1. solutions browser:
   1. add the ability to move 10 solutions forward/backward  (<</>>) (instead of just 1 per click).
   2. assign soft constraints:
      1. redefine the adjust the initial "teacher availability" to add "nice to have" constraints
         1. i.e. add more unavailability hours/days and check if any solution is still available
   3. save found solutions:
      1. reformat the parameters file and save everything into a binary file.
2. school definition UI:
   1. when defining a new school allow for a more fine-grained definition of learning times:
      1. define a value of "class duration", with a default (i.e. 45 minutes)
      2. can be table with 5 minutes timeframes (selection of hours in the same manner as the teacher availability or the student availability UI)
         1. because the class duration is already set, while the time frame is in 5 minutes, when a location is selected, it automatically fills the cells that proceeds it according to the class duration value.
      3. (optional) can be a list like that is refilled into a table (might be easier as a UI, but not sure)
         1. example: you select Sunday: and add a list: 8:00 - 8:45, 9:00-9:45, 10:15-11:00
3. student's availability UI:
   1. a student starts with a selection of a school
      1. once a school is selected, the possible learning times are automatically loaded into the UI according to that school definition in part 2.
      2. the UI then allows to select the student availability (same as we had it basically).
4. schools' relations:
   1. the school's relations UI, should add the ability to define the transit time (with a default value) from each school capsule
5. editing the solution:
   1. manually from the solutions UI:
      1. add a new student -> choose its school (define the school if none existing) -> define student availability (and hours requirements)
         1. mark with GREEN all the hours that are valid to add (none taken by others and no constraints are violated) -> do so iteratively after each selection of an hour, until all hours have been chosen
      2. edit a selected time -> click on an assigned hour from the solution
         1. in Green - all available hours that are not taken by any one nor violates a constraint
         2. in Yellow - all hours that can be switched between the selected hour and another hour that is taken by a different student (and the switch does not violate any constraint).
   2. automatically, run the algorithm to try and assign the new student with as little constraints as possible.
6. define search parameters
   1. allow for cancelation of search and remaining with the solutions that were found
   2. define a maximum running time
7. Optimize:
   1. Update algorithms.
   2. SIMD (?)

color legend:

red - high priority

orange - medium

green - low

user story (what window they see at what order):

1. define teacher availability
2. define schools
   1. define school relations
3. define students (based on school's time schedules) -> from here we should have the ability to define a school aswell
4. find solution
5. browse solutions
   1. simple browse
   2. add soft constraints
   3. edit.