DESIGN AND IMPLEMENTATION OF DDISMEDIA: AN ONLINE DIGITAL MEDIA LIBRARY FOR PEST DIAGNOSIS

By

SHIWEI ZHANG

A THESIS PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING

UNIVERSITY OF FLORIDA

2008

To my family

ACKNOWLEDGEMENT

I express my sincerest gratitude to the members of my supervisory committee, Dr. Fedro Zazueta, Dr. Tim Momol, Dr. Jiannong Xin, and Dr. Thomas Burks, for their valuable advice, guidance, and support in the past three years.

I thank Dr. Zazueta, my advisor, for teaching me research skills, and all his efforts to solve my problems promptly in achieving my master's degree. I thank Dr. Momol for supporting me my research assistantship, and his valuable feedbacks and comments for this project. I am also grateful to Dr. Xin for his great input in my work. He taught me new information technologies and guided me throughout my work. It was lucky for me to work with Dr. Xin for three years. I also thank Dr. Burks for his understanding and support.

Special thanks go to my parents for their tremendous love and support in my life. Finally, my heartfelt appreciation goes to my wife, Nan Feng, for her love, help, encouragement, and accompaniment. Without them, I could not achieve this goal.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract of Thesis Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Master of Engineering

DESIGN AND IMPLEMENTATION OF DDISMEDIA: AN ONLINE DIGITAL MEDIA LIBRARY FOR PEST DIAGNOSIS

By

Shiwei Zhang

December 2008

Chair: Fedro Zazueta
Cochair: Tim Momol
Major: Agricultural and Biological Engineering

A Distance Diagnostic and Identification System (DDIS) was developed at the University of Florida to assist in pest diagnosis. Since its original deployment in 1999, thousands of digital pest photos have been collected and catalogued in DDIS. Our study involves DDISMedia, a user-friendly rich Internet digital media library application developed to assist specialists in pest diagnosis. DDISMedia, coupled with the DDIS diagnosis environment, provides a new tool for specialists to improve diagnosis and identification of pests. The DDISMedia has a peer-reviewed media database, which contains a collection of digital media produced by specialists of different disciplines including plants, plant insects and diseases. This system allows registered users to submit pest images or video clips into the database. Techniques used in implementing DDISMedia include a peer review process, library management, and extensive search functions. The media library of the DDISMedia could also be used for research, teaching/learning, and educational programs.

CHAPTER 1
INTRODUCTION

1.1 Background

The web-based Distance Diagnostic and Identification System (DDIS) was developed to provide a unique environment to conduct pest diagnosis through the Internet. Since its first release in 1999, DDIS has served as an effective tool to assist the collaboration and communication among first detectors, extension specialists and diagnosticians to share information on plant insects and diseases. This system uses field data and digital media as primary sources to facilitate the diagnosis and identification of plant disease, insect, weed, invasive species, plant management, and livestock (Xin et al., 1998; 2003).

The primary information transferred from agricultural extension agents to diagnosticians contains massive photographic images. Therefore, thousands of these photographic images have been collected since 1999, serving as an archived database for research, educational programs, and teaching/learning purpose in diagnosing plant diseases (Holmes et al., 2000). However, the following defects in the current system interfere with the management and usability of the photographic images. First of all, it lacks effective image management component. For instance, inconsistency of image quality caused by the fact that the majority of the photos were taken by extension agents, not a professional photographer in laboratories of agricultural fields, usually damages further application of DDIS in the field of research and educational programs. The second defect derives from the fact that this system does not have the capability of direct image searching. In DDIS, images are by-products of DDIS samples, which means users can not obtain images unless they browse a DDIS sample. In addition, DDIS samples are only available to approved registered DDIS users. Agricultural extension clientele and other external users who are interested in getting information from DDIS may not be able to access the system.

Because of defects of the current DDIS discussed above and the feedbacks from system managers and users, it is desirable to develop a digital media library in DDIS to serve as an important supplement of DDIS. This media library would include high quality video clips and/or photos of pests, plants, and invasive species that are submitted from any registered users worldwide. To ensure the media are described correctly and completely by their corresponding text information, an accurate and expertise-based assessment process has to be performed. Once media are approved after the assessment, their can be retrieved and browsed by the public. In addition, a powerful direct searching functionality is also a necessity to help users obtain their inquiries efficiently. To solve the problems discussed above, an online digital media library: DDISMedia is designed and developed in this research. DDISMedia will be embedded into DDIS and eventually serve as an important resource for DDIS in pest diagnosis.

1.2 Existing Plant and Pest Digital Libraries

Before the design of the DDISMedia, other existing online digital image libraries were visited and explored. Forestry Images and PDIS Library are two representatives of the existing digital image libraries, and some features of these two libraries are worth using for references in the design of DDISMedia.

1.2.1 Forestry Images

Forestry Images provides an archive of high quality images related to forest health and silviculture, with particular emphasis on educational applications. In most cases, images shown in this system were taken by outside photographers.

This system mainly collects images from forest pests, trees and plants, silviculture, invasive species, and wild life. Currently, this system contains approximately 90,000 images in about 12,000 subjects. Moreover, one feature of this system is to allow public users to search images based on different options, such as location, subject, and photographer. However,

invariably errors may occur during the image submission process due to the fact that only the submitter has the responsibility to ensure the text information describing the image is accurate. In other words, the Forestry Images system lacks quality control procedure for the submitted images.

1.2.2 PDIS Library

PDIS Library is designed to provide a large database of plant diagnostic related images that faculty, students, and government agencies can use for publications, presentations, and reports. This image library is searchable by: taxonomic serial number, common name synonyms, scientific name synonyms, disease name, pest name, photographer, submitting institution, county and state, description, and image size. In this system, each image is approved by specialists and classified using taxonomic and common name synonyms. Only individuals in the PDIS participating institution can submit images into the library.

Like Forestry Images, PDIS Library does not support video clips either, and only allows single image to be uploaded in one submission.

CHAPTER 2
SYSTEM OVERVIEW

2.1 Requirement Analysis

Before the design of DDISMedia, potential features and requirements for this system have been discussed and determined.

First of all, as a web application in the area of agriculture and biology, DDISMedia would likely face users without much computer and internet knowledge. Thus, a well-designed graphic user interface (GUI) is one of the key factors in the system design. The following three aspects are considered in the design of the GUI, (1). Each page should emphasize one topic to minimize possible distraction and confusion for users, (2). Operation flow of the system should be straightforward and navigate users to their destinations easily, (3). Online Help with conceptual or background information will be added to the user interface.

Secondly, the media submission process of the DDISMedia should be user-friendly and reliable. On one hand, the submission interface should be simple to use. Complicated user interface may confuse and discourage people from using the system. Moreover, the DDISMedia should be designed to minimize user input required for the submission process. On the other hand, a reliable file upload module should be utilized in the DDISMedia. As discussed in Chapter 1, large size files, such as video clips and photographic images, are major components in user submission; therefore, a reliable upload module will likely shorten the submission time and minimize the failure possibility of submission.

Thirdly, it is also necessary to implement a method to ensure the high quality of media in DDISMedia. High quality means high-resolution of electronic media, and accurate and complete text information corresponds to the media, such as sample common name, scientific name, and family. The DDISMedia is designed as an important supplement and references for DDIS in pest

diagnosis, and will also be used as a reliable resource for research and educational programs in the future, thus, high quality of the media is a key factor for DDISMedia to success. Chapter 3 will discuss the methodology of media quality control.

The ability to perform extensive search is the fourth requirement for DDISMedia, and the following criteria are considered, (1). The search engine should be able to retrieve data in a short time and have the intelligence to give a result set highly related to users' inquiries, (2). Given the same search intelligence, the user interface for searching should be easy to use, (3). The display of search images should be well-organized and provide user concise text information. The search function of DDISMedia will be open to public; therefore, its quality will greatly affect the popularity of DDISMedia.

Last but not the least, since DDISMedia will eventually serve as an important resource for DDIS in pest diagnosis, a user interface should be implemented in DDIS diagnosis page to direct diagnosticians into DDISMedia to perform related search.

2.2 System Design

This section is an overview of the design of DDISMedia based on the requirements stated in the previous section.

Figure 2-1 gives an integral structure of DDISMedia, which includes four major components: (1) Media Submit, (2) Media Review, (3) Media Search/View, and (4) Assisted Diagnosis.

Any registered user can submit into DDISMedia via Media Submit. The registration is open to the general public. Users in DDIS, IPDN (International Pest Diagnosis Network) and CRDN (Caribbean Regional Diagnosis Network) are automatically granted submission right into DDISMedia. File uploading, as a key point in Media Submit, uses the Apache FileUpload package (Section 2.3.5) to ensure reliability and stability. Although submitters are allowed to

decide at will the number of media to be uploaded per time, it is strongly recommended to limit the total size of one uploading to 50 MB in DDISMedia. Besides, most text fields, except personal information, are optional in this component in order to speed the submission process. All missing or incorrect information will be rectified in the next component: Media Review.

The Media Review component is used to ensure the quality of media library. Each new submission must go through a peer review process, where the editor-in-chief, editors, and reviewers are responsible for the quality of media and also make sure the text information associated with the media is accurate and complete before its acceptance. If an error is found in the text information, the editor-in-chief/editor will make revision based on reviewers' recommendations. The Media Review component also provides a rich user interface that allows the editor-in-chief, editors and reviewers to interact during the review process. Only accepted media are ready to be retrieved in the following Media Search/View component.

