Time Sheet Web App
COMP3910

Assignment 2 Report

Tony Pacheco & Danny Di Iorio

November 18, 2018

Table of Contents

Purpose 3
Scope 3
Perspective 3
Class Diagram 4
Database Entity Relational Diagram 5
Use Cases 6
Interface Requirements 7
    General Users 7
    Administrators 7
Operating Environment 7
UI Design 8
Build and Run Instructions 12
    Configurations 12
    Importing Project into Eclipse 12
    Set Up Database and Datasource 12
    Build and Deploy Project on Server 13
User Accounts 13
Test Plan Results 13
    Known Bugs 14
Extra Credit Work 15

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

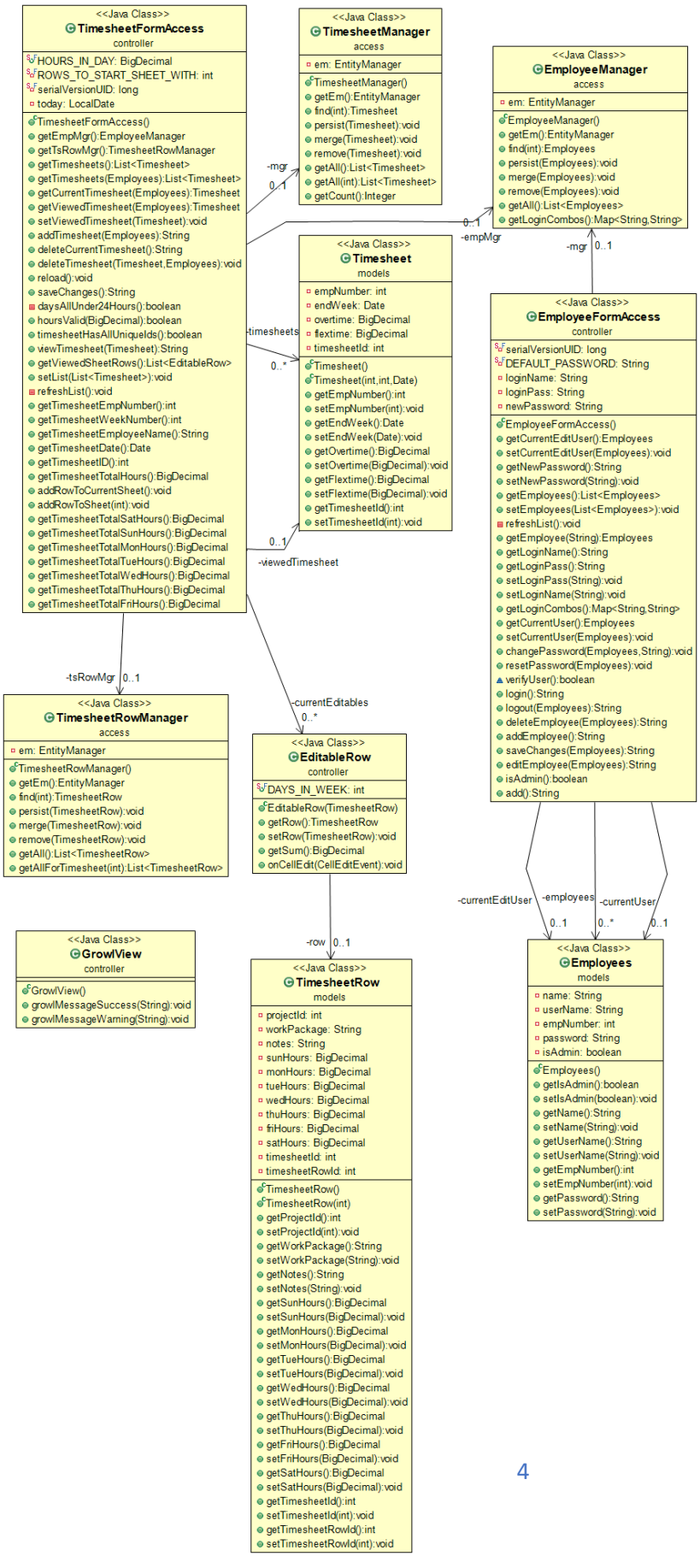
The application will store the following data in its database:

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

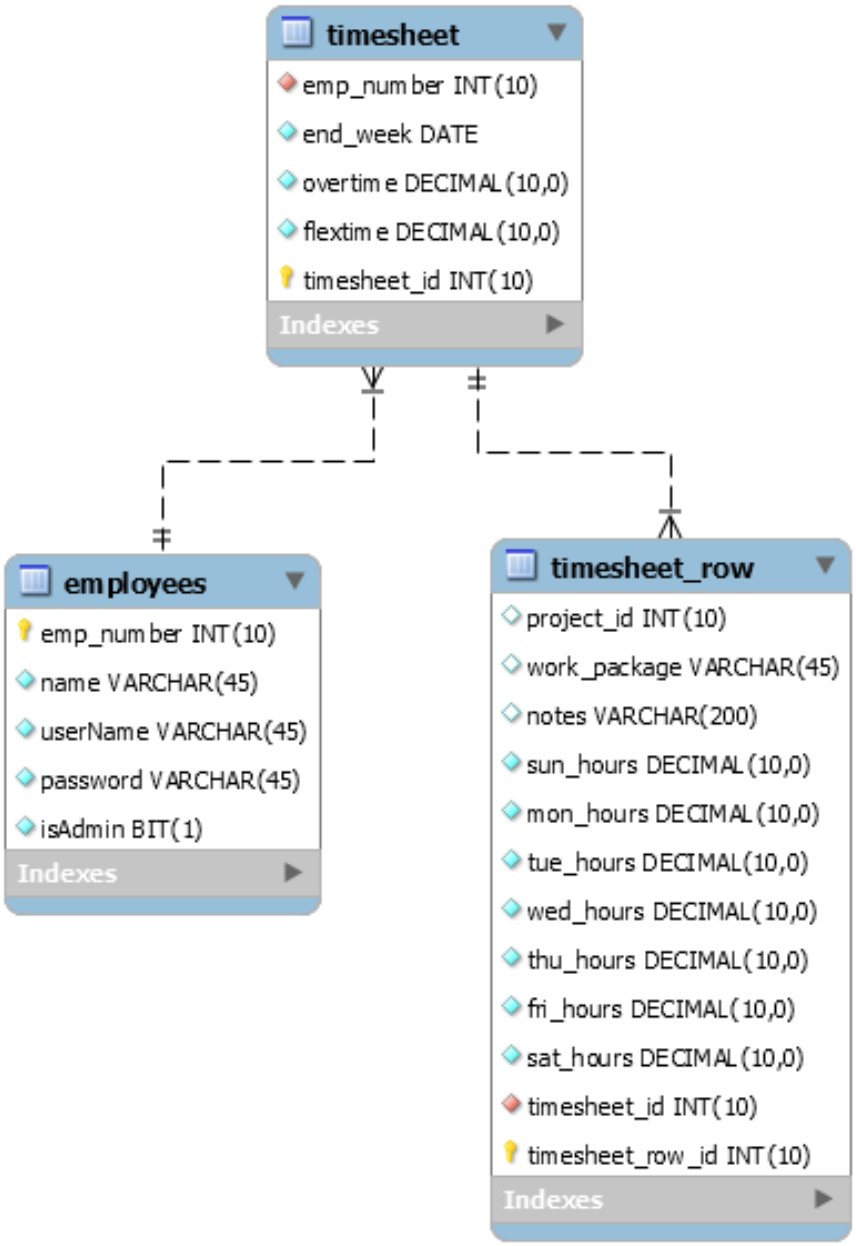
2. Timesheet Data:
    Each timesheet records the hours worked by a single employee on a given week. Each will contain the following data:
    a. Employee Number
    b. Employee Name
    c. Week Number (0-52)
    d. Week Specifier (the date on which the week ends – Friday)
    e. The following, each 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

