|  |  |  |
| --- | --- | --- |
|  | **What to deliver**  (to be filled out before evaluation starts) | **Needs to be done** |
| Documentation |  |  |
| Team, Roles, Responsibilities | Project Leader and Quality Manager: Felix Hoffmann  Design Manager: Pablo Rodriguez  Test Manager: Pablo Rodriguez  Development Manager: Tim Jauch | Test Manager: Together |
| Timeline | * M1: Kick off * M2: System Analysis * M3: System design * M4: Testing * M5: System acceptance |  |
| Mission Statement | Old: "Development of a database-driven workload planning system for universities to improve the management of lecture schedules and workload distribution.”  New:  “The purpose of **PlanningToolDB** database system is to maintain the data that is used and generated to support the planning for the faculty office.” | Remove the schedule!!! Only a list, nothing is missing. Calculating the workload of a professor. Maximum and minimum of work and transfer to the next semester |
| Mission Objectives | Old:  Specific goals: "Create a functional data model that correctly calculates workload capacities, generates reports, and prevents inconsistent data."   * Create Entity-Relationship Model * Import Data into Database * Create view for application   New:  New Mission Objectives   * To maintain data on staff * To perform searches on staff * Course filtering * Study program * To track the workload of staff * Course offerings for the upcoming semester * Workload of teaching personal * Service | Not an objective. E-book page: 392 |

Project Diary

Elect group members to their roll:

* Project Leader: Felix Hoffmann
  + Organization and management of the project
  + Makes sure deadlines are met
  + Communicates with professor
* Quality Manager: Felix Hoffmann
  + Makes sure all documents, code and processes meet the quality standards
  + Controls actual state with target state
  + Code reviews and quality checks
* Design Manager: Pablo Rodriguez
  + Creates and manages data models
  + Strong communication between development manager to make sure the reality reflects the models
  + Defines logic of the database
* Test Manager: All
* Development Manager: Tim Jauch
  + Technical implementation
  + Creates and optimizes SQL-Scripts for database creation
  + Develops functions for the system
  + Strong communication between development manager to make sure the reality reflects the models
  + Makes sure system works efficiently

Project questions:

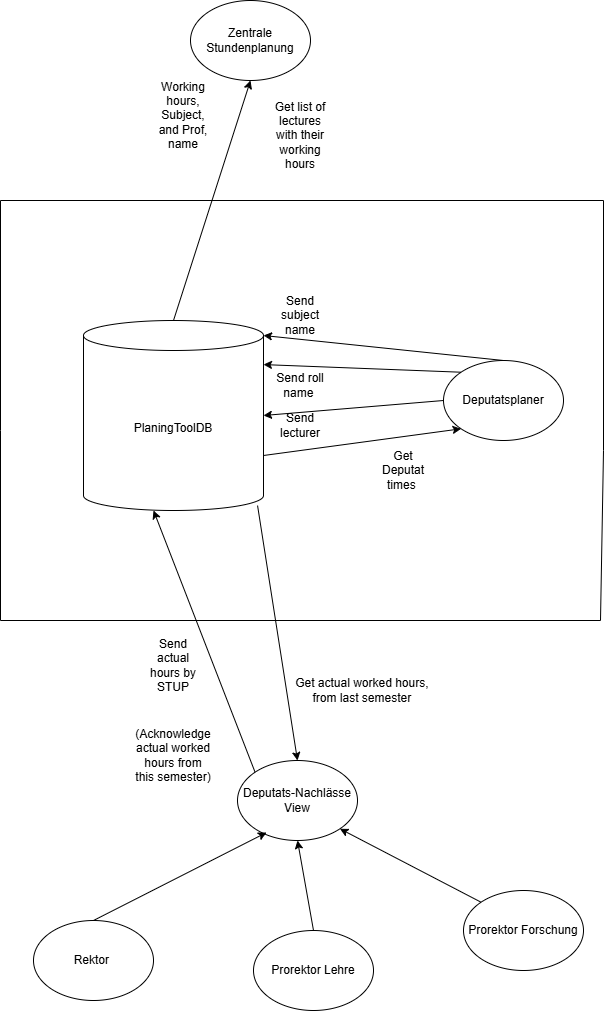
* Which technologies to use? When do we get access?
  + VM, ready to use, IBM installed, Case Tool installed, DB2 Database
* How should we view the document “PlanningTool-EN”/”Project for lecture Database”? Is it just a “here is an example”?
  + Milestone numbering is outdated (Now the first 3 Milestones are Milestone number 3), No graphical userface, no diary for each member (one for the group)
* Are the project diaries and MS-Forms the only things to hand in Moodle Milestones? Where do we hand in the project diaries?
  + Diary not needed to be handed in
* How and when will we transfer the data in the database? (from where too?)
  + Next week will be the beginning. Data will be given later
* Is a frontend application wanted?
  + No
* Can we have a bundled project diary where each member defines their doing?
  + Yes
* Which are the mandatory features that the database needs?
  + View Screenshots
* Is the basic task to fill the tables inside the database in the VM with the given data and create relationships so certain features can be performed?
  + Yes

