CONSTRUCTOR UNIVERSITY

Artificial Intelligence in Digital Societies and Business

Project Report

Name: Sagar Kumar & Betül Yurtman

Matriculation No.: 30000402 & 30008426

Course: MDSSBO-201-AI in DSSB

Semester: Fall 2025

Instructor: Prof. Dr. Wilhelm & Prof Dr. Brockmann

Word Count: ~3200 words

Index

Abstract

This report is part of the *Artificial Intelligence in Business and Society* course. We are tasked with designing and implementing a prototype AI-driven solution that addresses a real-world academic problem, while also reflecting on its technical design, use case, ethics, legal implications and more. In our case, we chose to create an AI-powered Quiz Generator that supports students in their exam preparation by automatically transforming course materials into personalized quizzes and provides instant feedback. The prototype is built in n8n and integrates document processing, AI prompt engineering and database storage into a single automated workflow. This project and report demonstrates how generative AI and workflows can be combined to create effective learning support tools while also discussing the technical, ethical, and practical considerations involved in deploying such tools.

All files for this project, including this report, can be found at the following public Github Repository: https://github.com/skumar1996github/AI_in_DSSB_Project

The link is also available as the first link in the References section on the last page.

The workload for the project was divided evenly between Sagar and Betül. The n8n workflow was built in close cooperation with each other in pairing sessions on Teams and the report was divided evenly between both team members with constant peer-reviews.

Note that the n8n JSON workflow file will not work right out of the box when importing into a n8n workspace because many things need to be configured before-hand like the gmail account, a free valid AI-model from OpenRouter and creating a manual database within the n8n system. For this reason we have provided screenshots of the workflow and key configuration steps to illustrate the system design and functionality without requiring full setup of the environment.

1. Introduction & Motivation

This report shows the development and concept of an AI-powered Quiz Generator created in n8n that is designed to support students with the preparation of their exams and studies. The report will explore how n8n is used as a low-code workflow tool to create the entire workflow: from ingesting course materials and user preferences to generating quizzes and feedback using generative AI models. The prototype implementation will use technical concepts like text processing, automated AI text generation with structured outputs and database management with read/writes.

As students ourselves, the motivation for this project comes from the common challenge we students face when learning large amounts of course material in a limited amount of time. Some students rely on passive study methods such as rereading slides or notes, which are often inefficient and make it difficult to identify knowledge gaps. Also, creating practice questions manually is time-consuming and you don't always make sure you are covering every topic you need. This project is motivated by the idea that we can use generative AI to reduce the effort by automatically converting learning materials into quizzes and feedback. Another advantage of the tool is that feedback is immediate and tailored specifically for the student. This will allow students to save time and quickly identify which topics they understand well and which areas they can still improve. By automating this process, we hope that our custom n8n tool can help students study more efficiently and in a more personalized manner.

2. Problem Definition & Use Case

2.1 Problem Statement

The core problem we are trying to solve in this project is that students often struggle to actively test their understanding when preparing for exams, as creating practice questions manually is time-consuming and typically depends on instructors to give feedback. The AI-powered Quiz Generator helps solve this by allowing students to quickly generate personalized quizzes directly from their course PDFs. This is convenient because the AI can use this information to make quizzes that are designed to include topics relevant for the exam and exclude topics that will not be tested, as they are not present in the pdf file. With the addition of user preferences like “quiz difficulty”, the user will also be able to learn gradually at their own pace. In addition, our tool provides instant feedback by collecting the users answers, comparing the results and then based on the question and topics of the quiz, the student will know

immediately which topics they performed the best and worst on. This will allow for continuous self-assessment and help students learn more efficiently without relying on professors.

2.1 Target Users and Stakeholders

A tool like this can be useful for **teachers**, **professors**, and **course designers** who want to provide students with effective self-assessment tools without creating quizzes manually. It is also useful for **independent tutors** and **coaching centers** that would like to offer personalized practice materials. Lastly, it is also relevant for **edtech companies** that are looking to integrate AI-driven learning features into their platforms and for students who want an efficient way to test their knowledge and receive instant feedback.

3. IT Infrastructure Description

In this section we will discuss the technical infrastructure used to build and run the AI-powered Quiz Generator, focusing on the platform, userflow, data sources, collection, storage and more. Since we only had a few months to develop the prototype, we will discuss the short-term implementation as it is during the time of submission and we will discuss the long-term goals for the project if we had more time to develop it.

3.1 Platform

The prototype tool is currently hosted and runs exclusively on n8n. The workflow itself is saved as a JSON file that can be downloaded and uploaded on different n8n accounts so the workflow can be shared. The workflow itself contains in-built nodes from n8n's node library. The long-term goal for this tool is to host it on a dedicated server or cloud-based environment that allows the tool to run independently of n8n and supports multiple users simultaneously. This setup would make it possible to improve scalability, and add more features such as user profiles etc.

3.2 Data Sources and Data Collection

For data collection, the current n8n workflow only accepts a PDF file containing the student's study materials, which is manually uploaded at the start of the workflow. This approach is sufficient for our prototype however it lacks in automatic updates and scalability. Therefore, in the long term, the tool could be extended to support automated and scheduled data ingestion. This will allow it to continuously collect new and updated course materials from learning platforms.

