

aiRchive

Team 2

Aekn Admal
aekn.admal@sjtu.edu

Huy Tran
huy.q.tran03@sjtu.edu

Gonul Eda Koker
gonul.koker@sjtu.edu

Table of Contents

Table of Contents.....	1
1 Project Overview & Requirements.....	2
1.1 Project Overview.....	2
1.2 System Environment.....	2
1.3 User Types and Functional Interactions.....	3
2 Project Design.....	6
2.1 Entities, Attributes, and Relationships.....	6
2.2 ERD.....	7
2.3 Relational Schema.....	8
2.4 Persisted Table Data.....	8
3 Project Implementation.....	16
3.1 System Overview and Architecture.....	16
3.2 Backend Architecture and Design.....	17
3.3 Functional Feature Demonstration.....	18
3.4 System Setup Instructions.....	69
4 Lessons Learned.....	73

1 Project Overview & Requirements

1.1 Project Overview

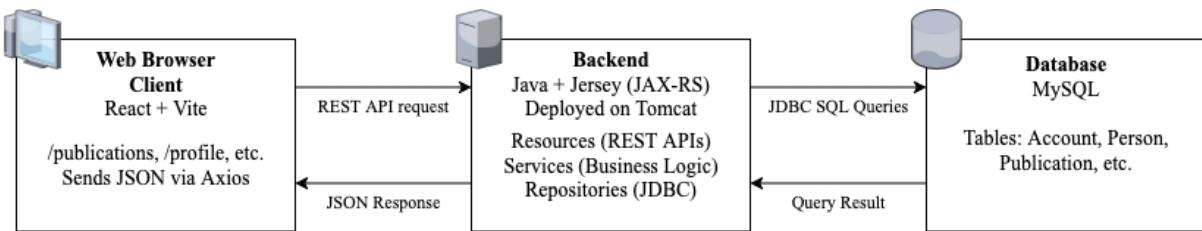
Our project, aiRchive, is a database web application inspired by platforms like arXiv and O'Reilly, designed to provide a centralized hub for curated AI publications including research papers, technical blogs, and educational articles. The key goal for our platform was to provide a personalized reading and research experience for students, industry professionals, and enthusiasts in the field.

Our motivation was to build a minimal and modern platform that not only hosts a wide range of AI content, but also provides personalized recommendations based on user interactions. The core features of the platform are its content management, curation, discovery, and recommendation systems. Alongside content curation and discovery, the platform features role-based access, an author verification system, administration functionalities, and a minimal yet modern UI.

1.2 System Environment

System Structure

[Open system structure diagram in a new page.](#)



Architecture

The application follows a three-tier architecture:

- **Frontend:** React with ShadCN UI, developed and served on Vite
- **Backend:** Java 17 with Jersey 2.35 (JAX-RS) REST API, built with Maven and deployed on Apache Tomcat 9.0.105
- **Database:** MySQL 8.0.42, accessed via JDBC using our custom repository layer

Languages and Tools Used

- **Programming Languages:** Java, JavaScript (React), SQL, HTML/CSS
- **Build Tools:** Maven, Vite
- **Database Tools:** MySQL CLI and Workbench
- **Development Tools:** IntelliJ IDEA, Eclipse, Git, Github

1.3 User Types and Functional Interactions

Guest (Unauthenticated User)

Guests are unregistered visitors. They have read-only access to publicly available content. Since they are unregistered, they will not benefit from the personalization and recommendation features supported by our application.

Key Functions and I/O:

- **Browse Publications:** Guests will be able to browse and filter publications by topics at the publication explore pages. These pages send API requests to our backend and are sent JSON response payloads containing publication metadata queried from the database.
- **Search publications:** Guests will be able to search for publications by title in the search bar component located in the navigation bar. The search parameters are sent to our backend and the results queried from the database are sent back to the client.
- **Browse Public Collections:** Guests will be able to browse public collections at the collection explore page.
- **View Publications & Collections:** Guests will be able to view individual publications or collections discovered through any of the previous methods. The client will send an API request with the publication's or collection's unique id (gathered from the publication/collection metadata received from any of the above discovery methods), and will be sent a response payload containing the publication/content data after querying from the database.

Reader (Registered User)

A *Reader* is a logged-in user who can interact with the platform's content, save items, and receive personalized recommendations based on their interaction history. Readers inherit all the functionalities that Guests have.

Key Functions and I/O:

- **Register an Account:** A Guest will be able to register an account by filling out a sign up form with several fields like username, email, password, etc. This request will be sent to the backend server which will validate the request with the database to ensure the to be registered account agrees with the database constraints. On successful validation the account will be created and a response payload containing a new session cookie will be sent to the client. A corresponding person entry will be created in the person table on successful registration. Additionally a default *Saved Publications* collection will be created for the new account.
- **Log In/Log Out:** Individuals will be able to log in using their username/email and password. Once again this request payload is sent to the server for validation and on success a session will be created and the user will be able to continue interacting with the platform through their account. Additionally, on successful login the server initiates a complete recalculation and database update of the user's affinity scores to ensure that their recommendations are up to date. The logged in user will be able to log out of their account, which will result in their session being terminated.

- **Like a Publication:** Logged in users will be able to like a publication. This action is sent to the server, which will then execute the methods needed to update the database entries corresponding to a publication's interaction history and the user's affinity scores. Users can also unlike previously liked publications which will result in a deletion of the corresponding entry in the publication's interaction table and an update to the user's affinity score.
- **Save to Collection:** Users will also be able to save publications to their collections. There are two methods of doing so: saving to a default collection and adding to a created collection. When a user saves a publication using any one of the save buttons on a publication card or page, the publication will be saved to their default *Saved Publications* collection. A user can also choose to add a publication to any one of their created collections through the *My Collections* page. Any saving of a publication will result in a recalculation of the user's affinity scores in the database. Users can also unsave a publication from any collection which will result in a deletion of the corresponding entry in the collection related tables and an update to their corresponding entries in the affinity score tables.
- **Managing Collections:** Users will be able to create, edit, and delete collections via the *My Collections* page. Users will be able to create a collection by choosing a title and setting an optional description. Users can also toggle between public or private visibility of their collection. All of this can be edited at any time by the user, and the corresponding entries in the database will reflect the changes. A user will also be able to delete a collection, which will result in a deletion of the corresponding entries in the database. Users can add publications to their collections via any of the methods outlined above.
- **Request Author Role:** Readers will be able to request an Author role upgrade. In their *My Profile* page if a Reader chooses to request an author role upgrade, a new database entry will be made in the corresponding table. The request will have a pending status until it is approved by an administrator. Once approved the database entries corresponding to the request's status will be changed to approved and the user's account will be updated to have the Author role.

Author (Privileged Reader)

An *Author* is a privileged Reader who has been approved to submit and publish publications. They can manage their publications by creating and editing drafts and their metadata, as well as publishing their drafts onto the platform. Authors inherit all the functionalities that Readers have.

Key Functions and I/O:

- **Manage Drafts:** Authors will be able to create drafts in the *My Publications* page. This is similar to how creating collections works. The user will be able to choose the publication type (paper, blog, article) and then input the appropriate fields such as title, page content, doi and pdf urls (paper only). This is sent to the backend server and upon successful validation of fields, a new database entry in the publication table will be made, with the publication status being set to draft. A drafted publication can

be edited as many times as the Author wishes and the changes will be reflected in the database.

- **Publish Drafts:** An Author can publish their draft through the *My Publications* page once they believe their draft is ready to be published. When choosing to publish a draft, the Author must list the authors of the publication. They can search and add authors based on their email (as it is stored in the database). If the author does not exist yet in the system, then they can choose to create a new person entity for that author. This is similar to how many other publication platforms work, to ensure that publications can be credited to their authors even if they do not have an account with the platform. After listing the authors, the submitter can choose up to three topics to associate their publication with. The topics and authors associated with the publication will be used for calculating affinity scores for users. The submitter can choose an optional publish date, and if none is provided the submission date will be used. After completing the form a publish request will be sent to the backend server, and upon successful validation the appropriate database entries will be made in tables like `publication_author`, `publication_topic`, etc. Once published a publication cannot and should not be edited or deleted.

Administrator (Privileged User)

An *Administrator* is any registered user who has their administrator privilege set to true in the database. Administrators can approve author role upgrade requests and manage topics on the platform. Administrators can be either a Reader or Author, and thus will inherit the appropriate functionalities.

Key Functions and I/O:

- **Approve Author Requests:** An Administrator can see and approve any author requests in their *My Administration* page. If they choose to approve an author request the `author_request` table in the database will update to an approved status and the corresponding account's role will be updated to author.
- **Manage Topics:** An Administrator can create new topics and delete existing topics in their *My Administration* page. An Administrator can create a topic with a short code and its full name. Upon successful creation of a topic a new database entry will be made in the `topic` table with the corresponding topic code and full name values. They can also delete topics which will result in a complete deletion of the topic from all tables using the entry.

2 Project Design

2.1 Entities, Attributes, and Relationships

Person

Represents a real-world identity, such as an author or user. This entity stores the core identity information of a person and can exist independently of an account.

Account

Represents a user account registered on the platform. Each account is linked to a person and includes the account's credentials and privileges.

Author Request

This entity tracks upgrade requests submitted by reader level users who want to become authors.

Publication

Represents a single piece of content submitted to the platform, including research papers, blogs, and articles.

Publication Author

Represents the many-to-many relationship between publications and their authors. All authors are person entities.

Publication Topic

Represents the many-to-many relationship between publications and their associated topics.

Publication Like

Represents a many-to-many relationship between publications and accounts who have liked the publication.

Publication View

Represents a many-to-many relationship between publications and accounts who have viewed the publication.

Collection

Represents a curated list of publications saved by an account. Collections may be private or public.

Collection Item

Represents the many-to-many relationship between collections and contained publications.

Topic

Represents a category relevant to AI, and is used to classify publications and build personalized recommendation features.

Topic Affinity

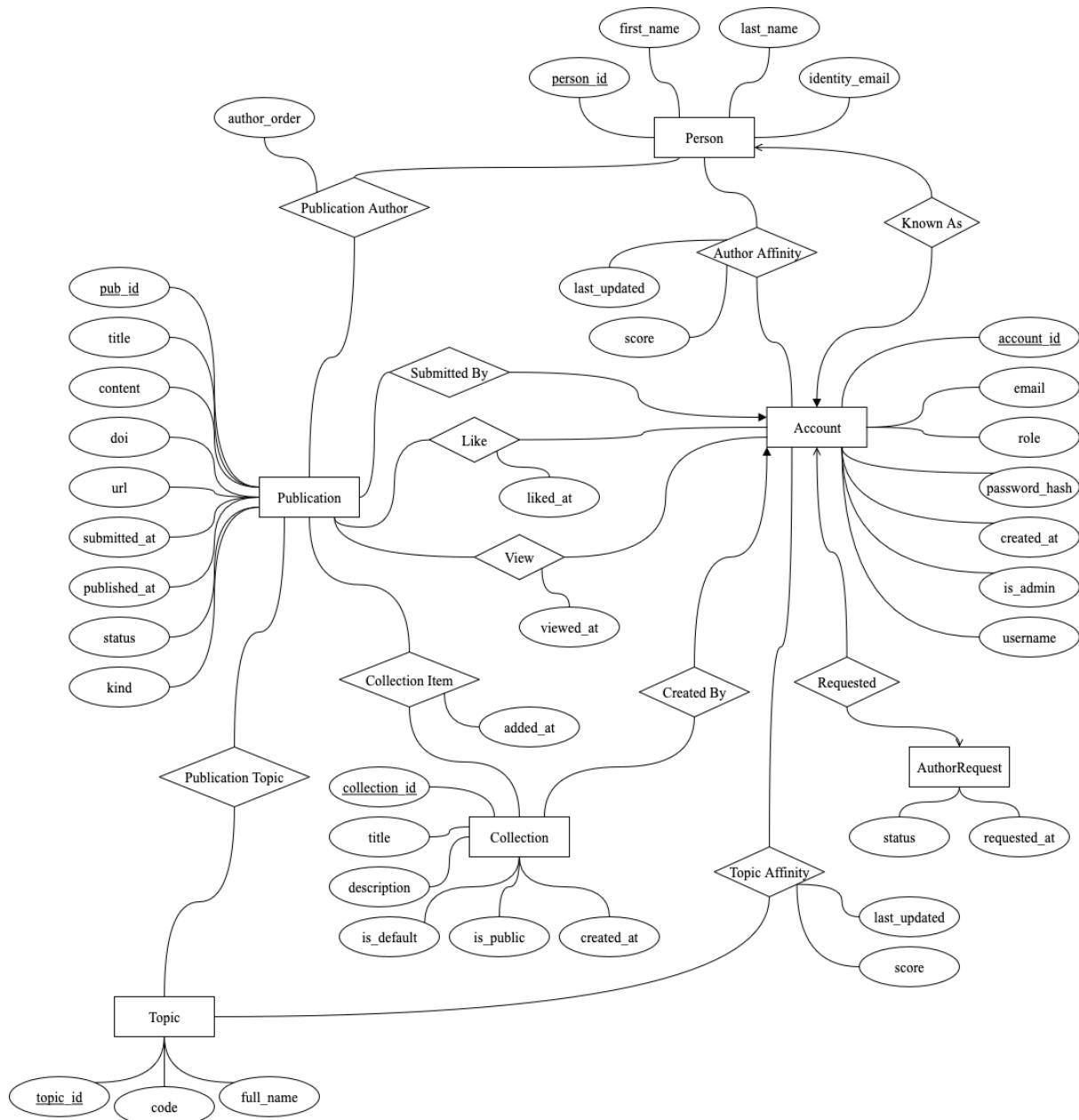
Represents a many-to-many relationship between an account and a topic. This relationship is calculated from interaction history, such as views, likes, and saves. This relationship stores a score representing an account's interest in a topic.

Author Affinity

Represents a many-to-many relationship between an account and an author (person). This relationship is calculated from interaction history, such as views, likes, and saves. This relationship stores a score representing an account's interest in an author.

2.2 ERD

[Open ERD in a new page.](#)



2.3 Relational Schema

```

person(person_id, first_name, last_name, identity_email)
account(account_id, person_id, email, username, password_hash, role, is_admin, created_at)
author_request(account_id, status, requested_at)
publication(pub_id, title, content, doi, url, kind, submitter_id, submitted_at, published_at,
status)
publication_author(pub_id, person_id, author_order)
topic(topic_id, code, full_name)
publication_topic(pub_id, topic_id)
publication_like(account_id, pub_id, liked_at)
publication_view(account_id, pub_id, viewed_at)
collection(collection_id, account_id, title, description, is_default, is_public, created_at)
collection_item(collection_id, pub_id, added_at)
topic_affinity(account_id, topic_id, score, last_updated)
author_affinity(account_id, author_id, score, last_updated)

```

2.4 Persisted Table Data

person Table Data

[Open screenshot in a new page.](#)

Result Grid			
person_id	first_name	last_name	identity_email
58	bla	bla	blabla2@gmail.com
57	waylon	jennings	whatwouldwaylondo@gmail....
56	Ozzy	Osbourne	ozzyluvsa@gmail.com
55	random	person	randomperson@gmail.com
54	new	user	newuse234234234234@gm...
53	new	user	newuse2222r@gmail.com
52	user	user	user@gmail.com
51	new	user	newuser@gmail.com
50	ash	ash	ash@gmail.com
49	asdfasdfsdf	asd	assd@gmail.com
48	aekn	admal	12341234@gmail.com
47	Some	Wonky...	wonky.dude@airchive.com
46	Stephan	ten Brink	stephan.tenbrink@airchive.c...
45	Florian	Euchner	florian.euchner@airchive.com
44	Phillip	Stephan	phillip.stephan@airchive.com
43	Lujia	Pan	lujia.pan@airchive.com
42	Ruichu	Cai	ruichu.cai@airchive.com
41	Keli	Zhang	keli.zhang@airchive.com
40	Lei	Zan	lei.zan@airchive.com
39	five	user	user200@gmail.com
38	Antonello	Rizzi	antonello.rizzi@airchive.com
37	Enrico	De Santis	enrico.desantis@airchive.com
36	Shubha...	Borse	shubhankar.borse@airchive...
35	Debasmit	Das	debasmit.das@airchive.com
34	Amir	Said	amir.said@airchive.com
33	Alexei	Bourd	alexei.bourd@airchive.com

account Table Data

[Open screenshot in a new page.](#)

