Mock Exam
iSAQB® Certified Professional for Software Architecture —

Foundation Level (CPSA-F®)

International Software Architecture Qualification Board e. V.

Document version: 2025.2
Based on curriculum - version 2025.1-rev2, 20250427

Examination Rules

This examination is a mock exam, which is based on the certification exam of the Certified Professional for Software Architecture – Foundation Level (CPSA-F®) in form and scope. It serves to illustrate the real iSAQB® CPSA® examination as well as to prepare for the corresponding exam.

The mock exam consists of 38 multiple-choice questions, which can be evaluated with 1 or 2 points depending on the level of difficulty. At least 60 percent must be achieved to pass the exam.

52 points can be achieved in this mock examination, you would need 31.20 points to pass.

The following general rules apply: Correct answers result in plus points, incorrect answers result in a deduction of points, but only with regard to the respective question. If the wrong answer to a question leads to a negative score, this question is evaluated with a total of 0 points.

The multiple-choice questions of the mock exam are divided into three types of questions:

A-Questions (Single Choice, Single Correct Answer) Select the only correct answer to a question from the list of possible answers. There is only one correct answer. You receive the specified score for selecting the correct answer. Depending on the level of difficulty, you can achieve a score of 1 or 2 points.

P-Questions (Pick-from-many, Pick Multiple) Select the number of correct answers given in the text from the list of possible answers to a question. Select just as many answers as are required in the introductory text. You receive 1/n of the total points for each correct answer. For each incorrect cross, 1/n of the points are deducted. The score is 1 or 2 points depending on the level of difficulty.

K-Questions (Allocation Questions, Choose Category) For a question, select the correct of the two options for each answer choice (”correct” or ”incorrect” or ”applicable” or ”not applicable”). You will receive 1/n of the points for each correctly placed cross. Incorrectly placed crosses result in the deduction of 1/n of the points. If NO answer is selected in a line, there are neither points nor deductions. The score is 1 or 2 points depending on the level of difficulty.

For a more detailed explanation of the question types and scoring system, further information is available in the CPSA-F examination guide.

The processing time is 75 minutes for native speakers and 90 minutes for non-native speakers. In order to ensure that the preparation for the exam is as authentic as possible, the processing time should be adhered to and any aids (such as seminar materials, books, internet, etc.) should not be used.

The exam can subsequently be evaluated using the solution for this mock exam.

Given that the iSAQB® e.V. is indicated as source and copyright holder, the present mock exam may be used in the context of training courses, for exam preparation or it may be passed on free of charge.

However, it is explicitly prohibited to use these exam questions in a real examination.

1 — *Id: Q-20-04-01*

How many definitions of “software architecture” exist? [1 Point]

A-Question: Choose one answer.

☐ a) Exactly one for all kinds of systems.

☐ b) One for every kind of software system (e.g. “embedded”, “real-time”, “decision support”, “web”, “batch”, . . .).

☐ c) A dozen ore more different definitions.

2 — *Id: Q-20-04-02*

Which of the following aspects are covered by the term “software architecture”? [1 Point]

*P-Question: From the following five answers select **three** that fit best.*

☐ a) Components

☐ b) Cross-cutting concepts

☐ c) (internal and external) Interfaces

☐ d) Coding conventions

☐ e) Hardware sizing

3 — *Id: Q-17-13-01*

Which FOUR of the following statements about (crosscutting) concepts are most appropriate? [2 Points]

*P-Question: From the following seven answers select **four** that fit best.*

☐ a) Uniform usage of concepts reduces coupling between building blocks.

☐ b) The definition of appropriate concepts ensures the pattern compliance of the architecture.

☐ c) Uniform exception handling can be achieved when architects agree with developers upon a suitable concept prior to implementation.

☐ d) For each quality goal there should be an explicitly documented concept.

☐ e) Concepts are a means to increase consistency.

- ☐ f) A concept can define constraints for the implementation of many building blocks.
- ☐ g) A concept might be implemented by a single building block.

