Time Sheet Web App
COMP3910 Assignment 2

Requirements & Design Document

Tony Pacheco & Danny Di Iorio

November 11, 2018

Table of Contents

Purpose

The Timesheet application will be an online system which allows users to login and access/edit workplace timesheets and allows an administrator to manage the user's accounts.

Scope

The Timesheet application UI will be a web interface consisting of numerous pages and forms to view, edit, add, and remove timesheet and employee data. Data storage and overall application functionality will be handled using a MySQL database and Java Persistence Architecture transactions.

Perspective

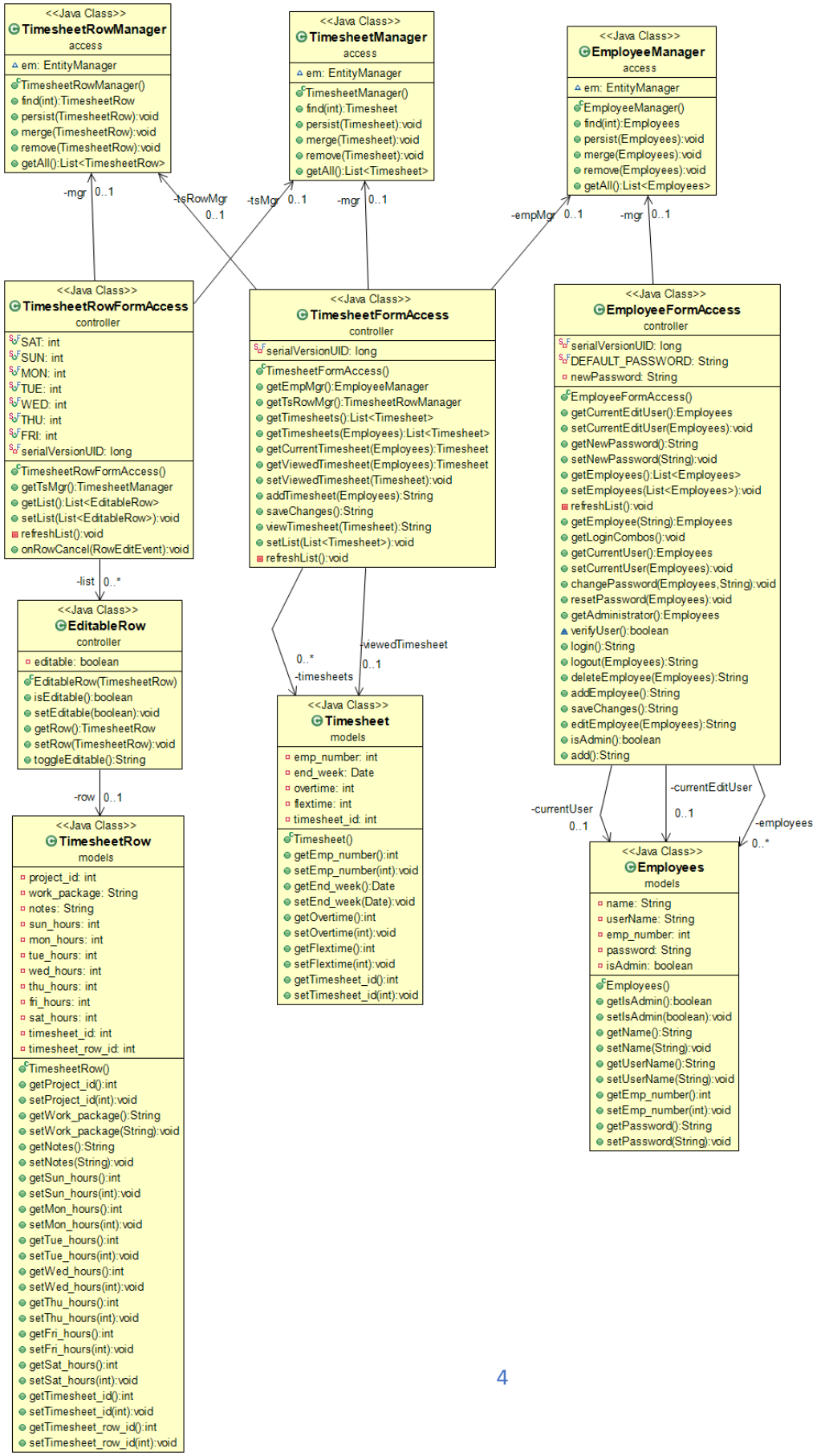
The application will store the following data in its database:

