EXAMINATION

The exam is open book, open laptop and open Internet. But you have to provide your individual solutions – the collaboration in any form is not allowed. You can provide your solutions either on paper or electronically. In case of the electronic solutions, please email the file to <rma@ut.ee> no later than 20:05 with a subject “[RE] exam solutions”.

Scenario
(adapted from IREB exercise given at REFSQ 2012)

PowerAB is an industrial group that runs a filling station at its plant in Traktoria. The filling station is used primarily for company vehicles but employees and their family members can also use it (and do so quite often due to the low prices for fuel charged there).

The filling station is open only on workdays during the regular working hours. It is run by one filling station attendant. A fuelling operation takes about six minutes (including the generation of the receipt). The fuelling operations naturally overlap because there are four pumps available to customers (1 for diesel, 1 for super, and two for lead-free). The filling station has nevertheless experienced a lot of congestion in recent months. The peak days are Fridays and Mondays.

When a company vehicle is fuelled, the receipt records the license number, the cost center (department to which vehicle belongs), the type of fuel, quantity and date. If an employee fills his/her private vehicle, the receipt includes the license number and personnel ID number (along with the other data, e.g., type of fuel, quantity, date and time). Family members of employees are entitled to pay cash when filling their vehicles.

At the end of his workday, the filling station attendant must give the company accounting department the daily cash, the current level of the fuel tanks and the receipts. These item are checked, either immediately or the next day (with any differences clarified with the filling station attendant). After this is done, the accountants are settled, i.e., fuel account is credited and the purchasers are debited. For purchases involving company vehicles, the cost centers for the departments are charged. For purchases involving employees, the personnel cost center is charged and the receipts are passed on to the personnel department. There the purchases are recorded as a deduction from the monthly salary account. Cash revenues are passed to the central cashier’s office in financial accounting.

The attendant is responsible for seeing to the refilling of the station’s fuel tanks. As soon as a minimum level is reached (with reserves for five workdays), he orders a tanker delivery from the fuel supplier. This account is settled directly with the company accounting department (debit against the fuel account and payment of the supplier).

PowerAB objective: to put an information system (i.e., software-intensive system) in place for substantially simplifying and streamlining administrative activities.
Main Examination Task

For the given scenario, prepare a requirements specification (document), which includes requirements for the new information system or its component.

When creating the specification you must pay attention to the following concerns: (20 points)

- Your requirements specification should be structured, organised, traced, and etc. and should follow any known requirements specification template (below please indicate which one).
- Requirements specification follows ___________________ template.
- Maintain the term glossary during the whole requirements specification process.
- Define the traceability model including both the traceability artefacts and traceability relationships. Maintain the traceability in the whole requirements specification.
- Unless indicated differently, the goals, domain properties, and (all types of) requirements must be documented using the “requirements shell” or any other structured-text template.
- Explain what parts correspond to what solutions of the examination tasks. You can use the following table:

<table>
<thead>
<tr>
<th>Examination task</th>
<th>Solution given in Section, Table, Figure, etc…</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 1</td>
<td></td>
</tr>
<tr>
<td>Task 2</td>
<td></td>
</tr>
<tr>
<td>Task 3</td>
<td></td>
</tr>
<tr>
<td>Task 4</td>
<td></td>
</tr>
<tr>
<td>Task 5</td>
<td></td>
</tr>
<tr>
<td>Task 6</td>
<td></td>
</tr>
<tr>
<td>Task 7</td>
<td></td>
</tr>
<tr>
<td>Task 8</td>
<td></td>
</tr>
<tr>
<td>Task 9</td>
<td></td>
</tr>
<tr>
<td>Task 10</td>
<td></td>
</tr>
<tr>
<td>Task 11</td>
<td></td>
</tr>
</tbody>
</table>

1. Carefully choose the problem scope!!! Clarify the problem scope along the four facets of the system context (i.e., subject facet, usage facet, IT facet, and development facet). (5 points)
2. What are **major stakeholders** and their goals? What should the software-intensive system do to satisfy the stakeholders’ dependency relationships? Explain your answer with the * strategic dependency model (technical viewpoint).

(10 points)

3. Document at least one goal (using the “requirements shell” or any other structured-text template) from model defined in Task 2.

(5 points)

4. Using **sequence diagram** define the interaction scenario between the software-intensive system and stakeholders. Explain how this scenario satisfies the stakeholder goals.

(10 points)

5. Take the documented goal (see Task 3). Build the goal hierarchy using the KAOS goal modelling language. The goal hierarchy should be at least of 3 levels. It should describe at least 2 alternative design solutions. The lower lever goals (of at least two branches) should be assigned to agents.

(10 points)

6. From the goal model defined in Task 5 derive at least 1 assumption/expectation and 3 requirements and document them using the “requirements shell” or any other structured-text template.

(10 points)

7. Create a class diagram, which would describe data used in the software-intensive system. Derive from the class diagram at least three solution-oriented requirements and document them (using the “requirements shell” or any other structured-text template).

(10 points)

8. Create a state diagrams for one object from the class diagram. Derive from the state diagram(s) at least one solution-oriented requirement and document it (using the “requirements shell” or any other structured-text template).

(10 points)

9. Elicit and document quality (non-functional) requirements:
   - one usability requirement,
   - one security requirement, and
   - one portability requirement.
   How do the quality requirements characterise one (select from your specification) functional concern (defined in terms of the requirements artefacts, like goals, scenarios or solution-oriented requirement) of your software system? What is the traceability between the quality requirements and solution-oriented requirements?

(10 points)

10. Create traceability matrix for the selected traceability artefacts. Your matrix should include at least three types of traceability artefacts and at least two types of traceability relationships. The created traceability matrix should be consistent with the traceability model.

(10 points)

11. What is the quality of your requirements specification? Which quality properties should be improved? Your answer should be supported with the quality evaluation results.

(10 points)