**Update log for Creel Application**

1. May 14, 2024 – initial push of Brook Beaverly’s creel planning tab.
2. June 6, 2024 – Dan’s fork with some ideas to streamline and tidy up parts of the planning tab and a few bug fixes.
   1. Bug-Winter quarter was getting 4 extra days instead of 2 extra for each strata because it is split by year. I added Qyear as a new year variable for picking dates and winter months of Jan and Feb were assigned the Qyear from the previous December’s year (i.e., year – 1), then I grouped on this variable when picking random sampling dates so only 2 extras were picked.
   2. Bug-occasionally, no target count times are produced (they were NA)
      1. This is occurring when random\_start\_time was exactly 300 less than end time (5h), which happens if runif(n()) is over 0.9949
         1. Fixed by deleting line checking if the time was < (endTime – 300) because with max runif(n()) value of 0.99999, the time will always be no greater than (endTime – 300), which allows 60 min for pressure count, 60 min between counts, and 60 more minutes for a final pressure count. There is no need for ifelse to produce NA values if time was exactly (endTime – 300)
         2. Also required Pressure\_count\_2 to be modified similarly (and no longer looking for na pressure\_count\_1)
   3. Moved military/standard time conversion outside main loop so it won’t recalculate dates just to convert between time formats...now handled in creelScheduleTimeFormated() reactive
      1. Also clean up all time conversions using my modified code that gets am/pm in all standard times and ensures military times have leading zeroes as needed
   4. Cleaned up data table display by using the creelScheduleTimeFormated() reactive code to select and then rename columns to be used in the data table and then used the output of this reactive as input for the renderDT statement.
      1. Got rid of extra variables used in calculations that user does not need to see and reorder data table that is displayed as follows: Quarter, date, dayName, shift, strata, start time, end time, direction, pressureCount1, pressureCount2
         1. This fixed issue where column order differed between the military and standard time display
      2. Renamed columns with more complete names that include spaces so the table displays well without messing up R naming conventions (ok to have names that are not very R compliant as we won’t do any more manipulation with this dataframe before displaying...so in future any edits of the creelScheduleTimeFormated() reactive should be made above the rename command)
   5. Set column width targets for several columns in data table (used columnDefffs option in datatable command) with long column labels to force them to line wrap rather than taking an inordinate amount of column space.
   6. Altered datatable to have 100 rows by default but can select from 50 to up to 400 using dropdown selector on interface.
   7. Use .groups= "drop\_last" (default action) for all summarize/summarise statements to avoid getting warning about overriding using .groups. This just helps keep the log clean so we can more easily find problems if they occur.
   8. Added lake name variable to data table
      1. Made some modifications so only one lake can be selected but no lake is selected by default, then used reactive value to show “No lake selected” until user picks a lake...this avoids error preventing display of datatable when no lake is yet selected.
   9. Altered download table to be the new table I created to display as data table so it matches what the user saw on screen (i.e., my code above altered the approach, so the data table being downloaded was now incorrect).
      1. This also fixed code to export table exporting wrong time codes (was using military time but with 2 digit hour and minutes as 2 digit decimal of an hour)
      2. Changed naming scheme to include lake name and start/end dates for entire creel
   10. Rearranged the boxes at the top of the page to make it take a bit less space and show more of the table on the screen when scrolled to the top of the page.
3. 11/21/2024 First official betta version pushed to server...not sure I captured all of the changes, but here is most of the changes made
   1. Fixed issues with daylight savings being one hour in the wrong direction.
   2. Fixed bug that caused variable number of days in each quarter...was solved by re-coding a different approach to prevent alternates from being picked from days already as standard days (was just throwing those out, now we pick from a list of days with standard days removed ahead of time).
   3. Scale number of alternate days so it is 2 weekeday/weekend minimum, but does an additional 2 days each for every 10% more effort selected.
   4. Sort output by date but with alternates at the end of the quarter rather than interspersed with standard days
   5. Fixed some issues where we were coming up 1 day short of the number of days we wanted due to difftime not counting either first or last day.
   6. Wrote code to set default starting date equal to the first day of the next quarter unless the current date is within 2d of the start of the current quarter, in which case it uses the current date (but this will throw 1 week of the same season at the back end of the year...so might not want to allow this past the first day of the month just to essentially avoid splitting a quarter at all).
   7. Rewrote date selection code block to ensure it does not ever pick both a morning and evening shift on the same day...did this by first picking the proper number of days from a list of available dates (not the full list of am/pm shifts), then use an innerjoin and slice\_sample(n=1) on each date to pick which shift is used. This logic also needed to be used to account for Friday am being weekday and Friday pm being weekend (used a filter to remove picked dates to deal with this)
4. 11/25/2024 – Several enhancements as itemized below
   1. Fixed bug that sorted seasons alphabetically instead of chronologically
   2. Rewrote parts of the code and added bookmarks for better navigation of code
      1. Separated military/am/pm time code from other functions
5. 12/10/2024 – bug fixes by Dan Shoup
   1. Several afternoon shifts in November and December had incorrect end time (used morning end time).
      1. Was a copy-paste error in code (related to code fix for new way of handling daylight savings time that used lubridate’s sdt\_start and dst\_end functions)
      2. In fixing this I noticed times appeared flipped for daylight savings vs standard time. I fixed all of these so the DST is the later time and standard times are the earlier time for both spring and fall
   2. In testing, I also noticed that if you manually type a date in start\_date dateInput box, it locks (no error in the console...just everything freezes). It appears this is because it is updating with an invalid start date before the user leaves the dateInput box
      1. This is caused by the fact that dateInput boxes will try to render a date from the partial characters...for example 01-01-202 will give January 1 in the year 202...which is negative in the as.date mode and causes math errors in our app
      2. To fix this, I produces some error trapping (good\_dates() and end\_date\_updated() flag) so start\_date cannot ever be earlier than 01-01-1970 (the date = 0 in as.Date format), must be earlier than end\_date, and not earlier than 7d before today. Also produced code so end\_date does not update if not needed (checks end\_date\_updated flag). Also use the above procedures to block other code from running until both dates are set. This prevented rendering the output table twice.
   3. Found issues calculating end date for leap years, fixed with series of if statements for picking initial dates for start and end date that check for leap years on either date.
   4. Created error messages for user if start and end date are not meeting validation rules.
6. ????-next update
   1. Fixed bypo on military vs AM/PM button (had “sm” rather than “am”)