an improved performance on the application. (R is used to define the APPENDER - see below.)

***log4j.appender.R=org.apache.log4j.RollingFileAppender***

This property value MUST NOT be changed. All log category designated as R (see above) will be sent to the RollingFileAppender module to be written to the actual physical file on the system. Rolling file appender creates a log file (text) and monitors the size (see below for the size property). If the log file (text) reaches the designated size, it will archive the first one and create a new file. This process of archiving old files and creating new files will be repeated 10 times, after which the log file (text) will start overwriting the old files.

***log4j.appender.R.MaxFileSize=1MB***

The above represents the maximum size of the log file. The consultant recommends that a small log file size be maintained as the small file allows for easier usability within the application.

***log4j.appender.R.MaxBackupIndex=10***

This property defines the number of log files to keep. Since this is a rolling file system, after the number of files reaches this number, the oldest file is rewritten.

Resource files found under src\org\system\resources

***SystemApp.properties***



**# DataSource** is used as a key for looking up the Java Naming and Directory Interface (JNDI) context for the JDBC calls within SYSTEM. The application uses the value of this property and uses it to look up the DataSource object from within the JNDI of the application server. The JNDI context is described in the DataSource.xml file either by the SE or edited by hand in the file provided.

DataSource=jdbc/SystemDS



**#Hostname** is the host name of the other system machine (e.g., the server where the the other system database is available).

***Final SYSTEM Implementation Plan***

5

**#Port** is the port number for the Oracle Listener on the server (see Hostname) for the other system.

Hostname=XXX.XXX.GOV



Port=1532



**#SID** is the identifier for the other system oracle instance running on the Server identified by the Hostname property and the Port property.

SID=pclient

***Datasource.xml***

This is provided to the SE as an example. The DataSource.xml file defines the lookup name and properties for the JDBC connection object stored inside the Application Server; normally defined by the SE using the User Interface provided by the Oracle 9i Application Server.

**2.4**

**Implementation Procedures**

This section describes the prerequisites and the steps the SE has to take to correctly install the system application.

Prerequisite installations for the building the WAR file:









JDK 1.3.1

Oracle 9i Application Server

ANT (if using the ANT build script to create an EAR file) JDeveloper 9.0.3 (if using JDeveloper to create the deployment file)

Database:





Assumption: Create Database [based on CLIENT standards].

Create a Table Space to host the application [Recommend a 200MB table space called HSDATA].

Create another table space called HSINDEX [Recommended size 100MB]. Create a User called XXXX [privilege – connect, resource, create a snapshot].

Create a Database Link (DBLink) called CLIENTPROD that points to the other system (Production environment).

**For Setting up the Training Database Instance**

o Import Data from the Export Dump File.











Use Oracle Utility IMP to import the DUMP file [as provided in the installation package: hsfp.dmp] that contains all the system tables, snapshots, stored procedures, packages, triggers.



From a DOS (UNIX) command prompt, go the directory hosting hsfp.dmp file type: “imp hsfp/hsfppassword@hsdb fromuser=hsfp touser=hsfp file=hsfp.dmp log=import\_hsfp.log” to import the dump file.

***Final SYSTEM Implementation Plan***

6

Check if the Application Host machine can access the other system server.

o Run the following scripts provided in the ZIP release package

/system/dbsrc/training\_deployment (for the training instance) and

/system/dbsrc/production\_deployment (for the production instance):











sqlplus in to hsdb as hsfp;

sqlplus>@C:\{path holding .sql files}\refresh\_snapshots.sql; sqlplus>@C:\{path holding .sql files}\reset\_grant\_table.sql; sqlplus>exec pop\_master\_for\_training; and

sqlplus>exec sp\_pop\_for\_training @C:\{path holding .sql files}\reset\_other\_table.sql.



**For Setting up the Production Database Instance:**

This information will be provided as part of production release activities.

Miscellaneous prerequisites:



The following information must be added to the property files:

DataSource name: ejb-location value in data-sources.xml (in WEB-INF/ directory) must match the value of the DataSource property in org/system/resources/SystemApp.properties file.