4 — *Id: Q-17-13-02*

In your project, three architects and seven developers are working on the documentation of the software architecture. Which methods are appropriate in order to achieve a consistent and adequate documentation, and which are not? [1 Point]

K-Question: Assign all the answers.

appropriate	not appropriate		
☐	☐	a)	The lead architect coordinates the creation of the documentation.
☐	☐	b)	Identical templates are used for the documentation.
☐	☐	c)	All parts of the documentation are automatically extracted from the source code.

5 — *Id: Q-17-13-03*

Which of the following techniques are best suited to illustrate the workflow or behavior of the system at runtime? [2 Points]

*P-Question: From the following eight answers select **four** that fit best.*

- ☐ a) Flowcharts
- ☐ b) Activity Diagrams
- ☐ c) Depiction of screen flows (sequence of user interactions)
- ☐ d) Sequence diagram
- ☐ e) Linear Venn diagram
- ☐ f) Numbered list of sequential steps
- ☐ g) Tabular description of interfaces
- ☐ h) Class diagrams

6 — *Id: Q-17-13-04*

Which of the following principles apply to testing? [1 Point]

*P-Question: From the following five answers select **three** that fit best.*

- ☐ a) In general, it is not possible to discover all errors in the system.
- ☐ b) In components with many known previous errors, the chances for additional errors are high.
- ☐ c) Sufficient testing can show that a program is free of errors.
- ☐ d) Testing shows the existence of errors rather than the absence of errors.
- ☐ e) Functional programming does not allow automated testing.

7 — *Id: Q-17-13-05*

Which of the following statements regarding the information hiding principle are true and which are false? [2 Points]

K-Question: Assign all the answers.

true	false		
☐	☐	a)	Adhering to the information hiding principle increases flexibility for modifications.
☐	☐	b)	Information hiding involves deliberately hiding information from callers or consumers of the building block.
☐	☐	c)	Information hiding makes it harder to work bottom-up.
☐	☐	d)	Information hiding is a derivative of the approach of incremental refinement along the control flow.

8 — *Id: Q-20-04-03*

What are the most important goals of software architecture? [1 Point]

*P-Question: From the following four answers select **two** that fit best.*

- ☐ a) Improve accuracy of patterns in structure and implementation.
- ☐ b) Achieve quality requirements in a comprehensible way.
- ☐ c) Enable cost-effective integration and acceptance tests of the system.
- ☐ d) Enable a basic understanding of structures and concepts for the development team and other stakeholders.

9 — *Id: Q-20-04-12*

Put yourself in the position of a software architect for a large, distributed business application in the banking or insurance domain. Which of the following statements is true and which is false? [1 Point]

K-Question: Assign all the answers.

true	false		
□	□	a)	The architect collaborates with the stakeholders to determine where the requirements and constraints will change often (e.g., business processes, technologies), and designs the architecture such that changes can occur without requiring extensive restructuring of the software architecture.
□	□	b)	Required product qualities should drive your architectural decisions.
□	□	c)	The software architecture can be designed completely independent of the hardware and infrastructure.

10 — *Id: Q-20-04-03*

What are your most important responsibilities as a software architect with respect to requirements? [1 Point]

P-Question: From the following five answers select ***three*** *that fit best.*

□ a) Support the business people to specify explicit and concrete quality requirements.

□ b) Help to identify new business opportunities based on your technology know-how.

□ c) Reject business requirements that contain technical risks.

□ d) Capture all business requirements in a terminology that can be understood by your development team.

□ e) Check requirements for technological feasibility.

11 — *Id: Q-20-04-07*

You are responsible as an architect for keeping a legacy system up and running according to the ongoing requirements of your business. What are the most important action items on your agenda? [1 Point]

P-Question: From the following five answers select ***three*** *that fit best.*

□ a) Negotiating the maintenance budget for your team

□ b) Assuring up-to-date documentation of the deployed system

□ c) Analyzing the impact of new requirements on the current system

□ d) Encouraging the team members to learn new programming languages

□ e) Suggesting technology updates in addition to the business requirements to your management

12 — *Id: Q-21-05-01*

Which of the following statements regarding architecture decisions are true, which are false? [2 Points]

K-Question: Assign all the answers.

true	false		
□	□	a)	Architecture decisions never need to be written down because they are already known to the development team.
□	□	b)	An architecture decision record helps to make the decision's context understood.
□	□	c)	Once a decision has been made on a central or fundamental framework (e.g. persistence framework), that decision must not be changed.
□	□	d)	Quality requirements help significantly with architecture decisions.

