Name:

Studentnr:

Software Architecture Exam

27 January 2015, 13:30 – 16:30

- This exam consists of 6 questions on 19 pages. Please check first whether you have properly obtained **all** pages.
- Enter the answers in the space allocated. In case you need more space, you can **use the back** of the pages. Make a proper reference to such an extra part on the back.
- When you have finished the exam, you should submit the complete package stapled in the correct order.
- Read carefully the question before you start answering!
- Reread your answers to check whether you really answered the question posed!

– Question	Max. points	Awarded points
1	10	
2	15	
3	20	
4	20	
5	20	
6	15	
Total:	100	
Grade:	**Total / 10**	

Good luck with the exam!

1. Basic concepts (10p)

The definition of software architecture of a system used throughout the course is of Bass et al: "the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both".

Consider the following model based on Rozanski & Woods explaining the relations between the different elements within the field of software architecture.
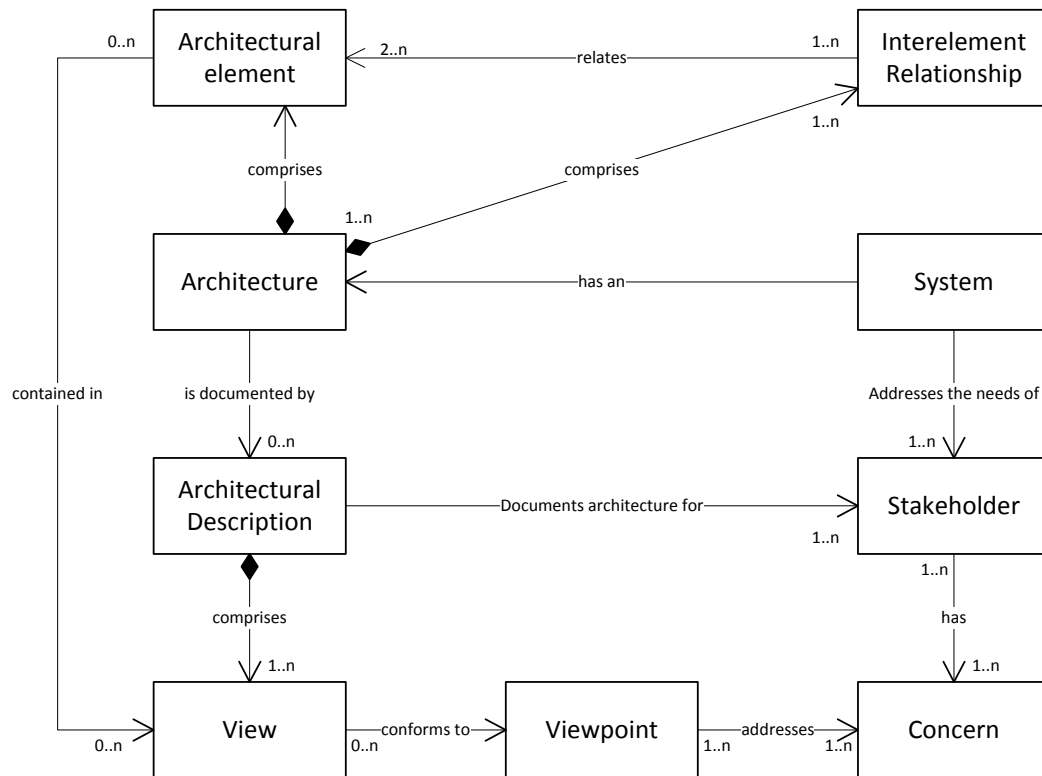
a. Give an example of how an architect can use Viewpoints to address Concerns of Stakeholders (1p)

b. Explain how an architect can use Viewpoints to address Concerns of Stakeholders (2p)

c. Explain using the above model why an architect models Structures but documents Views. (3p)

Another view on architecture is by Taylor et al. in their book "Software Architecture: Foundations, Theory, and Practice": "A software system's architecture is the set of principal design decisions about the system".

d. Explain how this definition complements the definition used throughout the course (Bass et al.) (4p)

2. Software product lines (15p)

Consider the definition of a Product Line: "A product line of software is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way".