1. User (Employee) Data for each user in the system:
    a. Name
    b. Employee Number
    c. Username
    d. Password

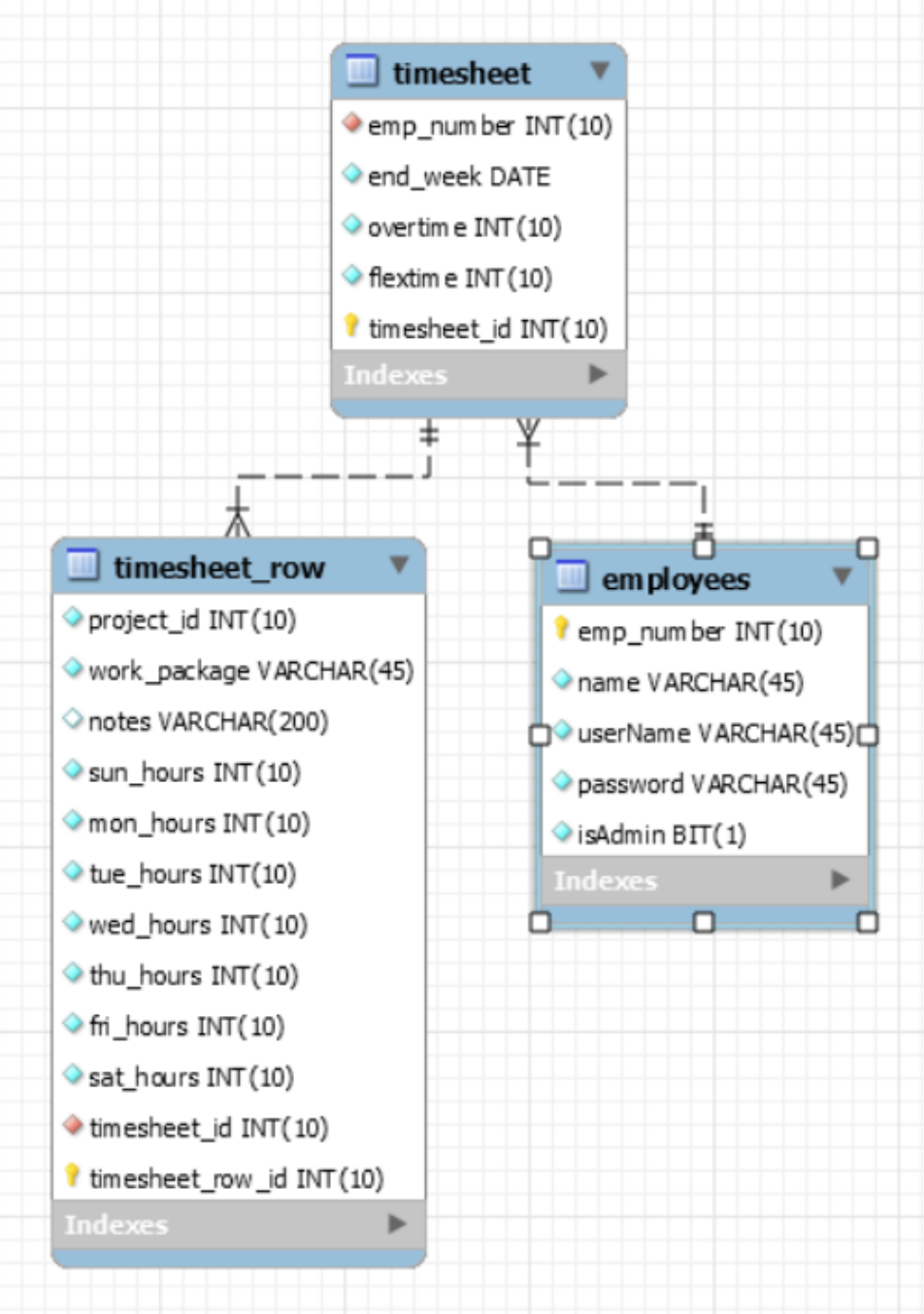
2. Timesheet Data:
    Each timesheet records the hours worked by a single employee on a given week. Each will contain the following data:
    a. Employee Number
    b. Employee Name
    c. Week Number (0-52)
    d. Week Specifier (the date on which the week ends – Friday)
    e. The following, each as a set of data to displayed as the rows of a timesheet table
        a. Project Number
        b. Work Package identifier
        c. Total number of hours worked for the week
        d. A column for each day of the week containing the number of hours worked that day
        e. Additional optional notes

        Each row above represents a week of work hours on a given work package of a specific project.

