

2005

## An End-User Development Approach to Building Customizable Web-Based Document Workflow Management Systems

Stacy Hutchings  
*University of North Florida*

Follow this and additional works at: <https://digitalcommons.unf.edu/etd>



Part of the [OS and Networks Commons](#)

---

### Suggested Citation

Hutchings, Stacy, "An End-User Development Approach to Building Customizable Web-Based Document Workflow Management Systems" (2005). *UNF Graduate Theses and Dissertations*. 263.  
<https://digitalcommons.unf.edu/etd/263>

This Master's Thesis is brought to you for free and open access by the Student Scholarship at UNF Digital Commons. It has been accepted for inclusion in UNF Graduate Theses and Dissertations by an authorized administrator of UNF Digital Commons. For more information, please contact [Digital Projects](#).  
© 2005 All Rights Reserved

AN END-USER DEVELOPMENT APPROACH TO BUILDING  
CUSTOMIZABLE WEB-BASED DOCUMENT WORKFLOW MANAGEMENT SYSTEMS

by

Stacy Hutchings

A project submitted to the  
Department of Computer and Information Sciences  
in partial fulfillment of the requirements for the degree of

Master of Science in Computer and Information Sciences

UNIVERSITY OF NORTH FLORIDA  
DEPARTMENT OF COMPUTER AND INFORMATION SCIENCES

December, 2005

The project "An End-User Development Approach to Building Customizable Web-Based Document Workflow Management Systems" submitted by Stacy Hutchings in partial fulfillment of the requirements for the degree of Master of Science in Computer and Information Sciences has been

Approved by the project committee:

Date

**Signature Deleted**

\_\_\_\_\_  
Arturo Sánchez-Ruiz, Ph.D.  
Project Director

12/16/05

**Signature Deleted**

\_\_\_\_\_  
Judith L. Solano, Ph.D.  
Chairperson of the Department

12/16/05

**Signature Deleted**

\_\_\_\_\_  
Charles N. Winton, Ph.D.  
Graduate Director

12/16/05

## ACKNOWLEDGEMENT

I would like to thank Dr. Arturo Sánchez-Ruíz for his time and feedback during the implementation of this project. His encouragement and his belief in my work ethic and commitment helped me achieve this important milestone.

I also wish to thank my family and extended family, especially my husband "Hutch," my son Jimi, and my daughter Caitlyn, for their unwavering love and encouragement over the past four years. Without their support, this accomplishment would not have been possible. For the many sacrifices they have made, I am forever grateful.

## CONTENTS

List of Figures .....	vi
Abstract .....	ix
Chapter 1: Introduction .....	1
1.1 The Workflow Reference Model .....	2
1.2 Related Work .....	5
1.3 System Overview: Our Approach .....	6
Chapter 2: System Architecture .....	9
2.1 Technologies Used .....	9
2.2 Architecture .....	11
Chapter 3: Implementation .....	16
3.1 Security and Encryption .....	18
3.2 User Management .....	19
3.3 System Management .....	20
3.4 Definition Modes and Versions .....	21
3.5 Form Management .....	23
3.6 Workflow Management .....	24
3.6.1 States .....	24
3.6.2 Transitions .....	25
3.6.2.1 Assignment Strategies .....	26
3.6.2.2 Transition Strategies .....	27
3.7 Definition Management .....	29

3.7.1	Transition Rules.....	30
3.7.2	Form Rules.....	30
3.7.3	Rule Definition.....	31
3.7.3.1	The Required Validator.....	31
3.7.3.2	The Compare Validator.....	31
3.7.3.3	The Range Validator.....	32
3.8	Task Management.....	32
Chapter 4:	Conclusions.....	34
4.1	Summary.....	34
4.2	Future Enhancement Recommendations.....	35
References	.....	37
Appendix A:	Database Logical Model.....	40
Appendix B:	System Demonstration.....	41
Vita	.....	91

## FIGURES

Figure 1: The Workflow Reference Model .....	3
Figure 2: Architecture Overview .....	12
Figure 3: System Implementation .....	17
Figure A1: Database Logical Model .....	40
Figure B1: The Log In Screen .....	41
Figure B2: Register Account Screen .....	42
Figure B3: The Standard User View .....	43
Figure B4: The Workflow Engineer View .....	43
Figure B5: The Administrator View .....	43
Figure B6: The Manage User - User Tab .....	44
Figure B7: The Manage User - User Groups Tab .....	45
Figure B8: The Manage Groups - Group Tab .....	46
Figure B9: The Manage Groups - Account Members Tab .....	47
Figure B10: The Manage Groups - Group Members Tab .....	47
Figure B11: The Manage Systems - System Tab .....	48
Figure B12: The Manage Systems - System Processes Tab .....	49
Figure B13: The Manage Form Lists - Category Tab .....	50
Figure B14: The Manage Form Lists - Item Tab .....	51
Figure B15: The Design Form - Form Tab .....	52
Figure B16: The Design Form - Pages Tab .....	53
Figure B17: The Design Form - Sections Tab .....	54

Figure B18: The Design Form - Controls Tab .....	55
Figure B19: The Design Form - Results of View Tab .....	56
Figure B20: The Design Form - Results After Update .....	56
Figure B21: The Manage Roles - Role Tab .....	57
Figure B22: The Manage Roles - Account Members Tab .....	58
Figure B23: The Manage Roles - Group Members Tab .....	58
Figure B24: The Design Workflow - Workflow Tab .....	59
Figure B25: States Tab: Pending Employee Request .....	60
Figure B26: States Tab: Pending Manager Review .....	61
Figure B28: Transitions Tab: Transition One .....	62
Figure B29: Transitions Tab: Transition Two .....	63
Figure B30: Transitions Tab: Transition Three .....	64
Figure B31: The Design Workflow - From Strategies Tab ....	65
Figure B32: The Design Workflow - To Strategies Tab .....	65
Figure B33: The Design Form - Form Tab .....	66
Figure B34: The Design Workflow - Workflow Tab .....	66
Figure B35: The Design Definition - Definition Tab .....	67
Figure B36: The Design Definition - Queue Fields Tab .....	68
Figure B37: The Design Definition - Rules Tab .....	69
Figure B38: Rules Tab: T1 - Request Type Required .....	70
Figure B39: Rules Tab: T1 - Start Date Required .....	70
Figure B40: Rules Tab: T3 - Reason For Denial Required ...	71
Figure B41: The Design Definition - View Rules Tab .....	71
Figure B42: The Design Definition - Definition Tab .....	72

Figure B43: The Manage Systems - System Processes Tab . . . .	73
Figure B44: The Home Page - Select A System Menu . . . . .	73
Figure B45: The UNF System - Home Page . . . . .	74
Figure B46: The UNF System - View Workflow Queue . . . . .	75
Figure B47: The UNF System - View Workflow Queue . . . . .	76
Figure B48: The UNF System - Add New Instance . . . . .	77
Figure B49: The UNF System - J Hutchings' Pending Queue ..	78
Figure B50: View Instance Details - Summary Tab . . . . .	79
Figure B51: View Instance Details - Comments Tab . . . . .	80
Figure B52: View Instance Details - Transition Tab . . . . .	81
Figure B53: View Instance Details - Reassign Tab . . . . .	81
Figure B54: Pending Tasks for Stacy Hutchings . . . . .	82
Figure B55: View Instance Details - Comments Tab . . . . .	83
Figure B56: View Instance Details - Transition Tab . . . . .	83
Figure B57: View Workflow Queue - Closed . . . . .	84
Figure B58: View Instance Details - Comments Tab . . . . .	84
Figure B59: View Instance Details - Validation Error . . . . .	85
Figure B60: View Instance Details -View Only Mode . . . . .	86
Figure B61: Pending Tasks for Stacy Hutchings . . . . .	87
Figure B62: The UNF System - View Instance Details . . . . .	87
Figure B63: Transition Tab: Select Participant . . . . .	88
Figure B64: Pending Tasks for Caitlyn Hutchings . . . . .	89
Figure B65: View Instance Details - Transition Tab . . . . .	89
Figure B66: The UNF System - View Workflow Queue Closed ..	90

## ABSTRACT

As organizations seek to control their practices through Business Process Management (BPM - or the process of improving the efficiency and effectiveness of an organization through the automation of tasks), workflow management systems (WFMS) have emerged as fundamental supporting software tools. A WFMS must maintain process state while managing the utilization of people and applications (resources), data (context), and constraints (rules) associated with each of the tasks [Baeyens04]. It must also be configurable so it can be easily adapted to manage specific workflows within any application domain. Finally, the WFMS should be flexible enough to allow for changing business needs. In order to meet these challenges, a WFMS must provide access to process and document definition tools as well as administrative tools.

In this project we have used an "End User Development" (EUD) approach [Repenning04] to build a stand-alone web-based WFMS which offers the non-technical end user the ability to design, launch, and manage multiple automated

workflows and their associated documents. It empowers end users to build and customize their own systems without requiring from them skills other than those associated with their domain of expertise.

## Chapter 1

### INTRODUCTION

The benefits of automating workflow are often the selling points of a WFMS system. Documented on many commercial WFMS websites are the following: it can reduce the amount of paper, it can reduce the amount of administrative work, it can reduce errors, it can reduce cost, it can reduce time, and through process monitoring tools, it can prove its own efficiency. Numerous case studies support the finding that businesses gain many improvements through the use of automated workflows. This section begins with a discussion of the Workflow Reference Model, which delineates the essential components of a WFMS. A section that reviews related work follows, which reveals how our approach differs from others that have been documented in the current literature. Finally, the last section gives an overview of the system we built, highlighting its contributions.

## 1.1 The Workflow Reference Model

The Workflow Management Coalition (WfMC) was established in 1993 to address the "computerized facilitation or automation of a business process, in whole or in part" [WfMC]. A business process, manual or automated, can be defined as a series of tasks that are performed by a person, persons, or applications in order to achieve an overall goal. Rules accompany the process by governing such things as the order the tasks are performed, as well as the amount and type of information presented and exchanged at each step. Each task should be defined in enough detail to determine the *who*, the *what*, the *when*, and the *how* of the process detail. It is important to know at each step, *who* is responsible for the task, *what* information needs to be collected, and *when* the task should occur. In addition, privileges and constraints control *how* it should occur (this includes determining the next step(s) as defined by the network of tasks).

In 1995, continuing their efforts to provide common ground in the field of workflow, the WfMC published the Workflow Reference Model [WfMC-TC-1003] depicted in Figure 1 to illustrate the major components and interfaces of a generic WFMS architecture. The model suggests five interfaces, or functionalities, a WFMS should support:

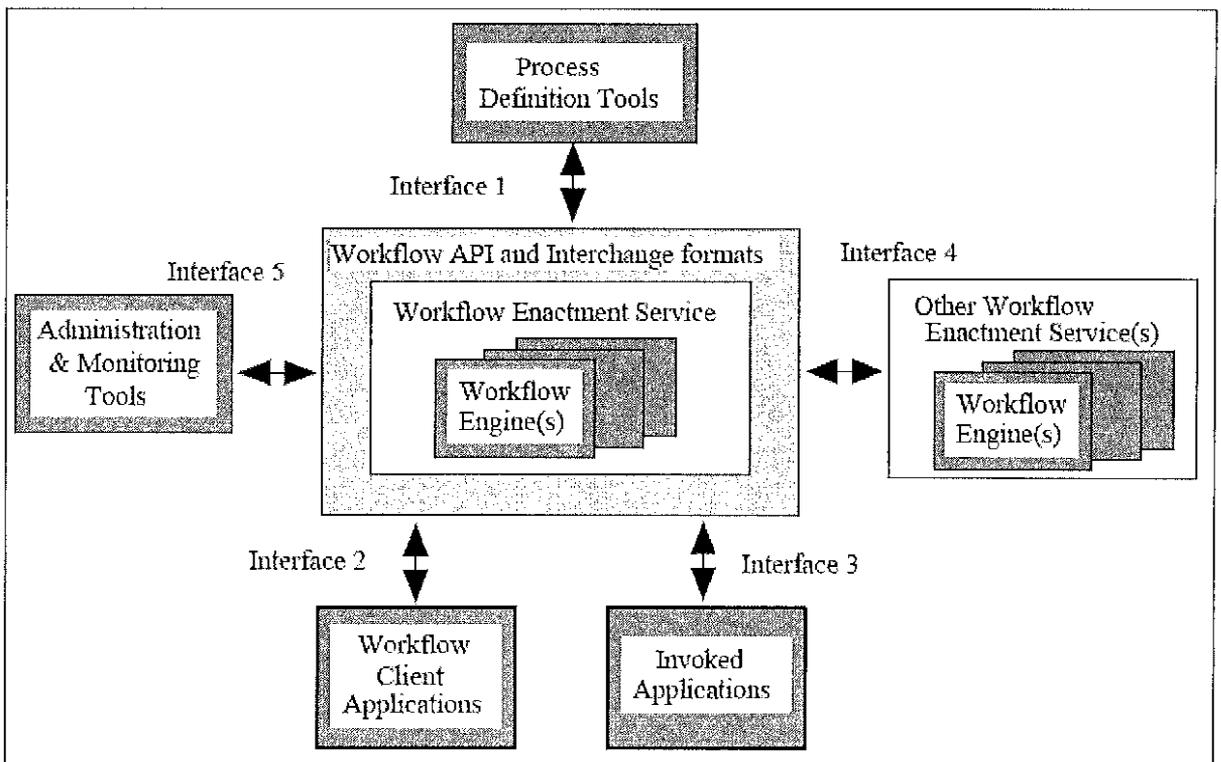


Figure 1: The Workflow Reference Model

- Interface 1 is the process definition interface: to define, modify, or import a process definition that will be interpreted by the workflow enactment service.

- Interface 2 is the workflow client interface: to present pending tasks to workflow participants for viewing or processing purposes.
- Interface 3 is the invoked application interface: to interact with other applications, such as the ability to invoke a messaging service to notify participants of their tasks.
- Interface 4 is the workflow enactment service interface: to interact with other workflow systems.
- Interface 5 is the administration tool and monitoring interface: to support process monitoring and management utilities for operating the entire WFMS system.

The five interfaces surround the core of the WFMS, a workflow enactment service comprised of one or more workflow engines. The responsibility of the workflow enactment service is to coordinate actions among the workflow engines that interpret the process definition and activate external resources.

## 1.2 Related Work

There are many commercial WFMS products available and the list of open source products is growing [e.g., Perez, Topicus, Baeyens04]. Very few claim to be complete WFMS, but rather workflow engines. In the review of seven open source Java-based workflow products [EnhydraShark, jBPM, JFolder, OpenWFE, wfmOpen, XFlow, YAWL], not one offered the end user a complete WFMS solution, implemented with the recommended interfaces of the Workflow Reference Model and without the need for technical knowledge or systems development skills. Instead, most adopted a minimalist view of the scope of a WFMS by providing a framework that leaves much of the implementation details to the users of the system. It is worth mentioning that there are two levels of "users" of WFMS, namely developers, who must customize WFMS frameworks to generate actual workflow applications, and end users, who will use the system as part of a BPM approach within an organization. The motivation and focus of this project has been on the needs of the latter, by empowering them with the ability to build their own workflow systems without the intervention of developers (the former).

### 1.3 System Overview: Our Approach

The goal of this project was to develop a customizable web-based document WFMS. Web-based document management technology is concerned with managing the lifecycle of documents, which are represented in our system through the use of web forms. The workflow enactment service is responsible for routing all or part of a form among individuals for approval, processing, or collaboration. Rules govern which parts of a form are required by a particular role. In addition, complex or simple workflow routing decisions are made based on form data.

Our system differs from other open source workflow products in a number of ways. First, it was designed to specifically address document management workflow technology. Web forms provide the means for collecting data from the participant. The workflow enactment service is responsible for displaying all or part of a form as well as routing the form among participants. In contrast, the reviewed open source workflow projects do not address specific workflow system technology.

Second, our system fully implements three of the prescribed WfMC's Workflow Reference Model interfaces through a centralized, web-based application interface. It presents the end user with different functionality depending on one of three user types: administrator, workflow engineer, and standard workflow participant. The standard workflow participant interface addresses Interface 3 of the Workflow Reference Model. It provides the ability to manage workflow tasks. The workflow engineer interface addresses Interface 1 of the Workflow Reference Model. It provides the ability to define a form, a workflow, and a process definition. The administrator interface addresses Interface 5 of the Workflow Reference Model. It provides the ability to monitor workflow progress and to perform such tasks as user and system management.

Finally, our system uses a different approach to defining and evaluating workflow rules. All reviewed open source workflow systems define routing decisions within the process definition itself. The problem with this approach is it limits the reusability of the same definition among other workflows. Our project implements a new process definition approach. It uses three separate

definitions to make up the entire workflow: a form definition, a workflow definition, and a process definition. Form and workflow definitions are created independent of one another. Once created, the process definition bridges a unique form with a unique workflow. Rules are defined within the process definition. This approach encourages the reusability of individual forms and workflows. In addition, it extends the rule definition possibilities to include form validation, as well as routing decisions.

## Chapter 2

### SYSTEM ARCHITECTURE

#### 2.1 Technologies Used

The system we constructed, the Workflow Generator and Tracking System, is a web-based system developed using the Eclipse IDE [Eclipse]. The Java 2 Enterprise Edition (J2EE) provides the programming back-end for the web browser-based user interface. The J2EE environment is provided through a Jakarta Tomcat 5.5 (version 5.5.9) [Tomcat] application server. Tomcat 5.5 requires Java 2 Standard Edition Runtime Environment (JRE) version 5.0 or later.

Persistent data is stored in a MySQL 5.0 (version 5.0.11-beta) [MySQL] InnoDB database. The data is accessed from the J2EE program via the Connector/J JDBC (version 3.1.10) [MySQL Connector/J] connection interface. Both the iBATIS Data Mapper framework (version 2.0) [iBATIS] and the iBATIS Data Access framework (version 2.0) were used to significantly reduce the amount of Java code needed to access the database. Using eXtensible Mark up Language

(XML) descriptor files, the iBATIS software mapped Java Beans to SQL statements as well as provided a data access abstraction layer.

Apache Software Foundation's Struts framework with Tiles (version 1.1) [Struts] was used to provide the view and controller components of the architecture. The overall system structure is an implementation of the Model-View-Controller (MVC) software architecture. With this architecture, the interface (view) is loosely coupled to the business logic (controller) and the business logic is loosely coupled to the data (model). The Struts framework provides a controller component for web-based applications that use a J2EE compliant application server. Tiles are view components that allow the interface to be assembled "in pieces." The framework uses an XML configuration file to organize, lay out, and reuse the individual components or tiles. Tiles for headers, footers, and navigation bars were placed in the same location on all pages to provide a consistent display format

The user interface is implemented using Java Server Pages (JSP) (version 2.0), part of the J2EE platform, and Cascading Style Sheets (CSS). The Struts framework

provides several JSP tag libraries to simplify access to controller and data objects from within JSP. In addition, the project also utilizes the Java Standard Tag Library (JSTL) [JSTL], as well as the Struts Layout Tag Library [StrutsLayout]. Javascript is used sparingly throughout the project. CSS was implemented to give end users the ability to customize the look and feel of their system without having to modify the JSP. The project's main logo as well as the CSS table-less design were adopted for use from the tutorials presented on the MaxDesign website [MaxDesign].

## 2.2 Architecture

At the highest level, the tracking system implements the MVC design pattern. The view layer is the user interface. The view never directly interacts with the model, which is the database. Data that is changed or events that occur in the view component are sent to the appropriate controller component. The controller determines what action should be taken for the event. The controller, through defined action form classes, passes control to the business layer. The business layer calls methods on the Data Access Object (DAO) interface in order to pass and receive data from the

model component. The implementation of this architecture can be seen in Figure 2.