a. Explain "sharing a common, managed set of features". What means common? What means managed? (3p)

b. Explain: "common set of core assets in a prescribed way". What are core assets? What means prescribed way? (3p)

c. Give an example of a well-known product line. (1p)

d. Explain why your example in question c. is a product line. (2p)

e. Explain the notion of a Variation Mechanism in product lines. (3p)

f. Explain how plug-ins and add-ons realize a variation mechanism in a product line. (3p)

3. Patterns and quality attributes (20p)

An architectural pattern establishes a relationship between a context, a problem and a solution [1].

a. Explain how an architectural pattern establishes this relationship. (3p)

b. Explain the relation between a viewpoint and an architectural pattern. (3p)

Given is the following summary and evaluation of the Broker pattern

Summary:
Providers register their function to a Broker. When a Client requires some functionality, it sends its request to the broker. The Broker decides which Provider to contact, sends the request, and forwards the response of the Provider to the Client.

When to use it:
Unknown number of providers that offer certain functionality

When to avoid it:
Requests or responses are too large to be routed via a Broker

    c.   Explain the Broker-pattern in terms of context, problem and solution. (3p)

Context:

Problem:

Solution:

Another example of an architectural pattern is the SOA Registry.

Summary:
Providers register their function to a Register. When a Client requires some functionality, the Register returns an appropriate Provider. The Client then contacts the Provider to handle its request.

When to use it:
Requests or responses are too large to be routed via a Broker

When to avoid it:
Providers not always responsive

d. Explain the Registry pattern in terms of context, problem and solution. (3p)

Context:

Problem:

Solution:

A solution for a pattern is determined and described by Bass et al.:

- A set of element types
- A set of interaction mechanisms or connectors
- A topological layout of the components
- A set of semantic constraints covering topology, element behavior and interaction mechanisms.

    e. Describe the role and use of each of these four aspects for the solution of the Register-pattern. (4p)

A set of element types:

A set of interaction mechanisms or connectors:

A topological layout of the components:

A set of semantic constraints covering topology, element behavior and interaction mechanisms:

Within a public-transport organization, the software architect is designing a payment system that will be used in all buses within the region. The region the organization operates in is a rural area with a few villages and a large city. Within the city, the WiFi network coverage is good, however between the villages, there is no WiFi connection, and the GSM network stutters from time to time. The main quality attribute the architect is concerned with is availability.

Based on the current requirements, the architect is considering whether to use the Broker or the Registry pattern.

    f.   Explain **and show** how the architect can use tactics to decide which of the two patterns is best applicable in his situation. (4p)

4. Quality attributes (20p)

a. Functional requirements and quality attributes go hand-in-hand. Explain the relationship between functional requirements and quality attributes. (2p)

b. Explain how quality attributes are addressed in the model presented in **question 1**. (3p)

c. Tactics can be used to achieve required quality attributes. Explain the relationship between tactics and the quality attribute scenario. (3p).

To analyze whether a software architecture satisfies the specified quality attributes, one can:

- Model the architecture to enable Quality Attribute Analysis
- Use Quality Attribute Checklists
- Perform thought experiments and Back-of-the-envelope analysis
- Create prototypes

d. Explain how each of the approaches can be used to assess the quality attribute, and give for each an example. (5p)

Model the architecture to enable Quality Attribute Analysis:

