

# P.R.I.S.M.

## PROJECT RECORDING INFORMATION SYSTEM MANAGEMENT

### PRELIMINARY DESIGN

**Presented By:**



**Shannon Pfohl: Team Leader**

**Paul Cherrier: Web Master**

**Ryan Egan: Information Specialist**

**Jordan Holoboski: Database Administrator**

**Kathleen O'Hara: System Administrator**

**Julian Thomas: Development Director**

## Contents

1. Summary of the Client’s Problem .....	5
2. Use Case Narratives .....	5
2.1 Employee .....	5
2.2 Team Leader.....	5
2.3 Supervisor .....	6
2.4 System Administrator .....	6
3. UML Diagrams.....	7
3.1 Use Case Diagram .....	7
3.1.1 UML Use Case Legend.....	7
3.1.2 Use Case Diagram .....	8
3.2 Deployment Diagram .....	9
3.2.1 Deployment Diagram Legend .....	9
3.2.2 Deployment Diagram .....	10
3.3 Activity Diagram .....	11
3.3.1 Activity Diagram Legend: .....	11
3.3.2 Activity Diagram: Login .....	12
3.3.3 Activity Diagram: Record Hours.....	13
3.3.4 Activity Diagram: Create Evaluation .....	14
3.4 Website Maps .....	15
3.4.1 Website map Legend .....	15
3.4.2 Website Map: Main Page.....	16
3.4.3 Website Map: Employee.....	17
3.4.4 Website Map: Team Leader .....	18
3.4.5 Website Map: Supervisor.....	19
3.4.6 Website Map: System Administrator.....	20
4. Dataflow Diagrams.....	21
4.1 Data Flow Legend.....	21
4.2 Context Diagram .....	22
4.3 Level 0 Diagram.....	23
4.4 Level 1, Level 2, and Level 3 Diagrams.....	24
4.4.1 Log In.....	24

4.4.2 Average Hours.....	25
4.4.3 Create Evaluation.....	26
4.4.4 Record Hours.....	27
4.4.5 Confirm Team Hours.....	33
4.4.6 Create Team Leader Evaluation.....	34
4.4.7 Record Attendance.....	35
4.4.8 Send Email Notification.....	36
4.4.9 Set Up Projects.....	37
5. Structure Diagram.....	38
5.1 Login Structure.....	38
5.2 Employee Structure.....	39
5.3 Team Leader Structure.....	40
5.3.1 Team Leader Hours Structure.....	41
5.3.2 Team Leader Evaluations Structure.....	42
5.3.3 Team Leader Attendance Structure.....	42
5.4 Supervisor Structure.....	43
5.4.1 Supervisor Select Project Structure.....	44
5.4.2 Supervisor Supervised Employees Structure.....	45
5.5 System Administrator Structure.....	46
5.5.1 System Administrator Hours Structure.....	47
5.5.2 System Administrator Evaluations Structure.....	48
5.5.3 System Administrator Attendance Structure.....	48
5.5.4 System Administrator Admin Access Structure.....	49
6. Functional Requirements Inventory.....	50
6.1 Employee.....	50
6.2 Team Leader.....	50
6.3 Supervisor.....	51
6.4 System Administrator.....	51
7. Logical Data Dictionary.....	52
8. Prototype Screens.....	60
8.1 Record Hours.....	60
8.2 View hours.....	62

- 8.3 System Administrator Home Page ..... 63
- 8.4 System Controls Page..... 64
- 9. Testing Plan ..... 65
  - 9.1 Overview and Strategy ..... 65
  - 9.2 Acceptance Test ..... 65
  - 9.3 Unit Tests ..... 65
    - 9.3.1 Test Cases..... 65
    - 9.3.2 Unit Test: Directory ..... 66
    - 9.3.3 Unit Test: Login ..... 67
    - 9.3.4 Unit Test: Average Hours ..... 69
    - 9.3.5 Unit Test: Record Hours ..... 71
- 10. Development Environment and Production Environment ..... 74
  - 10.1 Development Environment..... 74
  - 10.2 Operating Environment ..... 74
  - 10.3 Maintenance ..... 74
- 11. Appendices..... 75
  - 11.1 Appendix A: Glossary of Terms ..... 75
  - 11.2 Appendix B: Timeline ..... 77

## 1. Summary of the Client's Problem

The client, Dr. Fryling is looking for a system that can keep track of various data that her employees are required to keep track of. This data includes the hours the employees work on certain projects and evaluations the employees create on their team members. The system the client needs will also require a hierarchy that gives certain employees different abilities within the system. The top of the hierarchy will be the System Administrator. This user will have the rights to edit and view all of the data within the system. The Team Leader user will need the capabilities to see and edit their team member's hours, give evaluations on their team members, and send email notifications to their team members. The Supervisor user will be able to see the hours of their subordinates, see the evaluations created about their subordinates, and will also be able to send email notifications to their subordinates. The client is looking for this system to be easy to use so that the employees can easily keep track of the information.

## 2. Use Case Narratives

### 2.1 Employee

The Employee will log into PRISM with a username and password. All users are Employees. The Employee can be a System Administrator, a Team Leader, or a Supervisor. The Employee will log into the system and record the hours worked on a project. Comments on completed work during the hours recorded can be made by the Employee. The Employee can work on multiple projects. The Employee can view the average hours recorded of other teams if the other team's average hours are public. At the end of each phase of a project, the Employee must complete evaluations on every Employee that worked on the project as well as the Employee's Team Leader. The Employee can view the average score the Employee received from evaluations.

### 2.2 Team Leader

The Team Leader is an employee. The Team Leader can log into PRISM with a username and password. The Team Leader can log the hours the Team Leader works on projects. The Team Leader can make the team evaluations on the team members on their teams. The Team Leader can view the hours they have previously logged into PRISM. The Team Leader can lead multiple teams and can be an Employee on other teams. The Team Leader can also be a System Administrator or a Supervisor. The Team Leader has the capability of approving the hours completed by the Employees on the project that the Team Leader is responsible for. Team Leaders can create evaluations on the members of the Team Leader's team. The Team Leader can only see the hours that the Employees record for the project that the Team Leader is responsible for.

## 2.3 Supervisor

The Supervisor is an employee. The Supervisor can log into PRISM with a username and password. The Supervisor can log the hours the Supervisor works on projects. The Supervisor can make the team evaluations on the team members on their teams. The Supervisor can view the hours they have previously logged into PRISM. There can be multiple levels of Supervisors. The Supervisor can have other Supervisors, Team Leaders, or Employees below them in the system. The Supervisor can see the hours that the Supervisor's subordinates record. The Supervisor cannot see team project hours unless the Team Leader has made the project hours public. The Supervisor can see all evaluations made on the Supervisor's Employees. The Supervisor can take on multiple roles such as Team Leader or System Administrator.

## 2.4 System Administrator

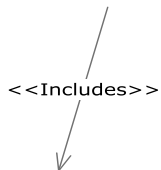
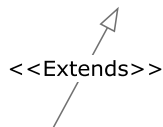
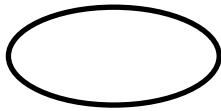
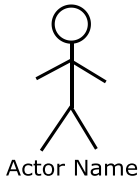
The System Administrator is an employee. The System Administrator can log into PRISM with a username and password. The System Administrator can log the hours the System Administrator works on projects. The System Administrator can make the team evaluations on the team members on their teams. The System Administrator can view the hours they have previously logged into PRISM. The System Administrator will have the responsibility of maintaining and configuring PRISM. The System Administrator will have access to adding and manipulating all accounts within PRISM. The System Administrator has the capability to add projects to the systems as well as adding all employees and team leaders to the projects. The System Administrator also has the capability to remove any employees and team leaders from projects. There can be multiple System Administrators for PRISM. System Administrators can also be Supervisors, Team Leaders, and Employees. System Administrators has access to all privileges and information recorded in PRISM. Email notifications about items due, such as recorded hours and evaluations, will be generated by the System Administrator and distributed to the appropriate Employees.

## 3. UML Diagrams

### 3.1 Use Case Diagram

UML Use Case Diagrams are used to give an overview of the functionality of a system. The diagram is based off of the use case narratives. The diagram is a brief snapshot of the use case narratives. The UML Use Case Diagram shows how each user interacts with the system and other users. The diagram also displays what actions each user can perform.

#### 3.1.1 UML Use Case Legend



**System Boundary:** This is where all of the interactions occur. Represents what is within the system and outside of it. Scenarios go on the inside and actors go on the outside.

**Actor:** Actors interact with the system through scenarios. Actors can be human or non-human. Human actors go on the left side of the system boundary. Non-human actors go on the right side. Actor name gets displayed below the actor.

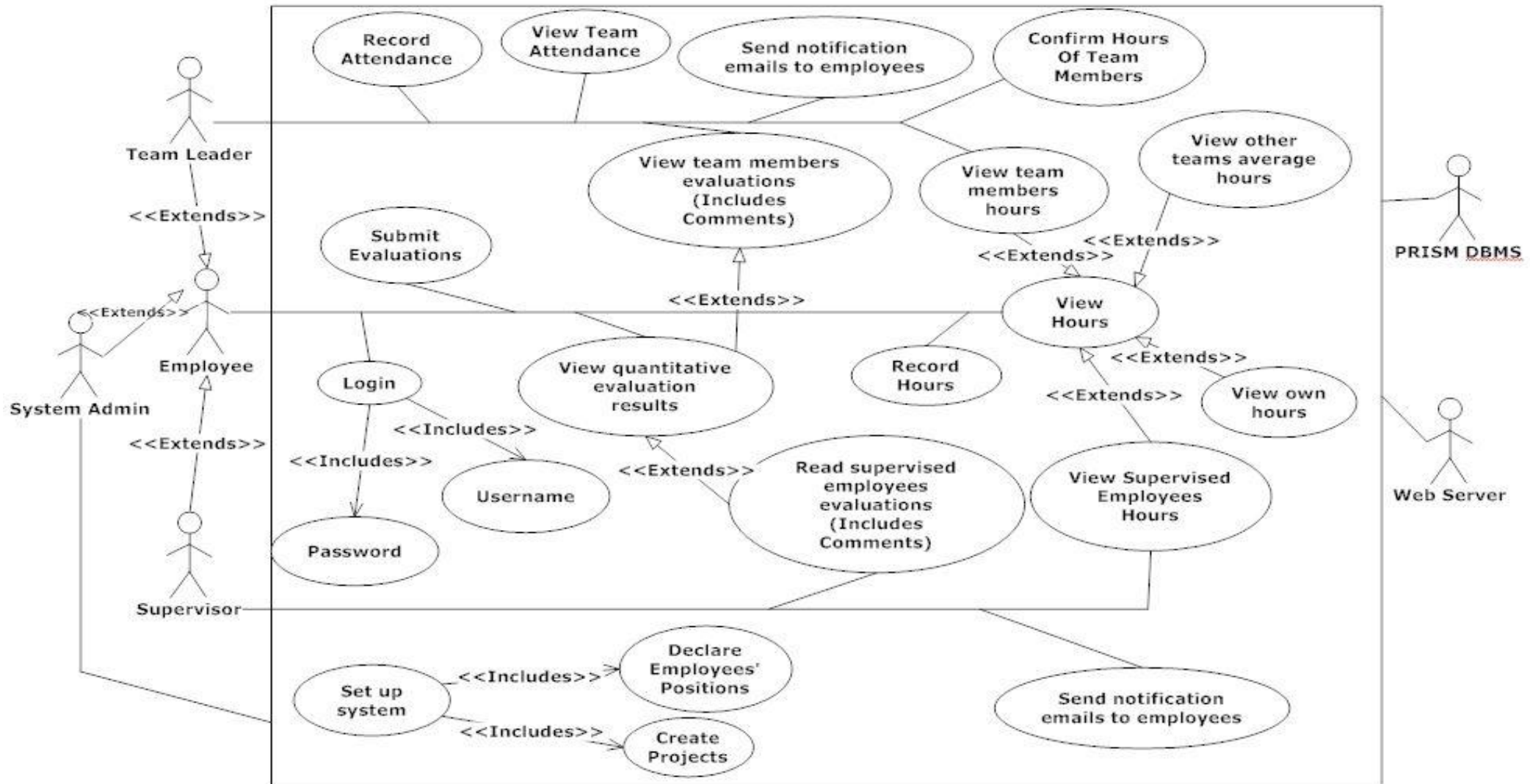
**Scenarios:** The actions that occur within a system and how the user interacts with the system.

**Participation Line:** Shows what scenarios an actor can interact with.

**Inheritance Arrow:** An arrow that points from one scenario to another. The scenario being pointed at is the parent and the other is the sub.

**Inclusion Arrow:** An arrow that points from a scenario to another scenario to show that something must be included for the scenario.

### 3.1.2 Use Case Diagram





## 3.2 Deployment Diagram

A deployment diagram in the Unified Modeling Language (UML) models the physical deployment of devices and execution environments for a system. The PRISM Deployment Diagram represents the physical design of the system. The diagram shows that PRISM will be connected to the internet via HTTP. The database that is used will be connected to PRISM via ODBC. The development environment will be connected to PRISM via SCP.

### 3.2.1 Deployment Diagram Legend

<HTTP>

**HTTP** - Hypertext Transfer Protocol defines how messages are formatted and transmitted, and what actions web servers and browsers should take in response to various commands.

<SCP>

**SCP** - Securely transfers computer files between a local host and a remote host

<ODBC>

**ODBC** - Open Database Connectivity is a standard programming language middleware for accessing database management systems.

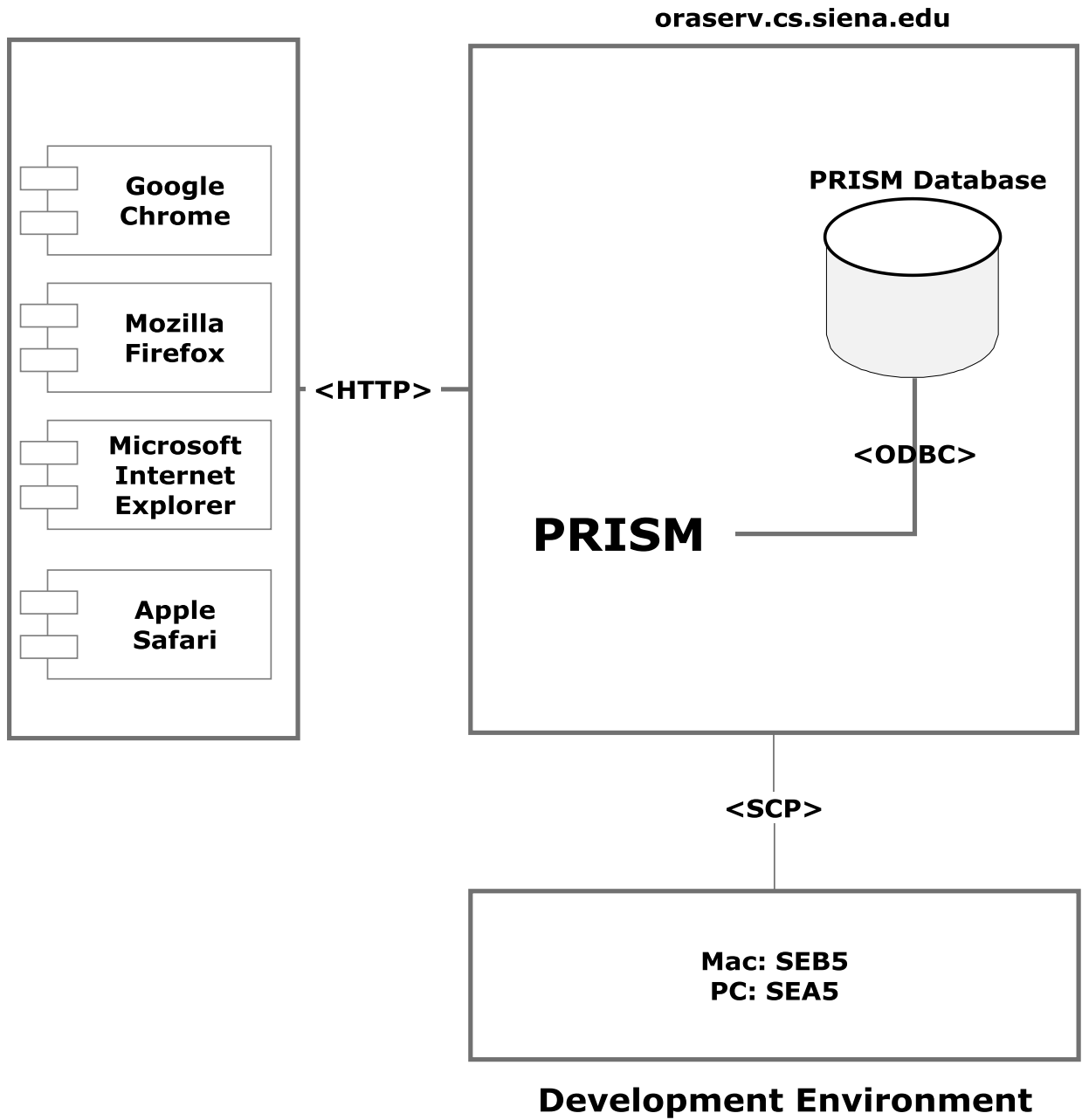


**System Boundary** - This is where all the interactions occur. Represents what is within the system and outside of it.



**Connection** - Displays a relationship between boundaries.

### 3.2.2 Deployment Diagram



### 3.3 Activity Diagram

A UML Activity Diagram is a step by step diagram of the actions within a process. Each step will start the process, end the process, perform an action, or make a decision. The activity diagrams can also display the objects that will be needed and will be given to users of the process. Below are the nodes and other objects used in the UML Activity Diagrams for PRISM's main processes.

#### 3.3.1 Activity Diagram Legend:



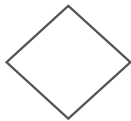
**Initial Node** – This is the first node in the process. The initial node is the starting point for all movement.



**Final Node** – The final node is the last node in the process. When the activity flow has reached here, the process is over.



**Activity Node** – The activity node describes the activity or step to be done to help complete the process when the flow reaches the node.



**Decision Node** – The decision node is used to branch the activity. A decision node is usually posed in question form, with multiple unique answers. The flow must follow one of the branches after the decision.



**Data Object** - Data that is used during the process. The data object can be either input or output.

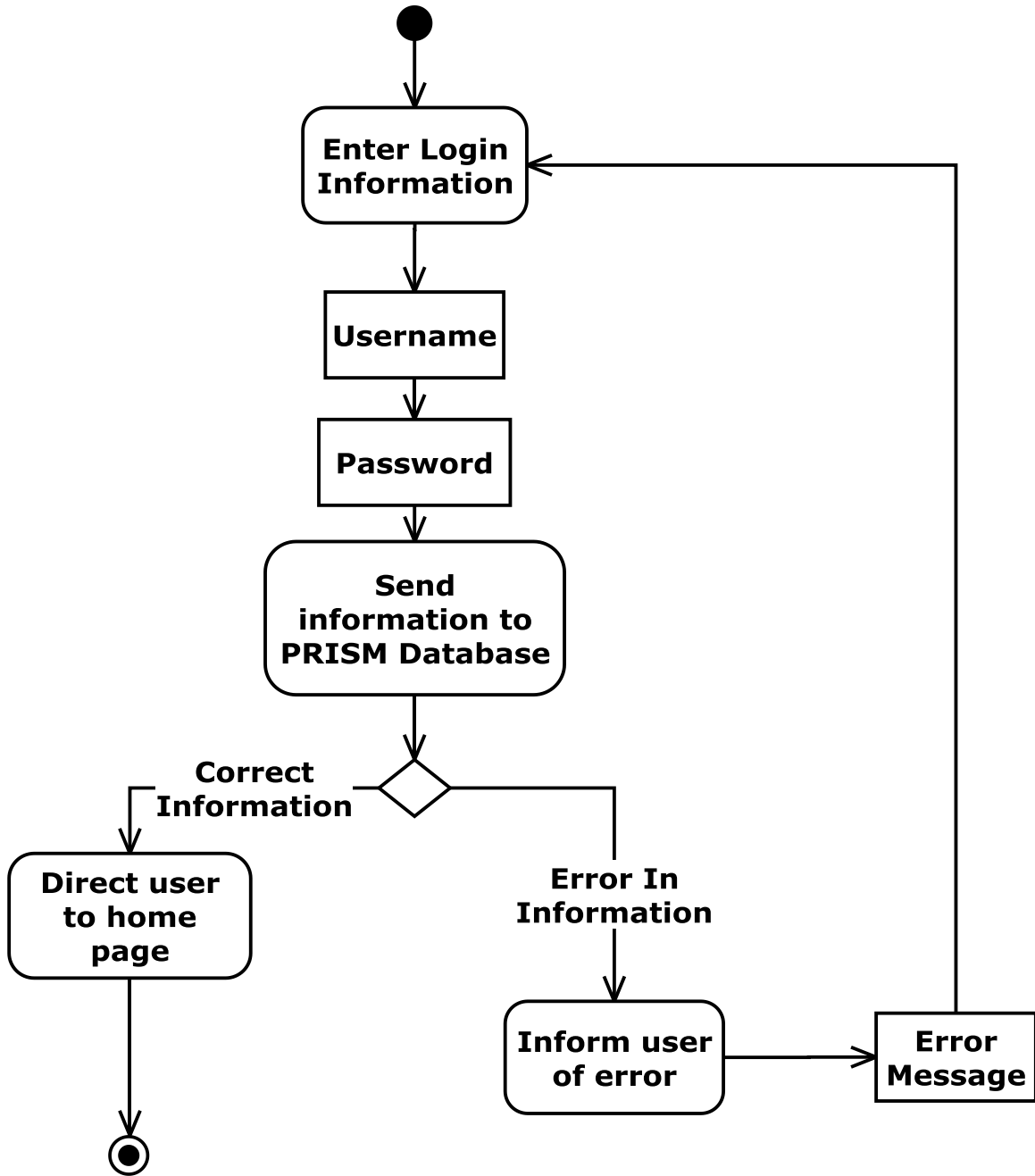


**Split/Join** – Can either separate activity flow to run two activities simultaneously or join them back together after simultaneous activities are completed.