Class Diagram

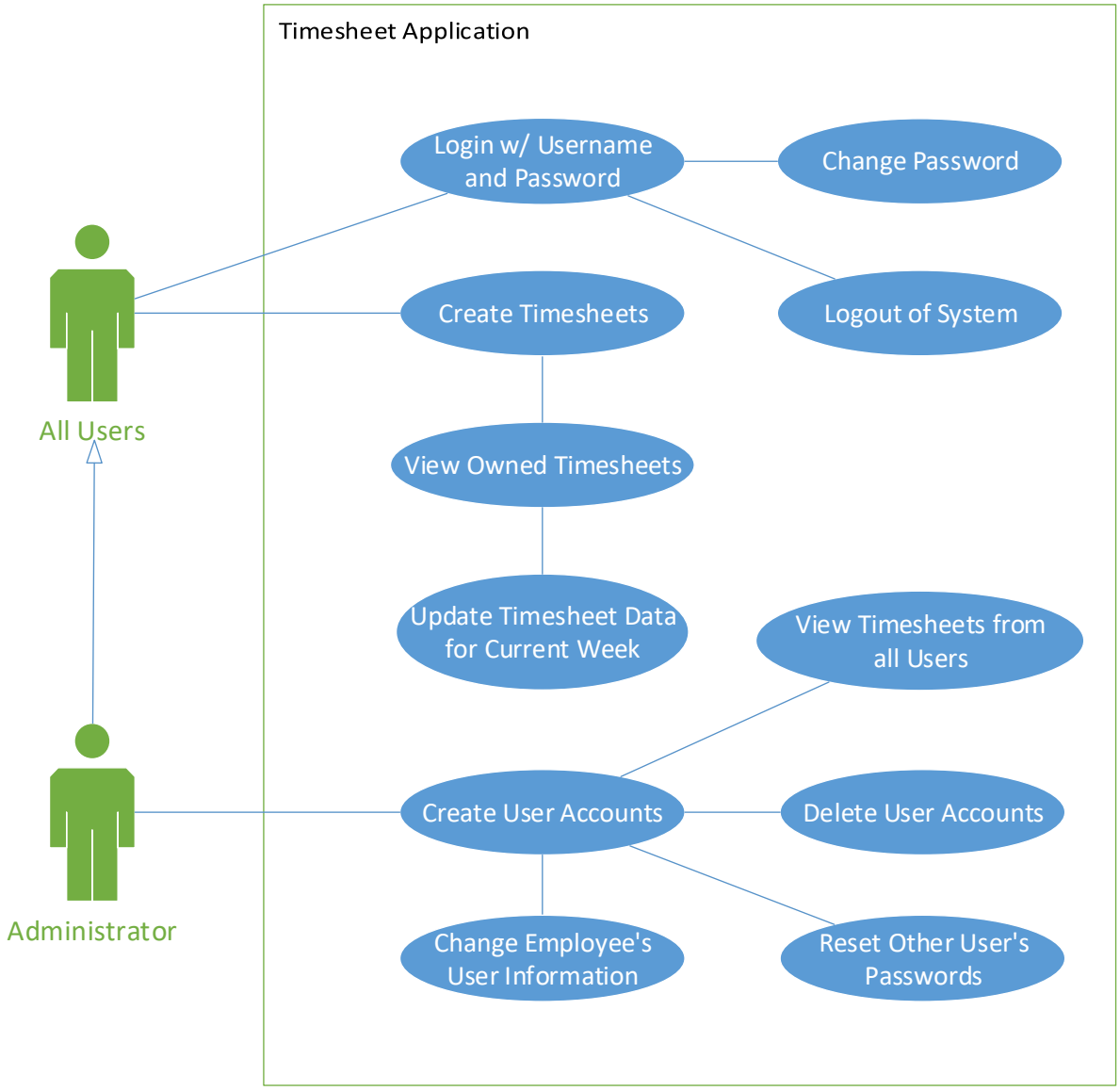


Database Entity Relational Diagram



Use Cases

The two types of users which the application will support are general users and the system's administrator. General users will be able to manage their own account settings and create and edit their own timesheets. Administrators will be able to do anything that general users can do, as well as manage the creation, modification, and deletion of other user accounts. Administrators can also view timesheets from all users in the system.



