



# Acceptance Test



Web Organization and Registration Management System

---

**For:**

**Dr. Darren Lim,  
Dr. Tim Lederman**

**Produced By:**



**Alyssa Nghiem – Team Leader / QA Lead**

**Eduardo Pinto Barbosa – Team Leader / QA Lead**

**Ryan Kennedy – Lead Programmer**

**Chris Small – Assistant Programmer**

**Tyler Vorpahl – Web Designer**

5/1/2013



# 1. Table of Contents

- 1. Table of Contents..... 1
- 2. Product Overview..... 4
- 3. User Case Narratives..... 4
  - System Administrator..... 4
  - Conference Chair..... 4
  - Submission Chair ..... 5
  - Reviewer..... 5
  - Submitter..... 5
  - Programming Contest Team Advisor..... 5
  - Programming Contest Participant..... 6
  - Conference Attendee..... 6
  - Vendor ..... 6
- 4. Use Case Diagram ..... 7
  - UML Use Case Diagram Legend ..... 8
  - UML Use Case Diagram ..... 9
- 5. Data Flow Diagrams ..... 10
  - Data Flow Diagram Legend ..... 10
  - Context Diagram ..... 11
  - WORMS Level 0 Diagram ..... 12
  - Process 1: Create Account..... 13
  - Process 1.2: Validate Account Information ..... 14
  - Process 2: Login..... 15
  - Process 3: Review ..... 16
  - Process 3.3: Check Reviews ..... 17
  - Process 3.3.1: Get Scholarly Work..... 18
  - Process 4: Submit ..... 19
  - Process 4.3: Store Submission ..... 20



Process 4.3.1: Generate File Name ..... 21

Process 5: Maintain System..... 22

Process 5.1: Configure ..... 23

Process 6: Configure Conference ..... 24

Process 7: Register for Conference ..... 25

6. Structure Diagram..... 26

7. Functional Requirements Inventory..... 27

    System Administrator ..... 28

    Conference Chair ..... 28

    Submission Chair ..... 28

    Reviewer ..... 29

    Submitter..... 29

    Programming Contest Team Advisor..... 30

    Programming Contest Participant..... 30

    Conference Attendee..... 31

    Other..... 31

8. Non-functional Requirements ..... 31

9. Future Enhancements ..... 31

10. Entity-Relationship Diagram and Table Relations..... 32

    ERD Legend ..... 33

    Entity-Relationship Diagram ..... 34

    Table Relations ..... 35

11. User Displays ..... 35

12. Appendix A: Glossary of Terms ..... 36

13. Appendix B: Project Timeline..... 38

14. Appendix C: Data Dictionary..... 39

15. Appendix D: Test Requirements and Results ..... 42

    Overview and Strategy ..... 42

    Unit Tests and Test Cases..... 42

    Integration Test..... 42



System Test .....	43
Functional Requirements.....	43
Non-functional Requirements.....	46
16. Appendix E: Database Schema.....	53
17. Appendix F: User Displays .....	59



## 2. Product Overview

Dr. Lim will be holding the Consortium for Computing Sciences in Colleges Northeast (CCSCNE) Conference in April 2013. The organization of this conference is very complex and there is a need for software that can assist in the handling attendee registration, scholarly work submission, scholarly work review, as well as scheduling and general information dissemination. The WORMS web application will provide these services.

## 3. User Case Narratives

User Case Narratives describe all the different users and their functionalities of the system. Listed below are all the possible users of WORMS and a description of what each user is able to do.

### System Administrator

The System Administrator is the person responsible for configuring the details of the system and monitoring the system. The System Administrator will log into a built-in account on the web site. The System Administrator will be able to configure a list of the email addresses of Conference Chairs. The System Administrator will also be able to configure a list of the email addresses of Submission Chairs and the type of scholarly work submission that each Submission Chair is responsible for. The System Administrator will be able to configure a list of questions that users will be asking when registering for the conference. The System Administrator will be able to view a list of all submitted scholarly works. The System Administrator will be able to view and remove accounts from the system. The System Administrator will be able to configure the list of topics that scholarly works can be categorized into.

### Conference Chair

A Conference Chair is one of the people responsible for organizing the conference. A Conference Chair will create an account in the WORMS web application. In order to have the necessary permissions, a Conference Chair must have created the account using an email address on the list of Conference Chair emails configured by the System Administrator. A Conference Chair will be able to log into the WORMS web application. A Conference Chair will be able to schedule the times and locations of events at the conference. A Conference Chair will be able to edit other conference information such as parking information, nearby hotels, and nearby attractions. A Conference Chair will be able to configure banquet meal selections.



### **Submission Chair**

A Submission Chair is one of the people responsible for organizing the submission and review of a certain type of scholarly work to the conference. A Submission Chair will create an account in the WORMS web application. In order to have the necessary permissions, a Submission Chair must have created the account using an email address on the list of Submission Chair emails configured by the System Administrator. Each Submission Chair will be associated with a specific type of scholarly work. A Submission Chair will log into the WORMS web application. A Submission Chair will specify deadlines for submissions of scholarly works. A Submission Chair will be able to view a list of scholarly works submitted and the person that submitted each scholarly work. A Submission Chair will be able to view the reviews given to scholarly works. A Submission Chair will choose whether or not a scholarly work should be accepted to the conference. A Submission Chair will confirm the credentials of each Reviewer.

### **Reviewer**

A Reviewer is a person with sufficient credentials who wishes to review submitted scholarly works. A Reviewer will create an account in the WORMS web application. A Reviewer will log into the WORMS web application. The Reviewer will enter credentials into a form in the WORMS web application and apply to become an accepted Reviewer. After a Reviewer has been accepted, the Reviewer will be able to view scholarly works that have been submitted and assigned to that Reviewer. An accepted Reviewer will be able to submit a review for each scholarly work that was assigned to that Reviewer.

### **Submitter**

A Submitter is a person who wishes to submit a scholarly work to the conference. A Submitter will create an account in the WORMS web application. A Submitter will log into the WORMS web application. A Submitter will select the type of scholarly work that the Submitter wants to submit. A Submitter will select the topic of the scholarly work that the submitter will submit. A Submitter will upload a PDF (Portable Document Format) file containing a paper or an abstract for a scholarly work. A Submitter will be able to check whether or not the scholarly work that the Submitter submitted has been reviewed.

### **Programming Contest Team Advisor**

A Programming Contest Team Advisor is a faculty member responsible for organizing a programming team that will compete at the programming competition. A Programming



Contest Team Advisor will create an account in the WORMS web application. A Programming Contest Team Advisor will register a programming contest team. A Programming Contest Team Advisor will provide the email addresses of the students on the programming contest team. A Programming Contest Team Advisor will pay the fee necessary to participate in the contest.

### **Programming Contest Participant**

A Programming Contest Participant will create an account in the WORMS web application. A Programming Contest Participant will register to attend the conference. A Programming Contest Participant will be able to view a confirmation that the Programming Contest Team Advisor has added the Programming Contest Participant to a team. A Programming Contest Participant will be able to view details about the programming contest.

### **Conference Attendee**

A Conference Attendee will create an account in the WORMS web application. A Conference Attendee will log into the WORMS web application. If a Conference Attendee wishes to attend the banquet, the Conference Attendee will make a meal selection. A Conference Attendee will pay the fee for attending the conference. A Conference Attendee may choose to purchase extra banquet tickets. A Conference Attendee will be able to view parking and hotel information. A Conference Attendee will be able to view the schedule of events during the conference. A Conference Attendee will be able to view conference sponsor information.

### **Vendor**

The conference allows Vendors to appear at the conference and set up displays for products. Vendors will use the WORMS application in the same way as a Conference Attendee. Vendors will pay a registration fee to attend the conference. Any organization required for the Vendors' displays will be handled outside of the WORMS application.



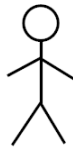
## 4. Use Case Diagram

The UML Use Case Diagram shows all the different users of the system, including both human and non-human. These are called actors. The diagram also shows the functions each user can perform called uses. Each user of WORMS is connected to different functions in the diagram by lines. Below is the legend of the WORMS UML diagram, followed by the actual diagram.

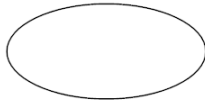




### UML Use Case Diagram Legend



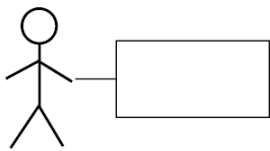
Actor: an external entity that interacts with the system. This may or may not be a human user



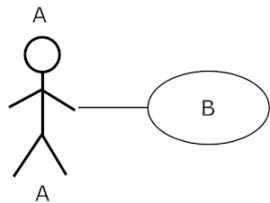
Use Case: a process within the system that actors may interact with



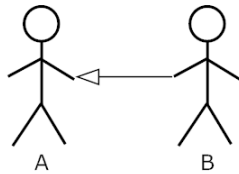
System Boundary: the boundary between the system internals and external Actors



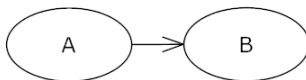
Actor A interacts with all Use Cases in the system



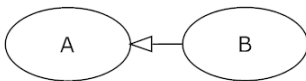
Actor A interacts with Use Case B



Actor B interacts with call Use Cases that Actor A is able to interact with.



