GPM

User Interface Design for

Assignments → Classes → Groups

Assumptions

- **Modules** have already been defined
- **Assignments** have already been defined
  - *Assignments are components of Modules*
- **Classes** have already been defined
  - *Classes are components of Assignments*
    - *The same assignment may have several classes distinguished by year only (no parallel classes allowed)*
- **Lists of students** have already been imported
  - *Students are components of classes*
  - *Each class has exactly one list of students*
  - *A list of students may be associated with several classes*

Main Requirements

- To assign **students** in a **class** to **groups** for that class
  - Students will be identified *according to NILE conventions* by a unique **Student-ID**, a **First Name** (aka Forename, Given Name) and a **Second Name** (aka Last Name, Family name, Surname)
  - Groups will be identified by a **Group ID** and a **Group Name** (*optional: short and long*)
- The **number of students** in a class is variable
- The **number of groups** in a class is variable
- **All** students will be assigned to a **single** group only
  - the groups will cover the whole class (**all**)
  - the groups will non overlap (**single**)

Auxiliary Requirements 1

- Student ID's and other identifying attributes will be provided by the **faculty**
- Group ID's and Group Name will be provided by the **tutor**
- Within a group, students will be **sorted** according to their Family Name (and Given Name, if necessary)
- Within a group, students get an **initial student number** from 1 to **group-size** (after sorting)
  - *Note: the Student-ID would be enough to identify students; a short number or code is convenient for labelling the rows and columns in the peer matrix*

Auxiliary Requirements 2

- If a student **leaves a group** permanently, the other students will **keep their number** in the group
- If a **new student** enters the group, he / she will get the next higher number – even if there were a free student number (because of leaving students)

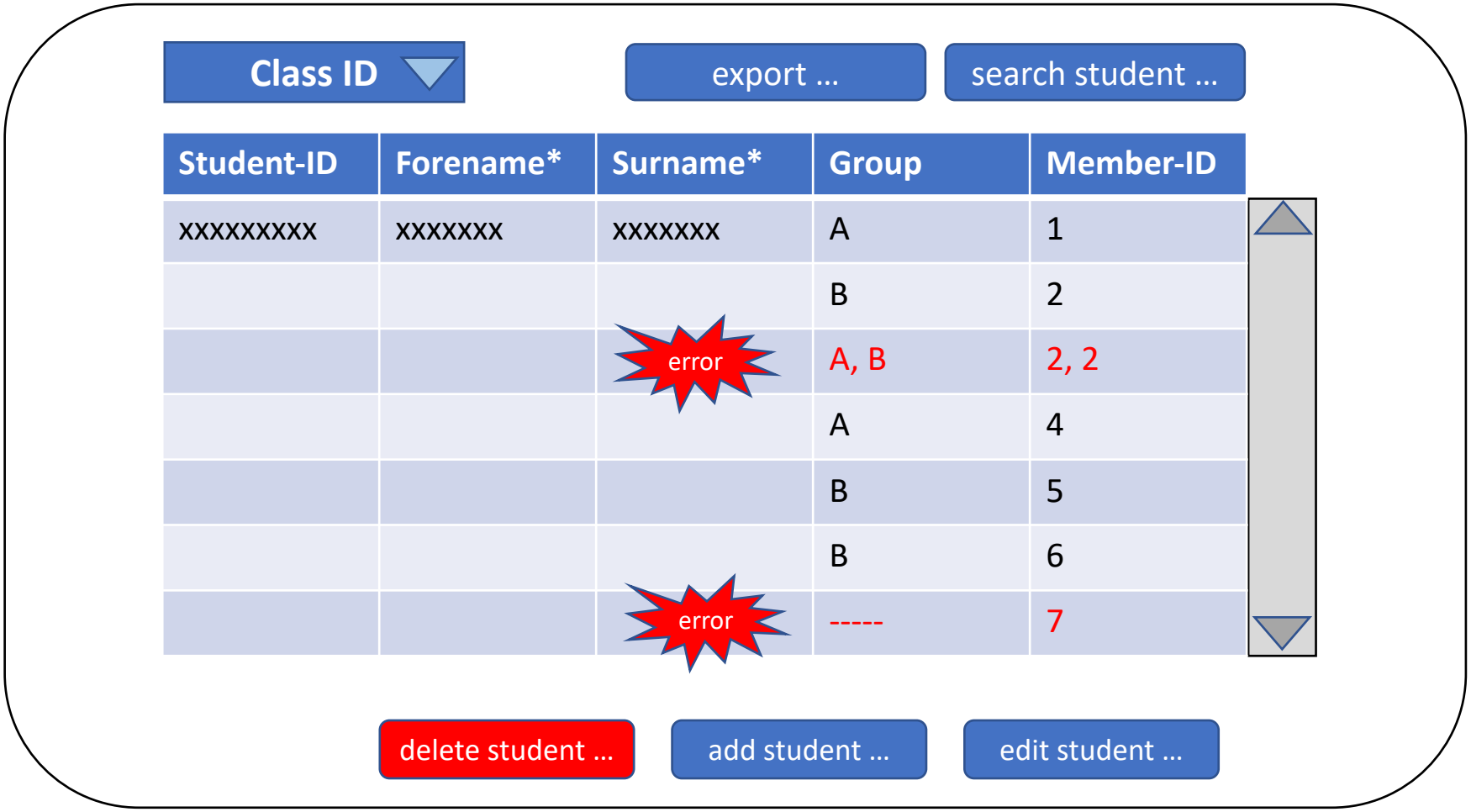
Student	Initial Number	After B leaves	After D enters
A	1	1	1
B	2	-	-
C	3	3	3
D	-	-	4

Auxiliary Requirements 3

- On request, the module/assignment/class/group **structure** (*with/without students*) will be
  - displayed on the screen
  - saved as a CSV file

  using all **identifying attributes** and the date of displaying/printing (for later checking)
- On request, it shall be possible to find out if a student has already been **allocated to a group** and if so, to which group
- On request, it shall be possible to find out if by mistake a student has been **allocated to more than one group**

Checking Students within Class / Group



Class ID is a drop-down menu; affix "..." means: a small dialogue window will pop-up in which to specify the request; export to csv file; each column may be sorted alphanumerically; may be used as a simple way to find out if every student has been assigned to one and only one group; usually, edit/add/delete a student will not be necessary, but once in a while it may come in handy to be able to manually correct (annotate) an entry instead of having to import the complete list of students from NILE, thereby losing all group info!

