SylLabus for the Course "Testing, Verification and Validation of Artificial Intelligence Systems"

1. Abstract of the discipline

The course Testing, Verification, and Validation of AI Systems aims to develop an engineering culture for the design, verification, and operation of intelligent digital systems.

Modern artificial intelligence systems fundamentally differ from traditional software systems. Their behavior is shaped not only by algorithmic logic but also by statistical models trained on data, making the challenges of reliability, correctness, and interpretability particularly complex.

The course covers:

- the nature of errors and unexpected behaviors («glitches») in AI models;
- the differences between testing, verification, and validation;
- methods for diagnosing and analyzing model behavior;
- architecture of AI quality control systems;
- the role of agent-based systems and mutual model checking;
- methods for integrating expert knowledge into validation processes;
- human-machine architectures for decision verification.

Particular attention is given to systematic result verification methods, including:

- automated testing of AI components;
- mutual model checking (LLM cross-verification);
- agent-based systems for result analysis;
- human-in-the-loop verification processes;
- domain-specific validation.

Large language models (LLMs) are used in the course not only as objects of analysis but also as tools for testing, critical analysis, and building verification systems for AI solutions.

2. General characteristics of the discipline

The course belongs to the professional module of the Master's degree program AI-Augmented Digital Systems Engineering (direction 09.04.02 Information Systems and Technologies).

The total volume of the course is 6 ECTS credits (216 academic hours).

The discipline is implemented in 3rd and 4th semesters.

Class workload:

Semester 3:

- theoretical classes: 24 hours;
- practical classes: 24 hours;
- **Independent work**: 24 hours.

Semester 4:

- theoretical classes: 20 hours;
- practical classes: 20 hours;
- **Independent work**: 32 hours.

Form of interim assessment:

- exam in the 3rd semester;
- exam in the 4th semester.

The course follows the AI-Augmented Engineering Learning format, which involves the active use of large language models in the educational process.

3. The place of the discipline in the structure of the educational program

The discipline is part of the educational track Artificial Intelligence Engineering (AI Systems Engineering). It develops the engineering competencies required to create reliable and verifiable intelligent systems.

The course builds on the learning outcomes of:

- Fundamentals of Artificial Intelligence and Large Language Models;
- Neural Networks, Machine Learning, and Deep Learning;
- Algorithms and Data Structures.

It provides a methodological foundation for:

- Natural Language Processing and Large Language Models;
- Computer Vision;
- Edge AI;
- Architecture of Intelligent Digital Systems;
- Master's project activities.

While previous courses focus on building models, this course emphasizes testing and controlling their behavior. The course bridges the transition: model → system → quality control → operational reliability.

4. Objectives of mastering the discipline

The aim of the course is to develop an engineering understanding of methods for verifying, testing, and validating intelligent systems as an integral part of the AI solution lifecycle.

Key educational outcomes:

- understanding the nature of AI system errors;
- mastering methods for testing models and AI components;
- studying methods for verifying AI results;
- mastering approaches to systemic validation of decisions;
- understanding the role of humans in AI verification processes;
- mastering the architecture of AI quality control systems;
- developing skills in building automated result verification systems;
- fostering a culture of critical analysis of model behavior.

5. Learning objectives

The main objectives are:

- studying the nature of errors and unstable behavior in AI models;
- analyzing limitations of statistical models;
- learning methods for testing AI components;
- mastering techniques for analyzing model errors;
- studying result verification methods;
- exploring the architecture of AI testing systems;
- mastering automated result verification methods;
- studying mutual model checking techniques;
- examining agent-based decision analysis systems;
- applying human-in-the-loop verification methods;
- integrating domain knowledge into verification systems;
- designing AI quality control systems.

6. Planned learning outcomes

Know:

- the difference between testing, verification, and validation;
- the nature of AI model errors;
- sources of hallucinations and incorrect conclusions;
- methods for analyzing model behavior;
- AI testing system architecture;

- automated result verification methods;
- the role of expert knowledge in validation;
- architecture of human-in-the-loop systems.

Be able to:

- identify errors and unstable behavior in AI models;
- design tests for AI components;
- analyze the correctness of model results;
- develop verification procedures;
- apply LLMs to analyze model behavior;
- integrate expert verification into AI systems;
- design AI quality control systems.

