Sprint 2

Scope:

Class cards:

-Class cards will render with fake temporary data

-Class cards will render with real data from API

Credit Hours:

-The total will update and display as classes are added/removed (removing is not Implemented yet.)

Schedule:

-The UI of the schedule grid will be implemented

-Schedule cards will be displayed (fake temp data)

-Unscheduled classes display

Permutations

-Scheduling algorithm is implemented

-Pinning and blocking prepared for next print

File Structure Breakdown

Same as sprint 1

Structure components:

ClassCard.tsx: Update class cards with selected card data

ScheduleCard.tsx: render section times and instructor

ClassGrid.tsx: Display blocked time slots

PermutationSelector.tsx: Users view and cycle through permutations of schedules

TotalCreditHours.tsx: Display updates to credit hour from selected courses.

Features:

ScheduleSlice.ts: Manage new state values blocked times, increment through permutations.

Utils:

Utilities.ts: Added functions getScheduled section, getUnscheduledSections, getPermutations.

getCourse.ts: Parse data from SearchServer

Server:

SearchServer.ts: connect to API, functions getSearchMessage() and get()

Components:
Data Flow:
Search Input - The Search bar sends term to SearchServer

Data Parsing - data is parsed through utilities.parseHTMLResponse into ClassData and SectionData, and updates ScheduleState.

Rendering:
ClassCard display course data

ClassGrid, GridBox render times based on ScheduleTime

PermutationsSelector users scroll through permutations

TotalCreditHours updates when users remove or add classes

Scheduling Algorithm:
Computes all valid permutations of classes that do not overlap and stores them in the Redux state. Current view stored by currentPermutation and change with incramentPermutation and decramentPermutation