Another significant component in DDISMedia is Media Search/View, which is further divided into two parts: search module and browse module. The search module provides five search options: (1) search by keyword, (2) advanced search, (3) browse all, (4) associated search, and (5) tag-based search. Keyword search is a generic inexact-match search based upon user-ranked keywords. Advanced search allows user to select different terms to specify their search criteria. Browse all enables user browse square images of all accepted media in database. Associated search option enables user search related samples to current search result through genius, species, family, media submitter or photographer as associated fields. Tag-based search extracts and ranks popular tags in current database and allows user to search based on each tag. The browse module retrieves complete information of one submission and provides a user-friendly platform for user to browse media and its associated text information.

The Assisted Diagnosis component connects DDIS and DDISMedia, and provides DDIS diagnosticians a functionality to use the digital media library as a reference tool during their pest diagnosis.

Other components in DDISMedia include database connection component, by which all other components communicate with database server, and suggested fill component that provides users input suggestions according to their inputs.

2.3 Technologies and Tools

To implement functions and components discussed in the previous section, the following technologies and tools were studied and utilized in the development of DDISMedia, including: HTML, JavaScript, Java Servlet, JSP, Apache FileUpload package, JDBC, SQL, and Ajax.

2.3.1 HTML

HTML (HyperText Markup Language) is the primary markup language for Web pages. It describes the structure of text-based information in a document by denoting certain text as links, headings, paragraphs, lists, and so on, and to supplement that text with interactive forms, embedded images, and other objects. HTML is written in the form of tags, surrounded by angle brackets. HTML can also describe, to some degree, the appearance and semantics of a document, and can include embedded scripting language code (such as JavaScript introduced below) which can affect the behavior of web browsers and other HTML processors (Musciano et al., 2007; Willard, 2007).

HTML markup consists of several key components, including elements and their attributes, character-based data types, and character references and entity references. A simple example is given out below:

```
<html>
<head>
```

```
<title>DDISMedia</title>
</head>
<body>
<p>Welcome to DDISMedia!</p>
</body>
</html>
```

These eight lines of HTML code displays text “Welcome to DDISMedia!” on the web browser within the page titled “DDISMedia.” HTML is the mostly used technology in the programming on client-side in DDISMedia.

2.3.2 JavaScript

JavaScript is a scripting language often used for client-side web development. It is a dynamic, weakly typed, prototype-based language, it also supports constructing new functions during the execution of a program, storing them in data structures, passing them as arguments to other functions, and returning them as the values of other functions (Goodman et al., 2004; Flanagan, 2006).

The primary use of JavaScript is to write functions that are embedded in or included from HTML pages and interact with elements of the page. Some simple examples of this usage are:

- Popping up a dialogue to notify user an error message.
- Validating web form input values to ensure that they will be accepted before they are submitted to the server.
- Changing text size or color as the mouse cursor moves over them: this effect is often used to attract the user's attention to important contents on the page.

JavaScript code can run locally in a user's browser and can respond to user actions quickly, making an application feel more responsive. JavaScript code can also detect user actions which HTML alone cannot, such as individual keystrokes. Therefore, JavaScript is used in the

development of DDISMedia to help interact between users and pages and reduce burden of the server.

2.3.3 Java Servlet

A Servlet is an object that receives a request and generates a response based on that request. The basic servlet package defines objects representing servlet requests and responses, as well as objects reflecting the servlet's configuration parameters and execution environment. It contains package defining Hypertext Transfer Protocol (HTTP)-specific subclasses of the generic servlet elements, including session management objects that track multiple requests and responses between the Web server and a client.

Java Servlet Application Programming Interface (API) allows a software developer to add dynamic content to a Web server using Java platform. Java Servlet defines the expected interactions of a Web container and a servlet. A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a Uniform Resource Locator (URL) to a particular servlet and ensuring that the URL requester has the correct access rights (Hunter et al., 2001; Perry, 2004; Hurniawan, 2002).

Usually Java Servlet is used in conjunction with JavaServer Pages (JSPs), which is introduced below. In the implementation of DDISMedia, such conjunction is widely used: Java Servlet API is used in server-side programming, while JSP is used to pass parameters values between the server and clients.

2.3.4 JSP

JSP is a Java technology that allows software developers to dynamically generate HTML or other types of documents in response to a Web client request. The technology allows Java code and certain pre-defined actions to be embedded into static content, such as HTML codes

(Perry, 2004; Bergsten, 2003). In DDISMedia, this technology is used to pass values of parameters between the server and clients.

2.3.5 Apache FileUpload

The Apache FileUpload package is an open-source and robust library handling file uploads and is developed and maintained by the Apache Software Foundation. FileUpload creates new file items using a FileItemFactory, which gives FileUpload most of its flexibility. The factory has ultimate control over how each item is created. The factory implementation that currently ships with FileUpload stores the item's data in memory or on disk, depending on the size of the item. Due to its reliability and stability in JSP/Java Servlet environment, Apache FileUpload package is selected as the file uploading tool for DDISMedia.

2.3.6 SQL

Structured Query Language (SQL) is a database computer language designed for the retrieval and management of data in relational database management systems (RDBMS), database schema creation and modification, and database object access control management (Ramakrishnan et al., 2002; Groff, 2002).

Being a standard interactive and programming language for querying and modifying data and managing databases, SQL is mainly used to form commands that allow the retrieval, insertion, updating, and deletion of data in DDISMedia.

2.3.7 JDBC

Java Database Connectivity (JDBC) is an API for the Java programming language that defines how a client may access a database. It provides methods for querying and updating data in a database. JDBC is oriented towards relational databases.

JDBC allows multiple implementations to exist and be used by the same application. JDBC provides a mechanism for dynamically loading the correct Java packages and registering

them with the JDBC Driver Manager. The Driver Manager is used as a connection factory for creating JDBC connections.

JDBC connections support creating and executing statements, which may be update statements such as SQL's CREATE, INSERT, UPDATE and DELETE, or query statements such as SELECT. Additionally, stored procedures may be invoked through a JDBC connection. JDBC API is used in DDISMedia to operate database connection.

2.3.8 Ajax

Asynchronous JavaScript and XML (Ajax), is a group of interrelated web development techniques used for creating interactive web applications or rich Internet applications. With Ajax, web applications can retrieve data from the server asynchronously in the background without interfering with the display and behavior of the existing page.

In many cases, different pages on a website share common contents. Using traditional methods, these common contents would have to be reloaded on every request. However, using Ajax, the common contents do not have to be updated on every request, which consequently reduces bandwidth usage and load time. The use of Ajax also allows the client's Web browser user interface to be more interactive and to respond quickly to inputs, and also allows sections of pages to be reloaded individually.

However, with the application of Ajax, an obvious problem may occur. If a user’s browser does not support Ajax or JavaScript, or simply has JavaScript disabled, Ajax module will lose its functionality. In addition, the lack of a standard body behind Ajax leads to the absence of a best practice to test Ajax applications (Crane et al., 2005; Mahemoff, 2006; Paulson, 2005).

In DDISMedia, Ajax is mainly used to provide input suggestions and pop up online “Help” and enhance user rich Internet experience.

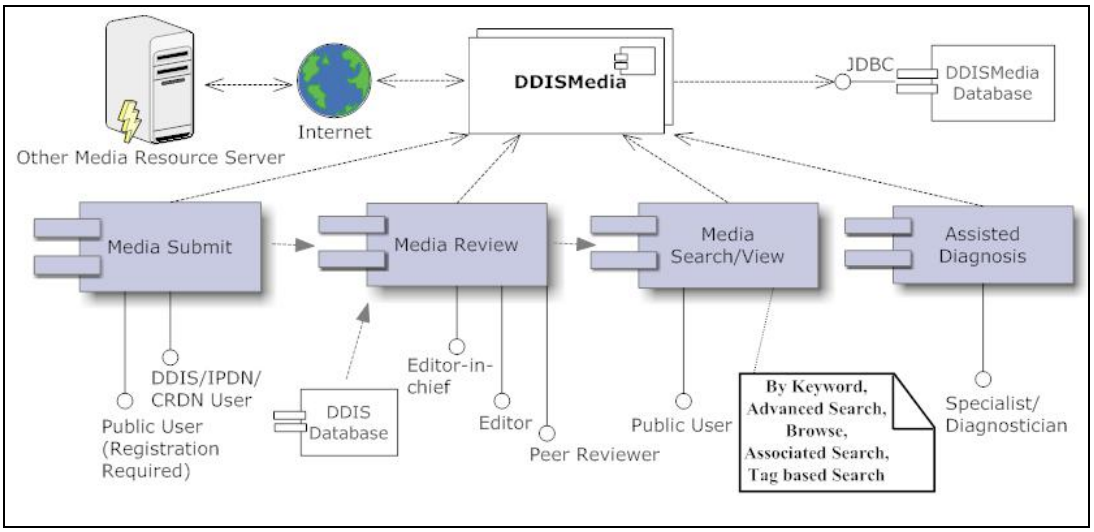

Figure 2-1. Integral structure of DDISMedia

CHAPTER 3
SYSTEM IMPLEMANTATION

This chapter discusses the implementation of three major components in DDISMedia, including Media Submit, Media Review, and Media Search/View.

3.1 Media Submit

The Media Submit component collects information for DDISMedia and save them into database. Media Submit is open to any registered users and registration is open to the public. Registered users in DDIS, IPDN, and CRDN are automatically granted submission rights to DDISMedia.