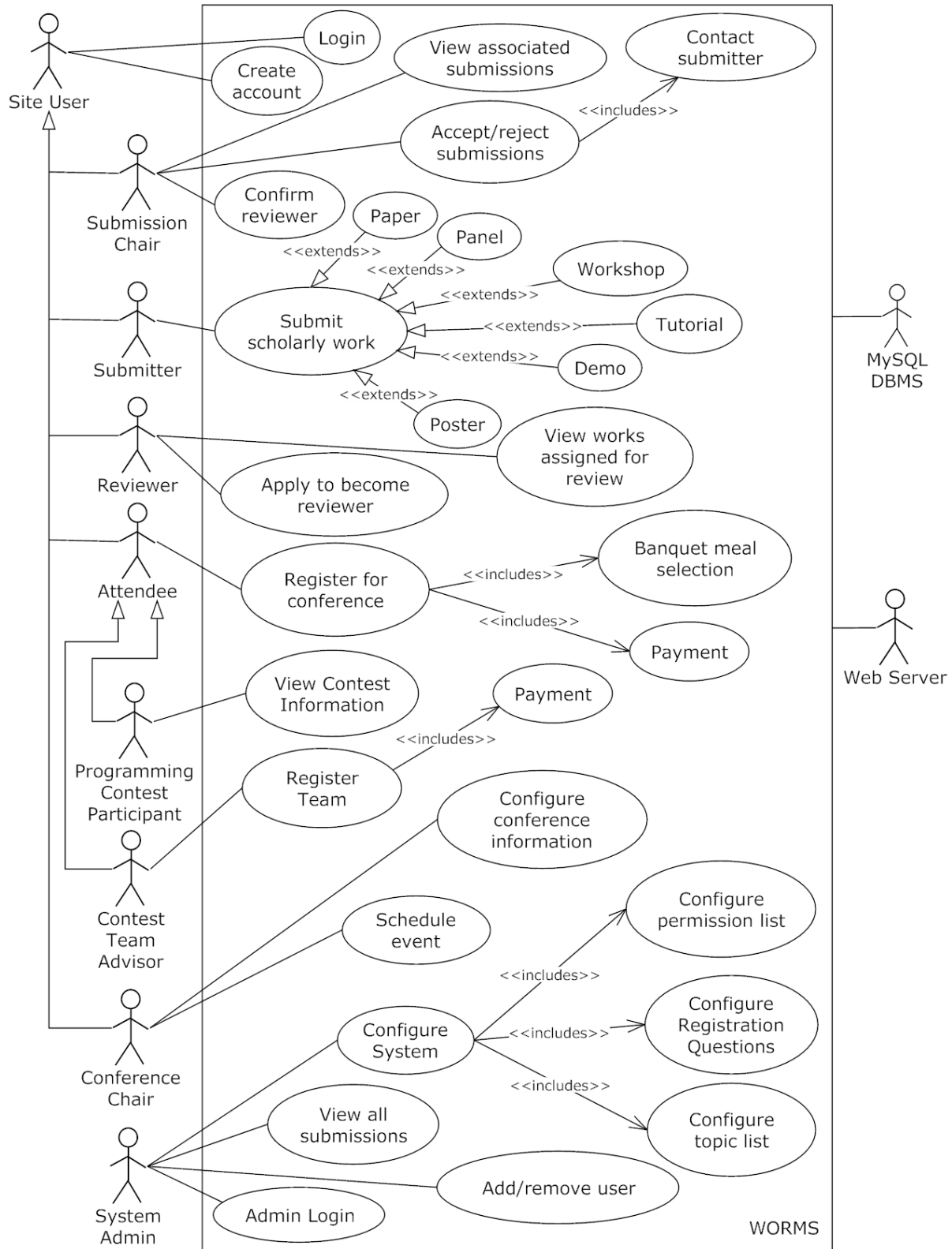
Use Case A includes Use Case B



Use Case B is a more specific type of Use Case A



UML Use Case Diagram

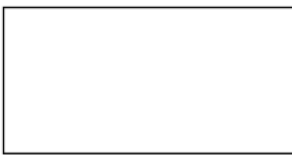




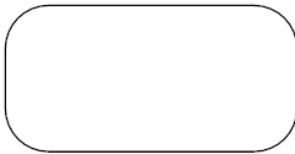
## 5. Data Flow Diagrams

This section will contain data flow diagrams for the system. These diagrams represent the movement of data between processes in the system as well as the movement of data between processes and external entities outside the system. The diagrams are a tool for analyzing the structure of the system and the ways in which data will be stored and retrieved by different processes. These diagrams model data flows at different levels of detail in the system. The symbols that are used in the diagrams are defined as follows.

### Data Flow Diagram Legend



Entity - either a source or destination of data for the system



Process - System functions that can receive data, modify data, and output data.



Data Store - A storage location for persistent data in the system.



Data Flow - movement of a data entity.

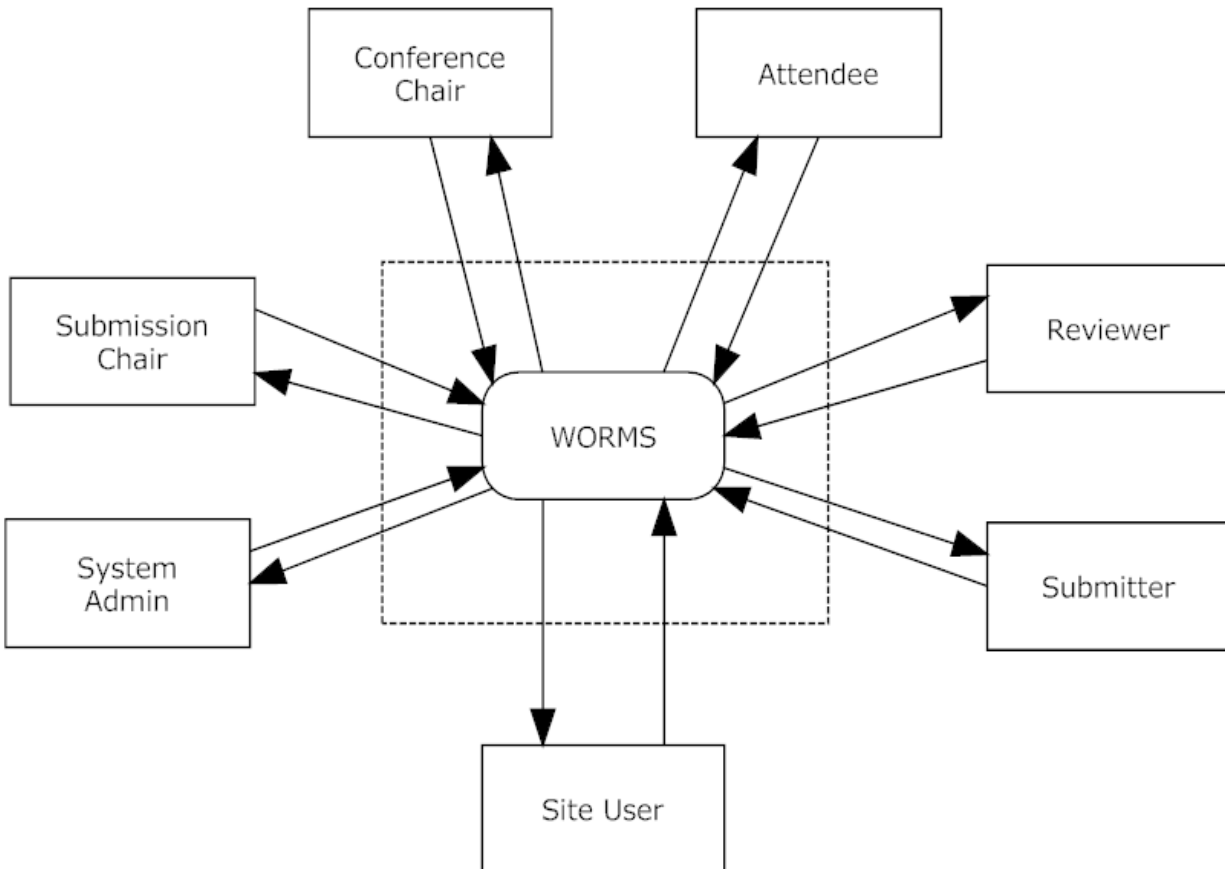


System boundary - the boundary between the system and external entities.



### Context Diagram

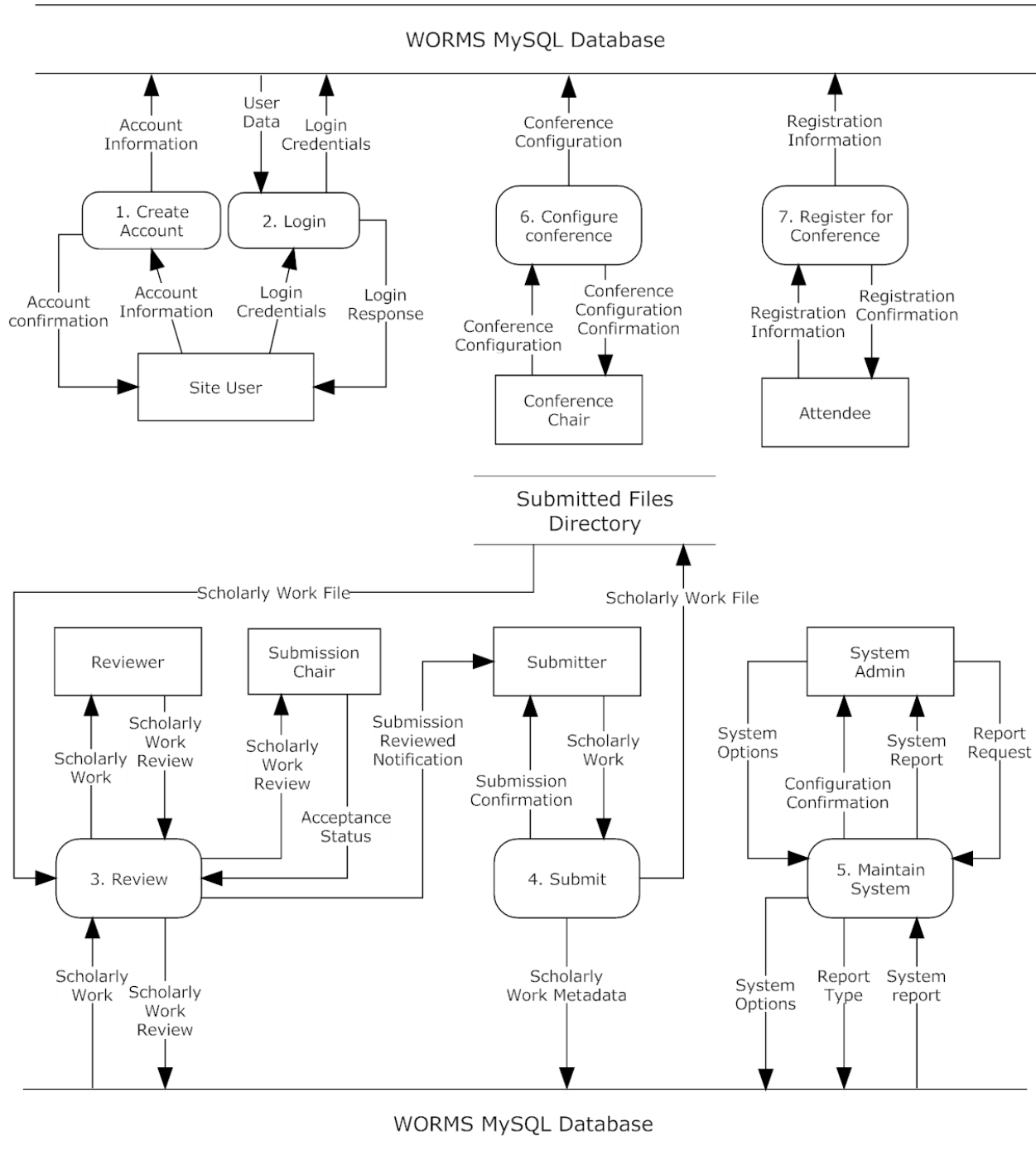
A Context Diagram is shown below. This is the general overview of each of the different entities interacting with the system.





