

## Technische Anforderungen

1. Strict application of the Database System Development Lifecycle to ensure efficiency of development and to avoid data redundancy
2. Strict separation of database, database application and GUI
3. All data, constraints and business rules are stored in the database (application).
4. The GUI provides means for data access and input only.
5. The GUI, to be designed with MS Access, serves the only purpose to insert, update, access and delete data.
6. The data model has to be implemented with the modelling tool Innovator.
7. The language of the application (data model, GUI) is German.

## Anforderungen Datenbankaufbau

1. The GUI must not implement any data constraints.
2. The data model, constraints and business logic in
3. the database must be designed to forbid any inconsistent data.
4. Each modelling element (entity, role, constraint) has to have a description defining the meaning of the entity precisely to avoid misunderstandings. Example: Possible definitions for entity „Semester “
  - „Identification of a specific academic half year, e.g. winter semester 2012/13. “
  - „Identification of a generic academic half year and all its modules according to the examination regulations, e.g. 6<sup>th</sup> semester of Softwaretechnik (SWT6). “
  - „Group of students having to take the same lectures as stipulated

## Aufbau Stundenplan

1. The examination regulations define which modules have to be offered. Note that not all lectures are defined in the examination regulations (e.g. Wahlfach X, elective X).
2. Modules can consist out of various elements (lectures, labs, projects, ...) having a certain number of weekly semester hours (SWS).
3. Each module element is taught by professors and/or external lecturers (Lehrbeauftragte).
4. A semester group may be split into various teaching groups (e.g. IT3A, IT3B) and sub-groups (e.g. lab groups 1 and 2 of IT3A).
5. Two groups may share the same lecture (e.g. SWT, SWM share Databases 2) having different names for the groups.

## Anforderungen Dozent

1. A lecturer cannot be professor and external lecturer at the same time. However, an external lecturer can become a professor and a professor can become an external lecturer after reaching his pension age.
2. An external lecturer is associated to a specific department (Fakultät).
3. A module element has a certain number of SWS assigned in the examination regulations. The number of hours in the time schedule and the number of
4. SWS as work-load of the lecturer might be different to those, e.g. DB2 Project: Student SWS = 2, Lecturer SWS = 1, hours in time schedule = 0.
5. Professors have to work a certain amount of SWS each semester. The work can be teaching or assigned tasks (dean, running a lab room, research, ...).
6. Currently each full-time professor has to work 18 SWS on average (required work load = Deputat). The assigned number of SWS per semester might be higher or lower but not below 9 SWS.

7. Professors might work part-time, be ill for a longer time or be on sabbatical, i.e. the deputat is reduced.
8. External lecturers do not have a required work load and cannot take other assigned tasks.
9. The application has to document the work load of each professor over the years.

### **Anforderungen Report**

1. List of each task of each professor for a selected semester also giving the total work load balance (Stundenkontostand – accumulated real work load versus required work load (Deputat))
2. List of module elements offered in a selected academic half year for a selected degree (Studiengang)
3. List of external lecturers, their SWS for a selected academic half year and their addresses
4. List of services provided, i.e. list of module elements taught by IT professors for a different department (name of module element, name of the lecturer,
5. SWS, department which the service is provided for)
6. List of services used, i.e. list of module elements taught to IT students by a lecturer of another department (name of module element, name of the lecturer, SWS, department which the service is provided by)

### **Anforderungen GUI**

1. It shall be possible to select different views
2. New lectures and lecturers can be added
3. Existing lectures can be edited throughout the graphical user interface
4. Optional: multiple output possibilities can be selected (textview, file, print)
5. Optional list of classrooms and room occupancy