login - done

logout - done

set business hours - done (under revision)

alter availability - done (under revision)

select employee - done (under revision)

generate schedule - chad (does this include generate schedule? since they both display schedule) separate cases

display schedule - chad

-----------------

name: login

participating actors: employee or schedule manager (User)

entry condition: null

exit condition: User is on the Main Menu

flow of events:

1. CDJ Scheduler displays with login form: text boxes to enter username and password along with a submit button

2. User (Employee or Schedule Manager) enters login information and then clicks the submit button.

3. CDJ Scheduler queries data store with submitted user information, then compares to see if login information is valid.

a. If invalid user information has been submitted, CDJ Scheduler shows error message and the user is able to attempt to login again.

b. Once valid login information is submitted and verified, CDJ scheduler closes login screen and displays main menu according to user role.

quality requirements:

-----------------

----------------------------------

name: logout

participating actors: employee or schedule manager (User)

entry condition: User is on the Main Menu

exit condition: User is on the Login Screen

flow of events:

1. User clicks logout button from main menu

2. CDJ scheduler closes main menu, and brings up login display

quality requirements:

----------------------------------

----------------------------------

Name: Set Business Hours

Participating Actors: Schedule Manager

Entry Condition: Schedule Manager is logged into CDJ Scheduler

Exit Condition: If the set business hours are saved, then data is saved to data store and Schedule Manager returns to main menu. If business hour are not saved, Schedule Manager just returns to main menu.

Flow of events:

1. Schedule Manager initializes use case by clicking Set Business Hours on the Main Menu.
2. CDJ Scheduler replaces the Main Menu screen with the Set Business Hours screen.
3. Schedule Manager selects the days that the business will be open for business by setting checkbox values to true.
4. Schedule Manager then selects the open and close times for the days selected.
5. Schedule Manager allocates the number of resources needed by filling in the Number of Employees textbox.
6. Once information is filled, Schedule Manager clicks the Save or Back button.
   1. If the save button is clicked, the information is then saved to the data store and the Schedule Manager clicks Back to return to the CDJ Schedulers Main Menu.
   2. If the Save button is not clicked, the Back button can be clicked and will return the Schedule Manager back to the CDJ Schedulers Main Menu.

quality requirements:

----------------------------------

----------------------------------

name: alter availability

participating actors: employee or schedule manager (User)

entry condition: user is logged in to CDJ Scheduler

exit condition: if valid data has been submitted to be saved, then the data store is updated with this new information

flow of events:

1. User initializes use case by pressing the Alter Availability button on the Main Menu.

2. a. If user is Schedule Manager, CDJ Scheduler displays Select Employee screen (see Select Employee use case)

b. If user is Employee, CDJ Scheduler displays a screen with a table, a save button, and a back button (purposely similar to Set Business Hours screen). The column headers of the table are the days of the week, each with a checkbox. The first row header is “Start” and the row contains a drop down menu of times in each column. The second row header is “End” and the row also contains a drop down menu of times in each column. CDJ Scheduler defaults the values of the checkboxes and drop down menus to the values set by the Schedule Manager on the Set Business Hours screen, unless the user has previously saved data, in which case the values will default to the most recently saved values.

3. User updates availability by checking boxes of days he/she is available to work and selecting available times from the drop down menus.

4. User then has the option of attempting to save by pressing the save button(which will fail if a availability is not within business hours or if user selects negative hours), storing the schedule in the data store, and then pressing the back button to return to the Main Menu.

quality requirements:

----------------------------------

----------------------------------

name: select employee

participating actors: Schedule Manager

entry condition: Schedule Manager is logged in to CDJ Scheduler

exit condition: Schedule Manager is on Alter Availability screen for selected employee

flow of events:

1. Schedule Manager initializes use case by pressing the Alter Availability button on the Main Menu.

2. CDJ Scheduler displays a screen with a drop down menu of employees, a submit button, and a back button

3. Scheduler manager selects an employee from the drop down menu and hits the submit button.

4. CDJ Scheduler displays the Alter Availability screen (see Alter Availability use case) for selected employee.

quality requirements:

----------------------------------

----------------------------------

name: generate schedule

participating actors: Schedule Manager

entry condition: Schedule Manager is on the Main Menu

exit condition: Schedule Manager is on the Main Menu

flow of events:

1. Schedule Manager initializes use case by pressing the generate schedule button on the Main Menu.

2. CDJ Scheduler displays a screen with a drop down menu with options for selecting scheduling priority, a button each for generate schedule, save, and back, and a blank text area.

3. Scheduler manager chooses scheduling priority and hits the generate schedule button.

4. CDJ Scheduler attempts to create a schedule based on given employees availability.

a. If it is successful, then the schedule is displayed in the text area on the screen.

b. If there is an error, then a message describing the problem and information to help solve that problem is provided in the text area.

5. The Schedule Manager then has the option of attempting to save by clicking the save button (which will fail if a schedule was not successfully generated), storing the schedule in the data store, and then pressing the back button to return to the Main Menu.

quality requirements:

----------------------------------

----------------------------------

name: display schedule

participating actors: employee or schedule manager (User)

entry condition: Schedule Manager is on the Main Menu

exit condition: Schedule Manager is on the Main Menu

flow of events:

1. User initializes use case by pressing the display schedule button on the Main Menu.

2. CDJ Scheduler displays a screen with a text area and back button.

a. If there is a schedule in the data store, that schedule is displayed in the text area for the User.

b. If there is not a schedule in the data store, an error message is displayed in the text area.

1. User can then view the schedule and press the back button to return to the Main Menu.

quality requirements:

----------------------------------

notes:

-display schedule being included in generate schedule? (we suck at understanding these association relationships in use cases) Chad- I do not believe so. They do the same sort of thing, but it is simpler to say they are separate

-does user viewing the main menu need to be listed in the entry condition or is it implicitly understood (perhaps it is obvious that the use case is initialized from the point that it can be initialized at, which is listed as the first step in the flow of events) Chad- We should list viewing the main menu as the entry and exit conditions of almost all of these. The way Onyeka described the conditions was how does the user know? Which differs from what I felt the meaning was out of the book.

-would this be our exit condition for each of the update functionalities or, since you can back out without actually updating, is there technically no real exit condition "if valid data has been submitted to be saved, then the data store is updated and current with this new information" Chad- I think that we mention this in the flow, and go with above exit condition

-does select employee include or extend alter availability Chad- The way we describe the system now, it is definitely includes. If we want to do extends, then select employee cannot be accessible on its own, and has to be initialized by the alter availability use case. If it can be accessed directly, then it is first class. And the way we discussed it, technically, as Schedule Manager, you access select employee first and then alter availability (which we should note that there are two different ways to enter into alter availability in the description)