3.3 API Infrastructure

The prototype uses external APIs from OpenRouters LLM's to support generative AI for creating quiz questions and generating feedback. These APIs are connected directly within the n8n workflow's "AI Agent" node. It allows for data and structured outputs to flow neatly between steps. In the short term, API usage is kept simple and limited to essential functions. However, in the long term, additional APIs such as user authentication and file storage can be added to improve personalization and security.

3.4 Data Storage

In the current prototype, all data is stored using n8n's built-in data storage features, in the "Quizzes" database. This data table contains quiz questions, general topic user answers correctly and some meta data which is generated during each workflow run. This data is used for the feedback process in the workflow. While this approach is sufficient for testing and demonstration purposes, it is limited in terms of scalability and long-term data management. In the long term, the system could be extended to use an external, scalable database such as MongoDB. This would allow persistent storage of user profiles, quiz history, performance tracking over time etc.

4. Background of AI Tools

In this section, we will provide the details of the artificial intelligence technologies behind the Quiz Generator, focusing on the concepts of Generative AI, Large Language Models (LLMs), and Prompt Engineering. Additionally, we explain the role of the OpenRouter API as the infrastructure layer.

4.1 Generative AI and Large Language Models (LLMs)

A Large Language Model (LLM) is a type of deep learning algorithm that can recognize, summarize, translate, predict, and generate text based on knowledge gained from massive datasets. Unlike traditional rule-based software that follows rigid, pre-defined commands, LLMs operate probabilistically; they predict the most likely next "token" (word or character) in a sequence to generate coherent and context-aware responses (Zhao et al., 2023).

In the context of this project, Generative AI serves as the core reasoning engine. It allows the system to:

- **Process Unstructured Data:** It can read raw text extracted from PDF course materials and understand the semantic meaning of the content.
- **Generate Structured Outputs:** It converts this unstructured text into a strict JSON format required for the quiz database, identifying key concepts to form questions, correct answers, and distractors (wrong options).
- **Provide Reasoning:** It analyzes student performance to generate personalized feedback and study plans, mimicking a human tutor's advice.

Merely having access to an LLM is not sufficient for automation; the model must be guided via **Prompt Engineering**. In our workflow, we utilize custom system prompts to instruct the AI to act as a "quiz generator" based on specific constraints like difficulty level (Beginner, Intermediate, Advanced).

4.2 OpenRouter API

To implement these LLM capabilities, the prototype utilizes **OpenRouter**, an API aggregation interface. OpenRouter acts as a standardized gateway that provides unified access to various state-of-the-art models (such as OpenAI's GPT-4, Anthropic's Claude, or open-source models like Llama and Mistral) (OpenRouter, 2025).

Using OpenRouter provided access to "free" and low-cost models, which was essential for developing a student-focused prototype without incurring high operational costs.

5. User Flow & Prototype Implementation Details

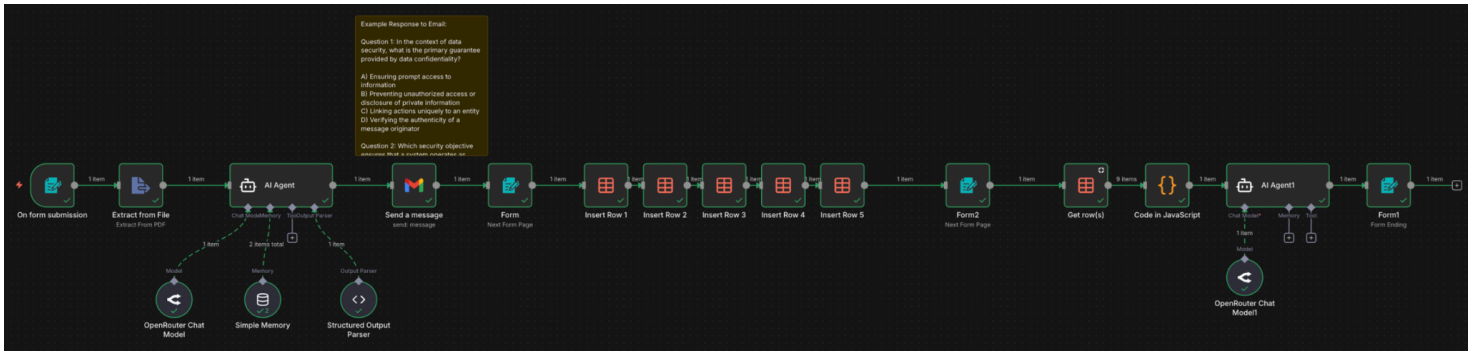
In this section, we will describe how the prototype is implemented in n8n and explain the user flow from the moment a student uploads their study material to the point where quizzes and feedback are generated.