Submitted information is usually divided into two parts: text information and digital samples. In one submission, users are allowed to provide one copy of text information and upload as many digital samples as they will. But the total size of one submission is strongly recommended to be limited to 50 MB.

3.1.1 Data Schema

In the database level, there are two tables storing submitted information. One table titled "img_lib" stores text information for each submission. Table 3-1 lists the database schema of this table.

Attribute "LibID" is the library ID and each submission is assigned a unique library ID once the submission enters the database. This unique ID is currently composed of eight characters: two numeric characters representing submission year, one dash character, and five random numeric characters created by a Java program on server-side.

Attributes "CommonName", "Genus", "Species", and "SubSpecies" represent the common name, genus, species, and sub species of the sample in a submission, respectively. Genus, species, and sub species constitute the scientific name of the sample. Similarly, "HostPlant",

“HostGenus”, “HostSpecies”, and “HostSubSpecies” represent common name and scientific name of a host.

Attribute “SampleType” is a required field for each submission. Same as in DDIS, six sample types exist in DDISMedia, and they are: (1) plant disease, (2) insect (plant), (3) insect (non-plant), (4) plant/weed, (5) management/physiology/nutrient, and (6) livestock.

Attribute “Family” specifies the family to which a sample belongs. It might not be an applicable attribute since some sample types do not have a family.

Attributes “County”, “City”, “State”, “Country”, “Longitude”, and “Latitude” are all geographical attributes which represent the location of samples. These attributes could be used in sample distribution map, which is a function in DDIS, and also for statistical purpose.

Attribute “CollectionDate” is the date when the sample was collected, and is a required field in submission interface.

Attribute “AdditionalDscp” allows submitter to add additional descriptions to the sample.

Attributes “PhotographerName”, “PhotographerInstitute”, and “PhotographerEmail” save personal information of the author of digital samples.

Attribute “SubmitDate” files the date when a sample is submitted.

Attribute “SubmitBy” refers to the user ID of a submitter. This user ID is either from DDISMedia, DDIS, IPDN or CRDN.

Attribute “Status” indicates the current status of a sample in the library. The default status value is “0” meaning the submission is new and waiting for a peer review process that (Section 3.2). Status “1” means the sample has been reviewed and approved to be accepted. Status “2” denotes the sample is in review. If a sample has a status “3”, it means the sample has been

reviewed but rejected. Only samples with status “1” are ready to be retrieved and browsed by public users.

Attributes “EditorUpdateDate”, “EditorBy”, “EditorDecision”, and “EditorComments” are four attributes used by Media Review component (Section 3.2).

Attribute “CopyrightNote” allows submitter to claim copyright for their submissions.

In Media Submit, table “img_media” is used to store information for each digital media. Table 3-2 lists the data schema of Table “img_media.” Each attribute shown in Table 3-2 is described in details in the following.

Attribute “LibID” in Table 3-2 must be an existing library ID in Table “img_lib.” Text information in Table “img_lib” and media information in Table “img_media” for one submission share the same library ID. Library ID in Table 3-2 is not necessary to be unique, since it is possible that there are several digital media in one submission.

Attribute “ImgID” is a serial number assigned to each submitted media. Combination of this attribute and “LibID” above is treated as the key of Table 3-2.

Attribute “FileName” is the original name of the media file before being submitted to the database. Once a media is submitted to the database, it will be renamed based on the combination of “LibID” and “ImgID.”

Attribute “FileType” indicates the file type of the submitted media file.

Attribute “Media” stores binary data of submitted media file and serves as a backup in case any emergency occurs on the physical disk of media storing server.

3.1.2 Work Flow

When a submission request is received by Servlets, a Java program is called to generate a random string and this string is assigned to the submission as its unique ID, as introduced above. To ensure the uniqueness of the library ID in Table “img_lib”, validation is required. If the

generated ID exists in the current database, the process of ID generation will be repeated until a new unique ID obtained. An ID generation flow chart in Media Submit is shown in Figure 3-1.

Once a library ID is assigned to a submission, the Apache FileUpload package is used to obtain text information and media file list from HTML pages. By using this file upload package, media files are loaded into memory and are ready to be processed. At this point, Media Submit writes a serial number onto the image, generates square copy, thumbnail copy, and normal-sized copy of the original image, and finally writes these images (i.e., square copy, thumbnail copy, normal-sized copy, and original copy) onto physical disk of media storing server. In terms of a video clip, except for the above steps, Media Submit needs an additional step to capture a picture from the clip before image processing.

After media upload is completed, the text information and media information is written into the database. Considering the possibility that one submission may contain more than one media file, text information and media information are stored in two different tables in the database. All text information describing digital samples with unique ID is stored in the Table "img_lib", while media information is stored in the Table "img_media." Once a new submission is input into the database, its status is assigned value "0", which means the submission is new and waiting for peer review and approval.

3.2 Media Review

DDISMedia is a specialized pest and plant media database, where expert knowledge is required to not only ensure quality of the media, but, more importantly, verify completeness and correctness of text information associated with the media. Media Review is such a component designed to conduct this task. A procedure named peer review process is used.

3.2.1 User Role

Similar to the journal publication review process, media submitter, editor-in-chief, editor, and reviewer are participating in the peer review process.

The media submitter submits new submissions to the editor-in-chief.

The editor-in-chief takes in charge of assigning new submissions to editor or reviewer. A new submission always goes to the editor-in-chief first, who then makes arrangement according to the sample type of this submission. Editor-in-chief has the right to make a final decision of accepting or rejecting a submission. Generally speaking, only one editor-in-chief is assigned to the peer review process.

The editor is responsible for assigning submissions received from editor-in-chief to reviewer. Editor can also make a final decision of accepting or rejecting a submission. There are usually several editors within different specialties.

The reviewer is a specialist in a certain area and provides comments and recommendations to editor-in-chief and editor. The peer review process does not have a fixed reviewer group in DDISMedia. However, thanks to their professional knowledge, all DDIS specialists can potentially be selected as reviewers in this process.

3.2.2 Data Schema

In database level, three tables are used in the peer review process. Attributes “EditorUpdateDate”, “EditorBy”, “EditorDecision”, and “EditorComments” in Table “img_lib” (Section 3.1) are updated when a final decision is made.

Table “img_editor_process” stores editor’s actions in the peer review process. Its data schema is shown in Table 3-3. Each attribute shown in Table 3-3 is described in details in the following.

Attribute “UserID” represents editor’s user ID in DDIS system.

Attribute “LibID” is the library ID of submission in Table “img_lib.” Combination of “LibID” and “UserID” is treated as the key in this table.

Attribute “ReceiveDate” records the date when the submission is assigned to the editor.

Attribute “Decision” is an attribute representing a decision made by the editor. After a decision is made, one of the following three values is assigned to this attribute. Value “0” indicates “Accept”; value “1” means “Accept with revision”; value “2” represents “Reject.”

Attribute “DecisionDate” records the date when the decision is made. By default, the value of this field is “NULL”, which means editor has not made any decision.

Attribute “Status” is an attribute which indicates the status of a submission in editor’s peer review process. Value “0” means the submission is in editor’s “Unassigned” list and waiting for review and approval. Value “2” represents the submission is in editor’s “In Review” list.

Attribute “Comment” stores messages from editor to submitter when they make a decision.

Table “img_reviewer_process” is used to record comments and recommendations from reviewer. Table 3-4 shows the data schema of Table “img_reviewer_process.” Each attribute shown in Table 3-4 is described in detail in the following.

Attribute “AutoID” is a unique ID representing a record in this table. It is automatically generated by the database system.

Attribute “LibID” is a foreign key which can be found in table “img_lib.”

Attribute “UserID” is reviewer’s user ID in DDIS.

Attributes from “CommonName” through “Family” in Table 3-4 have the same meaning as their counterparts in Table “img_lib” (Section 3.1).

Attribute “Comments_Private” is the field where reviewer inputs comments to editor-in-chief/editor only. Attribute “Comments_Public” is the field for reviewer to store comments for both submitter and editor-in-chief/editor.

Attribute “ReviewerDecision” represents reviewer’s decision on the submission. This field by default is set to be “NULL”, which indicates reviewer has not conducted any review. One of the following three values is assigned when a decision is made by reviewer. Value “0” represents acceptance of the submission, value “1” means acceptance of the submission but revision is needed according to reviewer’s comments, value “2” indicates rejection of the submission.

Attributes “AssignDate” and “ReviewDate” record the dates when the submission is assigned and reviewed, respectively, for management purpose.

3.2.3 Work Flow

The peer review process mimics a journal publication review process. A technical editorial board including one editor-in-chief and several editors is formed first to manage the whole process. Figure 3-2 shows the peer review process in DDISMedia.

As shown in Figure 3-2, a new submission is first included in the “Unassigned” list of editor-in-chief with a status value of “0.” The editor-in-chief next assigns editors or reviewers to initiate a review process. Then the submission is placed into the editor-in-chief’s “In Review” list by changing its status from “0” to “2” in Table “img_lib.” At the same time, the access to the submission is authorized to the assignee. Emails are also sent out by the system to notify submitter and assignee the initiation of the peer review process.

If the submission is assigned to an editor from the editor-in-chief, a new record containing the library ID and editor’s user ID is inserted in Table “img_editor_process” with a status value of “0”, which means the submission is currently in editor’s “Unassigned” list. To assign a reviewer for the submission, a new record with the library ID and reviewer’s user ID is inserted

in Table “img_reviewer_process”, and the submission is then included in the editor’s “In Review” list. Emails are sent out to inform submitter, reviewer, and editor-in-chief that the submission has been assigned to reviewer.