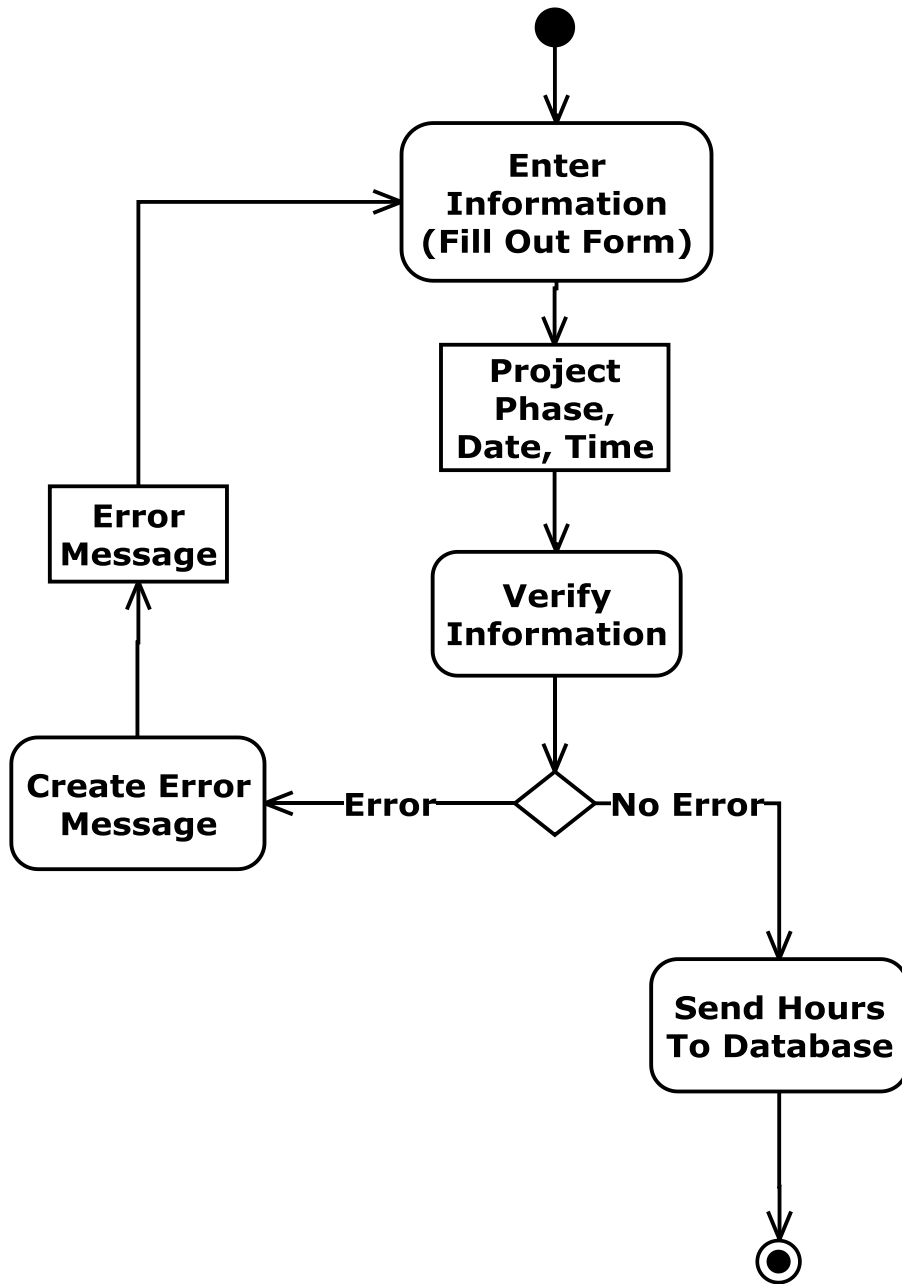


**Flow** – Shows the movement of action from one node to another.

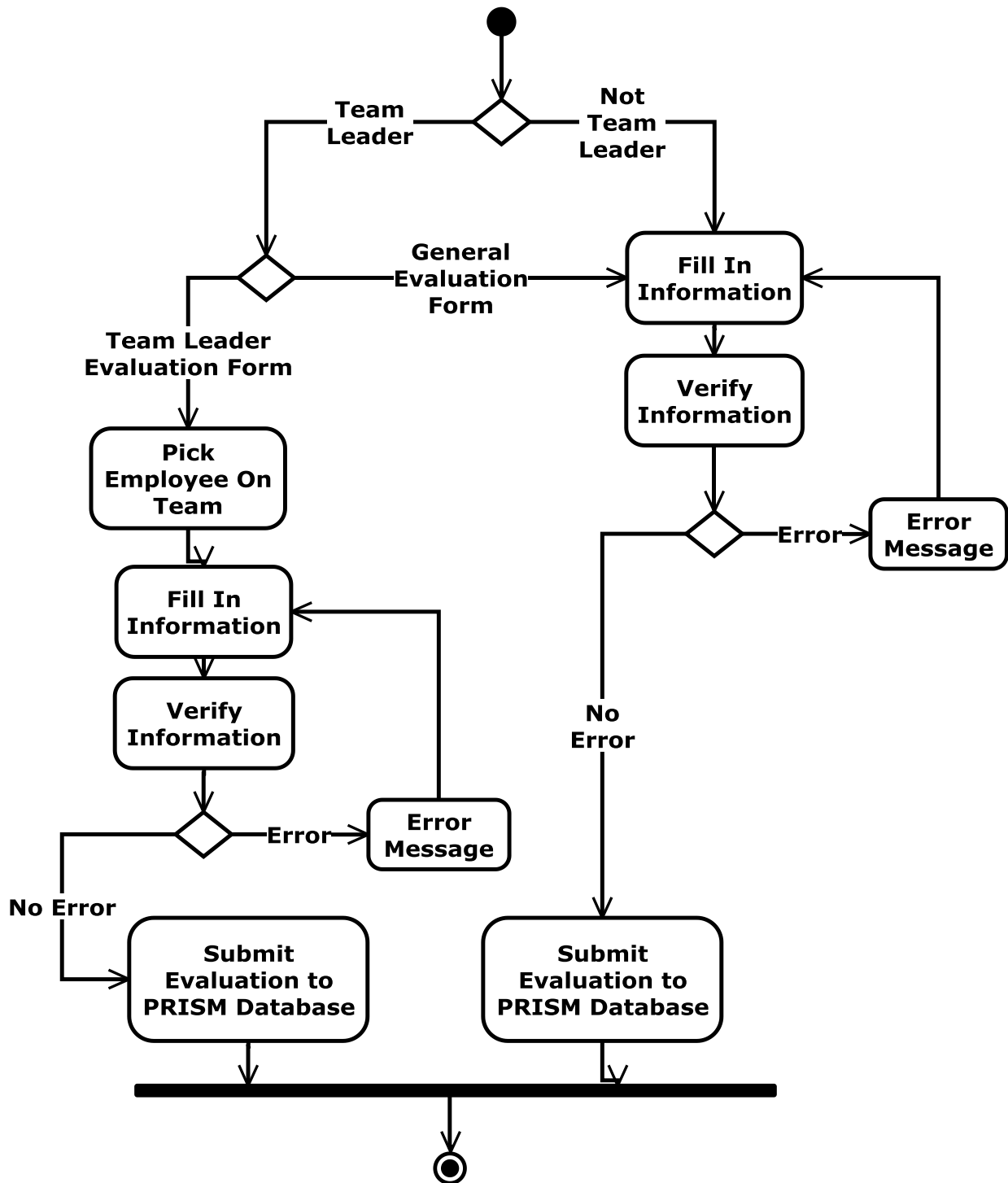
### 3.3.2 Activity Diagram: Login



3.3.3 Activity Diagram: Record Hours



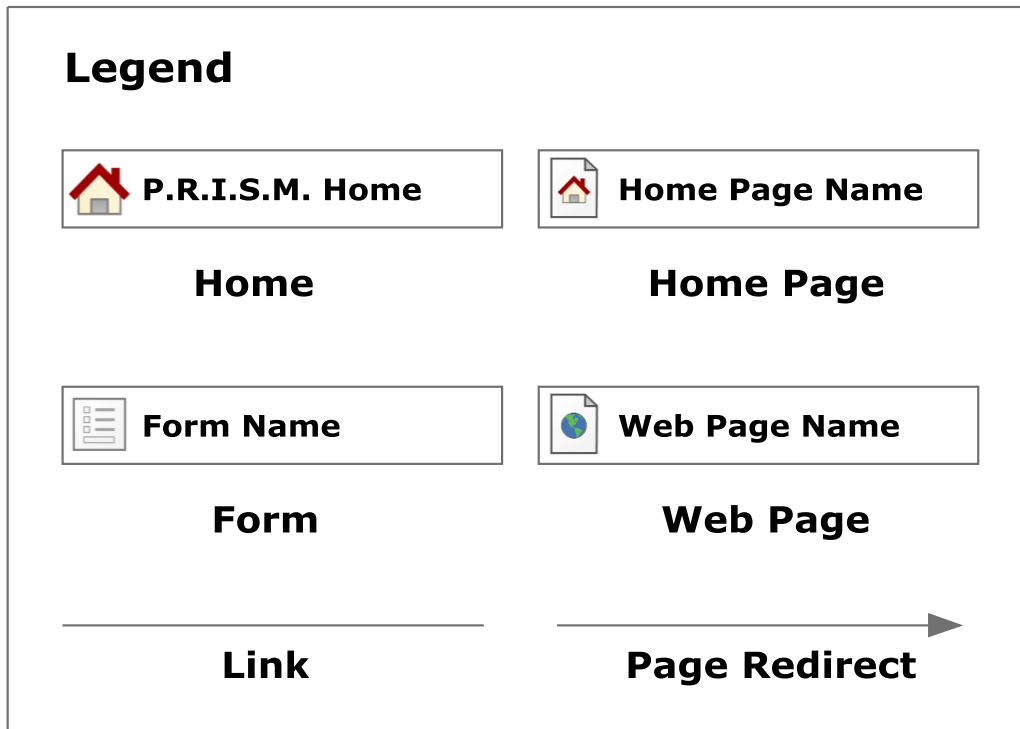
### 3.3.4 Activity Diagram: Create Evaluation



### 3.4 Website Maps

The web site map shows the structure of the PRISM website. The map outlines the links and forms associated with each as well as the functionality of these elements. Below are the various symbols associated with our web site map.

#### 3.4.1 Website map Legend



**Home** - This represents the main page of PRISM prior to a user logging in. Various options will be presented at this time.

**Home Page** - Represents the user's main page when they initially login to their account.

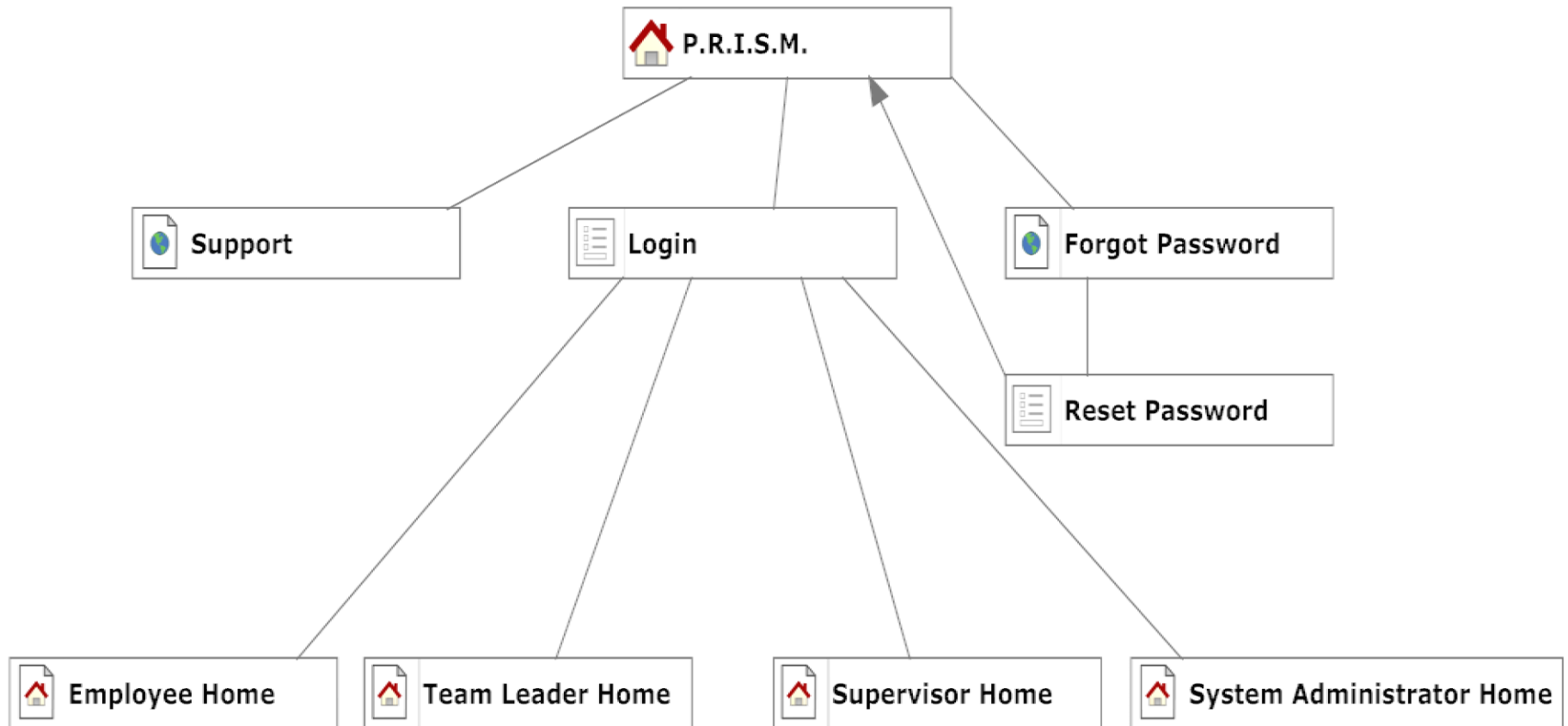
**Form** - Represents data fields which require user input.

**Web Page** - Represents a web page within the PRISM system.

**Link** - Represents a page being accessible from another page.

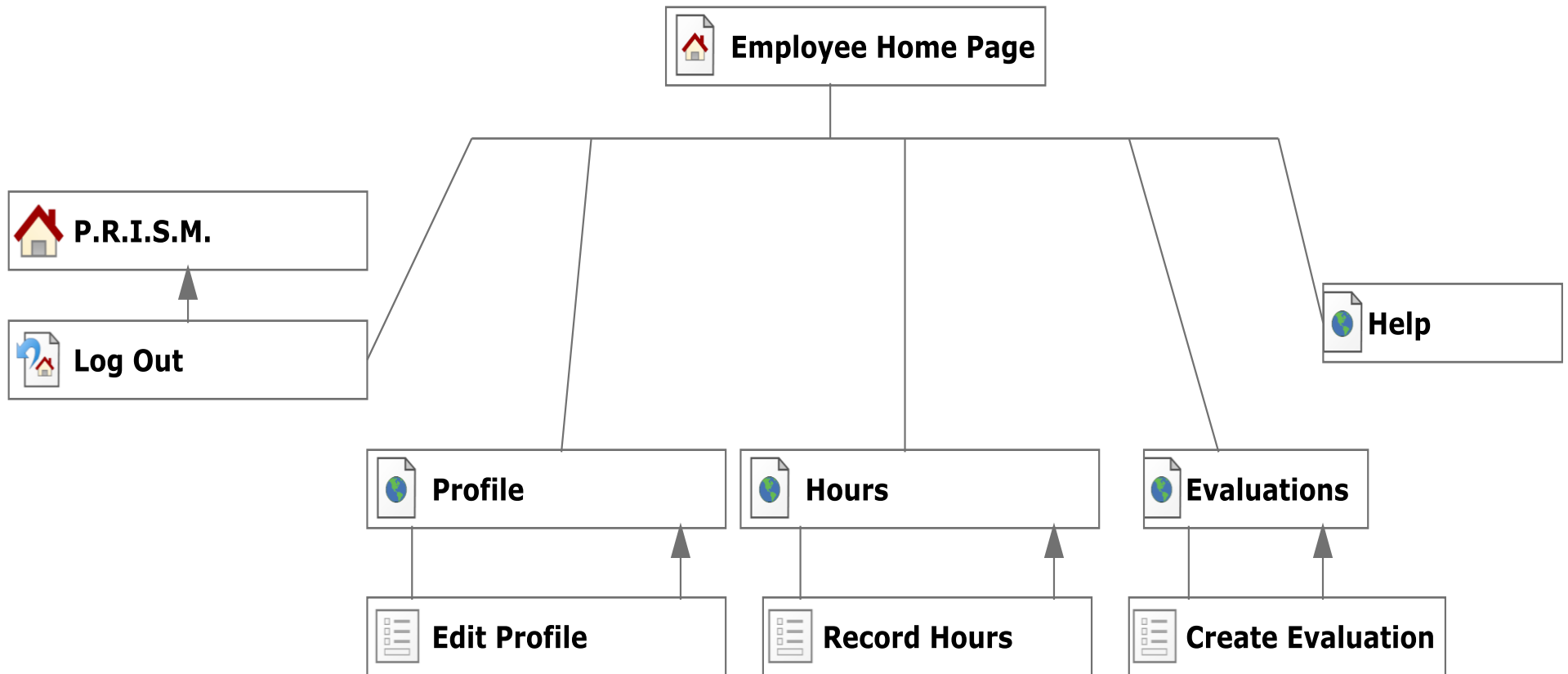
**Page Redirect** - Indicates a forced reroute to a new page depending on the user's action.