5.1 Workflow Overview

The following is an overview of the n8n workflow. It consists of multiple nodes strung together to automate the process from ingesting study materials to generating quizzes. We start with the "On form submission" node where the user is prompted to provide the study materials inside a form. The text is processed by the "Extract from File" node which is then sent to the first "AI Agent" node which converts the text to a JSON structure which contains information about the quiz. A copy of the quiz is then sent to the user's email from the "Send a message" node and

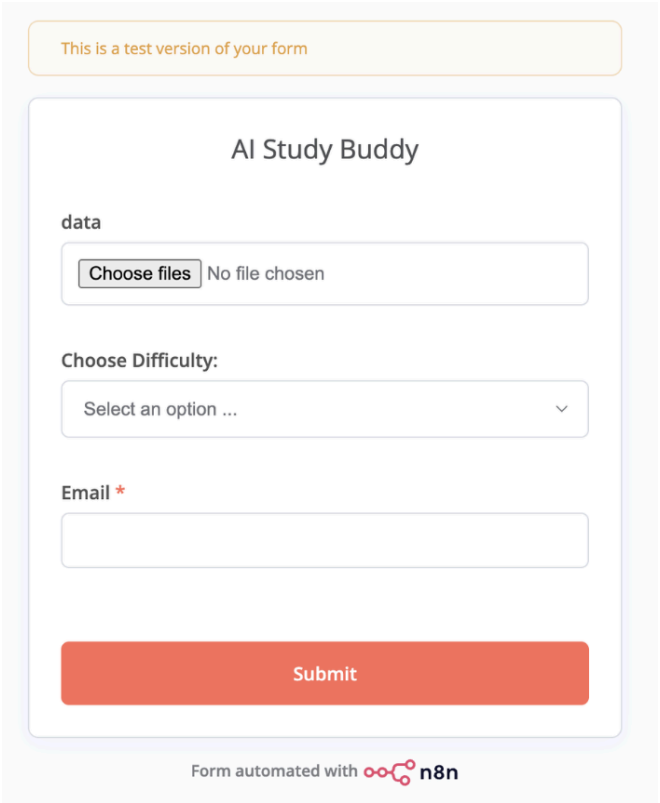
once they finish the quiz, their answers are stored in the database by the “Insert Row [number]” nodes. The user is then shown their results and after they continue their results are fed to the second “AI Agent” which processes the results and outputs feedback for the student.

We will now take a closer look at the user flow. It is divided into 4 distinct steps: upload content, take the quiz, get results and get feedback.



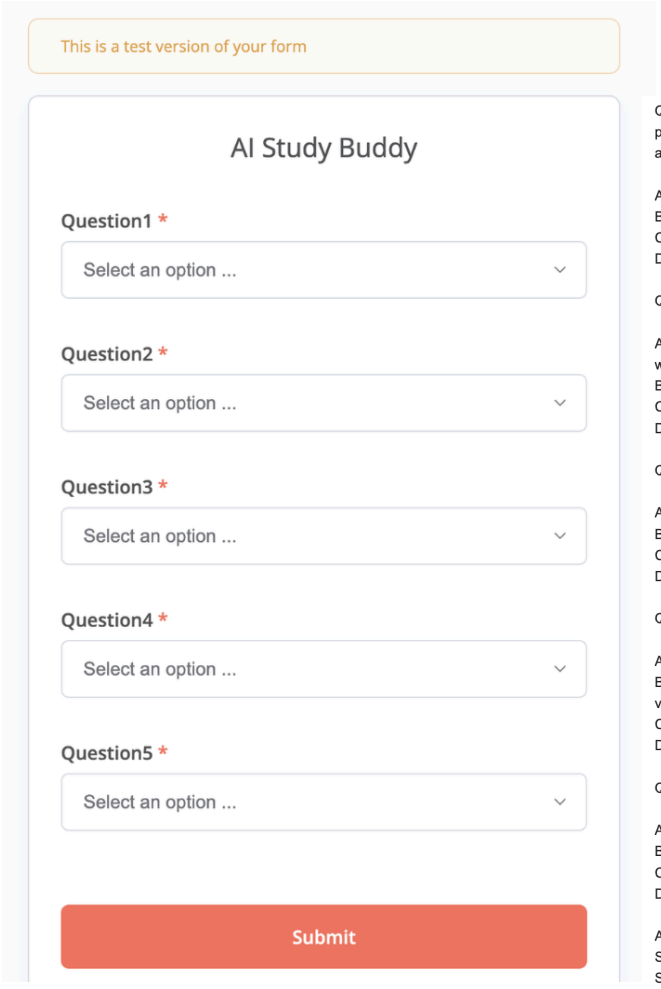
Step 1: Upload Content and Set Preferences

The first step of the user flow is to input the study material by clicking the “Choose files” button and uploading a PDF file. The user can also choose a difficulty between “Beginner”, “Intermediate” and “Advanced”. This option determines the difficulty of the quiz. The user can then also provide an email to which they will be sent a copy of the quiz.