13 — *Id: Q-20-04-09*

Decide for each of the following statements whether it is true or false. [2 Points]

K-Question: Assign all the answers.

true	false		
□	□	a)	Each iteration of an agile development approach could have an impact on the fundamental architecture decisions.
□	□	b)	The total effort spent on architectural work is much higher in iterative projects compared to waterfall projects.
□	□	c)	Agile projects do not need architecture documents since the development team uses daily standup-meetings to communicate decisions.
□	□	d)	If your systems consist of a set of microservices there is no need for a central architecture document since each service is free to choose its technologies.

14 — *Id: Q-20-04-10*

Which of the following statements regarding project goals and architectural goals is true and which is false. [2 Points]

K-Question: Assign all the answers.

true	false		
☐	☐	a)	Project Goals can include functional requirements as well as quality requirements.
☐	☐	b)	Architectural goals are derived from the quality requirements for the system or product.
☐	☐	c)	Business stakeholders should concentrate on business goals and not interfere with architectural goals.
☐	☐	d)	To avoid conflicts, business goals and architectural goals should be non-overlapping sets.

15 — *Id: Q-20-04-11*

What does the rule “explicit, not implicit” mean for architecture work? Choose the best-fitting answers. [1 Point]

*P-Question: From the following five answers select **two** that fit best.*

☐ a) Architects should avoid recursive structures and replace them by explicit loops.

☐ b) Architects should make the assumptions leading to decisions explicit.

☐ c) Architects should explicitly insist on natural language explanations (i.e. comments) for each building block.

☐ d) Architects should explicitly insist on written or at least verbal justifications for development effort estimates from their team.

☐ e) Architects should make prerequisites for their decisions explicit.

16 — *Id: Q-20-04-32*

There are many approaches that lead to a software architecture. Which of the following are the most often found in practice? [1 Point]

*P-Question: From the following five answers select **three** that fit best.*

☐ a) User interface driven design

☐ b) Domain driven design

☐ c) View based architecture development

☐ d) Bottom-up design

☐ e) Majority voting

17 — *Id: Q-20-04-38*

Several architecture development methods suggest a view-based approach. Which of the following views are most often used? [1 Point]

*P-Question: From the following six answers select **three** that fit best.*

☐ a) Physical database view

☐ b) Context view

☐ c) Building Block/Component view

☐ d) Test-driven view

☐ e) Configuration view

☐ f) Runtime view

18 — *Id: Q-20-04-22*

When documenting a building block of your software architecture, what information should the black-box description contain? [1 Point]

*P-Question: From the following four answers select **two** that fit best.*

☐ a) Public interfaces.

☐ b) Responsibility of the building block.

☐ c) Internal structure of the building block.

☐ d) Specification of the implementation details.

19 — *Id: Q-20-04-17*

Which prerequisites have to be fulfilled before developing a software architecture? Pick the most appropriate answers. [1 Point]

*P-Question: From the following five answers select **two** that fit best.*

☐ a) The requirements specification for the system is complete, detailed and consistent.

☐ b) The most important qualities for the system are known.

☐ c) Organizational constraints are known.

☐ d) The programming language has been selected.

- ☐ e) Hardware for the development team is available.

20 — *Id: Q-20-04-18*

Which factors can influence the design of a software architecture? Pick the most appropriate answers. [1 Point]

P-Question: From the following four answers select ***three*** *that fit best.*

- ☐ a) Political.
- ☐ b) Organizational.
- ☐ c) Technical.
- ☐ d) Virtual.

21 — *Id: Q-20-04-28*

Which of the following qualities can most likely be improved by using a layered architecture? [1 Point]

A-Question: Choose one answer.

- ☐ a) Runtime efficiency (performance).
- ☐ b) Flexibility in modifying or changing the system.
- ☐ c) Flexibility at runtime (configurability).
- ☐ d) Non-repudiability.