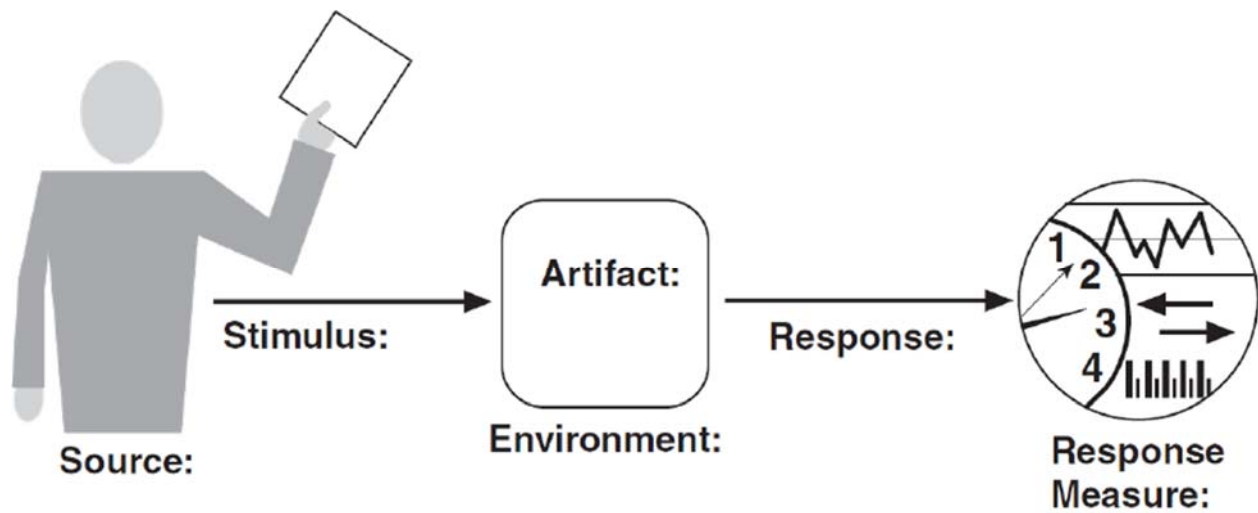Example:

Use Quality Attribute Checklists:

Example:

Perform thought experiments and Back-of-the-envelope analysis:
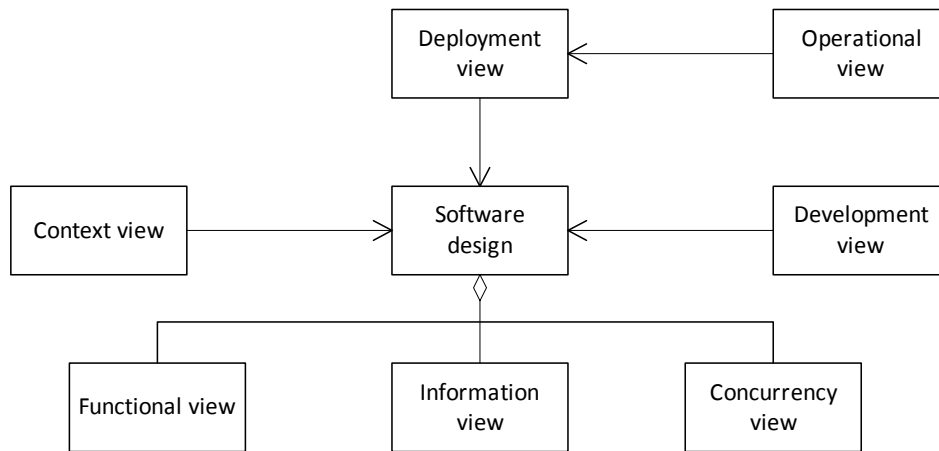
Ëxample:

Create prototypes:

Example:

e. An Architectural Significant Requirement in the public transport system as described in
   question 3 is availability of the Broker. Create a concrete scenario for this quality attribute in the
   generic scenario depicted below (3p).

Source: Stimulus: Artifact: Environment: Response: Response Measure:

f. Explain your scenario. (4p)

5. Functional architecture (20p)

A functional architecture model is an architectural model from a usage perspective.

a. Is the functional architecture model an informal, semi-formal or a formal notation? Explain your answer. (3p)

b. One of the uses of a functional architecture model is roadmapping. Explain how the functional architecture can be used for roadmapping. (3p)

The viewpoints of Rozanski & Woods in "Software Systems Architecture" are catalogued in 7 viewpoints, as shown in the figure below.

c.   Explain how the Functional Architecture can be used in the Context view. (3p)

d.   Explain how the Functional Architecture can be used in the Information view (3p).