If the submission is assigned to a reviewer from the editor-in-chief or editor, a new record containing the library ID and reviewer’s user ID is inserted in Table “img_reviewer_process.” The reviewer is able to see submissions to be reviewed in his/her working list. After a reviewer accomplishes his/her review process by providing comments and recommendations to the editor-in-chief/editor, data fields in Table “img_reviewer_process” are updated. Especially, attribute “ReviewerDecision” is assigned a non-null value indicating the submission is removed from the working list of reviewer. Emails are sent out to inform editor and editor-in-chief that the submission has been reviewed.

The last step of the peer review process is the decision-making process conducted by the editorial board. If a submission is accepted, attribute “Status” in Table “img_lib” is changed from “0” to “1.” If a submission is accepted with revision, text information of the submission should be revised based on the comments of the editorial board, and then the status of the submission is changed from “0” to “1.” If a submission is rejected, the status of the submission is changed from “0” to “2.” Once a decision is made by the editorial board, the submission is removed from the editor-in-chief/editor’s “In Review” list, and an email is sent out to notify submitter the decision on the submission.

3.3 Media Search/View

The Media Search/View component provides functionalities to public users to search and view submissions which have been accepted (with revision). This component includes two modules, Media Search and Media View. Media Search retrieves data that qualify search criteria, and Media View organizes and shows the search results.

3.3.1 Media Search: Extensive Search Options

Media Search module provides several search options with the aim of improving user experience and search accuracy in DDISMedia. After Media Search receives request from user, it calls the corresponding SQL store procedure to retrieve matching records. Extensive search options are listed as follows.

- **Keyword search**: This is a generic search based upon the keyword that user input. The keyword search procedure checks attributes “CommonName”, “Genus”, “Species”, “SubSpecies”, “HostPlant”, “HostGenus”, “HostSpecies”, “HostSubSpecies”, “Family”, “AdditionalDscp”, “SubmitBy” and “PhotographerName” in the Table “img_lib.” If a record in database has any attribute containing the keyword, this record is included in the result set.

- **ID search**: This option allows user retrieve the submission with exact library ID.

- **Advanced search**: User may select different terms to customize their search criteria. Search terms include common name, scientific name which is composed of genus, species and sub species, common name and scientific name of host plant, sample type, family, country, state and date range. If user specifies two or more search terms, the search results must meet all search requirements.

- **Browse all**: This option allows users to easily retrieve all accepted submissions in the library, and does not require user to set any search conditions.

- **Tag-based search**: Tag is defined as the keyword which appears in the common name field or common name of host plant field of one submission. Up to 100 tags with the highest frequency in database are dynamically extracted by the system at each time when user accesses this feature. A tag can be used as a keyword in the Keyword Search option.

3.3.2 Media Browser

Media Browser is a module which retrieves and shows complete information of one submission specified usually by a library ID. If a submission has been accepted (with revision), the sole task for Media Browser is to retrieve and exhibit data. If a submission is in the peer review process, extra work is needed before retrieving data: Media Browser checks if a user has access to the submission.

Table 3-1. Data schema of Table “img_lib”

Attribute name	Data type	Allow null
LibID	char(17)	Unchecked
CommonName	varchar(125)	Checked
Genus	varchar(75)	Checked
Species	varchar(75)	Checked
SubSpecies	varchar(75)	Checked
HostPlant	varchar(125)	Checked
HostGenus	varchar(75)	Checked
HostSpecies	varchar(75)	Checked
HostSubSpecies	varchar(75)	Checked
SampleType	char(10)	Checked
Family	varchar(75)	Checked
County	char(45)	Checked
City	char(45)	Checked
State	char(45)	Checked
Country	char(45)	Checked
Longitude	char(10)	Checked
Latitude	char(10)	Checked
CollectionDate	Datetime	Checked
AdditionalDscp	Ntext	Checked
PhotographerName	varchar(256)	Checked
PhotographerInstitute	varchar(85)	Checked
PhotographerEmail	varchar(50)	Checked
Status	Smallint	Unchecked
SubmitDate	Datetime	Checked
SubmitBy	char(12)	Checked
EditorUpdateDate	Datetime	Checked
EditorBy	char(12)	Checked
EditorDecision	char(1)	Checked
EditorComments	Ntext	Checked
CopyrightNote	Ntext	Checked

Table 3-2. Data schema of Table “img_media”

Attribute name	Data type	Allow null
LibID	char(17)	Unchecked
ImgID	char(2)	Unchecked
FileName	char(25)	Checked
FileType	char(35)	Checked
Media	image	Checked

Table 3-3. Data schema of Table “img_editor_process”

Attribute name	Data type	Allow null
UserID	char(12)	Unchecked
LibID	char(17)	Unchecked
ReceiveDate	datetime	Checked
Decision	smallint	Checked
DecisionDate	datetime	Checked
Status	smallint	Checked
Comment	ntext	Checked

Table 3-4. Data schema of Table “img_reviewer_process”

Attribute name	Data type	Allow null
AutoID	Int	Unchecked
LibID	Char(17)	Unchecked
UserID	Char(12)	Unchecked
CommonName	varchar(125)	Checked
Genus	varchar(75)	Checked
Species	varchar(75)	Checked
SubSpecies	varchar(75)	Checked
HostPlant	varchar(75)	Checked
HostGenus	varchar(75)	Checked
HostSpecies	varchar(75)	Checked
HostSubSpecies	varchar(75)	Checked
SampleType	Char(10)	Checked
Family	varchar(75)	Checked
Comments_Private	Ntext	Checked
Comments_Public	Ntext	Checked
ReviewerDecision	Smallint	Checked
AssignDate	Datetime	Checked
ReviewDate	Datetime	Checked

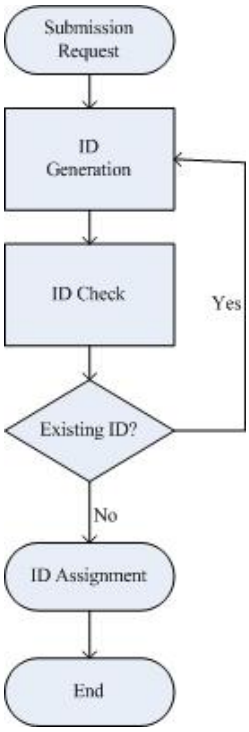

Figure 3-1. The ID generation flow chart in Media Submit

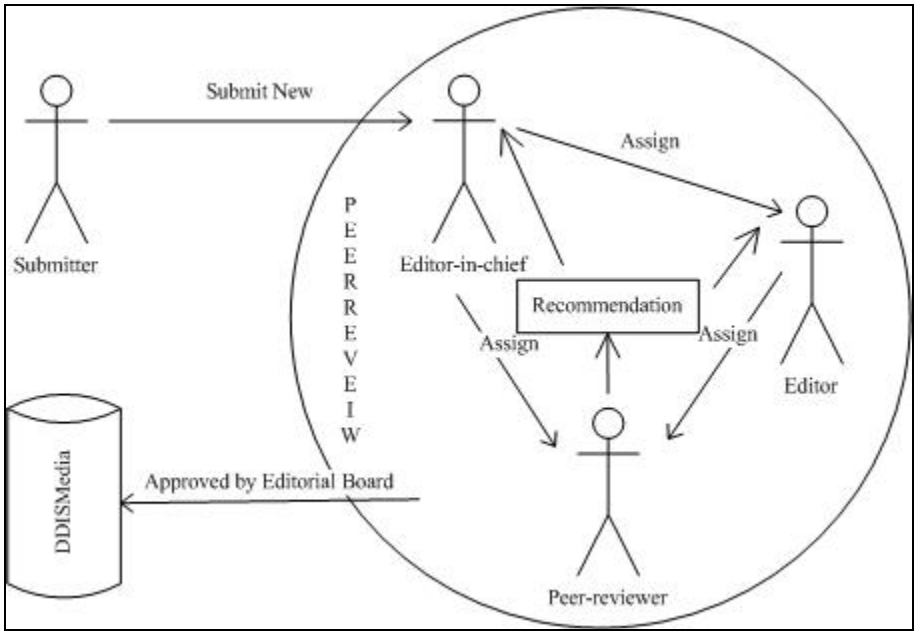

Figure 3-2. Peer review process in DDISMedia

CHAPTER 4
USER INTERFACE DESIGN

The user interface (UI) is the aggregate of means by which people interact with the system. A well designed UI makes the process of using the system effective, efficient and satisfying. In DDISMedia which is a web application, UI especially refers to web user interfaces (WUI). It accepts input and provides output by generating web pages which are transmitted via the Internet and viewed by the user using a web browser program. Some technologies such as JavaScript, Ajax, are utilized to provide real time interactions between users and pages, eliminating the need of refreshing a traditional HTML-based web browser. This chapter exhibits the user interface of each component in DDISMedia.

4.1 Media Submission UI

One design concept of DDISMedia is to provide a UI which enables quick submission for submitter. Only a small number of input fields in the submission are required. The submission form is composed of six parts: Sample Information form, Location form, Additional Description form, Copyright Notes form, Photographer/Video Producers form, and Upload Media form.

