Design Proposal Report for an Academic Research Online Agent: A Hybrid BDI Approach to Data Retrieval, Processing, and Storage

[Student ID]

[Module]

[Date]

Table of Contents

Part 1: Team Design Proposal Report

Introduction

The intelligent agents are now critical in automating knowledge-intensive jobs in research and industry. In research, manual information retrieval is inefficient and prone to errors due to the exponential growth of digital content. Scalable solutions through agent-based systems address the evidence-based decision-making requirement by collecting, processing, and organising relevant information autonomously (Wooldridge, 2009). Nevertheless, critics observe that unfiltered automated systems can increase misinformation or recall irrelevant information (Saeidnia et al., 2025). This proposal describes how to design an agent system to fit Academic Research Online, emphasising data retrieval, processing, and structured storage. The purpose is to illustrate that agent-based methods may minimise the information load and ensure ethical and technical adherence to the research principles.

System Requirements

The suggested system will need software libraries and hardware functionality to maintain efficient performance. Its development will be based on Python 3.11+ because it is flexible and has extensive web scraping library support, including Scrapy and BeautifulSoup, a natural language filtering library, NLTK and a data structuring library, Pandas (Searle,1969). PostgreSQL will support structured data, and MongoDB will support unstructured content, allowing the flexibility of various types of data. The hardware specifications are a multi-core processor, 16GB RAM, and a stable network used to connect with concurrent queries. Compliance requirements include encrypted storage and secure APIs, data integrity in accordance with the GDP and copyright regulations (Hussain et al., 2020). Open-source frameworks offer cost-effectiveness and flexibility over proprietary tools, but require greater technical expertise and vigorous maintenance habits, which not all organisations can easily afford.

Design Decisions

Academic Research Online over Digital Forensics was chosen because of the greater generalisability of academic search tools and decreased ethical risks of working with non-sensitive data. The system uses a Belief-Desire-Intention (BDI) model where the agents can act with a goal-directed behaviour but are still flexible in a dynamic search environment. A crawler agent, parser, processor, and storage module were selected as a modular architecture. This isolation of concerns increases scalability and fault isolation. PostgreSQL was chosen to store structured bibliographic records, whereas MongoDB will store semi-structured metadata.

Despite strong consistency, relational databases may not be as efficient with large unstructured data sets as NoSQL solutions (Kunda and Phiri, 2017). Finally, the hybrid solution is more efficient and more accurate.

Methodology & Development Approach

The project will be developed using an Agile approach, whereby designing, testing, and refining agent components via iteration will be possible. Agile is selected instead of the classic Waterfall model because academic research systems are subject to changing demands, and flexibility is essential (Gupta et al., 2024). The functional modules under each sprint, e.g. crawler or data filter, will facilitate early feedback and minimise the risk of late failure. Agile also promotes collaboration, continuous integration, and team-based development. Nevertheless, Agile can face challenges like scope creep when requirements are not acutely managed. To counter this, sprint backlogs will be highly prioritised. In addition, the test-driven development (TDD) will be employed to verify the reliability and maintainability of the agents. Though TDD increases the effort of initial development, it mitigates long-term maintenance costs owing to early identification of errors (Ramzan et al., 2024). This is a feedback-driven, iterative process of keeping the system up to date with user needs and compliance requirements.

Challenges & Solutions

Developing an Academic Research Online agent presents several technical, ethical, and operational challenges. One key issue is data retrieval, where web scraping may encounter restrictions such as CAPTCHA, dynamic content, or anti-bot measures, potentially limiting scalability (Wahed et al., 2024). Furthermore, unfiltered retrieval risks collecting irrelevant or misleading information, which can compromise academic integrity. The system integrates natural language filtering and ontologies to address this, enabling semantic matching to improve result relevance. Ethical challenges arise with copyright and GDPR compliance. Del-Real et al. (2024) found that systems lacking privacy-by-design often face legal and operational barriers, underscoring the need for encrypted storage, anonymisation, and controlled access in the proposed agent. Mitigation involves using APIs where available and implementing secure, encrypted storage. A further challenge is balancing reactive vs. deliberative behaviour: reactive designs offer speed but lack reasoning depth, while deliberative systems provide accuracy at the cost of computational overhead (Bratman et al., 1988). The proposed hybrid BDI approach balances these trade-offs by enabling both rapid responses and structured reasoning.

Graphical Designs

Two UML models illustrate different system views. The Use Case Diagram (Figure 1) represents researcher interactions with agent modules, crawler, parser, processor, and storage, mapping functional requirements to user goals such as query submission and structured output. The Sequence Diagram (Figure 2) captures dynamic behaviour, showing the order of operations from query input through parsing, filtering, and storage. These diagrams complement each other: use cases highlight scope and roles, while sequences reveal process flow and dependencies. Alternatives such as Class Diagrams provide static structural detail, but they may underrepresent communication and temporal aspects crucial in agent systems (Polhill and Salt, 2017). By combining use case and sequence perspectives, the design balances clarity of functionality with insight into agent coordination, while acknowledging UML's abstraction limits.