22 — *Id: Q-20-04-33*

Which type of problems provide a good fit for the Pipes & Filter Pattern? [1 Point]

P-Question: From the following four answers select ***two*** *that fit best.*

- ☐ a) Management of global application state
- ☐ b) IT systems which process data streams
- ☐ c) Decoupling multiple steps of an execution
- ☐ d) Temporal decoupling of an application

23 — *Id: Q-20-04-20*

Which goals are you trying to achieve with the dependency inversion principle? [1 Point]

A-Question: Choose one answer.

- ☐ a) Big building blocks shall not depend on small building blocks.
- ☐ b) Components shall be able to create dependent components more easily.
- ☐ c) Building blocks shall only depend on each other via abstractions.

24 — *Id: Q-20-04-21*

What are characteristics of tight (high) or loose (low) coupling? [2 Points]

K-Question: Assign all the answers.

tight coupling	loose coupling		
☐	☐	a)	Building blocks directly call dependent building blocks, i.e., without using indirect calls via interfaces or abstractions.
☐	☐	b)	Building blocks use shared complex data structures.
☐	☐	c)	Building blocks use a shared table (for read- and write operations) within a relational database.
☐	☐	d)	When designing building blocks, you have consistently applied the dependency inversion principle.

25 — *Id: Q-20-04-14*

Which statements about the principle “Don’t repeat yourself” (DRY) fit best? In other words: What could happen, if parts of the source code or configuration do exist in multiple copies in the system? [1 Point]

P-Question: From the following five answers select ***two*** *that fit best.*

- ☐ a) DRY reduces security.
- ☐ b) Strict adherence to DRY could lead to higher coupling.
- ☐ c) The components of the system that contain redundant code can be improved independently of each other.
- ☐ d) Adherence to DRY leads to additional attack vectors in IT security.
- ☐ e) Applying the Layer patterns allows a consistent application of the DRY principle.

26 — *Id: Q-20-04-15*

You can communicate aspects of your software architecture verbally and/or in writing. How do these variants correlate? Decide for each of the following statements whether it is true or false. [2 Points]

K-Question: Assign all the answers.

true	false		
□	□	a)	Verbal communication should supplement written documentation.
□	□	b)	Feedback to architecture decisions should always be done in writing to ensure traceability.
□	□	c)	Written documentation should always precede verbal communication.
□	□	d)	Architects should pick one variant (verbal or written) and stick to this choice during the whole development.

27 — *Id: Q-20-04-37*

Which of the following statements about notations for architectural views is true and which is false? [2 Points]

K-Question: Assign all the answers.

true	false		
□	□	a)	Business Process Model & Notation (BPMN) should only be used by Business Analysts and not for architecture documentation.
□	□	b)	UML deployment models are the only way to document the mapping of software components to infrastructure.
□	□	c)	UML Package Diagrams can be used to capture the building-block view of software architectures.
□	□	d)	As long as the notation is explained (e.g. by a legend), any notation can be sufficient to describe building block structures and collaboration.

28 — *Id: Q-20-04-13*

Which architectural views have the most practical application for developing software architectures? [1 Point]

P-Question: From the following four answers select ***two*** *that fit best.*

□ a) Pattern View.

□ b) Observer View.

☐ c) Building-Block View (Component View).

☐ d) Deployment View.

29 — *Id: Q-20-04-23*

The context view might contain a business context and a technical context, or both. Pick the most appropriate answers that apply to the technical context. [1 Point]

*P-Question: From the following five answers select **two** that fit best.*

☐ a) The technical context contains the physical channels between your system and its environment.

☐ b) The technical context contains all the infrastructure on which the components of your system are deployed.

☐ c) The technical context should include hardware pricing or pricing of cloud services used as infrastructure for your architecture.

☐ d) The technical context contains information about the chosen programming language as well as all frameworks used to implement your software architecture.

☐ e) The technical context might contain different elements than the business context.

