

Use Case Scheduling

Scheduling by cooperation of Main Lecturers

Actors	Program Manager Main Lecturer GUI
Scope	GUI
No. Iteration	2
Precondition	System is installed and running correctly. Program Manager has been granted this role by Program Administrator. Main Lecturer has been granted this role by Program Manager. Program Manager is logged in. Main Lecturer is logged in.
Postcondition	System is still running, both are still logged in and no side effects on database..
Postcondition on success	A Schedule has been created.
Basic course of events	(1) Program Manager creates a new Schedule and chooses to involve Main Lecturers [cooperative mode]. (2) GUI notifies Main Lecturers that a new Schedule is available. (3) Main Lecturer selects this Schedule. (4) Main Lecturer chooses to create a new Scheduling for his CourseInstances. (5) Main Lecturer predefines parameters. (6) Main Lecturer starts Scheduling. (7) GUI shows result. (8) Main Lecturer makes changes and creates a new conflict. (9) Main Lecturer marks the Scheduling as a proposal. (10) GUI notifies Program Manager about the new proposal. (11) Program Manager views the Schedule. (12) Program Manager resolves conflicts. (13) Program Manager freezes the Schedule. (14) GUI notifies staff about the freeze.
Alternative paths	(5a) Main Lecturer does not predefine parameters. (8b) Main Lecturer makes changes but does not create a conflict. (12b) Program Manager does not need to resolve conflicts. (8c) Main Lecturer does not make changes (12c) Program Manager does not need to resolve conflicts.
Open Questions	-
Implementation notes	-

Scheduling by Program Manager

Actors	Program Manager GUI
Scope	GUI
No. Iteration	2
Precondition	System is installed and running correctly. Program Manager has been granted this role by Program Administrator. Main Lecturer has been granted this role by Program Manager. Program Manager is logged in. Main Lecturer is logged in.
Postcondition	System is still running, both are still logged in and no side effects on database..
Postcondition on success	A Schedule has been created.
Basic course of events	(1) Program Manager creates a new Schedule and chooses to do it alone [single mode]. (3) Program Manager chooses to create a new Scheduling for all CourseInstances. (4) Program Manager predefines parameters. (5) Program Manager starts Scheduling. (6) GUI notifies Program Manager that Scheduling is complete. (7) Program Manager views result. (8) Program Manager makes changes. (9) Program Manager freezes the Schedule. (10) GUI notifies staff about the freeze.
Alternative paths	(5a) Main Lecturer does not predefine parameters. (8b) Main Lecturer makes changes but creates a conflict. Now Program Manager has to resolve that conflict before Schedule can be frozen. (8c) Main Lecturer does not make changes
Open Questions	Q: How is the difference between the two modes decided. Would a mixture of both (i.e. being able to switch modes at any time) be useful and feasible?
Implementation notes	-