### 3.4.2 Website Map: Main Page

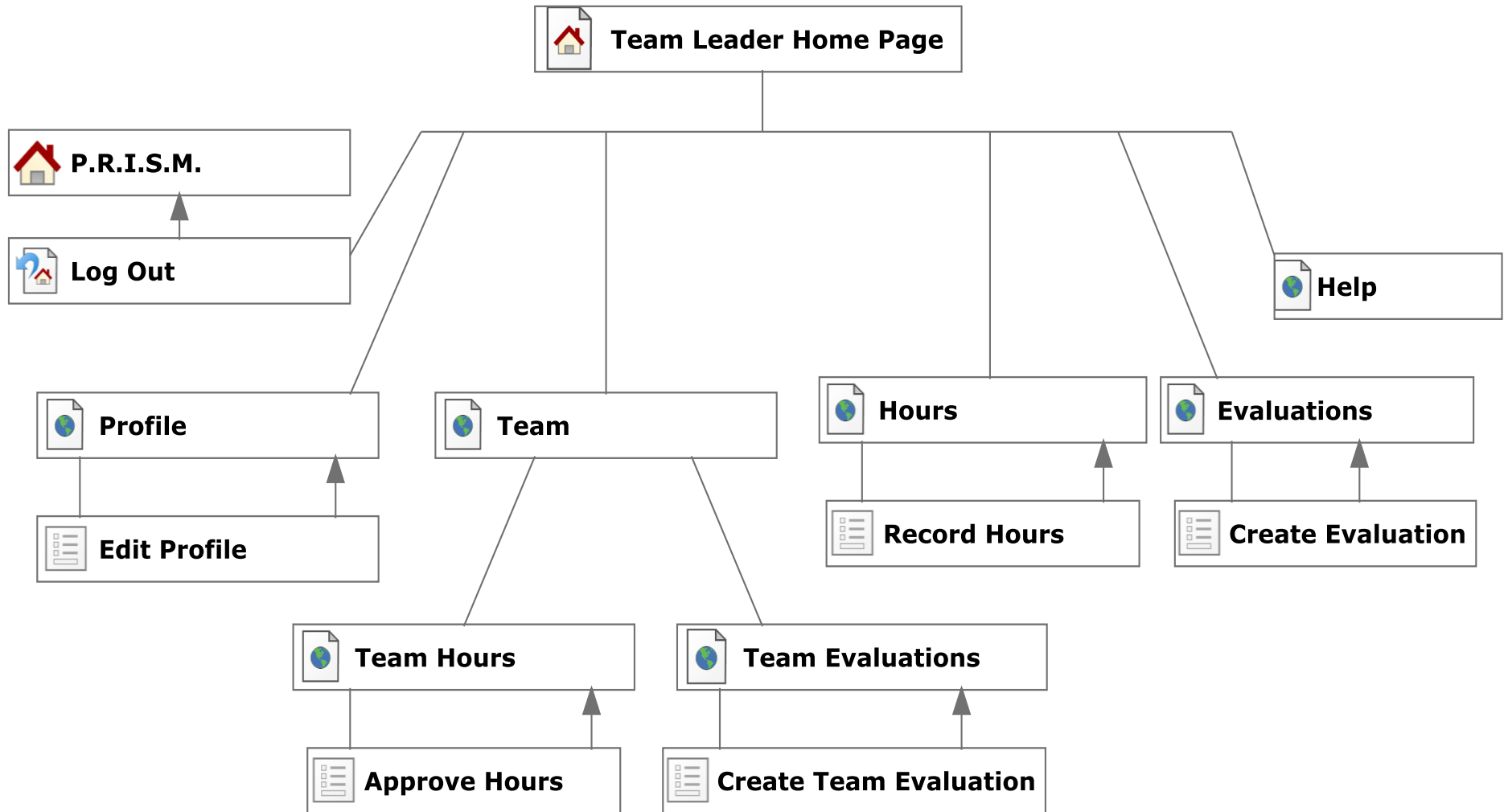




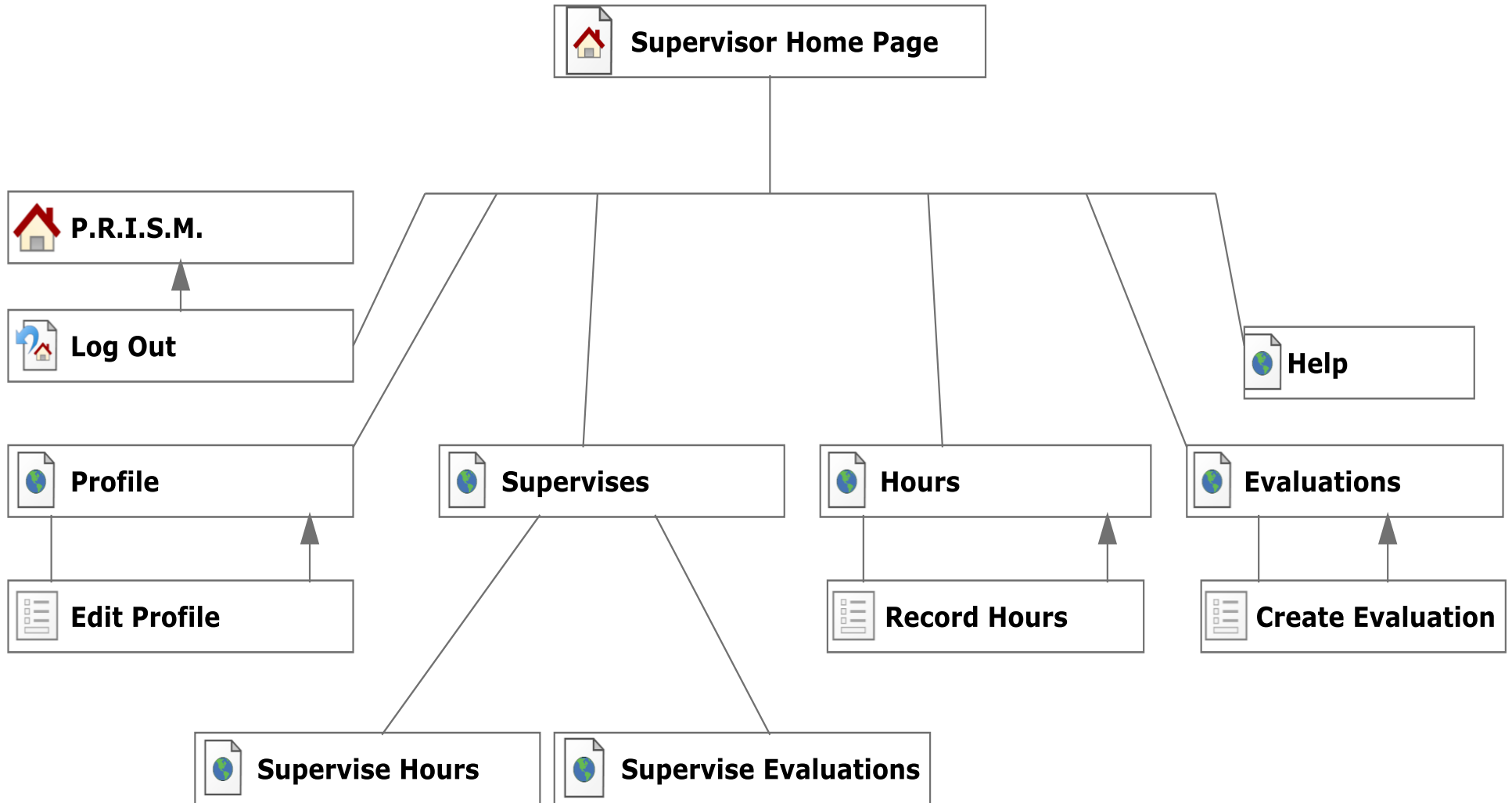
### 3.4.3 Website Map: Employee



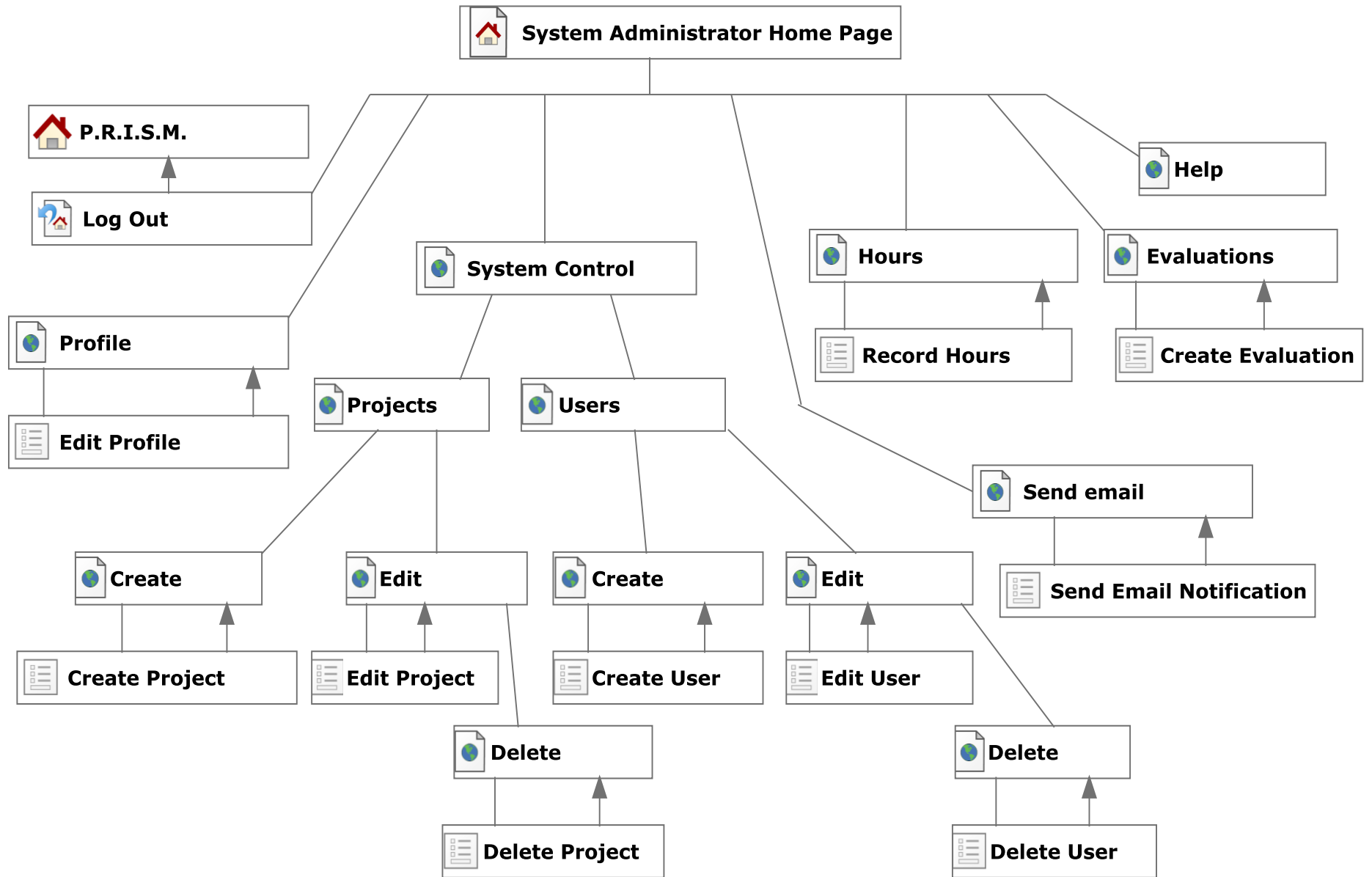
### 3.4.4 Website Map: Team Leader



### 3.4.5 Website Map: Supervisor



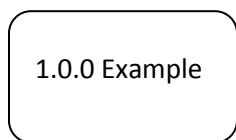
### 3.4.6 Website Map: System Administrator



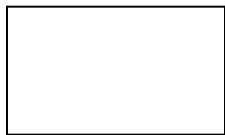
## 4. Dataflow Diagrams

### 4.1 Data Flow Legend

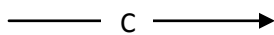
Data Flow Diagrams represent the movement of data between processes in the system as well as the movement of data between processes and external entities outside the system. The diagrams are a tool for analyzing the structure of the system and the ways in which data will be stored and retrieved by different processes. These diagrams model data flows at different levels of detail in the system. The following symbols will be used within the Data Flow Diagram:



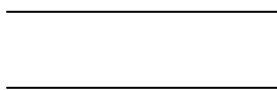
**Process** - Transforms or manipulates data where the number represents the stage in the system and the string 'Example' is the process.



**External Entity** - Contributes data or information to the system or which receive data/information from it.



**Data Flow** - Data/information flowing to or from a process where C is the data/information.



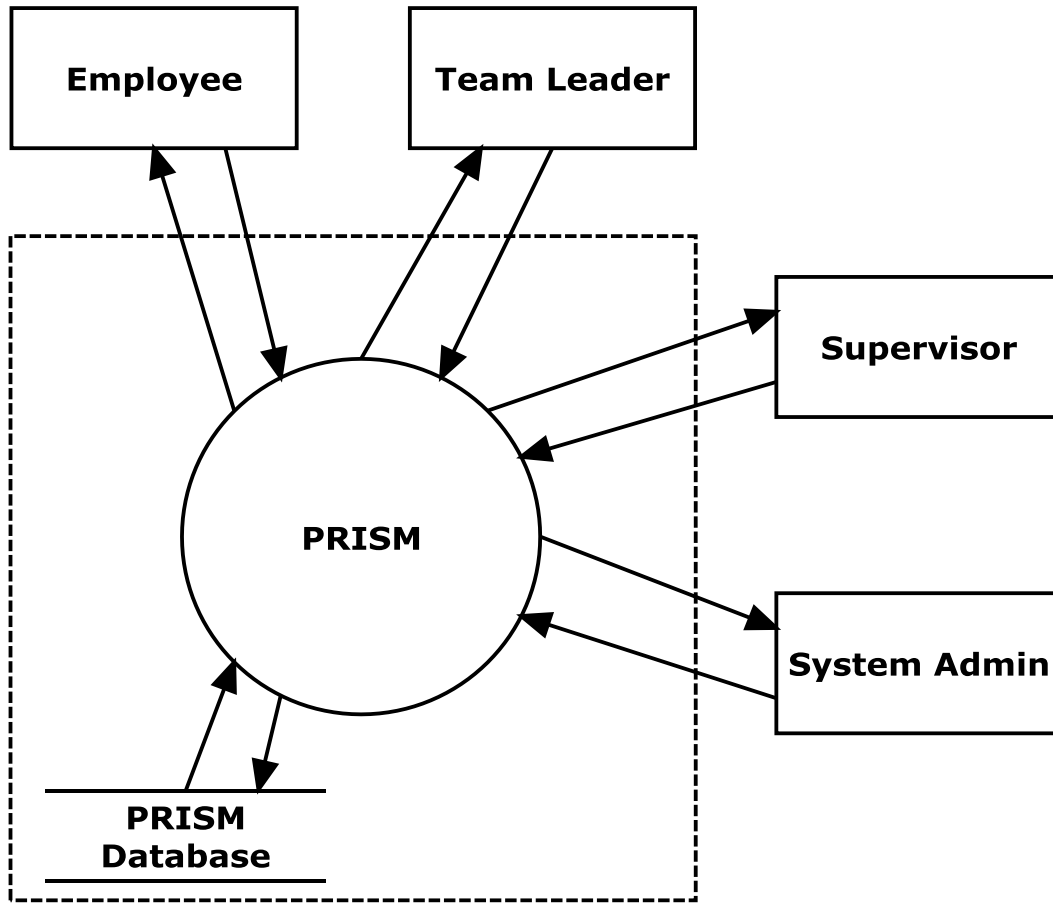
**Data Store** - Location where data is held temporarily or permanently.



**System Boundary** - This is where all of the interactions occur. Represents what is within the system and outside of it.

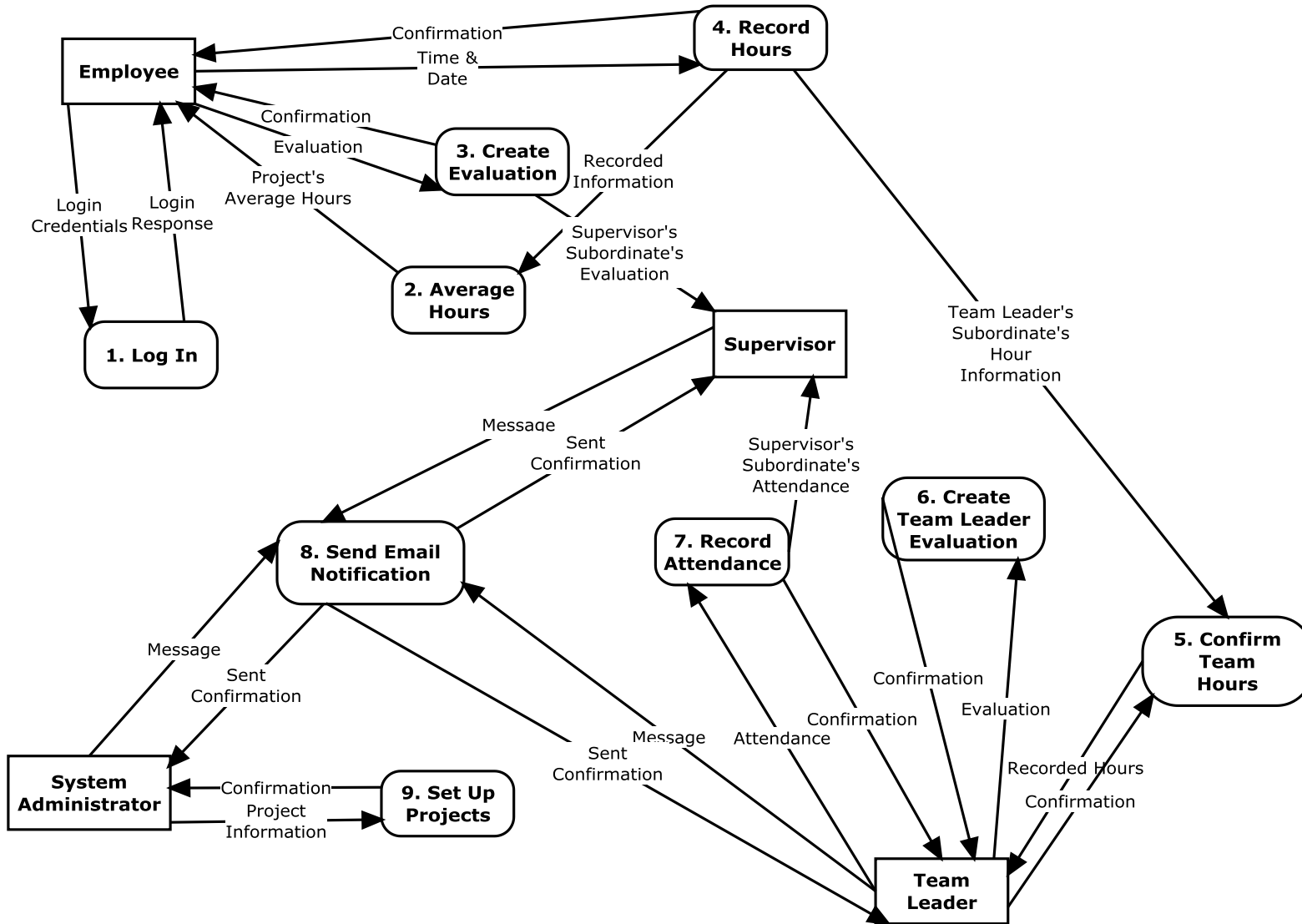
### 4.2 Context Diagram

Below is the Context Diagram which is an overall simplified view of the system and the system's interactions with the users and database.



### 4.3 Level 0 Diagram

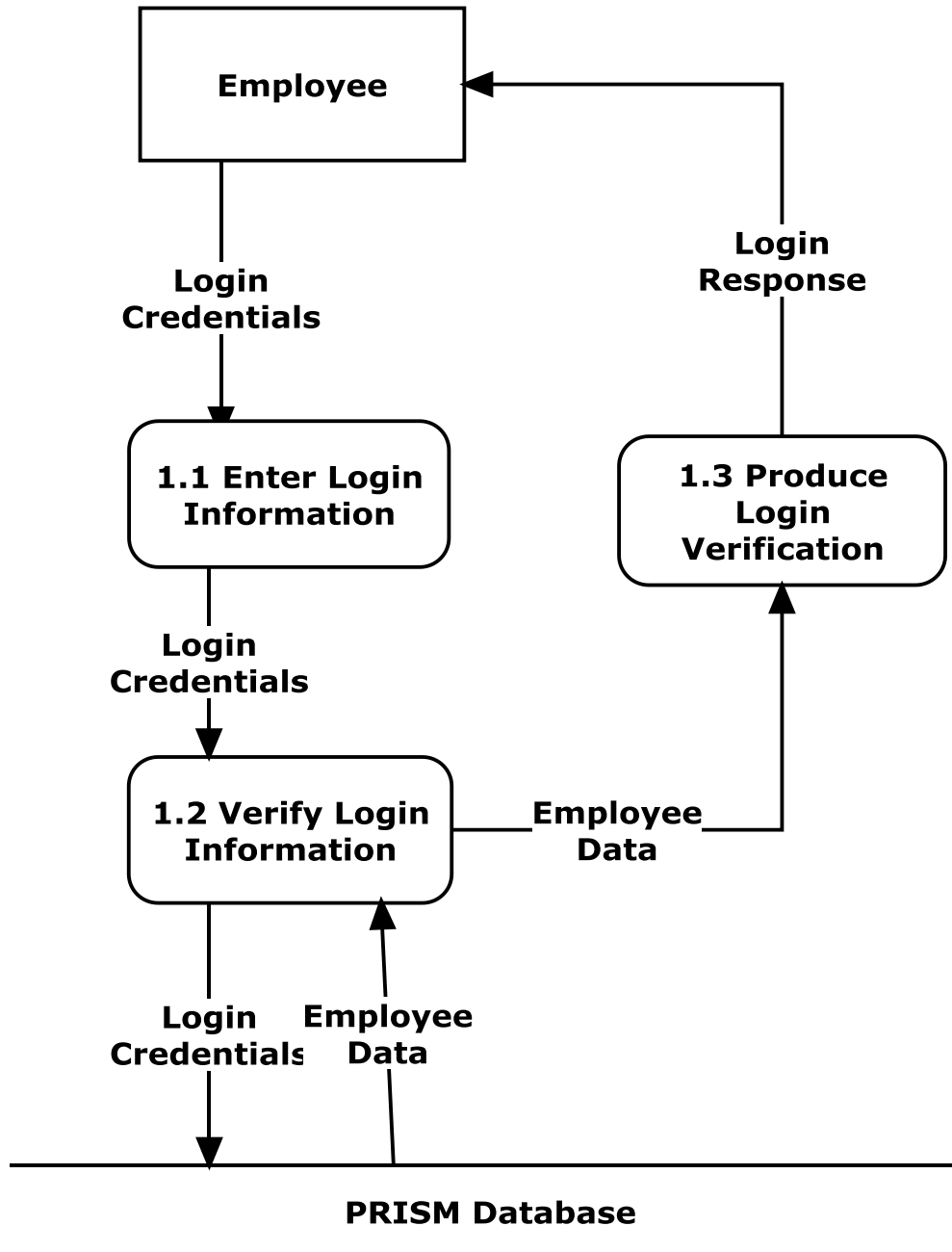
Below is the Level-0 diagram which shows the major subsystems shown in the Context Diagram.



