Distributed Operating Systems – project

The task:

Design and implement a distributed database system with the following properties:

- The database should implement data replication.
- The database should use a subset of SQL defined by the team. Minimal requirements include table creation and dropping, insertion and deletion of records and SELECT queries with at least a minimal table joining capability.
- The database must be robust. Temporary nonavailability of any node must not cause any loss of functionality.
- The database must provide an increase in performance and/or robustness as a result of distribution.
- A way of displaying the current contents of each node (locally) must be present for testing purposes.
- Presentation should take place in the laboratory.

Additional information:

Teams may have 3, 4 or 5 members. Each team is represented by a selected person, the project leader. During the first meeting the following roles must be assigned to team members:

- Project manager, responsible for task assignment, project management, final design decisions and contact with the teacher. In 4-person teams the project leader may only do small programming tasks[1] – the implementation is mostly a role for the rest of the team. In 5-person teams the project leader may not program.
- Documentation – compiling and formatting of the documentation written by all members.
- Testing – preparing the test cases, automatic testing scripts and test documentation (in phase III, subject to reformatting by the Documentation role holder).
- Presentation – the final presentation of the project, requires good command of the English language and some showmanship. Includes automation of parts of the live presentation and preparation of data for this task.
- Repository manager – creating the repository system, regular backups.
- Database engineer – team's expert on the database engine used on each node or main designer of that engine if it is created from scratch.

Every person must have at least one of these roles. Every person (except the project leader in 5-person teams) also holds the role of a programmer.

The testing and presentation should be highly automatic, not heavily interactive. Each phase of the project must be well documented. Documentation must include project management documents – minutes from all team meetings, project decisions made by the team, the tasks assigned to each member and the status of previously assigned tasks. The project manager should include in the documentation of each phase an evaluation of the work of all team members – this will influence the individual notes.

Grading:

The project is worth 50 points, assigned as follows:

- After each phase the teacher will give the team (as a whole) points: 0-7 after phase 1, 0-10 after phase 2 and 0-13 after phase 3. The sum of these points (max. 30) is the average team note. The project managerr is given exactly the average. The points for other team members are assigned by the project manager, so that no person is given less than 0 points and that the average value is as given by the teacher or lower. The maximal note for each person should not normally exceed 30 points, but in special circumstances this may be negotiated by the project manager with the teacher (e.g. if one person was very clearly the most hard working team member but a high average of the team doesn't allow recognizing it properly without exceeding 30 points).
- The other 20 points are given independently to each student by the teacher, depending on his work on the project (pay attention to documenting this!) and on the quality of the aspect of the project corresponding to his role.
- Achieving more than 50 points is not possible.

The basis for awarding points is the documentation provided in each phase. Delays will result in lowering the team's score.

Team changes during the project

Reassignment of roles within the team is allowed, but requires some paperwork – the project manager must obtain permission of the teacher, document the old and new assignment and document the state of the given roles at the moment of reassignment (for grading purposes).

Project manager may be overturned by unanimous decision of the other team members. If this is necessary for the good of the project, all the other team members will contact the teacher in person. The teacher will accept or deny the decision after consultation with team members and the manager.

Teams of less than 3 persons are normally not allowed. If such a team appears due to random circumstances (prolonged illness, etc.) transfer from another team must be negotiated. Only if this fails, even after mediation from the teacher, is a smaller team

[1] Interfaces between modules or modules integrating other modules are most natural tasks for a leader.

possible.

Voluntary transfers between teams are allowed at milestones if managers of interested teams agree. The transfer must be reported to the teacher and may not reduce the number of persons in any team below 3. Fully documented reassignment of roles in interested teams is necessary.

Voluntary transfers between milestones are in general not allowed. Exceptions can be granted by the teacher, but the request can also be denied without explanation.

Transfers, both voluntary and forced, between milestones require some paperwork – the project leader of the previous team must write a full description and evaluation of the transferred person's work in the current phase. The grading of that person during the next milestones is done by both managers separately and weighted by the time spent in each team during that phase.

Project phases:

Phase 1:

Prepare a functional specification of the project (requirements, architecture, major technical assumptions). Document the roles assigned to each member of the team. Prepare the repository and prepare a short instruction for other team members. Assign the tasks for the second phase.

Points awarded for:

- Documentation.
- Presentation of a working repository.

Phase 2:

Implement the prepared design. Prepare data for testing and a specification of the testing procedures. Prepare demonstration scenarios.

Points awarded for:

- Documentation.
- Presentation of a working prototype (may be buggy and incomplete, but a significant part of the functionality must be present).
- Code review, if deemed necessary by the teacher.

Phase 3:

Finish the technical documentation of the project. Run the tests, debug the code. Document testing results. Prepare the demonstration. Demonstration and documentation are both a basis for grading.

Points awarded for:

- Documentation.
- Presentation of the program.
- Quality of code.
- Testing results.

NOTE FOR ALL PHASES: ALL DOCUMENTATION MUST BE DELIVERED IN PDF FORMAT BY E-MAIL TO AKOZAKIE@ELKA.PW.EDU.PL.

This is the responsibility of the team manager.
A printed copy is welcome but not required **UNLESS** the documentation is not sent at least one day before the grading session – in that case it is absolutely necessary.

Laboratory sessions and milestones:

Date	Session
2013.10.17	Introduction, role assignment.
2013.10.31, 2013.11.21	Normal sessions (voluntary). *It is strongly suggested to discuss a draft version of the specifications with the teacher at one of these sessions to avoid any misunderstandings that might cause loss of points.*
2013.11.28	**End of phase I**
2013.12.12/2013.12.19	Normal sessions (voluntary).
2014.01.09	**End of phase II**
2014.01.16/2014.01.23	Normal sessions (voluntary).
2014.01.30	**End of phase III**