Step 2: Take The Quiz

The next step is to take the quiz itself. In the example below, I have uploaded course material from the course “Data Security and Privacy” from the second semester of the Masters in Data Science programme. The questions and answers key is sent to the user’s email, provided in the first step. Therefore, the user now has access to the quiz and the form to complete it. There are 5 questions per workflow run and they are chosen from at least two different topics so the quiz covers multiple topics. The dropdown menu has four options: A, B, C and D. Once the student is finished with the quiz, they can press Submit. Note that there is no time limit, the user can take as much time as they need. Therefore, one way to improve our prototype would be the ability to simulate exam conditions so the student is able to take a mock exam for themselves.



Question 1: Which security objective primarily encompasses strategies for protecting personal privacy and maintaining legally mandated limits on information access and disclosure?

A) Integrity
B) Availability
C) Confidentiality
D) Accountability

Question 2: System integrity ensures that a system:

A) operates as planned, free from intentional or unintentional illegal modifications, without interruption
B) has prompt and reliable access to information
C) provides confidence in the authenticity of the message originator
D) links actions uniquely to an entity

Question 3: In the OSI Security Architecture, a security mechanism is defined as:

A) any activity that compromises the security of data
B) a processing or communication service that strengthens security
C) a procedure intended to detect, prevent, and recover from security breaches
D) a potential danger to an information system

Question 4: Which of the following correctly differentiates a threat from an attack?

A) Threats are always intentional, while attacks can be unintentional
B) A threat is a potential danger or risk, while an attack is an actual exploitation of a vulnerability
C) Attacks are potential risks, while threats cause actual harm
D) Both threats and attacks are actual exploitations of vulnerabilities

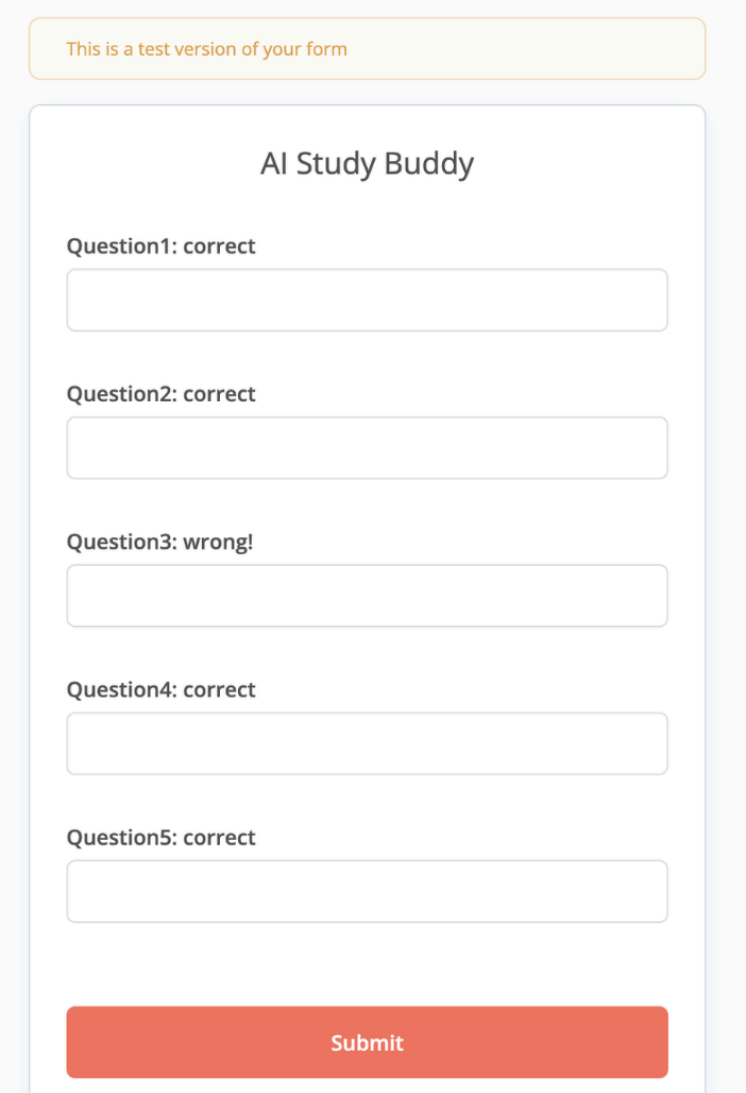
Question 5: Which of the following is classified as an active attack?

A) Traffic analysis
B) Release of message content
C) Masquerade
D) Eavesdropping

Answer Sheet:
Solution 1: C
Solution 2: A
Solution 3: C
Solution 4: B
Solution 5: C