General Answers

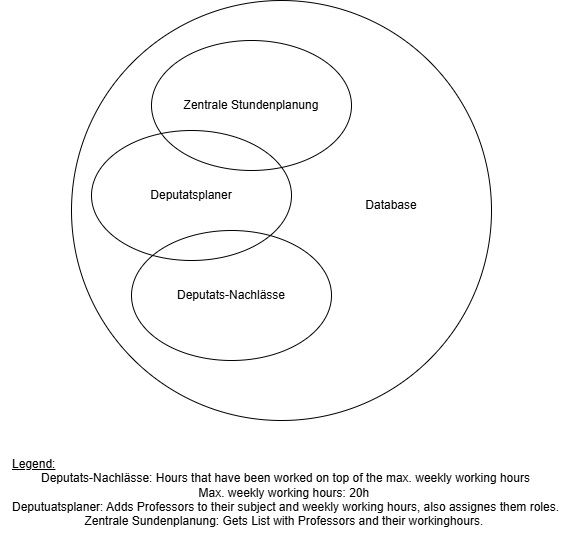
* Screenshots: Data we need to provide. Provide a system which can hold this information of the screenshots. Copying this system, replacing the tool in the screenshots. Main purpose: Tool for who will teach what in the next semester. (Not a system for scheduling)
* A list of courses to be offered
* Data: We will get later
* We will be making the Timetables (Stundenplan planen)

|  |  |  |
| --- | --- | --- |
|  | **What to deliver**  (to be filled out before evaluation starts) | **Needs to be done** |
| Documentation |  |  |
| System Boundary | e.g. context diagram:   * Shows system and interactions with external users (Planner, Admin) |  |
| User Views | Fully specified, based on user rolls:  The system is designed for 2 user roles, the workload planner and the Database Admin.  Deputatsnachlass:   * Provides the information system with working hours of the past semester   Deputatsplaner:   * Assigns professors to lectures, tutorials and labs * Manages teaching load reductions * Generates reports for the rectorate * Maintains time account balances of professors * Semester- and degree-specific lecture lists   Zentrale Studenplanung:   * Receives the course offerings from the information system |  |
| Requirements | System Requirements for **PlanningToolDB** Database System:   1. There are approximately 50 members of staff working for over 75 modules. 2. There are approximately 4 study programs available. 3. There are approximately 7 semesters with an average of 5 modules and a maximum of 7 modules per semester. 4. Having a complete data set for at least three consecutive terms. 5. Differentiate between professors and adjunct professors. 6. Having couplings of two lectures and of three lectures. 7. Having labs with a workload greater than lecture hours from curriculum. 8. Having professors with teaching workload only and professors with additional workload. 9. Having for each professor its workload per term. 10. Having for each professor its total workload balance. 11. Having all imported service for a given department. 12. Having all exported service for a given department. |  |