Interface Requirements

General Users

When a user first starts the application, they will see a login page, if they have a registered account, they can login with their username and password. If they do not have a registered account, an administrator will need to create an account for them.

After a registered user successfully logs in, they will see a page with the list of their saved timesheets, organized by date. Each timesheet can be opened and edited by clicking on it.

Users can create a new timesheet by clicking on "New Timesheet" in the header. This will open a page with a new timesheet with five empty rows. By clicking on a pencil icon in the last column of the table, the user will be able to edit the contents of that row in the table, which allows them to input/edit hours worked for any day of the week.

This page will also have buttons on a top toolbar which allow them to view a different week's timesheet, create a new timesheet, change their password, or logout of the application.

Administrators

If an administrator logs in to the application, they will see a page with the list of saved timesheets from all users, organized by date. Each timesheet can be opened and edited by clicking on it.

Admins have a header link to see a page with a list of all the users in the system. They will be able to click on an edit button beside each user to edit them, click a button beside each user to remove them, or click a button at the bottom of the list to create a new user. In the edit user page, the admin can edit all fields and have the option to reset their password to default.

The administrators will also have the same options in the toolbar as the general users so that they can create and edit their own timesheets. They will have access to view other users' timesheets.

Operating Environment

The operating environment will be split up into three layers: *presentation tier*, *business objects tier*, and *persistence tier*.

The *presentation tier* will use Java Server Faces, Prime Faces, and CDI beans to comprise the user interface. Front-end styling is handled by a custom stylesheet and Materialize 3rd party CSS library. Templates will be used for the main layout, header, and footer, and a message bundle for all text where possible.

The *business objects tier* will use JPA session and entity beans to handle the application functionality and represent the data within the database, respectively. We have designed entity beans to

represent each database table, with entity manager Java classes to provide CRUD services to these beans.