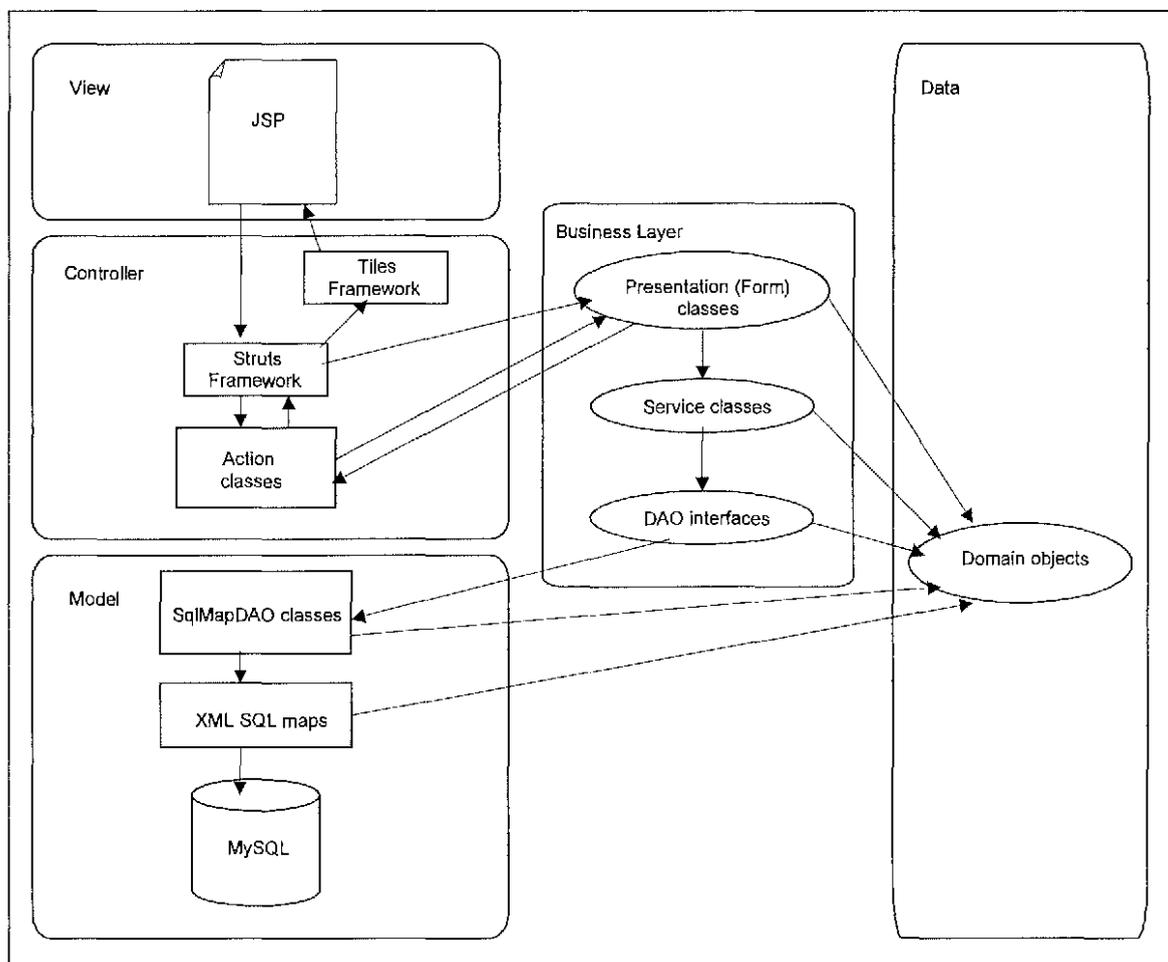


Figure 2: Architecture Overview

As a brief description of the Struts framework, the framework relies on one or more XML configuration files that associate three main components: Action classes, Form classes, and JSP or Tile names. When a JSP document is requested from the server, it is requested via an action

call. The action call is defined in the configuration file along with a fully qualified action class name and a fully qualified action form class name. The controller executes the appropriate action method then instantiates the associated action form object. In addition, it is the controller's responsibility to determine the appropriate page to display before passing the request to the server.

The server is responsible for compiling the embedded Javascript and JSP tags that determine which elements are displayed. The document is then rendered as an HTML page containing data fields on a form and presented to the user. When the form is submitted, another action is called. The controller populates the instantiated action form object with the appropriate form values using JavaBean property methods. The action form can then interact with the business layer as designed.

The business layer design for this project was modeled after the iBATIS JPetStore 4.0.5 [iBATIS] example. The domain classes are the entities of the workflow application. Each domain class represents a JavaBean. Each domain class contains only variables that define its properties as well as "getter" and "setter" methods. The

domain objects are utilized by both the model and business layer components.

Each presentation class extends the Struts ActionForm class. The controller uses the presentation class to retrieve, populate, and present data in the view component of the architecture. Initially, the action class calls a method in the presentation class to handle the business logic. The presentation class, in turn, interacts with the service class each time access to the model layer is requested. Finally, the service class utilizes the DAO interfaces to extract information from the database.

Access to the database has been implemented using the Data Access Object pattern [Gamma95] with iBATIS' Data Access Objects API and SQL Data Mapper framework. The DAO API provides for the dynamic configuration of various persistence mechanisms. Since the service classes interact only with the DAO interface, the actual implementation details are hidden from the rest of the application. This project uses the iBATIS SQL maps as its persistence mechanism. The SQL map DAO class methods interact directly with the SQL Data Mapper framework.

The SQL Data Mapper framework uses one or more XML configuration files to map domain objects to SQL tables or query results. For this project, multiple configuration files were used to represent either a single table in the database or a collection of related tables. Each configuration file contains an XML mapping of table column names to domain object property names, as well as the defined SQL queries for the appropriate tables. A separate XML file holds the database connection properties that also support the configuration of various Database Management Systems (DBMS).

Whenever the service class calls a method on the DAO interface, the implemented SQL Map DAO passes the name of the query and any parameters to the framework. The Data Mapper framework does the rest. It performs the necessary database connection operations (open, close, as well as pooling), executes the query, and returns a populated domain object as a result

## Chapter 3

### IMPLEMENTATION

The Workflow Generator and Tracking System was built to align with the Workflow Reference Model, as Figure 3 shows. It fully implements three of the five recommended interfaces. Interface 5 provides the administration and management tools of the application. In order to demonstrate interactions with organizational or resource data, this application provides user management and system management capabilities to those with Administrator level access. Interface 1 provides the workflow definition tools of the application. Workflow Engineers have the ability to create form definitions, workflow definitions, and process definitions. In addition, they are responsible for creating roles for use in the creation of a workflow. Interface 2 provides task management functionality for users of the workflow system. The Workflow Generator and Tracking System produces each workflow so it may be accessed as its own, unique web-based system. Within the customized system, users in the appropriate roles have the

ability to create a new instance of a workflow, view an instance of a workflow, modify an instance of a workflow, and transition an instance of a workflow.

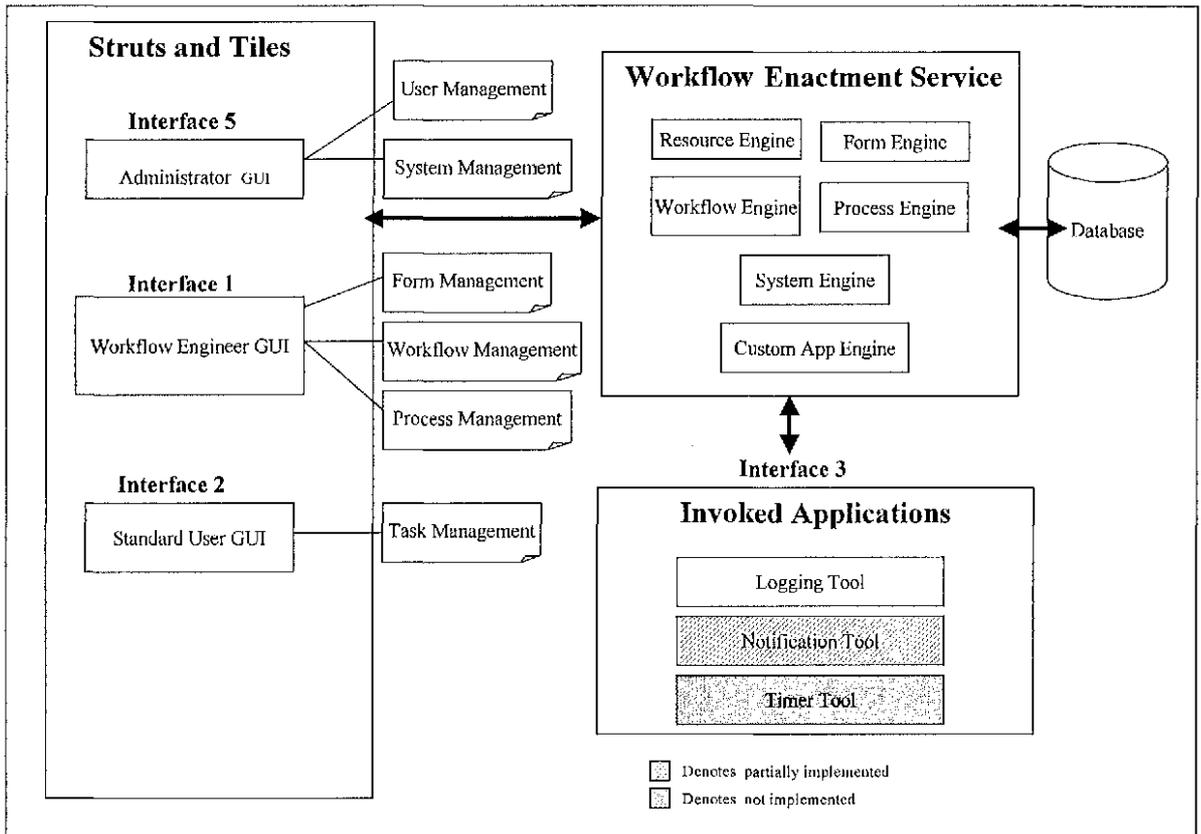


Figure 3: System Implementation

The workflow enactment service is made up of six engines that handle the various requests of the system. The Resource Engine handles all user related tasks. This includes such things as logging in and out of the system, as well as creating and modifying user accounts. The Form Engine handles all tasks related to the creation of a form.

The Workflow Engine handles all tasks related to the creation of a workflow. The Process Engine handles all tasks related to the creation of a process definition. The System Engine handles all tasks related to the creation of a new system. The Custom App Engine handles all tasks associated with individual workflow task management, such as viewing, transitioning, and assigning a workflow task.

The database contains forty tables that are used to store information regarding users, forms, workflows, process, rules, systems, and workflow instances. The design supports the creation of multiple workflows independent of application domain and without the need for additional tables, by extracting the common elements shared among instances and storing all form values as text in a single table. Appendix A represents a logical diagram of the database.

### 3.1 Security and Encryption

The Workflow Generator and Tracking System provides three access levels: administrator, workflow engineer, and standard user. Those with standard user privileges, upon log in, are restricted to accessing published systems and

their associated workflows only. Users with workflow engineer privileges are granted access to the form management, workflow management, and definition management tools. Users with administrative access have all privileges plus access to system management and user management tools. A single, super user was also created for the system.

All users have the ability to edit their personal account information including changing their password. Each individual is responsible for selecting their security question and answer at their initial log in. This information is maintained for use in the event a person forgets her password. The security information is private and is not accessible to the administrator at anytime. All system passwords and security answers are encrypted in the database using the MD5 encryption algorithm.

### 3.2 User Management

The User Management option is available only to Administrators. In this component, Administrators have the ability to create, edit, activate, and de-activate users of

the Workflow Generator and Tracking System. The information stored for each user includes first name, last name, phone number, email, username, password, password confirmation, access level, and active status.

Within this component, Administrators also have the ability to create new and edit existing user groups.

Administrators can add individual users or existing groups as members of another group. They also have the ability to remove members from a group. The application provides two system-defined groups: the All User group and the System Level - Admin group. As a user is created, she is added to the All User group. The All User group provides a convenient way to access all users of the system. An Administrator can also grant administrative access at the individual system level by placing a user or group in the System Level - Admin group.

### 3.3 System Management

The System Management component gives the Administrator the ability to create, edit, activate, and de-activate custom systems that operate "outside of" the Workflow Generator and Tracking System. When defining a new system, the

Administrator creates a website name and chooses a style sheet for a custom "look and feel." Once created, the new system appears as a selection under the Select System menu option. The Administrator can then associate one or more published process definitions with the newly created system. Any user, including those with Standard User permissions, can link to and begin using the newly created web-based system for task management.

### 3.4 Definition Modes and Versions

The Workflow Generator and Tracking System houses three definitions that together comprise a workflow: a form definition, a workflow definition, and a process definition. Each definition has been designed to support three modes: design mode, test mode, and published mode. In design mode, anything is possible. The Workflow Engineer can add, modify, or delete any part of a definition. In test mode, the definition cannot be modified except to be placed back into design mode. In published mode, some modifications are permitted, depending on the type of definition; however, a published definition can never return to any of its previous states. It will remain permanently in published mode.

When a Workflow Engineer initiates the creation of a definition, it begins in design mode. Definitions in design mode are not visible to any other component. When a form or workflow definition is ready for testing, the engineer places the definition into test mode. In test mode, it becomes visible to the process definition component. The Workflow Engineer can now assemble the process definition comprised of a single form and a single workflow.

When the process definition is ready for testing, the engineer would normally place it in test mode. Due to time constraints, however, the test feature was not implemented. At this point, the engineer must by pass the test phase and publish the definition. When the engineer places the process definition into published mode, the associated form and workflow are published as well.

The system was also designed to support versioning of each type of definition. The idea was to allow new versions of a published definition to be created and published along side of an already existing version. This can be demonstrated, however, the GUI tools for creating

additional versions were unable to be built, again due to time constraints.

### 3.5 Form Management

The Form Management component provides a Workflow Engineer with the tools to create forms. An Engineer can create, edit, and delete any form in design mode. Published forms may be modified; however, editing capabilities are limited. Forms are built incrementally using the designer interface.

Forms support multiple pages, sections, and controls.

There are eleven choices for form controls: Checkbox, Currency, Date, Double, Dropdown, Email, Float, Integer, Radio, Text, and Text area. The controls generate HTML form objects and provide built in validation. For example, by selecting the Email control, the form engine will generate a text box for data collection and insure that any data entered passes email validation. If Integer control were selected, the form engine would generate a similar text box for data collection. However, it would insure the information collected passes integer validation instead. Workflow Engineers also have the ability to customize their own lists for Dropdown, Radio, and Checkbox controls. In

addition, they have the ability to add labels, tool tips, default values, as well as size and order their controls on the form.

### 3.6 Workflow Management

The Workflow Management component provides a Workflow Engineer with the tools to create workflows. An Engineer can create, edit, and delete any workflow in design mode. Published workflows may be modified; however, editing capabilities are limited. Workflows are built incrementally using the designer interface. Workflows are comprised of three main parts: states, transitions, and workflow pattern strategies [van der Aalst].

#### 3.6.1 States

A state represents a transitioning point in a workflow. Each state is dependent upon a participant responsible for signaling the end of a task. This system supports multiple states, one of which must be indicated as the initial state.

Each state is associated with a given role. Roles are comprised of one or more individuals or groups in an organization. At each state, the Workflow Engineer determines whether or not a given role should have the permission to cancel a workflow instance, close a workflow instance, or reassign a workflow instance to another participant within the same role. In addition to providing the tools to manage states, this system provides the Workflow Engineer with the ability to create and manage roles.

### 3.6.2 Transitions

A transition occurs when a participant has signaled the completion of a task. The workflow moves from one state to the next state according to the defined path. The Workflow Engineer defines a transition by indicating a "from state" and a "to state." There can be multiple transitions originating from a single state, as well as multiple transitions converging on a given state. The next two sections discuss assignment and transition strategies.

### 3.6.2.1 Assignment Strategies

This system implements three assignment strategies: Pull, Push, and Auto-Lookup. The Pull strategy [Muehlen04] routes tasks to a general queue without requiring the name of a specific individual. The general queue holds all unassigned tasks until a role member is able to assign the task to herself. The tasks are available only to individuals who are part of the associated role. Once checked out, the workflow instance for that state becomes the responsibility of the individual user. Once assigned, the task is no longer available to other role members.

The Push strategy [Muehlen04] requires that the transitioning role member manually assign the instance to a role member in the next state. Upon transition, the system presents the current participant a list of individuals to choose from. Once selected, the system transitions the instance to the next state and the task appears in the queue of the selected participant.

The Auto-Lookup strategy is not based on any research. It was included to demonstrate a third assignment strategy. The Auto-Lookup strategy automatically assigns the instance

to the next role member, if the list of participants is made up of only a single member or if the state has been visited previously by the current instance.

#### 3.6.2.2 Transition Strategies

In addition to the assignment strategies, there exist six recognized transition strategies: Deferred Choice, And Split, Or Split, Sequence, Merge-First Wins, and Merge-Last Wins. All of the implemented strategies are based on published workflow pattern research [van der Aalst].

The Sequence strategy is implemented by the system any time there is a single transition originating from a given state. Since there is only one decision path for the given state, the system automatically transitions to the next state.

The And Split and Or Split strategies are necessary when there are multiple transitions originating from a given state. If multiple transitions exist, the system presents the Workflow Engineer with a choice of using the Or Split strategy or the And Split strategy. If the And Split strategy is selected, no further rules are required. At

the time of transition, the workflow instance will automatically transition to all states included as part of the And Split strategy. If the Or Split strategy is selected, the Workflow Engineer must define at least one transition rule for each transition included as the Or Split strategy. Rule definition occurs at the next stage, during the creation of the process definition.

The Merge-First Wins and the Merge-Last Wins strategies are necessary when there are multiple transitions converging on a given state. If multiple transitions exist, the system presents the user with a choice of using the Sequence strategy, the Merge-First Wins strategy, or the Merge-Last Wins strategy. The engineer can decide which, if any, transitions should participate in a merge. The Merge-Last Wins strategy prevents the workflow instance from transitioning to the next node until all states participating in the merge have indicated completion of their task. At the time the final merge participant signals completion, the instance transitions to the next state. The opposite is true of the Merge-First Wins strategy. Once the very first merge participant indicates completion of their task, the instance transition to the next state. The remaining participants are still required

to complete their task; however, the instance closes silently without transitioning to the next state.

The Deferred Choice strategy is the only strategy that presents a list of transition choices to the user, prior to her decision to transition to the next state. All other transition strategies are calculated by the system after the user signals the completion of the task. In other words, for these strategies, transition choices are not visible to the user. To implement a deferred choice strategy, the Workflow Engineer must select yes to the deferred choice option on the transition tab.

### 3.7 Definition Management

The Definition Management component provides a Workflow Engineer with the tools to create a process definition. An Engineer can create, edit, and delete any process definition in design mode. Published process definitions may be modified; however, editing capabilities are limited. Process definitions are comprised of a single form and a single workflow. Once the form and the workflow have been selected, the Workflow Engineer can create the rules that govern each. The Definition Management component gives the

engineer the ability to create, edit, or delete form as well as transition rules. All rules are based on information contained in the form.

### 3.7.1 Transition Rules

Transition rules are required only when an Or Split transition strategy has been indicated for the given workflow. The creator must define at least one rule for each transition involved in the Or Split group. If the rule evaluates to true upon transition, the transition path will be selected. The rules may be defined to be exclusive or multiple paths may evaluate to true. The implementation is in the hands of the Workflow Engineer.