The Sample Information form collects sample information. As shown in Figure 4-1, common name and scientific name (including genus, species, and sub species) of pest, common name and scientific name of host plant, sample type, and family are input fields in this form. All fields in this form, expect the “Sample Type” field, are optional because the peer review process is conducted to ensure all information is complete after media submission. Sample type is required because editor-in-chief would assign submissions to corresponding editor according to its values. The value of sample type is in the following pre-defined categories: Plant Disease, Insect (Plant), Insect (Non-plant), Plant/Weed, Management/Physiology/Nutrient, and Livestock. Family filed is usually applicable for submissions whose sample type is insect or plant.

The Location form is shown in Figure 4-2. This form aims to collect sample geographic information such as country, state/province, city, county, longitude, and latitude. Country, State/Province, and Media Collection Date are three required fields. Clicking the calendar icon pops up a small window which helps user choose and input the media collection date. Information in this form is mainly for statistical purposes.

Considering the complexity and inconvenience of inputting longitude and latitude, DDISMedia provides a nice feature to aid user to find their geo-codes. When a user clicks link "Get GeoCode" in the Location form, a geo-code input panel shown in Figure 4-3 is popped up. This panel is implemented by embedding the Google Map API. On the left side of this panel, it is allowed to generate geo-code by clicking "Generate Geocode" button. Once an address in the "Location" section is verified by the Google Map API, a pair of longitude and latitude coordinates is automatically filled into the corresponding fields in the "GeoCode" section. On the right side of this panel where a Florida map is shown by default, users are able to drag the green star icon on the map to locate any place in the world. When the star icon is dragged, the geo-code information in the "GeoCode" section is updated accordingly. By clicking "Use this geocode and close this page" button, values in fields "Longitude" and "Latitude" are input into corresponding fields in Figure 4-2.

Both the Additional Description form and the Copyright Notes form include optional fields where user input additional description and copyright messages, respectively. The user interfaces for these two forms are shown in Figure 4-4.

Photographer/Video Producers form, which is shown in Figure 4-5, is the interface to gather photographer/video producers' personal information, such as name, institute, and email. If the submitter is the person who produces photos or video clips, these fields could be skipped by

checking the checkbox in this form: the system is able to complete related information for the user.

Figure 4-6 shows the Upload Media form. By clicking the “Browse…” button, a submitter can select media files from his/her local disk. By clicking link “Add” and “Delete”, a submitter is able to add and delete input fields, respectively. A submitter is allowed to decide the number of media in one submission at their will; however, it is strongly recommended to limit the total file size to 50 MB to avoid heavy burden on the network. DDISMedia only accepts image files in jpeg, gif or png format, and video files in mpg or wmv format.

Clicking the “Submit” button in Figure 4-6 is the last step in media submission. If any required information is missing or an invalid value exists in the submission form, a dialogue is popped up to inform submitter that error occurs and the submission is invalid. If a submission is processed successfully on the server, a message is returned to the client notifying the success of submission.

4.2 Input Suggestions for Pest/Host Names

It is a goal to design a user interface collecting consistent user input, since data inconsistency could jeopardize the search ability and library quality of DDISMedia. However, the length and complexity of the common name and the scientific name of the pest/host makes it easy for a user to input inconsistent names or invalid input. To solve this problem, input suggestions are provided to users when input characters are captured. Ajax, a newly developed technology which allows interaction between pages and users without refreshing, is used to accomplish this task. An example of Ajax-based input suggestions is shown in Figure 4-7.

As shown in Figure 4-7, immediately after the characters “or” are captured, a list of options starting with “or” appears. The user can select one of these options, and the selected string will appear in the “Pest Common Name” field. Options are extracted from a pre-defined

list containing over 16,000 names, which is maintained by the National Pest Diagnostic Network (NPDN). Due to the large number of names in this list, this input suggestion feature is enabled only when two or more characters are input.

4.3 Media Review UI

The user interfaces in Media Review component are composed of the following parts: Working List UI, Assignment UI, View Comments UI, Review UI, and Decision UI.

4.3.1 Working List UI

Working list is available to all three user groups (i.e. editor-in-chief, editor, and reviewer) involved in the peer review process. It is technically a table containing submissions waiting for assignment and/or review. Working list for editor-in-chief/editor is further divided into "Unassigned" list and "In Review" list. Figure 4-8 shows an example of "In Review" working list for editor-in-chief/editor. In this table, the first four columns containing basic information such as library ID, submitter's name, sample type, and submission date, are identical for all three user groups. The library ID is linked to the media viewer (Section 4.4). The "Action" column, however, is more complex. For editor-in-chief and editors, if it is in the "Unassigned" working list, only "Assign/View Reviewer" link is shown in "Action" column; if it is in the "In Review" working list, both "Assign/View Reviewer" and "View Reviewer's Comments" links are shown in "Action" column as showed in Figure 4-8. For reviewers, the only link in the "Action" column is "Review."

4.3.2 Assignment UI

"Assign/View Reviewer" link as shown in Figure 4-8 navigates editor-in-chief/editor to the Assignment UI, which helps editor-in-chief to assign editors and also helps editor-in-chief/editor to assign reviewers for a submission. Figure 4-9 shows an example where the editor-in-chief is assigning editors and/or reviewers for submission "08-15428." In the "ASSIGN EDITOR"

section, the editor-in-chief is able to designate any user within the “Assign/Change Editor” pull-down list as the editor of this submission. If a submission has not been assigned to any editor, message “Media has not been assigned.” is showed under text “ASSIGN EDITOR”; otherwise, the assignee’s user ID and assignment date are shown. Currently, a submission can be assigned to only one editor; however, the assigned editor is allowed to be changed if necessary.

Similarly, as shown in Table 4-9, the “ASSIGN REVIEWER” section is the place to assign a submission to reviewers. All specialists in DDIS are reviewer candidates. Ajax is utilized to provide input suggestions for the “Assign a reviewer” text field. If a submission is assigned to a reviewer, the assignee’s name and user ID and assignment date will also be showed on the screen. As shown in the example, submission 08-15428 has been assigned to a reviewer whose name is “j Xin” and user ID is “imglib” on Jun. 18, 2008. A submission can be assigned to multiple reviewers.

As for the Assignment UI for editors, the only difference from the example above is the absence of the “ASSIGN EDITOR” section since editor is not allowed to assign a submission to other editors.

4.3.3 Review UI

The “Review” link in the reviewers’ working list navigates reviewers to the Review UI. An example of the Review UI is shown in Figure 4-10. In this UI, reviewers are allowed to provide recommendations and comments to editor-in-chief and editor. There are three options of recommendation: Accept, Accept with revision, and Reject. The “Comments to Submitter and Editor” text area collects comments to both submitter and editor-in-chief/editor, while the “Comments to Editor” text area collects comments that are only viewable by editor-in-chief/editor.

If the option “Accept with revision” is selected by the reviewer, the screen will be different from Figure 4-10 and an example is shown below. As shown in Figure 4-11, an extra section is appended to the “Comments to Editor” text area. By default, submitter’s original inputs are retrieved from database and filled into the corresponding fields. Reviewer is allowed to modify common name, scientific name, host common name, host scientific name, and family.

4.3.4 View Comments UI

“View Reviewer’s Comments” link shown in Figure 4-8 navigates editor-in-chief/editor to the View Comments UI, where editor-in-chief and editors can view recommendations and comments provided by the reviewers. In an example of the View Comments UI as shown in Figure 4-12, a submission with library ID 08-43881 was reviewed by a reviewer on September 30, 2008, whose name is “aba zhang” and user ID is “aba.”

All information input by reviewer through the Review UI is shown in the “COMMENTS” message box. Since multiple reviewers can be assigned for one submission, more than one message boxes may exist at the same time. If the reviewers do not provide any comment or recommendation, the text message “There is no comment right now.” will be shown in the “COMMENTS” section instead.

4.3.5 Decision UI

In Decision UI, editor-in-chief and editors make their final decision for a submission. As shown in Figure 4-13, this Decision UI is very similar to reviewer’s Review UI. A decision can be made from the following three options: Accept, Accept with revision, and Reject. And again, as discussed earlier, the selection of “Accept with revision” would result the UI as shown in Figure 4-11.

4.4 Media Search/View UI

The Media Search/View UI is divided into two parts: Media Search UI and Media View UI.

4.4.1 Media Search UI

The Media Search UI is shown in Figure 4-14. As shown in the screen, there are four tabs corresponding to four search options: basic search, media ID search, tag-based search (i.e. tag cloud), and advanced search. The blue tab represents the active search option, while gray tabs represent inactive search options.

The user interface for “Basic Search” is shown in Figure 4-14. The keyword could be any string composed of several characters. The user interface for “Media ID Search” is very similar to the screen shown in Figure 4-14: instead of keyword, a valid library ID is needed to get a result for the “Media ID Search.”

A “Tag Cloud” screen (Figure 4-15) is an example of the user interface of tag-based search option. Tags are arranged alphabetically in the tag box. Tags in bigger font size indicate higher frequencies in the database. It is worth mentioning that a tag is not limited to one word in DDISMedia. For instance, “Tomato Spotted Wilt Virus” is treated as one tag. Tags are always up-to-date because they are generated every time when the page is refreshed.

The Advance Search UI is showed in Figure 4-16. Users can customize search criteria using the following fields: common name and scientific name of pest/host, sample type, family, country, state/province, and submission date range. If users specify multiple fields, the search results must qualify all conditions simultaneously.