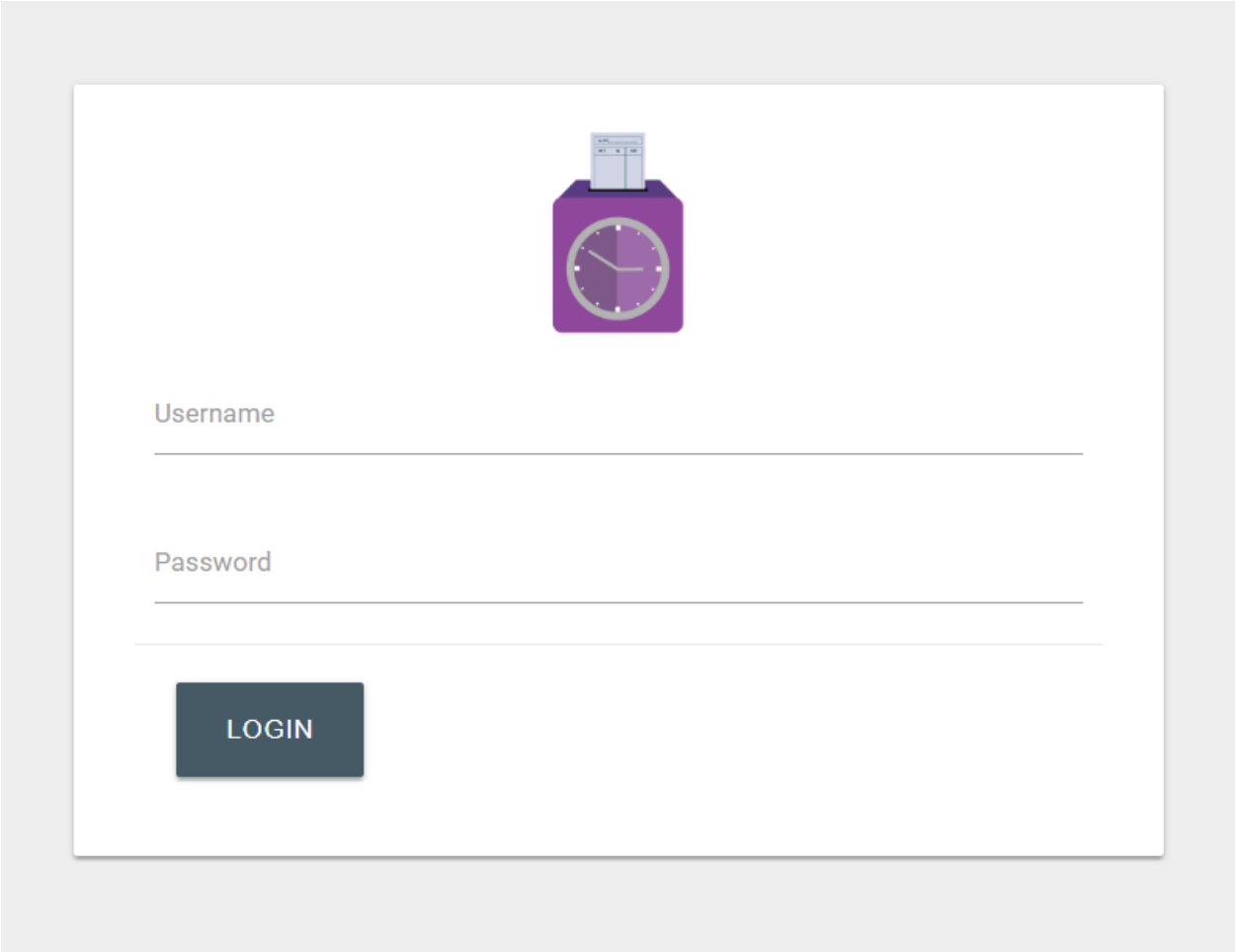
The *persistence tier* consists of a MySQL relational database designed by us. The database will be connected to the other layers with a data source running on JBoss Wildfly 13 server.

UI Design

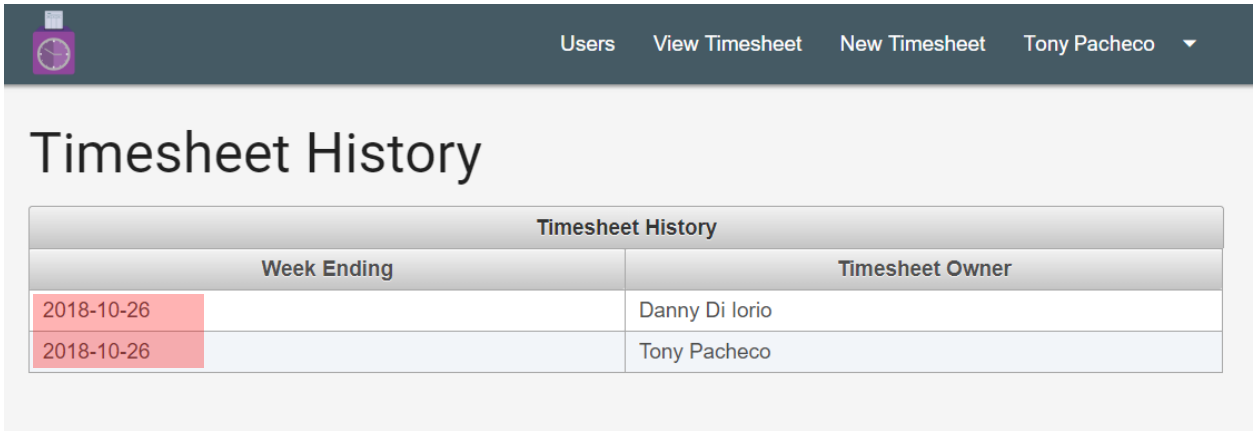
We plan to use the overall UI design created for Assignment 1, with these minor changes in mind (highlighted on the screen shots where possible):

- Improved representation of the links to each timesheet on the Timesheet History page, instead of just text
- Improved representation of the ability for the user to edit each row in a new timesheet – previously simply represented by a pencil icon at the end of each row.
- Improvements to growl message look – change based on severity or type of message, and content – either remove message header or fix bug of header repeating the message itself.