### 3.7.2 Form Rules

Form rules are optional. As described within the form management section, the system provides for basic validation according to control type. For example, a date control will check for valid date entries. However, if a Workflow Engineer would like to make a particular form field required, she must define an additional form rule. In addition, the Workflow Engineer has the option of

selecting whether a form rule should be evaluated on every transition or only on a given transition.

### 3.7.3 Rule Definition

The methods for defining form rules and transition rules are exactly the same. This system's rule implementation approach is modeled after the ASP.NET Validation Server Controls [MSDN]. The Workflow Engineer can choose to implement the Required Validator, the Compare Validator, or the Range Validator.

#### 3.7.3.1 The Required Validator

The Required Validator is the simplest of all validation types. It checks to make sure the form field has been populated. If validation fails, a custom error message, as defined by the engineer, will be displayed to the user.

#### 3.7.3.2 The Compare Validator

The Compare Validator is used when a given form control needs to be compared to another form control or value,

based on the following operator types: Equal, Not Equal, Greater Than, Greater Than Equal, Less Than, and Less Than Equal. If the given form control is to be compared to another form control, the system presents the Workflow Engineer with a list of controls matching the given control type. On the other hand, if the given form is to be compared to a value, the Workflow Engineer must enter a value appropriate for the given control type.

#### 3.7.3.3 The Range Validator

The Range Validator is used when the given form control must be validated according to a lower bound and an upper bound of values. The Workflow Engineer must select both a minimum and maximum value appropriate for the given control type. For both the Compare Validator and the Range Validator, if validation fails, a custom error message is presented to the user.

### 3.8 Task Management

The Workflow Generator and Tracking System uses the information collected by the Workflow Engineer and the Administrator to produce the Task Management interface as a

separate, functioning web system. The component itself may house one or more process definitions. For each definition, users with Standard User permission have the ability to select a workflow, create a new instance of a workflow, modify an existing instance of a workflow, view the history and comments associated with a workflow, as well as the ability to assign and transition a workflow. Tasks are initialized using the Add New Instance option. Instances of each document are created and managed according to the form, workflow, and process definitions. Tasks remain open until a user successfully closes or cancels the instance.

## Chapter 4

### CONCLUSIONS

#### 4.1 Summary

In summary, after having researched various approaches to building this class of systems, it can be said that while the scope of a workflow varies among products, all seem to share in common the ability to define a process and a set of classes that can interpret the generated definitions. Some provide only the workflow engine API. Others offer more of a framework, along the lines of the WfMC Reference Model. In either situation, the end user is dependent upon the system developer to produce a functioning WFMS. Our observation was that current open source workflow products seemed to exist as tools for the system developer interested in workflow, rather than as a solution for the end user in need of a WFMS.

The goal of this project was to use an EUD approach to produce a complete, out-of-the-box WFMS solution for web-based document management. Its main purpose was to give the business-oriented end user the ability to automate

workflow without the need to program. The Workflow Generator and Tracking System accomplishes this goal. It addresses specifically three types of users: administrators, workflow engineers, and standard workflow participants, by providing the appropriate tools for monitoring, developing, and coordinating automated workflow tasks. In addition, it incorporates information (the form), as well as organization (the participants) within the overall process (workflow) design to produce a *complete* WFMS [Hollingsworth04].

#### 4.2 Future Enhancement Recommendations

The author acknowledges the scope of this project was greater than initially anticipated. The overall size of the effort did not allow for the implementation of every feature originally envisioned or specified in the proposal. In particular, desirable features that were not implemented include a scheduler of tasks, a method to recall transitioned tasks, the end user tools to create definition versions, the ability to operate in test mode, and the full implementation of notification strategies such as email.

Although the list above appears lengthy, the items on it were close to being within reach. For example, the current system design allows for versioning support of all definition types. However, the GUI tools that would have allowed the end user to create them have not been developed.

Recommendations for future enhancements include incorporating more transition and assignment options, enhancing rule functionality, incorporating recall options, implementing messaging services, implementing visibility and edit constraints on form fields according to role, and implementing a task scheduler.

## REFERENCES

[Baeyens04]

Baeyens, T., "The State of Workflow," (May, 2004).

<http://www.theserverside.com/articles/article.tss?l=Workflow>

[Eclipse]

<http://www.eclipse.org/>

[EnhydraShark]

<http://shark.objectweb.org/>

[Gamma95]

Gamma, E., et al., Design Patterns Elements of Reusable Object-Oriented Software, Addison-Wesley Professional Computing Series, 1995.

[Hollingsworth04]

Hollingsworth, D., "Workflow Reference Model:10 Years On,"  
A chapter from the Workflow Handbook 2004, Fujitsu  
Services, United Kingdom, 2004.

[iBATIS]

<http://ibatis.apache.org/>

[jBPM]

<http://www.jboss.com/products/jbpm>

[JFolder]

<http://www.powerfolder.org/>

[JSTL]

<http://jakarta.apache.org/>

[MaxDesign]

<http://www.maxdesign.com.au/>

[MSDN]

<http://msdn.microsoft.com/>

(REFERENCES, Cont'd)

[Muehlen04]

Muehlen, M.Z., "Organizational Management in Workflow Applications - Issues and Perspectives," Information Technology and Management Journal, Kluwer Academic Publishers, 5, 3 (2004), pp. 271-291.  
[http://www.workflow-research.de/Publications/PDF/MIZU-ITM\(2004\).pdf](http://www.workflow-research.de/Publications/PDF/MIZU-ITM(2004).pdf)

[MySQL]

<http://www.mysql.com/>

[MySQL Connector/J]

<http://dev.mysql.com/downloads/connector/j/3.1.html>

[OpenWFE]

<http://web.openwfe.org/>

[Perez]

Perez, C, <http://www.manageability.org/>

[Repenning04]

Repenning, A. and A. Loannidou, "End-user development: tools that empower users to create their own software solutions," Communications of the ACM - Special Issue, 47, 9 (September, 2004) pp. 43-46.

[Struts]

<http://struts.apache.org/>

[StrutsLayout]

<http://struts.application-servers.com/>

[Tomcat]

<http://tomcat.apache.org/>

[Topicus]

<http://www.topicus.nl/topwfm/>

[van der Aalst]

<http://is.tm.tue.nl/research/patterns/patterns.htm>

[WfMC]

<http://www.wfmc.org>

(REFERENCES, Cont'd)

[WFMC-TC-1003]  
Workflow Management Coalition, The Workflow Reference  
Model, 1995.

[wfmOpen]  
<http://wfmopen.sourceforge.net/>

[XFlow]  
<http://xflow.sourceforge.net/>

[YAWL]  
<http://www.yawl.fit.qut.edu.au/>

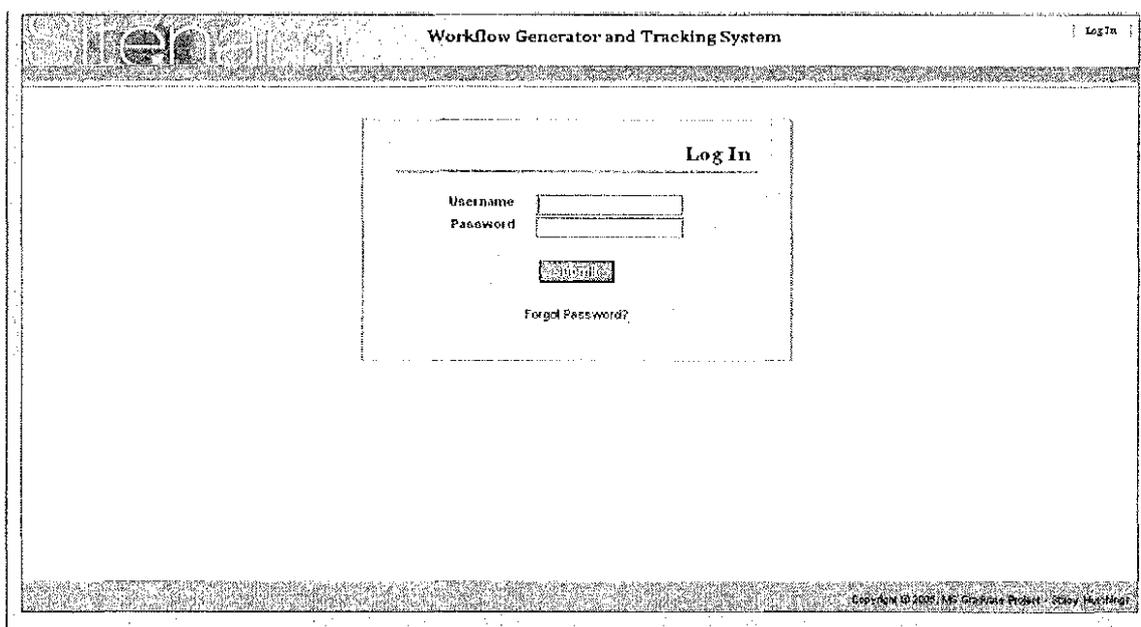


## APPENDIX B

### SYSTEM DEMONSTRATION REQUEST A DAY OFF WORKFLOW EXAMPLE

#### Log In

When a user accesses the website, the system presents the user with a log in screen shown in Figure B1. At initial log in, the user must enter the username and password assigned to her by the system administrator.



The screenshot shows a web browser window with the title "Workflow Generator and Tracking System" and a "Log In" link in the top right corner. The main content area contains a "Log In" form with the following elements:

- A "Log In" title above the form fields.
- A "Username" label followed by a text input field.
- A "Password" label followed by a password input field.
- A "Log In" button.
- A "Forgot Password?" link below the button.

At the bottom right of the browser window, there is a small copyright notice: "Copyright © 2007, MS Graduate Project - Steve Hutchings".

Figure B1: The Log In Screen

## Register Account

When a user accesses the system for the first time, she is presented with the screen shown in Figure B2 that displays her information. The system requires the user update any incorrect information; modify and confirm her password; and select a security question and answer. The security question and answer are stored in the event the user forgets her password. When the user has completed her changes, she selects the register account option to return to the log in screen.

The screenshot shows a web browser window with the title 'Workflow Generator and Tracking System' and a 'login' link in the top right corner. The main content area is titled 'Register Account' and contains the following form fields:

- First Name: Stacy
- Last Name: Hutchings
- Phone: [Empty]
- Email: hutch002@unf.edu
- Username: hutch002
- Password: [Masked with asterisks]
- Confirm Password: [Masked with asterisks]
- Security Question: No security question selected (dropdown menu)
- Security Answer: [Empty]

At the bottom of the form is a button labeled 'REGISTER ACCOUNT'. In the bottom right corner of the browser window, there is a copyright notice: 'Copyright © 2005, The University of North Florida - Stacy Hutchings'.

Figure B2: Register Account Screen

## Access Levels

The Figures B3, B4, and B5 show each of the navigational views according to user access level.

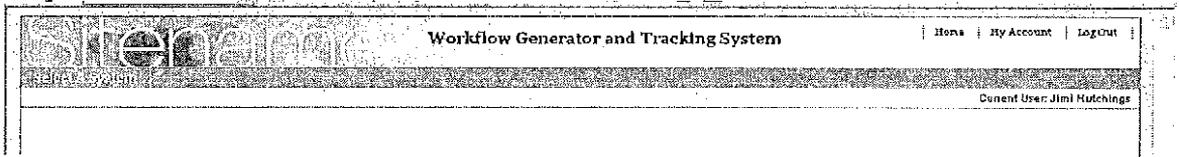


Figure B3: The Standard User View

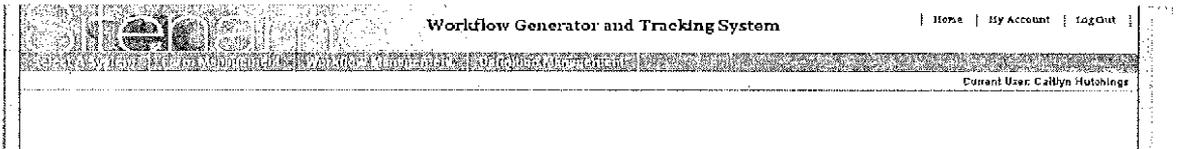


Figure B4: The Workflow Engineer View

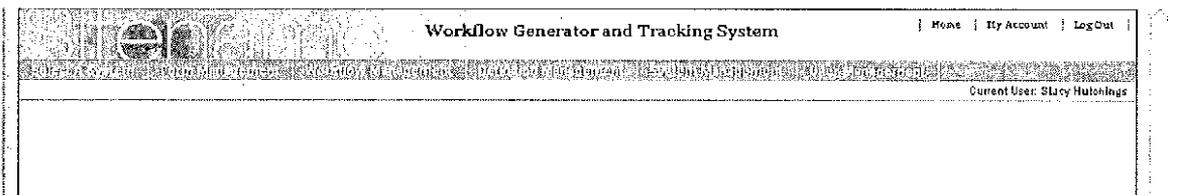


Figure B5: The Administrator View

## User Management

The User Management menu gives the Administrator two options: Manage Users and Manage Groups.

### Manage Users

The User tab shown in Figure B6 is the tool used to add or edit users. When adding a new user, the administrator populates all of the fields and chooses one of the three provided access levels. She also has the ability to edit any existing user by selecting her from the drop down list of active users. The Administrator can de-activate any existing user by un-selecting the Active checkbox.

The screenshot shows a web browser window titled "Workflow Generator and Tracking System". The browser's address bar contains the URL "http://www.localintranet.com/WorkflowGenerator/Default.aspx?tabid=1". The page header includes "Home", "My Account", and "Log Out" links, and a "Current User: Sissy Hutchings" indicator. The main content area is titled "Create or Modify A User" and contains the following form elements:

- A dropdown menu labeled "Please select a user" with a downward arrow.
- An "Active" checkbox with a downward arrow.
- A "User:" label followed by a dropdown menu showing "User: Sissy Hutchings".
- Input fields for "First Name:", "Last Name:", "Phone:", and "Email:", each with a corresponding text box.
- Input fields for "Username:", "Password:", and "Confirm Password:", each with a corresponding text box.
- An "Access Level:" label followed by a dropdown menu showing "Standard User".
- An "Active:" checkbox with a checked mark.
- A "Last Accessed:" label followed by an empty text box.
- Two buttons at the bottom: "ADD NEW USER" and "UPDATE USER".

The browser's status bar at the bottom shows "Done" on the left and "Local Intranet" on the right.

Figure B6: The Manage User - User Tab

(User Management, Cont'd)

The User Groups tab is shown in Figure B7. This screen displays the current user groups for the selected user, as well as allows the Administrator the ability to add and remove groups as needed. Each new user added to the system is automatically placed in the All Users group.

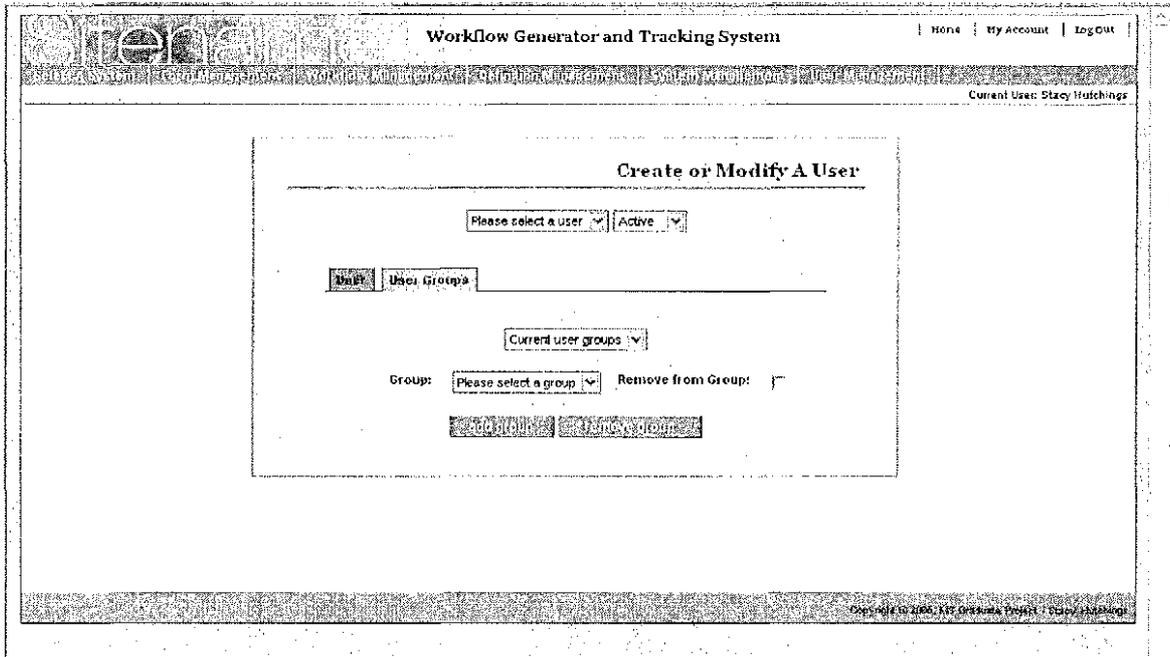


Figure B7: The Manage User - User Groups Tab

(User Management, Cont'd)

## Manage Groups

The Group tab is shown in Figure B8. Administrator adds and modifies groups within this interface. Administrators enter a group name and optional description to add a new group. To edit the group, the Administrator would select the group from the list provided, make changes, then update.

The screenshot shows a web application interface titled "Workflow Generator and Tracking System". At the top right, there are links for "Home", "My Account", and "Logout". Below these links, it says "Current User: Stacy Hutchings". The main content area is titled "Create or Modify A Group". At the top of this section is a dropdown menu with the text "Please select a group (V)". Below this are three tabs: "Group", "Account Members", and "Group Members". The "Group" tab is currently selected. Underneath the tabs, there are several input fields: "Group:" with a text box, "Description:" with a larger text area, "Created By:" with a text box, and "Modified By:" with a text box. At the bottom of the form area, there are two buttons: "Save" and "Cancel".

Figure B8: The Manage Groups - Group Tab

(User Management, Cont'd)

The Account Members tab, Figure B9, and the Group Members tab, Figure B10, allow the Administrator to add individual and group members by selecting a member, then selecting the add button. They may also remove members from the group by selecting the current member of choice, checking the Remove Member checkbox, then selecting the update button.