Context diagram



User view diagram



Questions:

* Is the base data for modeling the ERM? -> Just Information, how it looks like (Data will be given in a different way
* Are the user views enough? -> NO
* What are attribute domains? -> Possible values of the columns (Like enumerations)

Answers:

* No view needed for the Database Admin (Data is already there)
* Planer has 2 roles:
* The list will be given to the Zentrale Stundenplanung who then makes the schedule
* From System dataflow to the Zentrale Stundenplanung
* Planer is inside the tool and the other users are outside of the tool (Planer is part of the system)
* Another user view tracks weekly hours (Should/Is) so the knowledge can be applied to the planning of the next semester
* More requirements in to be checked in the next milestones
* Figure 10.11 Major user views for the DreamHome database system. Do exactly this table
* User view: Which data needs a certain user to see/access/update?

|  |  |  |
| --- | --- | --- |
|  | **What to deliver**  (to be filled out before evaluation starts) | **Needs to be done** |
| Documentation |  |  |
| Model documentation from Tool | Specification of all diagrams, information objects, attributes, relationships |  |
| System Design |  |  |
| Local data model for each user view | In the planning tool |  |
| Global Data model | In the planning tool |  |
| Attribute domains | Possible values – in the planning tool |  |
| Key Attributes | (Maybe) in the planning tool  Course:  -fnr  Lecturer:  -ppruefer  Subject:  -pnr  -group  -ppruefer  -fnr  -semesterId  Semeter:  -uId |  |
| Enhanced Modeling Concepts | What enhanced modeling concepts do we have  -ER Model ffrom MIlestone 2  -Key’s and Relationships in and inbetween tables  -Normalized tables to 3rd |  |
| Normalization | At least 3rd NF |  |
| User Transactions | Functionality of the database:  e.g.   * Workload   **Transaction 1: Assign professor to a course**  **Action**: The planner assigns a professor to a specific course for a given semester  **Entities involved**: Professor, Course, Semester, Assignment  **Database operation**: INSERT INTO Assignment (professor\_id, course\_id, semester\_id, hours\_per\_week)  **Purpose**: Tracks who teaches which course and how much teaching load they receive  **Transaction 2: Register Deputatsnachlässe**  **Action**: The planner registers Deputatsnachlässe for a professor due to extra responsibilities.  **Entities involved**: Professor, Deputatsnachlässe  **Database operation**: INSERT INTO Deputatsnachlässe (professor\_id, semester\_id, reason, hours)  **Purpose**: Ensures correct teaching obligation is calculated after applying Deputatsnachlässe (workload reduction), due to other responsibilities.  **Transaction 3: Generate Workload Report**  **Action**: The planner generates a report listing all lectures, hours and total workload for a professor.  **Entities involved**: Professor, Assignment, Deputatsnachlässe  **Database operation**: SELECT … FROM Assignments JOIN Deputatsnachlässe WHERE professor\_id = ?  **Purpose**: Reporting to rectorate or for planning the next semester.  **Transaction 4: View semester lecture plan**  **Action**: The planner views a list of all courses offered in a specific semester, per study program.  **Entities involved**: Course, Program, Assigment  **Database operation**: SELECT \* FROM Course JOIN Program WHERE semester\_id = ?  **Purpose**: Verify that enough lectures are planned for each semester.  **Transaction 5: Update Teaching Hours**  **Action**: The planner updates a professor’s teaching hours because of a course split.  **Entities involved**: Assignment  **Database operation**: UPDATE Assignments SET hours\_per\_week = 4 WHERE assignment\_id = 1001  **Purpose**: Ensure accurate calculation of actual workload. |  |
| Integrity Constraints | 1. Value can not be NULL:   sbjNo, sbjLevel, studyPrg, sbjName, elective, numCurr, numSchd, srvProvider, srvClient  lecNo, lecName, isProf, lecDept  offeringId, sbjNo, lecNo, term, cntLec, cntCurr, cntSchd  term  reductionId, term, lecNo, jobTitle, reduction   1. Value must be INTEGER:   sbjLevel, numCurr, numSchd  lecNo  offeringId, lecNo, cntCurr, cntSchd  reductionId, lecNo, reduction  Business rules (NULL values...) |  |