### WORMS Level 0 Diagram

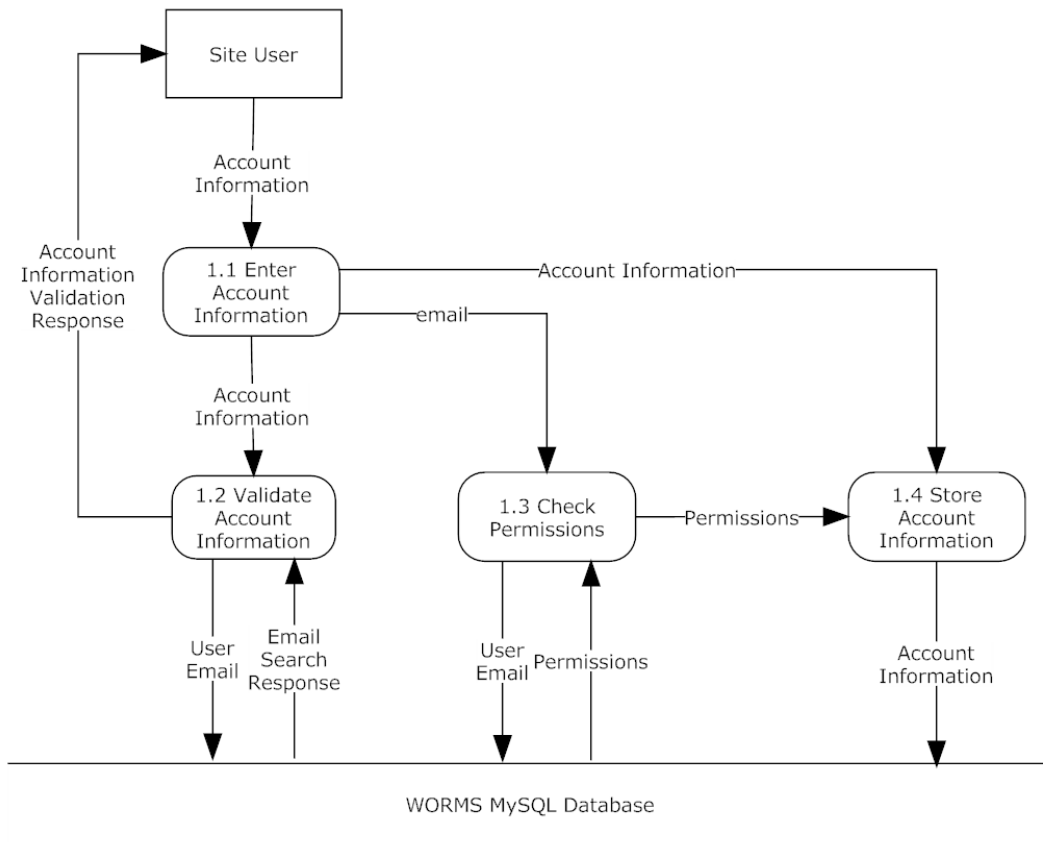
The Level 0 Diagram shows the major processes of the system and how they interact. Arrows are drawn to show the flow of data between the entities and processes.





**Process 1: Create Account**

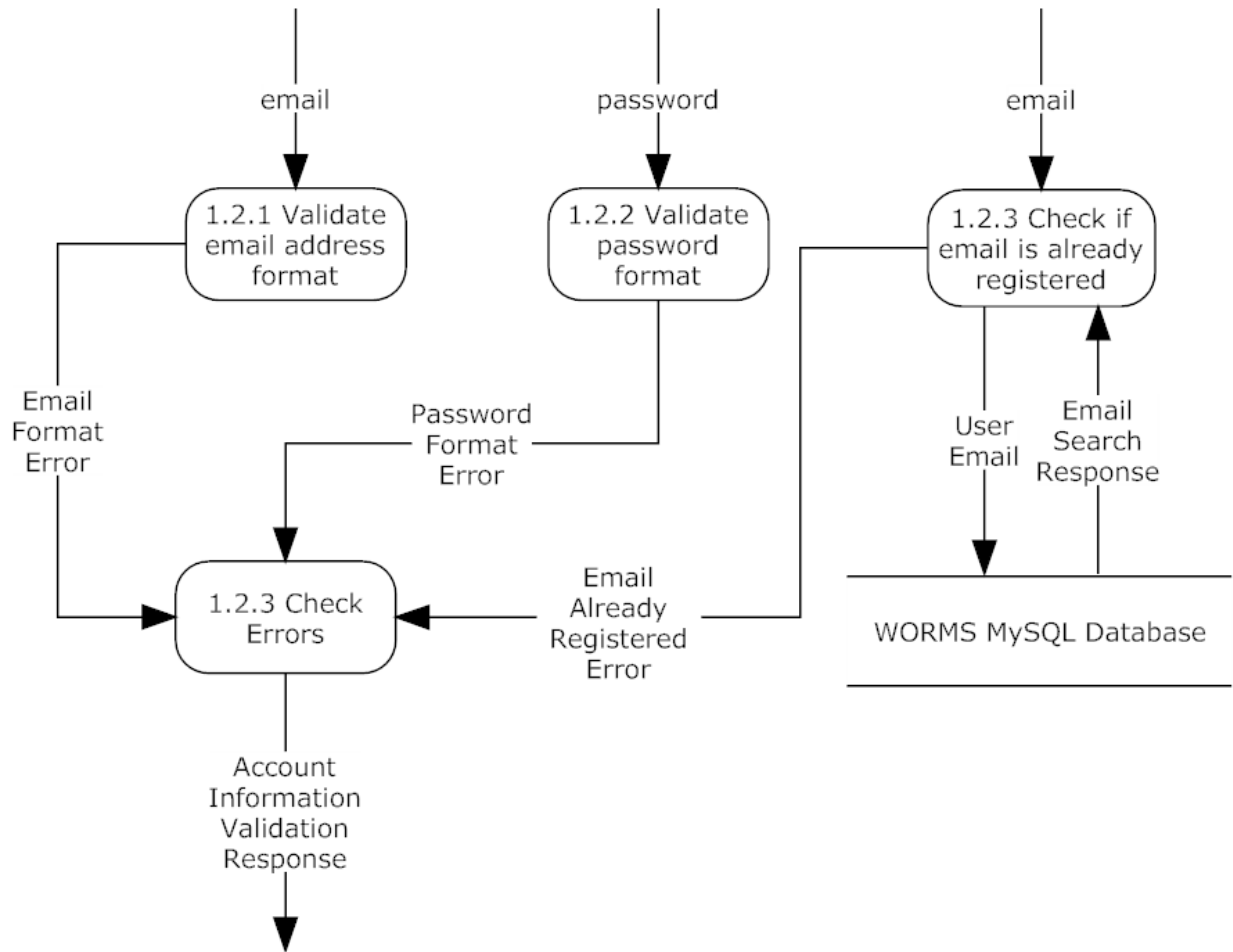
This Level 1 Diagram shows the Process 1: Create Account. This process is used to set up an account for a new user of the system.





**Process 1.2: Validate Account Information**

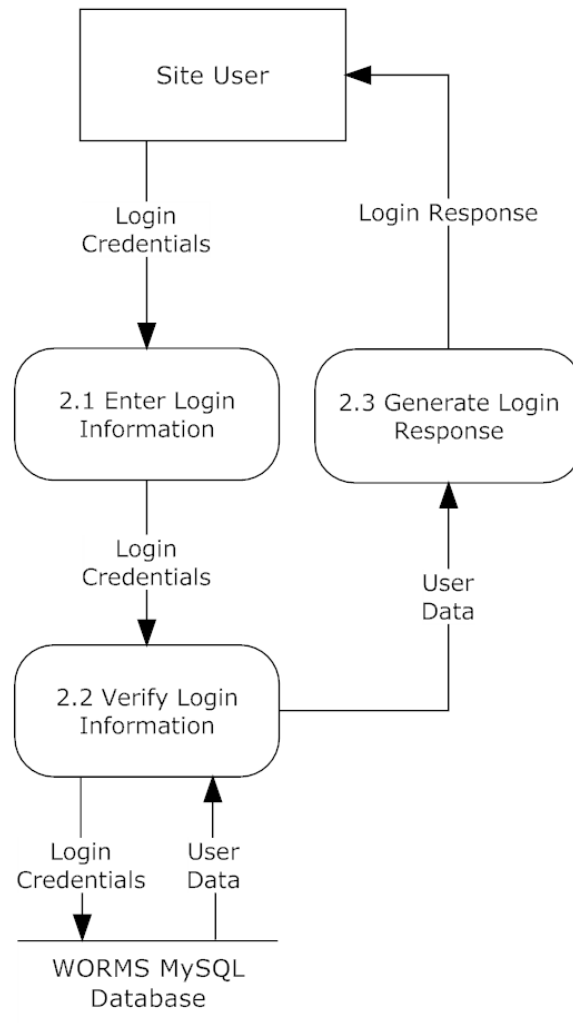
This Level 2 Diagram shows the Process 1.2: Validate Account information. This process is used each time an account is created to ensure that the information provided by the user is valid.