Other system database information in the org/system/resources/SystemApp.properties file needs to be pointed to the preferred database to be used for user validation.

log4j.properties for log file name (in /WEB-INF line 11, log4j.appender.R.File=C:\\temp\\SystemApp.log) need to point to a proper location to store the log message (e.g., in the case of UNIX - (/logs/SystemApp.log)).

Session timeout in web.xml (in WEB-INF) is set to 30 (<session-timeout>30</session- timeout>).











Application Implementation:

Consultant assumes that the SE will be using the JDeveloper tool to compile and create the WAR file within the client environment3.

***JDeveloper 9.0.3 Instructions:***





Unzip the provided source code into the Project directory for Jdeveloper.

Using the previously created project, the SE can compile the java files, update the properties, and create the WAR file for deployment.

Using the Deployment tool or the Oracle 9i Application Server Administration front end, the SE can install the application into the production Application Server.



***ANT Script Instructions:***







Unzip the provided source code into the Project directory. Change your current directory (cd) to the [install]/system directory.

Assuming the ANT and JDK 1.3.1 have been installed correctly, the SE can execute the build script by typing “ant makewar” Or “ant makeear.”

This will make the WAR/ EAR file for the application under the SYSTEM deploy directory. Follow Step 3 of JDeveloper instructions.





***Final SYSTEM Implementation Plan***

7

Responsible for the following:

* Install the database and connection to the other database.
* Create / install the Web Archive from the provided source code.

Confirmation of a Proper Install:





The SE must be able to log into the Application as another system user.

Upon successful login, the “Master Plan Summary” page will be shown on the web browser for the user’s region.

**2.5**

**Entry and Exit Criteria**

This section describes the Entry and Exit criteria for the installation process of the system. The Entry Criteria describes the steps taken before the actual installation of the code and the Exit Criteria describes the steps the SE can take to verify the correct installation of the application.

Entry Criteria:

The implementation of the system database and the connection to other system must be verified (and set in the property files) before continuing the implementation steps. The steps to install the database can be found in *Section 2.4 – Database*.

Exit Criteria:

Once the database implementation is completed, the SE must follow the application implementation procedures specified in *Section 2.4.* After complete implementation, the SE must log onto the application page. The Login page for the system should appear. The SE can test if the application is in working order by logging in as a valid other system user or by checking the log files to see that system recorded no anomalous messages.

**2.6**

**Change Control Procedure**

The only change control allowed on the site is for the client to change the configuration properties as listed in *Section 2.1.* Since these are specific changes to the core client environment, it is assumed that the client SE and DBA will be adding the correct information in these configuration files.

Consultant recommends that the client document any configuration changes for future fixes and patch/ upgrade work.

**3.**

**ROLES AND RESPONSIBILITIES**

The following table outlines the key roles and responsibilities for both client and consultant personnel.

***Final SYSTEM Implementation Plan***

8

**Position**

**Responsibilities**

**Assigned Staff**

***CLIENT Staff***

**CLIENT SE /**

**Implementation Lead**

John Doe

**4.**

**PROJECT SCHEDULE**

The client SE / Implementation Lead, supported by the client DBA and appropriate consultant personnel (as outlined in *Section 3*), will implement the system within the client environment on Monday.

***Final SYSTEM Implementation Plan***

9

* Confirm that the installation was a success.

**CLIENT DBA**

The DBA will be supporting the client SE / Implementation Lead in the implementation of the database components of the system.

Jane Doe

***Consultant Staff***

**Consultant Developer**

Core developer for the system. Available as on-call support for the CLIENT SE / Implementation Lead and CLIENT DBA.

X

**Consultant DBA**

The DBA for the system development effort. Available as on-call support for the client SE / Implementation lead and client DBA.

Y

**Consultant Development Lead**

Lead developer for system. Available as on-call support for any miscellaneous issues surrounding implementation or patch-fixing for the client SE / Implementation Lead and client DBA.

Z