

## Requirements Specification

### 1. Scope

#### 1.1 Overview

The Time Entry custom component is part of the Time Tracker application. It provides an abstraction of a time log entry that an employee enters into the system on a regular basis. This component handles the persistence and other business logic required by the application.

#### 1.2 Logic Requirements

##### 1.2.1 Time Entries

A time entry represents the date and amount of time an employee has worked for a particular project and client. It is normally used for payroll and billing purposes.

The following attributes will be maintained:

- Description – a brief description of the time entry
- Date – the date for the time entry
- Hours – the number of hours the employee worked (cannot be negative)
- Task Type – the type of task performed
- Status – the status of the time entry
- Billable – a flag to indicate whether the entry is billable to client

The database schema to store time entries is as follows:

```
CREATE TABLE TimeEntries (  
    TimeEntriesID    integer NOT NULL,  
    TaskTypeID       integer, NOT NULL,  
    TimeStatusesID   integer, NOT NULL,  
    Description       varchar(64) NOT NULL,  
    Date             date NOT NULL,  
    Hours            float NOT NULL,  
    Billable         smallint NOT NULL,  
    CreationDate      date NOT NULL,  
    CreationUser      varchar(64) NOT NULL,  
    ModificationDate  date NOT NULL,  
    ModificationUser  varchar(64) NOT NULL,  
    PRIMARY KEY (TimeEntriesID)  
);
```

##### 1.2.2 Task Types

Each time entry has an associated type of task performed by the employee.

The following task types have been identified:

- Component Specification
- Component Design
- Component Development
- Information Architecture
- Project Management

- Meeting
- Sales
- Miscellaneous

Note: New task types may be added or modified as per application requirements.

The database schema to store task types is as follows:

```
CREATE TABLE TaskTypes (  
    TaskTypesID      integer NOT NULL,  
    Description       varchar(64) NOT NULL,  
    CreationDate      date NOT NULL,  
    CreationUser      varchar(64) NOT NULL,  
    ModificationDate  date NOT NULL,  
    ModificationUser  varchar(64) NOT NULL,  
    PRIMARY KEY (TaskTypesID)  
);
```

#### 1.2.3 Time Entry Statuses

Each time entry has a status assigned to it. The status will be modified by an administrator or the supervisor of the employee.

The following statuses have been identified:

- Pending Approval – the time entry is pending approval by an administrator or supervisor
- Approved – the time entry has been approved by an administrator or supervisor
- Not Approved – the time entry is not approved by an administrator or supervisor

Note: New statuses may be added or modified as per application requirements.

The database schema to store time entry statuses is as follows:

```
CREATE TABLE TimeStatuses (  
    TimeStatusesID   integer NOT NULL,  
    Description       varchar(64) NOT NULL,  
    CreationDate      date NOT NULL,  
    CreationUser      varchar(64) NOT NULL,  
    ModificationDate  date NOT NULL,  
    ModificationUser  varchar(64) NOT NULL,  
    PRIMARY KEY (TimeStatusesID)  
);
```

#### 1.2.4 Pluggable Persistence

All entities defined above will be backed by a database. The design will provide the necessary API to store and retrieve data from the database.

For the initial version, the Informix database system will be used as persistence storage for this component and the Time Tracker application. Other database systems should be pluggable into the framework.

### 1.3 Required Algorithms

None.

### 1.4 Example of the Software Usage

The Time Tracker application will use this component to perform operations related to time entries.

### 1.5 Future Component Direction

Other database systems maybe plugged in for some client environments.

## 2. Interface Requirements

### 2.1.1 Graphical User Interface Requirement

None.

### 2.1.2 External Interfaces

None.

### 2.1.3 Environment Requirements

- Development language: Java 1.4
- Compile target: Java 1.3, Java 1.4

### 2.1.4 Package Structure

com.topcoder.timetracker.entry.time

## 3. Software Requirements

### 3.1 Administration Requirements

#### 3.1.1 What elements of the application need to be configurable?

None.

### 3.2 Technical Constraints

#### 3.2.1 Are there particular frameworks or standards that are required?

None.

#### 3.2.2 TopCoder Software Component Dependencies:

- Configuration Manager
- DB Connection Factory

\*\*Please review the [TopCoder Software component catalog](#) for existing components that can be used in the design.

#### 3.2.3 Third Party Component, Library, or Product Dependencies:

Informix Database.

#### 3.2.4 *QA Environment:*

- Solaris 7
- RedHat Linux 7.1
- Windows 2000
- Windows Server 2003
- Informix

### 3.3 **Design Constraints**

The component design and development solutions must adhere to the guidelines as outlined in the TopCoder Software Component Guidelines. Modifications to these guidelines for this component should be detailed below.

### 3.4 **Required Documentation**

#### 3.4.1 *Design Documentation*

- Use-Case Diagram
- Class Diagram
- Sequence Diagram
- Component Specification

#### 3.4.2 *Help / User Documentation*

- Design documents must clearly define intended component usage in the 'Documentation' tab of Poseidon.