4.4.2 Media View UI

The Media View UI organizes and shows search results and are composed of the following three components: square image viewer, thumbnail image viewer, and media viewer.

The square image viewer is a user interface showing results retrieved by the “Browse All” search option. As shown in Figure 4-17, each page is designed to show forty images in the order of their submission dates. A page navigation bar is available at the bottom to navigate user to their desired page. Each image has a mouse over to show common name and scientific name of pest.

The thumbnail image viewer is a UI to browse search results of the following search options: basic search, media ID search, tag-based search, and advanced search. In the example shown in Figure 4-18, the word “tomato” is used as the keyword. The left side of the upper tool bar shows the number of search results and search conditions, and the right side allows user to set the number of images showed in one page. Below the tool bar are thumbnail images of search results. The common name and scientific name of pest is showed below each image.

By clicking any image in the square image reviewer or the thumbnail image viewer, detailed information of the corresponding submission is showed in media viewer, including text information and all digital media in this submission. Figure 4-19 shows an example of media viewer. If a media is an image, its normal sized copy with a link navigating to the original image is shown in the media viewing panel. If a media is a video clip, it is embedded in the media viewing panel and playable. Below the media viewing panel is the area showing all related text information for the submission, such as common name, scientific name, location, copyright, etc. Two tool boxes are located at the right side of the media viewer. The upper tool box lists the thumbnail images of all digital media in one submission. Clicking any thumbnail image in this tool box would let the corresponding media shown in the media viewing panel. The lower tool box provides search links implementing simple associated search functions. Associated search topics include scientific name of pest, scientific name of host, photographer, and submitter.

4.5 Interfaces within DDIS

The media library database of DDISMedia is coupled with DDIS pest diagnosis by embedding a function in DDIS diagnosis user interface (Figure 4-20). By default, the button "Search related topics in DDIS Media Library" is disabled; however, it can be activated once any character is input in host or diagnosis field. For example, "Tomato" is used as the common name of host. By clicking the "Search related topics in DDIS Media Library" button, the Media Search/View component will be called to retrieve and show the media in a pop-up window, whose host common name starts with "Tomato."

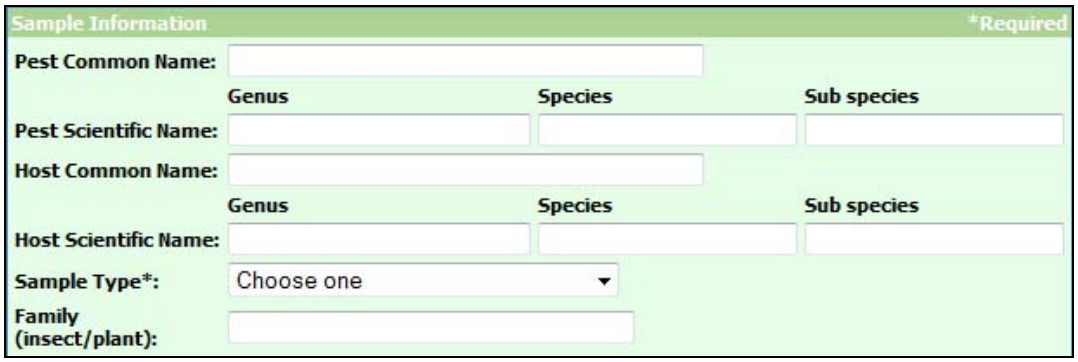

Figure 4-1. Sample information form

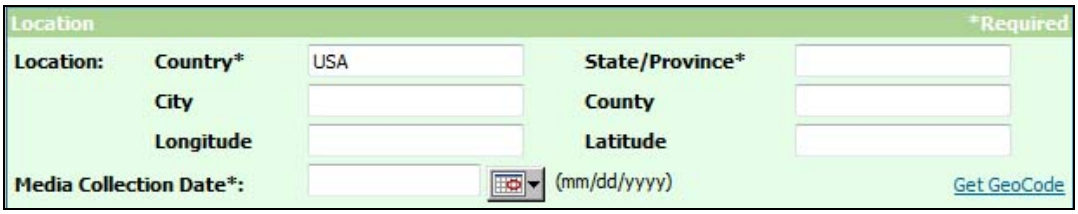

Figure 4-2. Location form

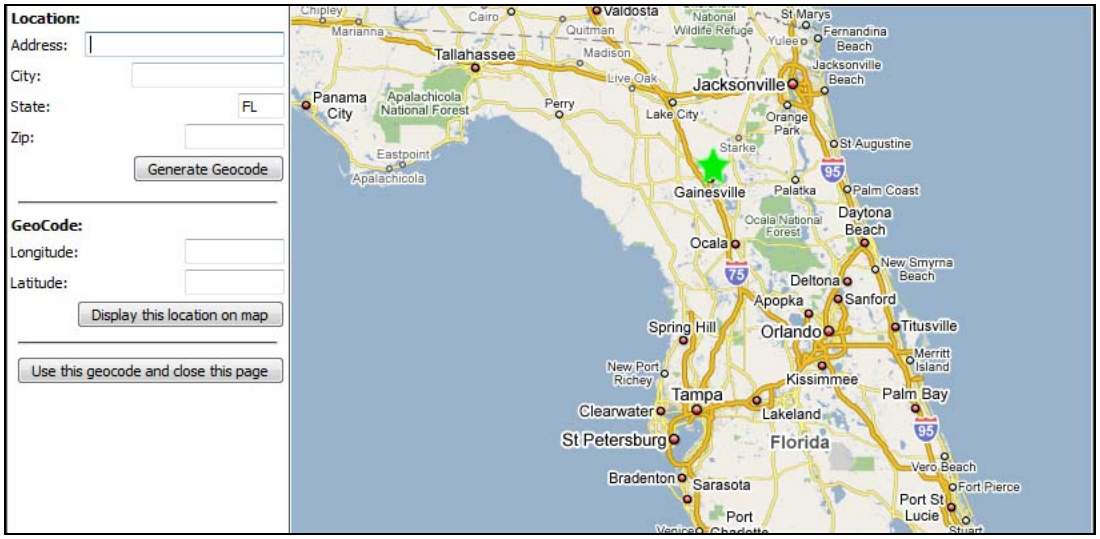

Figure 4-3. Geo-code input panel

Figure 4-4. Additional description form and copyright notes form

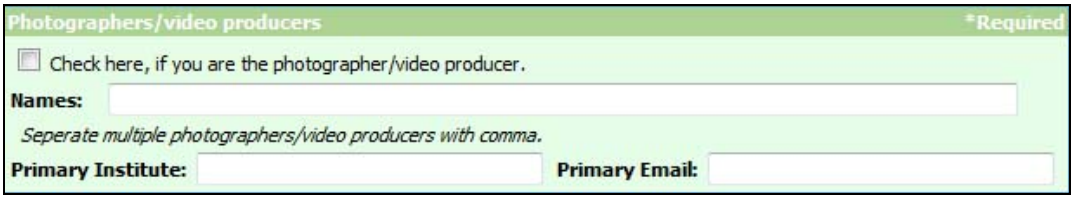

Figure 4-5. Photographer/video producers form

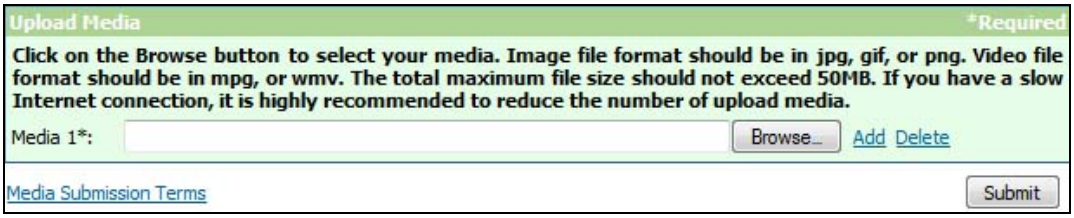

Figure 4-6. Upload Media form

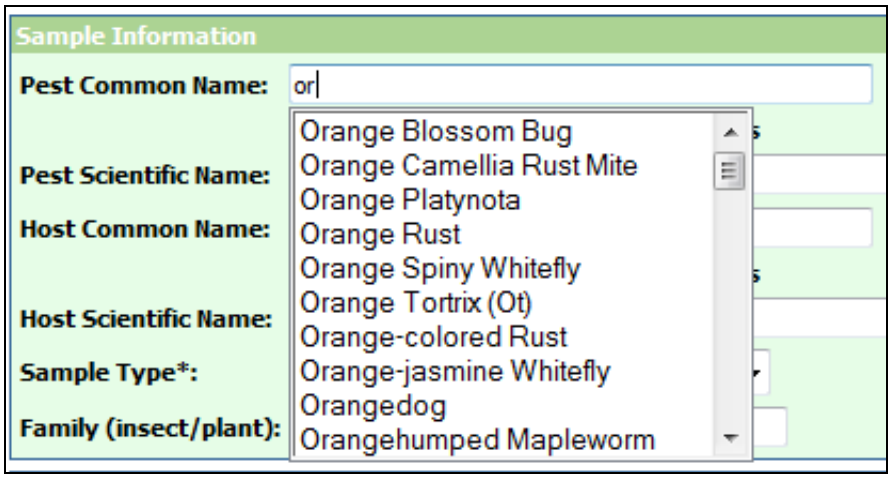

Figure 4-7. Example of Ajax-based input suggestions

ID	Submited By	Sample Type	Submission Date	Action
08-15428	shiwei zhang	Insect (Plant)	04/22/2008	Assign/View Reviewer, View Reviewer's Comments