Have skills in:

- diagnosing model behavior;
- constructing test scenarios;
- analyzing model errors;
- conducting comparative model testing;
- designing AI verification systems;
- using LLMs as analytical verification tools.

7. Methodological concept of the discipline

The discipline is built around the idea that an AI system is a human-computer system where:

- some decisions are made by the model,
- Some decisions are made by humans.

Accordingly, review processes must account for both decision-making natures.

Methodological axis of the course: goal setting → requirements → model → behavior → verification → interpretation → validation.

7.1 Human-machine nature of AI systems

AI systems are considered as hybrid systems that include:

- algorithms,
- trained models,
- data,
- expert knowledge,
- human decisions.

Validation of results in such systems is impossible without human intervention.

7.2 Differences in verification levels

The discipline distinguishes three control levels:

Testing: checking the correct operation of specific system components.
Verification: ensuring system behavior meets specified requirements.
Validation: confirming the result aligns with the actual needs and purposes of system application.

7.3 Verification begins with goal setting

Any verification system should start with analysis:

- goals of the system;
- user needs;
- criteria for result correctness.

Goal setting determines:

- acceptable error margins;
- quality criteria;
- system applicability limits.

7.4 The nature of AI system errors

Particular attention is paid to analyzing:

- model hallucinations;
- logical reasoning errors;
- statistical artifacts;
- effects of biased training data;
- incorrect generalizations.

7.5 Experimental Verification Culture

The course fosters an understanding that AI system testing should be based on:

- experiments;
- comparative tests;
- model behavior analysis;
- systematic error diagnostics.

8. The role of LLMs in the educational process

Large language models are used in the discipline as tools for engineering activities.

8.1 LLM as a testing object

LLMs serve as examples of complex models whose behavior must be:

- tested;
- analyzed;
- verified.

8.2 LLM as an analytical tool

LLMs are used for:

- analyzing model results;
- detecting errors;
- formulating alternative hypotheses;
- analyzing decision logic.

8.3 LLM as a tool for peer review

The course explores cross-model verification **Practice**s, where one model analyzes the results of another.

8.4 LLM as a component of agent-based verification systems

LLMs can act as components in systems for:

- automatic error diagnostics;
- result analysis;
- test generation;
- hypothesis testing.

8.5 Principle of mandatory verification

Any result obtained using LLMs must undergo additional verification through:

- experimentation;
- model comparison;
- expert assessment.

Professional principle of the course: trust the verification, not the confidence of the model.

9. Educational technologies

The discipline employs:

- **Lecture** classes in the format of engineering analysis;
- **Seminar**s and discussions;
- digital **Lab**oratory work;
- AI model behavior analysis;
- building AI testing systems;
- project activities;
- using LLMs as analytical assistants.

10. Differentiated assessment model

The assessment is based on three levels.
Basic level

- correct execution of model testing;
- understanding error sources;
- basic result analysis.

Advanced level

- development of verification procedures;
- comparative analysis of models;
- explanation of the model's behavior.

Research level

- construction of automatic verification systems;
- use of agent-based analysis systems;
- development of validation system architecture.

11. Final certification
An exam is held each semester.
The exam consists of two parts.

Theoretical part
Understanding is checked:

- the nature of AI errors;
- methods for testing models;
- methods of verification of results;
- AI verification system architectures.

Practical part

The student defends the results of work on an end-to-end project.
Presented:

- task description;
- system architecture;
- model testing methods;
- methods of checking results;
- system error analysis;
- suggestions for improving reliability.

Curriculum schedule for the course

SDS - **Independent work** (Self-Directed Study)