**Process 2: Login**

This Level 1 Diagram shows Process 2: Login. Each user interacts with this process in order to gain access to the main functionality of the system.

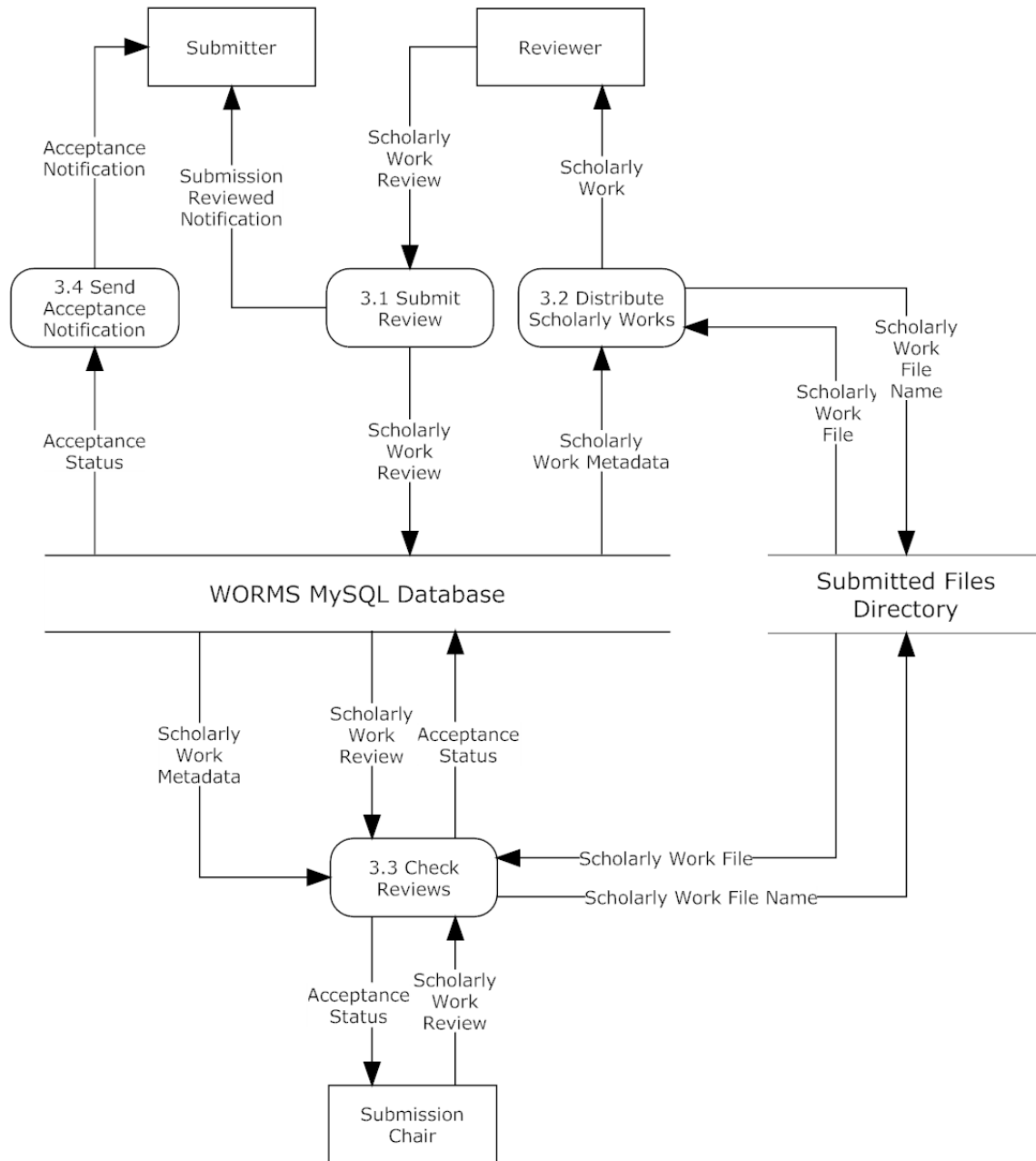






**Process 3: Review**

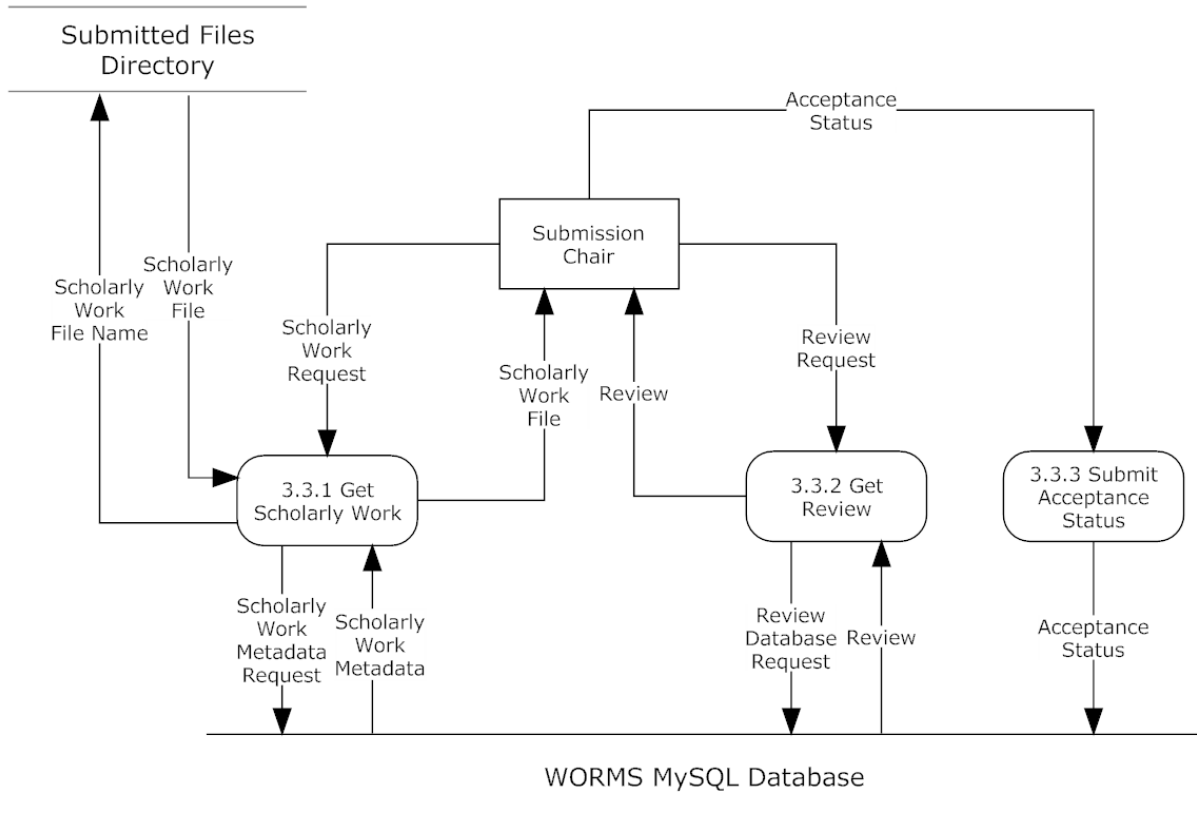
This Level 1 Diagram shows the Process 3: Review. This process involves the creation of reviews for each Scholarly Work and the determination of which Scholarly Works will be accepted to the conference.





**Process 3.3: Check Reviews**

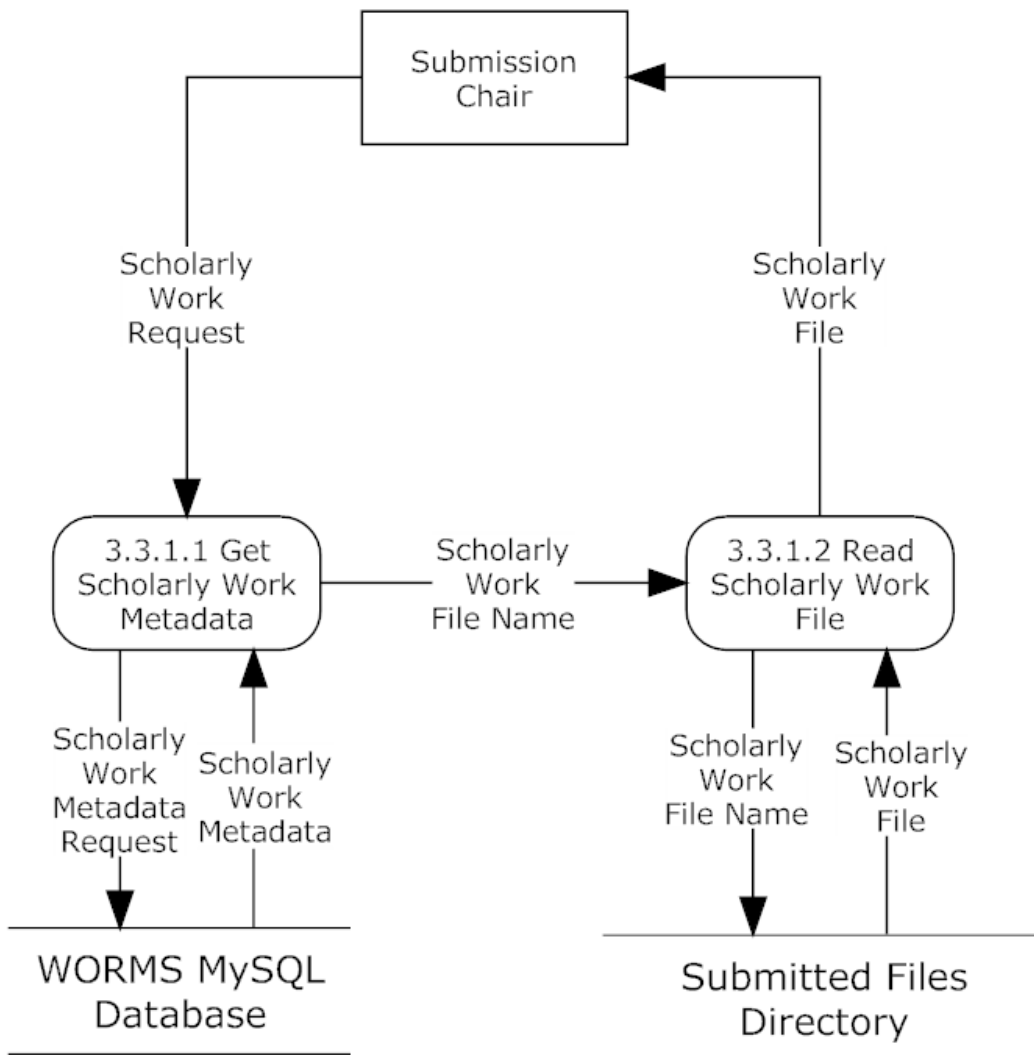
This Level 2 Diagram shows the Process 3.3: Check Reviews. This diagram shows the process of Check Reviews within the Review process. This process is used to determine which submitted scholarly works will be accepted into the conference.