Figure 1 The Use Case Diagram

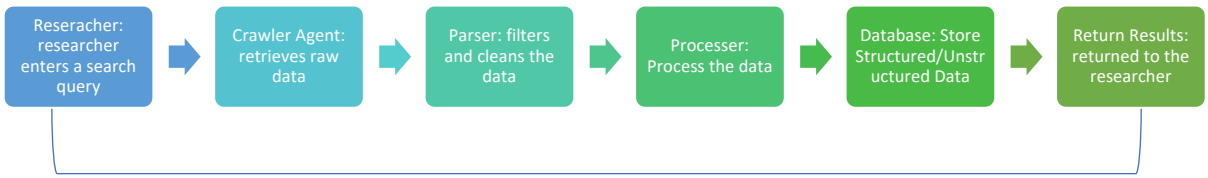

Figure 2 Sequence Diagram

Critical Evaluation of Design Choices

The adoption of a hybrid BDI architecture offers adaptability and reasoning, but it introduces complexity in implementation. (Cassenti and Kaplan, 2021) argued that reactive agents excel in speed and robustness, especially under uncertainty, while (Dlouhá and Pospíšilová, 2018) highlighted the importance of plans and deliberation for achieving long-term goals. Combining these perspectives, the chosen design balances rapid query handling with structured reasoning for relevance and quality. The dual-database strategy further reflects a trade-off: relational storage ensures consistency, while NoSQL systems offer flexibility with unstructured metadata (Shantharajah and Maruthavani, 2021). Although managing both increases overhead, it reduces the risks of performance bottlenecks. Alternatives such as single-database or fully reactive architectures would simplify the system, but at the expense of either precision or scalability. Overall, the design prioritises reliability, compliance, and researcher usability.

Conclusion

This proposal shows that an Academic Research Online agent can autonomously access, process, and present scholarly information under ethical and technical criteria. A hybrid BDI model, modular architecture, and dual-database design give the system scalability,

flexibility, and compliance. Visual representations made the functionality and coordination clear, and the critical analysis made trade-offs in agent design apparent.

References

Cassenti, D. and Kaplan, L.M. (2021). Robust uncertainty representation in human-AI collaboration. *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications III*. doi:https://doi.org/10.1117/12.2584818.

Del-Real, C., De Busser, E. and van den Berg, B. (2024). Shielding software systems: A comparison of security by design and privacy by design based on a systematic literature review. *Computer Law & Security Review*, [online] 52, p.105933. doi:https://doi.org/10.1016/j.clsr.2023.105933.

Dlouhá, J. and Pospíšilová, M. (2018). Education for Sustainable Development Goals in public debate: The importance of participatory research in reflecting and supporting the consultation process in developing a vision for Czech education. *Journal of Cleaner Production*, 172, pp.4314–4327. doi:https://doi.org/10.1016/j.jclepro.2017.06.145.

Gupta, S., Pushan Kumar Dutta, Roy, S., Aparna Bhaduri, Kalyan Maji, Choudhary, A. and Sarkar, A. (2024). Evaluating Waterfall vs. Agile Models in Software Development for Efficiency and Adaptability. *Advances in logistics, operations, and management science book series*, pp.142–148. doi:https://doi.org/10.4018/979-8-3693-3318-1.ch008.

Hussain, F., Hussain, R., Noye, B. and Sharieh, S. (2020). Enterprise API Security and GDPR Compliance: Design and Implementation Perspective. *IT Professional*, 22(5), pp.81–89. doi:https://doi.org/10.1109/mitp.2020.2973852.

Kunda, D. and Phiri, H. (2017). A Comparative Study of NoSQL and Relational Database. *Zambia ICT Journal*, 1(1), p.1. doi:https://doi.org/10.33260/zictjournal.v1i1.8.

Polhill, G. and Salt, D. (2017). The Importance of Ontological Structure: Why Validation by 'Fit-to-Data' Is Insufficient. *Understanding complex systems*, pp.141–172. doi:https://doi.org/10.1007/978-3-319-66948-9_8.

Ramzan, H.A., Ramzan, S. and Kalsum, T. (2024) February. Test-driven development (tdd) in small software development teams: Advantages and challenges. In *2024 5th International Conference on Advancements in Computational Sciences (ICACS)* (pp. 1-5). IEEE.

Saeidnia, H.R., Hosseini, E., Lund, B., Tehrani, M.A., Zaker, S. and Molaei, S. (2025). Artificial intelligence in the battle against disinformation and misinformation: a systematic review of challenges and approaches. *Knowledge and Information Systems*, 67. doi:https://doi.org/10.1007/s10115-024-02337-7.

Searle, J.R. (1969) *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press. Searle, J.R. (1969) *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press. Searle, J.R. (1969) *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press.

Shantharajah, S.P. and Maruthavani, E. (2021). A Survey on Challenges in Transforming No-SQL Data to SQL Data and Storing in Cloud Storage based on User Requirement. *International Journal of Performability Engineering*, 17(8), p.703. doi:https://doi.org/10.23940/ijpe.21.08.p6.703710.

Wahed, M.A., Alzboon, M.S., Alqaraleh, M., Ayman, J., Al-Batah, M. and Bader, A.F. (2024). Automating Web Data Collection: Challenges, Solutions, and Python-Based Strategies for Effective Web Scraping. *In 2024 7th International Conference on Internet Applications, Protocols, and Services (NETAPPS)*, [online] pp.1–6. doi:https://doi.org/10.1109/netapps63333.2024.10823528.

Wooldridge, M.J. (2009) *An Introduction to Multiagent Systems*. Chichester: John Wiley & Sons.