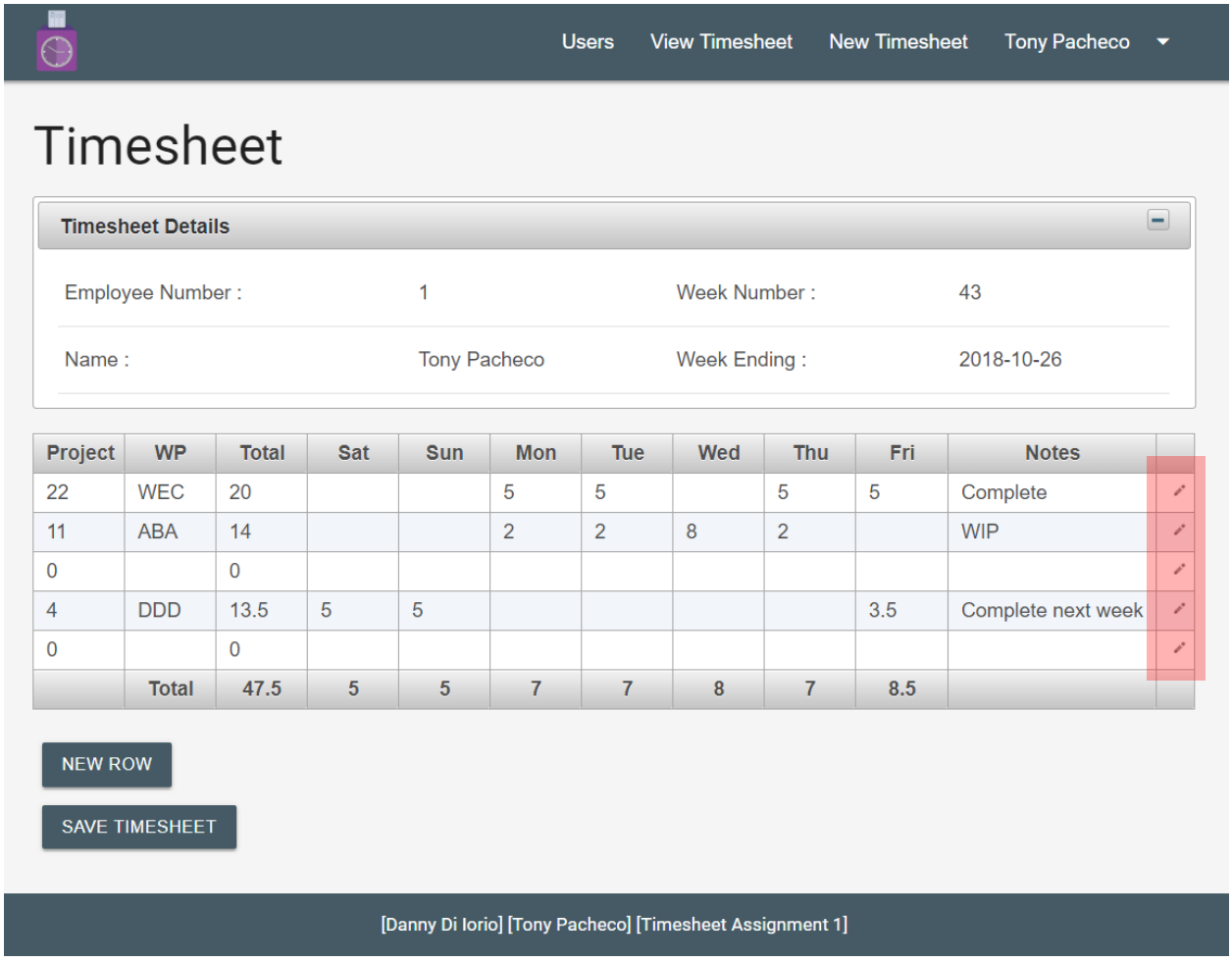
Login page



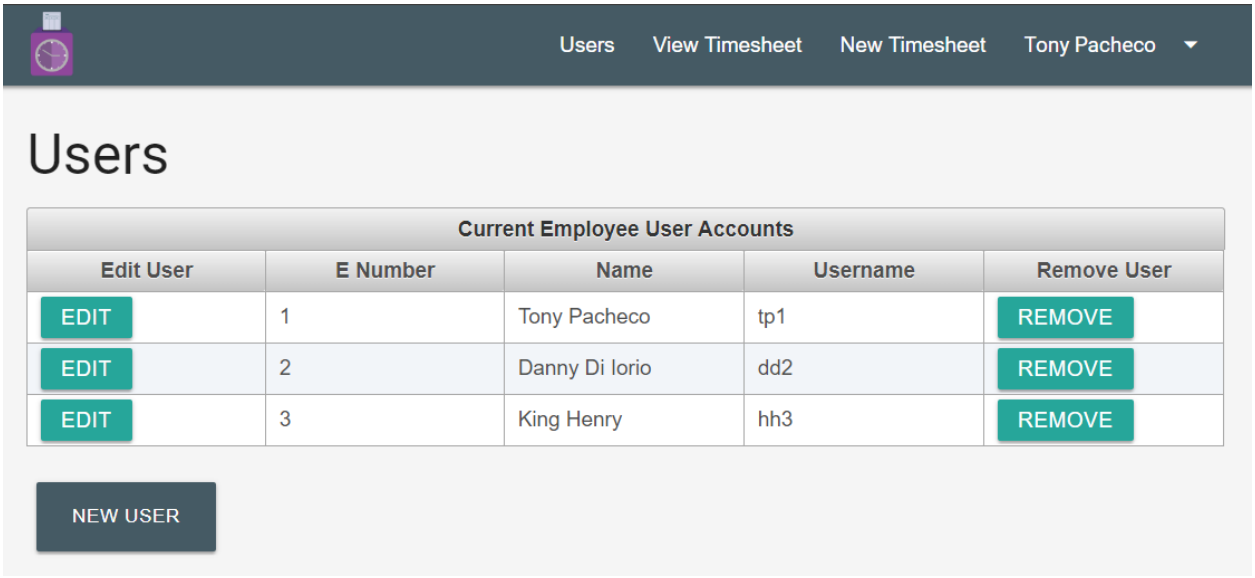