Result Grid							
account_id	person_id	email	username	password_hash	role	is_admin	created_at
29	58	blabla2@gmail.com	blabla2	\$2a\$12\$EbsaEGuUWiY6wf...	READER	0	2025-08-09 05:10:39
28	57	whatwouldwaylondo@gmail....	whatwouldwaylondo	\$2a\$12\$PwXDAEMNzcf1H...	READER	0	2025-08-08 20:32:22
27	56	ozzyluvssai@gmail.com	ozzyluvssai	\$2a\$12\$SPsPTiSj0BfnSY10...	READER	0	2025-08-05 22:32:37
26	54	newuse234234234234@gm...	newuser15	\$2a\$12\$zOfs9LjH5imCf2k6...	READER	0	2025-08-05 13:31:42
25	53	newuse2222r@gmail.com	newuser2222	\$2a\$12\$se Tg0Nd19ia07L...	AUTHOR	0	2025-08-05 13:29:04
24	52	user@gmail.com	user	\$2a\$12\$ziEIWJyutUXOVVe...	READER	0	2025-08-05 13:25:23
23	51	newuser@gmail.com	newuser111	\$2a\$12\$pdqCty38v9Cd.V...	READER	0	2025-08-05 13:09:14
22	50	ash@gmail.com	ashash	\$2a\$12\$swtEVJ5PNwNBja...	READER	0	2025-08-05 12:15:08
21	49	assd@gmail.com	asd	\$2a\$12\$.J73aSJ1iBc/g0kpV...	AUTHOR	0	2025-08-05 11:59:31
20	48	12341234@gmail.com	123412341234	\$2a\$12\$BOPm1O0Cs/REW...	READER	0	2025-08-05 05:51:14
19	39	user200@gmail.com	user200	\$2a\$12\$w3lkXgRZLedyhW...	READER	0	2025-08-05 04:31:34
18	26	publisher2@airchive.com	publisher2	\$2a\$12\$XDomxTHIJnI47vg...	AUTHOR	0	2025-08-05 03:18:20
17	25	admin@gmail.com	admin	\$2a\$12\$wp7xtlmfUkwQl6xr...	AUTHOR	1	2025-08-04 19:29:07
16	24	asdfsdlkjf@gmail.com	asdfasdfadsf	\$2a\$12\$YYvxIBF0APlc2Xqv...	AUTHOR	0	2025-08-04 19:27:42
15	23	asdf@gmail.com	asdfasdf	\$2a\$12\$bvfFCyt1lzKpzQqL...	AUTHOR	0	2025-08-04 19:22:47
14	22	tommylee@gmail.com	mcrue	\$2a\$12\$zC0/ZZXs7LKFUC...	AUTHOR	0	2025-08-04 00:54:26
13	21	ben10@gmail.com	ben10	\$2a\$12\$oRBwnSwj3wfBisY...	READER	0	2025-08-04 00:19:04
12	20	tenthduke@gmail.com	tenthduke	\$2a\$12\$JHtlc3ffFvYHsyvk...	READER	0	2025-08-03 22:47:24
11	19	thing10@gmail.com	thing10	\$2a\$12\$qUdbfJDgEspy8hs...	READER	0	2025-08-03 19:32:01
10	14	newdude@gmail.com	newdude	\$2a\$12\$0017u5uYLPCAkr...	AUTHOR	0	2025-08-03 17:23:33
9	9	thing3@gmail.com	thing3	\$2a\$12\$4cQ94ufdnUE2Ec...	READER	0	2025-08-02 23:14:10
8	8	thing2@gmail.com	thing2	\$2a\$12\$9pIG.80.kAaOYGT...	READER	0	2025-08-02 22:57:01
7	7	user4@gmail.com	user4	\$2a\$12\$ptzuJ4VyacbC4Mjs...	AUTHOR	0	2025-08-02 22:51:21
6	6	user3@gmail.com	user3	\$2a\$12\$NIBGzE9TGeTNDhi...	AUTHOR	0	2025-08-02 22:48:40
5	5	user2@gmail.com	user2	\$2a\$12\$hGg4s0q1/hdG0UE...	READER	0	2025-08-02 22:47:28
4	4	anotheremail@gmail.com	anotherguy	\$2a\$12\$8oqpGGp4rrZL8rm...	READER	0	2025-08-01 19:50:05

author_request Table Data

[Open screenshot in a new page.](#)

Result Grid		
account_id	status	requested_at
29	PENDING	2025-08-09 05:13:33
25	APPROVED	2025-08-05 13:30:58
24	PENDING	2025-08-05 13:26:40
22	PENDING	2025-08-05 12:15:12
21	APPROVED	2025-08-05 12:00:00
18	APPROVED	2025-08-05 03:57:33
17	APPROVED	2025-08-05 03:52:38
16	APPROVED	2025-08-04 19:28:30
15	APPROVED	2025-08-04 09:23:40
14	APPROVED	2025-08-04 07:04:02
10	APPROVED	2025-08-03 17:24:02
9	PENDING	2025-08-02 23:14:56
8	PENDING	2025-08-02 22:57:09
7	APPROVED	2025-08-02 22:51:26
6	APPROVED	2025-08-02 22:48:59
5	PENDING	2025-08-02 22:47:32
1	APPROVED	2025-08-01 17:01:04
HULL	HULL	HULL

publication Table Data

[Open screenshot in a new page.](#)

pub_id	title	content	doi	url	kind	submitter_id	submitted_at	published_at	status
19	link for erd	<p>link for erd	NULL	NULL	BLOG	1	2025-08-09 18:24:22	NULL	DRAFT
18	aiRchive: Final Report	Table of Content	NULL	NULL	ARTICLE	1	2025-08-09 17:02:30	NULL	DRAFT
17	random blog	<pre class="hljs"><code>as...</code></pre>	NULL	NULL	BLOG	1	2025-08-05 13:33:41	2025-08-03 07:00:00	PUBLISHED
16	A Blog About Nothing	<h2>nothing</h2><p>...</p>	NULL	NULL	BLOG	18	2025-08-05 04:45:53	2025-04-08 07:00:00	PUBLISHED
15	CSI Obfuscation: Single-Ant...	<p><code>The ability of modern tel...</code></p>	https://doi.org/10.48550/arXiv.2508.02...	https://arxiv.org/pdf/2508.02...	PAPER	18	2025-08-05 04:41:34	2025-08-04 07:00:00	PUBLISHED
14	CAMA: Enhancing Mathema...	<p><code>Large Language Models...</code></p>	https://doi.org/10.48550/arXiv.2508.02...	https://arxiv.org/pdf/2508.02...	PAPER	18	2025-08-05 04:37:06	2025-08-04 07:00:00	PUBLISHED
13	Noosemia: toward a Cogniti...	<p><code>This paper introduces a...</code></p>	https://doi.org/10.48550/arXiv.2508.02...	https://arxiv.org/pdf/2508.02...	PAPER	18	2025-08-05 04:13:30	2025-08-04 07:00:00	PUBLISHED
12	LoRA-X: Bridging Foundatio...	<p><code>The rising popularity of ...</code></p>	https://doi.org/10.48550/arXiv.2501.16...	https://arxiv.org/pdf/2501.16...	PAPER	18	2025-08-05 04:05:29	2025-02-04 08:00:00	PUBLISHED
11	Sort-free Gaussian Splatting...	<p><code>Recently, 3D Gaussian...</code></p>	https://doi.org/10.48550/arXiv.2410.18...	https://arxiv.org/pdf/2410.18...	PAPER	18	2025-08-05 04:02:38	2025-04-09 07:00:00	PUBLISHED
10	system architecture link	<p><code>system architecture link</code></p>	NULL	NULL	BLOG	1	2025-08-09 17:02:55	NULL	DRAFT
9	example	<p><code><sd></code></p>	NULL	NULL	ARTICLE	1	2025-08-05 01:56:09	NULL	DRAFT
8	asdf	<p><code>asdfasdf!</code></p>	asdf	NULL	PAPER	1	2025-08-05 01:52:18	NULL	DRAFT
7	Trustworthy Distributed AI S...	<p><code>Emerging Distributed AI ...</code></p>	https://doi.org/10.1145/3645...	https://dl.acm.org/doi/pdf/10...	PAPER	1	2025-08-03 19:06:16	2025-02-10 08:00:00	PUBLISHED
6	Composable OS Kernel Arc...	<p><code>As intelligent systems p...</code></p>	https://doi.org/10.48550/arXiv.2508.00...	https://arxiv.org/pdf/2508.00...	PAPER	1	2025-08-03 19:00:59	2025-08-01 07:00:00	PUBLISHED
5	bla bla bla	<p><code>Python Developers, wan...	NULL	NULL	ARTICLE	1	2025-08-03 17:20:48	NULL	DRAFT
4	buncha random stuff	<p><code>where you can write stuf...	NULL	NULL	ARTICLE	1	2025-08-03 08:59:02	NULL	DRAFT
3	Our Schema	<p><code>I really hope this doesn't...</code></p>	NULL	NULL	BLOG	1	2025-08-03 08:32:48	2025-08-03 07:00:00	PUBLISHED
2	Implementing MCP Servers...	<p><code>Python Developers, wan...	NULL	NULL	ARTICLE	1	2025-08-03 06:33:16	2025-08-03 06:33:16	PUBLISHED
1	Spatio-Causal Patterns of S...	<p><code>Different statistical samp...	https://doi.org/10.1613/jair.1...	https://www.jair.org/index.ph...	PAPER	1	2025-08-03 05:46:26	2025-07-31 07:00:00	PUBLISHED
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

publication_author Table Data

[Open screenshot in a new page.](#)

pub_id	person_id	author_order
17	55	2
17	1	1
16	47	1
15	46	3
15	45	2
15	44	1
14	43	4
14	42	3
14	41	2
14	40	1
NULL	NULL	NULL

topic Table Data

[Open screenshot in a new page.](#)

1 • SELECT * FROM airchive.topic LIMIT 10;		
Result Grid		
topic_id	code	full_name
2	NLP	Natural Language Processing
3	CV	Computer Vision
4	ML	Machine Learning
5	DL	Deep Learning
6	RL	Reinforcement Learning
7	LLM	Large Language Models
8	XAI	Explainable AI
9	ETH	AI Ethics and Fairness
10	KG	Knowledge Graphs
11	IR	Information Retrieval
NULL	NULL	NULL

publication_topic Table Data

[Open screenshot in a new page.](#)

1 • SELECT * FROM airchive.publication_topic ORDER BY pub_id DESC LIMIT 10;	
Result Grid	
pub_id	topic_id
17	35
17	34
17	26
16	36
16	21
16	2
15	46
15	45
15	4
14	37

publication_like Table Data

[Open screenshot in a new page.](#)

1 • `SELECT * FROM airchive.publication_like ORDER BY liked_at DESC LIMIT 10;`

Result Grid | Filter Rows: | Search | Edit: | Export/Import: | Fetch rows: |

account_id	pub_id	liked_at
18	2	2025-08-08 20:00:41
1	16	2025-08-05 23:08:41
27	13	2025-08-05 23:00:54
27	17	2025-08-05 23:00:49
27	6	2025-08-05 23:00:45
27	16	2025-08-05 22:59:58
27	14	2025-08-05 22:34:00
24	7	2025-08-05 13:26:22
24	2	2025-08-05 13:26:09
23	15	2025-08-05 13:10:17
NULL	NULL	NULL

publication_view Table Data

[Open screenshot in a new page.](#)

1 • `SELECT * FROM airchive.publication_view ORDER BY viewed_at DESC LIMIT 10;`

Result Grid | Filter Rows: | Search | Edit: | Export/Import: | Fetch rows: |

account_id	pub_id	viewed_at
1	3	2025-08-09 05:27:57
1	16	2025-08-09 05:10:15
1	12	2025-08-09 05:10:02
1	2	2025-08-09 05:09:52
28	3	2025-08-08 20:34:33
1	2	2025-08-08 20:31:19
1	7	2025-08-08 20:31:16
1	12	2025-08-08 20:19:14
18	2	2025-08-08 20:00:17
18	17	2025-08-08 20:00:12
NULL	NULL	NULL

collection Table Data

[Open screenshot in a new page.](#)

Result Grid							
collection_id	account_id	title	description	is_default	is_public	created_at	
43	29	bla bla's first collection	<p>This is my first collection...	0	1	2025-08-09 05:13:09	
42	29	Saved	Your saved publications	1	0	2025-08-09 05:10:40	
41	28	Saved	Your saved publications	1	0	2025-08-08 20:32:22	
40	27	Saved	Your saved publications	1	0	2025-08-05 22:32:37	
38	26	Saved	Your saved publications	1	0	2025-08-05 13:31:42	
37	25	Saved	Your saved publications	1	0	2025-08-05 13:29:04	
36	24	Saved	Your saved publications	1	0	2025-08-05 13:25:23	
35	23	Saved	Your saved publications	1	0	2025-08-05 13:09:14	
34	22	Saved	Your saved publications	1	0	2025-08-05 12:15:08	
33	18	random title	<p>asdfa</p><p>sdf</p><p>...	0	0	2025-08-05 12:11:38	
NULL	NULL	NULL	NULL	NULL	NULL	NULL	

collection_item Table Data

[Open screenshot in a new page.](#)

Result Grid		
collection_id	pub_id	added_at
43	16	2025-08-09 05:13:09
42	15	2025-08-09 05:11:09
42	16	2025-08-09 05:10:54
1	16	2025-08-09 05:10:11
41	14	2025-08-08 20:33:48
41	17	2025-08-08 20:32:56
1	1	2025-08-05 23:13:46
1	6	2025-08-05 23:06:56
1	7	2025-08-05 23:06:52
1	15	2025-08-05 23:06:51
NULL	NULL	NULL

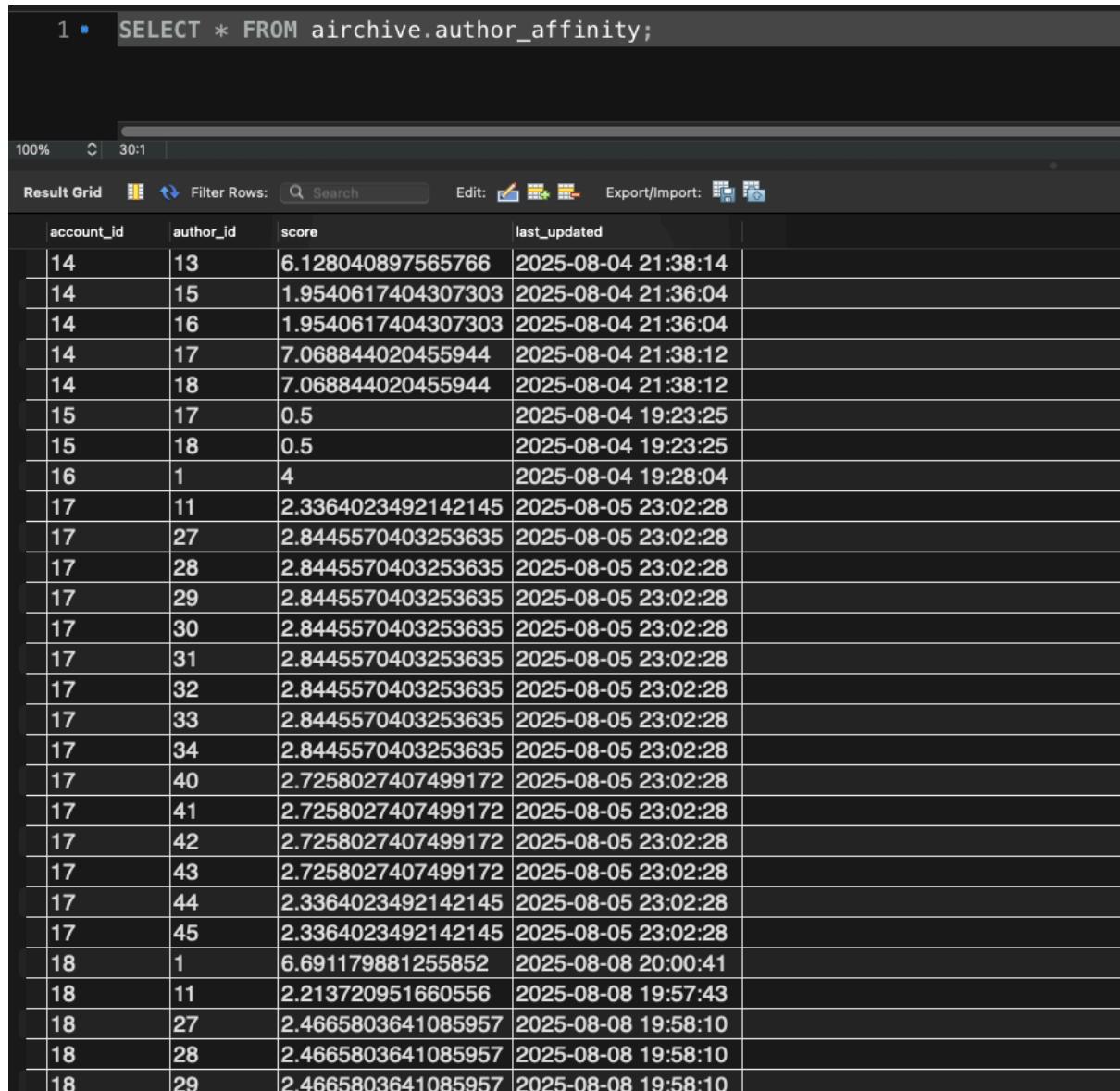
topic_affinity Table Data

[Open screenshot in a new page.](#)

1 • SELECT * FROM airarchive.topic_affinity;				
Result Grid				
account_id	topic_id	score	last_updated	
14	34	3.4557254531396597	2025-08-04 21:38:18	
14	36	1.9540617404307303	2025-08-04 21:36:03	
14	37	1.9540617404307303	2025-08-04 21:36:03	
14	38	7.068844020455944	2025-08-04 21:38:12	
14	39	7.068844020455944	2025-08-04 21:38:12	
14	40	7.068844020455944	2025-08-04 21:38:12	
15	38	0.5	2025-08-04 19:23:25	
15	39	0.5	2025-08-04 19:23:25	
15	40	0.5	2025-08-04 19:23:25	
16	25	4	2025-08-04 19:28:04	
17	2	0.38940039153570...	2025-08-05 23:02:28	
17	3	2.8445570403253635	2025-08-05 23:02:28	
17	4	7.398607439178347	2025-08-05 23:02:28	
17	21	0.38940039153570...	2025-08-05 23:02:28	
17	33	2.3364023492142145	2025-08-05 23:02:28	
17	34	2.7258027407499172	2025-08-05 23:02:28	
17	35	2.7258027407499172	2025-08-05 23:02:28	
17	36	1.1316954846787557	2025-08-05 23:02:28	
17	37	3.8574982254286727	2025-08-05 23:02:28	
17	38	0.8245627548392087	2025-08-05 23:02:28	
17	39	0.8245627548392087	2025-08-05 23:02:28	
17	40	0.8245627548392087	2025-08-05 23:02:28	
17	43	0.38940039153570...	2025-08-05 23:02:28	
17	45	2.3364023492142145	2025-08-05 23:02:28	
17	46	2.3364023492142145	2025-08-05 23:02:28	
18	2	2.4976509274553855	2025-08-08 20:00:08	

author_affinity Table Data

[Open screenshot in a new page.](#)



The screenshot shows a MySQL Workbench interface with a result grid displaying the data from the `airchive.author_affinity` table. The table has four columns: `account_id`, `author_id`, `score`, and `last_updated`. The data consists of approximately 30 rows, each representing a unique combination of account and author IDs with their corresponding score and update timestamp.

account_id	author_id	score	last_updated
14	13	6.128040897565766	2025-08-04 21:38:14
14	15	1.9540617404307303	2025-08-04 21:36:04
14	16	1.9540617404307303	2025-08-04 21:36:04
14	17	7.068844020455944	2025-08-04 21:38:12
14	18	7.068844020455944	2025-08-04 21:38:12
15	17	0.5	2025-08-04 19:23:25
15	18	0.5	2025-08-04 19:23:25
16	1	4	2025-08-04 19:28:04
17	11	2.3364023492142145	2025-08-05 23:02:28
17	27	2.8445570403253635	2025-08-05 23:02:28
17	28	2.8445570403253635	2025-08-05 23:02:28
17	29	2.8445570403253635	2025-08-05 23:02:28
17	30	2.8445570403253635	2025-08-05 23:02:28
17	31	2.8445570403253635	2025-08-05 23:02:28
17	32	2.8445570403253635	2025-08-05 23:02:28
17	33	2.8445570403253635	2025-08-05 23:02:28
17	34	2.8445570403253635	2025-08-05 23:02:28
17	40	2.7258027407499172	2025-08-05 23:02:28
17	41	2.7258027407499172	2025-08-05 23:02:28
17	42	2.7258027407499172	2025-08-05 23:02:28
17	43	2.7258027407499172	2025-08-05 23:02:28
17	44	2.3364023492142145	2025-08-05 23:02:28
17	45	2.3364023492142145	2025-08-05 23:02:28
18	1	6.691179881255852	2025-08-08 20:00:41
18	11	2.213720951660556	2025-08-08 19:57:43
18	27	2.4665803641085957	2025-08-08 19:58:10
18	28	2.4665803641085957	2025-08-08 19:58:10
18	29	2.4665803641085957	2025-08-08 19:58:10

3 Project Implementation

This section details the full-stack implementation of our aiRchive application. Here we will explain the architectural decisions, code structure, key features, REST API design, SQL integration, recommendation engine, and system deployment. The implementation maps to the functional requirements outlined earlier.