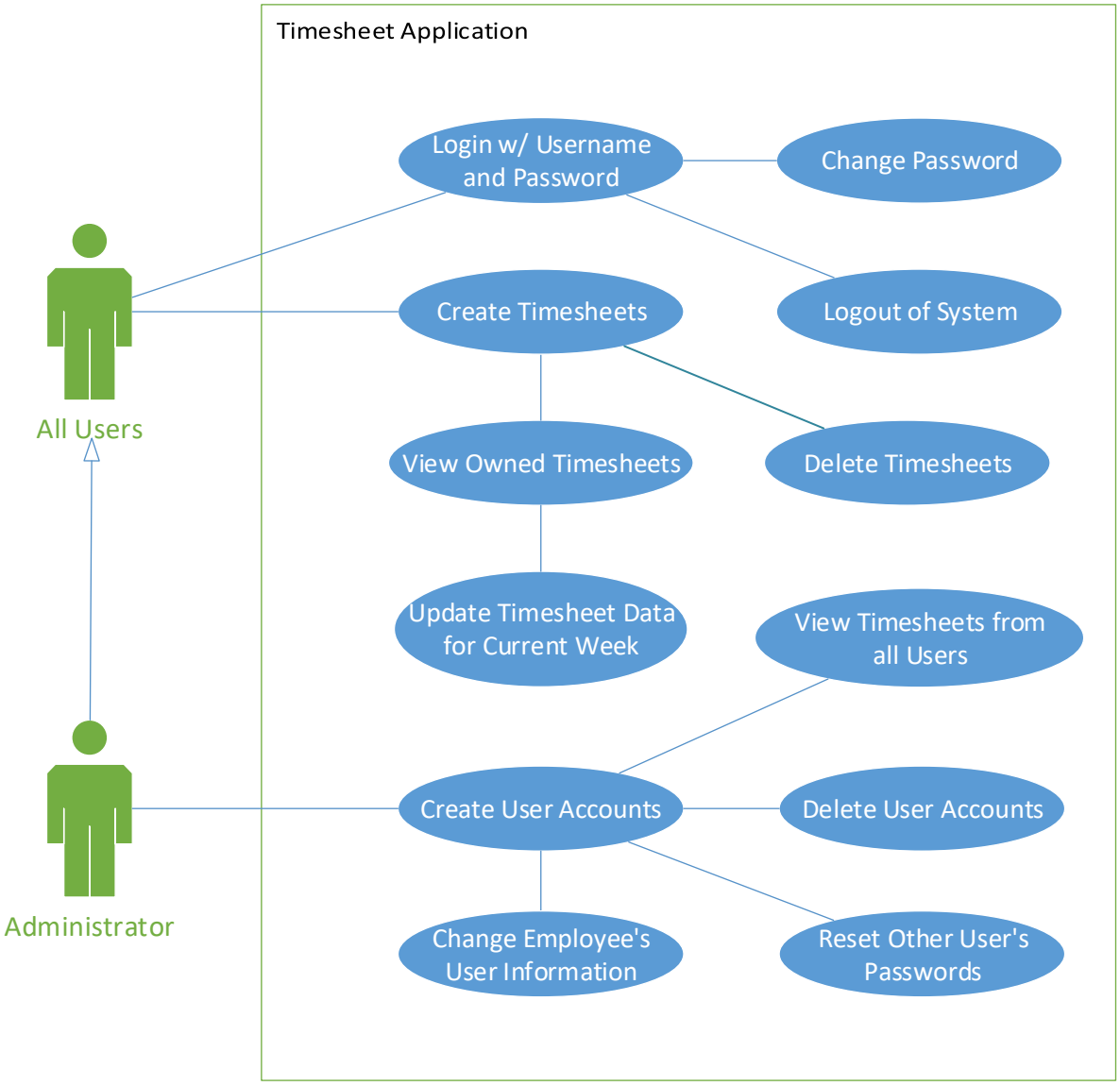


Database Entity Relational Diagram



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, edit, and delete 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 each cell, the user will be able to edit the contents of that cell in the table, which allows them to input/edit data as required.

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 name and username 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.

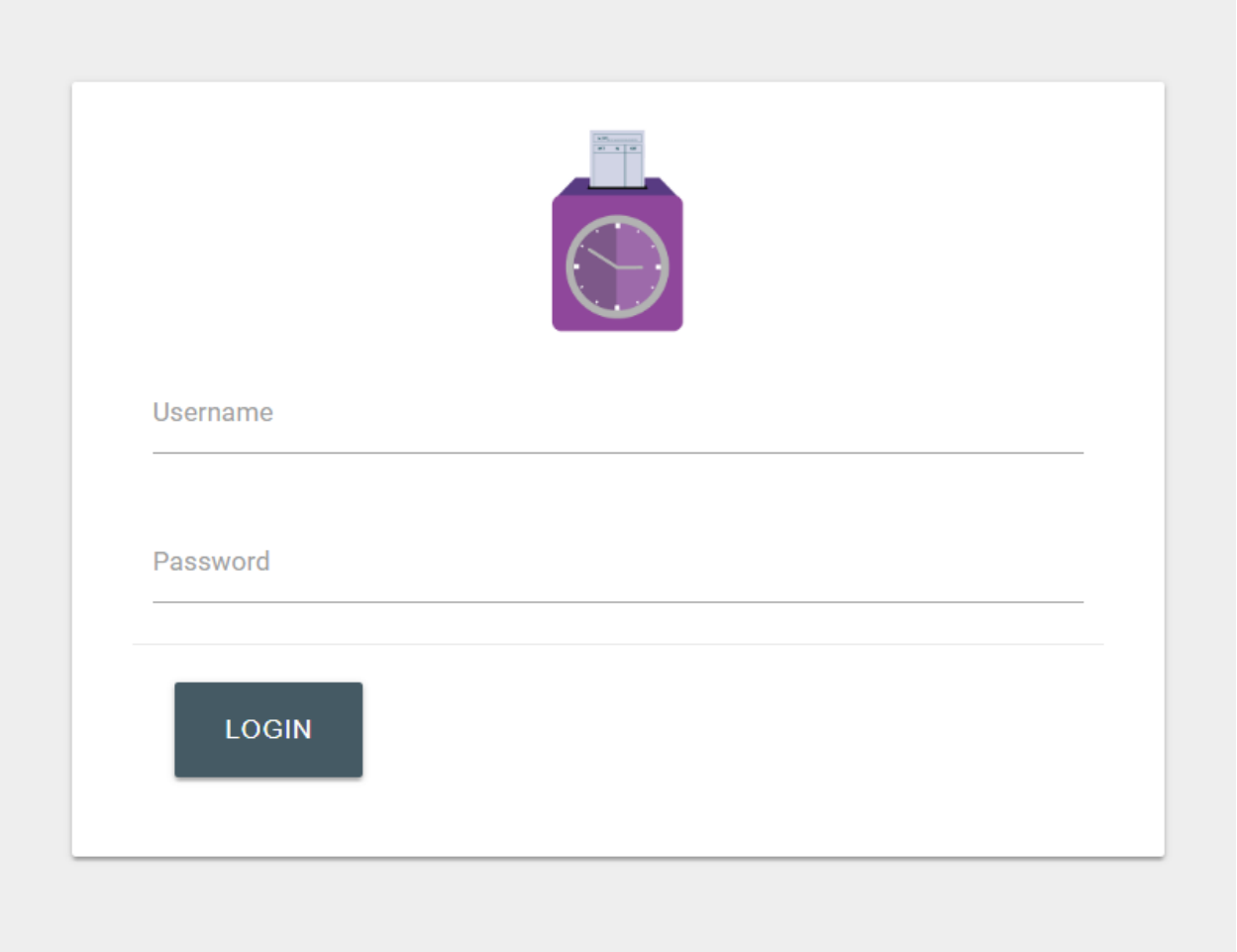
The *persistence tier* consists of a MySQL relational database designed by us. The database will be connected to the other layers with a data source running on JBoss Wildfly 13 server.