View Timesheets page (admin)



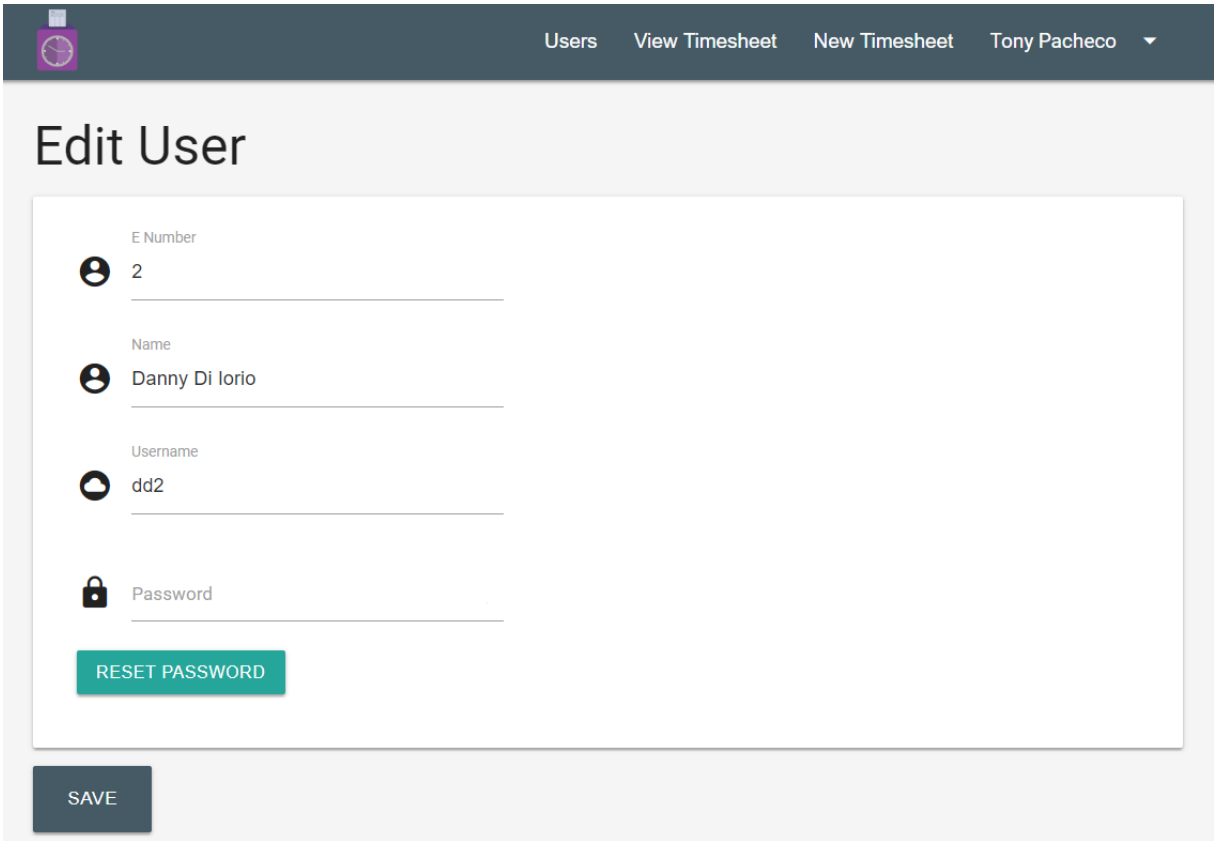
New Timesheet page (admin)