3.1 System Overview and Architecture

We implemented aiRchive as a three-tier application (as we briefly mentioned above) consisting of a React frontend, a Java-based REST API backend, and a MySQL relational database.

Frontend: React with ShadCN UI and Vite

We initially used JSP and Servlets to render HTML and handle our sign in and sign up form submissions. This was fine for prototyping our basic pages, but it quickly became a limiting factor once we started implementing more features, such as: filtering publications by topics, conditionally displaying components based on user roles, and managing dynamic page content such as view and like counts.

In my opinion, the biggest inconvenience was trying to show author-only or admin-only components on a page. To do this we had to create separate JSP files and servlet endpoints, with basically the same exact page content plus a few extra components needed only for specific roles. Also, we had to use stuff like AJAX to avoid reloading our pages whenever we wanted to rerender with new content.

As a result, we decided to transition to [React](#), which is a framework literally designed for building dynamic and interactive web applications. Since React uses a component-based model it allowed us to manage our UI with less work as we could reuse layouts and components across our pages. Also since React is designed to build single-page applications, it lets us update content without having to reload the entire page.

After switching to React, we restructured our backend to a dedicated REST API, which allowed us to separate client and server responsibilities such as fetching and rendering data.

Backend: Java 17 with Jersey 2.35 (JAX-RS) and Apache Tomcat

We implemented our backend in Java 17 and using [Jersey 2.35 \(JAX-RS\)](#) for our [RESTful APIs](#). We use Apache Tomcat 9.0.105 as the servlet container for deployment. After transitioning to React, we redesigned our backend from JSP and Servlets to a REST API handling business logic and data access.

Our backend follows a three-layer architecture:

- Repository Layer: All SQL logic is implemented here using JDBC. Each table has a corresponding repository class that executes queries and maps results to Java records.
- Service Layer: This is the layer where we handle all business logic such as validation, role-based access control, error handling, and data consistency. We designed this to be the “middleman” between our resource controllers and repositories.

- Resource Layer: This is where we explore our REST API to the client through JAX-RS annotations. Each endpoint validates session status when necessary, receives and maps JSON payloads to records, and delegates tasks to the appropriate service methods.

Database: MySQL 8.0.42

We use MySQL 8.0.42 as our relational database. We iteratively refined our schema, throughout the semester, such that all relations are normalized to 3NF and BCNF, where applicable. All SQL queries were manually written in the repository layer. We also added indexes to several columns to optimize queries and we also added a FULLTEXT index to support [natural-language full text searches](#) for publication titles and topic names. We also integrated [HikariCP](#) as a connection pool provider. By default, our application uses DriverManager to create connections, but connection pooling can be conditionally enabled by setting the `db.usePool` property in the database properties file. Connection pooling is extreme overkill for our demo, but it felt nice to have.

3.2 Backend Architecture and Design

We implemented our backend using the well-known Repository-Service-Resource pattern. With this approach we were able to separate logic concerns and easily add new features. As we outlined earlier each layer is responsible for a specific core aspect of the system.

Repository Layer (Data Access)

As outlined previously, our repository layer is responsible for all SQL interactions with the MySQL database. Each table has a dedicated repository class (e.g. `AccountRepository`, `PublicationRepository`) that contains SQL queries and logic to execute prepared statements, map `ResultSets` to Java records, and to handle data access exceptions like `SQLException`. With our repository layer we were able to keep all SQL-related logic out of our service and resource layers.

Service Layer (Business Logic)

Our service layer implements all of the core application logic. It validates inputs, enforces role-based access control, wraps exceptions, and coordinates operations that involve multiple tables and steps. All of our use case flows such as publishing a draft, liking a publication, getting recommended publications are handled in these classes. All our role-based access control logic is enforced here via our `SecurityUtils` class. This layer is also where we take our results from the repository layer and reorganize them in a way that our client expects.

Resource Layer (REST API)

The resource layer handles all HTTP requests using JAX-RS annotations like `@Path`, `@Get`, `@Post`, `@Produces`, etc. We designed our resource layer such that each class acts as a controller for a specific domain. For example our `PublicationResource` class handles

all API endpoints regarding publications. Each method in our resource classes extract input from the request body or request parameters, validates the requesting session, delegates logic to the service layer, and responds with our JSON-parsed Java records. We have a package exclusively for these DTO (Data Transfer Objects) records, so that we could define standardized request and response bodies to work with.

Session Authentication

Finally, all user sessions are handled using `HttpSessions`. When a user logs in or creates an account, a session is created on our server. This session stores the user's account id, role, and some other identity information necessary for identification and validation. The server then sends a `Set-Cookie` header to the client containing a `JSESSIONID` cookie. This cookie is then sent by the browser in all following requests to our server. This way we were able to appropriately deny or accept requests based on their privileges. To ensure that the session information is up to date we decided to create an endpoint which the client calls every once in a while which kicks off a refreshing of their session context.

3.3 Functional Feature Demonstration

Sorry about the unorderedness of the following pages, the numerous screenshots do not format well on Google Docs.

Account Creation

- Create an account page.

The screenshot shows a web browser window with the URL <http://localhost:5173/signup>. The page is titled "aiRchive" and features a dark-themed "Create an Account" form. The form includes fields for First Name (Filled: Final), Last Name (Filled: Report), Username (Filled: finalreporttest), Email (Filled: finalreporttest@gmail.com), and Password (Filled with dots). A "Sign Up" button is at the bottom of the form. At the bottom of the page, there is a link "Already a user? Login". The browser interface shows standard controls like back, forward, and search, along with a "Sign Up" button in the top right corner of the header.

- POST Request.

Developer Tools — airChive — http://localhost:5173/

Status Method Domain File Initiator Type Transferred Size Headers Cookies Request Response Timings Stack Trace

200	GET	localhost...	me	axios.js:1672 (xhr)	xml	108 B	0 B
200	GET	localhost...	recommendations?page=	axios.js:1672 (xhr)	json	2.23 kB	2.08 kB
200	GET	localhost...	platform	axios.js:1672 (xhr)	json	183 B	42 B
200	POST	localhost...	register	axios.js:1672 (xhr)	json	446 B	153 B
200	GET	localhost...	recommendations?page=	axios.js:1672 (xhr)	json	2.23 kB	2.08 kB
200	GET	localhost...	platform	axios.js:1672 (xhr)	json	183 B	42 B
200	GET	localhost...	stats	axios.js:1672 (xhr)	json	171 B (raced)	30 B
200	GET	localhost...	interactions?limit=3	axios.js:1672 (xhr)	json	142 B (raced)	2 B
200	GET	localhost...	2	axios.js:1672 (xhr)	json	156 B	15 B
200	GET	localhost...	7	axios.js:1672 (xhr)	json	156 B (raced)	15 B
200	GET	localhost...	1	axios.js:1672 (xhr)	json	156 B (raced)	15 B
200	GET	localhost...	16	axios.js:1672 (xhr)	json	156 B	15 B
200	GET	localhost...	6	axios.js:1672 (xhr)	json	156 B (raced)	15 B
200	GET	localhost...	3	axios.js:1672 (xhr)	json	156 B	15 B

POST http://localhost:5173/api/auth/register

Status: 200 OK

Headers:

- Access-Control-Allow-Origin: http://localhost:5173
- Connection: close
- Content-Length: 153
- Content-Type: application/json
- Date: Sun, 10 Aug 2025 04:15:25 GMT
- Set-Cookie: SESSIONID=1FBBB348337E8D6C90268675EC38D65;Version=1;Path=/;Max-Age=2592000;HttpOnly
- Vary: Origin

Request Headers (936 B):

- Accept: application/json, text/plain, */*
- Accept-Encoding: gzip, deflate, br, zstd
- Accept-Language: en-US,en;q=0.5
- Connection: keep-alive
- Content-Length: 128
- Content-Type: application/json
- Cookie: username=localhost-8888-*310010175251147323;username=localhost-8888192c912c9ybmftzS1GIC9Zm1MGU2NWJ0DE0MmFjOGZizDcwZjMyOTkwZGU4NyjICJuYV1lj9ogKfub255bw91cy8NbmvZSlstCjkaXnbwDF5X25hbWUjOiaQW5vbnlbt3VsE1uZW1llwgImIusXRpWxzrjgkPNwiqimNvbGsyjognvsh0=jf19994b8642b47d6tc6c89ddfe1af740116154102454bc7fb68d4907f95b";_xrf=2|23037628|ddc8cd807ba5e42a6fbe3eb2b6ef7741762511473
- DNT: 1
- Host: localhost:5173
- Origin: http://localhost:5173
- Priority: u=0
- Referer: http://localhost:5173/signup
- Sec-Fetch-Dest: empty
- Sec-Fetch-Mode: cors
- Sec-Fetch-Site: same-origin
- Sec-GPC: 1
- User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:141.0) Gecko/20100101 Firefox/141.0

Request Headers (Raw):

```
Accept: application/json, text/plain, */*
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Content-Length: 128
Content-Type: application/json
Cookie: username=localhost-8888-*310010175251147323;username=localhost-8888192c912c9ybmftzS1GIC9Zm1MGU2NWJ0DE0MmFjOGZizDcwZjMyOTkwZGU4NyjICJuYV1lj9ogKfub255bw91cy8NbmvZSlstCjkaXnbwDF5X25hbWUjOiaQW5vbnlbt3VsE1uZW1llwgImIusXRpWxzrjgkPNwiqimNvbGsyjognvsh0=jf19994b8642b47d6tc6c89ddfe1af740116154102454bc7fb68d4907f95b";_xrf=2|23037628|ddc8cd807ba5e42a6fbe3eb2b6ef7741762511473
DNT: 1
Host: localhost:5173
Origin: http://localhost:5173
Priority: u=0
Referer: http://localhost:5173/signup
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Sec-GPC: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:141.0) Gecko/20100101 Firefox/141.0
```

DOMContentLoaded: 199 ms Load: 2.13 s

- Newly created account and person entry.

Result Grid Filter Rows: Search Export:

account_id	person_id	email	username	password_hash	role	is_admin	created_at	person_id	first_name	last_name	identity_email
31	60	finalreporttest@gmail.com	finalreporttest	\$2a\$12\$BmkVkJcB7VasiS...	READER	0	2025-08-09 21:15:25	60	Final	Report	finalreporttest@gmail.com

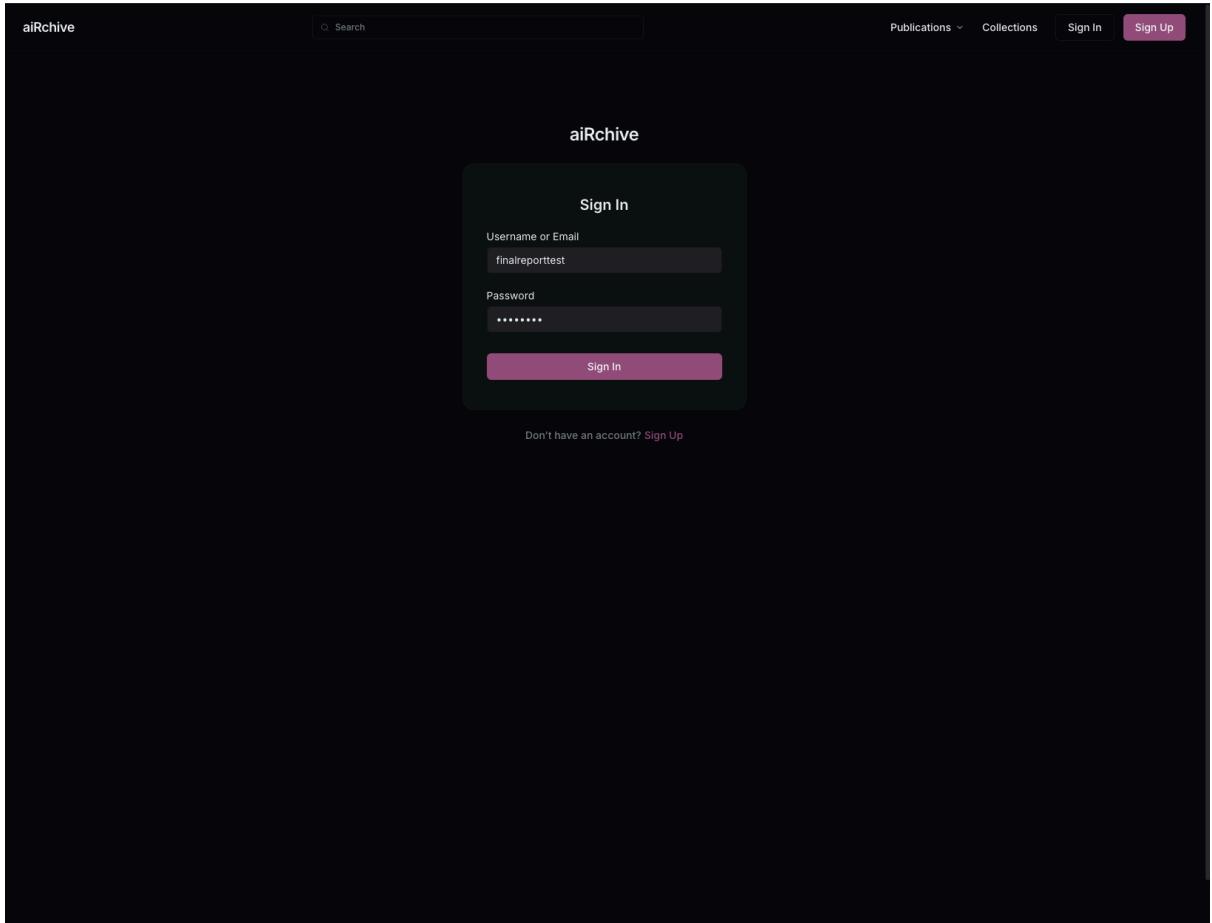
- Default collection created on account registration.

Result Grid Filter Rows: Search Edit: Export/Import:

collection_id	account_id	title	description	is_default	is_public	created_at
45	31	Saved	Your saved publications	1	0	2025-08-09 21:15:25

Sign In

- Sign in page.



- Homepage of signed in user.

The screenshot shows the aiRchive homepage with a dark theme. At the top, there's a search bar and navigation links for 'Publications' and 'Collections'. A sidebar on the right displays 'Platform Stats' (12 publications, 675 total views, 21 total likes) and 'Your Stats' (0 Read, Liked, Saved). The main area features a 'Welcome back, Final' message with a hand icon. Below it is a 'Recommended For You' section with several cards:

- ARTICLE**: Implementing MCP Servers in Python: An AI Shopping... (by Akhn Admail, Aug 3, 2025)
- PAPER**: Trustworthy Distributed AI Systems: Robustness, Privacy... (by Wensi Wei, Feb 10, 2025)
- PAPER**: Spatio-Causal Patterns of Sample Growth (by Andre F Ribeiro, Jul 31, 2025)
- BLOG**: A Blog About Nothing (by Some Wonky Dude, Apr 8, 2025)
- PAPER**: Composable OS Kernel Architectures for Autonomou... (by Rajpreet Singh, Aug 1, 2025)
- BLOG**: Our Schema (by Akhn Admail, Aug 3, 2025)

At the bottom of the sidebar, there's a 'Recent Interactions' section stating 'No recent activity.'

- POST login request.