Figure 4-8. Example of "In Review" working list UI for editor-in-chief/editor

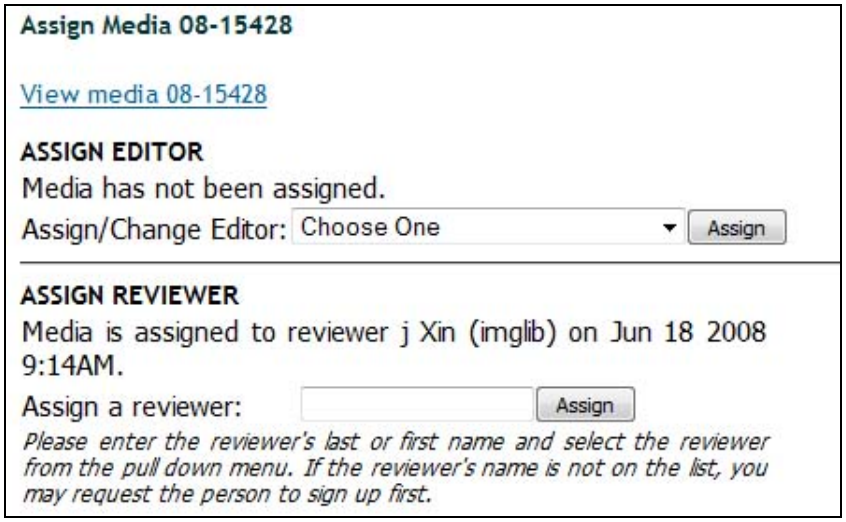

Figure 4-9. Example of Assignment UI for editor-in-chief

Figure 4-10. Example of Review UI for reviewer

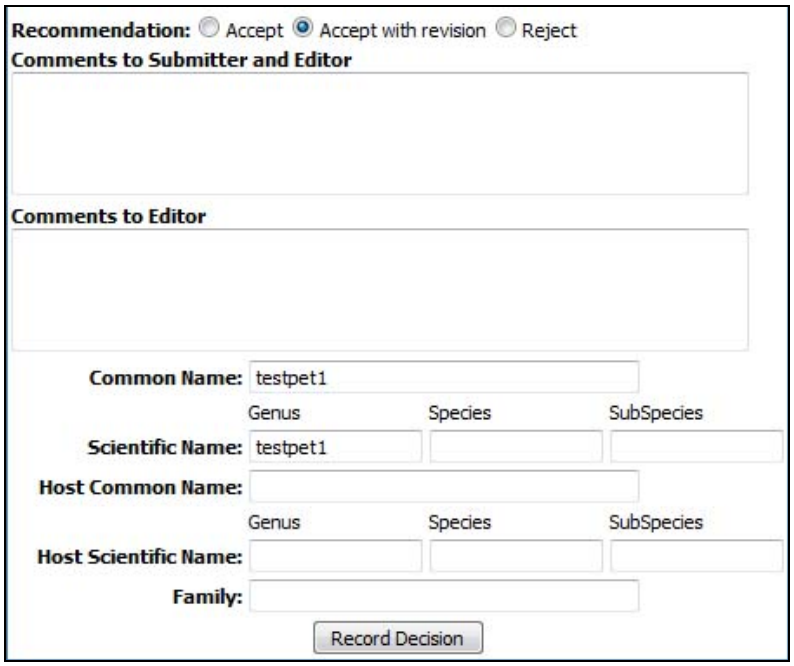

Figure 4-11. Example of user interface when selecting “Accept with revision” in Review UI