Questions:

* Calculating hours? If it’s possible to use the hours of the course from STUPO to calculate the workload for a professor or if those working hours must be written by the Planer when he/she assigns the course to the teaching personal?

Notes:

* In total you need 18 hours
* Deputatsnachlässe has nothing to do with the course (has nothing to do with previous semesters). It is manually inserted. Being head of a lab or similar will gain him a reduction (DN)

Needs to be done:

* Description of the user views
* Need functions in a table (Deputatsnachlässe)
* A function is a head of a Lab (for example Dean (Dekan))
* No separation of the Groups (Like Group A and B)
* Create a workload account (Can be done in SQL. Calculated by a view)

|  |  |  |
| --- | --- | --- |
|  | **What to deliver**  (to be filled out before evaluation starts) | **Needs to be done** |
| Documentation |  |  |
| Batch file to set up database | Logical model, Physical model, create Database schema and staging table, SQL inserts => fully populated tables |  |
| List of all Business Rules | * All Attributes are not nullable (except sbjNotes, lec1stn, lecRoom, lecNotes, supervisor, assNotes) * Term in workLoadReduction and courseOffering, has to be a valid one * A course can only be offered with a valid lecturer * A lecturer can only teach an existing subject * Only an existing lecturer can have a reduction (workLoadReduction) * Uniqueness of sbjNo, lecNo, offeringId, reductionId |  |
| How to ensure Business Rules  (Point out how your model ensure each business rule) | * Make the attributes NOT NULL * Term has to be in the term attribute of semester, else it is not a valid term in courseOffering and workLoadReduction * Foreign key of lecNo in courseOffering * Foreign key of sbjNo and lecNo in lecturer * Foreign key of lecNo in workLoadReduction * Enforced through primary keys |  |
| To be checked | |  |
| Having a complete data set for at least three consecutive terms. | |  |
| Differentiate between professors and adjunct professors. | |  |
| Having couplings of two lectures and of three lectures. | |  |
| Having labs with a workload greater than lecture hours from curriculum. | |  |
| Having professors with teaching workload only and professors with additional workload. | |  |
| Having for each professor its workload per term. | |  |
| Having for each professor its total workload balance. | |  |
| Having all imported service for a given department. | |  |
| Having all exported service for a given department. | |  |

Create your model inside the database and load the data into the database

Then it comes to the views (for workload table )

|  |  |  |
| --- | --- | --- |
|  | **What to deliver** (to be filled out before evaluation starts) | **Needs to be done** |
| Documentation |  |  |
| System Documentation coming from SE-Tool | PTDB DBSchema Documentation.pdf |  |
| Installation Manual | <https://github.com/Skalt47/InfosysLab>  README.md |  |
| Test manual and test report |  |  |
| Installation Files (Logical Model, Physical Model, Database, Test Routines) | Can also be found at: <https://github.com/Skalt47/InfosysLab>  all packed in one single zip-file named Team<nn>{ss|ws}<yy>.zip |  |
| Base data management |  |  |
| Add another elective |  |  |
| Hire another adjunct lecturer |  |  |
| An adjunct is no longer available. |  |  |
| A professor retires |  |  |
| A new professor is hired |  |  |