The screenshot shows the Network tab in a browser developer tools window, specifically the 'Headers' tab. A POST request for 'http://localhost:5173/api/auth/login' is selected. The request headers are as follows:

```

POST /api/auth/login HTTP/1.1
Host: localhost:5173
Content-Type: application/json
Accept: application/json, text/plain, */*
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Content-Length: 59
Cookie: username=localhost-8888="2|1:0|10:1752511473|23:username-localhost-8888|192:eyJcI2VybFIzSjI6Cjy2nM1MGU2NWJQJD0EM0mfOQZiZDcwZjMjY0TkwZ0UaNylsICJUWYlIl0gkFub255bW91cyBnbmVtZSisfJkxNwboF5X25nbWUOAIoW5vbntb3VzIeuZWlilwglmluXRpYVWxzJogkFNlwglnVbG9yijqbnvIsbh0-if1f9994ba642b47d81c6c89dde1aa074011615410245afbc71b68d49075f95b";_xsr=2|23037628|ddc8cd807ba5e42a6be3eb2b6e774|1752511473
DNT: 1
Origin: http://localhost:5173
Priority: u=0
Referer: http://localhost:5173/signin
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Sec-GPC: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:141.0) Gecko/20100101 Firefox/14.0

```

- POST affinity recalculations request.

The screenshot shows the NetworkMiner interface with the XHR tab selected. A single POST request is highlighted, showing the URL `http://localhost:5173/api/users/me/affinities`. The response status is **202 Accepted**. The response headers include `Content-Type: application/json`, `Content-Length: 0`, and `Date: Sun, 10 Aug 2025 04:22:37 GMT`. The request headers show `Origin: http://localhost:5173`.

- GET platform statistics request

The screenshot shows the NetworkMiner interface with the XHR tab selected. A single GET request is highlighted, showing the URL `http://localhost:5173/api/platform/statistics`. The response status is **200 OK**. The response body contains JSON data: `views: 675`, `likes: 21`, and `publications: 12`.

- GET recommended publications request. This just returns the globally popular publications as the new user doesn't have any interactions yet.

The screenshot shows the Network tab of a browser developer tools interface. It displays a list of requests and their responses. The requests are primarily GET and POST methods to local URLs like '/platform', '/stats', and '/recommendations'. The responses are JSON objects, with one response highlighted in blue. The highlighted response is for a publication with ID 0, titled "Implementing MCP Servers in Python: An AI Shopping Assistant with Gradio". The response includes fields such as 'publid: 2', 'title: 'Implementing MCP Servers in Python: An AI Shopping Assistant with Gradio'', 'kind: 'ARTICLE'', 'publishedAt: [2025, 8, 3, 6, 33, 16]', 'viewCount: 199', 'likeCount: 4', 'authors: []', and 'topics: []'. There are also other entries for publications with IDs 1, 2, 3, and 4, each with their own details like title, kind, publishedAt, viewCount, likeCount, authors, and topics.

Status	Meth..	Domain	File	Initiator	Type	Transferred	Size
200	GET	local...	platform	axios.js:1672	json	183 B	42 B
200	GET	local...	stats	axios.js:1672	json	171 B (raced)	30 B
200	GET	local...	interactions?limit=3	axios.js:1672	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672	json	156 B	15 B
200	GET	local...	7	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	16	axios.js:1672	json	156 B	15 B
200	GET	local...	6	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672	json	156 B	15 B
200	POST	local...	logout	axios.js:1672	json	279 B	25 B
200	GET	local...	recommendations?pa	axios.js:1672	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672	json	183 B (raced)	42 B
200	GET	local...	stats	axios.js:1672	json	171 B (raced)	30 B
200	GET	local...	interactions?limit=3	axios.js:1672	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	7	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	16	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	6	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672	json	156 B (raced)	15 B
200	POST	local...	logout	axios.js:1672	json	279 B	25 B
200	GET	local...	recommendations?pa	axios.js:1672	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672	json	183 B	42 B
200	GET	local...	stats	axios.js:1672	json	171 B	30 B
200	GET	local...	interactions?limit=3	axios.js:1672	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672	json	156 B	15 B
200	GET	local...	7	axios.js:1672	json	156 B	15 B
200	GET	local...	1	axios.js:1672	json	156 B	15 B
200	GET	local...	16	axios.js:1672	json	156 B	15 B
200	GET	local...	6	axios.js:1672	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672	json	156 B (raced)	15 B

42 requests 13.54 kB / 20.17 kB transferred Finish: 12.96 min DOMContentLoaded: 199 ms load: 2.13 s

- GET user stats request

The screenshot shows the Network tab of the DOM and Style Inspector. It lists 42 requests made to the local host. The selected request is a GET to '/api/users/me/stats' with a status of 200, type json, transferred 171 B, and size 30 B. The initiator is 'axios.js:1672'. Other requests include interactions, platform, login, logout, and recommendations endpoints.

Status	Meth...	Domain	File	Initiator	Type	Transferred	Size
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B (raced)	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B	15 B
200	POST	local...	logout	axios.js:1672 (...)	json	279 B	25 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	POST	local...	login	axios.js:1672 (...)	json	446 B	153 B
202	POST	local...	affinities	axios.js:1672 (...)	xml	166 B	0 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B
200	POST	local...	logout	axios.js:1672 (...)	json	279 B	25 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	POST	local...	login	axios.js:1672 (...)	json	446 B	153 B
202	POST	local...	affinities	axios.js:1672 (...)	xml	166 B	0 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B

- GET recent interactions

The screenshot shows the Network tab of the Firefox developer tools. It lists 42 requests made to the local host. The selected request is a GET to '/api/users/me/interactions' with a status of 200 OK, type json, transferred 142 B, and size 3 B. The initiator is 'axios.js:1672'. Other requests include stats, platform, login, logout, and recommendations endpoints.

Status	Meth...	Domain	File	Initiator	Type	Transferred	Size
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B (raced)	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B	15 B
200	POST	local...	logout	axios.js:1672 (...)	json	279 B	25 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	POST	local...	login	axios.js:1672 (...)	json	446 B	153 B
202	POST	local...	affinities	axios.js:1672 (...)	xml	166 B	0 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B
200	POST	local...	logout	axios.js:1672 (...)	json	279 B	25 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	POST	local...	login	axios.js:1672 (...)	json	446 B	153 B
202	POST	local...	affinities	axios.js:1672 (...)	xml	166 B	0 B
200	GET	local...	recommendations?pa...	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B

Sign Out

- Sign out button.

The screenshot shows the aiRchive platform interface. At the top, there is a search bar and a navigation bar with 'Publications', 'Collections', and a language switch ('FR'). A dropdown menu is open, showing 'My Profile', 'My Collections', and a red-highlighted 'Sign Out' option. The main content area features a 'Welcome back, Final 🙌' message and a 'Let's continue your journey through the latest in artificial intelligence.' sub-message. Below this is a 'Recommended For You' section with six items:

- ARTICLE**: Implementing MCP Servers in Python: An AI Shopping... by Aekn Admal (Aug 3, 2025) - LLM, AI Agents. 199 views, 4 likes.
- PAPER**: Trustworthy Distributed AI Systems: Robustness, Privacy... by Wengqi Wei (Feb 10, 2025) - PRIV, AI Governance, Distributed AI. 145 views, 4 likes.
- PAPER**: Spatio-Causal Patterns of Sample Growth by Andre F Ribeiro (Jul 31, 2025) - ML, Data Mining, CSS. 110 views, 2 likes.
- BLOG**: A Blog About Nothing by Some Wonky Dude (Apr 8, 2025) - NLP, MT, OS. 24 views, 2 likes.
- PAPER**: Composable OS Kernel Architectures for Autonomou... by Rajpreet Singh (Aug 1, 2025) - OS, AI. 53 views, 3 likes.
- BLOG**: Our Schema by Aekn Admal (Aug 3, 2025) - AV. 83 views, 2 likes.

On the right side, there are three boxes: 'Platform Stats' (Publications: 12, Total Views: 675, Total Likes: 21), 'Your Stats' (Read: 0, Liked: 0, Saved: 0), and 'Recent Interactions' (No recent activity).

- POST logout request.

Screenshot of a browser developer tools Network tab showing network traffic. The table lists requests and responses. A specific POST request for 'logout' is highlighted in blue.

Status	Meth..	Domain	File	Initiator	Type	Transferred	Size
200	GET	local...	me	axios.js:1672 (...)	xml	108 B	0 B
200	GET	local...	recommendations?pag	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	POST	local...	register	axios.js:1672 (...)	json	446 B	153 B
200	GET	local...	recommendations?pag	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B (raced)	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B	15 B
200	POST	local...	logout	axios.js:1672 (...)	json	279 B	25 B
200	GET	local...	recommendations?pag	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	POST	local...	login	axios.js:1672 (...)	json	446 B	153 B
202	POST	local...	affinities	axios.js:1672 (...)	xml	166 B	0 B
200	GET	local...	recommendations	axios.js:1672 (...)	json	2.23 kB (raced)	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B (raced)	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B (raced)	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B
200	POST	local...	logout	axios.js:1672 (...)	json	279 B	25 B
200	GET	local...	recommendations?pag	axios.js:1672 (...)	json	2.23 kB	2.08 kB

Selected Request Headers (254 B):

- Access-Control-Allow-Origin: http://localhost:5173
- Connection: close
- Content-Length: 25
- Content-Type: application/json
- Date: Sun, 10 Aug 2025 04:25:30 GMT
- Set-Cookie: SESSIONID=-Version=1;Path=/;Max-Age=0;HttpOnly
- Vary: Origin

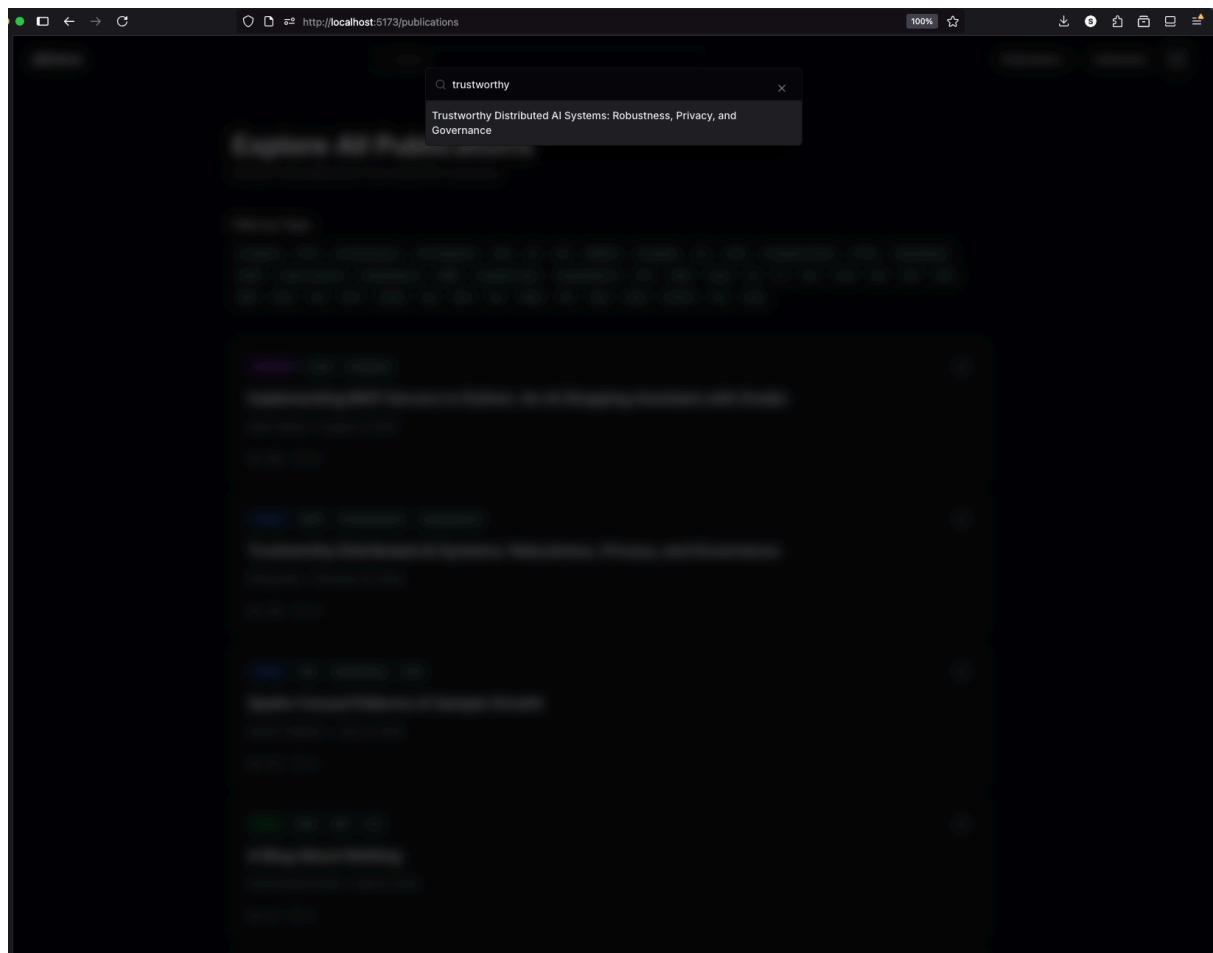
Selected Response Headers (254 B):

- Accept: application/json, text/plain, */*
- Accept-Encoding: gzip, deflate, br, zstd
- Accept-Language: en-US,en;q=0.5
- Connection: keep-alive
- Content-Length: 0
- Cookie: username=localhost-8888="2|1:0|10-1752511473|23:username-localhost-8888|192.eyJtc2VybmlzS16CjyZmM1MGU2NWJ0DE0Mnf0GzIDczw2JMyOTwzG0U4NyIsIjJyWlIlogKfu255BW9icyBnNmVzSisCjukXNwbGF5X2hbWUOIAW5vbntb3V2IEtuzWliliwglmluxRpxWxzljogKFNIwgmNvb9yjojbvNsbh0=tJf19994a6b642b47d1c6c89ddfa07074011615410245afbc7fb68a4907595b5_";xsrf=2|23037628ddc8cd807ba5e42a61be3eb2b6e774|1752511473;SESSIONID=679082C5BCD2FD9F371291461689BA4
- DNT: 1
- Host: localhost:5173
- Origin: http://localhost:5173
- Priority: u=0
- Referer: http://localhost:5173/_
- Sec-Fetch-Dest: empty
- Sec-Fetch-Mode: cors
- Sec-Fetch-Site: same-origin
- Sec-GPC: 1
- User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:141.0) Gecko/20100101 Firefox/14.0.0

Network Summary: 30 requests | 11.14 kB / 15.90 kB transferred | Finish: 12.08 min | DOMContentLoaded: 199 ms | load: 2.13 s

Browse & Search Publications

- Searching by title keyword “trustworthy”



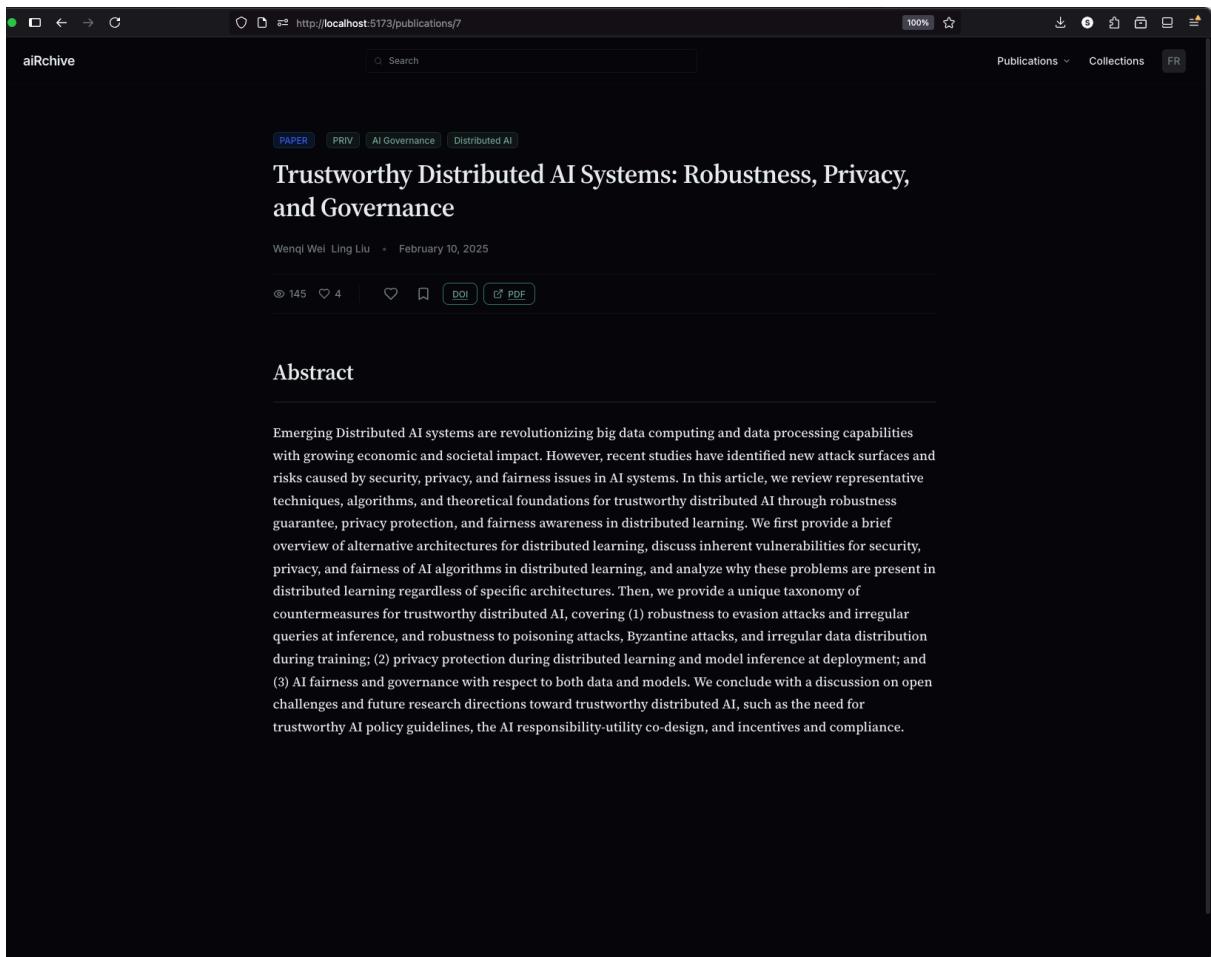
- GET request for search. This returns a MiniPublication DTO that contains the bare minimum metadata as the complete publication content is not needed for searching.