### 4.4 Level 1, Level 2, and Level 3 Diagrams

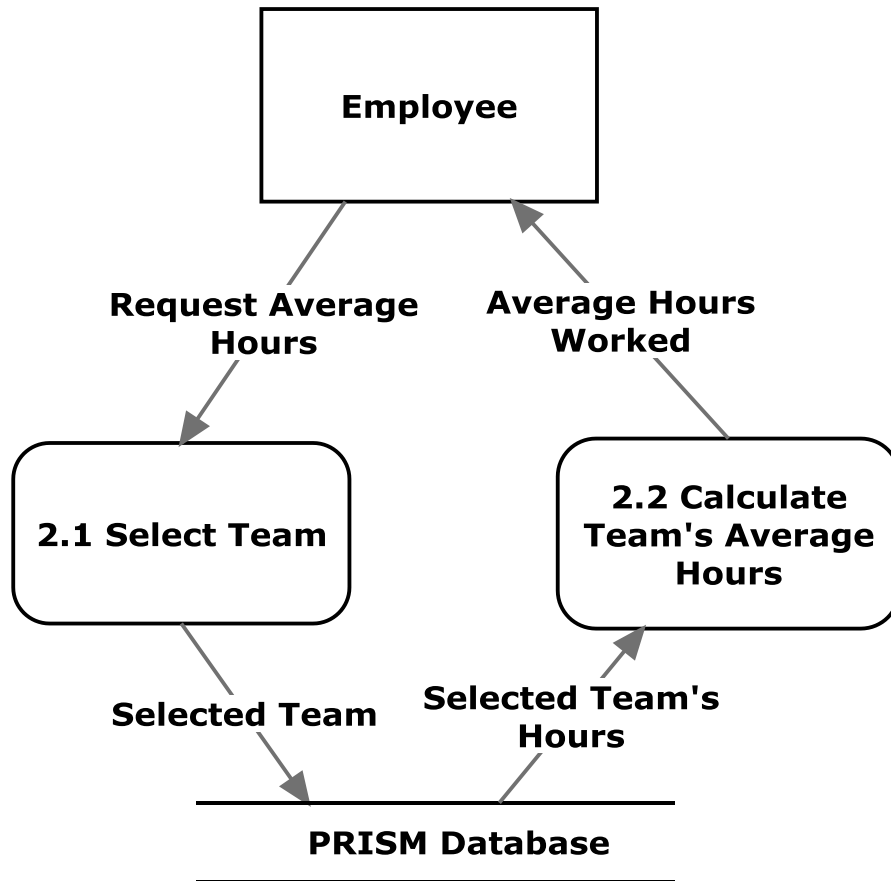
Below are Level-1 diagrams which show an expansion of processes from the Level-0 diagram.

#### 4.4.1 Log In

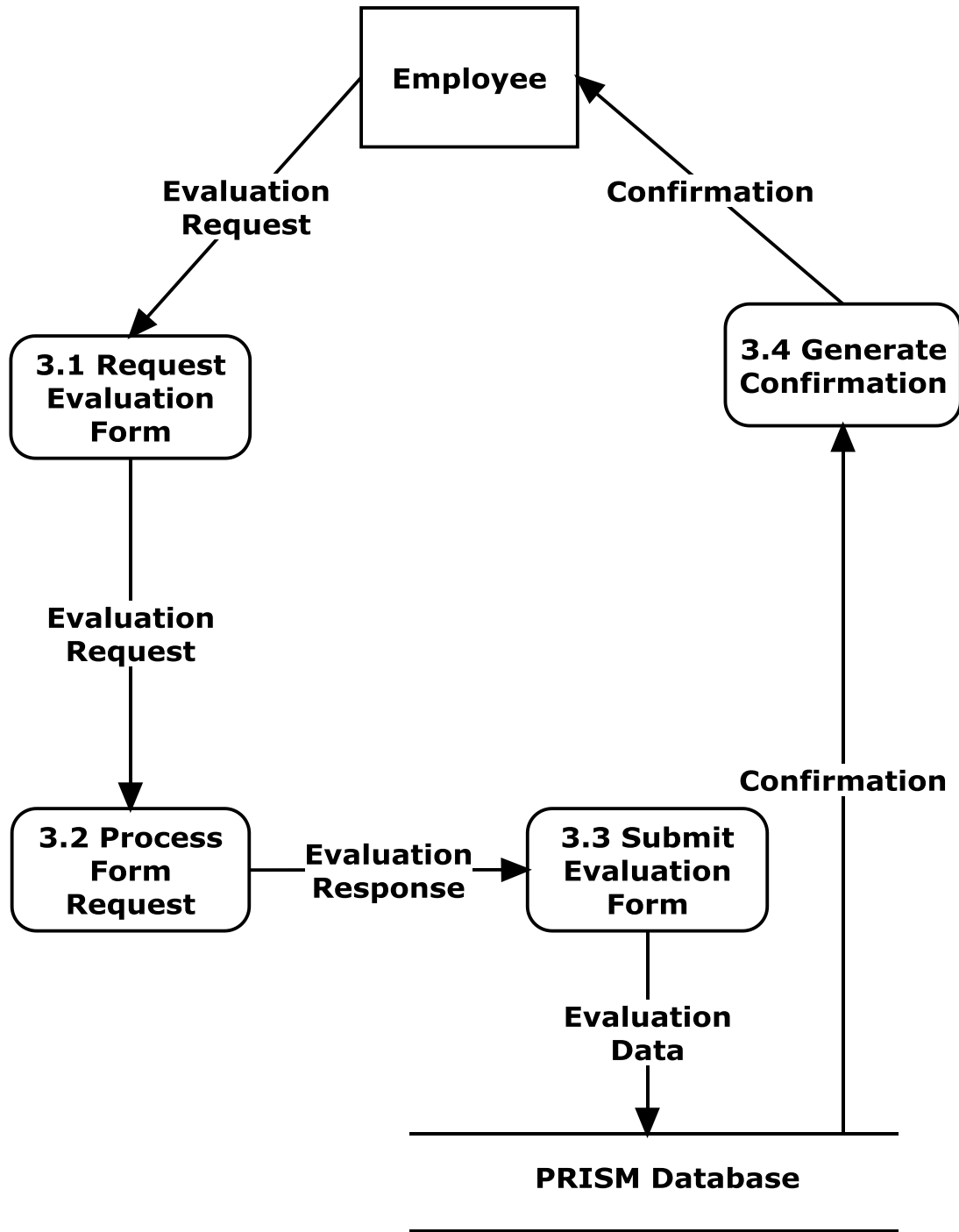




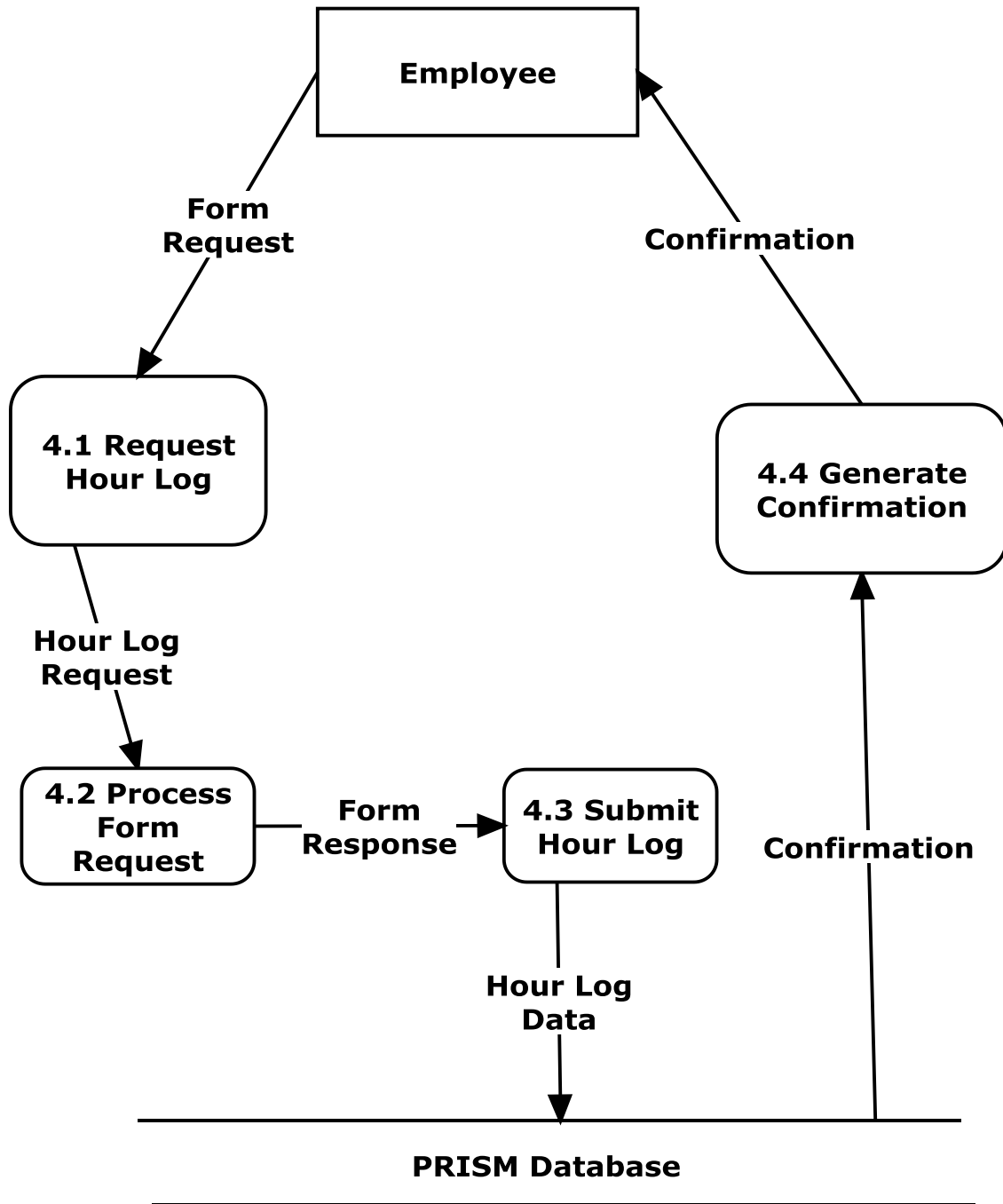
### 4.4.2 Average Hours



4.4.3 Create Evaluation

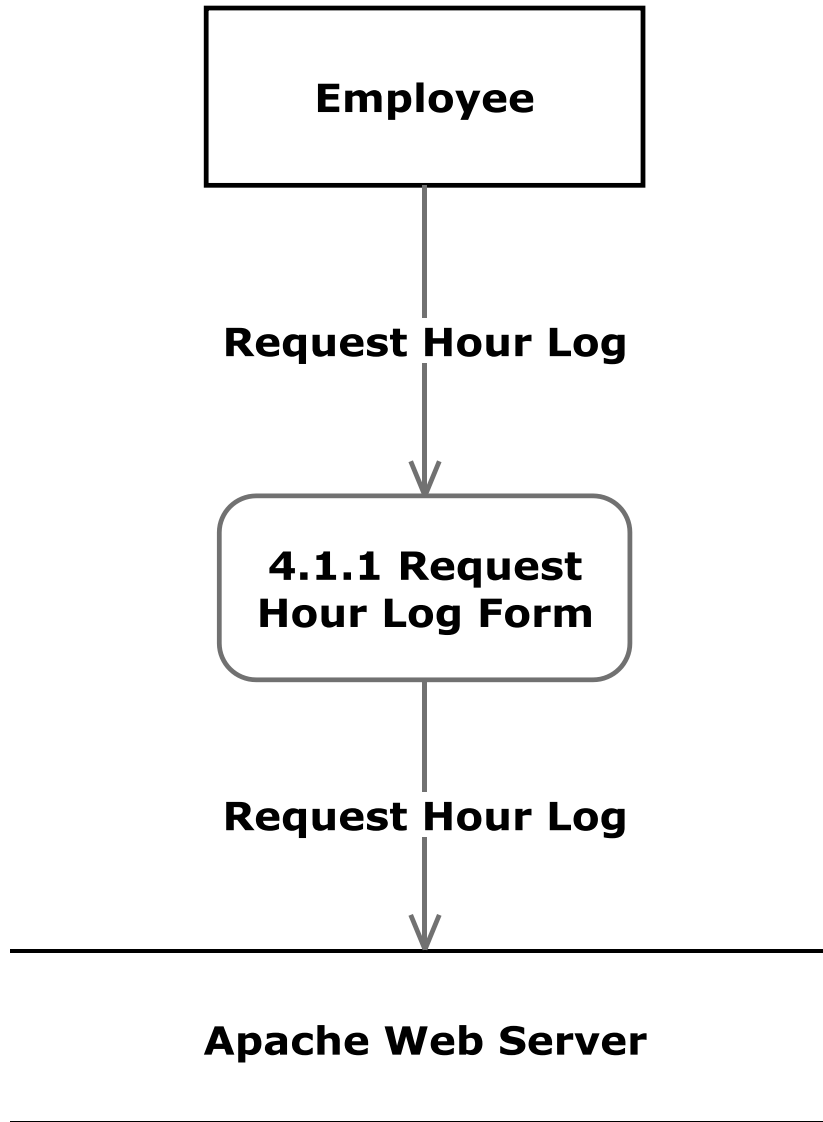


4.4.4 Record Hours

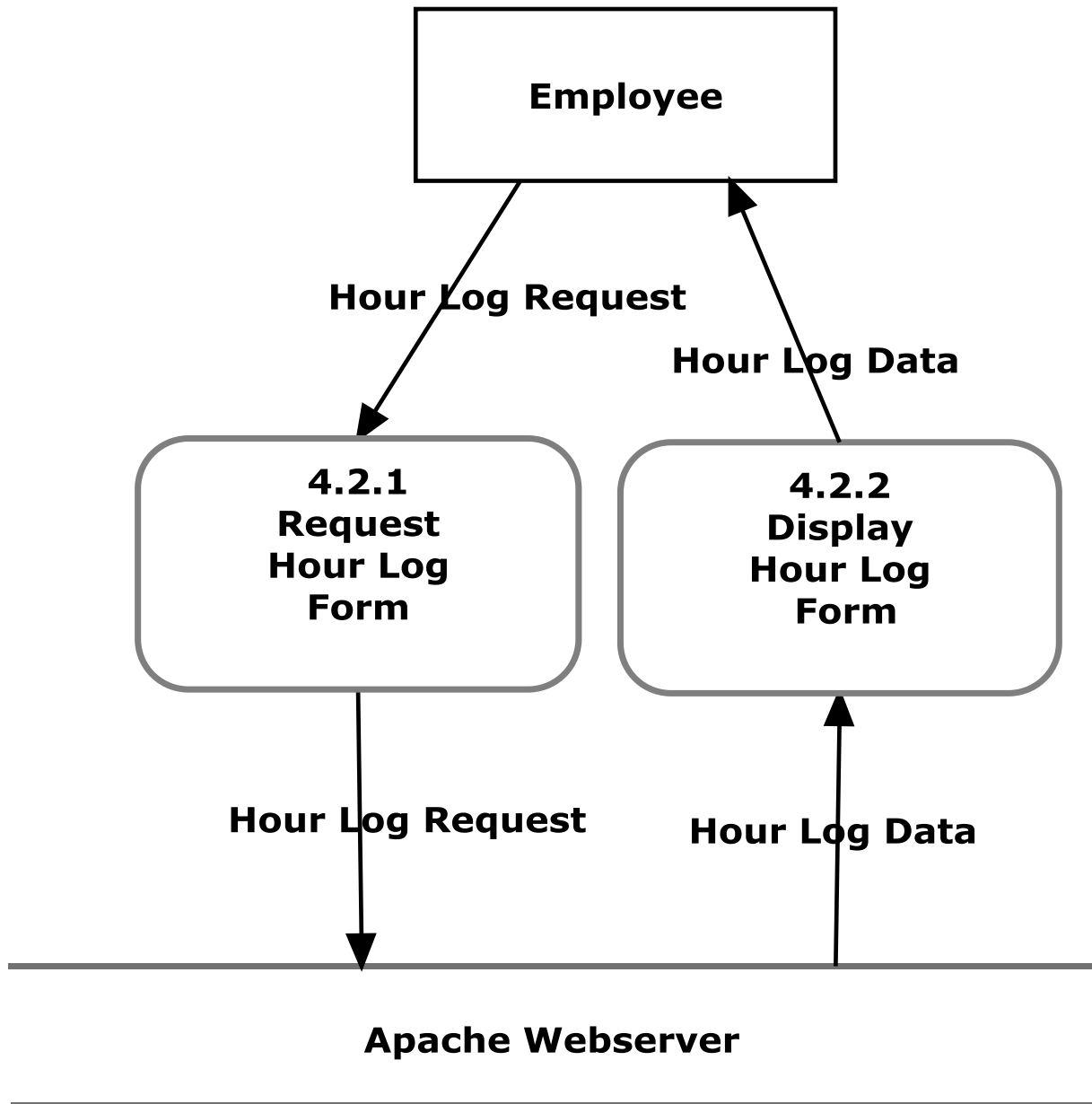


The following Level-2 diagrams demonstrate an even further expansion of processes in higher levels.

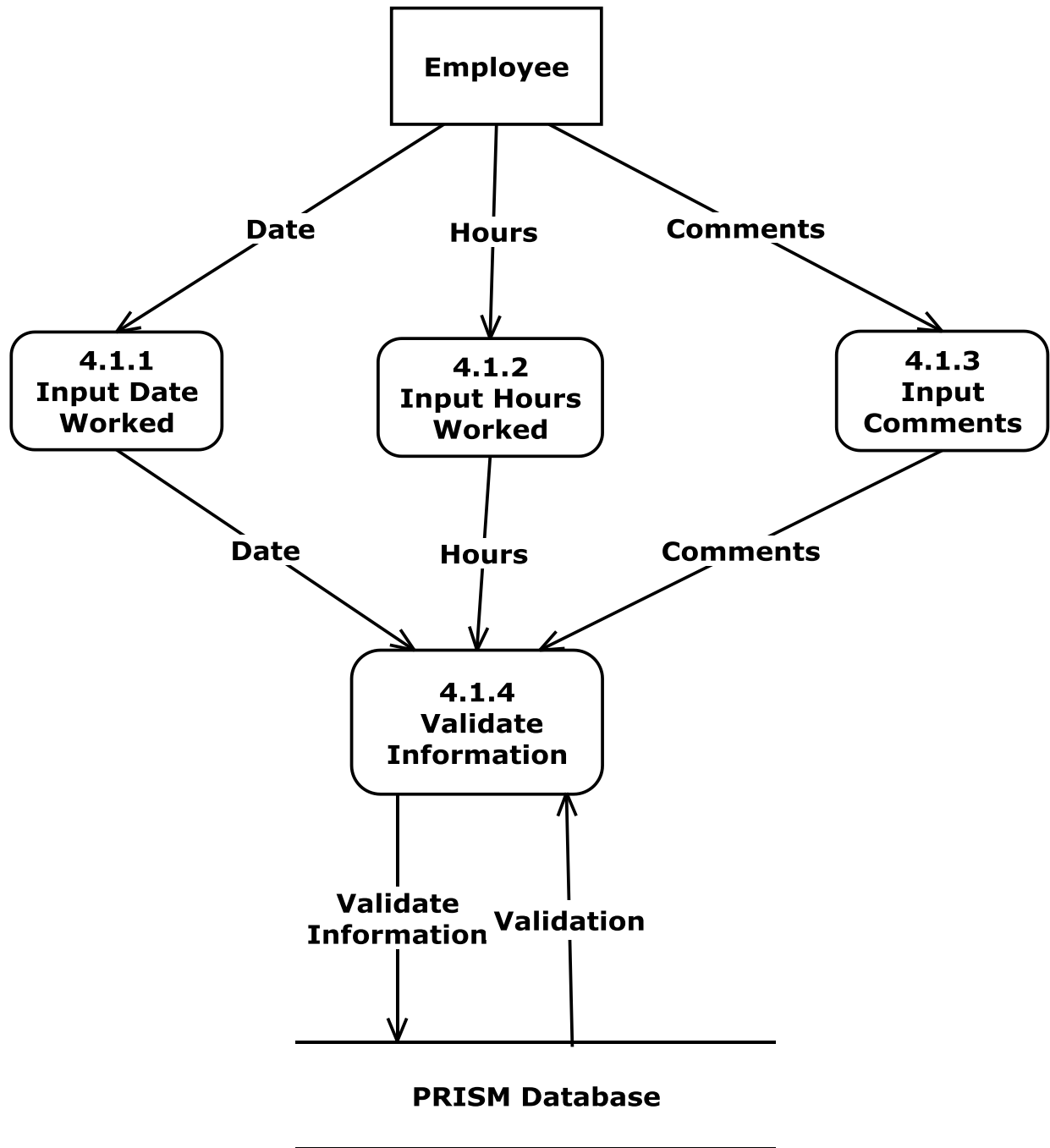
*4.4.4.1 Record Hours Level 2: Request Hour Log*