Semester 3

Week	Content	Lectures (hours)	Practice (hours)	SDS (h)
1	**Lecture**. AI systems as human-computer systems. The concept of an AI system as a human-machine decision support complex is examined. The differences between traditional software and artificial intelligence systems, as well as the role of humans in interpreting results and making decisions, are analyzed. **Seminar.** Analysis of real AI systems in medicine, engineering, and recommendation services. Highlighting the roles of humans and algorithms, discussing the distribution of responsibilities. **SDS.** Essay on the role of AI as a decision support tool. **Independent project workflow.** Selecting the subject area for the end-to-end project and describing the future system. **Working with LLMs**. Using LLMs to analyze AI application scenarios.	2	1	1
2	**Lecture**. Understanding goal setting in AI systems. The hierarchy of values, needs, goals, and objectives in the design of intelligent systems is examined. The need for explicit goal formulation as an initial development step is emphasized. **Seminar.** Analysis of the goals of social networking and credit scoring algorithms. Identification of explicit and hidden goals. **SDS.** Identification of system stakeholders and their interests. **Independent project workflow.** Formulating goals for the designed system. **Working with LLMs**. Using LLMs to identify alternative goal formulations.	2	1	1
3	**Lecture**. Goal consistency analysis. Goal conflicts in complex socio-technical systems and methods for identifying them are considered. **Seminar.** Analysis of cases involving conflicting goals and discussion of possible compromises. **Independent workflow**. Construction of a goal hierarchy for the selected system. **Independent project workflow.** Analysis of goal conflicts within the designed system. **Working with LLMs**. Identifying potential goal conflicts.	2	1	1

Week	Content	Lectures (hours)	Practice (hours)	SDS (h)
4	**Lecture**. Degrees of freedom in goal setting and constraints. The solution space formed by the system's goals and the influence of regulatory and technical constraints are considered. **Seminar.** Analysis of the limitations of real AI systems. **SDS.** Determining system constraints. **Independent project workflow.** Formation of the project's solution space. **Working with LLMs**. Generating alternative solutions and analyzing their feasibility.	2	1	1
5	**Lecture**. Criteria and metrics for evaluating AI systems. The transition from system goals to performance criteria and quantitative metrics is considered. **Practice**. Analysis of metrics for various AI systems. **Independent workflow**. Creating a goal-criterion-metric correspondence table. **Independent project workflow.** Defining project metrics. **Working with LLMs**. Generating and analyzing metrics.	2	1	1
6	**Lecture**. Requirements for AI systems. The translation of performance criteria into a system of functional and non-functional requirements is discussed. **Practice**. Developing requirements for an AI system. **Independent workflow**. Creating a requirements list. **Independent project workflow.** Preparing a requirements document. **Working with LLMs**. Analyzing requirements completeness.	2	1	1
7	**Lecture**. AI system architecture: data pipeline, ML pipeline, inference pipeline. Hands-on training. Analysis of existing AI system architectures. **SDS.** Description of the selected system architecture. **Independent project workflow.** Development of a project architectural diagram. **Working with LLMs**. Generation of architectural variants.	2	1	1
8	**Lecture**. Architecture of LLM systems and Retrieval-Augmented Generation (RAG). **Lab**. Creating a simple RAG application. **SDS.** Analysis of the architecture of RAG systems. **Independent project workflow.** Defining the role of retrieval mechanisms. **Working with LLMs**. Experiments with retrieval approaches.	2	1	1
9	**Lecture**. Context and subject area of AI systems. The dependence of the correctness of AI results on the subject context is considered. **Practice**. Analysis of AI system responses in various subject areas. **Independent work assignment**. Description of the system's subject area. **Independent project workflow.** Formation of the project's domain context. **Working with LLMs**. Using LLMs with domain sources.	1	2	2
10	**Lecture**. Enriching validation with domain knowledge. Methods for integrating knowledge bases and regulatory documents into the AI verification system are discussed. **Practice**. Working with domain sources and documents.	1	2	2

Week	Content	Lectures (hours)	Practice (hours)	SDS (h)
	Independent workflow. Collecting domain sources. **Independent project workflow.** Forming a project knowledge base. **Working with LLMs**. Using retrieval approaches.			
11	**Lecture**. Sources of AI system errors: data, models, architecture. **Practice**. Analysis of real model errors. **Independent workflow**. Identifying potential system errors.	1	2	2
12	**Lecture**. Glitches in generative models and hallucinations. **Practice**. Analysis of LLM errors. **Independent workflow**. Documentation of identified errors.	1	2	2
13	**Lecture**. Features of AI system testing. **Practice**. Development of test scenarios. **Independent workflow**. Creation of a test set.	1	2	2
14	**Lecture**. Adversarial testing and stress testing of models. **Practice**. Generation of adversarial queries. **SDS.** Development of stress tests.	1	2	2
15	**Lecture**. Verification and validation in AI systems. **Practice**. Analysis of AI verification cases. **Independent workflow**. Preparation of an analytical report.	1	2	2
16	**Lecture**. Testing the reasoning and factual validity of AI results. **Practice**. Analysis of language model responses. **Independent work**. Exam preparation.	1	2	2