**Process 3.3.1: Get Scholarly Work**

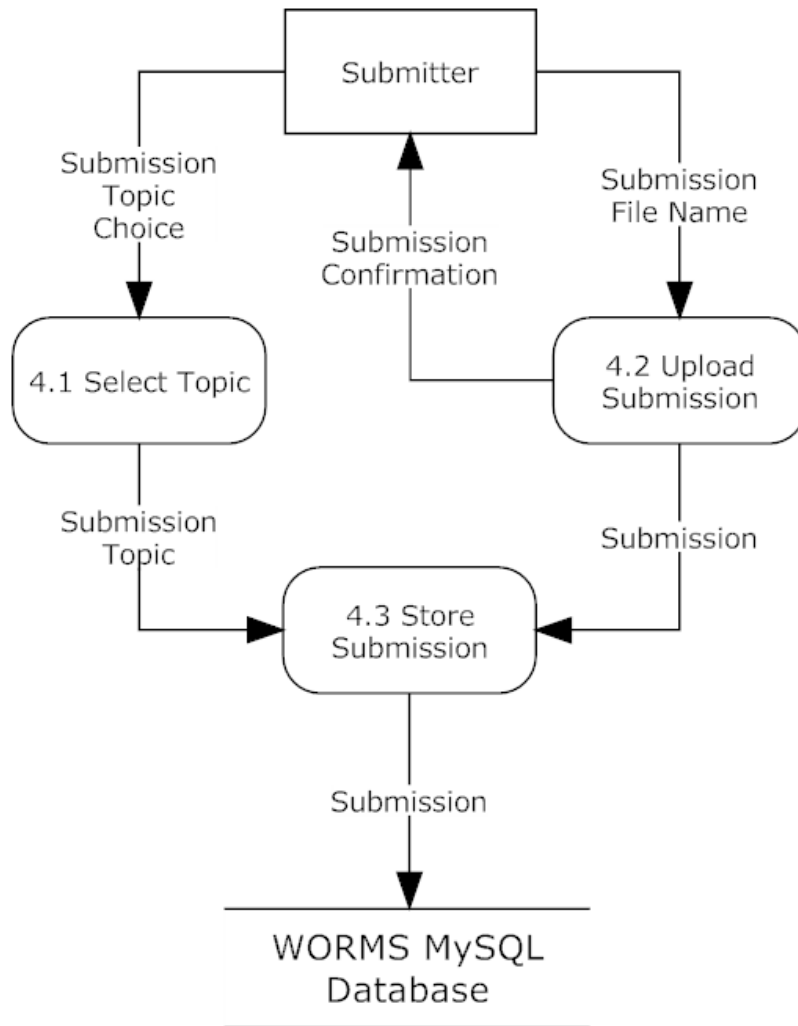
This Level 2 Diagram shows Process 3.3.1: Get Scholarly Work. This process is used to retrieve a Scholarly Work for examination by a Submission Chair.





**Process 4: Submit**

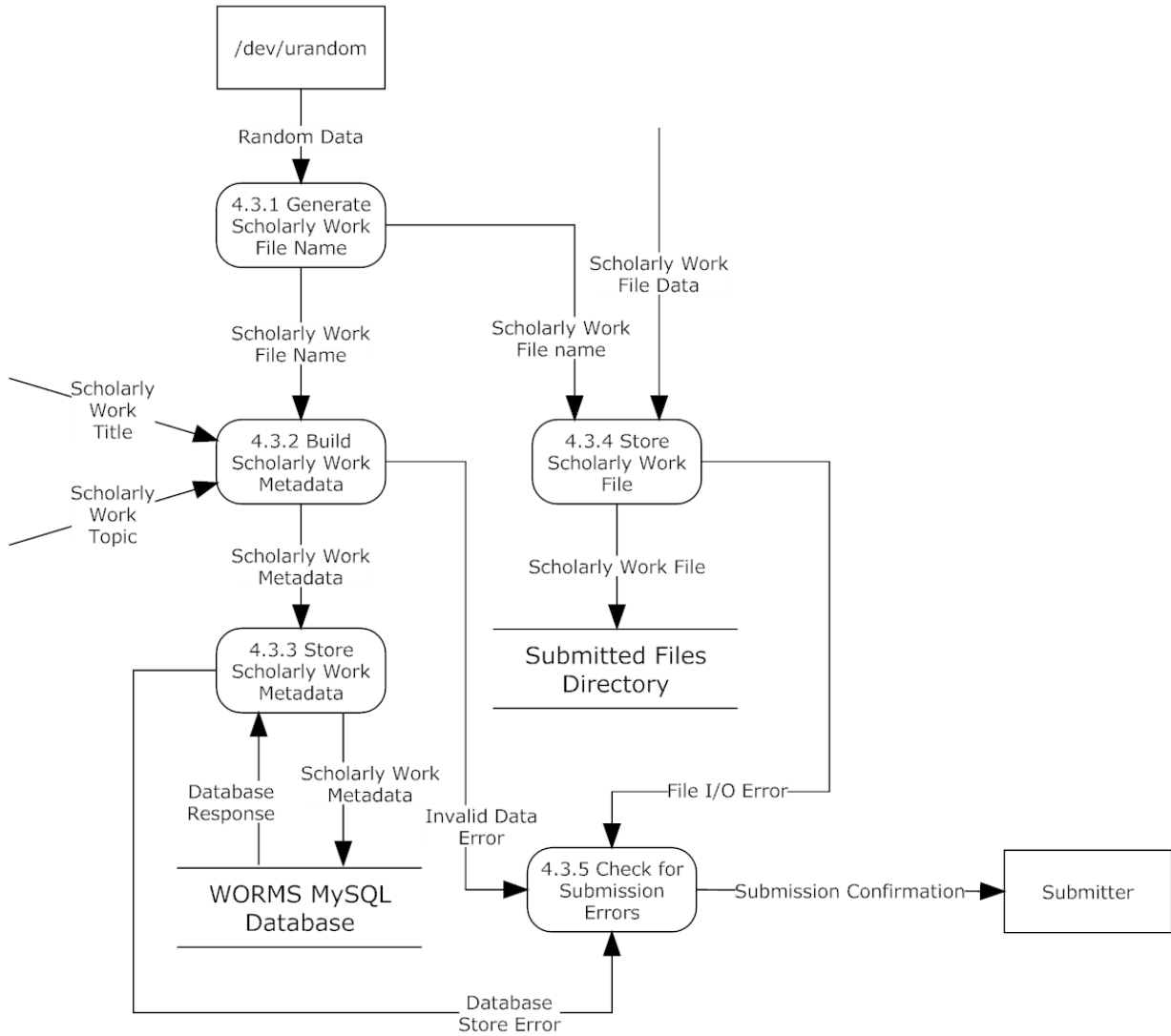
This Level 1 Diagram shows Process 4: Submit. This is the process of Scholarly Works being uploaded to the WORMS server and registered into the system.





**Process 4.3: Store Submission**

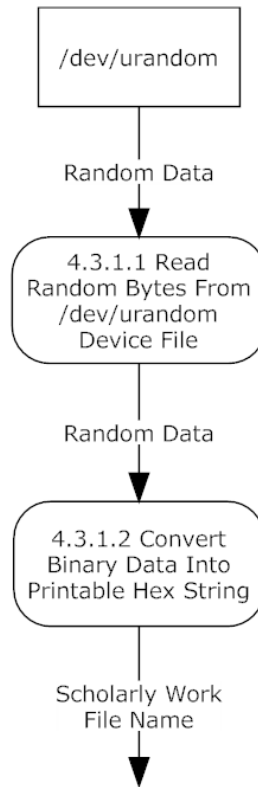
This Level 2 Diagram show Process 4.3: Store Submission. This process is used to store a Scholarly Work and the necessary metadata upon submission by a Submitter.