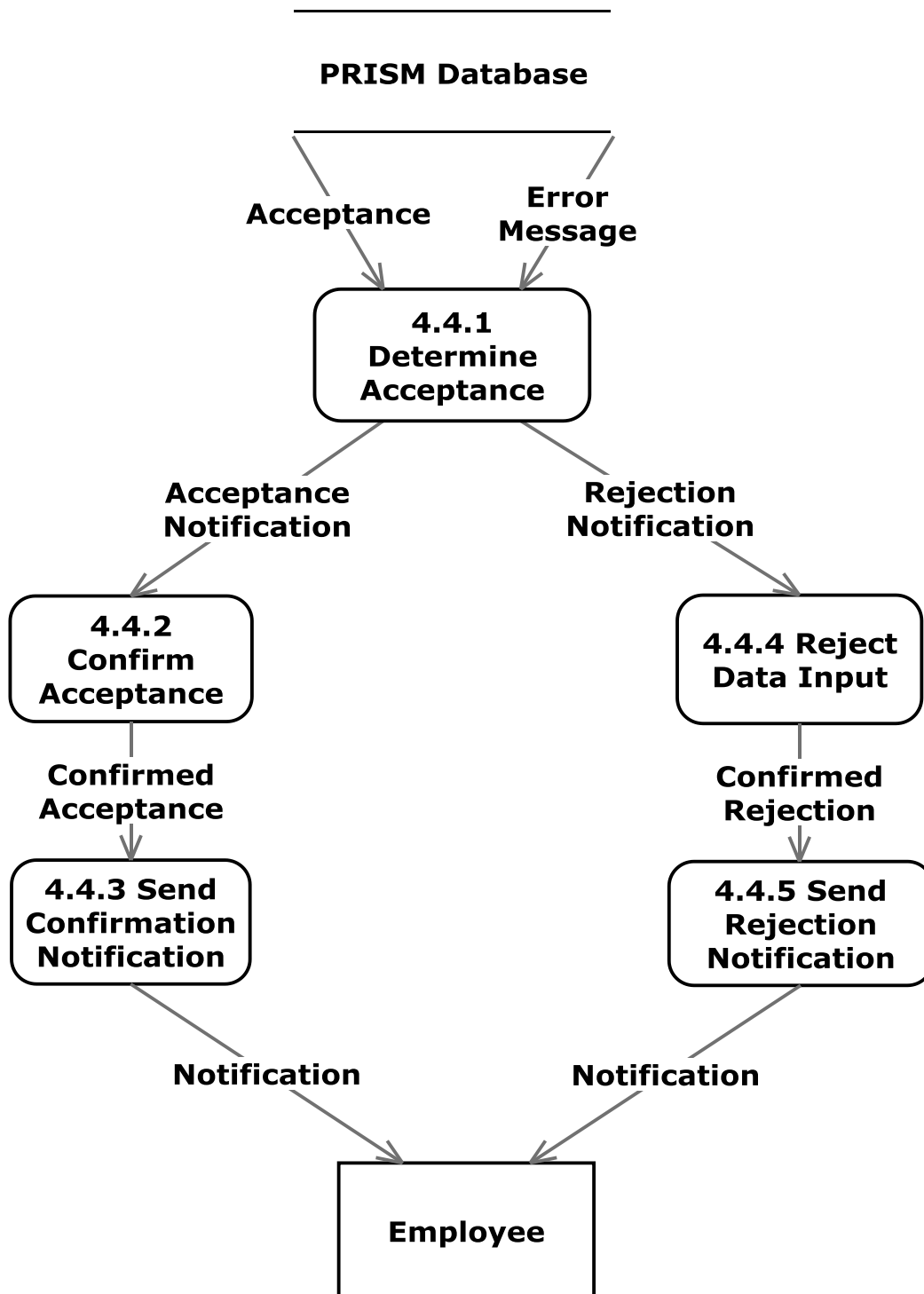
*4.4.4.2 Record Hours Level 2: Process Form Request*



4.4.4.3 Record Hours Level 2: Submit Hour Log

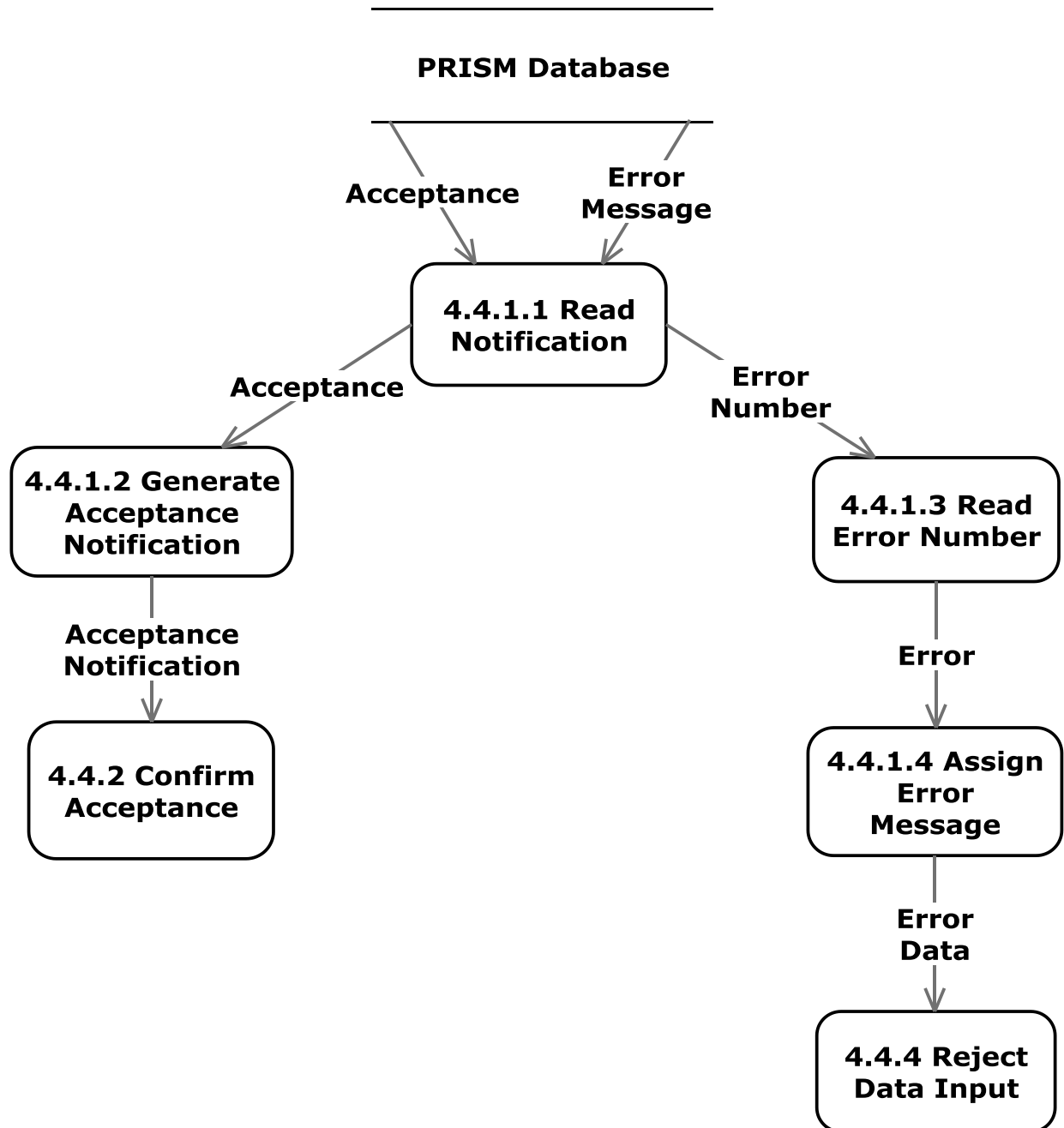


4.4.4.4 Record Hours Level 2: Generate Confirmation



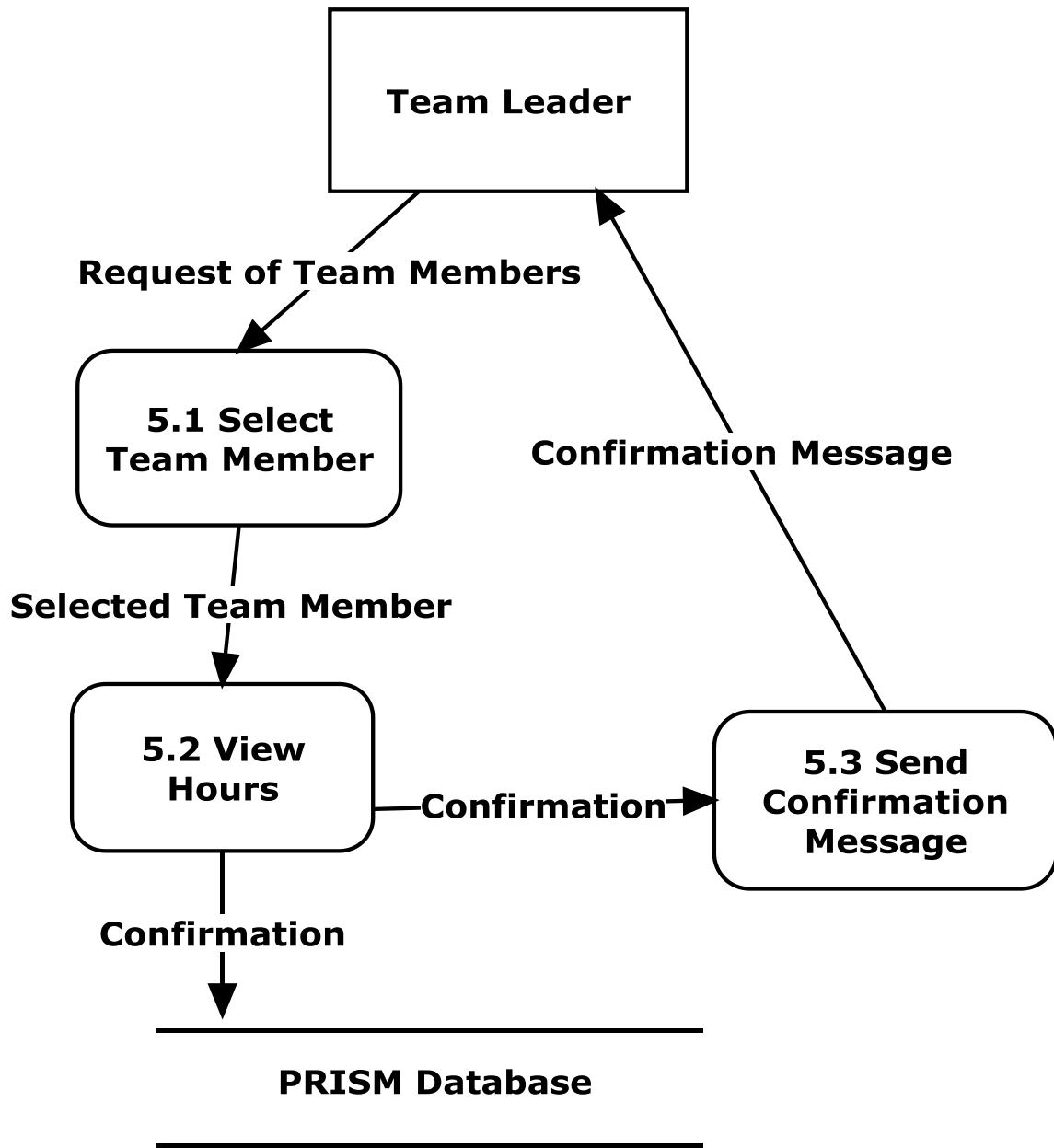
The following Level-3 diagram demonstrates an even further expansion of the Generate Confirmation process.