UI Design

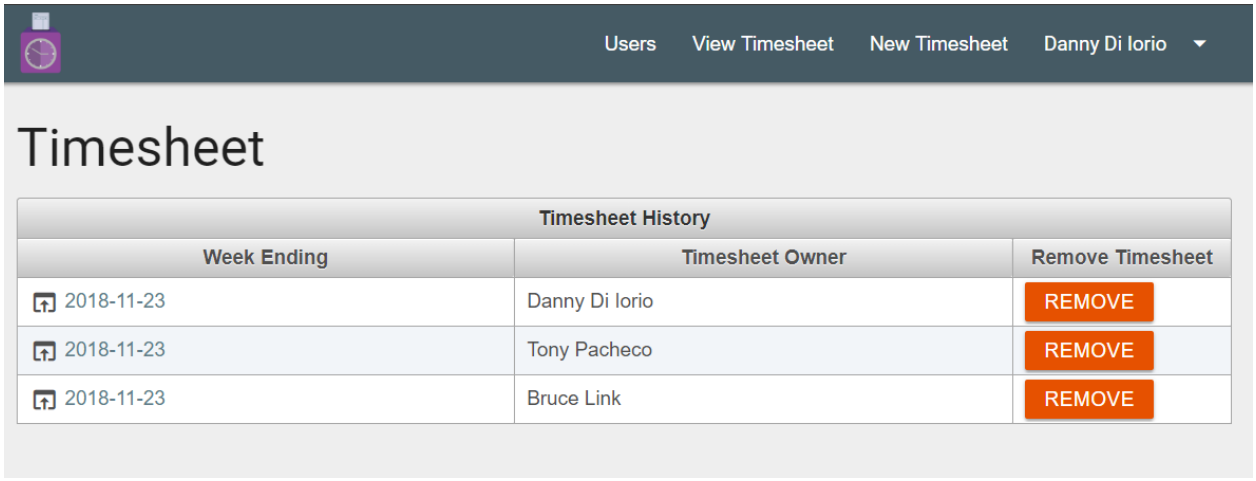
The overall UI design was adapted from Assignment 1, with these minor improvements added:

- Improved representation of the links to each timesheet on the Timesheet History page, instead of just plain text
- Timesheet cells are individually editable instead of the whole row changing into an editable row
- Improvements to growl message look by changing severity based on the type of message and content, and fixed bug duplicating message as both growl subject and content