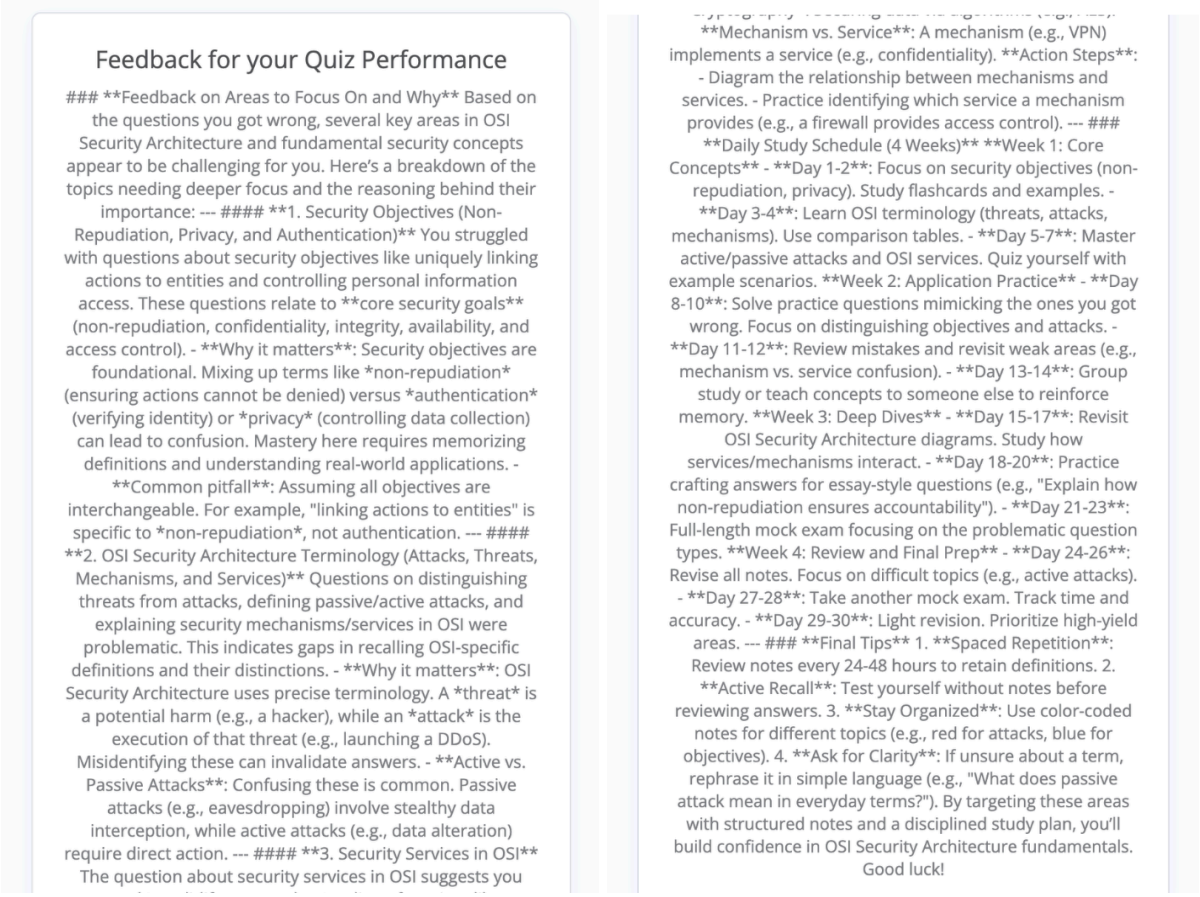
Step 3: Get Quiz Results

After completing the quiz the user can now see their results. If they answered the question correctly, the form header will say “correct”, otherwise “wrong!”. These results are subsequently stored in the “Quizzes” dataset along with metadata about the time they answered the question and more.



Step 4: Get Feedback

Once the quiz has been completed, the database stores all questions and answers answered by the user. Based on the questions they got right and wrong, the AI agent provides detailed feedback consisting of areas to focus more attention, a full-fledged 30-day study plan, studying tips and more.



Feedback for your Quiz Performance

Feedback on Areas to Focus On and Why Based on the questions you got wrong, several key areas in OSI Security Architecture and fundamental security concepts appear to be challenging for you. Here's a breakdown of the topics needing deeper focus and the reasoning behind their importance: --- #### **1. Security Objectives (Non-Repudiation, Privacy, and Authentication)** You struggled with questions about security objectives like uniquely linking actions to entities and controlling personal information access. These questions relate to **core security goals** (non-repudiation, confidentiality, integrity, availability, and access control). - **Why it matters**: Security objectives are foundational. Mixing up terms like *non-repudiation* (ensuring actions cannot be denied) versus *authentication* (verifying identity) or *privacy* (controlling data collection) can lead to confusion. Mastery here requires memorizing definitions and understanding real-world applications. - **Common pitfall**: Assuming all objectives are interchangeable. For example, "linking actions to entities" is specific to *non-repudiation*, not authentication. --- #### **2. OSI Security Architecture Terminology (Attacks, Threats, Mechanisms, and Services)** Questions on distinguishing threats from attacks, defining passive/active attacks, and explaining security mechanisms/services in OSI were problematic. This indicates gaps in recalling OSI-specific definitions and their distinctions. - **Why it matters**: OSI Security Architecture uses precise terminology. A *threat* is a potential harm (e.g., a hacker), while an *attack* is the execution of that threat (e.g., launching a DDoS). Misidentifying these can invalidate answers. - **Active vs. Passive Attacks**: Confusing these is common. Passive attacks (e.g., eavesdropping) involve stealthy data interception, while active attacks (e.g., data alteration) require direct action. --- #### **3. Security Services in OSI** The question about security services in OSI suggests you