**4.4.4.4.1 Generate Confirmation Level 3: Determine Acceptance**

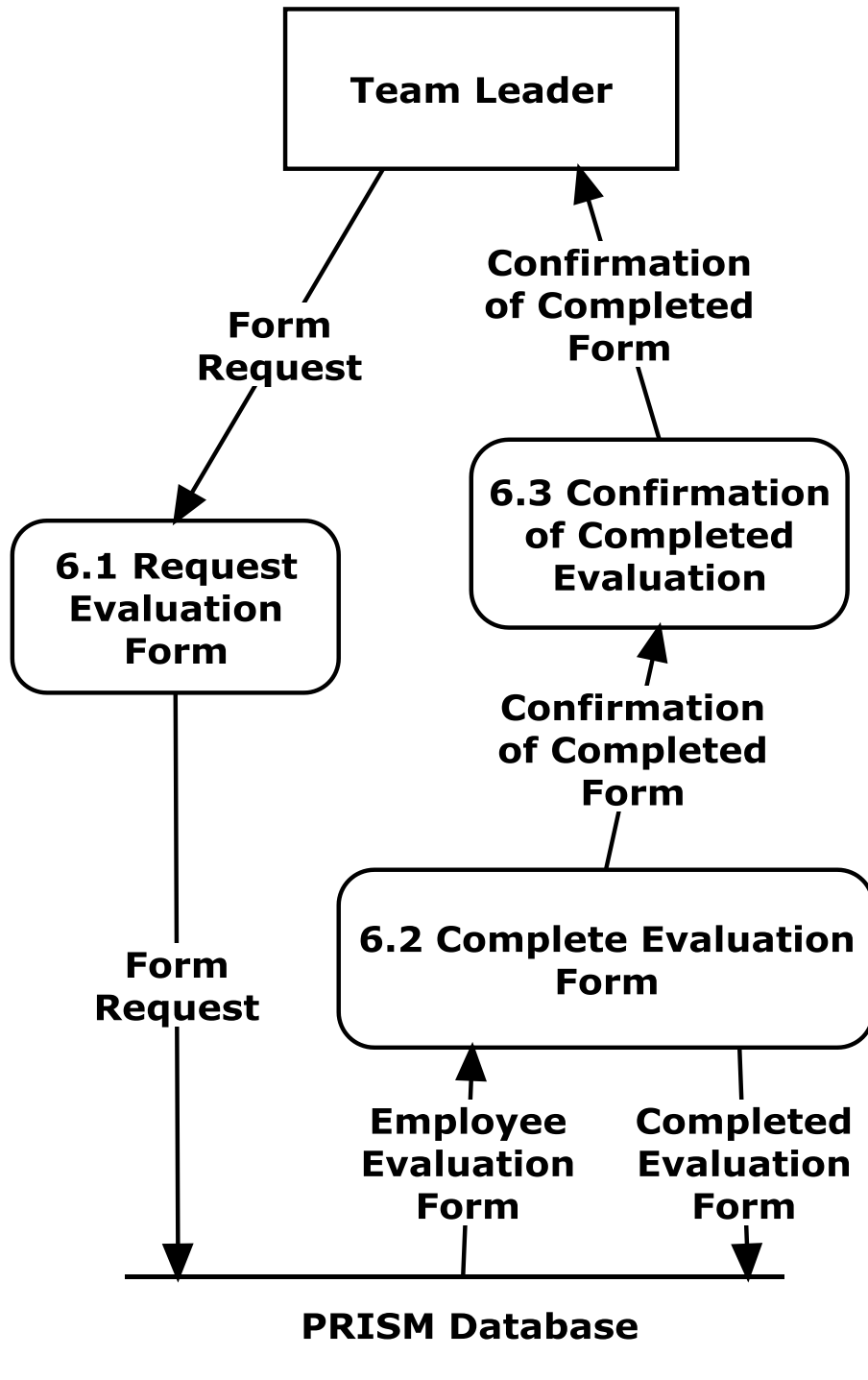




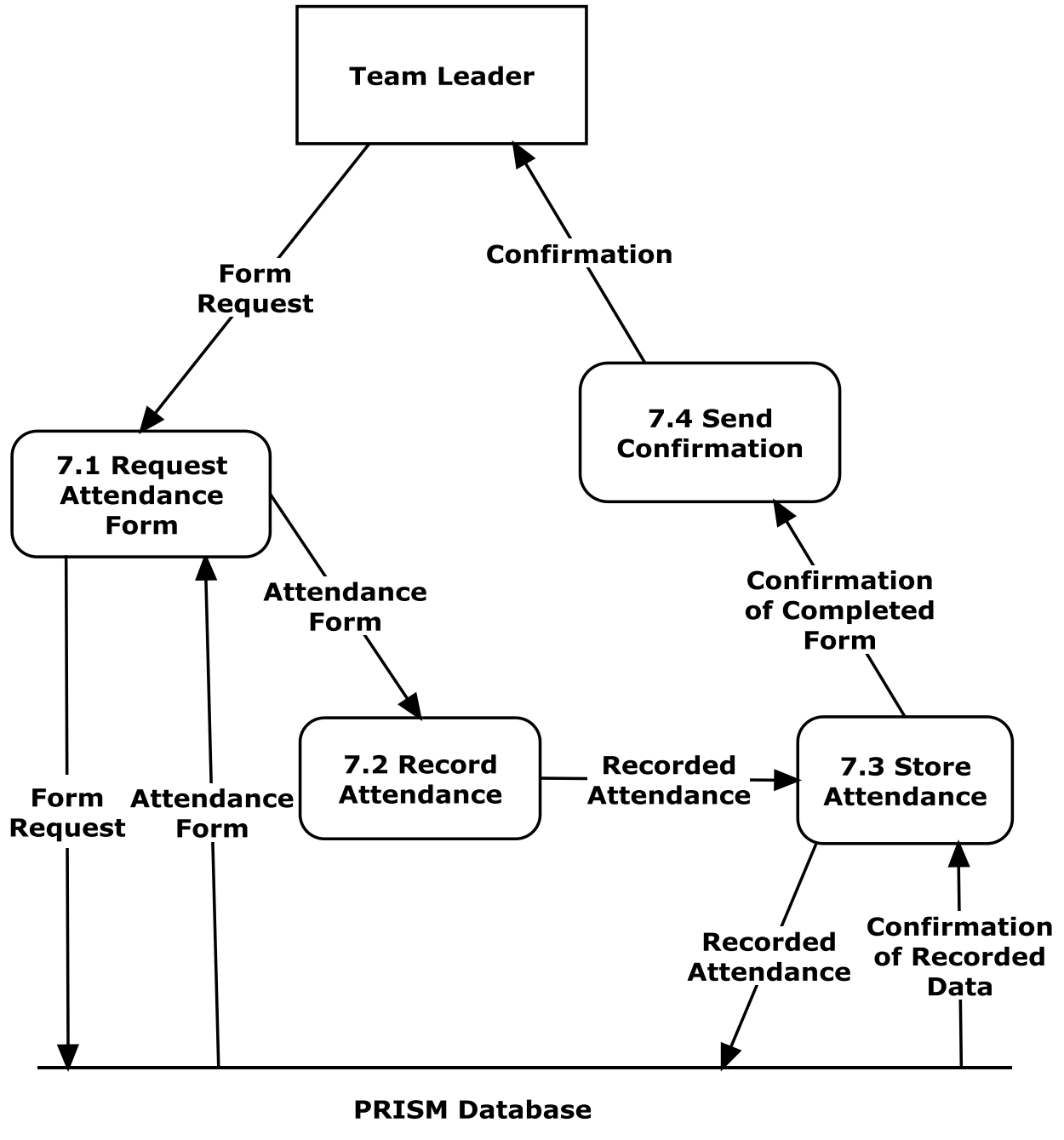
4.4.5 Confirm Team Hours



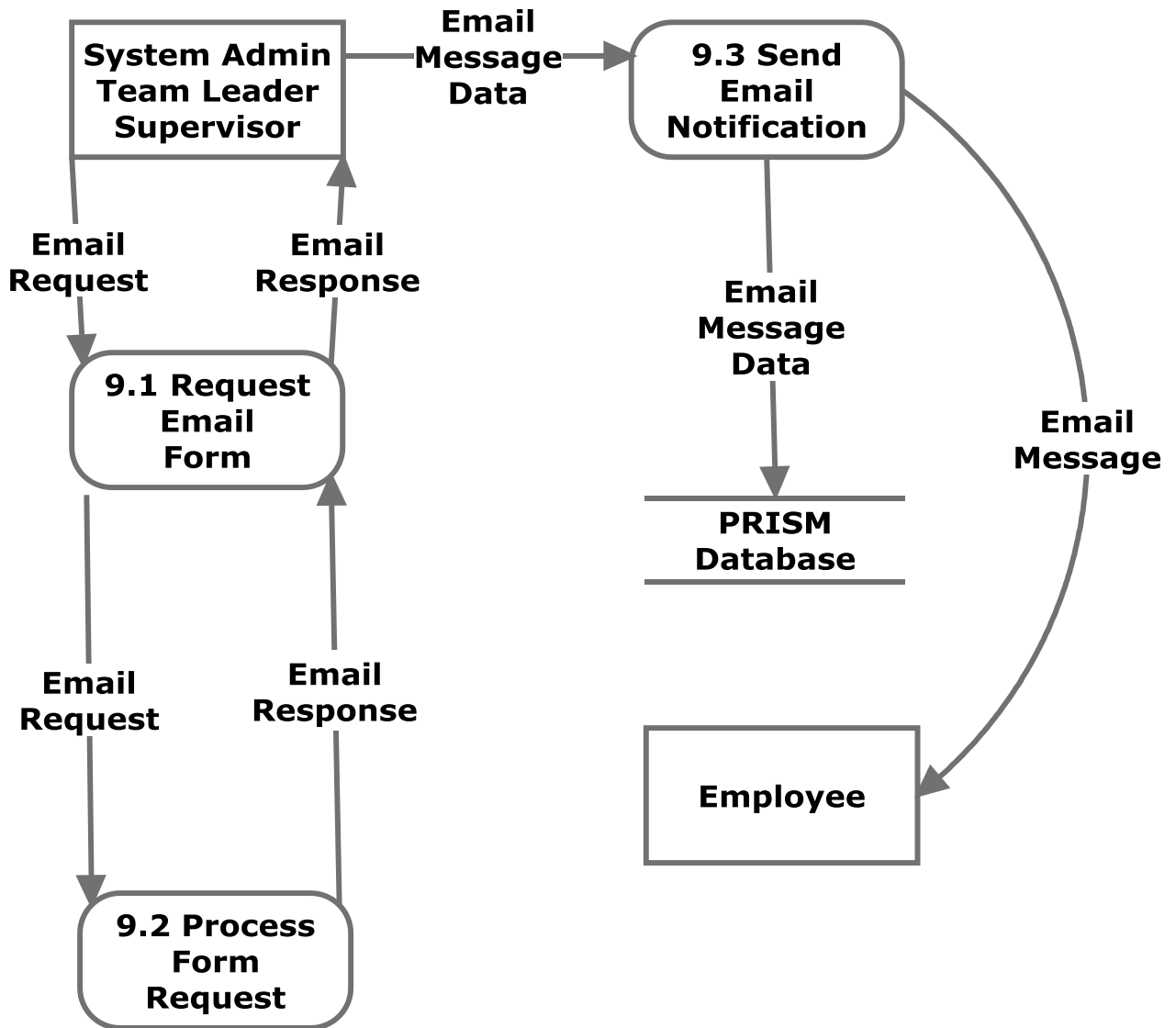
4.4.6 Create Team Leader Evaluation



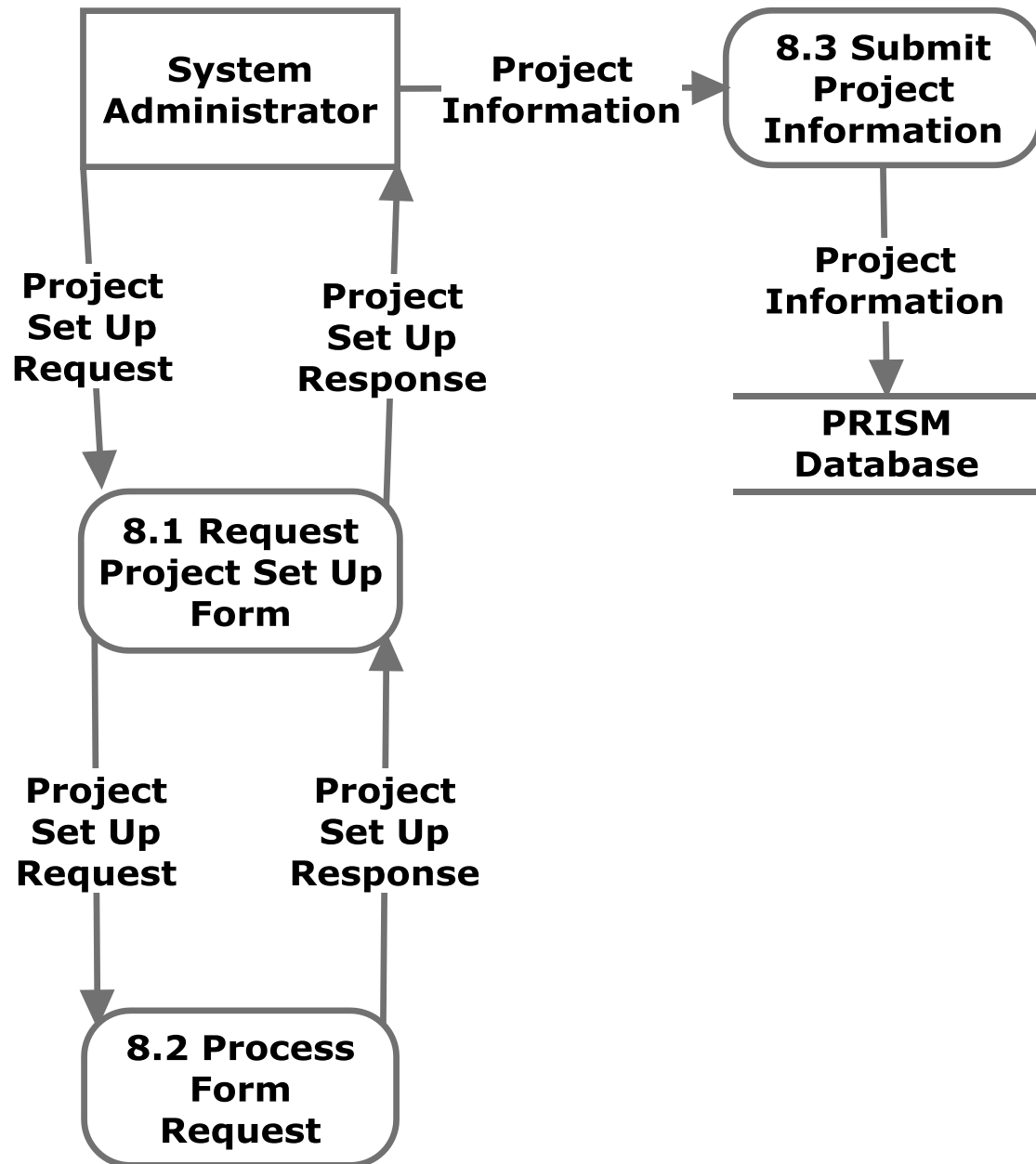
4.4.7 Record Attendance



4.4.8 Send Email Notification



### 4.4.9 Set Up Projects

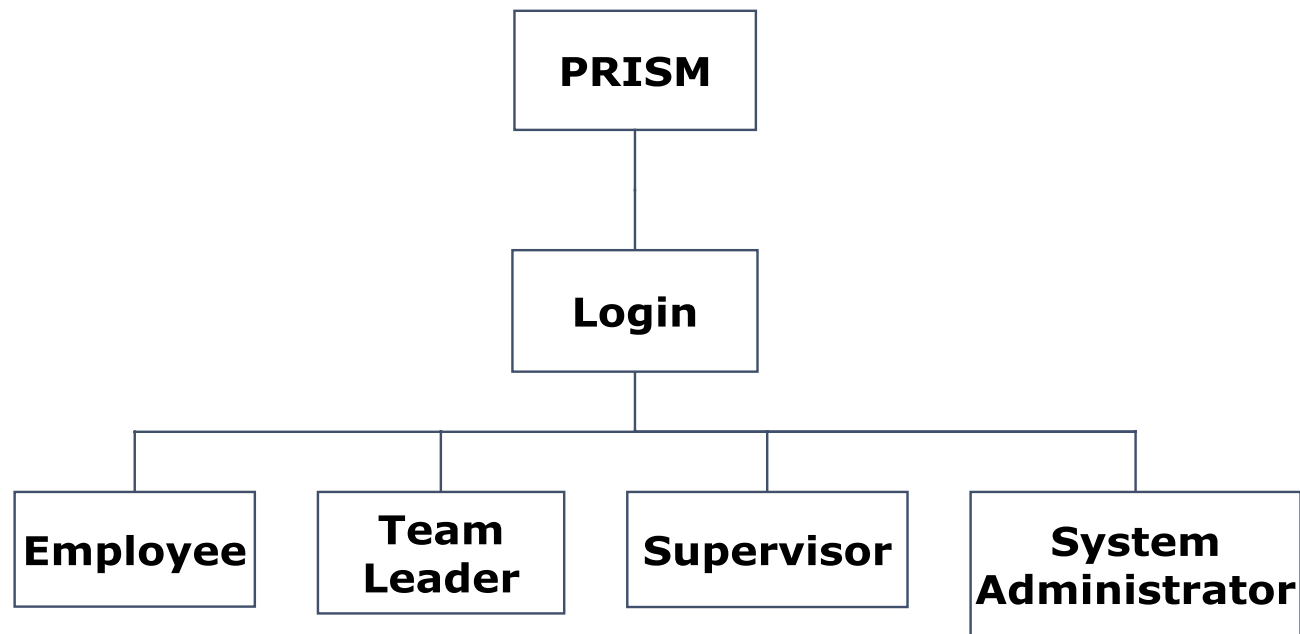


## 5. Structure Diagram

The structure diagram shows the hierarchy of all components within the PRISM system.

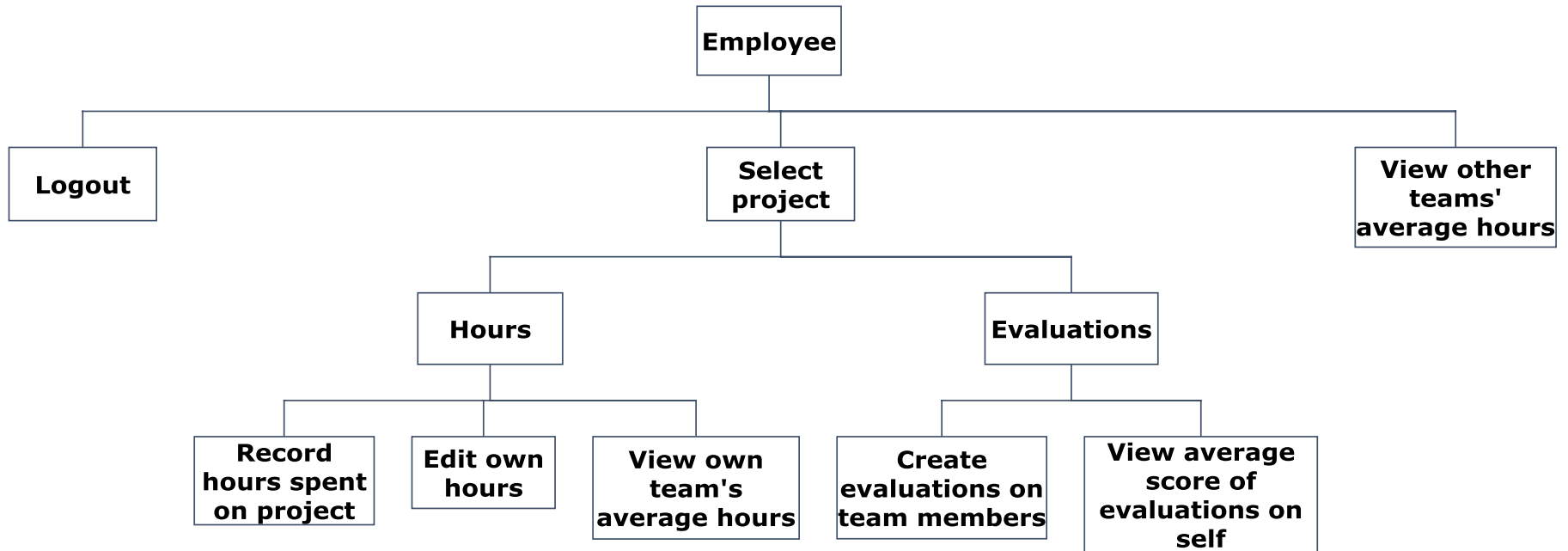
### 5.1 Login Structure

The login component leads to all other components of the PRISM system, and these components differ based on the status of the user logging into the system.



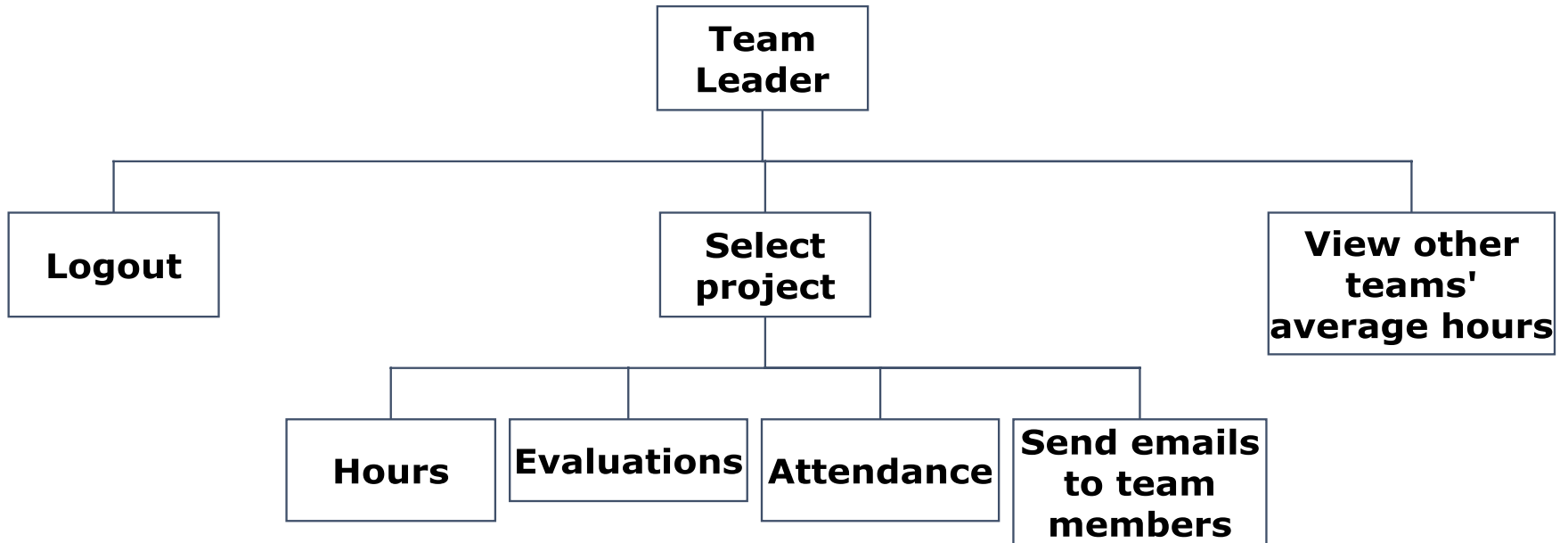
## 5.2 Employee Structure

This diagram shows the components that the Employee has access to, and shows the various detailed components within them.



### 5.3 Team Leader Structure

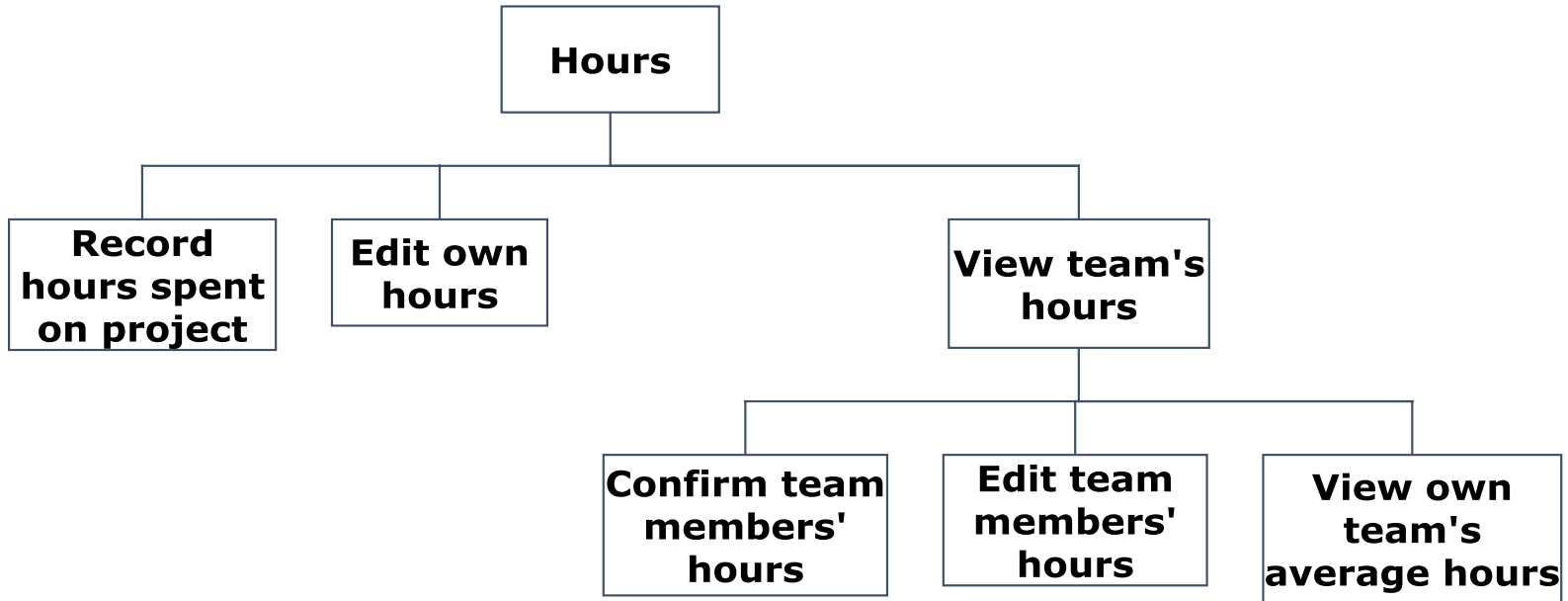
This diagram shows the components that the Team Leader has access to.





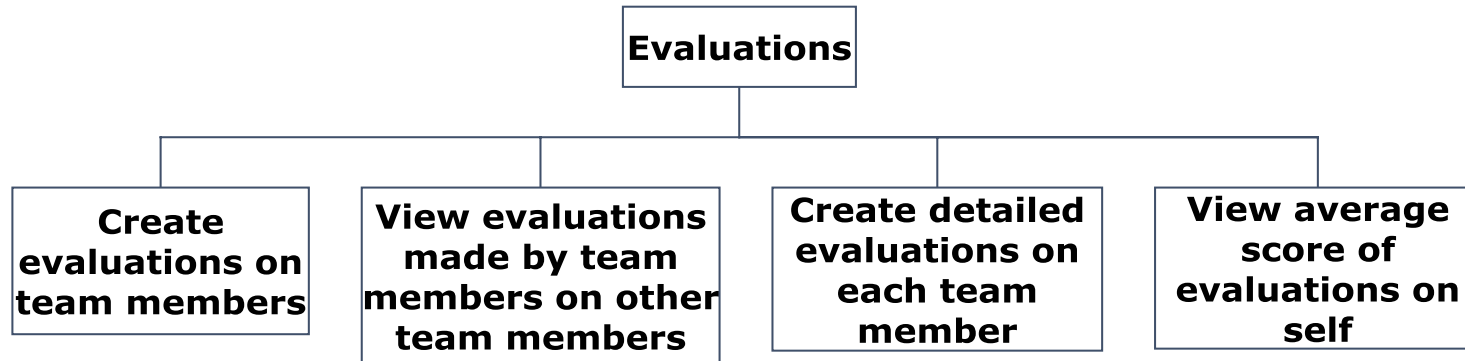
### 5.3.1 Team Leader Hours Structure

This diagram shows the more detailed components of the Team Leader's hours component.



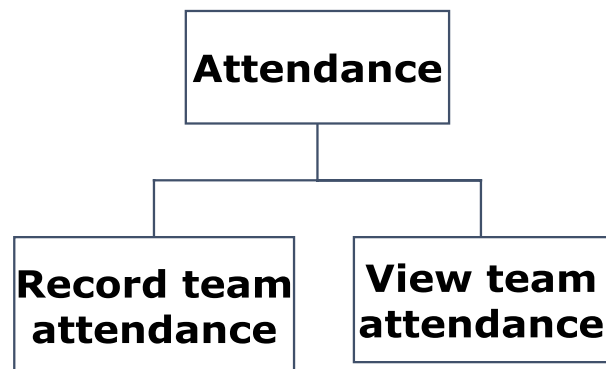
### 5.3.2 Team Leader Evaluations Structure

This diagram shows the more detailed components of the Team Leader's evaluations component.



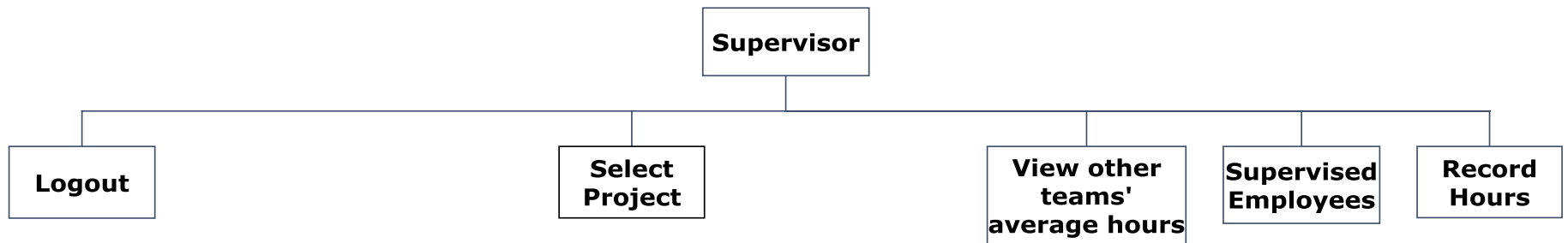
### 5.3.3 Team Leader Attendance Structure

This diagram shows the more detailed components of the Team Leader's hours component.



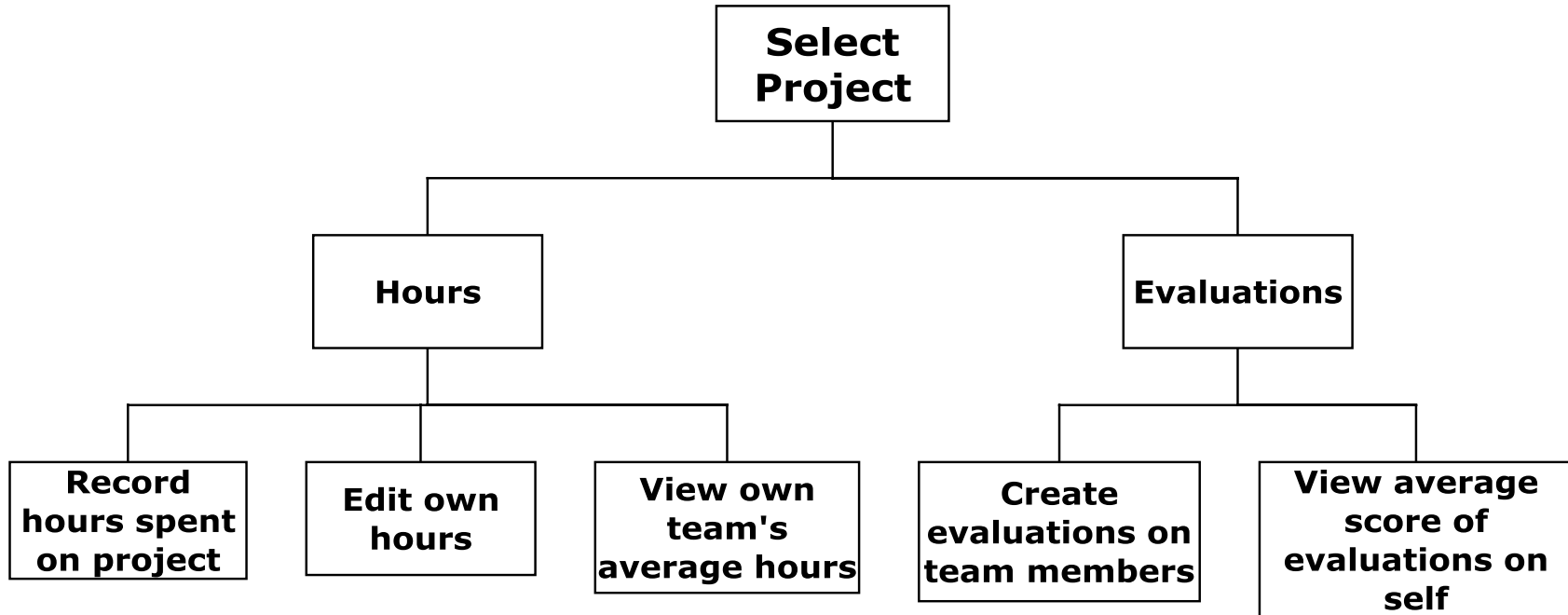
## 5.4 Supervisor Structure

This diagram shows the components that the Supervisor has access to.