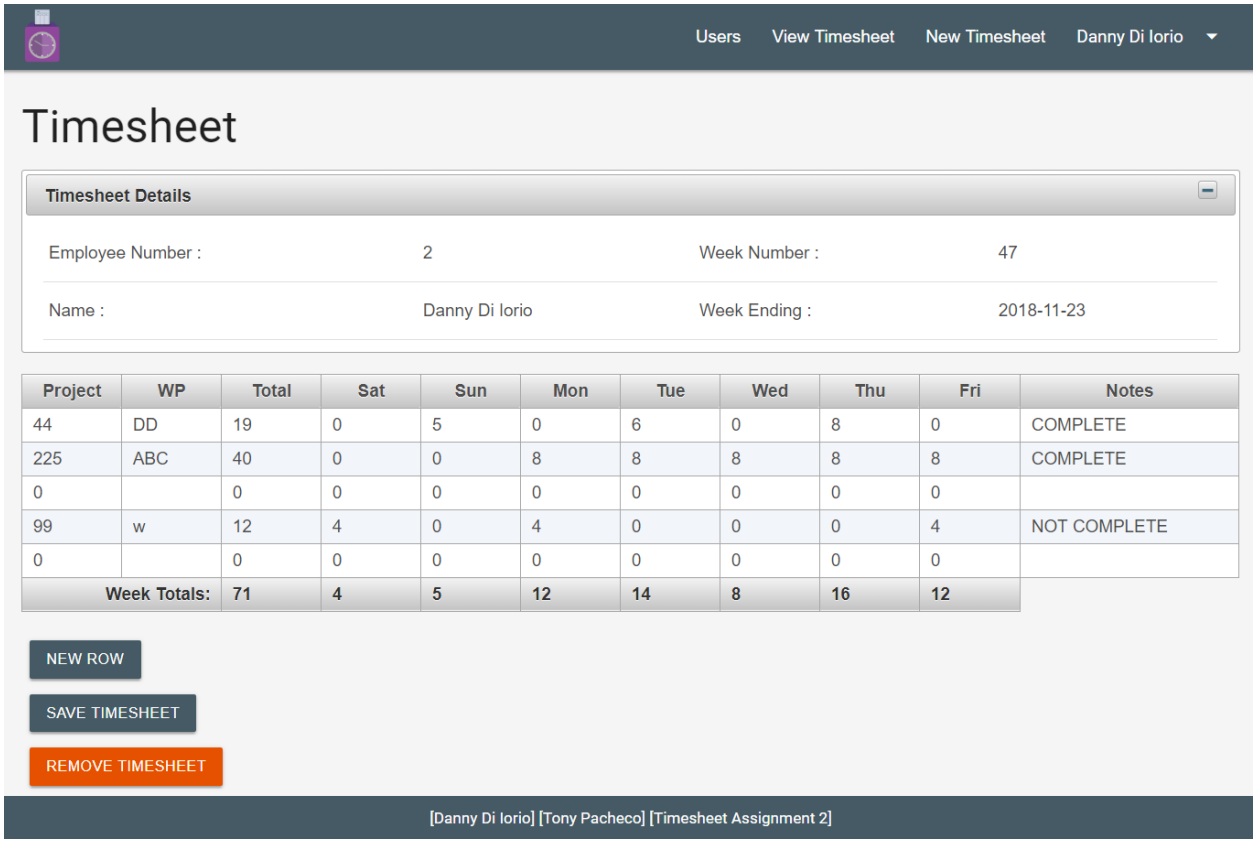
Login page



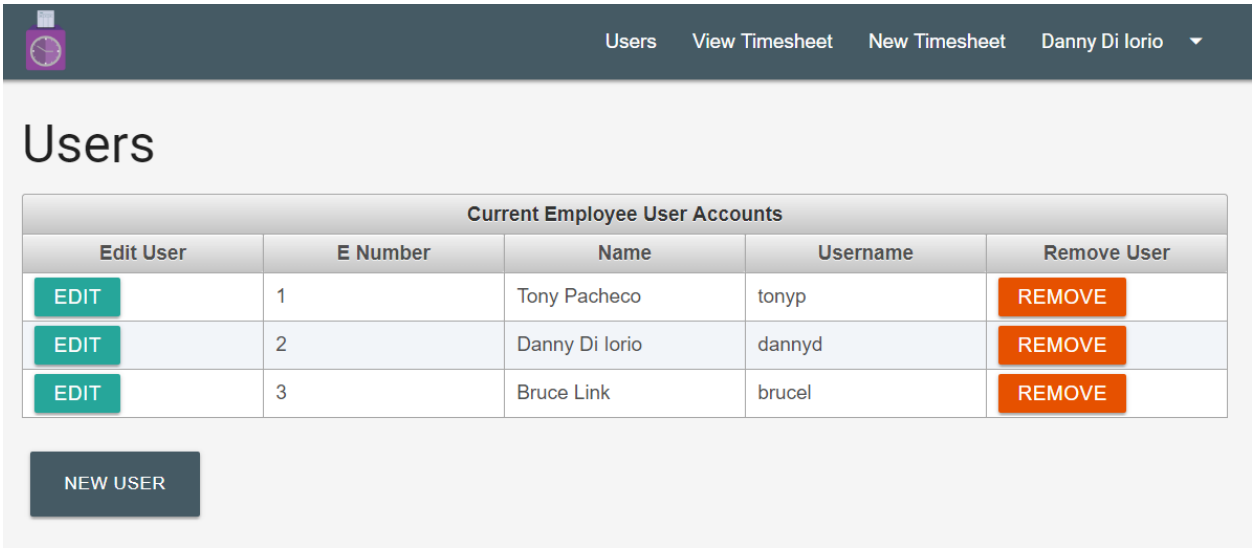
View Timesheets page (admin)