Mechanism vs. Service: A mechanism (e.g., VPN) implements a service (e.g., confidentiality). **Action Steps**: - Diagram the relationship between mechanisms and services. - Practice identifying which service a mechanism provides (e.g., a firewall provides access control). --- ### **Daily Study Schedule (4 Weeks)** **Week 1: Core Concepts** - **Day 1-2**: Focus on security objectives (non-repudiation, privacy). Study flashcards and examples. - **Day 3-4**: Learn OSI terminology (threats, attacks, mechanisms). Use comparison tables. - **Day 5-7**: Master active/passive attacks and OSI services. Quiz yourself with example scenarios. **Week 2: Application Practice** - **Day 8-10**: Solve practice questions mimicking the ones you got wrong. Focus on distinguishing objectives and attacks. - **Day 11-12**: Review mistakes and revisit weak areas (e.g., mechanism vs. service confusion). - **Day 13-14**: Group study or teach concepts to someone else to reinforce memory. **Week 3: Deep Dives** - **Day 15-17**: Revisit OSI Security Architecture diagrams. Study how services/mechanisms interact. - **Day 18-20**: Practice crafting answers for essay-style questions (e.g., "Explain how non-repudiation ensures accountability"). - **Day 21-23**: Full-length mock exam focusing on the problematic question types. **Week 4: Review and Final Prep** - **Day 24-26**: Revise all notes. Focus on difficult topics (e.g., active attacks). - **Day 27-28**: Take another mock exam. Track time and accuracy. - **Day 29-30**: Light revision. Prioritize high-yield areas. --- ### **Final Tips** 1. **Spaced Repetition**: Review notes every 24-48 hours to retain definitions. 2. **Active Recall**: Test yourself without notes before reviewing answers. 3. **Stay Organized**: Use color-coded notes for different topics (e.g., red for attacks, blue for objectives). 4. **Ask for Clarity**: If unsure about a term, rephrase it in simple language (e.g., "What does passive attack mean in everyday terms?"). By targeting these areas with structured notes and a disciplined study plan, you'll build confidence in OSI Security Architecture fundamentals. Good luck!

5.2 Implementation Details

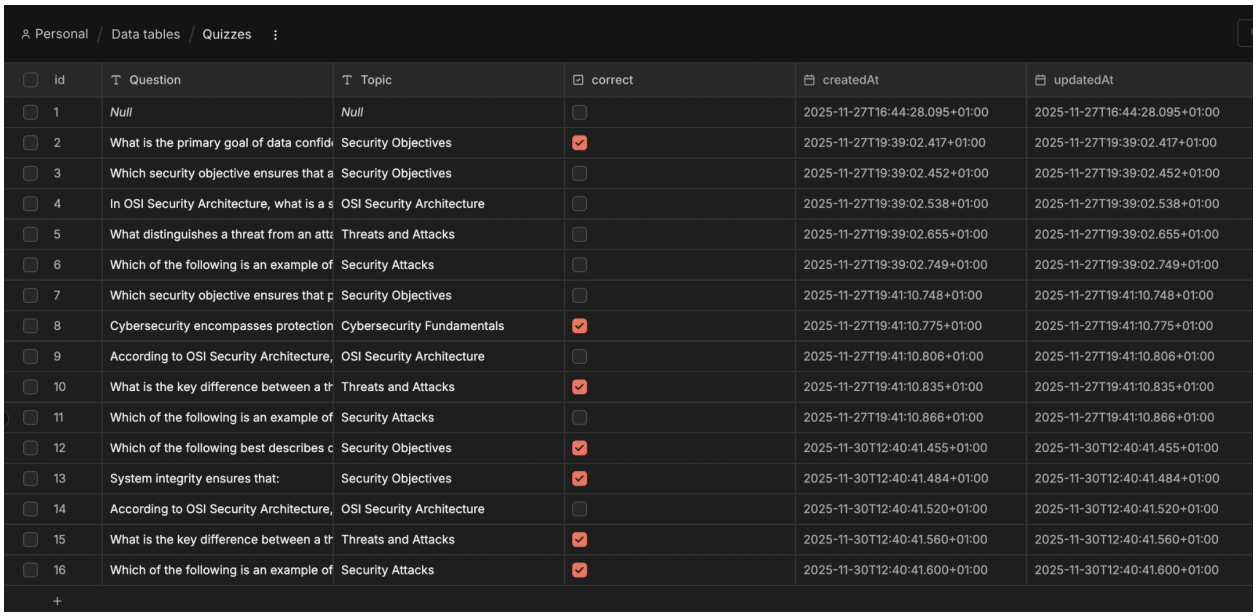
5.2.1 Artificial Intelligence for creating the Quiz