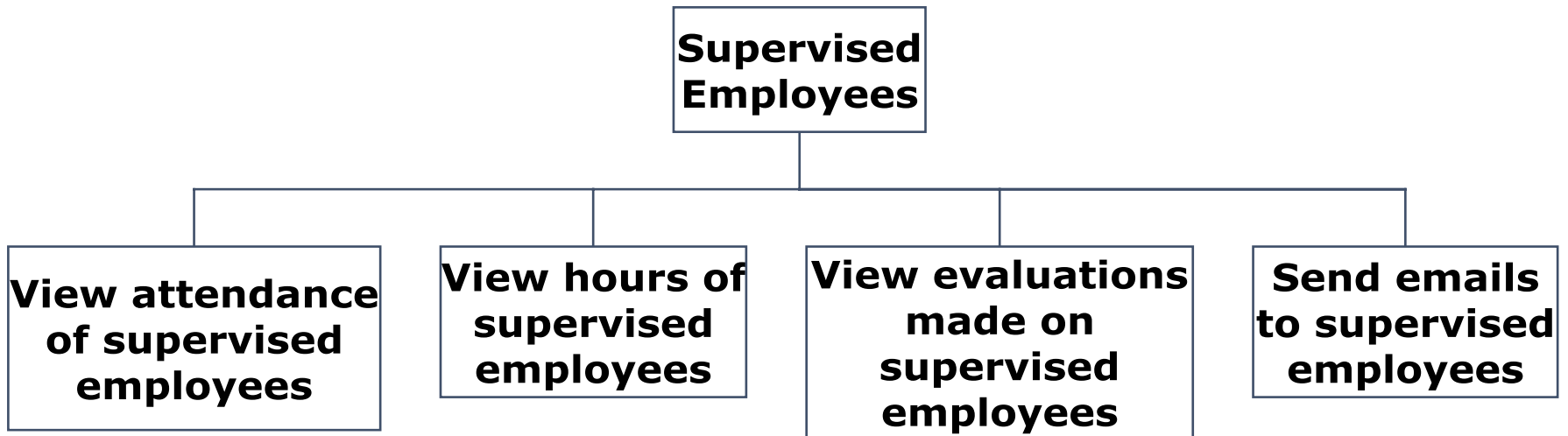
### 5.4.1 Supervisor Select Project Structure

This diagram shows the more detailed components of the Supervisor's select project component.



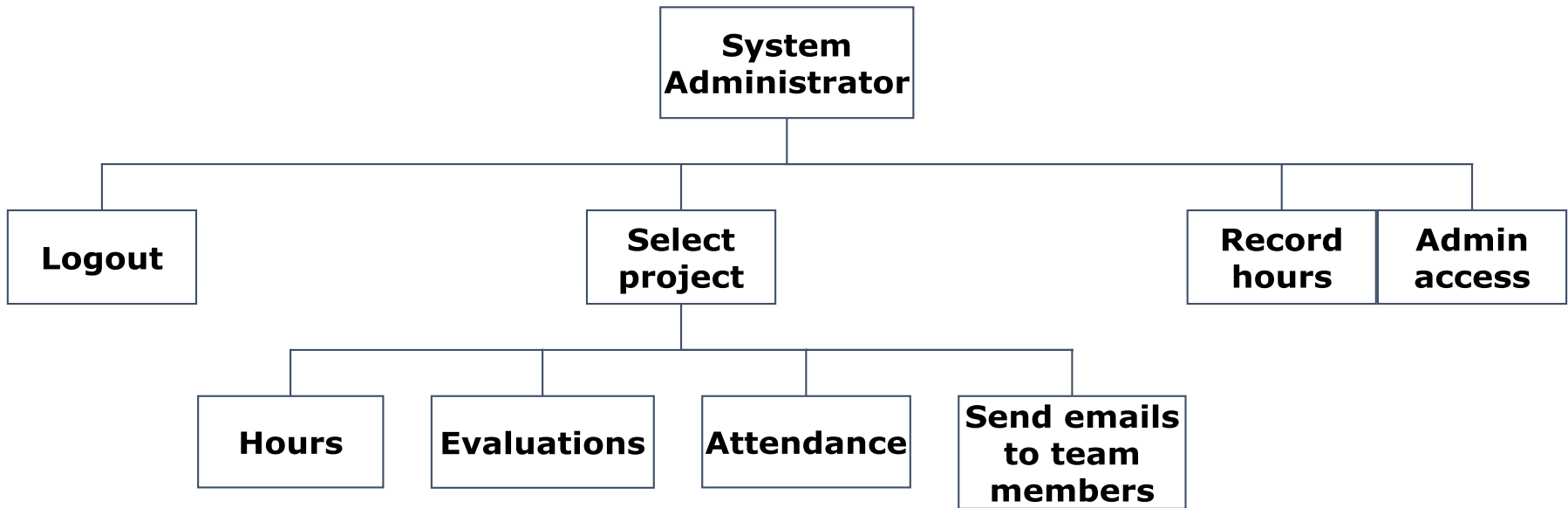
### 5.4.2 Supervisor Supervised Employees Structure

This diagram shows the more detailed components of the Supervisor's supervised employees component.



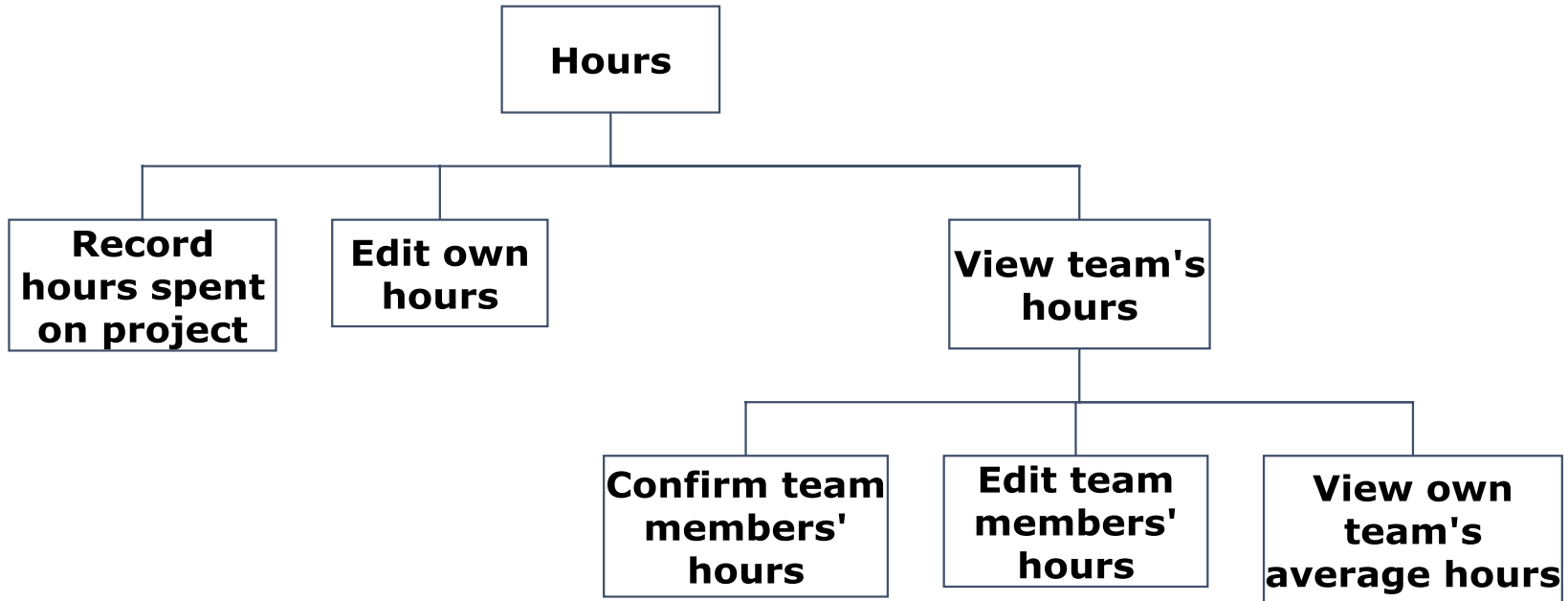
### 5.5 System Administrator Structure

This diagram shows the components that the System Administrator has access to.



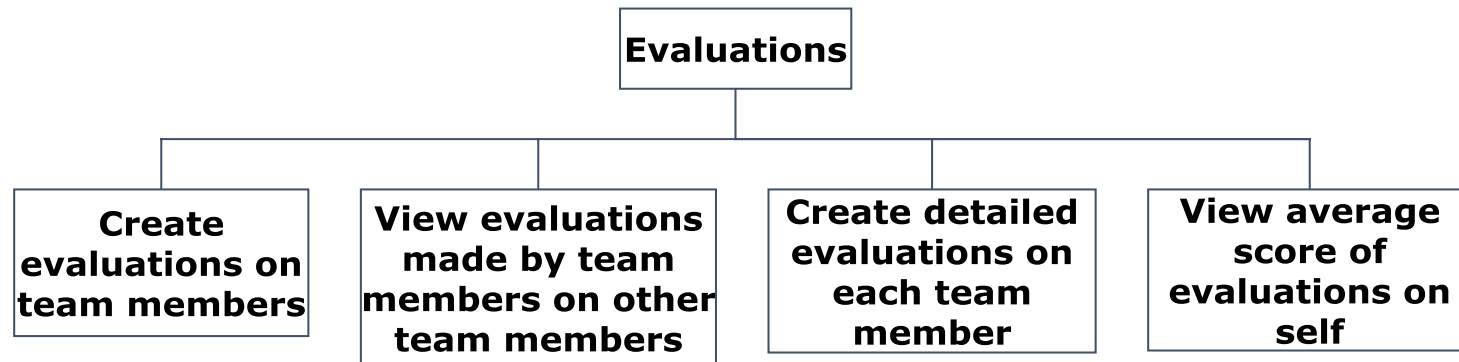
### 5.5.1 System Administrator Hours Structure

This diagram shows the more detailed components of the System Administrator's hours component.



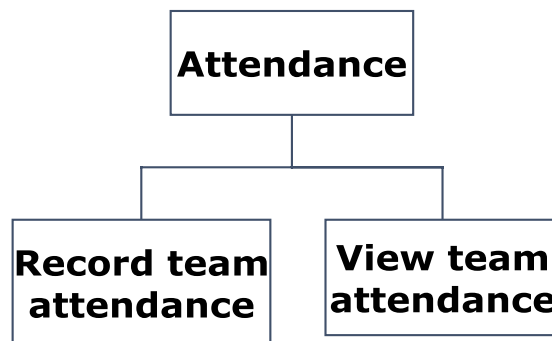
### 5.5.2 System Administrator Evaluations Structure

This diagram shows the more detailed components of the System Administrator's evaluations component.



### 5.5.3 System Administrator Attendance Structure

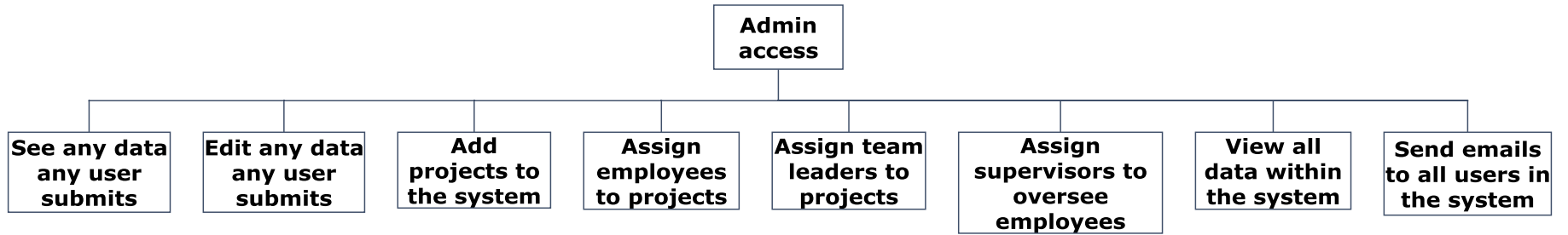
This diagram shows the more detailed components of the System Administrator's attendance component.





### 5.5.4 System Administrator Admin Access Structure

This diagram shows the more detailed components of the System Administrator's admin access component.



## 6. Functional Requirements Inventory

### 6.1 Employee

- Will be able to login to the system
  - A login attempt that fails will display an error message
- Will be able to record hours spent on projects
  - Includes hours spent working on projects and hours spent on project related meetings
- Will be able to edit own hours
  - Only true if the team leader has yet to confirm the employee's hours
- Will be able to view own recorded hours
- Will be able to view own team's average hours worked on a project
- Will be able to view other teams' average hours worked on a project
  - Only true if the other team(s) allow for average hours to be displayed publicly
- Will be able to submit evaluations of other team members
- Will be able to view own average score of evaluations submitted by other team members
- Will be able to log out

### 6.2 Team Leader

- Will be able to login to the system
  - A login attempt that fails will display an error message
- Will be able to record attendance of self and team members
- Will be able to view recorded attendance for own team
- Will be able to record hours spent on projects
  - Includes hours spent working on projects and hours spent on project related meetings
- Will be able to edit own hours
- Will be able to view own recorded hours
- Will be able to view hours recorded by team members
  - Will be able to confirm hours if the hours recorded by team members are valid; which will lock the hours from any future changes by the Employee
- Will be able to edit hours recorded by team members
- Will be able to view own team's average hours worked on a project
- Will be able to view other teams' average hours worked on a project
  - Only true if the other team(s) allow for average hours to be displayed publicly
- Will be able to submit evaluations of other team members
- Will be able to submit detailed evaluations of each individual team member
- Will be able to view evaluations made by other team members that are subordinate to the team leader
  - Will be able to view all scores and comments made on evaluations
- Will be able to view own average score of evaluations submitted by other team members
- Will be able to send emails to Employees on the team
  - Includes periodic emails on project deadline dates
- Will be able to log out

### 6.3 Supervisor

- Will be able to login to the system
  - A login attempt that fails will display an error message
- Will be able to record hours
- Will be able to record hours spent on projects if on a team
  - Includes hours spent working on projects and hours spent on project related meetings
- Will be able to edit own hours
- Will be able to view own recorded hours
- Will be able to view the hours of the Employees they supervise
- Will be able to view own team's average hours worked on a project
- Will be able to view teams' average hours worked on a project
  - Only true if the other team(s) allow for average hours to be displayed publicly
- Will be able to view attendance of Employees they supervise
- Will be able to submit evaluations of other team members
- Will be able to view own average score of evaluations submitted by other team members
- Will be able to view all evaluations made on the Employees, Team Leaders, and Supervisors they supervise
- Will be able to send emails to all Employees that they supervise
  - Includes periodic emails on project deadline dates to Employees that they supervise
- Will be able to log out

### 6.4 System Administrator

- Will be able to login to the system
  - A login attempt that fails will display an error message
- Will be able to record attendance of self and team members
- Will be able to view recorded attendance for own team
- Will be able to record hours
- Will be able to edit own hours
- Will be able to view own recorded hours
- Will be able to view hours recorded by team members
- Will be able to view teams' average hours worked on a project
- Will be able to submit evaluations of users
- Will be able to submit detailed evaluations of each user
- Will be able to view all evaluations made
- Will be able to create individual accounts
- Will be able to manipulate any user's account
- Will be able to edit any data any user submits
- Will be able to assign the role of a user
- Will be able to add projects to the system
- Will be able to assign specific Employees to projects
- Will be able to assign Team Leaders to projects
- Will be able to assign Supervisors to oversee Employees
- Will have complete access to all information recorded in the system by all Employees
- Will be able to send emails to all Employees
  - Includes periodic emails on project deadline dates
- Will be able to log out

## 7. Logical Data Dictionary

The purpose of the data dictionary is to display the variety of information and metadata that needs to be stored in order for the system to be functional. At the current phase, the data dictionary only showcases the data names and types, and also describes the purpose of each piece of data within the system. In the next document, the Detailed Design, the data dictionary will detail how these pieces are stored within the PRISM database.

Data Name	Applicable To	Data Type	Data Size	Description	Acceptable Input	Good Example(s)	Bad Example(s)	Notes
firstName	Add users to system	Varchar	1-30 characters	First name of a user	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	John	Joh2n\$%	
lastName	Add users to system	Varchar	1-50 characters	Last name of a user	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	Smith	Sm1t*	
userName	Add users to system, all screens user sees	Varchar	6-30 characters	Username of a user	A-Z, a-z, -, _, 0-9	js09smit	`js09\$mit	Usernames MUST be unique
password	Add users to system, change password, login, manage users	Varchar	8-30 characters	Password of a user	ASCII char 33 to ASCII char 126	d#5%3dSM NHJ~>!	( ), paA2, Password, 1213134	Must include one capital letter and one number

Data Name	Applicable To	Data Type	Data Size	Description	Acceptable Input	Good Example(s)	Bad Example(s)	Notes
email	Add users to system, forgotten password retrieval, send e-mails	Varchar	6-50 characters	E-mail of a user	Must be in proper e-mail format (i.e. name@website.com )	js09smit@QT.org	fjfe@, js09smit	e-mails MUST be unique
userHours	Record hours, view hours, confirm hours	Float	1-4 digits	Hours recorded by user	Proper number format (##.##)	4.75, 5, 10.00	94.322	Number must be less than 24 hours, and decimal places allowed will only be quarters (.00, .25, .5, .75)
hoursComments	Record hours, view hours, confirm hours	Varchar	0-256 characters	Description of what user did	ASCII char 32 to ASCII char 165	Team meeting #17, researched for req spec.	ℤ ℘ Ω Ū ǂ ı	
totalUserHours	View hours, confirm hours	Float	1-8 digits	Total hours recorded by user	Proper number format (##.##)	72.25, 532	92.43	Decimal places allowed will only be quarters (.00, .25, .5, .75), will be added up automatically

Data Name	Applicable To	Data Type	Data Size	Description	Acceptable Input	Good Example(s)	Bad Example(s)	Notes
totalTeamHours	View hours	Float	1-8 digits	Total hours recorded by team	Proper number format (##.##)	474, 32.00	847.303	Decimal places allowed will only be quarters (.00, .25, .5, .75), will be added up automatically
teamAvgHours	View hours, view other team's hours	Float	1-8 digits	Average hours recorded by team members	Proper number format (##.##)	32.52	9.0001	Decimal places will be limited to two digits, will be calculated automatically
beginTime	Record hours, view hours	Varchar	7-8 characters	Time where the user began work	Proper standard time format (##:## AM/PM)	10:30 AM	15:33, 12:12 AM	Time will be limited to :00, :15, :30, :45
endTime	Record hours, view hours	Varchar	7-8 characters	Time where the user ended work	Proper standard time format (##:## AM/PM)	12:30 PM	15:33, 12:12 AM	Time will be limited to :00, :15, :30, :46
beginDate	Record hours, view hours, take attendance, view attendance	Date	10 characters	Start date of hours recorded by user	MySQL Date format (Year-Month-Day)	2013-11-18	11/12/13	

Data Name	Applicable To	Data Type	Data Size	Description	Acceptable Input	Good Example(s)	Bad Example(s)	Notes
endDate	Record hours, view hours, take attendance, view attendance	Date	10 characters	End data of hours recorded by user	MySQL Date format (Year-Month-Day)	2013-11-19	11/13/13	
projectName	All screens relevant to a project	Varchar	2-128 characters	Name of a project	ASCII char 32 to ASCII char 165	PRISM	F	
projectLeader	Submit evaluation, record hours, view hours	Varchar	6-30 characters	Username of the leader(s) on a project	A-Z, a-z, -, _, 0-9	js09smit	`js09\$mit	
projectDesc	Add project to system, main page of a project	Varchar	0-4096 characters	Description of the project	ASCII char 32 to ASCII char 165	This is a time-management web-app for our client.	ℤ ℘ Ω Ū Š ı	
projectAddDate	Add project to system	Date	6-10 characters	Date project was added to system	MySQL Date format (Year-Month-Day)	2013-11-18	11/12/13	