Group Name Definition

Class ID …

export …

Group Number	Group ID	Group Info
1	xxxxxxxxxxx	blabla
2	bbbbbbbbbbbb	
3	aaaaaaaaaa	
4	6666666666666	
5	99999999	
6	wwwwwwww	

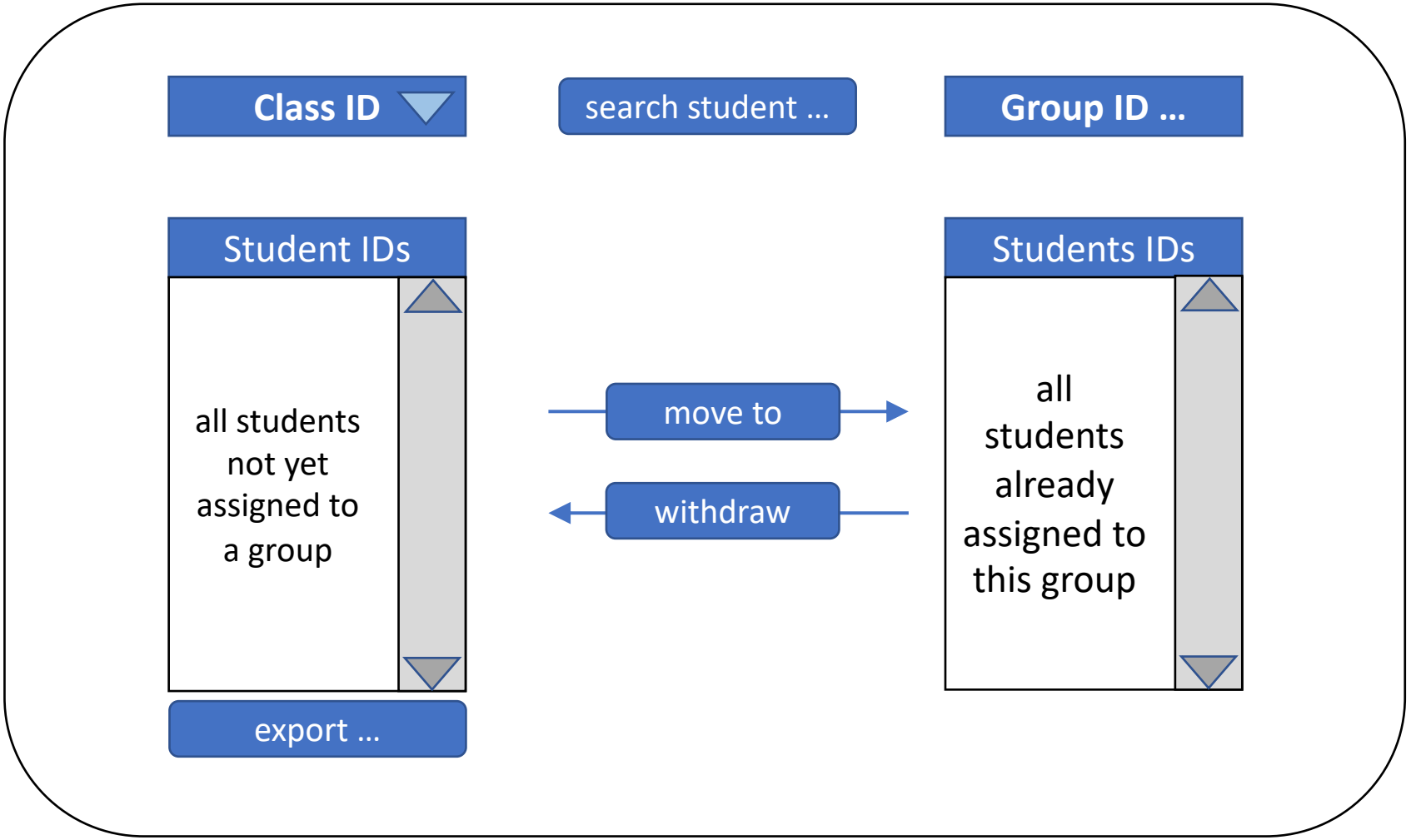
remove …

add …

edit …

Class ID *is a drop-down menu allowing quick change of focus from one class to another (but may interfere with the hierarchical access structure of GPM); probably, it will not make much sense to* ***use a scrolling list*** *for a handful of groups; it will be better to* ***let GPM expand/shrink the table*** *as groups are added or deleted – then, however, the remove/add/edit buttons have to be moved, too; sometimes, exporting group structure (without students) to a CSV file may come in handy.*

Defining Group Membership



Class ID *may be a drop-down menu allowing quick change of focus from one class to another class (but may interfere with the hierarchical access structure of GPM); because classes may be very large, a* ***search*** *button may be necessary (any field allowed); sometimes,* ***exporting*** *to a CSV file may come in handy;* ***GROUP ID*** *will be a clickable button/field so as to allow to go back to the* ***group name definition*** *window*