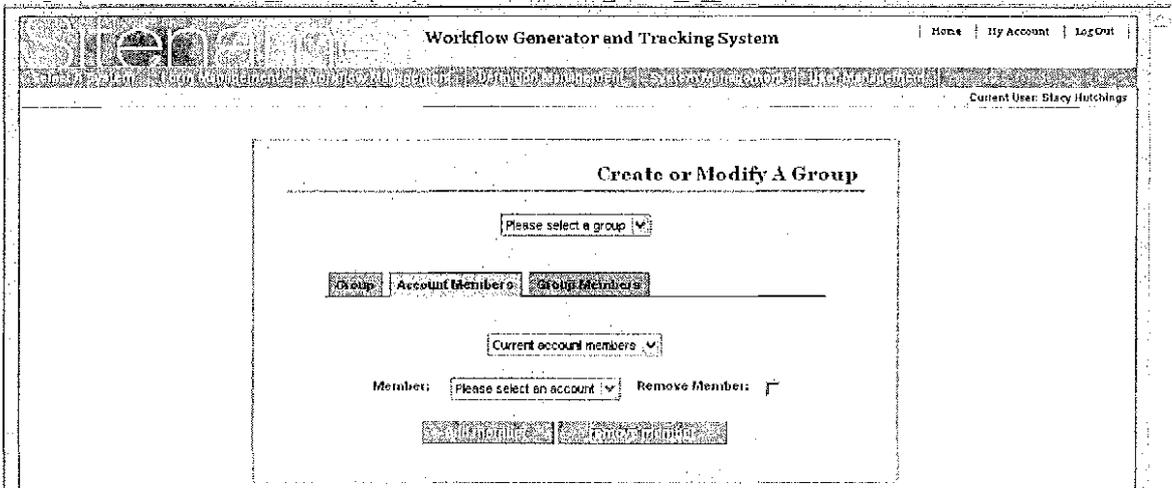


Figure B9: The Manage Groups - Account Members Tab

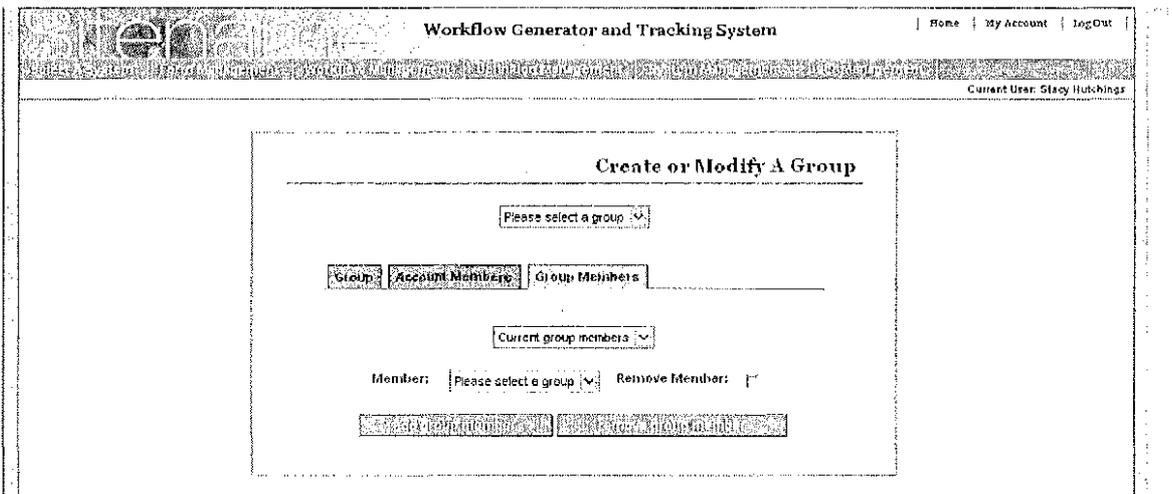


Figure B10: The Manage Groups - Group Members Tab

## System Management

The System Management menu gives the Administrator one option: Manage Systems.

### Manage Systems

The System tab shown in Figure B11 is the tool used to add or edit systems. Each system created will appear in the Select A System menu on the navigational bar for all users. The Task Management component is enacted within the custom generated system. To add a new system the Administrator, enters a label, an optional description, a title for the web system, enters the name of a style sheet, then selects the add system button. This information may also be edited for existing systems. As well, Administrators can activate and de-activate existing systems.

The screenshot shows a web application interface for 'Workflow Generator and Tracking System'. At the top right, there are links for 'Home', 'My Account', and 'Log Out'. Below these, it indicates 'Current User: Stacy Hutchings'. The main content area is titled 'Create or Modify A System' and contains a form with the following fields:

- A dropdown menu with 'LNF' selected and an 'Active' checkbox.
- Two tabs: 'System' and 'System Properties', with 'System Properties' selected.
- 'System Label': LNF
- 'Description': Created as an example for the MS Project User Manual
- 'Page Title': LNF Workflow System
- 'Stylesheet': styleLNF.css
- 'Active':
- 'Created By': Stacy Hutchings 12/01/2006 3:15 PM
- 'Modified By':

At the bottom right of the page, there is a copyright notice: 'Copyright © 2006, ILS, Generate Project, Stacy Hutchings'.

Figure B11: The Manage Systems - System Tab

(System Management, Cont'd)

The System Processes tab is shown in Figure B12. This screen allows the Administrator to associate any published process definitions with the given system. The Administrator chooses the system from the list of published definitions then selects the add process button. Administrators may also remove process definitions from existing systems by selecting the remove option then the update button.

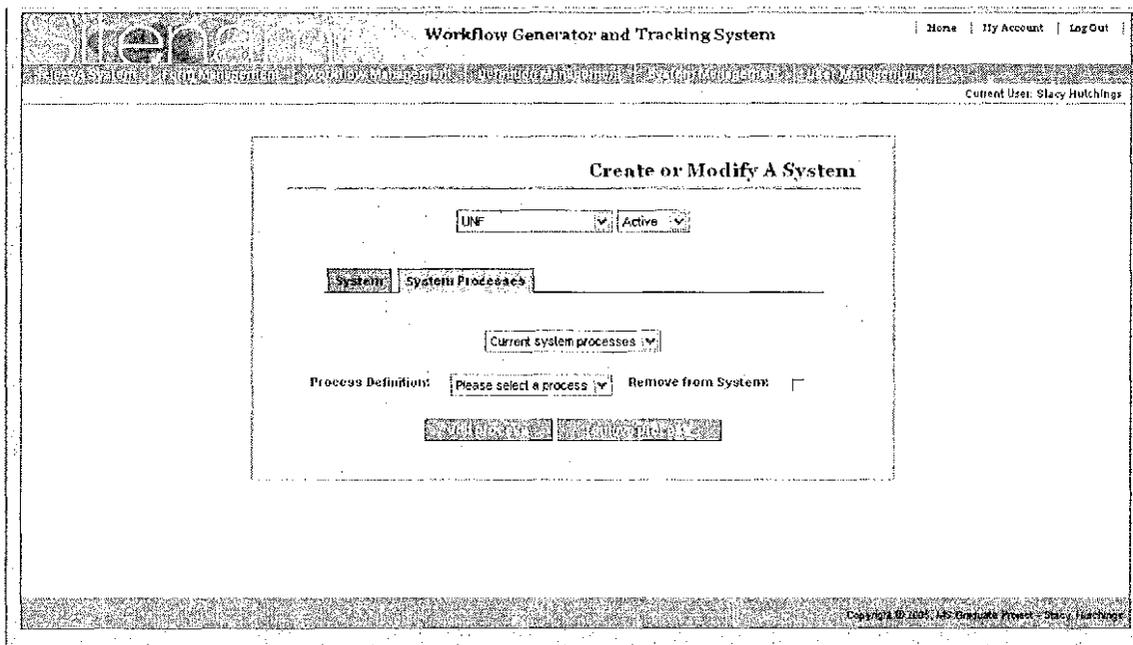


Figure B12: The Manage Systems - System Processes Tab

## Form Management

In order to demonstrate the form management tools, we will build a simple form for requesting a day off. It will be one page containing two sections, one for the Employee and one for the Manager. The Employee section will have three controls: a dropdown list control, containing the type of request; a date control, indicating the start date of the request; and a second date control, indicating the end date of the request. The Manager section will have a text area control for comments that will be required in the event the Manager rejects the request.

The Form Management menu gives the Workflow Engineer three options: Design Form, Modify Published Form, and Manage Form Lists.

### Manage Form Lists

The Category tab is shown in Figure B13. Workflow Engineers can create their own lists for use within Dropdown, Radio, or Checkbox controls. For this example, we need to create a list that contains each of the day off request types. We must enter the required information and select the add button.

The screenshot shows a web application interface for the 'Workflow Generator and Tracking System'. At the top right, there are links for 'Home', 'My Account', and 'Log Out'. Below these, it indicates 'Current User: Stacy Hutchings'. The main content area is titled 'Create or Modify A Form List'. It contains a form with the following elements: a dropdown menu labeled 'Please select a category' with a value of 'Active', a 'Category' label, a 'Name' text input field, a 'Category:' label with a value of 'Day Off RequestTypes', a 'Description:' label with a text area containing 'Types of days off', an 'Active:' checkbox which is checked, and labels for 'Created By:' and 'Modified By:'. At the bottom of the form, there are two buttons: 'Add' and 'Cancel'. The footer of the page contains the text 'Copyright © 2005, 148 Business Process - Stacy Hutchings'.

Figure B13: The Manage Form Lists - Category Tab

(Form Management, Cont'd)

The Item tab is shown in Figure B14. Within this tab, we add each of our custom day off request types. We will enter two types: Personal Day and Vacation Day.

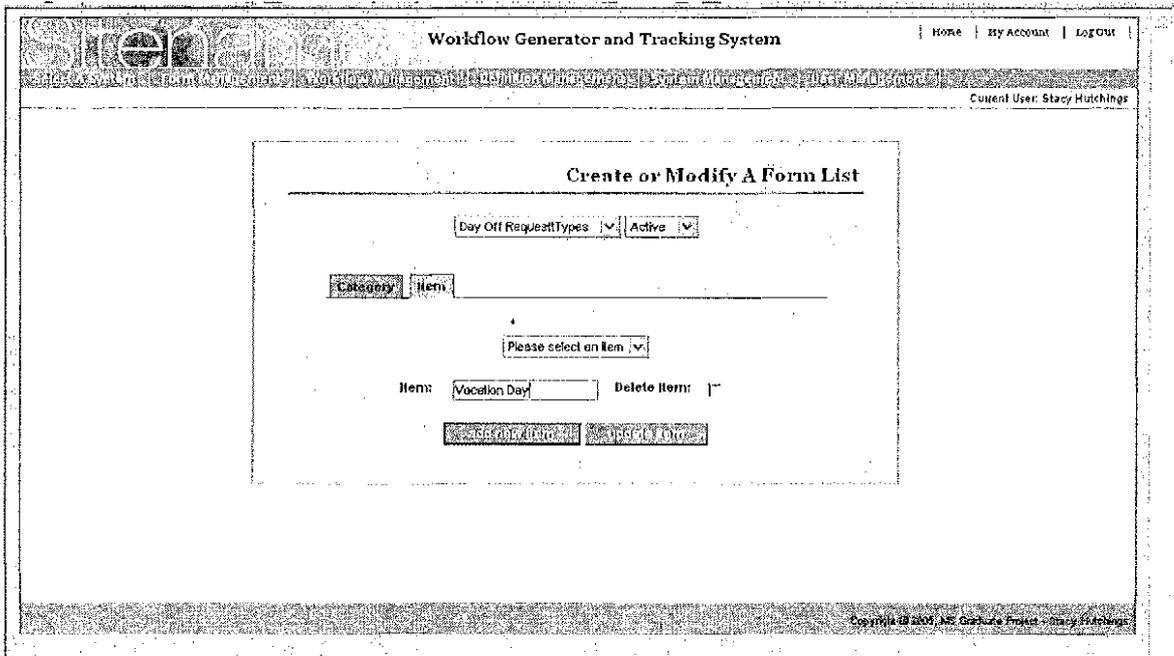


Figure B14: The Manage Form Lists - Item Tab

(Form Management, Cont'd)

## Design Form

The Form tab is shown in Figure B15. The Workflow Engineer enters a form label and an optional description before adding the form. The mode defaults to design mode so full editing capabilities are available to her.

The screenshot shows a web application titled "Workflow Generator and Tracking System". The page is for "Design A New Form". At the top right, there are links for "Home", "My Account", and "Log Out", and a message "Current User: Stacy Hutchings". Below the title, there is a dropdown menu that says "Please select a form". A navigation bar contains tabs for "Form", "Pages", "Sections", "Controls", and "View Form". The main form area includes:

- "Form Label:" with the text "Day Off Request Form" in a text box.
- "Description:" with a text area containing "A single form used for requesting a day off".
- "Mode:" with a dropdown menu set to "Design".
- "Delete Form:" with a checkbox.
- "Version:" with a text box.
- "Created By:" with a text box.
- "Modified By:" with a text box.

At the bottom of the form area, there are two buttons: "Add Form" and "View Form". A footer at the bottom right of the page reads "Copyright © 2006, MSU Graduate Program, Stacy Hutchings".

Figure B15: The Design Form - Form Tab

(Form Management, Cont'd)

The Pages tab is shown in Figure B16. The Workflow Engineer must select the form she would like to modify from the list of available forms. The form we are building will contain a single page, so we will add a page called "pageone." We enter "pageone" as the label and select the add button. We do not need to worry about the order since there is only one page.

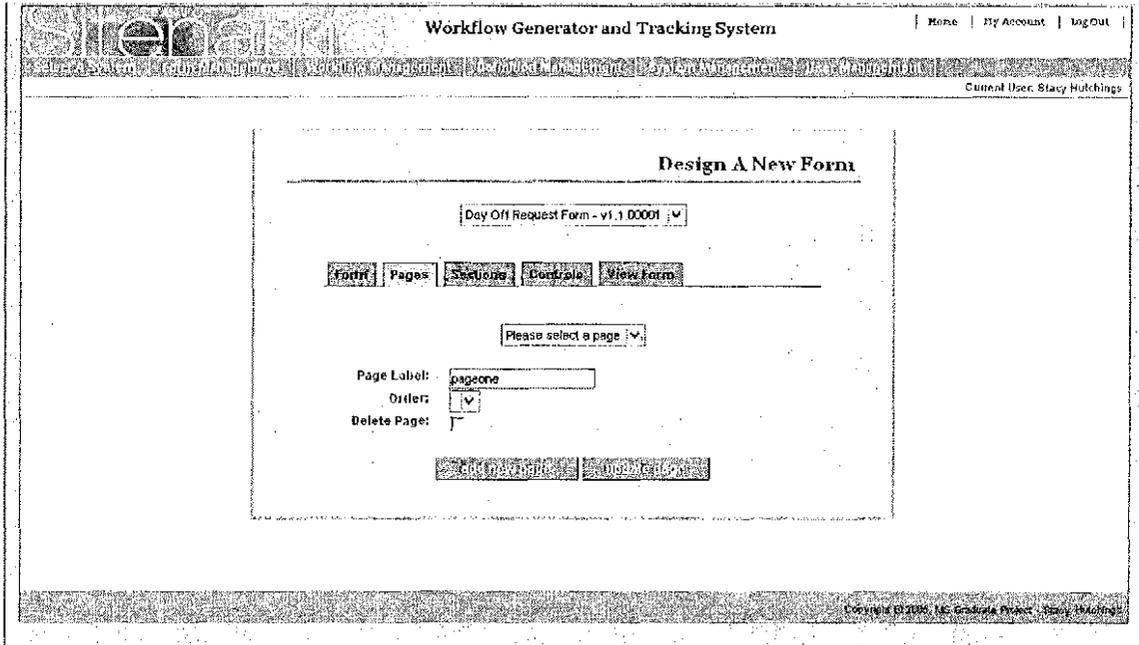


Figure B16: The Design Form - Pages Tab

(Form Management, Cont'd)

The Sections tab is shown in Figure B17. The form we are building will contain two sections, so we will add two sections, one called "Employee Information" and the other called "Manager Information." We enter the label for the section and select the add button. We repeat this for each section. In this example, we would like the Manager section to appear after the Employee section. After adding the Manager section, we will change the order number to 2 and select the update button.

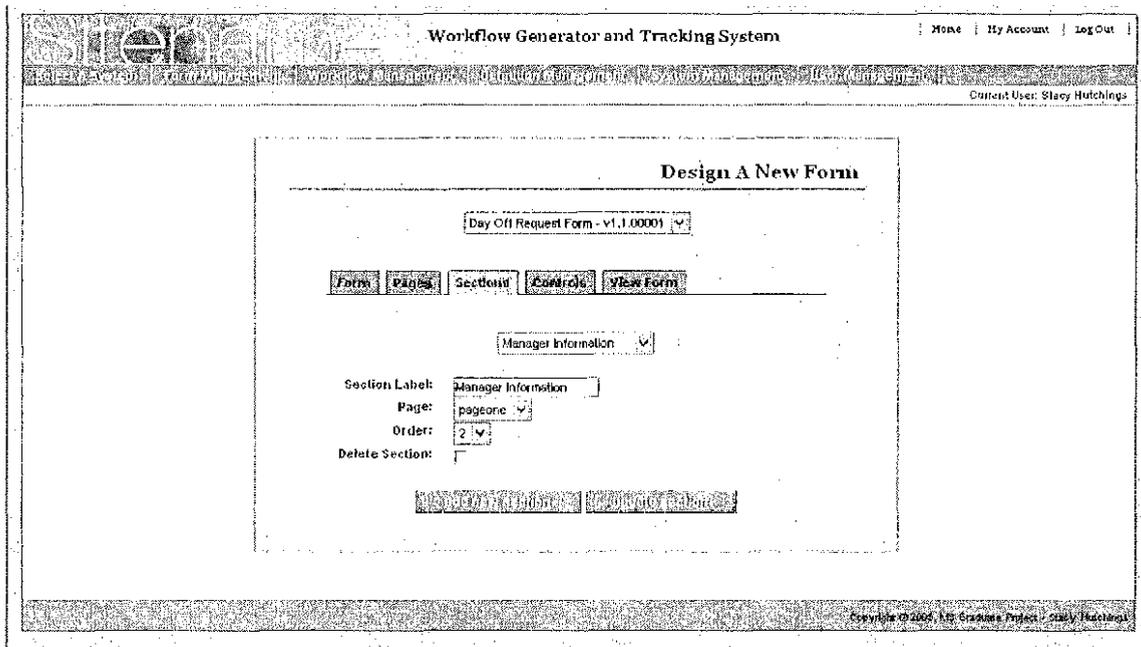


Figure B17: The Design Form - Sections Tab

(Form Management, Cont'd)

The Controls tab is shown in Figure B18. The form we are building will contain three controls for the Employee section and a single control for the Manager section. For the Employee section we will add one dropdown control and two date controls. For the Manager section, we will add one text area control.

First, we select a control type. We will select DROPDOWN from the list of choices. Notice that the Select List option becomes available. Because this is a Dropdown control, we must select the list we want to associate with this control. We will choose the Day Off Request Types list we created earlier. Next, we enter a label for our control as well as an optional tool tip. Once we confirm the appropriate section is selected, we add the control. We repeat this process for the remaining controls we want to appear on the form.

The screenshot shows the 'Design A New Form' application window. At the top, the title bar reads 'Design A New Form'. Below the title bar, there is a dropdown menu showing 'Day Off Request Form - v1.1.00001'. A set of tabs is visible, with 'Controls' selected. The main area contains a configuration panel for a control. At the top of this panel is a dropdown menu labeled 'Please select a control'. Below it, the following fields are visible: 'Control Type:' set to 'DROPDOWN', 'Select List:' set to 'Day Off RequestTypes', 'Control Label:' set to 'Request type:', and 'Tooltip:' set to 'Please choose the type of day off you are requesting'. There is also a 'Default Value:' field which is empty. Below these are 'Width:' and 'Height:' fields. The 'Section:' dropdown is set to 'pageone - Employee Information'. At the bottom of the panel, there is an 'Order:' dropdown and a 'Delete Control:' checkbox. At the very bottom of the application window, there are two buttons: 'OK' and 'Cancel'.

Figure B18: The Design Form - Controls Tab

(Form Management, Cont'd)

The results of selecting the View Form tab can be seen in Figure B19. The form we are building can be viewed at any time during the build process. Our form is almost complete. The controls appear in the correct order. However, the comment area looks too narrow. We can go back to the control tab, select the Reason For Denial control, and update the size of the control. The new results are shown in Figure B20.