Data Name	Applicable To	Data Type	Data Size	Description	Acceptable Input	Good Example(s)	Bad Example(s)	Notes
projectBeginDate	All screens relevant to a project	Date	6-10 characters	Date where work was begun on a project	MySQL Date format (Year-Month-Day)	2013-11-19	11/13/13	
projectComplete Date	All screens relevant to a project	Date	6-10 characters	Date where project was completed	MySQL Date format (Year-Month-Day)	2013-11-20	11/14/13	
projectBeginPhase	All screens relevant to a project	Date	6-10 characters	Date of the beginning of a phase of a project	MySQL Date format (Year-Month-Day)	2013-11-21	11/15/13	
projectEndPhase	All screens relevant to a project	Date	6-10 characters	Date of the end of a phase of a project	MySQL Date format (Year-Month-Day)	2013-11-22	11/16/13	
projectCurrentPhase	All screens relevant to a project	Varchar	1-128 characters	Name of the current phase of the project	ASCII char 32 to ASCII char 165	Preliminary Design	ℤ ℘ Ω Ū Ƴ ı	



Data Name	Applicable To	Data Type	Data Size	Description	Acceptable Input	Good Example(s)	Bad Example(s)	Notes
userStatus	Add users to system, manage user permissions, all screens user uses	Char	1 character	Denotes the status of a user's status in the system or on a project	E for employee, L for team leader, S for supervisor, A for system admin	A	Y, 2, &	
attendance	Record attendance, view attendance	Char	1 character	Attendance of a user on a date	A for absent, L for late, P for present	P	E, 55, #	
percentWorked	Submit evaluation	int	1-3 digits	Percent a user worked on a project compared to other team members	0-9	20	102, 103.4	Min value: 0, max value: 100
quality	Submit evaluation	int	1-3 digits	Quality of work produced for a phase of a project	0-9	95	-34, 332.7	Min value: 0, max value: 100

Data Name	Applicable To	Data Type	Data Size	Description	Acceptable Input	Good Example(s)	Bad Example(s)	Notes
averageEvalScore	View own evaluation score	Float	1-5 digits	Average quality score submitted by other team members' evaluations	Proper number format (##.##)	73.21	100.3343	Decimal places will be limited to two digits, will be calculated automatically
workQuality	Submit team leader individual evaluation	int	1-3 digits	Quality of a user's work	0-9	56	-34, 332.7	Min value: 0, max value: 100
workTimeliness	Submit team leader individual evaluation	int	1-3 digits	Timeliness of a user's work	0-9	100	-34, 332.8	Min value: 0, max value: 100
attendanceRating	Submit team leader individual evaluation	int	1-3 digits	Attendance rating of user	0-9	3	-34, 332.9	Min value: 0, max value: 100
participation	Submit team leader individual evaluation	int	1-3 digits	User's participation rating	0-9	23	-34, 332.10	Min value: 0, max value: 100
cooperation	Submit team leader individual evaluation	int	1-3 digits	User's cooperation rating	0-9	89	-34, 332.11	Min value: 0, max value: 100

<b>Data Name</b>	<b>Applicable To</b>	<b>Data Type</b>	<b>Data Size</b>	<b>Description</b>	<b>Acceptable Input</b>	<b>Good Example(s)</b>	<b>Bad Example(s)</b>	<b>Notes</b>
effort	Submit team leader individual evaluation	int	1-3 digits	User's effort extended	0-9	75	-34, 332.12	Min value: 0, max value: 100
userAddDate	Add users to system	Date	6-10 characters	Date user is added to system	MySQL Date format (Year-Month-Day)	2013-11-18	11/12/13	
userDeleteDate	Remove users from system	Date	6-10 characters	Date user is removed from system	MySQL Date format (Year-Month-Day)	2013-11-19	11/13/13	

## 8. Prototype Screens

### 8.1 Record Hours

Prototype on Next Page

The Record Hours form will appear when the user requests to record the hours worked on the project by that user. The user's name will automatically be inserted into the form based on the name for the current logged in user. The user will select the team and the project that the user will be recording hours for. The date will be set to the current date when the form was requested, however, the user may change the date via the dropdown calendar or manually type in the date. The time that was worked on the project from start to finish will be inputted by the user manually and AM or PM will be selected from a dropdown menu. If the user wishes, the user may insert comments about the recorded hours. When completed, the user will click on the 'Submit' button and will be prompted with a pop up asking for confirmation of submitted hours.

8.1 Record Hours (Cont)



# PRISM

## Record Hours

**Kathleen O'Hara**  
**Team Name:** Quantum Technologies ▾  
**Project Name:** Project PRISM ▾  
**Project Phase:** Requirements Specification ▾

Date : mm/dd/yyyy ▾  
 Start Time : 00 :00 AM ▾  
 End Time : 00 :00 AM ▾

Insert comment(s) here.

Submit

Date	Start Time	End Time	Duration	Comments
11/05/2013	9:00AM	10:00AM	1.0	Team Meeting
11/04/2013	9:00PM	10:00PM	1.0	Team Meeting
11/04/2013	4:15PM	7:30PM	3.25	Level-2 Diagram

## 8.2 View hours

The user will be able view all hours the user has submitted according to the specific project and project phase. The user will select the team name, project name, and project phase from a drop down menu and will click on the 'Submit' button. The 'Submit' button will generate a table that will display the hours the user has specified to view.



# PRISM

## View Hours

Kathleen O'Hara

Team Name:

Project Name:

Project Phase:

Date	Start Time	End Time	Duration	Comments
11/05/2013	9:00AM	10:00AM	1.0	Team Meeting
11/04/2013	9:00PM	10:00PM	1.0	Team Meeting
11/04/2013	4:15PM	7:30PM	3.25	Level-2 Diagram

[Quantum Technologies](#) | [Siena CS Department](#) | [Siena School of Science](#) | [Siena College](#)

Copyright (c) 2013 Quantum Technologies. All rights reserved.

### 8.3 System Administrator Home Page

The System Administrator’s home page will consist of displaying a shortcut to System Controls, a list of all active Projects with the according Team Leaders and Total Hours spent on the Project by that Team. The System Administrator has the availability of clicking on the Project Name which will direct the System Administrator to an email the entire Team on the Project. Also, by clicking on the Team Leader’s name, the System Administrator will be able to create an email which will be sent to that particular Team Leader. The System Administrator will also be able to view the average hours spent on all teams on the home page.



# PRISM

**System Administrator's Dashboard**

**System Controls**

Projects	Employees	Teams
<a href="#">Create Project</a>	<a href="#">Create Employee</a>	<a href="#">Create Team</a>
<b>Manage Project</b> PRISM ▾ <a href="#">Manage Project</a>	<b>Manage Employee</b> Kathleen O'Hara ▾ <a href="#">Manage Employee</a>	<b>Manage Team</b> Quantum Technologies ▾ <a href="#">Manage Team</a>

**Active Projects**

Project Name	Team Leader	Total Hours
PRISM	Shannon Pfohl	220
Project 2	John Doe	213
Project 3	Jane Doe	190
Project 4	Jim Doe	204

Team	Average
Quantum Technologies	259
Team 2	234

## 8.4 System Controls Page

The System Administrator will only have access to this page. The System Administrator will be able to create Projects, Teams, and Employees by clicking the particular link of which action the System Administrator would like to perform. By clicking on the links, the System Administrator will be redirected to a form to create a Project, Team, or Employee depending on which link is clicked. The System Administrator also has the privileges to manage Projects, Teams, and Employees. To manage a Team, the System Administrator will choose the Team to manage from the drop down menu and then will click the 'Submit' button which will direct the System Administrator to the 'Manage Team' form. To manage a Project, the System Administrator will choose the Project to manage from the drop down menu and then will click the 'Submit' button which will direct the System Administrator to the 'Manage Project' form. To manage an Employee, the System Administrator will choose the Employee to manage from the drop down menu and then will click the 'Submit' button which will direct the System Administrator to the 'Manage Employee' form.



# PRISM

## System Controls

**Projects**

[Create Project](#)

Manage Project

**Employees**

[Create Employee](#)

Manage Employee

**Teams**

[Create Team](#)

Manage Team



## 9. Testing Plan

### 9.1 Overview and Strategy

PRISM is a web application; therefore, PRISM will be tested to make sure it is functional on each major web browser. The web browsers to be tested on are Google Chrome, Internet Explorer, Mozilla Firefox, and Apple Safari. PRISM can also be tested on other web browsers if time allows. PRISM will be tested through a number of unit tests that are determined by the functional requirements. The non-functional requirements, which cannot be measured, will be evaluated by the members of Quantum Technologies to ensure each requirement has been satisfactorily met. A brief overview of the testing follows this section. In the detailed design there will be more detail for the testing plan.

### 9.2 Acceptance Test

Each component of PRISM will have individual tests that will be performed on the system. These tests will be done in order to make sure the individual parts of the system are working properly. Once the individual tests have been completed, an acceptance test will be executed to check if the functional requirements have been met. Once the acceptance test is over, Quantum Technologies and the client, Dr. Fryling, will decide whether all of the requirements have been sufficiently met. Quantum Technologies will design the testing plan using the functional requirements inventory.

### 9.3 Unit Tests

The unit tests are specific tests that will be run to ensure that the system is running properly as it is developed. The unit tests give step by step directions in the test cases on what to input and what to look for when a test is run. Certain tests need to be run together to ensure that all functions of the system work together properly and do not interfere with each other, these specific tests are listed in the "Integrated with these units" column.

#### 9.3.1 Test Cases

Each Unit is made up of a series of test cases. In these test cases there are directions on what to input into the system and in what order. The test cases also contain information on what the system should look like before the test is run and how it should respond after the test is run. If all of the test cases pass in the unit test, then the unit is considered to be functioning properly.

### 9.3.2 Unit Test: Directory

Pass/Fail Status	Unit Number	Unit Test Name	Date Last Tested	Comments or brief description	Integrated with these units
	1	<a href="#">Login</a>		Allows a User access into PRISM.	
	2	<a href="#">Average Hours</a>		Allows all Users to see the average hours on projects. Will be able to see total hours on a project and total hours on a specific phase of the project.	4,5
	3	<a href="#">Create Evaluation</a>		Allows all Users to create evaluations on their teammates .	
	4	<a href="#">Record Hours</a>		Allows all Users to Record the hours they have worked on specific projects.	2,5
	5	<a href="#">Confirm Team Hours</a>		Allows Team Leaders to view the hours of their team members and to confirm or reject the hours the team member has submitted. Once confirmed the team member will not be able to edit the data any further.	2,4
	6	<a href="#">Create TL Evaluation</a>		Allows Team Leaders to create more detailed evaluations on their team members.	
	7	<a href="#">Record Attendance</a>		Allows Team Leaders to Record Attendance for their team members in class and at meetings.	
	8	<a href="#">Send Email</a>		Allows Supervisors, Team Leaders, and System Administrators to send emails to employees, teams, or team leaders.	
	9	<a href="#">Set Up Project</a>		Allows the System Administrator to create and edit projects within PRISM	

## 9.3.3 Unit Test: Login

Test Cases										
Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
P	1.001	No Username, No Password	None	Press Enter	Blank Login Screen	Error Message, Redisplay Form				
P	1.002	Good Username, No Password	Login with just Username, no Password	Enter Username, Press Enter	Blank Login Screen	Error Message, Redisplay Form				
P	1.003	Bad Username, No Password	Login with incorrect Username, correct Password	Enter Incorrect Username, Press Enter	Blank Login Screen	Error Message, Redisplay Form				
P	1.004	No Username, Password	Login with no username and correct password	Enter Correct Password, Press Enter	Blank Login Screen	Error Message, Redisplay Form				
P	1.005	No Username, Bad Password	Login with no Username, incorrect Password	Enter Incorrect Password, Press Enter	Blank Login Screen	Error Message, Redisplay Form				

Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
P	1.006	Bad Username, Bad Password	Login with incorrect username and password	Enter Incorrect Username, Enter Incorrect Password, Press Enter	Blank Login Screen	Error Message, Redisplay Form				
P	1.007	Good Username, Bad Password	Login with correct username, incorrect password	Enter Correct Username, Incorrect Password, Press Enter	Blank Login Screen	Error Message, Redisplay Form				
P	1.008	Bad Username, Good Password	Login with incorrect username, and correct password	Enter Incorrect Username, Enter Correct Password, Press Enter	Blank Login Screen	Error Message, Redisplay Form				
P	1.009	Good Username, Good Password	Login with correct username, correct password	Enter Correct Username, Enter Correct Password, Press Enter	Blank Login Screen	Display Homepage				

**P = Unit Summary**      **100% passing**      **9 passed**      **Date of last test = 1/0/00**  
**9 tests**      **0 failed**

### 9.3.4 Unit Test: Average Hours

Test Cases										
Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
P	2.001	Check 1 Team Phase-Long Hours	Insert hours into one test project for one phase	Insert three different sets of hours from three different employees into a specific phase of the project. View the average chart	Average=0	Returns the average of the times submitted				
P	2.002	Check 2 Team Phase-Long Hours	Insert hours into two test projects for the same phase	Insert three different sets of hours from three different employees into a specific phase of the project. Insert thee different sets of hour from three different employees into the same specific phase of the project. View the average chart.	Average=0	Returns the averages of each project of the times submitted				

Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
P	2.003	Check 1 Team Project-Long Hours	Insert hours into one test projects for multiple phases	Insert three different sets of hours from three different employees into three different phases of the project. View the average chart	Average=0	Returns the averages of the times submitted				
P	2.004	Check 2 Team Project-Long Hours	Insert hours into two test projects for multiple phases	Insert three different sets of hours from three different employees into three different phases of the project. Insert thee different sets of hour from three different employees into three different phases of the project. View the average chart.	Average=0	Returns the averages of each project of the times submitted				

**P = Unit Summary**  
**4 tests**  
**100% passing**  
**4 passed**  
**0 failed**  
**Date of last test = 1/0/0**  
**0**

**9.3.5 Unit Test: Record Hours**

Test Cases										
Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
P	4.001	No Hours	None	Press Submit	Blank Hour Log Form	Error Message				
P	4.002	Date	Date	Enter date, Press Submit	Blank Hour Log Form	Error Message				
P	4.003	Clock in Partial Data	Partial Clock in Data	Enter Clock in Hour, Press Submit	Blank Hour Log Form	Error Message				
P	4.004	Clock out Partial Data	Partial Clock in Data	Enter Clock out Hour, Press Submit	Blank Hour Log Form	Error Message				
P	4.005	Date, Clock in Partial Data	Date, Partial Clock in Data	Enter Date, Enter Clock in Hour, Press Submit	Blank Hour Log Form	Error Message				
P	4.006	Date, Clock out Partial Data	Date, Partial Clock out Data	Enter Date, Enter Clock out Hour, Press Submit	Blank Hour Log Form	Error Message				

Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
P	4.007	Date, Clock in Full Data	Date, Full Clock in Data	Enter Date, Enter Clock in Hour, Enter Clock in Time, Enter Clock in AM/PM, Press Submit	Blank Hour Log Form	Error Message				
P	4.008	Date, Clock out Full Data	Date, Full Clock out Data	Enter Date, Clock out Hour, Enter Clock out Time, Enter Clock out AM/PM, Press Submit	Blank Hour Log Form	Error Message				
P	4.009	Date, Clock in Partial Data, Clock out Partial Data	Date, Partial Clock in Data, Partial Clock out Data	Enter Date, Enter Clock in Hour, Enter Clock out Hour, Press Submit	Blank Hour Log Form	Error Message				
P	4.010	Date, Clock in Full Data, Clock out Partial Data	Date, Full Clock in Data, Partial Clock out Data	Enter Date, Clock in Hour, Clock in Time, Clock in AM/PM, Enter Clock out Hour, Press Submit	Blank Hour Log Form	Error Message				





## 10. Development Environment and Production Environment

### 10.1 Development Environment

#### Computer #1 – Mac

- iMac
- Model Identifier: 12,1
- Mac OS X Lion 10.7.5 (11G3b)
- 21.4 inch (1920 x 1080 display)
- AMD Radeon HD 6750M 512 MB graphics
- Intel Core i5 (2.5GHz)
- 4 GB Memory
- 500 GB SATA Disk

#### Computer #2 – PC

- OptiPlex 760
- Windows Vista Enterprise
- Dual Dell Screens
  - Dell 2208WFP (1680 x 1050 display)
  - Dell 1908FP (1280 x 1024 display)
- Intel Core Duo CPU (2.93 GHz)
- 4 GB Memory
- 300 GB SATA Disk

#### Server

- Hostname: oraserv.cs.siena.edu
- CentOS 5.2 (final)
- Kernel: 2.6.18-92.el5
- Intel Xeon 2.66 GHz CPU
- 8 GB of Memory
- Java SE Runtime Environment (build 1.6.0 10-rc-b28)
- GCC Version 4.1.2 20071124 (Red Hat 4.1.2-42)
- Python 2.4.3

### 10.2 Operating Environment

Quantum Technologies will be using a web based application located on Siena College's oraserv database server. PRISM will consist of an Oracle database with an Apache Web server.

### 10.3 Maintenance

A bulk of the maintenance will be done in the security of the Software Engineering lab with the appropriate applications. These applications include, but are not limited to programs such as Adobe Photoshop, Dreamweaver and Notepad++ as well as internet browsers such as Google Chrome, Internet Explorer, Safari and Mozilla Firefox.

## 11. Appendices

### 11.1 Appendix A: Glossary of Terms

**Actor:** Actors that interact with the system through uses. Actors can be human or non human

**Adobe Dreamweaver:** Tool used for web application development

**Adobe Photoshop:** Graphic editing application

**Apache HTTP Server:** Apache HyperText Transfer Protocol Server, Web server application

**Apple Safari:** Web browser designed by Apple

**Data Flow:** Data/information flowing to or from a process in a Data Flow Diagram

**Data Flow Diagram:** A graphical representation of the "flow" of data through an information system

**Data Store:** Location where data is held temporarily or permanently in a Data Flow Diagram

**Database:** Organizes data, typically through a computer, so that the data is easily accessible

**External Entities:** A component of a Data Flow Diagram that represents any human or non-human user of a Software System

**Functional Requirements Inventory:** Defines what the system will be able to do and what is testable about the system

**Eclipse:** Programming environment developed by the Eclipse Foundation

**Gantt Chart:** Bar chart typically used to project scheduling, represents a timeline

**Google Chrome:** Web browser designed by Google

**HTML:** HyperText Markup Language, main language for creating web pages

**HTTP:** Hypertext Transfer Protocol, A protocol used to transfer hypertext requests and information between servers and browsers

**Inclusion Arrow:** An arrow that points from a scenario to another scenario to show that something must be included for the scenario

**Inheritance Arrow:** An arrow that points from one use to another; the use of being pointed at is the parent and the other is the sub

**Internet Explorer:** Web browser designed by Microsoft

**Level-0 Diagram:** A data flow diagram that represents a system's major processes, data flows, and data stores at a high level of detail

**Level-1 Diagram:** Provides an overview of the major functional areas of the undertaking

**Level- 2 & Level-3:** The Level-2 and Level-3 diagrams demonstrate an even further expansion of the higher level diagrams.

**Mozilla Firefox:** Web browser designed by Mozilla Foundation and the Mozilla Corporation

**Notepad++:** Text editor specializing in syntactic highlighting of various programming languages

**ODBC:** Open Database Connectivity is a standard API used for connecting to database

**Oracle Database:** An object-relational database management system produced and marketed by Oracle Corporation

**Oraserv Database:** Siena College's database server

**Participation Line:** Shows what scenarios an actor can interact with in a UML Use Case Diagram

**PHP:** PHP: Hypertext Preprocessor, a programming language used for developing client-side scripts for web browsers

**PRISM:** Project Recording Information System Management

**Process:** Transforms or manipulates data in a Data Flow Diagram

**Prototype:** An early sample, model or release of a product built to test a concept

**Scenarios:** The actions that occur within a system and how the user interacts with the system

**SCP:** Secure Copy Protocol is a network protocol used for secure data communication

**SQL:** Structured Query Language, language used to develop databases

**User Interface:** Space where a user can interact with a computer through inputs and outputs

**System Boundary:** The boundary between the system and the external entities in a Data Flow Diagram

**UML Use Case Diagram:** A type of behavioral diagram to present a graphical overview of the functionality provided by a system

**UML:** Unified Modeling Language is a specification language used in software engineering

**Unit Testing;** A testing method where the system is broken down into units and each unit is tested

**Website Map:** A list of pages of a website accessible to users

### 11.2 Appendix B: Timeline

**Quantum Technologies Project Timeline**

Number	Task	Start	End	Duration	% Complete	9/9	9/16	9/23	9/30	10/7	10/14	10/21	10/28	11/4	11/11	11/18	11/25	12/2	
1	Establish Team	9/9/2013	9/9/2013			◆													
2	Software plan	9/9/2013	9/17/2013	7	100.0	■													
3	Software plan due date	9/18/2013	9/18/2013				◆												
4	Software plan presentation	9/18/2013	9/18/2013				◆												
5	Requirement Specifications	9/18/2013	10/29/2013	29	100.0		■												
6	Required Documents due date	10/30/2013	10/30/2013		100.0								◆						
7	Requirement presentation	10/30/2013	10/30/2013										◆						
8	Preliminary Design	10/30/2013	12/3/2013	23	100.0								■						
9	Preliminary Design due date	12/4/2013	12/4/2013		100.0														◆
10	Preliminary design presentation	12/4/2013	12/4/2013		100.0														◆
11	Team Meeting	Monday	Monday			■	■	■	■	■	■	■	■	■	■	■	■	■	■
12	Client Meeting	Thursday	Thursday			□	□	□	□	□	□	□	□	□	□	□	□	□	□
13	Team Meeting	Friday	Friday			■	■	■	■	■	■	■	■	■	■	■	■	■	■

Legend	Symbol
Milestone	◆
Document	■
Team Meeting	■
Client Meeting	□
Deadline	▼
Completed Document	■