30 — *Id: Q-20-04-24*

Software architecture documentation could contain descriptions of cross-cutting concerns. Pick the best reasons why documentation of cross-cutting concerns is useful. [1 Point]

*P-Question: From the following four answers select **two** that fit best.*

☐ a) Cross-cutting concepts should focus on the domain and be free of technical information.

☐ b) Aspects or concepts that are used in multiple parts of your software architecture should be described in a non-redundant way.

☐ c) Cross-cutting concepts can be reused in more products within the same organization.

☐ d) Cross-cutting concepts should be implemented by specialists. Therefore, separate documentation is useful.

31 — *Id: Q-20-04-25*

What are guidelines for good interface design? Check which of the following statements are true and which are false. [2 Points]

K-Question: Assign all the answers.

true	false		
□	□	a)	Use of interfaces should be easy to learn.
□	□	b)	It should be possible to write client code for the interface, that is reasonably easy to understand.
□	□	c)	An interface should provide access to a comprehensive set of implementation details.
□	□	d)	Interface specifications should contain functional and non-functional aspects.
□	□	e)	Local and remote calls to an interface should behave identically in all aspects.

32 — *Id: Q-20-04-26*

One definition says: “Software architecture is the sum of all the decisions you have taken during development.” Check which of the following statements about architectural/design decisions are true and which are false. [2 Points]

K-Question: Assign all the answers.

true	false		
□	□	a)	Architectural decisions can impact the structure of the building block or components.
□	□	b)	Software architects shall justify all design decisions in writing.
□	□	c)	Architectural decisions can have interdependencies between each other.
□	□	d)	Tradeoffs between conflicting quality requirements should be explicit decisions.

33 — *Id: Q-20-04-31*

Which of the following statements are typical reasons for introducing an architecture documentation and which are not typical reasons? [2 Points]

K-Question: Assign all the answers.

typical	not typical		
□	□	a)	To support onboarding of new developers.
□	□	b)	To support the automated testing approach of the system.
□	□	c)	To support the work of distributed teams.
□	□	d)	To assist in future enhancements of the product.
□	□	e)	To conform to regulatory or legal constraints.
□	□	f)	To ensure that developers have enough work to do.

34 — *Id: Q-20-04-30*

Which of the following pairs of qualities are usually in conflict to each other, and which are not? [2 Points]

K-Question: Assign all the answers.

conflict	no conflict		
□	□	a)	Understandability – Readability
□	□	b)	Usability – Security
□	□	c)	Runtime configurability – Robustness
□	□	d)	Security – Functional correctness

35 — *Id: Q-20-04-27*

ISO 25010 provides generic quality characteristics for software systems. How can quality requirements concerning these characteristics be made more concrete? [1 Point]

A-Question: Choose one answer.

□ a) By developing UI prototypes.

□ b) By defining explicit interfaces.

□ c) By discussing or writing quality scenarios.

□ d) By creating automated tests.

36 — *Id: Q-20-04-28*

Which of the following points are best suited to support the qualitative evaluation of a software architecture with regard to the fulfillment of quality requirements? [2 Points]

*P-Question: From the following six answers select **four** that fit best.*

□ a) Quantitative dependency analysis.

☐ b) Architecture models.

☐ c) Quality scenarios.

☐ d) Team size.

☐ e) Log files.

☐ f) Organizational structure.

37 — *Id: Q-20-04-29*

You try to analyze your architecture quantitatively. Which are the most appropriate indicators for architectural problem areas? [1 Point]

*P-Question: From the following five answers select **two** that fit best.*

☐ a) High coupling of components.

☐ b) Names of public methods do not reflect their purpose.

☐ c) Missing comments.

☐ d) Clusters of errors in certain building blocks of the system.

☐ e) Number of test cases per component.

38 — *Id: Q-20-04-36*

You try to quantitatively analyze your architecture. Which three of the following properties can you measure reliably in your software architecture? Pick the best fitting answers. [1 Point]

*P-Question: From the following five answers select **three** that fit best.*

☐ a) Size of building blocks (e.g. LOC).

☐ b) Change rate of the source code of components.

☐ c) Cohesion of the architectural components.

☐ d) Security level of a component.

☐ e) Number of the developers that contributed to a specific component.