The screenshot shows a web-based form design interface. At the top, there is a header bar with the text "of VA Form 100-100-001". Below this, the form is divided into two main sections. The first section, titled "Employee Information", contains three fields: "Request Type" with a dropdown arrow, "Start Date" with a date picker icon, and "End Date" with a date picker icon. The second section, titled "Manager Information", contains a single text area labeled "Reason For Denial". At the bottom center of the form, there is a button labeled "Save".

Figure B19: The Design Form - Results of View Tab

This screenshot is identical to Figure B19, showing the same form design interface. However, the "Reason For Denial" text area in the "Manager Information" section is now significantly larger and wider, reflecting the update mentioned in the text. The "Save" button remains at the bottom center.

Figure B20: The Design Form - Results After Update

## Workflow Management

In order to demonstrate the workflow management tools, we will build a simple workflow for requesting a day off. This workflow will contain four states. The initial state will be displayed as "Pending Employee Request," the next state will be "Pending Manager Review," the next two states will be based on the Manager's decision. If the Manager approves the request, the instance will enter a Closed state. If the Manager denies the request, the instance will enter the "Request Denied" state, for the employee to review.

The Workflow Management menu gives the Workflow Engineer three options: Design Workflow, Modify Published Workflow, and Manage Roles.

### Manage Roles

The Role tab is shown in Figure B21. Workflow Engineers can create custom roles to contain the appropriate members of the organization. For this example, we need to create two roles: Employee and Manager. To create a role, enter the role name and an optional description, then select the add button.

The screenshot shows a web application interface titled "Workflow Generator and Tracking System". At the top right, there are links for "None", "My Account", and "Log Out". Below this, it says "Current User: Stacy Hutchings". The main content area is titled "Create or Modify A Role". It features a dropdown menu with the text "Please select a role". Below the dropdown are two tabs: "Account Members" and "Group Members". The "Account Members" tab is active. The form contains the following fields:

- Role:** A text input field containing the word "Employee".
- Description:** A text area containing the text "Contains all users of the system".
- Created By:** An empty text input field.
- Modified By:** An empty text input field.

At the bottom of the form, there are two buttons: "Add Role" and "Modify Role". In the bottom right corner of the application frame, there is a small copyright notice: "Copyright © 2005, SAS Institute, Inc. - Stacy Hutchings".

Figure B21: The Manage Roles - Role Tab

(Workflow Management, Cont'd)

The Account Members tab is shown in Figure B22. Within this tab, we will add Stacy Hutchings as the only member of the Manager role. We must have the Manager role selected before we begin. Since the current role member list is empty, we must select Stacy Hutchings from the list of members, then select the add button. We will repeat this process for the Employee role. However, we will add the All User group to this role as shown in Figure B23.

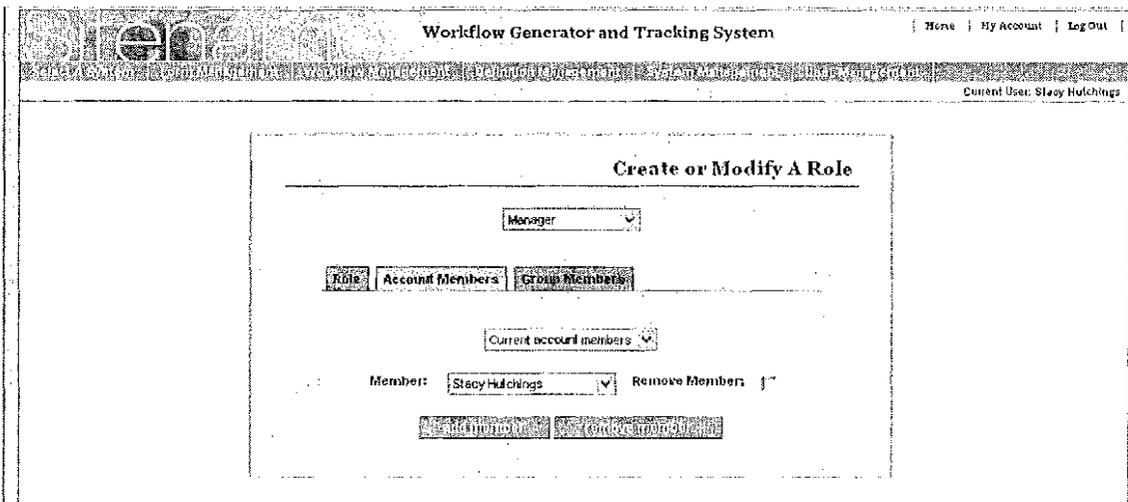


Figure B22: The Manage Roles - Account Members Tab

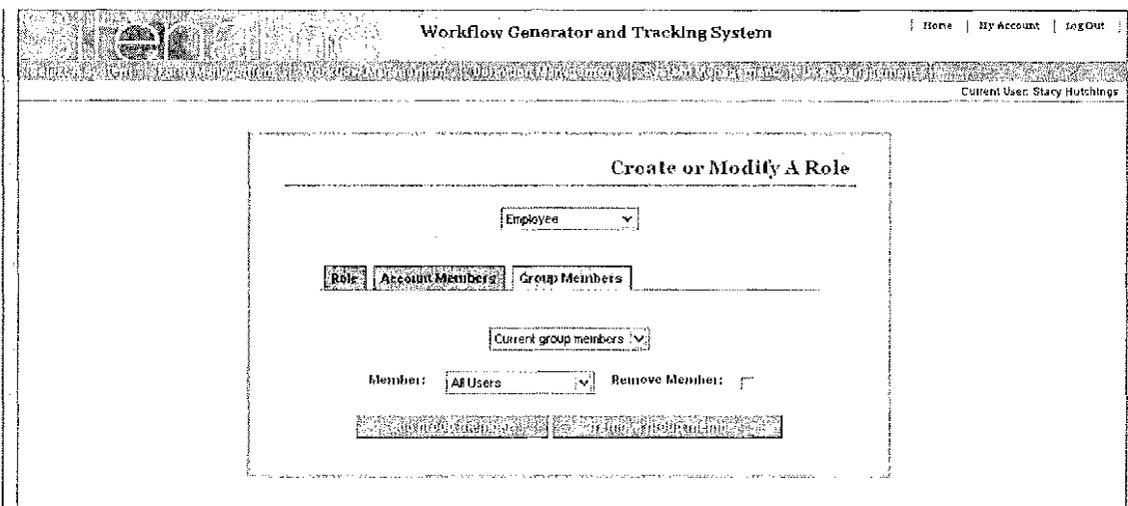


Figure B23: The Manage Roles - Group Members Tab

(Workflow Management, Cont'd)

## Design Workflow

The Workflow tab is shown in Figure B24. The Workflow Engineer enters a workflow label and an optional description before adding the workflow. The mode defaults to design mode so full editing capabilities are available to her.

The screenshot shows a web application interface for creating a workflow. The page title is "Design A New Workflow". At the top, there is a navigation bar with "Home", "My Account", and "Log Out" links. Below the navigation bar, the current user is identified as "Stacy Hutchings". The main content area contains a form with the following elements:

- A dropdown menu labeled "Please select a workflow" with a downward arrow.
- A set of five tabs: "Workflow", "States", "Transitions", "From Strategies", and "Tasks/Events". The "Workflow" tab is currently selected.
- A "Workflow Label:" field containing the text "Request Day Off Workflow".
- A "Description:" field containing the text "Created to demonstrate how to build a workflow".
- A "Mode:" dropdown menu set to "Design".
- A "Delete Workflow:" checkbox, which is currently unchecked.
- Fields for "Version:", "Created By:", and "Modified By:", which are currently empty.
- At the bottom of the form, there are two buttons: "Add Workflow" and "Cancel".

At the bottom right of the page, there is a small copyright notice: "Copyright © 2000, All Rights Reserved. Stacy Hutchings".

Figure B24: The Design Workflow – Workflow Tab

(Workflow Management, Cont'd)

The States tab is shown in Figure B25. The Workflow Engineer must select the workflow she would like to modify from the list of available workflows. The workflow we are building will contain four states. Since one of them will be transitioning to the Close state, which is already provided by the system, we only need to create three states.

The first state we enter will be our initial state, although the order we enter them does not matter. For this entry, we must change the initial state indicator to YES, on subsequent requests we will leave it as NO. We will assign the Employee role as the participant of this state. We will also give the participant permission to cancel this instance. However, we do not want them to be able to close the request or be able to reassign this request to another Employee.

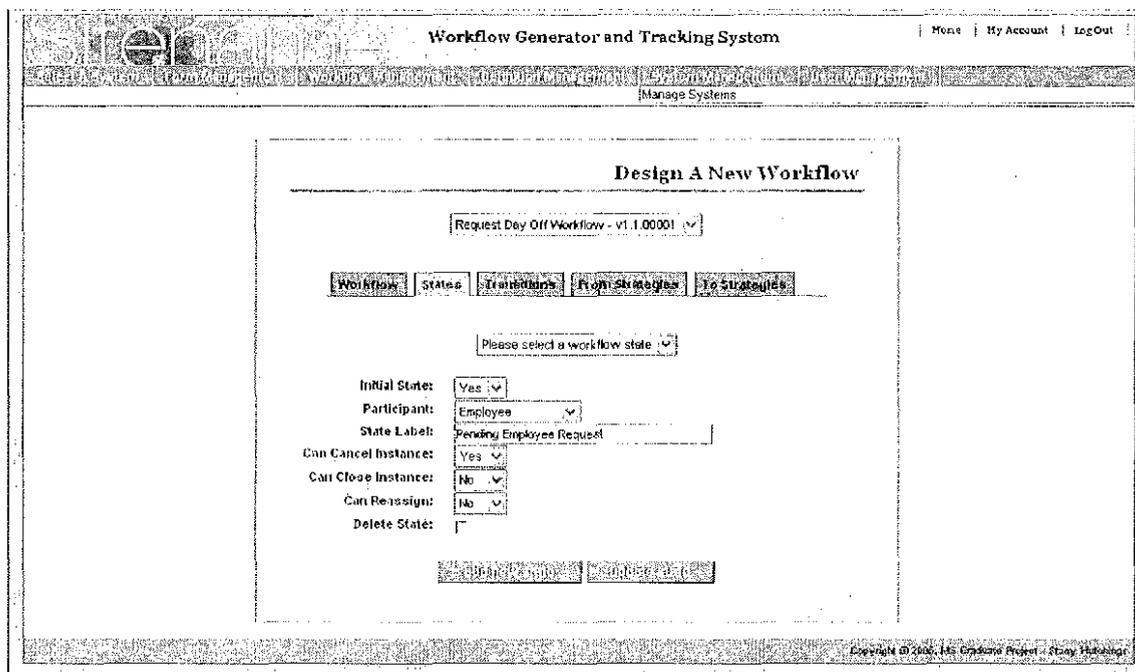


Figure B25: States Tab: Pending Employee Request

(Workflow Management, Cont'd)

Figure B26 and Figure B27 show the remaining states and their settings. The participant at the Request Denied state will be given permission to close the instance.

Workflow Generator and Tracking System | Home | My Account | LogOut  
Current User: Stacy Hutchings

### Design A New Workflow

Request Day Off Workflow - v1.1.00001

Workflow | States | Transitions | From Strategies | To Strategies

Please select a workflow state

Initial State: No  
Participant: Manager  
State Label: Pending Manager Review  
Can Cancel Instance: No  
Can Close Instance: No  
Can Reassign: No  
Delete State:

Cancel State | Delete State

Figure B26: States Tab: Pending Manager Review

Workflow Generator and Tracking System | Home | My Account | LogOut  
Current User: Stacy Hutchings

### Design A New Workflow

Request Day Off Workflow - v1.1.00001

Workflow | States | Transitions | From Strategies | To Strategies

Request Denied

Initial State: No  
Participant: Employee  
State Label: Request Denied  
Can Cancel Instance: No  
Can Close Instance: Yes  
Can Reassign: No  
Delete State:

Cancel State | Delete State

Figure B27: States Tab: Request Denied

(Workflow Management, Cont'd)

The Transitions tab is shown in Figure B28. The workflow we are building will contain three possible transitions: transition one, from Pending Employee Request to Pending Manager Review; transition two, from Pending Manager Review to Close; and transition three, from Pending Manager Review to Request Denied.

For transition one, we select the "from state," the "to state," give the transition a name, then select an assignment strategy. Since the manager role contains only one member, we use the Push-Auto Look Up strategy to let the system automatically assign the instance for us.

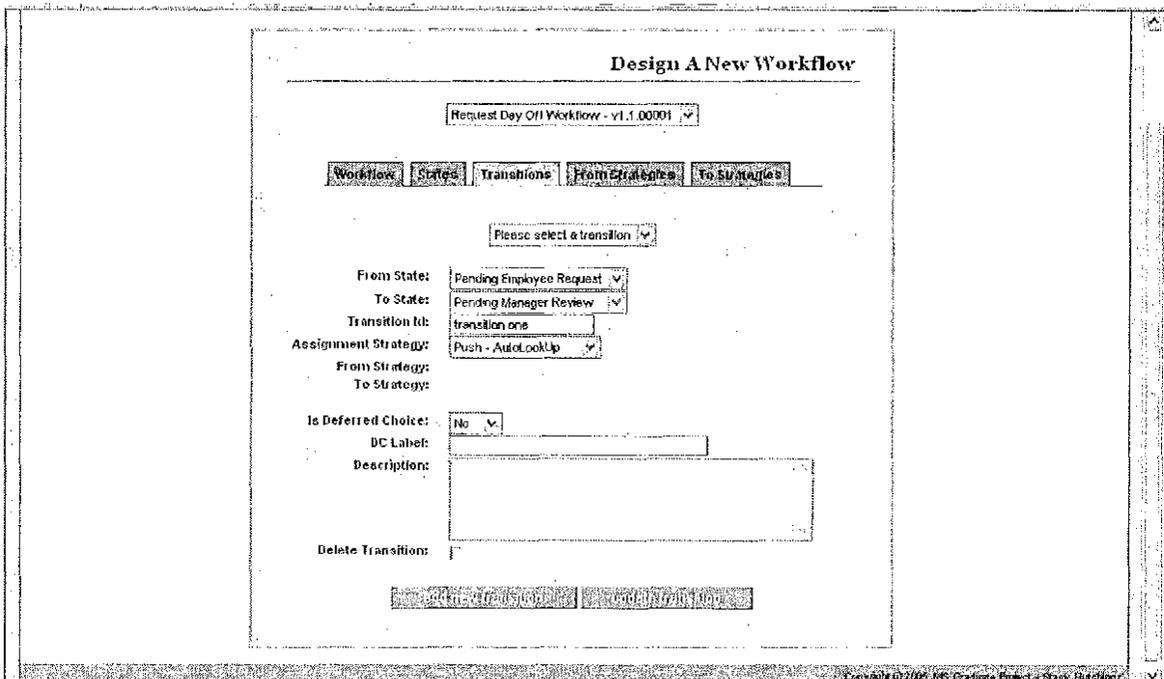


Figure B28: Transitions Tab: Transition One

(Workflow Management, Cont'd)

Figure B29 shows the selections made for transition two. The assignment strategy does not matter when transitioning to the Close state. For the transition strategy, we will use Deferred Choice to present the transition decision to the Manager. We change the Deferred Choice indicator to YES, customize our label, and add an optional description before selecting the add button.

The screenshot displays the 'Design A New Workflow' application window. At the top, the title bar reads 'Design A New Workflow'. Below it, a dropdown menu shows 'Request Day Off Workflow - v1.1.00001'. A navigation bar contains tabs for 'Workflow', 'States', 'Transitions', 'From Strategies', and 'To Strategies', with 'Transitions' currently selected. A central dropdown menu prompts 'Please select a transition'. The configuration fields are as follows: 'From State' is 'Pending Manager Review', 'To State' is 'Close', 'Transition Id' is 'transition two', 'Assignment Strategy' is 'Push - Manual', 'From Strategy' and 'To Strategy' are empty. The 'Is Deferred Choice' checkbox is checked and labeled 'Yes'. The 'DC Label' is 'Approve Request'. The 'Description' field contains the text: 'We will use the deferred choice strategy to present the decision to the Manager'. At the bottom, there is a 'Delete Transition' checkbox which is unchecked, and two buttons: 'Add Transition' and 'Cancel'.

Figure B29: Transitions Tab: Transition Two

(Workflow Management, Cont'd)

Figure B30 shows the selections made for transition three. The assignment strategy will be Push - Manual, since there are multiple users in the Employee role. For the transition strategy, we will again use Deferred Choice to present the transition decision to the Manager.

The screenshot displays the 'Design A New Workflow' application window. At the top, the title bar reads 'Design A New Workflow'. Below it, a dropdown menu shows 'Request Day Off Workflow - v1.1.00001'. A navigation bar contains tabs for 'Workflow', 'States', 'Transitions', 'From Strategies', and 'To Strategies', with 'Transitions' currently selected. A prompt 'Please select a transition' is visible. The main configuration area includes the following fields:

- From State:** Pending Manager Review
- To State:** Request Denied
- Transition Id:** transition three
- Assignment Strategy:** Push - Manual
- From Strategy:**
- To Strategy:**
- To Deferred Choice:** Yes
- DC Label:** Deny Request
- Description:** We will use the deferred choice strategy to present the decision to the Manager
- Delete Transition:**

At the bottom right of the window, there is a copyright notice: 'Copyright © 2005, MS Graduate Program - State Teachers' and a 'Local Intranet' icon.

Figure B30: Transitions Tab: Transition Three

(Workflow Management, Cont'd)

The From Strategies tab and the To Strategies tab provide a summary of the current transitions from two different perspectives. Figure B31 shows all transitions originating from a given state within the current workflow. Figure B32 shows all transitions converging on a given state within the current workflow.

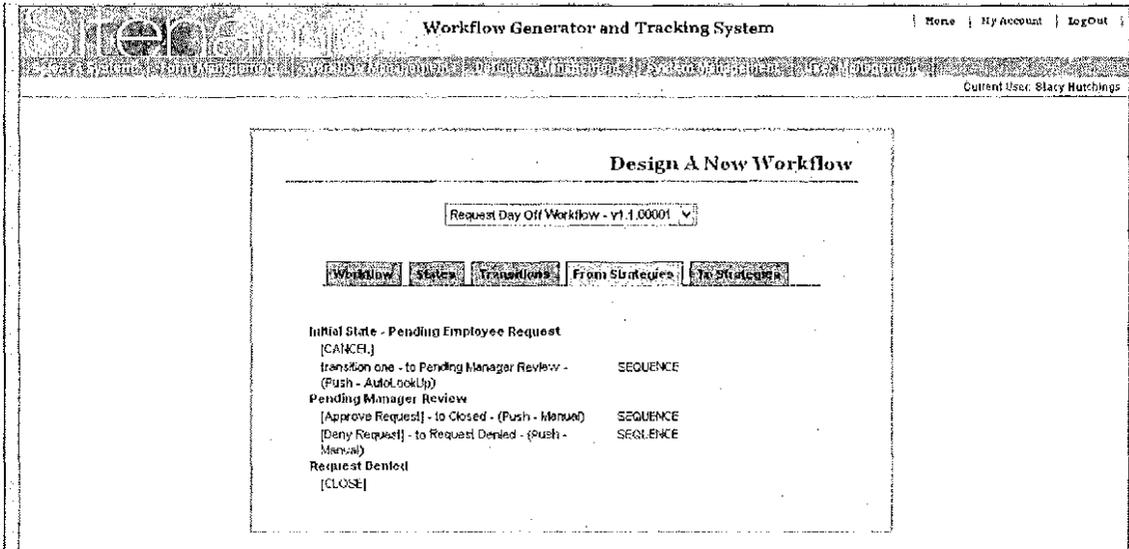


Figure B31: The Design Workflow - From Strategies Tab

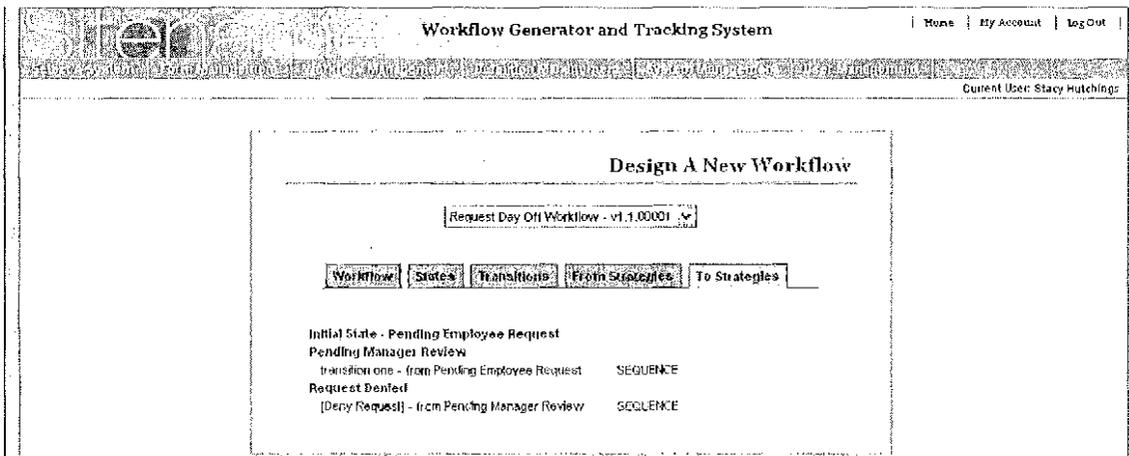


Figure B32: The Design Workflow - To Strategies Tab

(Workflow Management, Cont'd)

Before we can define the process definition, both the workflow and the form must be placed in test mode as shown in Figure B33 and B34.