The first “AI Agent” node in the workflow turns the extracted text from the uploaded PDF into a multiple-choice quiz. This is done using a custom prompt that tells the AI to generate five questions based on the course content and the difficulty level chosen by the user. The quiz is returned in a strict JSON format, which allows the next steps in the workflow to easily understand and store the data.

```
You are a quiz generator.
Based on the following content, and difficulty level, create a 5-question
multiple choice quiz.
Each question should have 4 options (A, B, C, D) and clearly state the correct
answer. Classify the question into a broad general topic, only asking questions
from 4 different topics at a time. The answer should be a random option amongst
the four options, not just option A every time.

Content:
{{ $json.text }}

Difficulty level:
{{ $('On form submission').item.json['Choose Difficulty: '] }}

Return the result strictly as valid JSON in this structure:

{
  "questions": [
    {
      "id": number,
      "topic": "string",
      "question": "string",
      "options": { "A": "string", "B": "string", "C": "string", "D": "string" },
      "correct_answer": "string"
    }
  ]
}

Do not enclose the JSON in quotes or escape characters. Output only raw JSON,
nothing else. Return only the raw JSON object as specified above, with no
additional text, explanation, or formatting. Do not return an array or any other
structure.
```

5.2.2 Data Storage

The results of the quiz are stored in a custom relational database, where each question, topic, and whether the answer was correct or not is saved for later analysis and feedback generation. All columns are stored as type “String”, but only the column “correct” is a boolean. We also store meta-data about the data point with createdAt and updatedAt timestamps.



Personal / Data tables / Quizzes

id	Question	Topic	correct	createdAt	updatedAt
1	*Null*	*Null*	☐	2025-11-27T16:44:28.095+01:00	2025-11-27T16:44:28.095+01:00
2	What is the primary goal of data confid	Security Objectives	☑	2025-11-27T19:39:02.417+01:00	2025-11-27T19:39:02.417+01:00
3	Which security objective ensures that a	Security Objectives	☐	2025-11-27T19:39:02.452+01:00	2025-11-27T19:39:02.452+01:00
4	In OSI Security Architecture, what is a s	OSI Security Architecture	☐	2025-11-27T19:39:02.538+01:00	2025-11-27T19:39:02.538+01:00
5	What distinguishes a threat from an atta	Threats and Attacks	☐	2025-11-27T19:39:02.655+01:00	2025-11-27T19:39:02.655+01:00
6	Which of the following is an example of	Security Attacks	☐	2025-11-27T19:39:02.749+01:00	2025-11-27T19:39:02.749+01:00
7	Which security objective ensures that p	Security Objectives	☐	2025-11-27T19:41:10.748+01:00	2025-11-27T19:41:10.748+01:00
8	Cybersecurity encompasses protection	Cybersecurity Fundamentals	☑	2025-11-27T19:41:10.775+01:00	2025-11-27T19:41:10.775+01:00
9	According to OSI Security Architecture,	OSI Security Architecture	☐	2025-11-27T19:41:10.806+01:00	2025-11-27T19:41:10.806+01:00
10	What is the key difference between a th	Threats and Attacks	☑	2025-11-27T19:41:10.835+01:00	2025-11-27T19:41:10.835+01:00
11	Which of the following is an example of	Security Attacks	☐	2025-11-27T19:41:10.866+01:00	2025-11-27T19:41:10.866+01:00
12	Which of the following best describes c	Security Objectives	☑	2025-11-30T12:40:41.455+01:00	2025-11-30T12:40:41.455+01:00
13	System integrity ensures that:	Security Objectives	☑	2025-11-30T12:40:41.484+01:00	2025-11-30T12:40:41.484+01:00
14	According to OSI Security Architecture,	OSI Security Architecture	☐	2025-11-30T12:40:41.520+01:00	2025-11-30T12:40:41.520+01:00
15	What is the key difference between a th	Threats and Attacks	☑	2025-11-30T12:40:41.560+01:00	2025-11-30T12:40:41.560+01:00
16	Which of the following is an example of	Security Attacks	☑	2025-11-30T12:40:41.600+01:00	2025-11-30T12:40:41.600+01:00

5.2.3 Artificial Intelligence for creating Feedback

Once the user takes the quiz and their results are stored in the database, we can now feed this information to the second “AI Agent” node in the workflow to get proper feedback based on the user's answers. We use the following prompt to generate personalized feedback and study guidance by providing the list of questions that the user answers incorrectly in the $json.questionString variable.

```
These are the list of questions I got wrong: {{ $json.questionString }}.
Based on these questions, give me feedback on which topics I should focus on and
why. and then construct me a study plan + notes to help me study for the
questions I got wrong. Make the output text atleast 400 words and output the
text iin a organized and pretty matter with clearly marked sections and newlines
to keep things spaced out. Add a newline after every section.
```

6. Evaluation/Validation of Performance

To validate the efficacy of the AI-powered Quiz Generator, we conducted a functional performance evaluation using real-world course materials. The primary goal was to verify that the n8n workflow could successfully transform unstructured PDF content into a structured, interactive learning experience without manual intervention.

6.1 Functional Testing and Relevance

The system was tested using lecture slides from the "Data Security and Privacy" course. The prototype successfully processed the complex academic text, demonstrating that the system effectively grounds the AI in the provided material. The generated questions accurately reflected key topics found in the PDF, proving the system's ability to identify relevant exam content automatically. Furthermore, the system successfully maintained the strict JSON schema required for the database throughout the process, ensuring a seamless user experience from upload to result generation.