### Process 4.3.1: Generate File Name

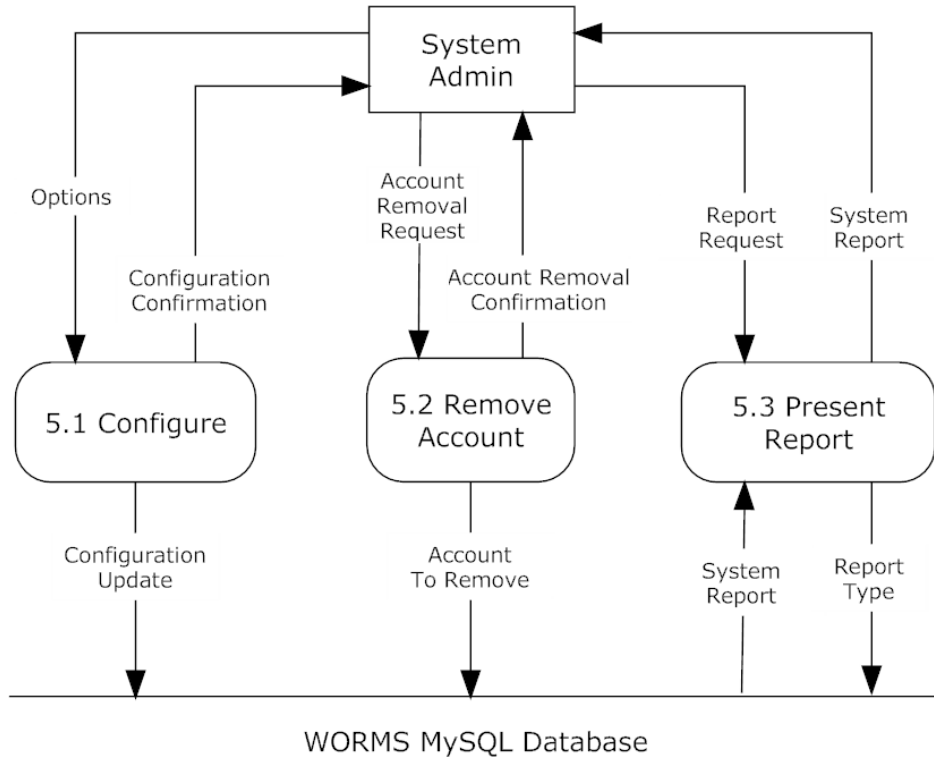
This Level 3 Diagram shows Process 4.3.1: Generate File Name. This process is used to generate a file name for the file where a submitted Scholarly Work will be stored.





### Process 5: Maintain System

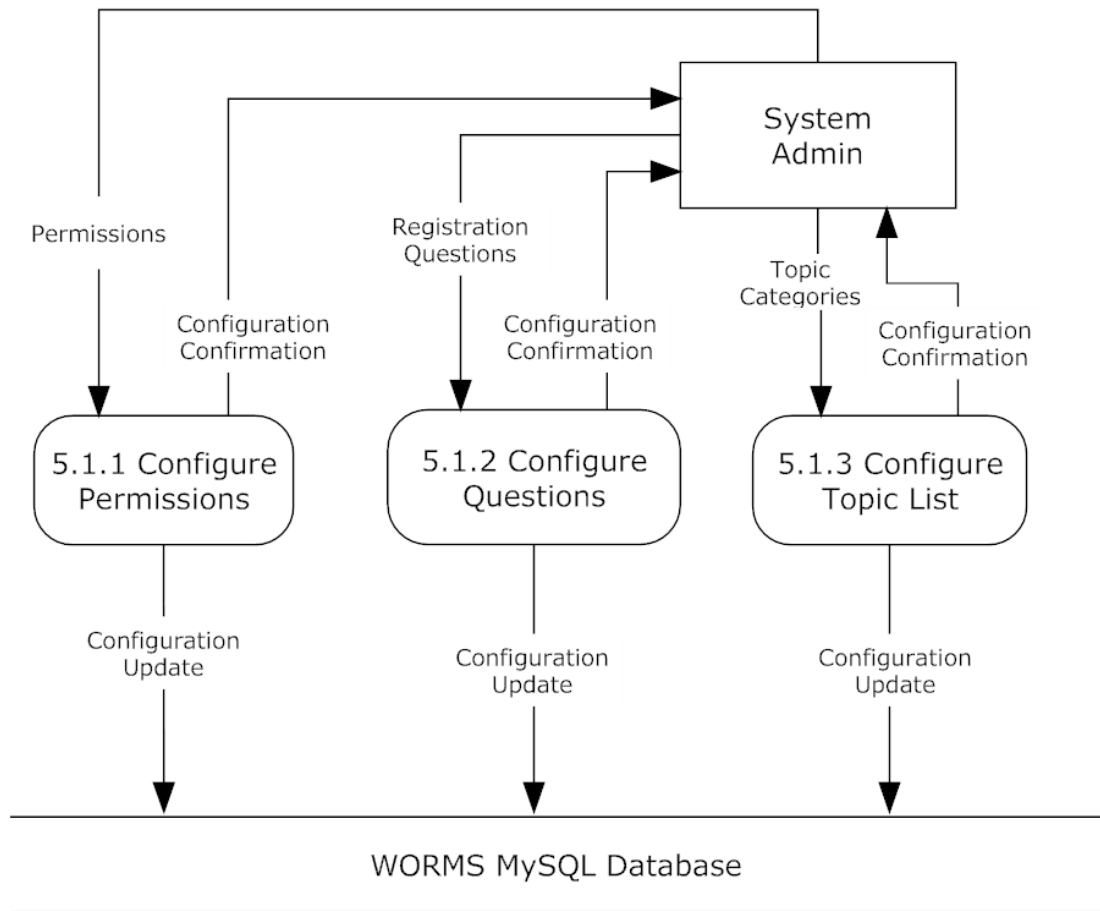
This Level 1 Diagram shows the Process 5: Maintain System. This is the process the System Administrator interacts with in order to initially set up the system and to keep it functioning properly.





**Process 5.1: Configure**

This Level 1 Diagram shows Process 5.1: Configure. This process is used when the System Administrator wishes to change system options and perform other administrative tasks.