Semester 4

Week	Content	**Lecture**s (hours)	Practice (hours)	**SDS** (h)
1	**Lecture**. Automated verification of AI results. The sca**Lab**ility issues of AI system verification and the limitations of fully manual expert validation are discussed. Automatic verification architectures and the role of verification in modern AI applications are analyzed. **Seminar**. Analysis of errors in real AI systems and discussion of automated verification capabilities. **SDS**. Preparing a review of methods for automatically verifying AI results. **Independent project workflow**. Identifying elements of the project system subject to automated verification. **Working with LLM**. Generating alternative solutions to problems and analyzing differences between model results.	2	2	3
2	**Lecture**. Using LLM to evaluate results (LLM as a judge). Methods of using language models to evaluate the responses of other models are considered. The advantages and limitations of this approach are analyzed. **Lab**. Using one model to evaluate the results of another model. **SDS**. Comparison of result scores obtained from different models. **SDS — project**. Development of a scheme for automatic	2	2	3

Week	Content	**Lectures** (hours)	Practice (hours)	**SDS** (h)
	results evaluation for the designed system. **Working with LLM**. Cross-evaluation of model responses.			
3	**Lecture**. Self-consistency and mutual checking of models. Methods for mutual checking of results from several models and the use of model ensembles to increase reliability are considered. **Lab**. Generating multiple solutions to a single problem and analyzing the consistency of the results. **SDS**. Experiments comparing the results of different models. **Independent project workflow**. Adding a mechanism for mutual checking of results. **Working with LLM**. Generating multiple answer options and analyzing their consistency.	2	2	3
4	**Lecture**. Retrieval-based verification. Methods for verifying AI system results based on external information sources and knowledge bases are discussed. **Lab**. Verifying model responses using search engines and documentary sources. **Independent work**. Analysis of cases of model hallucinations. **Independent project workflow**. Integrating verification mechanisms through knowledge sources. **Working with LLM**. Using retrieval systems for fact checking.	2	2	3
5	**Lecture**. Agent-based AI error detection systems. System architectures using specialized agents — critics, opponents, and fact checkers — are discussed. **Lab**. Building a simple agent-based model analysis framework. **SDS**. Designing the structure of an agent-based verification system. **Independent work**flow. Developing a project verification architecture based on agent-based models. **Working with LLM**. Using LLMs as intelligent agents.	2	2	3
6	**Lecture**. Systematic search for AI bugs. Methods for systematically identifying AI system errors and analyzing the space of possible errors are discussed. **Lab**. Generating adversarial queries that provoke model errors. **SDS**. Developing a set of test scenarios. **Independent project workflow**. Creating a test set for the project system. **Working with LLM**. Generating provocative queries.	2	2	3
7	**Lecture**. Expert validation of AI systems. The role of domain experts and human-AI interaction models in verifying results is examined. Practice. Developing expert criteria for evaluating AI system results. **Independent work**flow. Preparing an expert review checklist. **Independent project workflow**. Defining the expert's role in the system architecture. **Working with LLM**. Generating potential expert evaluation criteria.	2	2	3
8	**Lecture**. Contextual verification of AI systems. The dependence of the correctness of AI results on the subject context and domain knowledge is considered. Practice. Analysis of AI system responses in various subject areas. **Independent work**. Determining the subject context of the	2	2	3

Week	Content	**Lecture**s (hours)	Practice (hours)	**SDS** (h)
	system. **Independent project workflow**. Generating a domain description. **Working with LLM**. Using LLMs with domain sources and knowledge bases.			
9	**Lecture**. Verification pipeline. The architecture of the AI system testing pipeline is discussed: input validation, retrieval validation, response verification. Practice. Designing a verification pipeline for a specific system. **SDS**. Development of a verification pipeline diagram. **Independent work**flow. Formation of a project verification architecture. **Working with LLM**. Analysis of the pipeline architecture.	2	2	4
10	**Lecture**. AI reliability engineering. The **Lecture** covers the principles of ensuring the reliability of AI systems, including monitoring, observability, and continuous quality assessment. Practice. Presentation and discussion of the reliability architecture of the designed system. **SDS**. Preparation for final certification. **Independent project workflow**. Finalization of the system reliability architecture. **Working with LLM**. Analysis of the system architecture and risk identification.	2	2	4