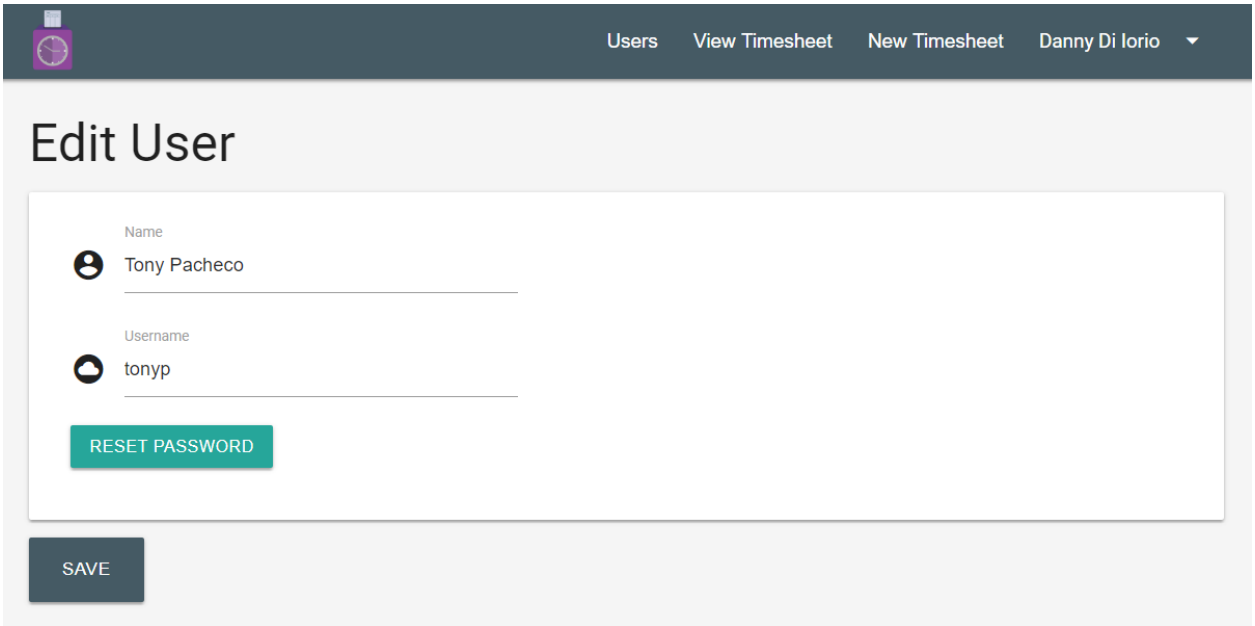
New Timesheet page (admin)



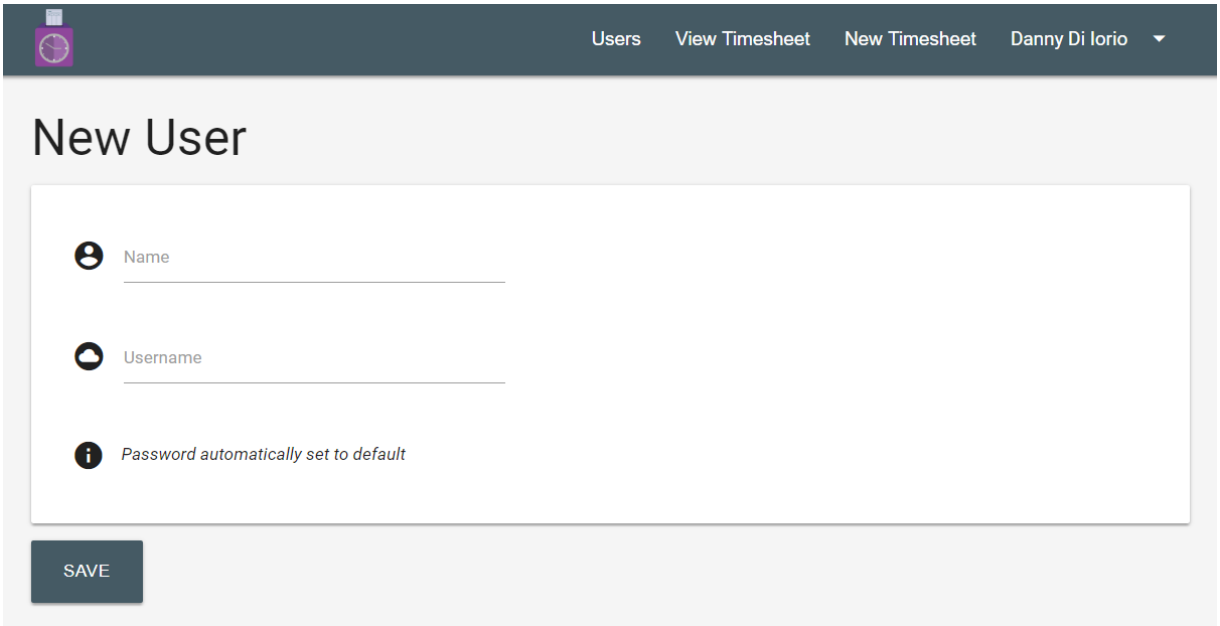
View users page (admin)