6.2 Feedback Accuracy

We also validated the quality of the AI-generated feedback. When incorrect answers were submitted during testing, the AI successfully identified the specific knowledge gaps and provided logical, structured explanations rather than simple corrections. This confirmed the tool’s potential to act as an effective automated tutor.

6.3 Future Enhancement

A crucial future enhancement involves implementing **Automated Cross-Validation** by establishing a **multi-agent debate** framework (Du et al., 2023). This entails adding a secondary **"Validator Agent"** to the workflow that independently solves the questions generated by the first AI to verify the answer key. Adopting this **"LLM-as-a-Judge"** approach (Zheng et al., 2023) allows the system to cross-reference questions before they ever reach the user. This strategy significantly reduces the risk of AI hallucinations or factual errors, ensuring the highest level of academic accuracy for students.

7. Ethical/Legal Considerations & Solutions

The integration of Artificial Intelligence into educational workflows introduces significant ethical and legal responsibilities. While the Quiz Generator offers efficiency, its deployment requires careful management of data privacy, intellectual property, and academic integrity to ensure it aids rather than hinders the learning process.

7.1 Data Privacy and GDPR Compliance

Since the tool processes student inputs, compliance with GDPR is paramount (GDPR Text, n.d.).

- **The Risk:** The prototype stores performance metrics (questions, topics, correctness) to generate feedback but explicitly excludes email addresses from persistent storage to protect anonymity. However, even without stored Personally Identifiable Information(PII), aggregated metadata requires secure handling to prevent potential re-identification.
- **The Solution:** We implement strict **Data Minimization**. We will continue the practice of not storing PII and ensure that temporary email transmission is encrypted. Furthermore, we strictly select API providers with "zero-retention" policies, ensuring that course data is processed in-memory and never used to train external AI models.

7.2 Accuracy and AI Hallucinations

A major ethical concern in educational AI is the potential for "hallucinations"—instances where the model generates plausible but factually incorrect questions or feedback (Ji et al., 2023).

- **The Risk:** If the AI generates a quiz with incorrect answers, it could negatively impact a student's exam preparation, leading to confusion rather than clarity.
- **The Solution:** As discussed in the evaluation section, implementing an **Automated Cross-Validation** (Validator Agent) workflow is the primary technical mitigation. Additionally, from a user interface perspective, the tool must include clear **Disclaimers** stating that the tool is a "study aid" and that students should verify critical information against their primary lecture notes.

7.3 Intellectual Property (Copyright)

The system relies on users uploading PDF course materials, which are often the intellectual property of the professor or the university.

- **The Risk:** There is a legal ambiguity regarding whether processing these files through an AI constitutes a violation of copyright, especially if the AI provider retains the data.
- **The Solution:** The tool operates under the principle of **"Fair Use"** for educational purposes, as the reproduction is temporary and transformative (converting slides to quizzes).

8. Conclusion

This project set out to address a common challenge in higher education: the inefficiency of passive study methods and the lack of personalized, immediate feedback for students. By leveraging the low-code capabilities of n8n and the reasoning power of Generative AI, we successfully designed and implemented a functional prototype of an **AI-powered Quiz Generator**.

The development process demonstrated that it is possible to automate the transformation of unstructured course materials (PDFs) into structured, interactive learning assessments. The prototype successfully validated the core user flow—from ingesting lecture slides to generating relevant multiple-choice questions and providing logic-driven feedback.

However, as a prototype, the current system has identified limitations regarding scalability and data persistence. While the use of n8n's internal tools was sufficient for proof-of-concept, a production-grade deployment would require the integration of external databases (like MongoDB) and the implementation of the proposed "Validator Agent" to ensure absolute academic accuracy.

In conclusion, this project illustrates the potential of integrating AI workflows into educational environments. By adhering to strict data privacy standards and focusing on "zero-retention" architectures, tools like this can provide significant value—saving students time and enabling active recall—while navigating the ethical complexities of AI in education.

References (APA-7)

Github Project Repository: https://github.com/skumar1996github/AI_in_DSSB_Project

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.-Y., & Wen, J.-R. (2023). **A Survey of Large Language Models.** *arXiv preprint arXiv:2303.18223*. https://doi.org/10.48550/arXiv.2303.18223

OpenRouter. (2025). *Unified interface for LLMs*. https://openrouter.ai/

Du, Y., Li, S., Torralba, A., Tenenbaum, J. B., & Mordatch, I. (2023). **Improving Factuality and Reasoning in Language Models through Multiagent Debate.** *arXiv preprint arXiv:2305.14325*. https://doi.org/10.48550/arXiv.2305.14325

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Eric, P., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). **Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena.** *Advances in Neural Information Processing Systems*, 36, 46595–46623. https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html

Intersoft Consulting. (n.d.) ***General Data Protection Regulation (GDPR)* – Official legal text.** GDPR-info.eu. https://gdpr-info.eu/

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., & Fung, P. (2023). **Survey of hallucination in natural language generation.** ACM Computing Surveys, 55(12), 1–38. https://doi.org/10.1145/3571730