Design A New Form

Day Off Request Form - v1.1.00001

Form Pages Sections Controls New Form

Form Label: Day Off Request Form

Description: A simple form used for requesting a day off

Mode: Test

Delete Form:

Version: 1.1.00001

Created By: Stacy Hutchings 12/01/2005 4:02 PM

Modified By: Stacy Hutchings 12/01/2005 7:16 PM

Save

Figure B33: The Design Form - Form Tab

Design A New Workflow

Request Day Off Workflow - v1.1.00001

Workflow States Transitions From Strategies To Strategies

Workflow Label: Request Day Off Workflow

Description: Created to demonstrate how to build a workflow

Mode: Test

Delete Workflow:

Version: 1.1.00001

Created By: Stacy Hutchings 12/01/2005 5:55 PM

Modified By: Stacy Hutchings 12/01/2005 7:10 PM

Save

Figure B34: The Design Workflow - Workflow Tab

## Definition Management

In order to demonstrate the definition management tools, we will add several form rules. We will also customize which fields should appear in the pending task queue within the task management component.

The Definition Management menu gives the Workflow Engineer two options: Design Definition and Modify Published Definition.

The Definition tab is shown in Figure B35. The Workflow Engineer enters a definition label and an optional description before adding the form. In addition, she must select a form and a workflow. The mode defaults to design mode so full editing capabilities are available to her.

The screenshot displays the 'Design A New Workflow Definition' interface. At the top, there is a navigation bar with 'Home', 'My Account', and 'Logout' links. Below this, the current user is identified as 'Stacy Hutchings'. The main content area is titled 'Design A New Workflow Definition' and contains a dropdown menu for selecting a definition. Below the dropdown are four tabs: 'Definition', 'Create Fields', 'Rules', and 'View Rules'. The 'Definition' tab is active, showing a form with the following fields and values:

- Definition Label: RequestADayOff
- Description: Process definition for the request a day off workflow
- Form: Day Off Request Form - v1.1.00001
- Workflow: Request Day Off Workflow - v1.1.00001
- Mode: Design
- Version: (empty)
- Created By: (empty)
- Modified By: (empty)

At the bottom of the form, there are two buttons: 'Save' and 'Cancel'.

Figure B35: The Design Definition - Definition Tab

(Definition Management, Cont'd)

The Queue Fields tab is shown in Figure B36. The Workflow Engineer must select the definition she would like to modify from the list of available process definitions. Each dropdown list contains the form controls available for display in the task management component. We will select Request Type, Start Date, and End Date.

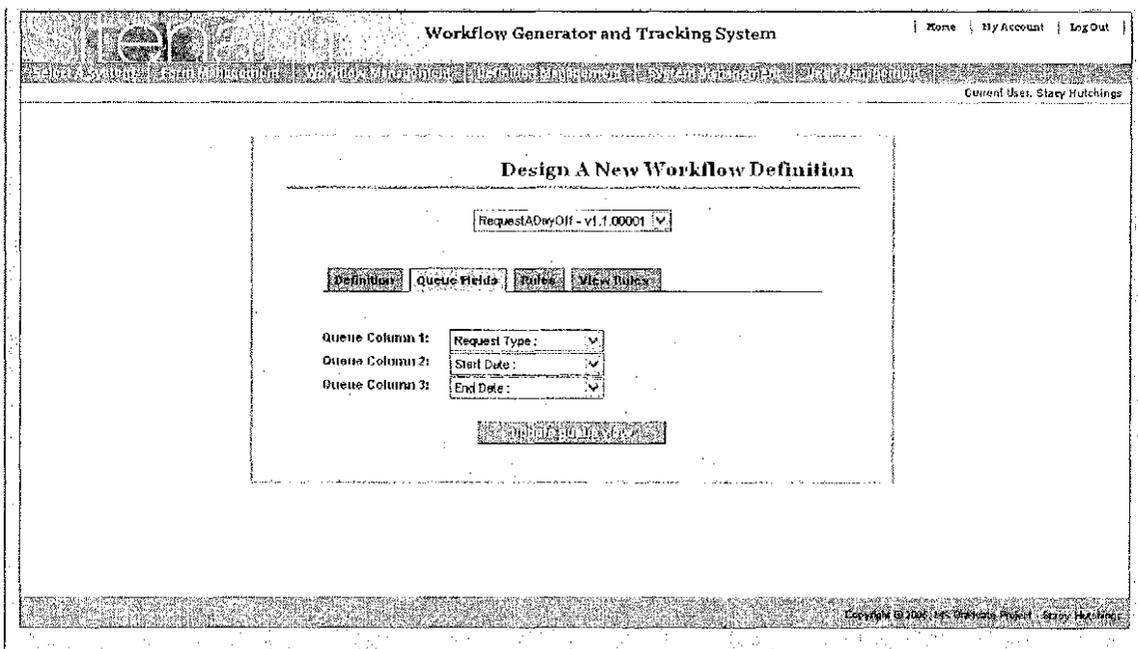


Figure B36: The Design Definition – Queue Fields Tab

(Definition Management, Cont'd)

The Rule tab gives the Workflow Engineer the ability to build transition as well as form rules. For this workflow, there are no required transition rules as shown in Figure B37. However, we would like to define several form rules. First, we would like to make the Request Type, Start Date, and End Date required for transition one. Second, we would like to add a rule to validate the Start Date is before the End Date. Third, we would like to make the Comments required for transition three.

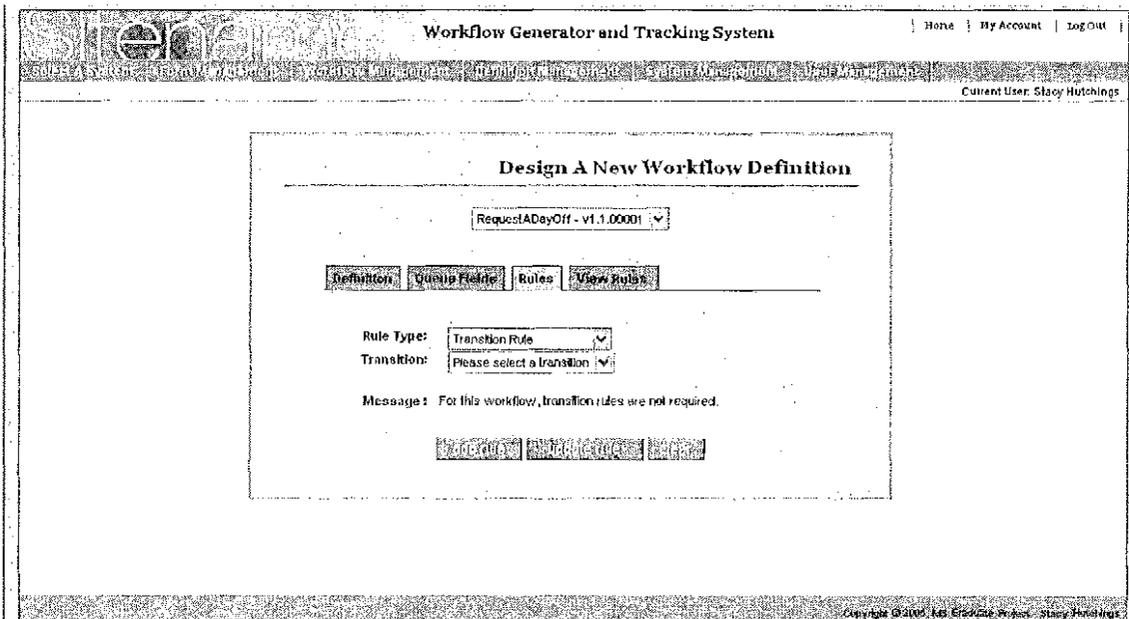


Figure B37: The Design Definition - Rules Tab

(Definition Management, Cont'd)

Figure B38 and Figure B39 show examples of the rule definitions for transition one. The type is Form Rule, the transition is transition one, the control to validate is indicated, and we are using the required validator.

Design A New Workflow Definition

RequestADayOff - v1.1.00001

Definition Queue Fields Rules View Rules

Rule Type: Form Rule

Transition: transition one

Control To Validate: Employee Information - Request Type

Control Type: STRING

Validator: RequiredValidator

Error Message: Request Type is required

Delete Rule: [X] [OK] [Cancel]

Figure B38: Rules Tab: T1 - Request Type Required

Design A New Workflow Definition

RequestADayOff - v1.1.00001

Definition Queue Fields Rules View Rules

Rule Type: Form Rule

Transition: transition one

Control To Validate: Employee Information - Start Date

Control Type: DATE

Validator: RequiredValidator

Error Message: Start Date is required

Delete Rule: [X] [OK] [Cancel]

Figure B39: Rules Tab: T1 - Start Date Required

(Definition Management, Cont'd)

Figure B40 shows the rule definitions for transition three. The type is Form Rule, the transition is transition three, the control to validate is indicated, and we are using the required validator. The View Rules tab, shown in Figure B41, provides a summary of all rules defined for the current process.

Design A New Workflow Definition

RequestADayOff - v1.1.00001

Definition Queue Fields Rules **View Rules**

Rule Type: Form Rule

Transition: transition three

Control To Validate: Manager Information - Reason For Denial

Control Type: STRING

Validator: RequiredValidator

Error Message: Comments are required

Delete Rule:

Back Cancel Save

Figure B40: Rules Tab: T3 - Reason For Denial Required

Design A New Workflow Definition

RequestADayOff - v1.1.00001

Definition Queue Fields Back **View Rules**

Form Rules:

- transition one - Request Type: is required
- transition one - Start Date: is required
- transition one - End Date: is required
- transition three - Reason For Denial is required

Figure B41: The Design Definition - View Rules Tab

(Definition Management, Cont'd)

We have completed our process definition. We are ready to publish the definition. Publishing the definition automatically publishes the form as well as the workflow and makes the process definition available for system assignment. Figure B42 shows the process definition in published mode.

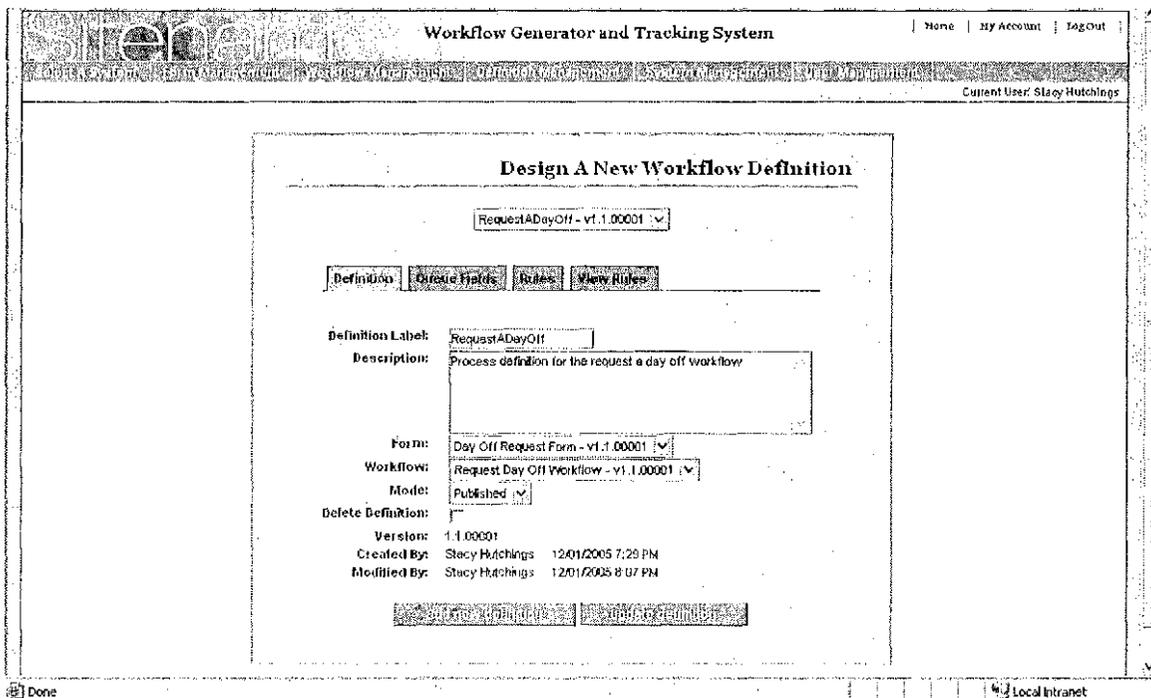


Figure B42: The Design Definition - Definition Tab

## System Management

Now that we have a published process definition, we are ready to add the process to an existing system. Figure B43 shows the Manage Systems - System Processes tab. We will add the RequestADayOff process definition to the UNF System. Figure B44 shows the choices beneath the Select A System menu.

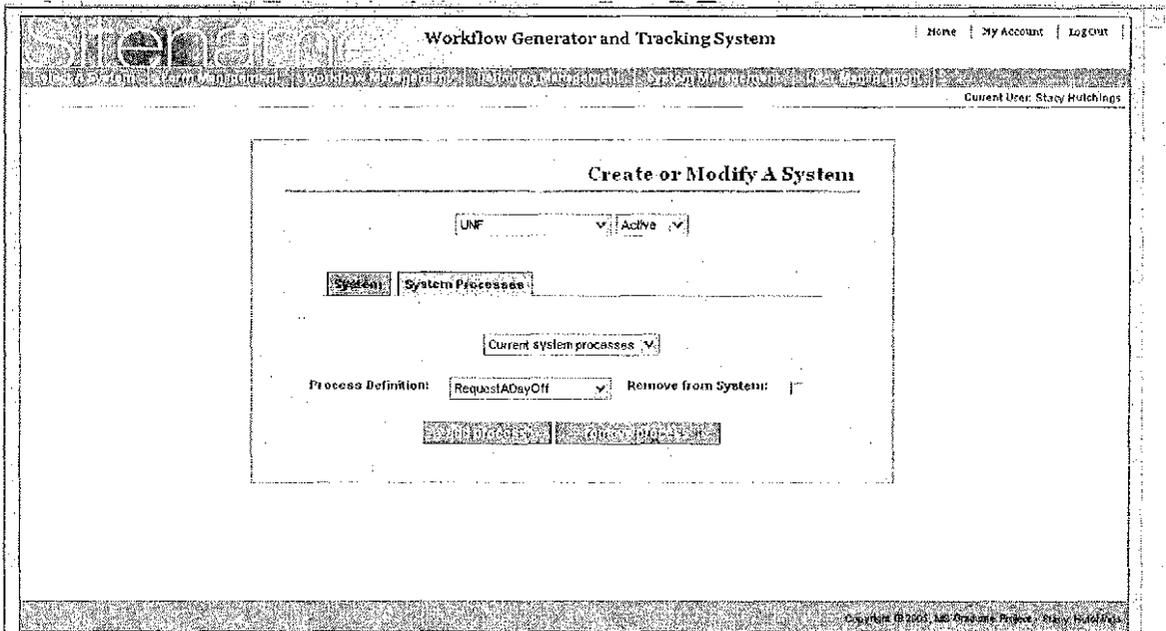


Figure B43: The Manage Systems - System Processes Tab

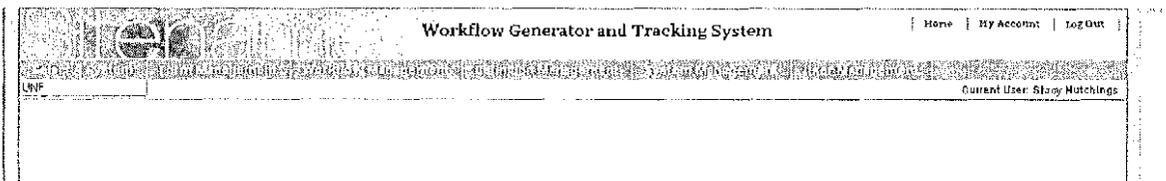


Figure B44: The Home Page - Select A System Menu

## Task Management

To access the task management component, a user must select a system. Figure B45 shows the resulting workflow system when the user selects the UNF system.

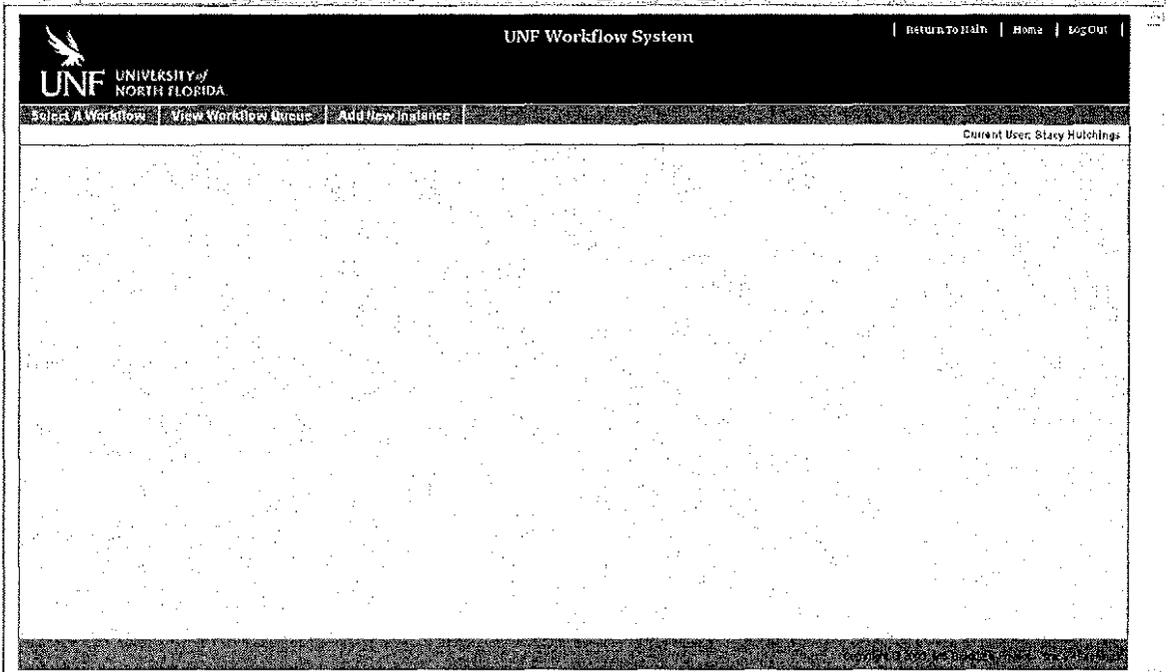


Figure B45: The UNF System - Home Page

(Task Management, Cont'd)

The Task Management component gives the Standard User three options: Select A Workflow, View a Workflow Queue, and Add A New Instance.