The screenshot shows the Network tab of a browser developer tools interface. On the left, a list of network requests is displayed, mostly showing GET requests for local resources like 'platform', 'stats', and various 'interactions' and 'recommendations'. One specific request, a GET for 'search?q=trustwor...', is highlighted in blue at the bottom of the list. On the right, a detailed view of this selected JSON response is shown. The response object has properties like 'publid: 7', 'title: "Trustworthy Distributed AI Systems: Robustness, Privacy, and Governance"', 'kind: "PAPER"', 'publishedAt: [2025, 2, 10, 8, 0]', 'viewCount: 145', 'likeCount: 4', 'authors: [{}]', and 'topics: (3)[..., ..., ...]'. The 'Response' tab is selected in the top navigation bar.

Status	Meth...	Domain	File	Initiator	Type	Transferred	Size
200	GET	local...	platform	axios.js:1672 (...)	json	183 B (raced)	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B (raced)	30 B
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B
200	POST	local...	logout	axios.js:1672 (...)	json	279 B	25 B
200	GET	local...	recommendations?paq	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	POST	local...	login	axios.js:1672 (...)	json	446 B	153 B
202	POST	local...	affinities	axios.js:1672 (...)	xml	166 B	0 B
200	GET	local...	recommendations	axios.js:1672 (...)	json	2.23 kB	2.08 kB
200	GET	local...	platform	axios.js:1672 (...)	json	183 B	42 B
200	GET	local...	stats	axios.js:1672 (...)	json	171 B	30 B
200	GET	local...	interactions?limit=	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	topics	axios.js:1672 (...)	json	2.87 kB (raced)	2.73 kB
200	GET	local...	recommendations?paq	axios.js:1672 (...)	json	3.66 kB	3.52 kB
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	15	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	11	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	12	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	14	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	search?q=trustwor...	axios.js:1672 (...)	json	142 B	2 B
200	GET	local...	search?q=trustworth...	axios.js:1672 (...)	json	548 B (raced)	406 B

56 requests | 20.34 kB / 28.95 kB transferred | Finish: 18.44 min | DOMContentLoaded: 199 ms | load: 2.13 s

- Rendered publication page.



- GET request for full publication content. As you can see this endpoint responds with the complete publication content. Also, as you can see from the following POST and GET requests there is a POST request to update view count, GET request to determine if the account has liked the publication, and a GET request to determine if the user has saved the publication.

The screenshot shows a browser's developer tools Network tab with several requests listed. The JSON response pane on the right displays the following publication data:

```

{
  "pubId": 7,
  "title": "Trustworthy Distributed AI Systems: Robustness, Privacy, and Governance",
  "content": "<p><b>Emerging Distributed AI systems are revolutionizing big data computing and data processing capabilities with growing economic and societal impact. However, recent studies have identified new attack surfaces and risks caused by security, privacy, and fairness issues in AI systems. In this article, we review representative techniques, algorithms, and theoretical foundations for trustworthy distributed AI through robustness guarantee, privacy protection, and fairness awareness in distributed learning. We... and robustness to poisoning attacks, Byzantine attacks, and irregular data distribution during training; (2) privacy protection during distributed learning and model inference at deployment; and (3) AI fairness and governance with respect to both data and models. We conclude with a discussion on open challenges and future research directions toward trustworthy distributed AI, such as the need for trustworthy AI policy guidelines, the AI responsibility-utility co-design, and incentives and compliance.</b></p>\n",
  "doi": "https://doi.org/10.1145/3645102",
  "url": "https://dl.acm.org/doi/pdf/10.1145/3645102",
  "kind": "PAPER",
  "submitterId": 1,
  "submittedAt": "(6) [ 2025, 8, 3, 19, 6, 16 ]",
  "publishedAt": "(5) [ 2025, 2, 10, 8, 0 ]",
  "status": "PUBLISHED",
  "viewCount": 145,
  "likeCount": 4,
  "authors": [
    {
      "personId": 17,
      "firstName": "Wenqi",
      "lastName": "Wei"
    },
    {
      "personId": 18,
      "firstName": "Ling",
      "lastName": "Liu"
    }
  ],
  "topics": [
    {
      "topicId": 38,
      "code": "PRIV",
      "fullName": "Security and Privacy"
    },
    {
      "topicId": 39,
      "code": "GOV",
      "fullName": "AI Governance"
    },
    {
      "topicId": 40,
      "code": "DAI",
      "fullName": "Distributed AI"
    }
  ]
}

```

- This is the result of the same query in MySQL Workbench

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL statement:

```
1 • SELECT * FROM airchive.publication WHERE pub_id = 7;
```

The results grid shows one row of data:

pub_id	title	content	doi	url	kind	submitter_id	submitted_at	published_at	status
7	Trustworthy Distributed AI S...	<p>Emerging Distributed AI...	https://doi.org/10.1145/3645...	https://dl.acm.org/doi/pdf/10...	PAPER	1	2025-08-03 19:06:16	2025-02-10 08:00:00	PUBLISHED

- This is the new view entry in the publication_view table

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL statement:

```
1 • SELECT * FROM airchive.publication_view ORDER BY viewed_at DESC;
```

The results grid shows three rows of data:

account_id	pub_id	viewed_at
31	7	2025-08-09 21:33:37
30	1	2025-08-09 19:27:41
1	2	2025-08-09 05:07:57

- This is the publication explore page

The screenshot shows the 'Explore All Publications' page on the aiRhive platform. At the top, there is a search bar and navigation links for 'Publications', 'Collections', and a language selector ('FR'). Below the header, the title 'Explore All Publications' is displayed, followed by a subtitle 'Discover new publications from across the community'. A 'Filter by Topic' section contains numerous small, semi-transparent buttons representing various research areas such as AI Agents, ETH, AI Governance, AI in Robotics, AGI, AI, AV, BENCH, Causality, CL, CSS, Computer Vision, CTRL, Data Mining, DATA, Deep Learning, Distributed AI, EMB, Example Topic, Explainable AI, FSL, GEN, hasb, IR, IT, KG, LLM, ML, MT, MA, MM, NLP, OS, OPT, PLAN, QA, REC, RL, PRIV, SP, SIM, ASR, SYNTH, SY, VQA. The main content area displays four publication cards:

- Implementing MCP Servers in Python: An AI Shopping Assistant with Gradio**
Aekn Admal • August 3, 2025
199 views, 4 likes
- Trustworthy Distributed AI Systems: Robustness, Privacy, and Governance**
Wenqi Wei • February 10, 2025
146 views, 4 likes
- Spatio-Causal Patterns of Sample Growth**
Andre F Ribeiro • July 31, 2025
110 views, 2 likes
- A Blog About Nothing**
Some Wonky Dude • April 8, 2025
24 views, 2 likes

- GET recommendations request

Filter IRI's
Pick an element from the page (Cmd+Shift+C or Cmd+Opt+C)

Status	Meth...	Domain	File	Initiator	Type	Transferred	Size
200	GET	local...	interactions?limit=3	axios.js:1672 (...)	json	142 B (raced)	2 B
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	topics	axios.js:1672 (...)	json	2.87 kB (raced)	2.73 kB
200	GET	local...	recommendations?page=1&pageSize=10	axios.js:1672 (...)	json	3.66 kB	3.52 kB
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	15	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	11	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	12	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	14	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	search?q=trustwor	axios.js:1672 (...)	json	142 B	2 B
200	GET	local...	search?q=trustworthy	axios.js:1672 (...)	json	548 B (raced)	406 B
200	GET	local...	7	axios.js:1672 (...)	json	2.35 kB (raced)	2.21 kB
200	POST	local...	view	axios.js:1672 (...)	xml	160 B	0 B
200	GET	local...	like	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	topics	axios.js:1672 (...)	json	2.87 kB (raced)	2.73 kB
200	GET	local...	recommendations?page=1&pageSize=10	axios.js:1672 (...)	json	3.66 kB (raced)	3.52 kB
200	GET	local...	2	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	7	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	1	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	16	axios.js:1672 (...)	json	156 B	15 B
200	GET	local...	6	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	3	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	15	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	11	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	12	axios.js:1672 (...)	json	156 B (raced)	15 B
200	GET	local...	14	axios.js:1672 (...)	json	156 B (raced)	15 B

All HTML CSS JS XHR Fonts Images Media WS Other Disable Cache No Throttling

Headers Cookies Request Response Timings Stack Trace

Filter properties

JSON Raw

```

▶ 0: { pubId: 2, title: "Implementing MCP-Servers in Python: An AI Shopping Assistant with Gradio", kind: "ARTICLE", ... }
▶ 1: { pubId: 7, title: "Trustworthy Distributed AI Systems: Robustness, Privacy, and Governance", kind: "PAPER", ... }
▶ 2: { pubId: 1, title: "Spatio-Causal Patterns of Sample Growth", kind: "PAPER", ... }
▶ 3: { pubId: 16, title: "A Blog About Nothing", kind: "BLOG", ... }
▶ 4: { pubId: 6, title: "Composable OS Kernel Architectures for Autonomous Intelligence", kind: "PAPER", ... }
▶ 5: { pubId: 3, title: "Our Schema", kind: "BLOG", ... }
▶ 6: { pubId: 15, title: "CSI Obfuscation: Single-Antenna Transmitters Can Not Hide from Adversarial Multi-Antenna Radio Localization Systems", kind: "PAPER", ... }
▶ 7: { pubId: 11, title: "Sort-free Gaussian Splatting via Weighted Sum Rendering", kind: "PAPER", ... }
▶ 8: { pubId: 12, title: "LoRA-X: Bridging Foundation Models with Training-Free Cross-Model Adaptation", kind: "PAPER", ... }
▶ 9: { pubId: 14, title: "CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge", kind: "PAPER", ... }

```

72 requests 28.07 kB / 39.86 kB transferred Finish: 27.87 min DOMContentLoaded: 199 ms load: 2.13 s

Publication Interactions, Recommendations, & Related Functionalities

- Explore publications page before any recommendation data is persisted for the user

The screenshot shows the 'Explore All Publications' page on the aiRhive platform. At the top, there is a search bar and navigation links for 'Publications' and 'Collections'. Below the header, the title 'Explore All Publications' is displayed, followed by a subtitle 'Discover new publications from across the community'. A 'Filter by Topic' section contains numerous small, semi-transparent topic tags such as 'AI Agents', 'ETH', 'AI Governance', etc. Four publication cards are listed:

- Implementing MCP Servers in Python: An AI Shopping Assistant with Gradio** by Aekn Admal (August 3, 2025). It has 199 views and 4 likes.
- Trustworthy Distributed AI Systems: Robustness, Privacy, and Governance** by Wenqi Wei (February 10, 2025). It has 146 views and 4 likes.
- Spatio-Causal Patterns of Sample Growth** by Andre F Ribeiro (July 31, 2025). It has 110 views and 2 likes.
- A Blog About Nothing** by Some Wonky Dude (April 8, 2025). It has 24 views and 2 likes.

- The new users uncalculated affinity tables before interacting with publications. The two entries in author_affinity and three entries in topic_affinity are because the publication I just viewed in the previous screenshots are associated with those authors

and topics.

```

1 - SELECT |
2     'AUTHOR' AS type,
3     aa.account_id,
4     a.username,
5     p.first_name AS author_first_name,
6     p.last_name AS author_last_name,
7     aa.score,
8     aa.last_updated
9  FROM airhive.author_affinity aa
10 JOIN airhive.account a ON aa.account_id = a.account_id
11 JOIN airhive.person p ON aa.author_id = p.person_id
12 WHERE a.username = 'finalreporttest'
13
14 UNION ALL
15
16 SELECT
17     'TOPIC' AS type,
18     ta.account_id,
19     a.username,
20     t.code AS topic_code,
21     t.full_name AS topic_name,
22     ta.score,
23     ta.last_updated
24  FROM airhive.topic_affinity ta
25 JOIN airhive.account a ON ta.account_id = a.account_id
26 JOIN airhive.topic t ON ta.topic_id = t.topic_id
27 WHERE a.username = 'finalreporttest';
28

```

type	account_id	username	author_first_name	author_last_name	score	last_updated
AUTHOR	31	finalreporttest	Wenqi	Wei	0.5	2025-08-09 21:33:37
AUTHOR	31	finalreporttest	Ling	Liu	0.5	2025-08-09 21:33:37
TOPIC	31	finalreporttest	PRIV	Security and Privacy	0.5	2025-08-09 21:33:37
TOPIC	31	finalreporttest	GOV	AI Governance	0.5	2025-08-09 21:33:37
TOPIC	31	finalreporttest	DAI	Distributed AI	0.5	2025-08-09 21:33:37

- I will now like and save the following publication which has ML, Causality, and AI as it's associated topics and Lei Zan, Keli Zhang, Ruichu Cai, and Lujia Pan as it's authors.

CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge

Lei Zan Keli Zhang Ruichu Cai Lujia Pan • August 4, 2025

PAPER ML Causality AI

Abstract

Large Language Models (LLMs) have demonstrated strong performance across a wide range of tasks, yet they still struggle with complex mathematical reasoning, a challenge fundamentally rooted in deep structural dependencies. To address this challenge, we propose the CAMA framework, a two-stage causal framework that equips LLMs with explicit, reusable mathematical structure. In the learning stage, CAMA first constructs the MCG (Mathematical Causal Graph), a high-level representation of solution strategies, by combining LLM priors with causal discovery algorithms applied to a corpus of question-solution pairs. The resulting MCG encodes essential knowledge points and their causal dependencies. To better align the graph with downstream reasoning tasks, CAMA further refines the MCG through iterative feedback derived from a selected subset of the question-solution pairs. In the reasoning stage, given a new question, CAMA dynamically extracts a task-relevant subgraph from the MCG, conditioned on both the question content and the LLM's intermediate reasoning trace. This subgraph, which encodes the most pertinent knowledge points and their causal dependencies, is then injected back into the LLM to guide its reasoning process. Empirical results on real-world datasets show that CAMA significantly improves LLM performance on challenging mathematical problems. Furthermore, our experiments demonstrate that structured guidance consistently outperforms unstructured alternatives, and that incorporating asymmetric causal relationships yields greater improvements than using symmetric associations alone.

- Here are the newly inserted collection_item, publication_view, and publication_like entries in the database reflecting these interactions.

```

1 • |SELECT
2     'VIEW' AS interaction_type,
3     p.pub_id,
4     p.title,
5     a.username,
6     v.viewed_at AS interaction_time
7 FROM airchive.publication_view v
8 JOIN airchive.publication p ON v.pub_id = p.pub_id
9 JOIN airchive.account a ON v.account_id = a.account_id
10 WHERE p.pub_id = 14 AND a.username = 'finalreporttest'
11
12 UNION ALL
13
14 SELECT
15     'LIKE' AS interaction_type,
16     p.pub_id,
17     p.title,
18     a.username,
19     l.liked_at AS interaction_time
20 FROM airchive.publication_like l
21 JOIN airchive.publication p ON l.pub_id = p.pub_id
22 JOIN airchive.account a ON l.account_id = a.account_id
23 WHERE p.pub_id = 14 AND a.username = 'finalreporttest'
24
25 UNION ALL
26
27 SELECT
28     'SAVE' AS interaction_type,
29     p.pub_id,
30     p.title,
31     a.username,
32     ci.added_at AS interaction_time
33 FROM airchive.collection_item ci
34 JOIN airchive.publication p ON ci.pub_id = p.pub_id
35 JOIN airchive.collection c ON ci.collection_id = c.collection_id
36 JOIN airchive.account a ON c.account_id = a.account_id
37 WHERE p.pub_id = 14 AND a.username = 'finalreporttest';
38

```

Result Grid Filter Rows: Search Export:

interaction_type	pub_id	title	username	interaction_time
VIEW	14	CAMA: Enhancing Mathema...	finalreporttest	2025-08-09 21:49:23
LIKE	14	CAMA: Enhancing Mathema...	finalreporttest	2025-08-09 21:49:24
SAVE	14	CAMA: Enhancing Mathema...	finalreporttest	2025-08-09 21:49:25

- Here are the updated topic and author affinity scores in the table reflecting these changes. As you can see the scores are now different, and new entries were added for the topics and authors associated with the newly viewed, saved, and liked publication.

```

1 •  SELECT
2      'AUTHOR' AS type,
3      a.username,
4      aa.author_id AS id,
5      CONCAT(p.first_name, ' ', p.last_name) AS name,
6      aa.score,
7      aa.last_updated
8  FROM airchive.author_affinity aa
9  JOIN airchive.account a ON aa.account_id = a.account_id
10 JOIN airchive.person p ON aa.author_id = p.person_id
11 WHERE a.username = 'finalreporttest'
12
13 UNION ALL
14
15 SELECT
16      'TOPIC' AS type,
17      a.username,
18      ta.topic_id AS id,
19      t.full_name AS name,
20      ta.score,
21      ta.last_updated
22  FROM airchive.topic_affinity ta
23  JOIN airchive.account a ON ta.account_id = a.account_id
24  JOIN airchive.topic t ON ta.topic_id = t.topic_id
25 WHERE a.username = 'finalreporttest';