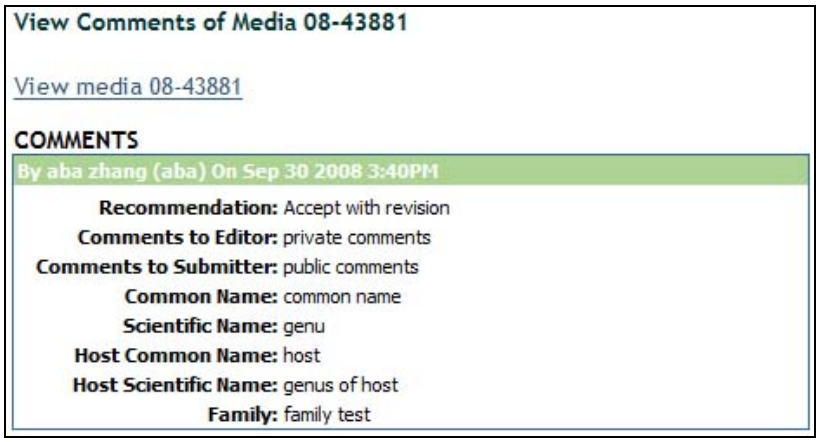

Figure 4-12. Example of View Comments UI for reviewer

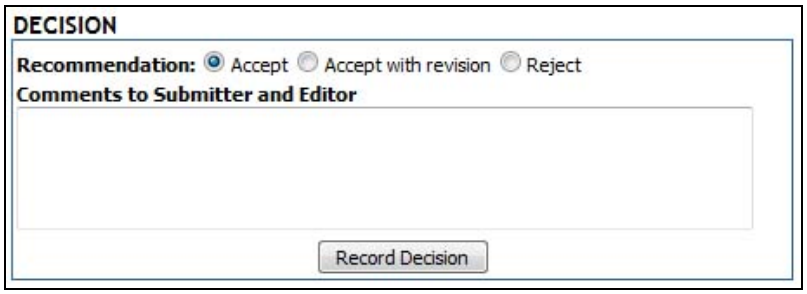

Figure 4-13. Example of Decision UI for editor-in-chief/editor

Figure 4-14. Media Search UI

Figure 4-15. Example of Tag Cloud UI

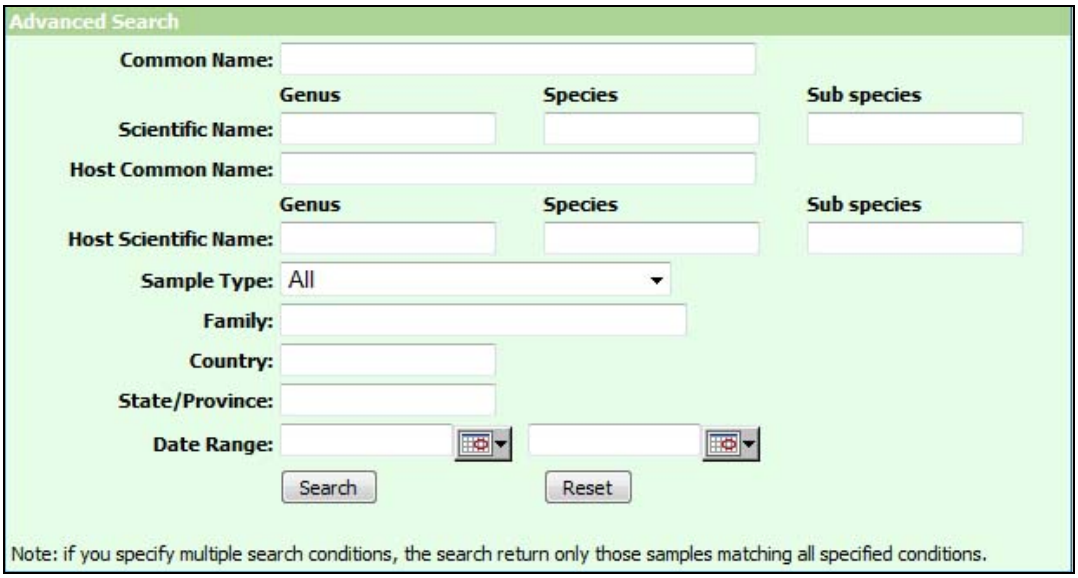

Figure 4-16. Advanced Search UI

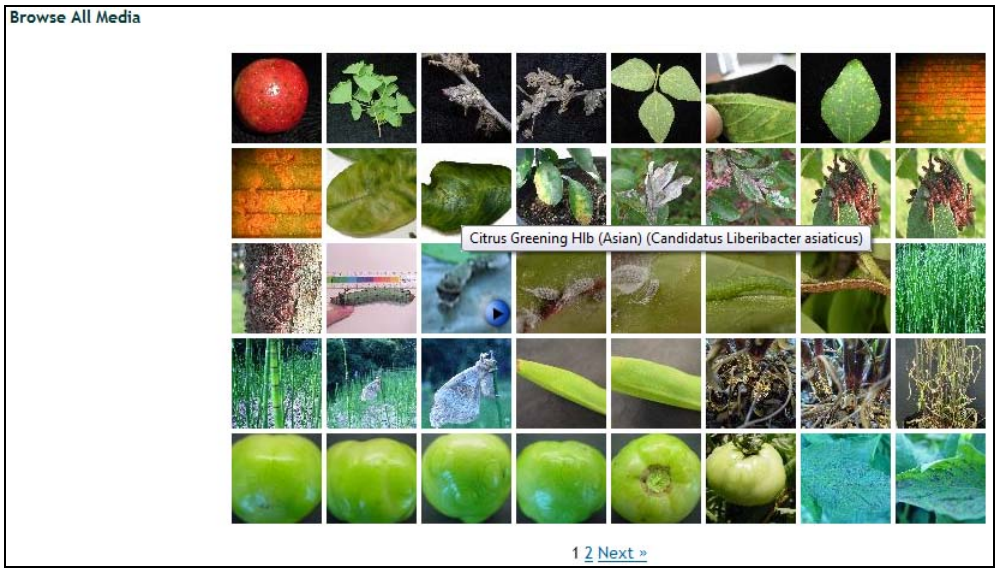

Figure 4-17. Example of square image viewer

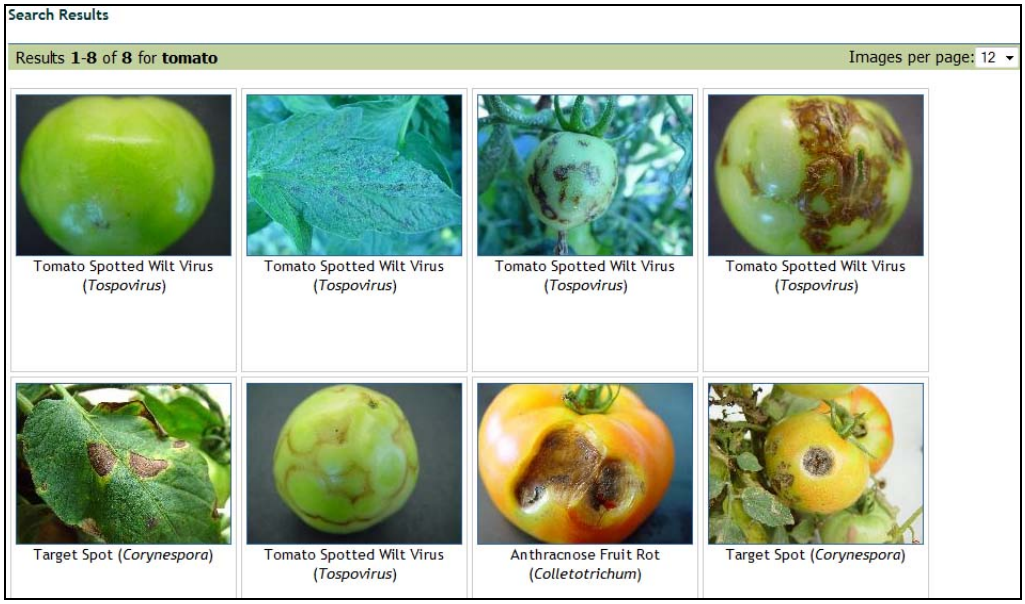

Figure 4-18. Example of thumbnail image viewer

Figure 4-19. Media viewer

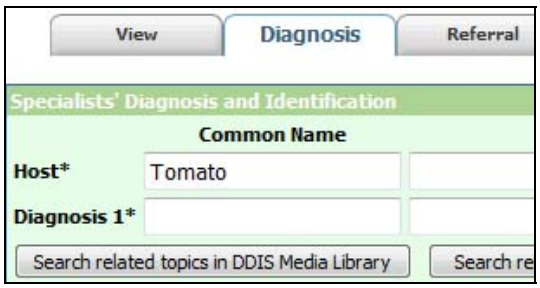

Figure 4-20. DDISMedia embedded in DDIS pest diagnosis

CHAPTER 5
SUMMARY, CONTRIBUTIONS, AND FUTURE DIRECTIONS

5.1 Summary

A study involved the design and implementation of DDISMedia: an online digital media library used in pest diagnosis, research, and education. This web-based system allows users to submit digital media, such as video clips and photographic images obtained from the field, clinics or diagnostic laboratories as an important reference for pest diagnosis in DDIS. DDISMedia utilizes a peer review process to ensure high-quality media and authoritative information within this library. The extensive search capability of DDISMedia provides users a good reference to identify and understand pest management.

Before designing DDISMedia, potential features and requirements for this system were determined, including well-designed graphic user interfaces, reliable media submission process, quality control peer review process, extensive search ability, and integration with DDIS pest diagnosis. The core functional components of DDISMedia contain “Media Submit”, “Media Review”, “Media Search/View”, and “Assisted Diagnosis.” “Media Submit” collects media information and implements a stable file uploading function. “Media Review” relies on a peer review process to ensure the quality of the digital library. “Media Search/View” provides public users extensive search ability and user-friendly browsers. “Assisted Diagnosis” embeds DDISMedia into DDIS diagnosis user interface to assist pest diagnosis in DDIS. The following technologies and tools have been used in the implementation of DDISMedia: HTML, JavaScript, Java Servlet, JSP, Apache FileUpload, SQL, JDBC, and Ajax.

5.2 Contributions

- As specialists increasingly depend on multimedia to help them obtain information and make decisions, DDISMedia currently serves as a useful reference resource for pest diagnosis and identification in DDIS. In addition to the textual field information, it is convenient to compare a diagnostic sample visually with the archived media in the library.

With the accumulation of media in DDISMedia, it will become an important reference tool for DDIS.

- The extensive search ability of DDISMedia provides a rich user interface to assist specialists with pest diagnosis. With the extensive search ability, the relationship of pest taxonomy is expressed, which is not available in the DDIS search functionality.

5.3 Future Directions

The current search capability of DDISMedia has room for improvement. For the current dominant relational database system, it is possible to fail to find out the relationship between two associated samples just due to the absence of similar keyword between them in the database. As one of the solutions to solve the problem, in very small and specified field, people can apply their perception and knowledge in computer preprocessing, such as clustering, or building decision keys in the database to improve the search function. However, when the scope grows, the manual work would increase exponentially and might be impractical for implementation. Therefore, future research on improving database technology could be performed.

Moreover, implementation of a user interface with the ability to visually show the sample relationship is another interesting topic. For example, a hierarchy or network structure might be one of the relationships among beings in biological chain. By using rich internet technologies, such as Ajax and Flex, it is feasible to implement this feature.

In addition, image processing applets can be applied to media viewer. For instance, improving image quality could make images more readable, and emphasizing biological features in an image could assist pest diagnosis and become an improvement in long-distance pest diagnosis.

At last, although other diagnostic labs or organizations might have good-quality media, it is difficult for DDISMedia to connect their database directly without permission. Therefore,

obtaining data from the search user interfaces of those severs and showing these data in DDISMedia to extend the search scope of DDISMedia is another topic deserved further research.

LIST OF REFERENCES

Bergsten, Hans, 2003, JavaServer Pages, O'Reilly, Cambridge, MA.

Crane, Dave, and Pascarello, Eric, 2005, Ajax in Action, Manning, Greenwich, CT.

Flanagan, David, 2006, JavaScript: The Definitive Guide, O'Reilly, Cambridge, MA.

Goodman, Danny and Morrison, Michael, and Eich, Brendan, 2004, JavaScript Bible, John Wiley and Sons, Hoboken, NJ.

Groff, James, and Weinberg, Paul, 2002, SQL: The Complete Reference, McGraw-Hill Professional, New York, NY.

Holmes, Gerald, Brown, Edward, and Ruhl, Gail, 2002, What's A Picture Worth? The Use of Modern Telecommunications in Diagnosing Plant Diseases, Plant Disease, Vol. 84 No. 12, 1256-1265.

Hunter, Jason, and Crawford, William, 2001, Java Servlet Programming, O'Reilly, Cambridge, MA.

Kurniawan, Budi, 2002, Java for the Web with Servlets, JSP, and EJB, Sams Publishing, Sebastopol, CA.

Mahemoff, Michael, 2006, Ajax Design Patterns, O'Reilly, Cambridge, MA.

Musciano, Chuck, and Kennedy, Bill, 2007, HTML & XHTML: The Definitive Guide, O'Reilly, Cambridge, MA.

Paulson, L.D., 2005, Building Rich Web Applications with Ajax, Computer, Volume 38, Issue: 10, 14-17.

Perry, Bruce W., 2004, Java Servlet and JSP Cookbook: Practical Solutions to Real-world Problems, O'Reilly, Cambridge, MA.

Ramakrishnan, Raghu, and Gehrke, Johannes, 2002, Database Management Systems, McGraw-Hill Professional, New York, NY.

Smith, K., 2006, Simplifying Ajax-style Web development, Computer, Volume: 39, Issue: 5, 98-101.

Willard, Wendy, 2007, HTML: A Beginner's Guide, McGraw-Hill Professional, New York, NY.

Xin, Jiannong, Beck, Howard, Halsey, Larry, Fletcher, James, and Zazueta, Fedro, 1998, Using Digital Cameras and the Internet to Identify Plant Insect and Disease Problems, Proceeding of 7th International Conference on Computers in Agriculture. Orlando, Florida.

Xin, Jiannong, Zazueta, Fedro, and Beck, Howard, 2003, A web-based distance diagnostic and identification system for extension, EFITA 2003 Conference, 460-465.

BIOGRAPHICAL SKETCH

Shiwei Zhang was born in 1981, in Chongqing, China. He received his Bachelor of Engineering degree in computer science and technology from Nankai University in 2003. He began his graduate study in a concurrent degree program at the University of Florida in spring 2006. After completing the Master of Science degree in the Computer Information and Science Engineering Department in December 2007, he continued the Master of Engineering degree in the Agricultural and Biological Engineering Department, in the area of information technology under the supervision of Dr. Fedro Zazueta. As a research assistant, his work focused on redesigning and implementation of DDIS, a Distance Diagnostic and Identification System, which is currently widely used in pest diagnosis by the state of Florida and some international organizations. He also published two academic papers at World Congress on Computers in Agriculture.