The user must select a workflow in order to access the other menu options. When the user chooses the RequestADayOff workflow, the system presents the current user with a list of pending tasks. In Figure B46, there are no pending tasks for Stacy Hutchings.

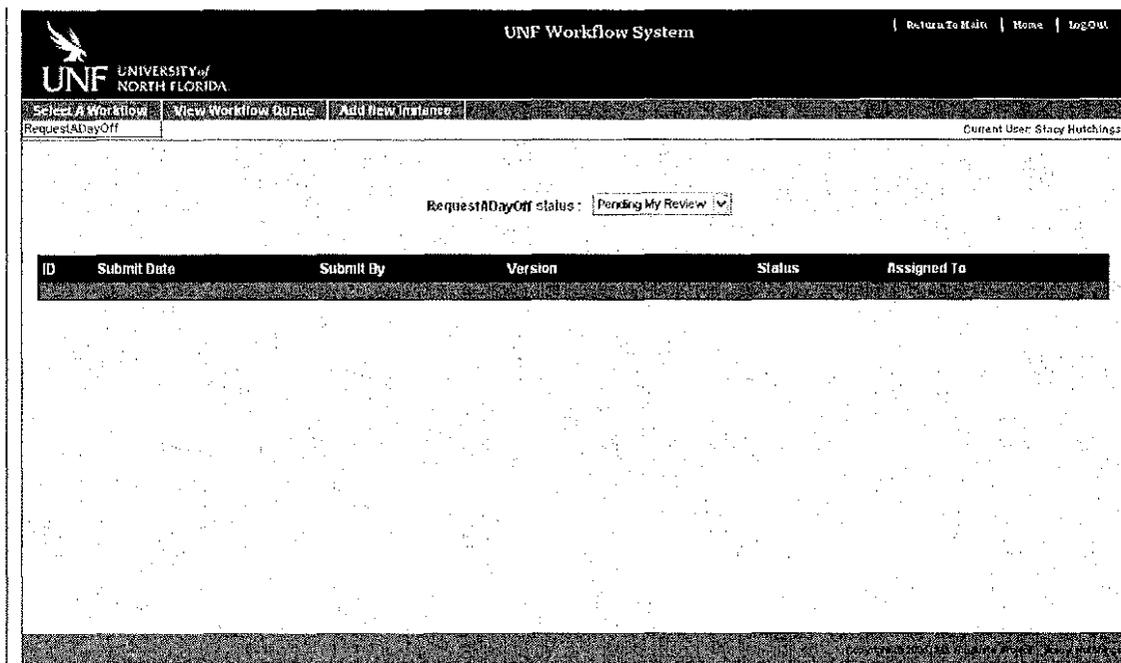


Figure B46: The UNF System - View Workflow Queue

(Task Management, Cont'd)

The View Workflow Queue option automatically defaults to the pending tasks of the current user. The user has the ability to view All Open, Canceled, and Closed tasks, see Figure B47. Any additional workflow states that contain tasks will also appear in the list to choose from.

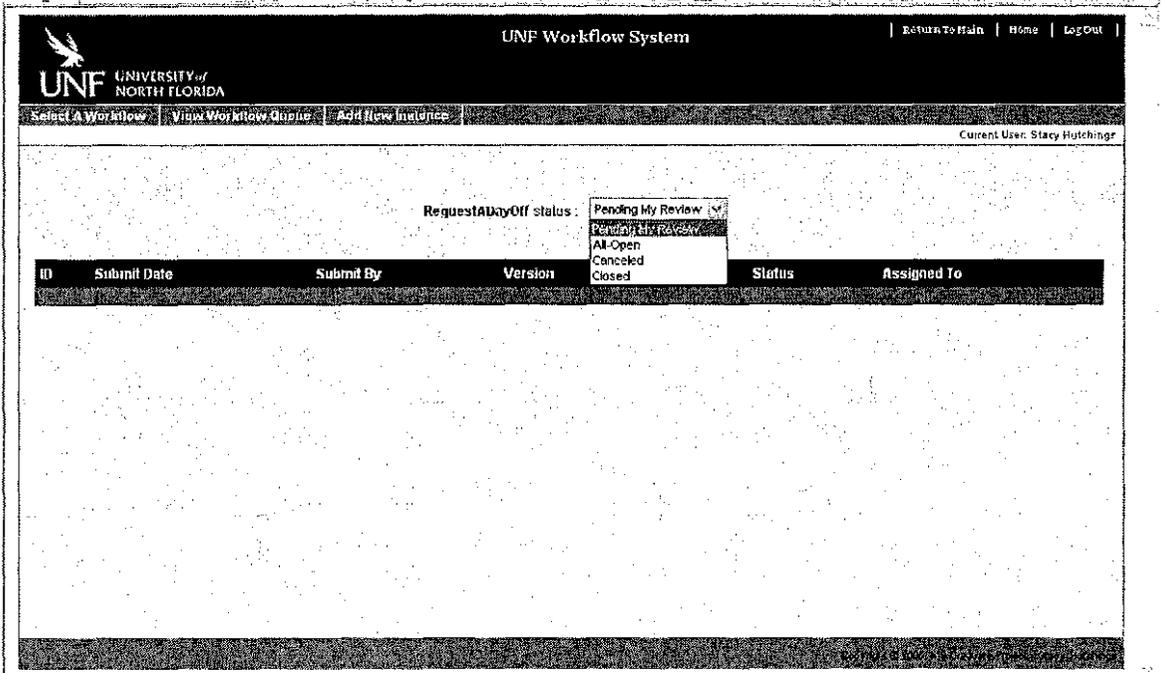


Figure B47: The UNF System - View Workflow Queue

(Task Management, Cont'd)

When the user is ready to add a new workflow instance, she selects the Add New Instance option. The system presents the user with the Day Off Request Form shown in Figure B48. This is the custom form built by the Workflow Engineer for the RequestADayOff process.

The screenshot displays the 'RequestADayOff - Day Off Request Form' within the UNF Workflow System. The page header includes the UNF logo and 'UNIVERSITY of NORTH FLORIDA'. Navigation links for 'Return to Main', 'Home', and 'Log Out' are present. A menu bar contains 'Select A Workflow', 'View Workflow Queue', and 'Add New Instance'. The current user is identified as 'Slacy, Hutchings'. The form itself is titled 'RequestADayOff - Day Off Request Form' and is divided into two main sections: 'Employee Information' and 'Manager Information'. The 'Employee Information' section contains a 'Request Type' dropdown menu, a 'Start Date' field with a calendar icon, and an 'End Date' field with a calendar icon. The 'Manager Information' section features a 'Reason For Denial' text area. At the bottom of the form is a button labeled 'add new instance'.

Figure B48: The UNF System - Add New Instance

## Request A Day Off Example in Action

To demonstrate the entire workflow process from within the Task Management component, Jimi Hutchings and Caitlyn Hutchings will play the part of the Employee. Stacy Hutchings will play the part of the Manager. Jimi and Caitlyn will each submit a request for time off. Stacy will approve the first request and deny the second request, to demonstrate the possible workflow transition paths.

To submit a request for time off, Jimi Hutchings selects the Add New Instance option. He is presented with the screen in Figure B48. He enters the type of request, the start date of his time off, and the end date of his time off. Once complete, he selects the add button. The request appears in his list of pending tasks, see Figure B49.

The queue displays the request ID, the date submitted, the name of the person who created the request, the version of the workflow definition, the three form fields selected by the workflow designer, the current status, and the person who the task is currently assigned to.

The screenshot shows the UNF Workflow System interface. At the top, there is a navigation bar with links for 'Return to train', 'Home', and 'LogOut'. Below this is a header for 'UNF UNIVERSITY of NORTH FLORIDA' and a menu with options: 'Select A Workflow', 'View Workflow Queue', and 'Add New Instance'. The user is identified as 'General User: Jimi Hutchings'. A dropdown menu for 'Request A Day Off status' is set to 'Pending My Review'. Below this is a table with the following data:

ID	Submit Date	Submit By	Version	Status	Assigned To			
15	12/04/2006 2:57 PM	Jimi Hutchings	1.1.00001	Personal Day	12/08/2005	12/09/2005	Pending Employee Request	Jimi Hutchings

Figure B49: The UNF System - J Hutchings' Pending Queue

(Request A Day Off Example in Action, Cont'd)

To view the task, Jimi selects the ID of the request. The system presents him with the view shown in Figure B50. Within the summary tab, the details of the form are displayed along with the option to update.

The screenshot displays the 'UNF Workflow System' interface. At the top, there are navigation links: 'Return To Main', 'Home', and 'Log Out'. Below this is the 'UNF UNIVERSITY of NORTH FLORIDA' logo. A menu bar contains 'Select A Workflow', 'View Workflow Queue', and 'Add New Instance'. The current user is identified as 'Jimi Hutchings'. The main content area has tabs for 'Summary', 'Comments', 'Transition', and 'Reassign', with 'Summary' selected. Below the tabs, the instance details are shown: 'ID: 15', 'Submitted By: Jimi Hutchings', 'Submit Date: 12/04/2005 2:57 PM', 'Current Status: Pending Employee Request', and 'Assigned To: Jimi Hutchings'. The form title is 'RequestADayOff - Day Off Request Form'. It is divided into two sections: 'Employee Information' and 'Manager Information'. The 'Employee Information' section contains 'Request Type' (Personal Day), 'Start Date' (12/09/2005), and 'End Date' (12/09/2005). The 'Manager Information' section contains a 'Reason For Denial' field. At the bottom of the form is a 'Update Instance' button.

Figure B50: View Instance Details - Summary Tab

(Request A Day Off Example in Action, Cont'd)

Within the comments tab, the history of the instance is displayed. The system records all transition as well as assignment events. This screen also gives all users the ability to add comments. Figure B51 shows the Comments tab.

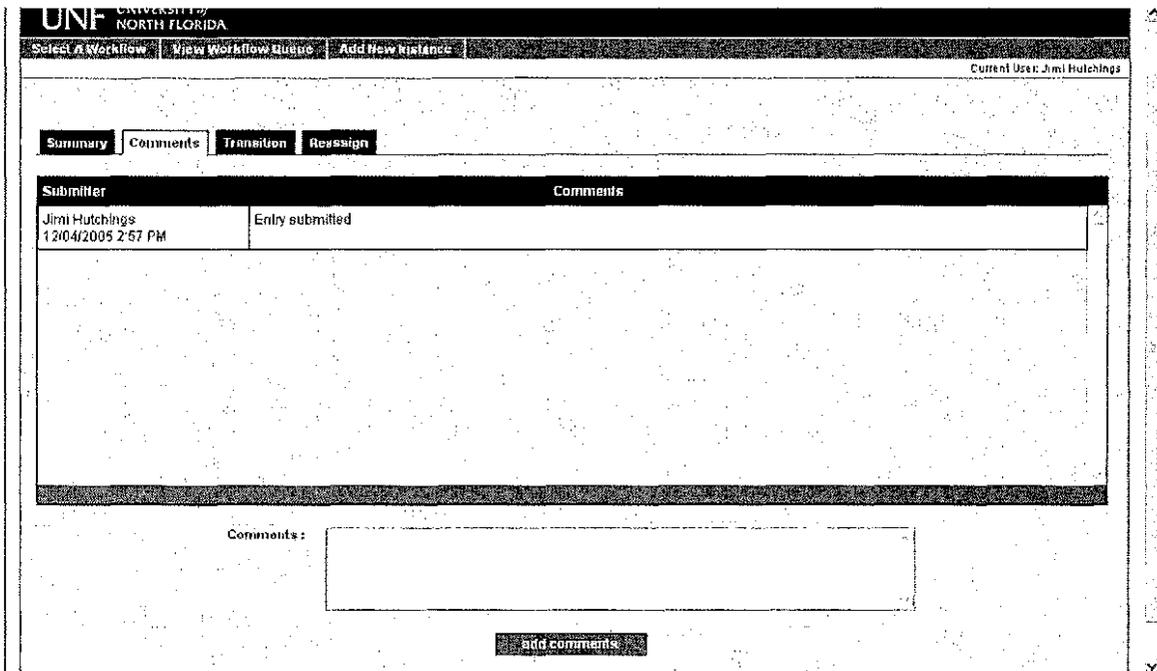


Figure B51: View Instance Details - Comments Tab

(Request A Day Off Example in Action, Cont'd)

The Transition tab and the Reassign tab display transition and reassignment choices to the user respectively. Both options are determined by the workflow definition. Figure B52 and Figure B53 show each tab. Jimi has two transition options and zero reassignment options.

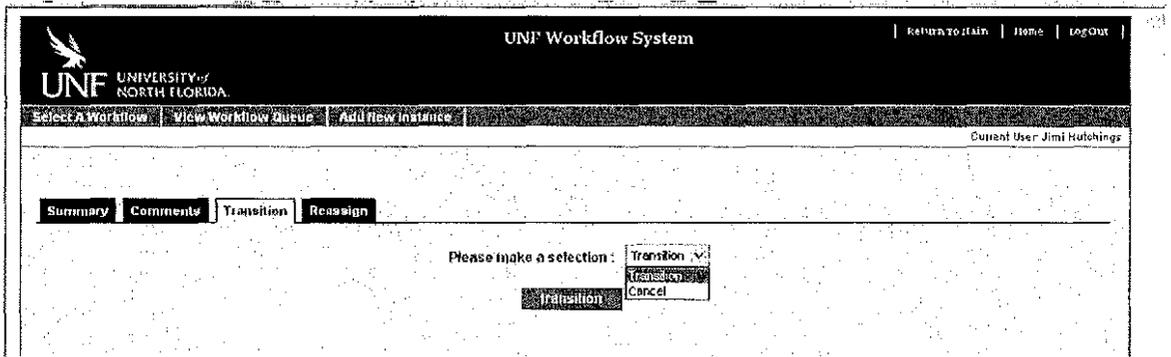


Figure B52: View Instance Details - Transition Tab

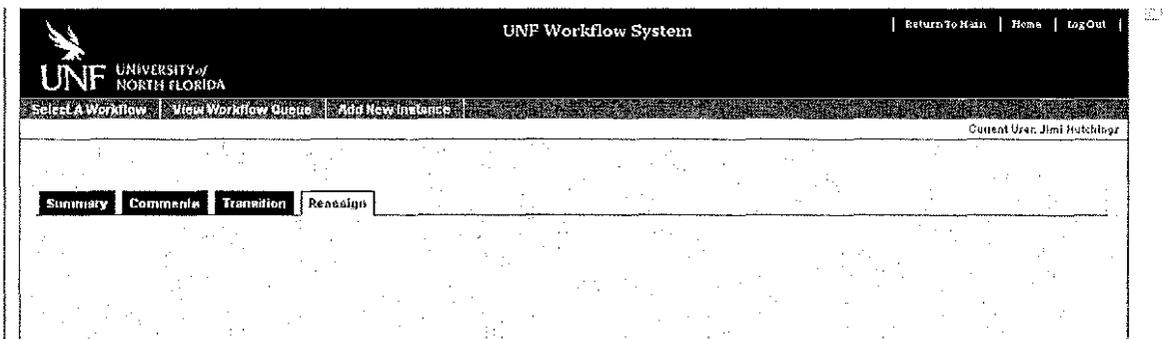
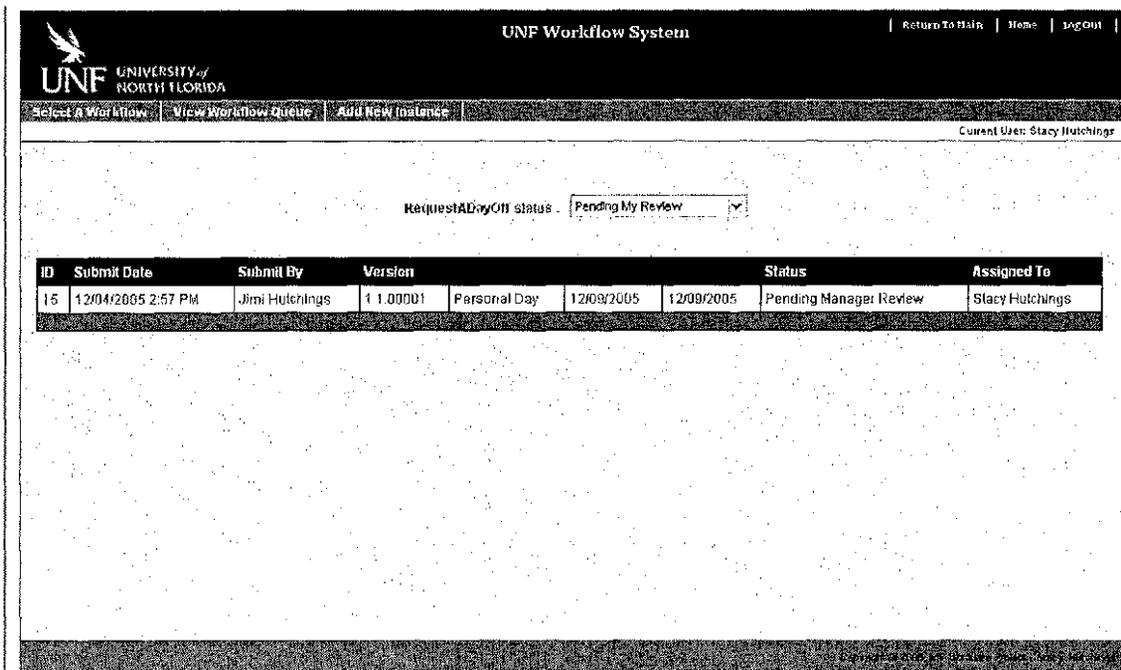


Figure B53: View Instance Details - Reassign Tab

(Request A Day Off Example in Action, Cont'd)

When Jimi is ready to send the request to his Manager, he will select the transition tab, select the transition option from the drop down list, then select the transition button. The request will be removed from Jimi's list of pending tasks and appear as a pending task in his manager's queue. Because the Auto-Look Up assignment strategy was selected for this transition, the system knows to automatically assign the task to Stacy Hutchings, who is listed as the only person for the Manager role. Figure B54 shows Stacy Hutchings' pending queue after the request has been transitioned.



The screenshot displays the UNF Workflow System interface. At the top, the UNF logo and 'UNIVERSITY of NORTH FLORIDA' are visible. The page title is 'UNF Workflow System'. Navigation links include 'Return to Main', 'Home', and 'Logout'. Below the header, there are buttons for 'Select a Workflow', 'View Workflow Queue', and 'Add New Instance'. The current user is identified as 'Stacy Hutchings'. A dropdown menu for 'RequestADayOff status' is set to 'Pending My Review'. A table below shows a single pending task:

ID	Submit Date	Submit By	Version	Status			Assigned To	
15	12/04/2005 2:57 PM	Jimi Hutchings	1.00001	Personal Day	12/08/2005	12/09/2005	Pending Manager Review	Stacy Hutchings

Figure B54: Pending Tasks for Stacy Hutchings

(Request A Day Off Example in Action, Cont'd)

When Stacy views the request, she sees the details of the request. The Comments tab in Figure B55 shows the transition history. The Transition tab in Figure B56 presents her with a new list of transition decisions. The Manager has the option to Approve or Deny the request.