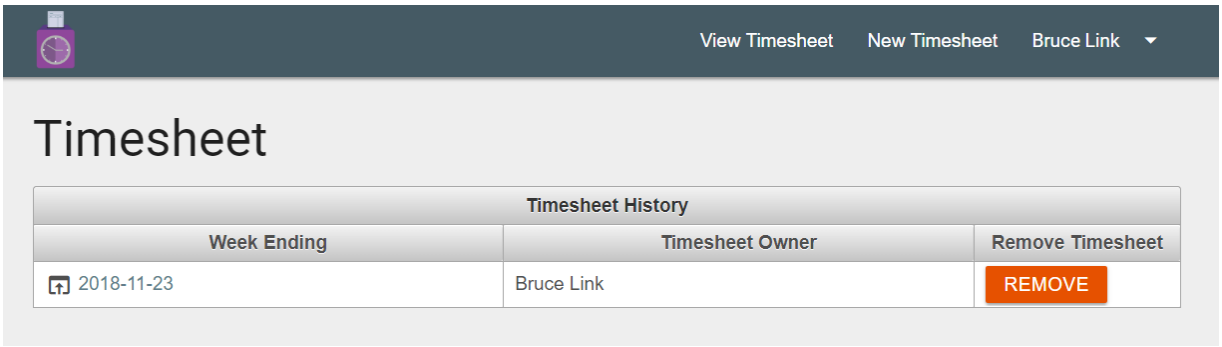
Edit user page (admin)



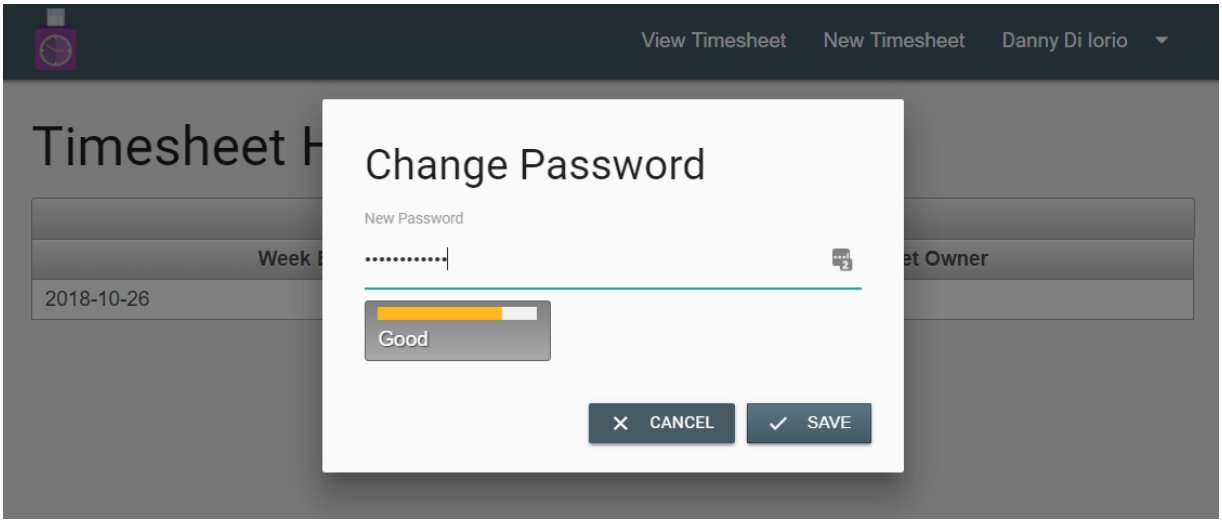
New user page (admin)



View Timesheets page (regular user)



Change password dialog



Build and Run Instructions

Configurations

- Java Development Kit 8
- Eclipse Java EE IDE
- Wildfly 13 and JBOSS_HOME defined
- MySQL 8
- MySQL user “timesheetuser” with password “password”
- MySQL Workbench