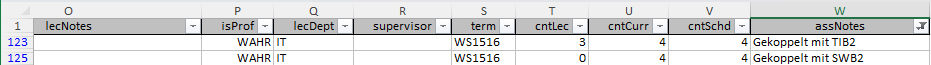
|  |  |  |
| --- | --- | --- |
|  | **What to deliver** (to be filled out before evaluation starts) | **Needs to be done** |
| Planning Offered Courses |  |  |
| Start a planning session for the upcoming term |  |  |
| Planning of a dedicated semester, e.g. SWB6 |  |  |
| Alert for missing courses | OUTER JOIN |  |
| Show workload of a professor |  |  |
| Check linked courses for not adding twice its workload |  |  |
| Reporting |  |  |
| Create a report of all offered courses |  |  |
| Budgeting |  |  |
| Update offered courses at the end of a term |  |  |
| Show the overall workload balance of a professor |  |  |

Notes of Milestone 5 Meeting:

|  |  |
| --- | --- |
| To do | Status |
| Get rid of offeringid (Use composite key instead for courseoffering) | Done |
| Credited hours=cntlec (Currently Datatype error, In created.sh cntlec had the datatype integer, now is decimal) | Done |
| Workloadbalance lecno=16, term=SS15 teaching hours: 7.6 would be correct Reduction hours: 11 would be correct | Done |
| When adding hours it seems to be adding it 2 times | Done |
| Look at the course offering and manage manually, when a room is booked 2 times, but the lecture is of course only done once, so it should only count once | Done |
| + 2 SWS Mathe-Zusatzübungen  Bedeutet das cntLec muss um 2\*(Anzahl Wochen des Semesters) erhöht werden? | Not done, because exercises, are not held lectures. |
| Update Readme.md | Done |

Not possible:

SWB2 & TIB2 WS1516 Softwaretechnik (Coupling will have no effect (TIB2, cntLec=0)):

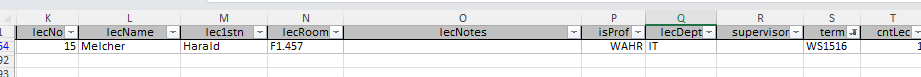


SWB3 WS1516 Rechnernetze (Coupling is not possible, because the courses are taught by 2 different lecturers and in a different room):

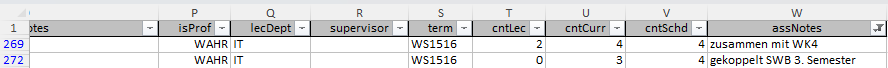


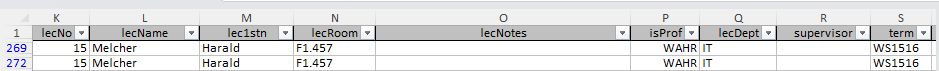


WKB3 WS1516:

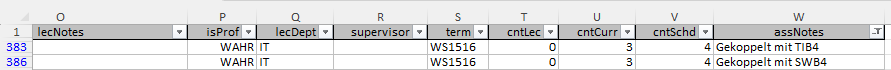


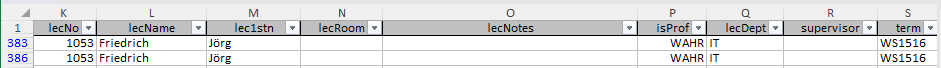
SWB3 & WKB4 WS1516 Internet-Technologien (Coupling will have no effect (WKB4, cntLec=0)):





SWB4 & TIB4 WS1516 Softwarearchitektur (Coupling will have no effect (Both, cntLec=0)):





SWB5, TIB5, WKB5 WS1516 Ingenieursmethodiken (Coupling not possible, because they are all different lecturers):

Ein Bild, das Text, Zahl, Reihe, Schrift enthält.

KI-generierte Inhalte können fehlerhaft sein.

Ein Bild, das Text, Zahl, Reihe, Schrift enthält.

KI-generierte Inhalte können fehlerhaft sein.

Possible:

TIB3 WS1516 Rechnernetze:





SWB3 WS 1516 Rechnernetze:





WKB3 WS 1516 Rechnernetze (Coupling not possible, different lecturer and room):  