```

100% ◇ 1:1

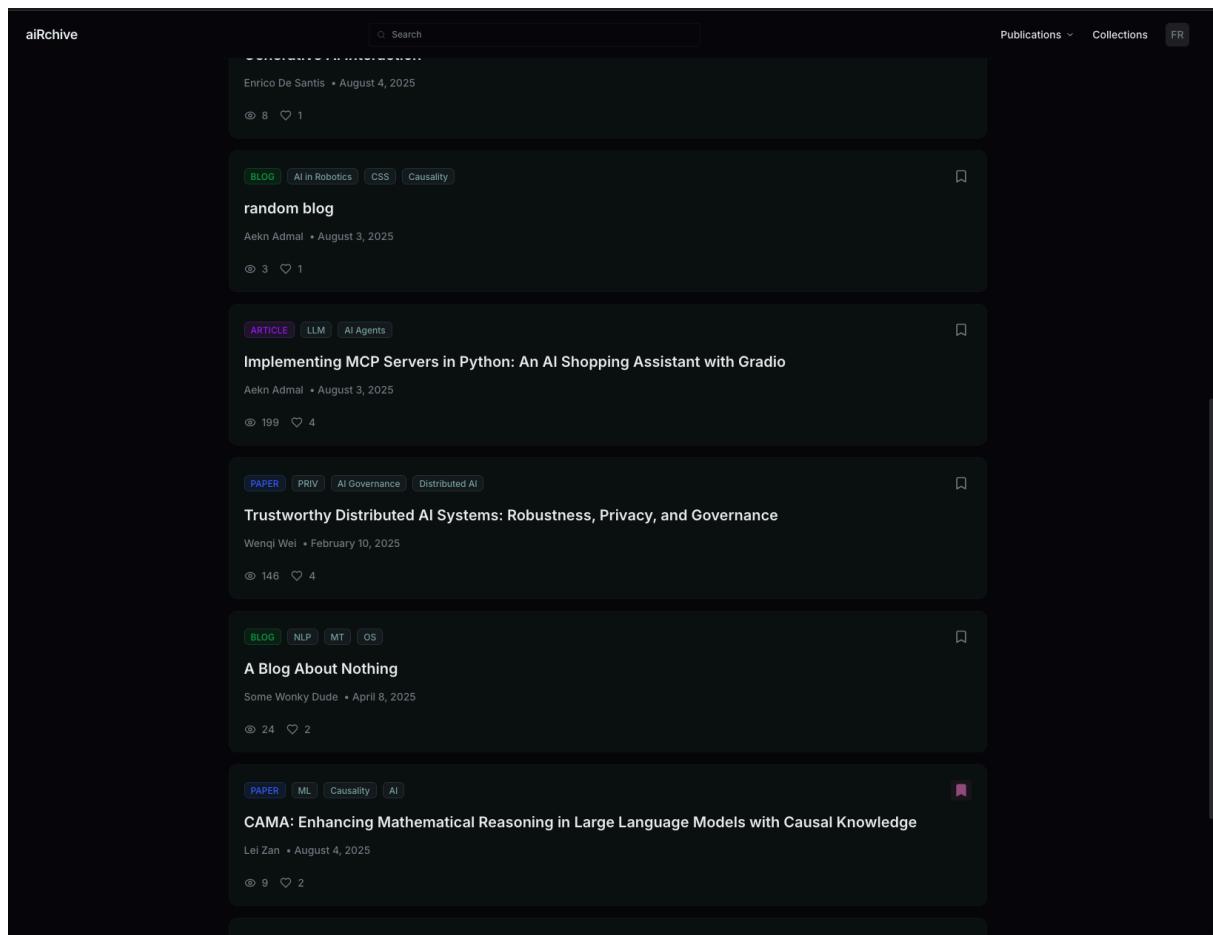
Result Grid Filter Rows: Search Export:

type	username	id	name	score	last_updated
AUTHOR	finalreporttest	17	Wenqi Wei	0.5	2025-08-09 21:33:37
AUTHOR	finalreporttest	18	Ling Liu	0.5	2025-08-09 21:33:37
AUTHOR	finalreporttest	40	Lei Zan	6	2025-08-09 21:49:25
AUTHOR	finalreporttest	41	Keli Zhang	6	2025-08-09 21:49:25
AUTHOR	finalreporttest	42	Ruichu Cai	6	2025-08-09 21:49:25
AUTHOR	finalreporttest	43	Lujia Pan	6	2025-08-09 21:49:25
TOPIC	finalreporttest	4	Machine Learning	6	2025-08-09 21:49:25
TOPIC	finalreporttest	35	Causality	6	2025-08-09 21:49:25
TOPIC	finalreporttest	37	Artificial Intelligence	6	2025-08-09 21:49:25
TOPIC	finalreporttest	38	Security and Privacy	0.5	2025-08-09 21:33:37
TOPIC	finalreporttest	39	AI Governance	0.5	2025-08-09 21:33:37
TOPIC	finalreporttest	40	Distributed AI	0.5	2025-08-09 21:33:37

- This is the publication explore page which will now display different publications because the system recognizes that the account has higher scores for certain topics and authors. As you can see all the ML and AI publications shot up. The recently saved publication shows up below a lot of them because the system knows the account has already viewed the publication so it is not recommended right away.

The screenshot shows a dark-themed web application titled "aiRhive". At the top, there's a navigation bar with a search input field and links for "Publications", "Collections", and "FR". Below the header, a main title "Explore All Publications" is displayed, followed by a subtitle "Discover new publications from across the community". A "Filter by Topic" section contains numerous small, semi-transparent tags representing various research areas like "AI Agents", "ETH", "AI Governance", etc. Below this, four publication cards are listed:

- Spatio-Causal Patterns of Sample Growth** by Andre F Ribeiro (July 31, 2025). Tags: PAPER, ML, Data Mining, CSS. Interaction counts: 110 views, 2 likes.
- Composable OS Kernel Architectures for Autonomous Intelligence** by Rajpreet Singh (August 1, 2025). Tags: PAPER, OS, AI. Interaction counts: 53 views, 3 likes.
- CSI Obfuscation: Single-Antenna Transmitters Can Not Hide from Adversarial Multi-Antenna Radio Localization Systems** by Phillip Stephan (August 4, 2025). Tags: PAPER, ML, IT, SP. Interaction counts: 9 views, 1 like.
- Noosemia: toward a Cognitive and Phenomenological Account of Intentionality Attribution in Human-Generative AI Interaction** by Enrico De Santis (August 4, 2025). Tags: PAPER, CSS, AI, CL. Interaction counts: 8 views, 0 likes.



- I will now save a publication at the bottom of the list which has the topic Computer Vision. I will save it without viewing it to demonstrate how publications that are not viewed are recommended over publications that are viewed. The following screenshots show the Computer Vision publications on the bottom, and then a screenshot of the page after saving the publication without refreshing the page, and then the relevant data in the database, and then the refreshed explore page with them ranked higher up on the page for the user.

The screenshot shows the aiRhive platform interface. At the top, there is a search bar and navigation links for 'Publications', 'Collections', and 'FR'. Below the header, a list of publications is displayed in a dark-themed card layout:

- Wenqi Wei** • February 10, 2025
146 views, 4 likes
[BLOG] NLP MT OS
- A Blog About Nothing**
Some Wonky Dude • April 8, 2025
25 views, 3 likes
- CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge**
Lei Zan • August 4, 2025
9 views, 2 likes
- Our Schema**
Aekn Admal • August 3, 2025
83 views, 2 likes
- Sort-free Gaussian Splatting via Weighted Sum Rendering**
Qiqi Hou • April 9, 2025
13 views, 0 likes
- LoRA-X: Bridging Foundation Models with Training-Free Cross-Model Adaptation**
Farzad Farhadzadeh • February 4, 2025
20 views, 0 likes

The screenshot shows the aiRchive platform interface. At the top, there is a search bar and navigation links for 'Publications', 'Collections', and 'FR'. Below the header, a list of publications is displayed in a dark-themed card layout:

- A Blog About Nothing** by Wengqi Wei (February 10, 2025) - 146 views, 4 likes. Tags: BLOG, NLP, MT, OS.
- CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge** by Some Wonky Dude (April 8, 2025) - 25 views, 3 likes. Tags: PAPER, ML, Causality, AI.
- Our Schema** by Lei Zan (August 4, 2025) - 9 views, 2 likes. Tags: BLOG, AV.
- Sort-free Gaussian Splatting via Weighted Sum Rendering** by Aekn Admal (August 3, 2025) - 83 views, 0 likes. Tags: PAPER, Computer Vision.
- LoRA-X: Bridging Foundation Models with Training-Free Cross-Model Adaptation** by Qiqi Hou (April 9, 2025) - 13 views, 0 likes. Tags: PAPER, Computer Vision.
- Farzad Farhadzadeh (February 4, 2025) - 20 views, 0 likes.**

```

1 •  SELECT
2      'AUTHOR' AS type,
3      a.username,
4      aa.author_id AS id,
5      CONCAT(p.first_name, ' ', p.last_name) AS name,
6      aa.score,
7      aa.last_updated
8  FROM airchive.author_affinity aa
9

```

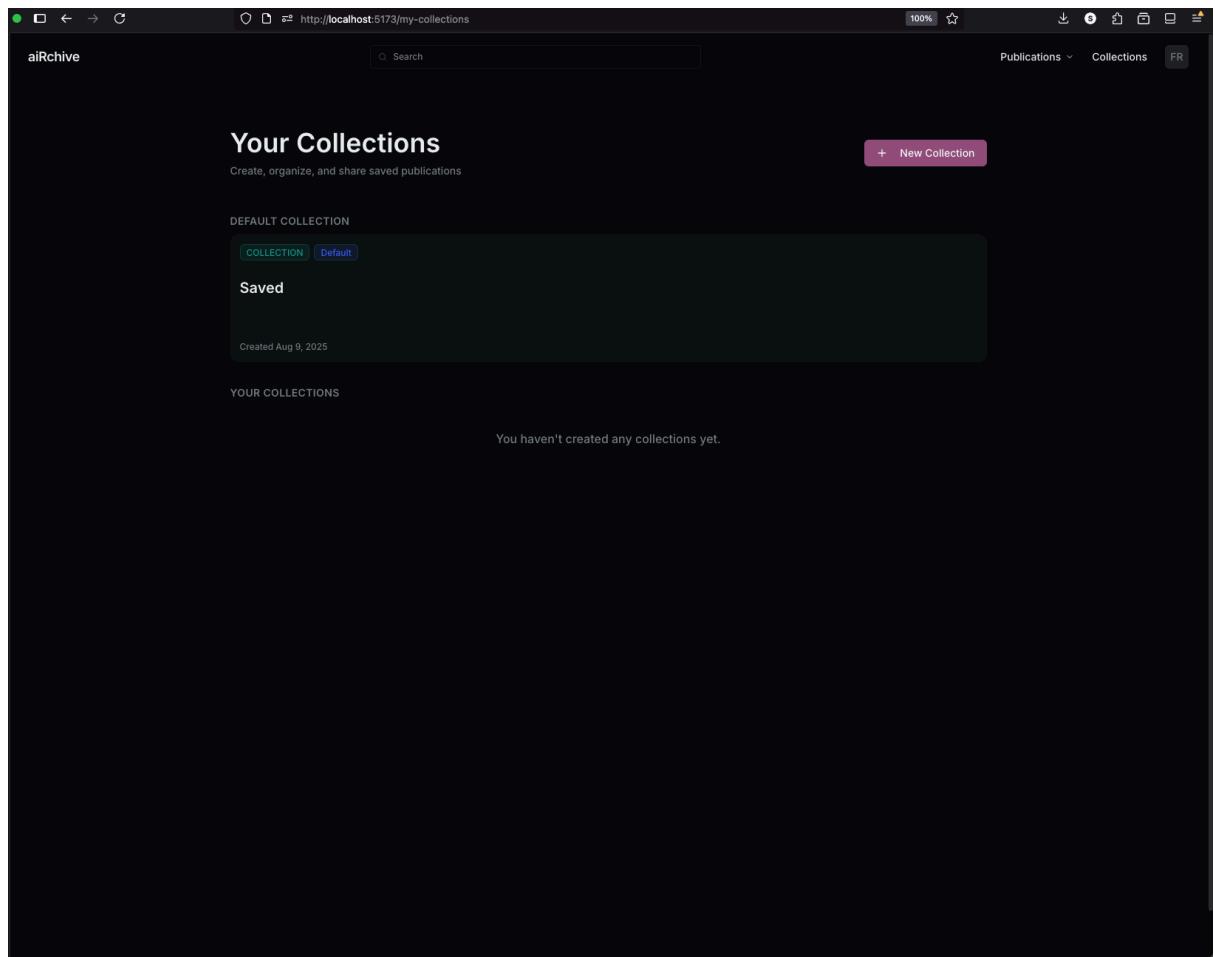
Result Grid Filter Rows: Search Export:

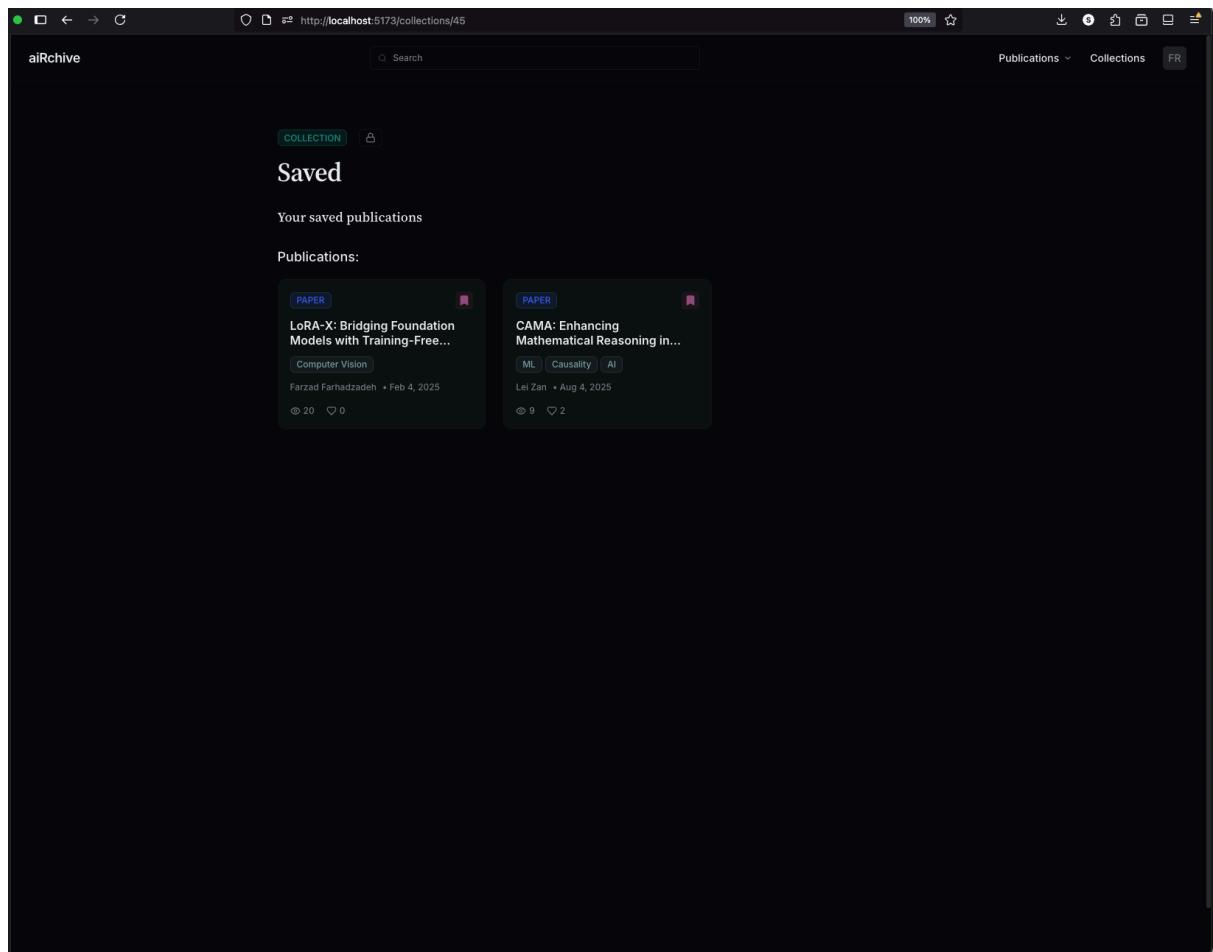
type	username	id	name	score	last_updated
AUTHOR	finalreporttest	17	Wenqi Wei	0.5	2025-08-09 21:33:37
AUTHOR	finalreporttest	18	Ling Liu	0.5	2025-08-09 21:33:37
AUTHOR	finalreporttest	31	Farzad Farhadzadeh	2.5	2025-08-09 22:02:23
AUTHOR	finalreporttest	32	Fatih Porikli	2.5	2025-08-09 22:02:23
AUTHOR	finalreporttest	35	Debasmit Das	2.5	2025-08-09 22:02:23
AUTHOR	finalreporttest	36	Shubhankar Borse	2.5	2025-08-09 22:02:23
AUTHOR	finalreporttest	40	Lei Zan	6	2025-08-09 21:49:25
AUTHOR	finalreporttest	41	Keli Zhang	6	2025-08-09 21:49:25
AUTHOR	finalreporttest	42	Ruichu Cai	6	2025-08-09 21:49:25
AUTHOR	finalreporttest	43	Lujia Pan	6	2025-08-09 21:49:25
AUTHOR	finalreporttest	47	Some Wonky Dude	6	2025-08-09 22:00:21
TOPIC	finalreporttest	2	Natural Language...	6	2025-08-09 22:00:21
TOPIC	finalreporttest	3	Computer Vision	2.5	2025-08-09 22:02:23
TOPIC	finalreporttest	4	Machine Learning	6	2025-08-09 21:49:25
TOPIC	finalreporttest	21	Machine Translation	6	2025-08-09 22:00:21
TOPIC	finalreporttest	35	Causality	6	2025-08-09 21:49:25
TOPIC	finalreporttest	36	Operating Systems	6	2025-08-09 22:00:21
TOPIC	finalreporttest	37	Artificial Intelligence	6	2025-08-09 21:49:25
TOPIC	finalreporttest	38	Security and Privacy	0.5	2025-08-09 21:33:37
TOPIC	finalreporttest	39	AI Governance	0.5	2025-08-09 21:33:37
TOPIC	finalreporttest	40	Distributed AI	0.5	2025-08-09 21:33:37

The screenshot shows the aiRhive interface with a dark theme. At the top, there's a navigation bar with 'aiRhive', a search bar, and links for 'Publications', 'Collections', and 'FR'. Below the navigation is a grid of publication cards. Each card contains the title, author, date, a small description, and interaction metrics (views, likes). The cards are separated by thin horizontal lines.