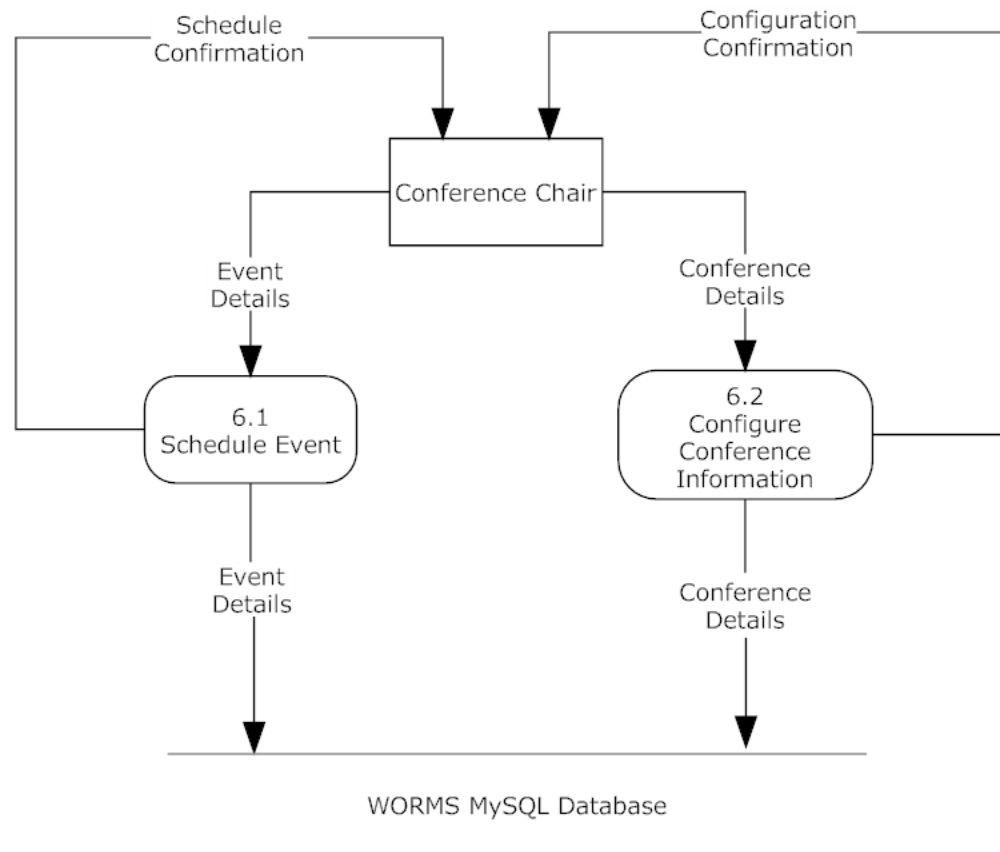






### Process 6: Configure Conference

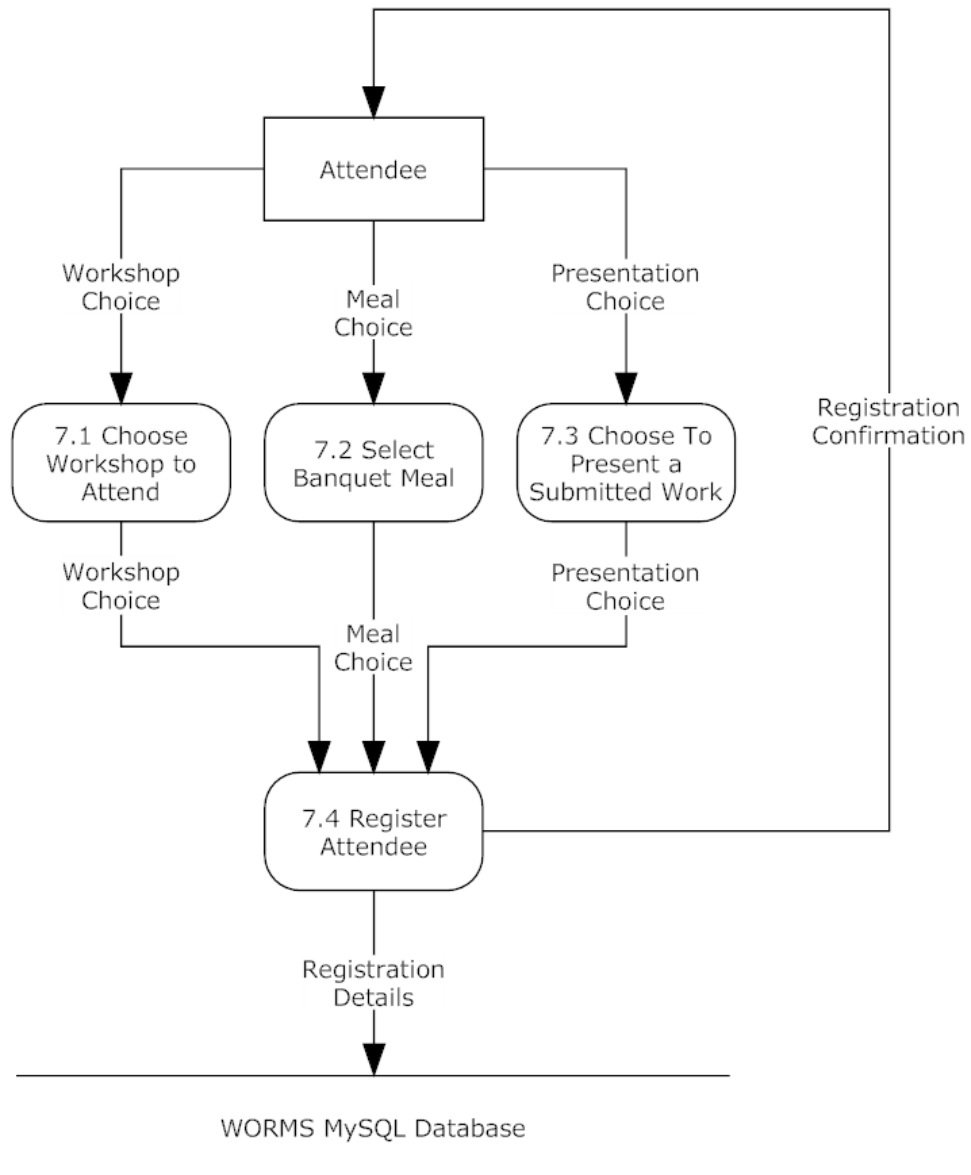
This Level 1 Diagram shows the Process 6: Configure Conference. This process allows the Conference Chair to schedule events and edit the conference information.





**Process 7: Register for Conference**

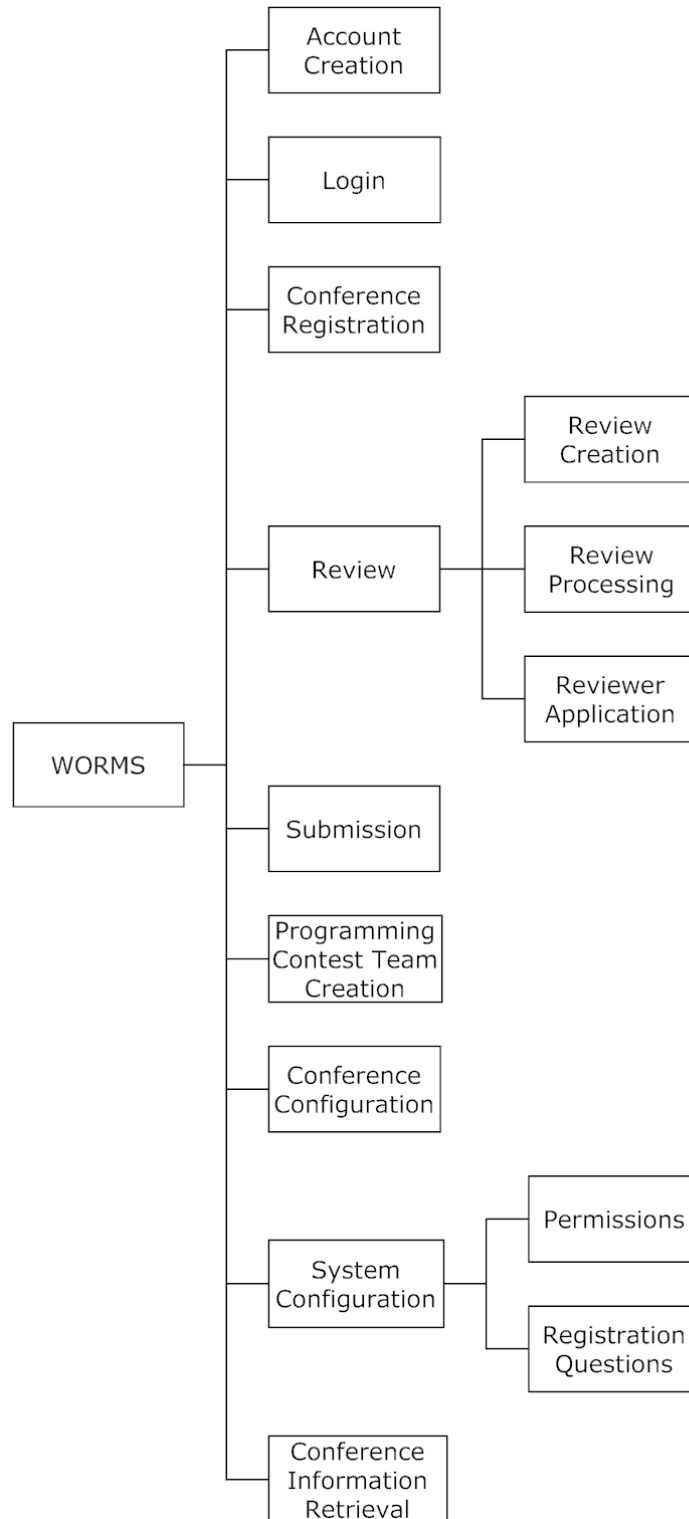
This Level 1 Diagram shows the Process 7: Register for Conference. This process allows a conference attendee to register for the conference.





## 6. Structure Diagram

This diagram models the hierarchical structure of the major components of the WORMS software.



## 7. Functional Requirements Inventory

Functional requirements specify the properties of each user.



### **System Administrator**

- Will be able to log onto WORMS with a provided username and password that is stored on the database
  - An incorrect username and/or password will result in appropriate error message
- Will be able to log off WORMS at any point
- Will be able to configure a list of email addresses stored on the database of the conference chairs
- Will be able to configure a list of email addresses stored on the database of the submission chairs
- Will be able to configure the types of scholarly works each Submission Chair is responsible for
- Will be able to configure a list of questions that users are asked while registering stored on the database
- Will be able to view a list of all submitted scholarly works stored on the database
- Will be able to view and/or remove accounts from the system
- Will be able to configure the topics scholarly works can be categorized into

### **Conference Chair**

- Will be able to create an account using a valid email address confirmed by the System Administrator
- Will be able to log onto WORMS with a provided username and password stored on the database
  - An incorrect username and/or password will result in appropriate error message
- Will be able to log off WORMS at any point
- Will be able to schedule times and locations of events to be stored on the database
- Will be able to edit information such as parking information, nearby hotels, and nearby attractions stored on the database
  - Will be able to view the final appearance of this information as a normal user will see it
  - This editing will take place in the same location on the web site where the final information will be presented to the user
- Will be able to configure banquet meal selections stored on the database

### **Submission Chair**

- Will be able to create an account using a valid email address confirmed by the System Administrator
- Will be able to log onto WORMS with created username and password stored on the database



- An incorrect username and/or password will result in appropriate error message
- Will be able to log off WORMS at any point
- Will be associated with a specific type of scholarly work
- Will be able to specify deadlines for submissions of scholarly works
- Will be able to view a list of scholarly works submitted stored on the database
  - Will be able to view each scholarly works and submitter who submitted the scholarly work stored on the database
- Will be able to view the reviews given to scholarly works stored on the database
- Will be able to download all submitted works
- Will be able to choose whether or not a scholarly work should be accepted to the conference
- Will be able to confirm the credentials of each reviewer stored on the database

### **Reviewer**

- Will be able to create an account using a valid email address
  - A confirmation email will be used to validate the email address
- Will be able to log onto WORMS with created username and password stored on the database
  - An incorrect username and/or password will result in appropriate error message
- Will be able to log off WORMS at any point
- Will register to become an accepted Reviewer
  - Will be able to check whether or not credentials have been accepted yet
- Will be able to view scholarly works that have been submitted and assigned stored on the database
- Will be able to review scholarly works assigned to the reviewer stored on the database
  - Will be able to fill out a review form for scholarly work
  - Will be able to submit review form of scholarly work to be stored on the database

### **Submitter**

- Will be able to create an account using a valid email address
  - A confirmation email will be used to validate the email address
- Will be able to log onto WORMS with created username and password stored on the database



- An incorrect username and/or password will result in appropriate error message
- Will be able to log off WORMS at any point
- Will be able to select the type of scholarly work the Submitter wants to submit
  - Will be able to select the topic of scholarly work the submitter will submit
  - Will be able to upload a PDF file containing a paper or abstract for a scholarly work to be stored in a file on the server
  - The name of the file containing the Scholarly Work will be stored in a specific folder with a randomly generated name. This file name will be associated with the scholarly work in the database.
  - Will be able to check whether or not the scholarly work has been reviewed
- Will receive a notification via email of submission acceptance or rejection

### **Programming Contest Team Advisor**

- Will be able to create an account using a valid email address
  - A confirmation email will be used to validate the email address
- Will be able to log onto WORMS with created username and password stored on the database
  - An invalid username and/or password will result in appropriate error message
- Will be able to log off WORMS at any point
- Will be able to register a programming team
  - Will be able to provide email addresses of the students participating in the programming contest to be stored on the database
  - Each account created with an email address specified by a Programming Contest Team Advisor will be associated with the team created by that Programming Contest Team Advisor
  - Will pay the fee necessary for registering a programming contest team

### **Programming Contest Participant**

- Will be able to create an account using a valid email address as specified by a Programming Contest Team Advisor on a team email list
  - A confirmation email will be used to validate the email address
- Will be able to log onto WORMS with created username and password stored on the database
  - An invalid username and/or password will result in appropriate error message



- Will be able to log off WORMS at any point
- Will be able to view information about the programming contest such as dates, times, and contest environment information

### **Conference Attendee**

- Will be able to create an account using a valid email address
  - A confirmation email will be used to validate the email address
- Will be able to log onto WORMS with created username and password stored on the database
  - An invalid username and/or password will result in appropriate error message
- Will be able to log off WORMS at any point
- Will be able to select a workshop to attend at the conference
- Will be able to view information about hotels and nearby attractions
- Will be able to register for the conference
- Will be able to view a schedule of events at the conference

### **Other**

- Users who are not Conference Chairs, Submission Chairs, or the System Administrator will be able to apply to become a Reviewer or Submit a Scholarly Work
- WORMS must function properly in the latest versions of Google Chrome, Mozilla Firefox, and Safari

## **8. Non-functional Requirements**

Non-functional requirements specify the properties of the system that are not necessarily specific features to be implemented. The only non-functional requirement for the WORMS software is that the software must be user friendly.