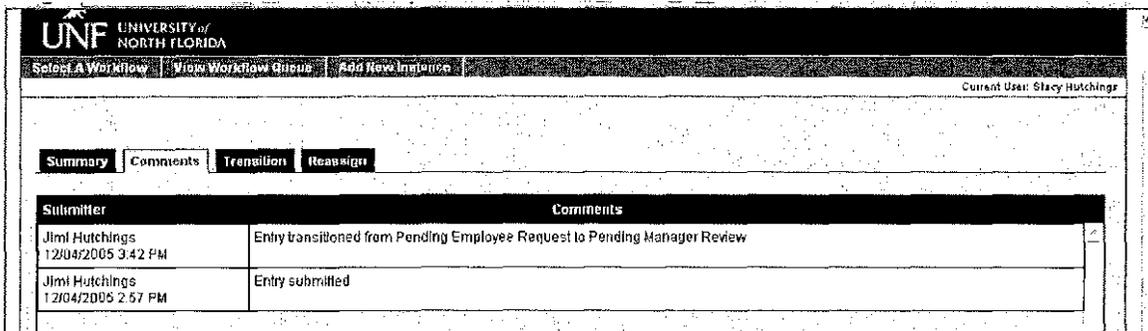


Figure B55: View Instance Details - Comments Tab

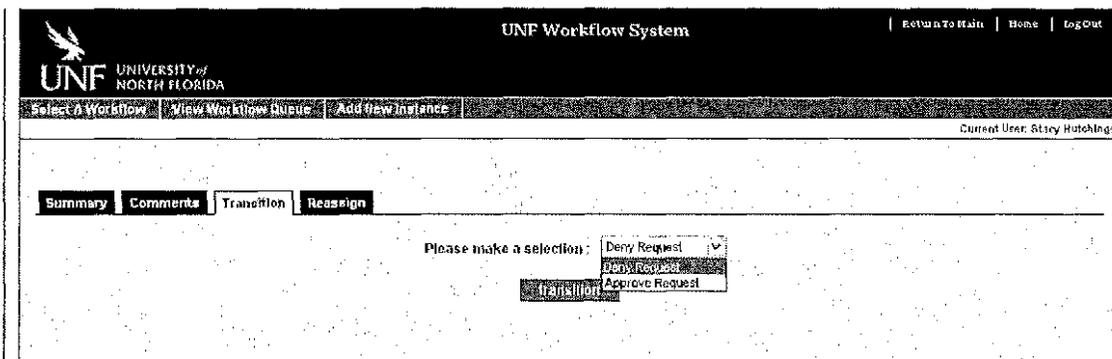


Figure B56: View Instance Details - Transition Tab

(Request A Day Off Example in Action, Cont'd)

For this request, Stacy will select the Approve Request option then select the transition button. The request transitions automatically to the Closed state. Figure B57 shows the tasks listed in the Closed queue. Figure B58 shows the Comments tab when you view details on the request.

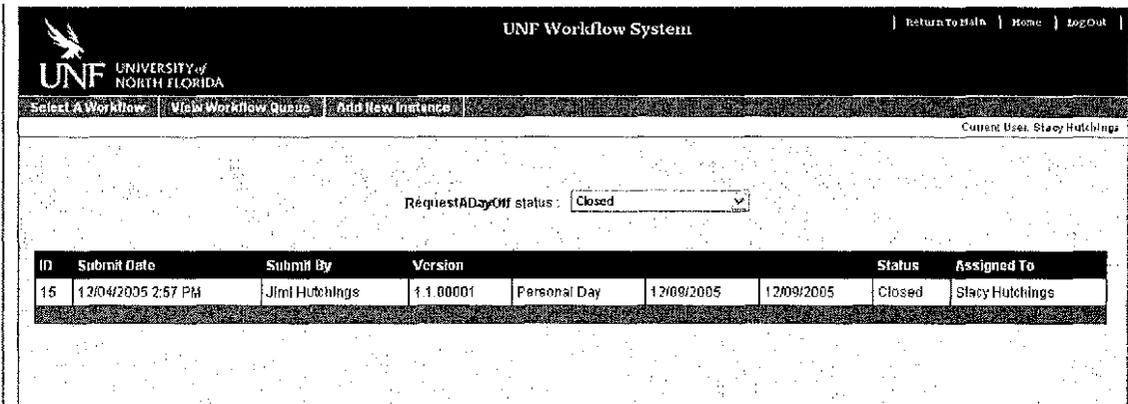


Figure B57: View Workflow Queue - Closed

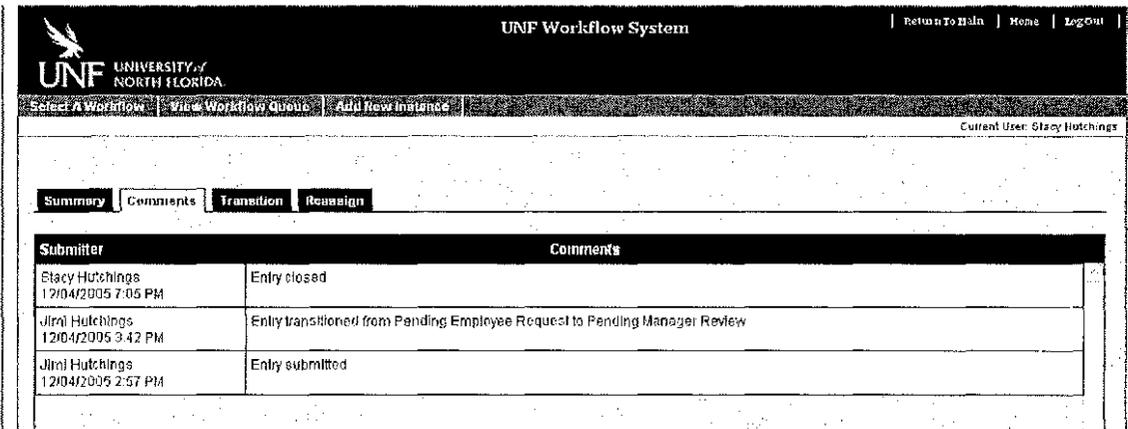


Figure B58: View Instance Details - Comments Tab

(Request A Day Off Example in Action, Cont'd)

To demonstrate what happens in the event the request is denied, Caitlyn Hutchings will initiate a request for the same day off. She will select the Add New Instance option, enter the details for the request, and when she is satisfied with her request, she will transition it to her manager. Figure B59 shows the validation error that occurs when a transition rule fails. In this example, Caitlyn has forgotten to select the type of day off for her request. The request will not transition until all errors are corrected.

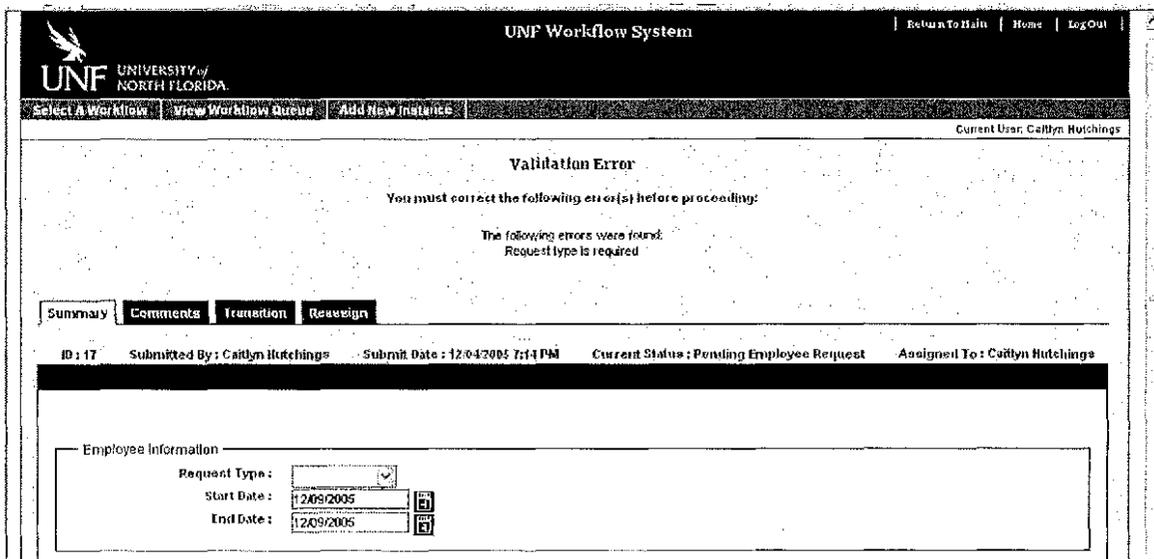


Figure B59: View Instance Details - Validation Error

(Request A Day Off Example in Action, Cont'd)

Figure B60 shows the request has transitioned successfully to the Manager role. When Caitlyn views her request, no longer in her pending list of tasks, the system reminds her the request is in View Only Mode and removes the ability to update, transition, and reassign the request.

The screenshot displays the UNF Workflow System interface. At the top, the UNF University of North Florida logo is on the left, and navigation links for 'Return To Main', 'Home', and 'Logout' are on the right. Below the header, there are tabs for 'Select A Workflow', 'View Workflow Queue', and 'Add New Instance'. The current user is identified as 'Caitlyn Hutchings'. The main content area is titled 'View Only Mode' and contains a 'RequestADayOff - Day Off Request Form'. The form is divided into two sections: 'Employee Information' and 'Manager Information'. The 'Employee Information' section includes fields for 'Request Type' (set to 'Personal Day'), 'Start Date' (12/09/2005), and 'End Date' (12/09/2005). The 'Manager Information' section includes a 'Reason For Denial' field. The form is currently in 'View Only Mode', as indicated by the text at the top of the content area.

RequestADayOff - Day Off Request Form	
<b>Employee Information</b>	
Request Type:	Personal Day
Start Date:	12/09/2005
End Date:	12/09/2005
<b>Manager Information</b>	
Reason For Denial:	

Figure B60: View Instance Details -View Only Mode

(Request A Day Off Example in Action, Cont'd)

The new request appears in the Manager's list of pending tasks shown in Figure B61. For this example, the Manager is going to select the Deny Request option then select the transition option without entering comments. Figure B62 shows the error message displayed.

The screenshot shows the UNF Workflow System interface. At the top, there is a navigation bar with 'Return To Main', 'Home', and 'Log Out'. Below this is a header for 'UNF UNIVERSITY of NORTH FLORIDA' and a menu with 'Select A Workflow', 'View Workflow Queue', and 'Add New Instance'. The current user is identified as 'Stacy Hutchings'. The main content area shows a 'RequestADayOff status' dropdown menu set to 'Pending My Review'. Below this is a table of pending tasks.

ID	Submit Date	Submit By	Version	Status	Assigned To
17	12/04/2005 7:14 PM	Caitlyn Hutchings	1.1.00001	Personal Day	12/09/2005 12/09/2005 Pending Manager Review Stacy Hutchings

Figure B61: Pending Tasks for Stacy Hutchings

The screenshot shows the UNF Workflow System interface displaying a validation error. The error message states: 'Validation Error. You must correct the following error(s) before proceeding: The following errors were found: Comments are required.' Below the error message are tabs for 'Summary', 'Comments', 'Transition', and 'Reassign'. The 'Summary' tab is selected, showing details for ID: 17, Submitted By: Caitlyn Hutchings, Submit Date: 12/04/2005 7:14 PM, Current Status: Pending Manager Review, and Assigned To: Stacy Hutchings. The 'Employee Information' section shows 'Request Type: Personal Day', 'Start Date: 12/09/2005', and 'End Date: 12/09/2005'. The 'Manager Information' section shows 'Reason For Denial' with an empty text area.

Figure B62: The UNF System - View Instance Details

(Request A Day Off Example in Action, Cont'd)

After the Manager fills in the reason for denial, the system presents the Manager with a list of participants for assignment. The Manager would like to return the request to Caitlyn so she selects her name from the list then selects the assign button, see Figure B63.

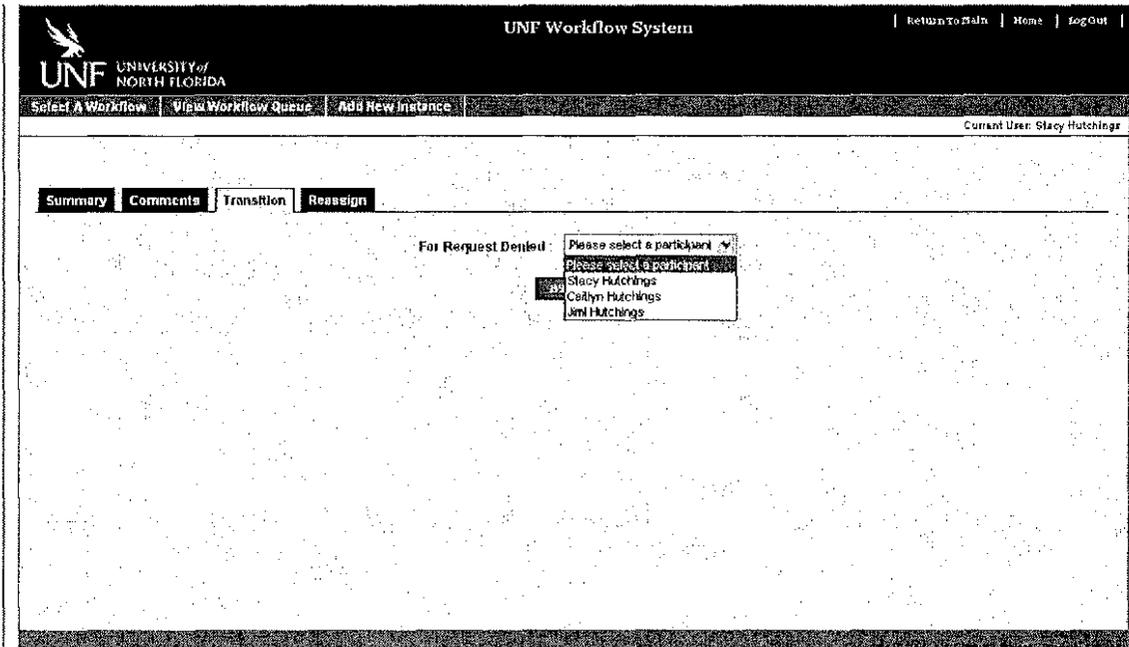


Figure B63: Transition Tab: Select Participant

(Request A Day Off Example in Action, Cont'd)

After denying the request, the request returns to Caitlyn's pending task queue with a status of Request Denied shown in Figure B64. Caitlyn has the ability to view the request and the reason for denial. Figure B65 shows the Transition tab with a single transition option of closing the instance.

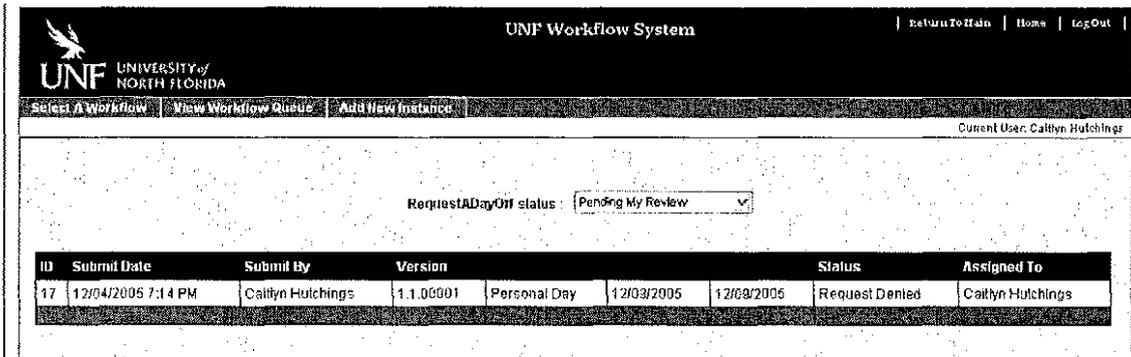


Figure B64: Pending Tasks for Caitlyn Hutchings

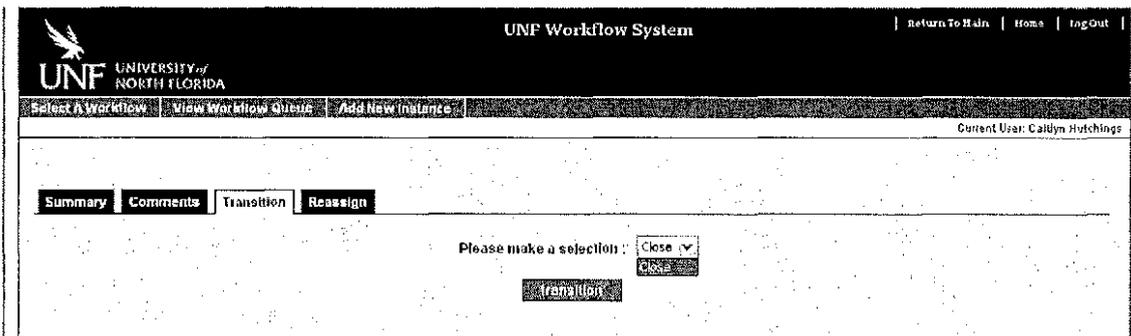


Figure B65: View Instance Details - Transition Tab

(Request A Day Off Example in Action, Cont'd)

Figure B66 shows both requests listed in the Closed task queue.

The purpose of the documentation was to demonstrate each of the components of the Workflow Generator and Tracking System and to provide a working example of how an end user would approach the task of creating and implementing a workflow.

The screenshot displays the UNF Workflow System interface. At the top, the UNF logo and 'UNIVERSITY of NORTH FLORIDA' are visible. The page title is 'UNF Workflow System'. Navigation links include 'Return To Main', 'Home', and 'Log Out'. Below the header, there are three buttons: 'Select A Workflow', 'View Workflow Queue', and 'Add New Instance'. The current user is identified as 'Callyn Hutchings'. A dropdown menu for 'Request A Day Off status' is set to 'Closed'. Below this, a table lists workflow instances:

ID	Submit Date	Submit By	Version				Status	Assigned To
15	12/04/2005 2:57 PM	Jimi Hutchings	1.1.00001	Personal Day	12/08/2005	12/09/2005	Closed	Stacy Hutchings
17	12/04/2005 7:14 PM	Callyn Hutchings	1.1.00001	Personal Day	12/09/2005	12/09/2005	Closed	Callyn Hutchings

Figure B66: The UNF System - View Workflow Queue Closed

## VITA

Stacy Hutchings has a Bachelor of Arts degree from Flagler College in Secondary Mathematics Education, 1992, and expects to receive a Master of Science in Computer and Information Sciences from the University of North Florida, December 2005. Dr. Arturo Sánchez-Ruíz of the University of North Florida served as Stacy's project director. Stacy is currently employed as a Senior Business Analyst at PHH Mortgage Company doing .NET web development and application support for the company's internal departments. Stacy has been in her current position for only three months, but with the company for 4 years.

Stacy has on-going interests in both .NET and J2EE web development. Stacy's academic work has included the use of Java, COBOL, and SQL. Married for the last 11 years, Stacy has two children, ages 8 and 9.