- Spatio-Causal Patterns of Sample Growth** by Andre F Ribeiro • July 31, 2025
- Composable OS Kernel Architectures for Autonomous Intelligence** by Rajpreet Singh • August 1, 2025
- CSI Obfuscation: Single-Antenna Transmitters Can Not Hide from Adversarial Multi-Antenna Radio Localization Systems** by Phillip Stephan • August 4, 2025
- LoRA-X: Bridging Foundation Models with Training-Free Cross-Model Adaptation** by Farzad Farhadzadeh • February 4, 2025
- Noosemia: toward a Cognitive and Phenomenological Account of Intentionality Attribution in Human-Generative AI Interaction** by Enrico De Santis • August 4, 2025

- After saving those two publications we can see the new default saved publication page for the user contains both publications.





Authors and Publishing Publications

Profile page before requesting author access:

The screenshot shows the aiRhive profile page at <http://localhost:5173/profile>. The interface has a dark theme with light-colored cards.

Profile Section:

- Final Report** (Reader)
- @finalreporttest
- finalreporttest@gmail.com
- [Request Author Access](#)

Your Stats:

Category	Value	Description
Read	3	
Liked	2	
Saved	2	

Recent Interactions:

- LoRA-X: Bridging Foundation Models with Training-Free Cross-Model Adaptation
You saved this paper.
- A Blog About Nothing
You liked this paper.
- A Blog About Nothing
You viewed this paper.
- CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge
You saved this paper.
- CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge
You liked this paper.

After requesting author access:

The screenshot shows the aiRchive user interface. At the top, there is a navigation bar with links for 'Publications', 'Collections', and 'FR'. A green notification bubble in the top right corner says 'Author request submitted!'. Below the navigation, there are three main sections: 'Final Report' (User: @finalreporttest, Email: finalreporttest@gmail.com) with a status 'Request Pending...', 'Your Stats' (3 Read, 2 Liked, 2 Saved), and 'Recent Interactions' (listing several recent activities like saving and liking papers).

The new request:

A screenshot of a database query results grid. The query is:

```
1 • SELECT * FROM airchive.author_request ORDER BY requested_at DESC;
```

The results grid shows two rows of data:

account_id	status	requested_at
31	PENDING	2025-08-09 22:06:31
29	PENDING	2025-08-09 05:13:33

I logged in as an administrative user to show the new request:

The screenshot shows the airRhive administration interface. At the top, there is a navigation bar with links for 'Publications', 'Collections', and a user icon. Below the navigation is a search bar. The main title 'Administration' is displayed, followed by a subtitle 'Manage author requests and topics'. A secondary navigation bar below the main title includes 'Author Requests' and 'Topic Management'. The main content area is titled 'EMAIL' and lists several author requests. Each request includes the email address, the date it was requested, and a purple 'Approve' button.

EMAIL	REQUESTED AT	
finalreporttest@gmail.com	Aug 9, 2025, 10:06 PM	<button>Approve</button>
blabla2@gmail.com	Aug 9, 2025, 5:13 AM	<button>Approve</button>
user@gmail.com	Aug 5, 2025, 1:26 PM	<button>Approve</button>
ash@gmail.com	Aug 5, 2025, 12:15 PM	<button>Approve</button>
thing3@gmail.com	Aug 2, 2025, 11:14 PM	<button>Approve</button>
thing2@gmail.com	Aug 2, 2025, 10:57 PM	<button>Approve</button>
user2@gmail.com	Aug 2, 2025, 10:47 PM	<button>Approve</button>

I'm going to approve it and go back to the finalreporttest's profile page. And as you can see I am now an author. I will also show the database entry.

The screenshot shows the airRhive dashboard interface. At the top, there is a search bar and navigation links for 'Publications', 'Collections', and a selected tab 'FR'. Below this, a dark card displays the user's profile: 'Final Report' (Author), handle '@finalreporttest', and email 'finalreporttest@gmail.com'. The profile picture is labeled 'FR'. Below the profile, a section titled 'Your Stats' shows three metrics: '3 Read', '2 Liked', and '2 Saved'. Underneath, a section titled 'Recent Interactions' lists five items, each with a small icon, the title, and a brief description. The items are: 'LoRA-X: Bridging Foundation Models with Training-Free Cross-Model Adaptation' (You saved this paper.), 'A Blog About Nothing' (You liked this paper.), 'A Blog About Nothing' (You viewed this paper.), 'CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge' (You saved this paper.), and 'CAMA: Enhancing Mathematical Reasoning in Large Language Models with Causal Knowledge' (You liked this paper.).

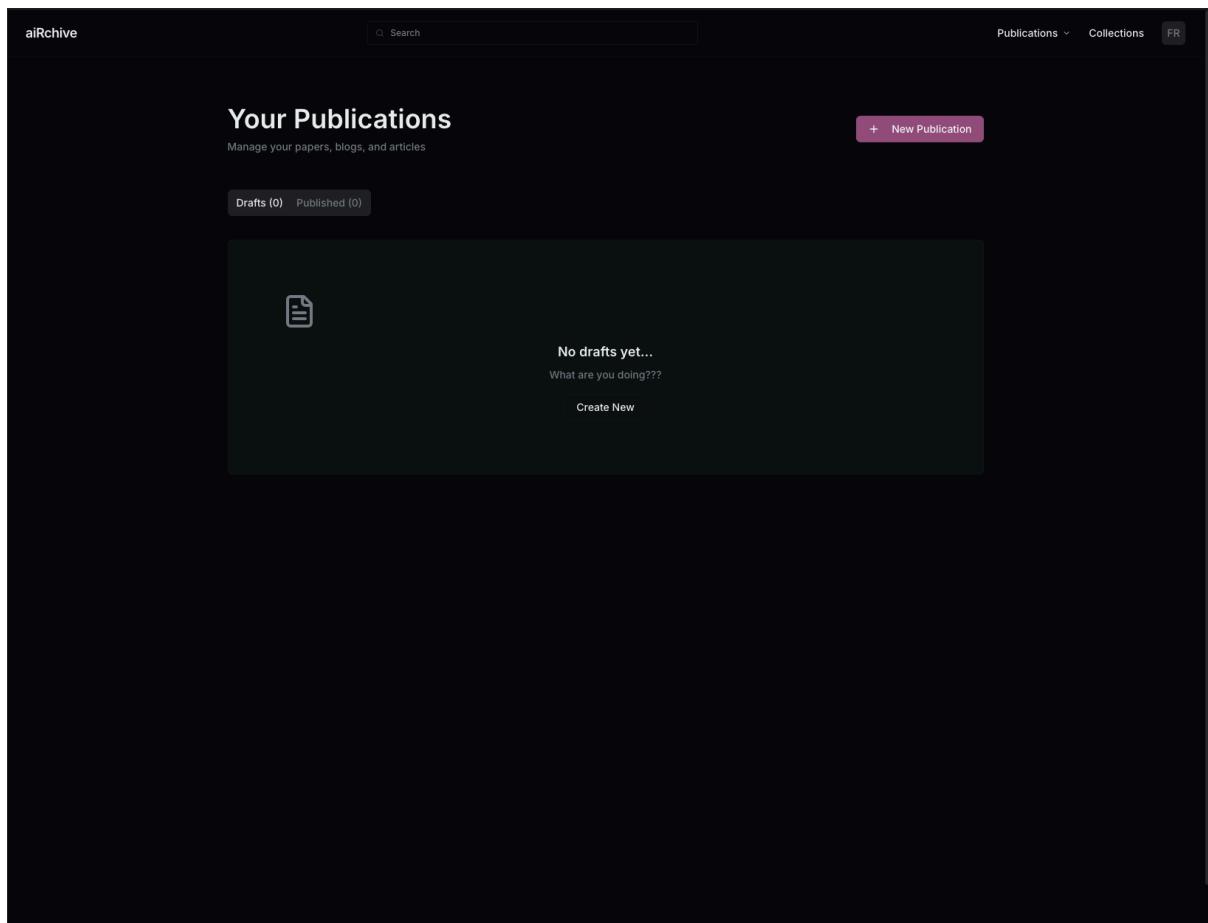
The screenshot shows a terminal window with a SQL query and its results. The query is:

```
1 • SELECT * FROM airchive.author_request ORDER BY requested_at DESC;
```

The results are displayed in a grid:

	account_id	status	requested_at
	31	APPROVED	2025-08-09 22:06:31
	29	PENDING	2025-08-09 05:13:33

I have no publications published or drafted on this account, and as proven by the database.



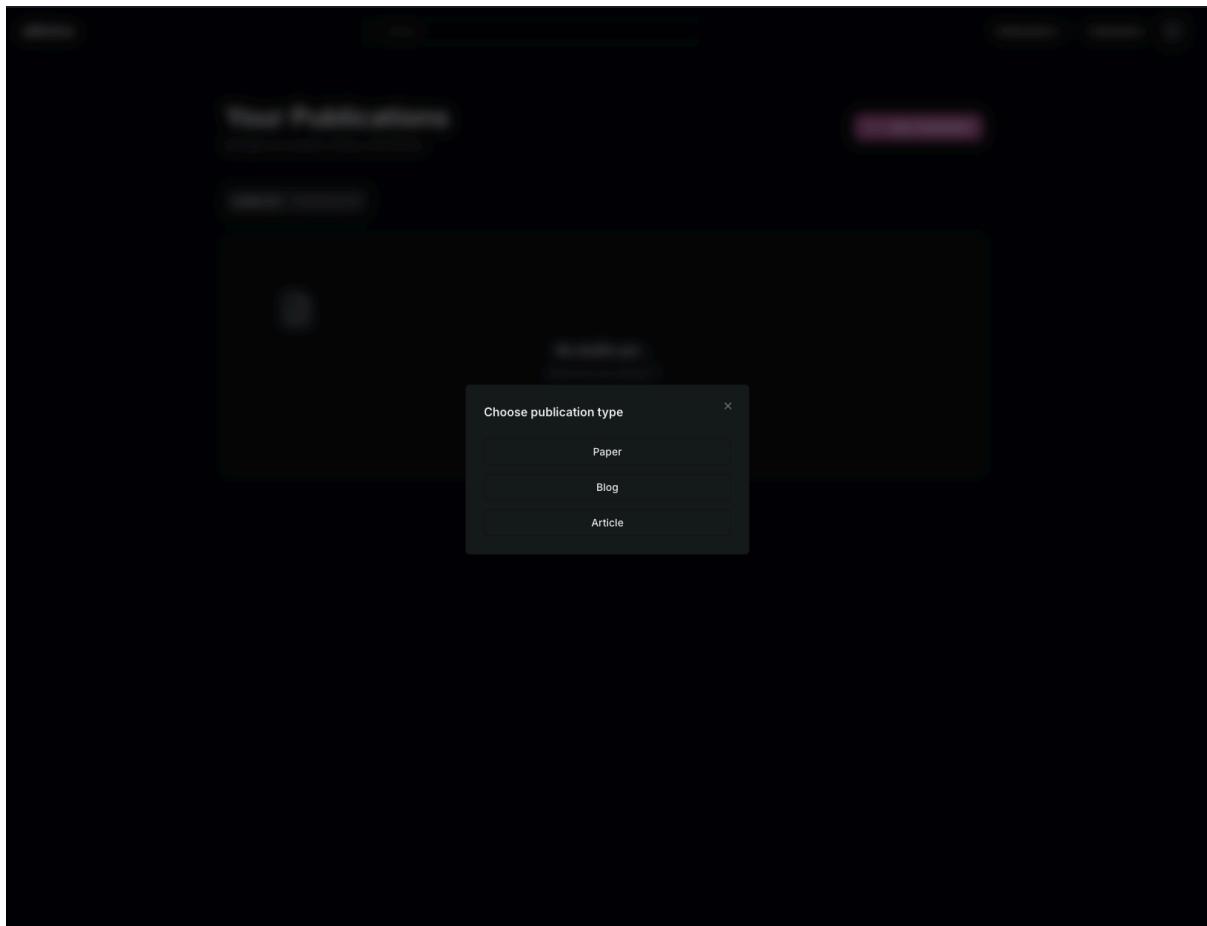
```
1  SELECT * FROM airchive.publication WHERE submitter_id = 31;
```

100% 24:1

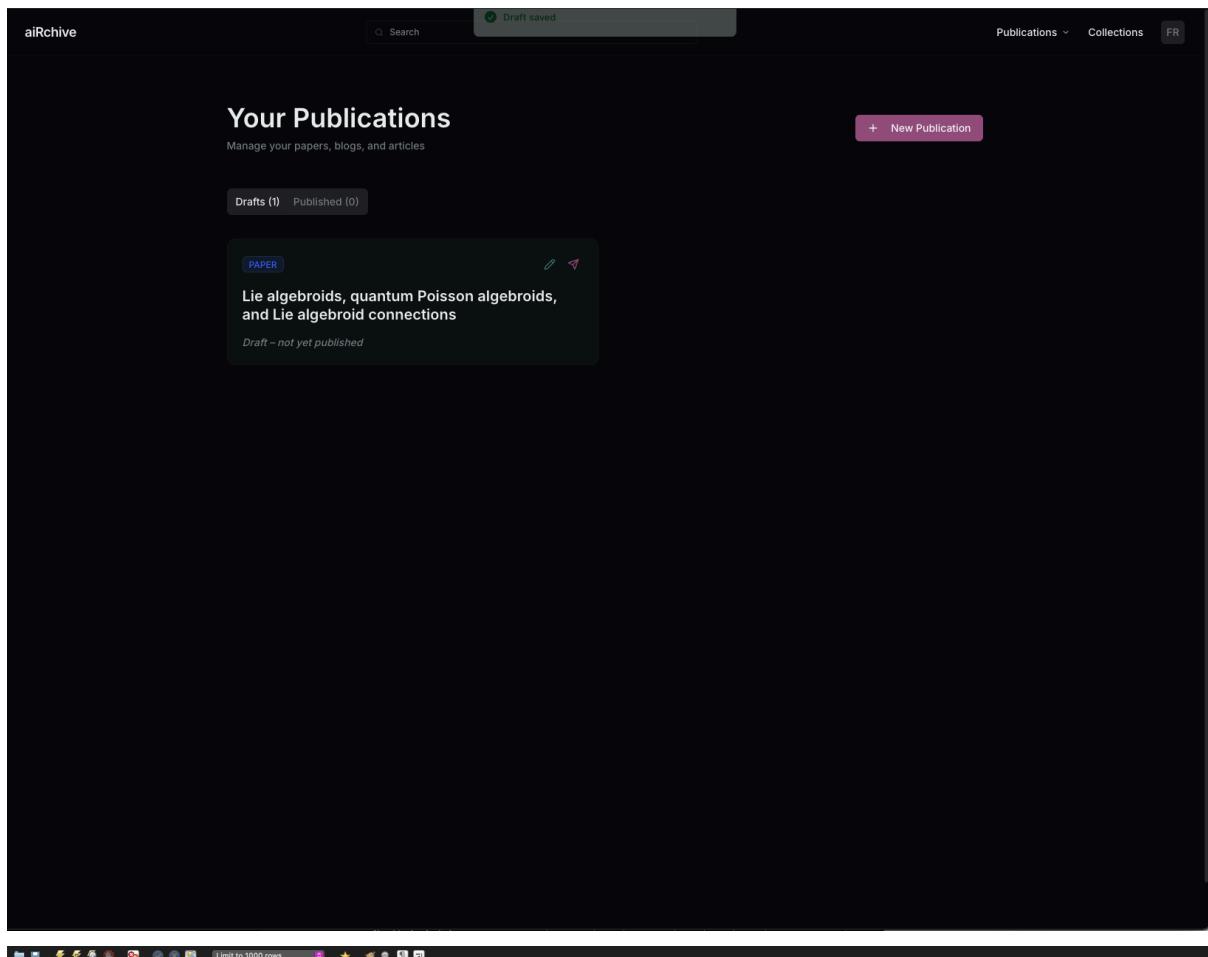
Result Grid Filter Rows: Search Edit: Export/Import:

pub_id	title	content	doi	url	kind	submitter_id	submitted_at	published_at	status
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Now I will create a draft of a paper and save the draft and show the new entry in the database. I will use a random publication on arXiv.



The screenshot shows a dark-themed web application interface for 'aiRhive'. At the top, there's a navigation bar with 'aiRhive' on the left, a search bar in the center, and 'Publications' and 'Collections' dropdown menus on the right. Below the header, a button labeled 'Back to My Publications' is visible. The main area is titled 'Create Draft' and has a sub-section 'PAPER'. A 'Title' field contains the text 'Lie algebroids, quantum Poisson algebroids, and Lie algebroid connections'. An 'Abstract' section follows, containing a block of text about Lie algebroids over ringed spaces, mentioning smooth manifolds, complex manifolds, analytic spaces, algebraic varieties, and schemes. It discusses the universal enveloping algebroid of a Lie algebroid, a filtration, and a correspondence between sheaves of quantum Poisson algebras and Lie algebroids. The text also mentions adjunction, bijective correspondence, and applications to twisted universal enveloping algebras, Lie-Rinehart algebras, and holomorphic Lie algebroids. It concludes with a discussion on deformation groupoids and hypercohomology. Below the abstract, there are sections for 'DOI' (with the URL <https://arxiv.org/abs/2508.05542>) and 'PDF URL' (with the URL <https://arxiv.org/pdf/2508.05542>). At the bottom right is a purple button labeled '+ Create Draft'.