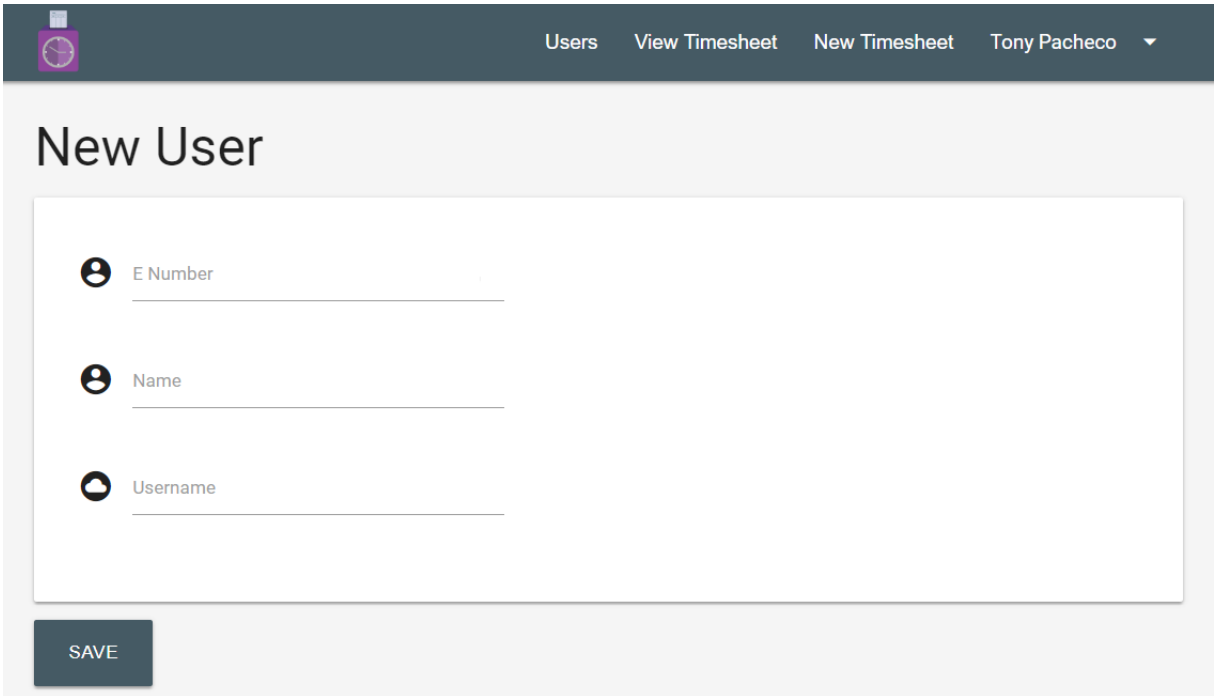
View users page (admin)



Edit user page (admin) – new user page is the same except no Password field or Reset Password button



New user page (admin)



View Timesheets page (regular user)