## **9. Future Enhancements**

At this point in time the WORMS software is incomplete. The software can be enhanced first by implementing all features specified in the Functional Requirements Inventory. Major missing features are marked with a bold "No" in the Acceptance Criteria section. In addition, the software could be made more user-friendly to fulfill the Non-functional Requirements.





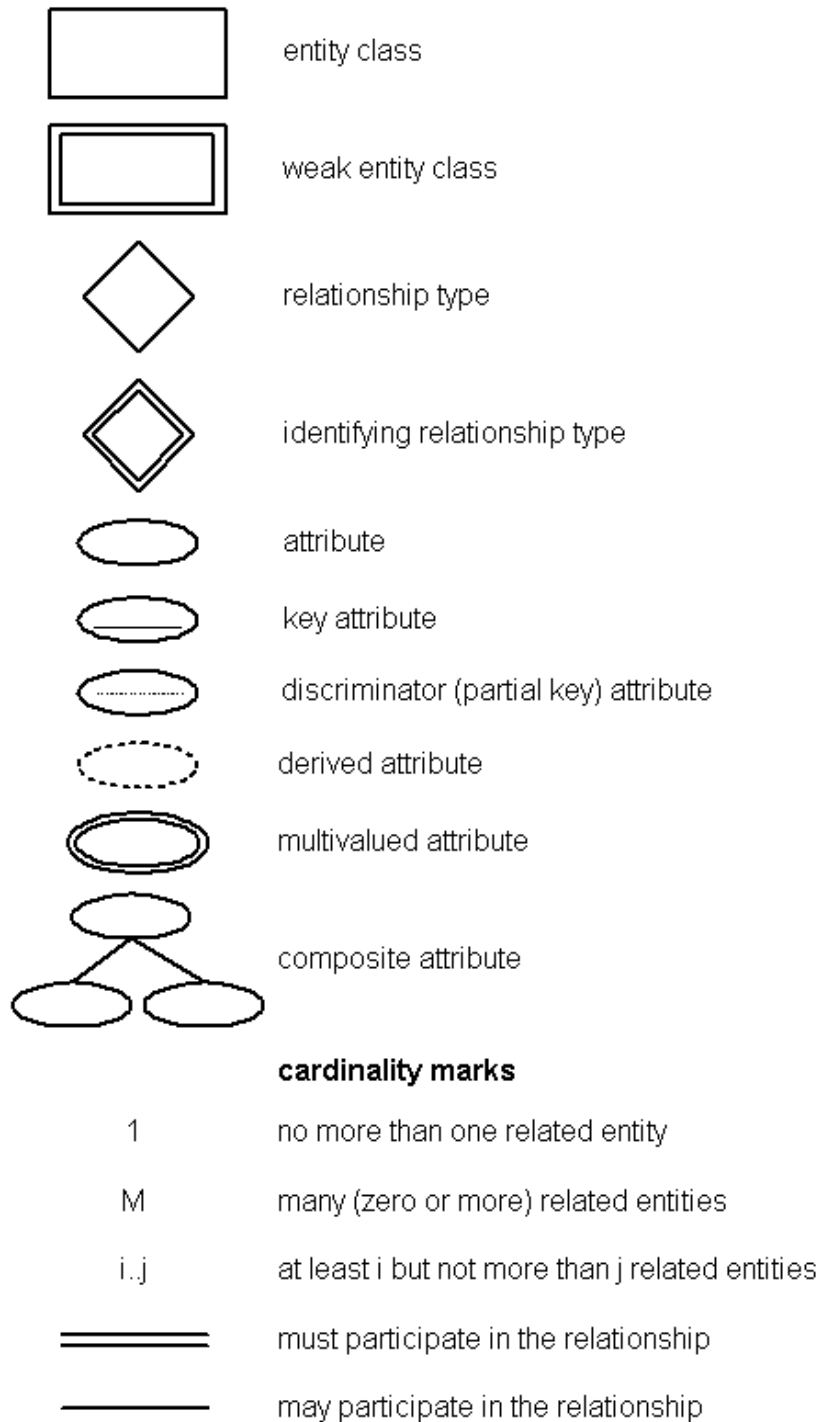
•

## **10. Entity-Relationship Diagram and Table Relations**

The Entity-Relationship Diagram models the different types of data stored in the WORMS database and the relationships between these types of data.

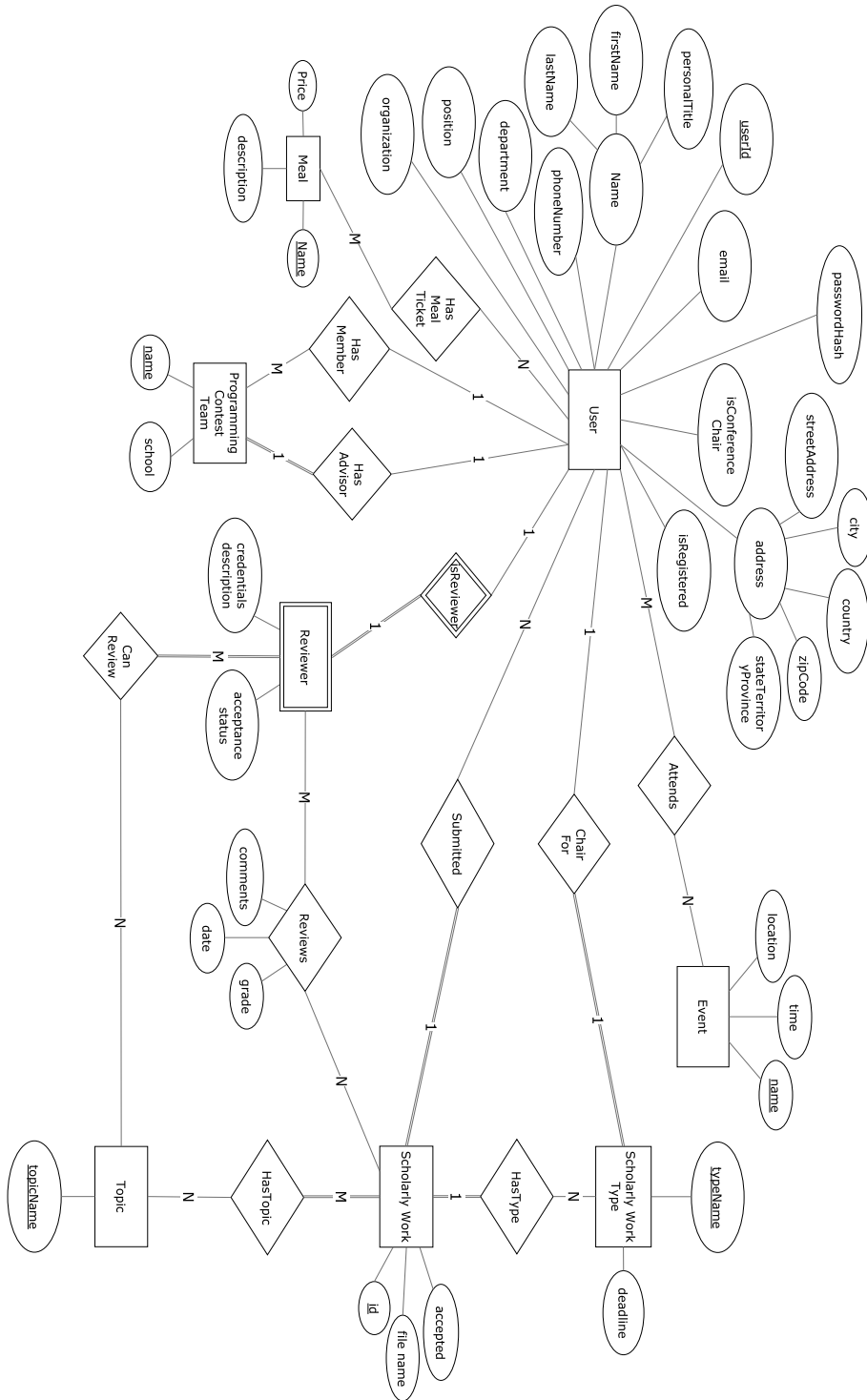


**ERD Legend**





### Entity-Relationship Diagram





### **Table Relations**

The full specification of the schema for all tables in MySQL Data Description Language is available in Appendix E.

## **11. User Displays**

Appendix F contains images showing the preliminary layout of several web pages in the WORMS application.



## **12. Appendix A: Glossary of Terms**

CCSCNE – Consortium for Computing Sciences in Colleges Northeast

CSS – Cascading Style Sheets, language used to style web pages

DBMS – Database Management System

Go – A compiled, statically and strongly typed programming language developed by Google

I/O – Input and output

HTML – Hypertext Markup Language, language used to structure web pages

JavaScript – browser side scripting language

Linux – an open source operating system kernel

MySQL – a database management system

Scholarly Work – one of the following: poster, paper, workshop, tutorial, panel

UML – Unified Modeling Language

WORMS – Web Organization Registration Management System