Importing Project into Eclipse

1) Unzip submitted zip file into your chosen directory
2) In Eclipse, select “File -> Import…”
3) Select “JSF Project”, select next
4) Find “web.xml” file from the project directory (WEB-INF folder)
5) Press the next button
6) Select 3.1 in the dropdown for Servlet Version
7) Ensure Wildfly 13 is selected as Target Server
8) Select “Finish”

Set Up Database and Datasource

1) Log in to your MySQL database with root user access
2) **Note:** If using MySQL 5.6 or older, manually delete “timesheetuser@localhost” and “timesheetuser@%” if they exist before running SQL script
3) Create the database using one of the `create.sql` files, use `create_mysql_5_6.sql` of you have version 5.6 or earlier of MySQL installed, or `create_mysql_5_7.sql` for MySQL 5.7 or later. ***Files are located in ‘Documentation’ project folder***
   a) Open MySQL Workbench and open the `create.sql` file that matches your MySQL version, run it to create database, OR
   b) Run the command “`source create.sql`” using command line – replacing `create.sql` with the name of the sql which matches your MySQL version as noted above
   c) This creates the timesheet database and grants timesheetuser all access
   d) Verify that the database has been created with some sample data inserted
4) Add datasource with the following configuration:
   a) JNDI: java:jboss/datasources/timesheetDB
   b) Choose MySQL as driver
   c) JDBC Connection URL: jdbc:mysql://localhost:3306/timesheet
   d) Enter username “timesheetuser” and password “password”

Build and Deploy Project on Server

1) In Eclipse, stop WildFly server and delete any previously deployed versions of this project
2) Start the WildFly server
3) Right click on the project folder in Eclipse
4) Select “Run As -> Run on Server”
5) **If the project is launched in the Eclipse browser, close it and open a browser (Chrome ideally)**
6) Enter “http://localhost:8080/assignment1” in the address bar to view

User Accounts

Admin:

- Username: tonyp
- Password: pass
- Username: dannyd
- Password: password

User:

- Username: brucel
- Password: pass

Test Plan Results

1) Login page – **ALL PASSED**
   a) Users will enter a correct combination of password and username
   b) Validation checks these fields and displays error messages appropriately
2) Logout – **ALL PASSED**
   a) User can logout of their account by pressing the logout button in the header dropdown menu
   b) After pressing the logout button, they are directed to the login page
3) User landing page – **ALL PASSED**
   a) Page with a list of saved timesheets (empty at first)
   b) Header containing links to View Timesheets, New Timesheet, User dropdown menu
4) Administrator landing page – **ALL PASSED**
   a) Same page as users
   b) Header will have an additional link Users, which will lead to a list of users
5) View timesheet – **ALL PASSED**
   a) This page shows a list of saved timesheets, organized by date
   b) The timesheet owner is shown
6) Create timesheet – **ALL PASSED**
   a) When a new timesheet is created five rows will be present
   b) User and week details are listed above the timesheet

c) It will be defaulted as current week
d) Clicking on the add row button will add a row to the timesheet table
e) Cells are all editable except totals
f) Time entered during the week must be in the unit of hour from 0.0 to 24.0, it may be integer or with one decimal place. Error message shows if validation fails
g) Error message appears if hours for a day column total more than 24.0, and timesheet is not saved until they are corrected
h) WP (Working Project) must have alphabetical value combined with numerical value
i) Total number of hours should appear before the column of the day of the week column
    i) The calculation is done automatically; however, it is only done when the user clicks the save button

7) View users page – **ALL PASSED**
    a) This page will show all the users in a list with all their information
    b) The admin can click on a button to edit, directing them to an edit page
    c) The admin can also click on a remove button to delete the user
    d) The admin can click a button to add a new user
8) Edit users page – **ALL PASSED**
    a) The admin can edit all fields related to the user
    b) The user's password can be reset to a default value ('*1234'*) by clicking the reset password button
    c) Validation messages show if required fields are empty
9) New user page – **ALL PASSED**
    a) This page allows a manager to create a new user by filling out the following fields:
        i) Name - required
        ii) Username - required
        iii) Password
        iv) Note – employee number is auto generated and not shown on this page
    b) Validation messages show if required fields are empty

Known Bugs

- Sometimes after the page refreshes, such as after deleting a user, the drop down in the header does not work. Clicking on a link to reload the pages fixes it
- Empty timesheet cells contain 0's instead of being blank
- In rare cases, after clicking the 'Remove' button on the Timesheet history page, the removed timesheet still shows in the table (although it is removed from database). Clicking View Timesheets in the header refreshes page and fixes it.

Extra Credit Work

- Primefaces tag library used throughout application, including custom growl messages for all validation
- Requirements and design document submitted early (November 11th)
- Application implemented using JPA instead of JDBC