The dynamic structure of the software is an important aspect to be taken into account. To model dynamic structures, we use Component-and-Connector structures.

e. What are the elements of a Component-and-Connector structure? Explain them and give an example for each. (2p)

f. Explain how the Component-and-Connector structure is embedded in the functional architecture model. (2p)

We distinguish two types of dynamic structures: horizontal and vertical behavior.

g. Explain horizontal and vertical behavior. (2p)

Horizontal behavior:

Vertical behavior:

An example of a Functional Architecture Diagram within the domain of the Public Transport organization is depicted below. It describes how the "Plan travel" module uses functionality of other modules to derive a list of possible routes and the associated costs
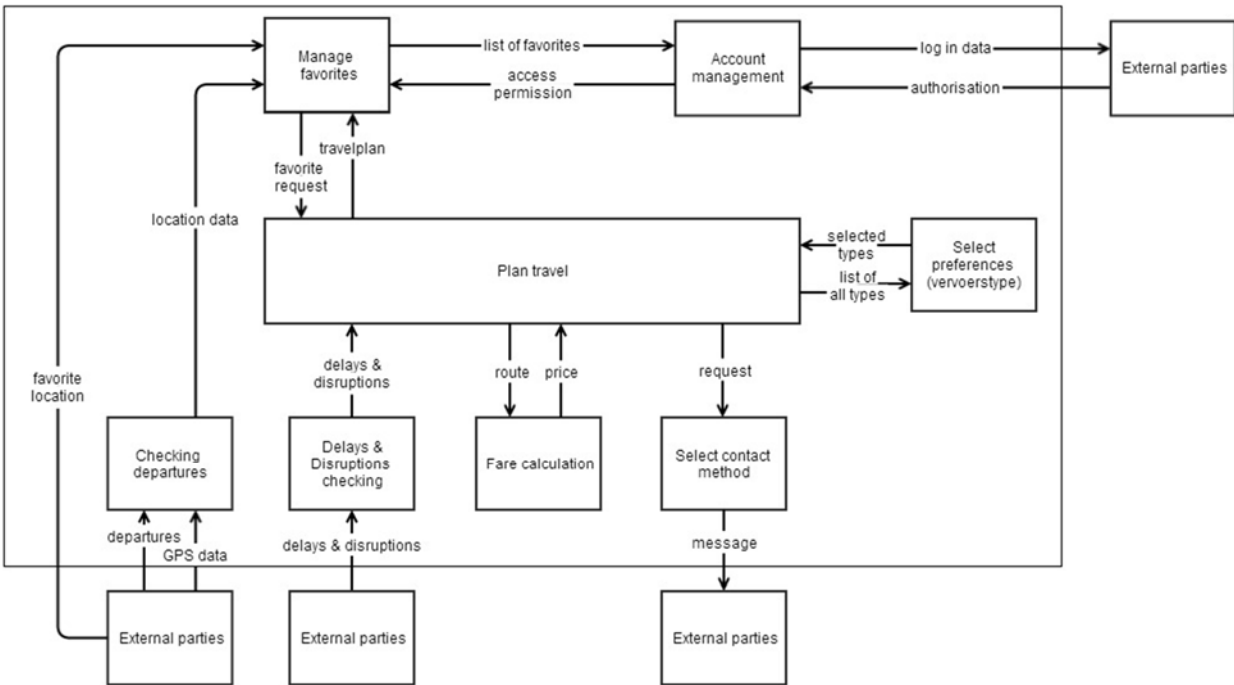
h.   Give an example of horizontal dynamic behavior using the figure above. (1p)

i.   Give an example of vertical behavior for this system. (1p)

6. Architecture Implementation & Evaluation (15p)

a. Architectural erosion is an imminent danger for any software product. In other words, the implementation of an architecture will almost certainly drift away from the intended architecture. Explain why. (3p)

b. Give two examples of how the risk of architectural erosion can be minimized. (3p)

To assess the quality of a software architecture, one can use different tools and techniques.

c. Explain the activities of Architecture Analysis and Architecture Evaluation and what their difference is. (3p)

An important tool to evaluate a software architecture is the Architecture Trade-off Analysis Method (ATAM).

    d. Explain how ATAM helps in improving the quality of a software architecture (3p)

    e. An important aspect of the ATAM are the intangible outputs of the method. What are intangible outputs, and why are these important? (3p)