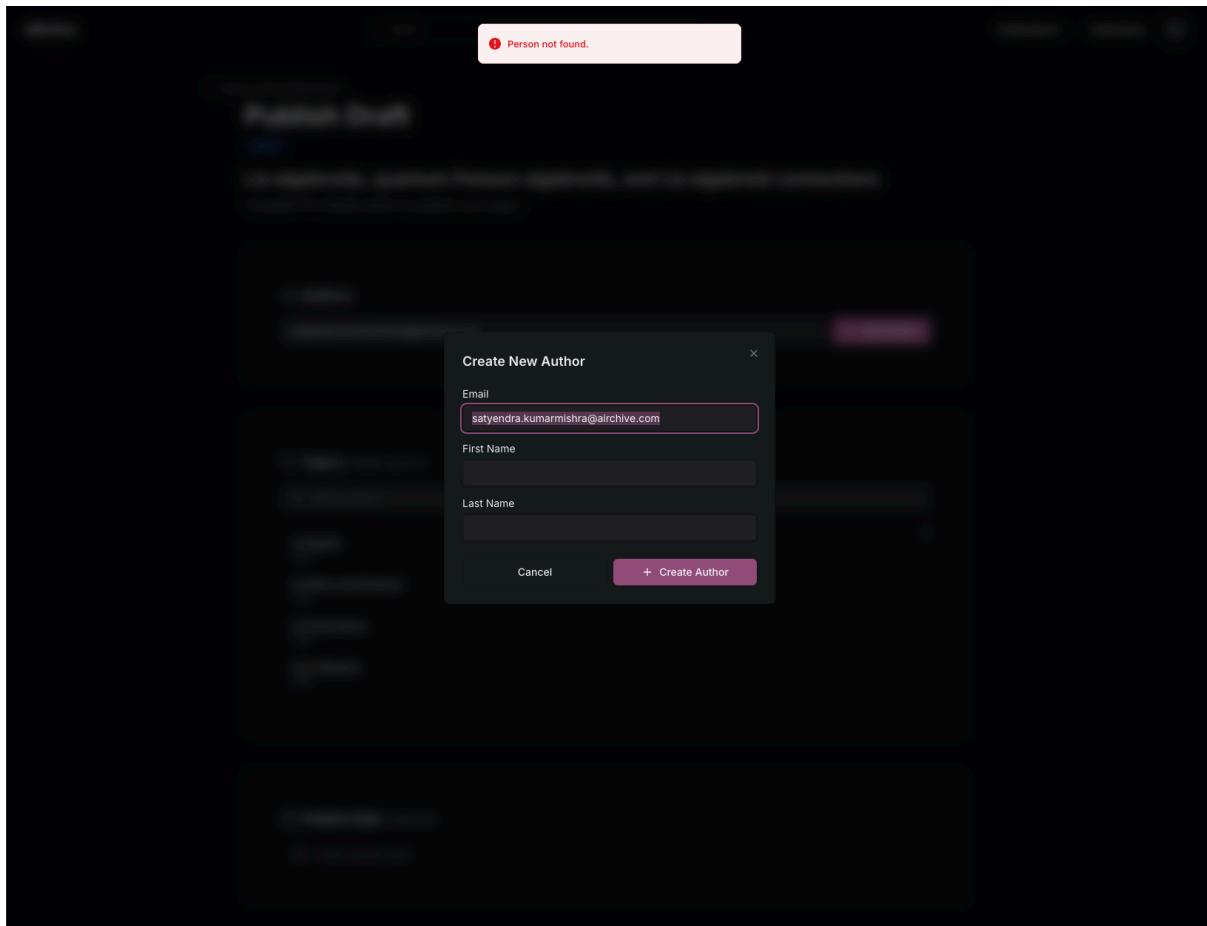


Result Grid										
Edit: Export/Import:										
pub_id	title	content	doi	url	kind	submitter_id	submitted_at	published_at	status	
20	Lie algebroids, quantum Pois... 	In this paper, we consider...	https://arxiv.org/abs/2508.05...	https://arxiv.org/pdf/2508.05...	PAPER	31	2025-08-09 22:12:56		DRAFT	

I will now showcase the publishing workflow.

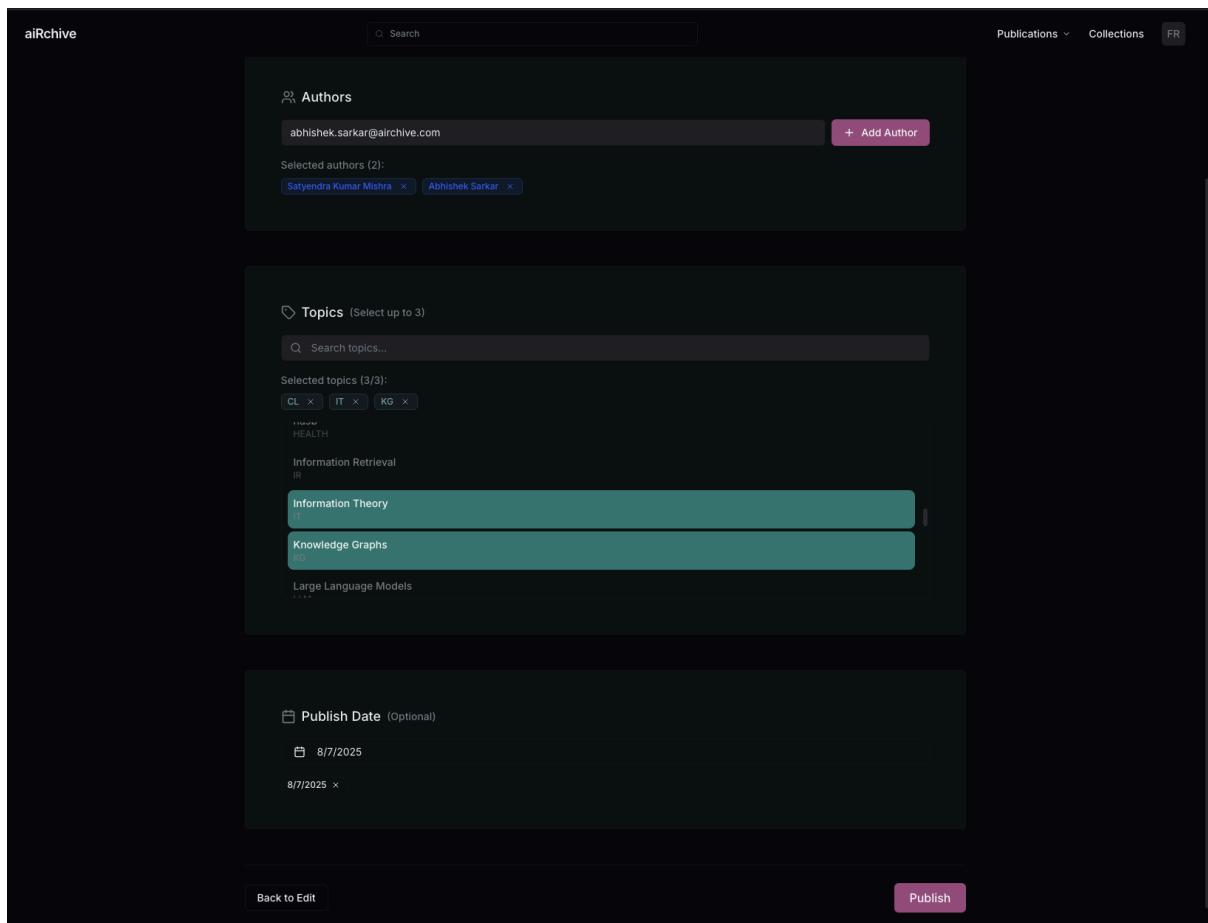
The screenshot shows the 'Publish Draft' section of the aiRchive platform. At the top, there's a navigation bar with 'aiRchive', a search bar, and links for 'Publications', 'Collections', and 'FR'. Below the navigation, a back-link '← Back to My Publications' and a title 'Publish Draft' are visible. A 'PAPER' button is highlighted. The main content area has a dark background with white text. It starts with a 'Authors' section containing a placeholder 'Enter author email address' and a '+ Add Author' button. Next is a 'Topics' section with a placeholder 'Search topics...' and a list of categories: 'AI Agents AGT', 'AI Ethics and Fairness ETH', 'AI Governance GOV', and 'AI in Robotics ROB'. Finally, there's a 'Publish Date' section with a placeholder 'Select publish date'.

Since the authors of this publication do not exist on the platform I will create a author profile for them.



The screenshot shows the aiRhive platform's 'Publish Draft' interface. At the top, a success message 'Author created and added successfully' is displayed. The main title of the publication is 'Lie algebroids, quantum Poisson algebroids, and Lie algebroid connections'. Below the title, it says 'Complete the details below to publish your paper'. The first section, 'Authors', shows two selected authors: 'Satyendra Kumar Mishra' and 'Abhishek Sarkar'. A button '+ Add Author' is available. The second section, 'Topics', allows selecting up to three topics from a list: 'AI Agents' (AGT), 'AI Ethics and Fairness' (ETH), 'AI Governance' (GOV), and 'AI in Robotics' (ROB). A search bar 'Search topics...' is provided. The third section, 'Publish Date', is optional and includes a 'Select publish date' button. The URL 'http://localhost:5173' is visible at the bottom left.

I will choose three random topics and a publishing date for this publication.



I will now publish and it will redirect to the rendered publication page. I will also show the newly updated and created publication entries.

Expand sidebar (^Z)

aRchive

Search

Publications ▾ Collections ER

PAPER KG CL IT

Lie algebroids, quantum Poisson algebroids, and Lie algebroid connections

Satyendra Kumar Mishra Abhishek Sarkar + August 7, 2025

0 0 0 DOI PDF

Abstract

In this paper, we consider Lie algebroids over commutative ringed spaces. Lie algebroids over ringed spaces unify the existing notion of Lie algebroids over smooth manifolds, complex manifolds, analytic spaces, algebraic varieties, and schemes. We show that the universal enveloping algebroid of a Lie algebroid possesses a natural filtration that yields a structure of a sheaf of quantum Poisson algebras. We establish a bijective correspondence between sheaves of quantum Poisson algebras and Lie algebroids. We show that this correspondence leads to an adjunction between the two categories. We discuss this bijective correspondence in particular cases of Lie algebroids over ringed spaces and highlight the subsequent results. To characterize non-flat Lie algebroid connections, we construct a sheaf of twisted universal enveloping algebras for a Lie algebroid using Lie algebroid (hyper) cohomology. We show that our construction yields some of the existing constructions for Lie-Rinehart algebras and holomorphic Lie algebroids. As another application, we study the deformation groupoid of a Lie algebroid using the second hypercohomology of the Lie algebroid.

```
1 SELECT * FROM archive.publication WHERE submitter_id = 31;
```

Result Grid Filter Rows: Search Edit Export/Import

pub_id	title	content	doi	url	kind	submitter_id	submitted_at	published_at	status
20	Lie algebroids, quantum Po... ↗ In this paper, we consid...	https://arxiv.org/abs/2508.05... ↗ https://arxiv.org/pdf/2508.05...	PAPER	31	2025-08-09 22:17:16	2025-08-07 07:00:00	PUBLISHED		

```

1 •  SELECT
2      'AUTHOR' AS type,
3      pa.pub_id,
4      pa.author_order AS sort_order,
5      p.person_id AS ref_id,
6      CONCAT(p.first_name, ' ', p.last_name) AS label,
7      p.identity_email AS extra
8  FROM archive.publication_author pa
9  JOIN archive.person p ON pa.person_id = p.person_id
10 WHERE pa.pub_id = 20
11
12 UNION ALL
13
14 SELECT
15     'TOPIC' AS type,
16     pt.pub_id,
17     NULL AS sort_order,
18     t.topic_id AS ref_id,
19     t.full_name AS label,
20     t.code AS extra
21  FROM archive.publication_topic pt
22  JOIN archive.topic t ON pt.topic_id = t.topic_id
23 WHERE pt.pub_id = 20;
24

```

100% 1:1

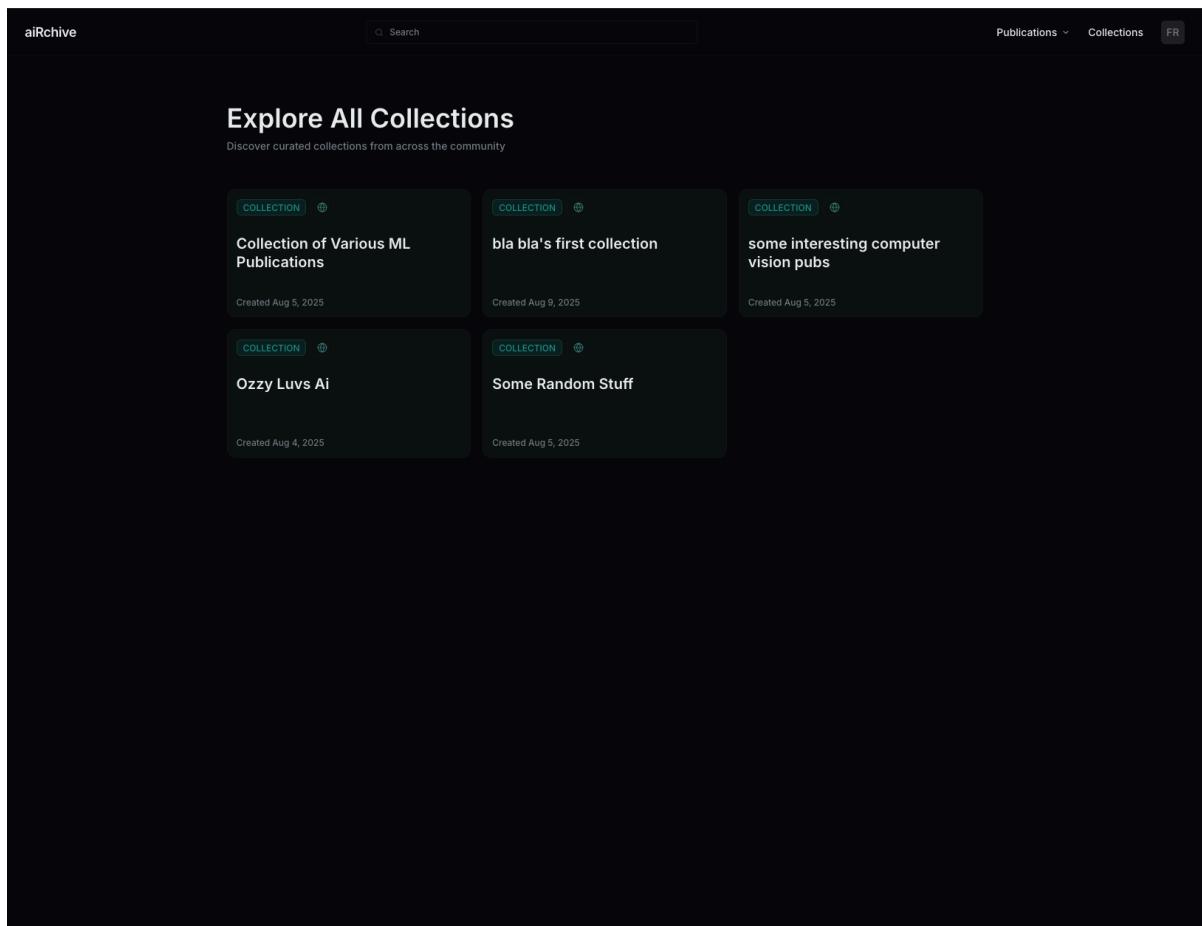
Result Grid Filter Rows: Search Export:

type	pub_id	sort_order	ref_id	label	extra
AUTHOR	20	1	61	Satyendra Kumar Mishra	satyendra.kumarmishra@air...
AUTHOR	20	2	62	Abhishek Sarkar	abhishek.sarkar@airhive.c...
TOPIC	20	NULL	10	Knowledge Graphs	KG
TOPIC	20	NULL	43	Computation and Language	CL
TOPIC	20	NULL	45	Information Theory	IT

As you can see the new publication was created and entries were created in the publication_author and publication_topic table to reflect the changes.

Collection Workflow

This is how browsing collections looks like.



Result Grid								
1 • SELECT * FROM airchive.collection WHERE is_public = true;								
100% 58:1								
Filter Rows:		Search		Edit:	Export/Import:			
collection_id	account_id	title		description		is_default	is_public	created_at
15	1	Ozzy Luvs Ai		<p>i don't think he does act...		0	1	2025-08-04 09:53:28
24	17	Collection of Various ML Pu...		<p>This collection contains...		0	1	2025-08-05 03:55:48
26	18	some interesting computer v...		<p>😊😊😊😊😊😊😊😊</p>		0	1	2025-08-05 05:00:31
27	18	Some Random Stuff		<p>random stufffffff</p><p>...</p>		0	1	2025-08-05 05:14:07
43	29	bla bla's first collection		<p>This is my first collection...</p>		0	1	2025-08-09 05:13:09
NULL	NULL	NULL		NULL		NULL	NULL	NULL

I will now show how the rendered collection page looks like.

The screenshot shows the aiRhive platform. At the top, there's a navigation bar with 'aiRhive', a search bar, and links for 'Publications', 'Collections', and 'FR'. Below the header, a collection titled 'Ozzy Luvs Ai' is displayed. The collection has a status message: 'i don't think he does actually 😊'. Under the heading 'Publications:', there are two items:

- ARTICLE**: 'Implementing MCP Servers in Python: An AI Shopping...' by Aekn Admal, published on Aug 3, 2025. It has 199 views and 4 likes.
- PAPER**: 'Trustworthy Distributed AI Systems: Robustness, Privacy...' by Wengi Wei, published on Feb 10, 2025. It has 146 views and 4 likes.

```
1 • | SELECT * FROM airchive.collection_item WHERE collection_id = 15;|
```

100% 65:1

Result Grid Filter Rows: Search Edit: Export/Import:

collection_id	pub_id	added_at
15	2	2025-08-05 05:38:32
15	7	2025-08-04 09:53:28
NUL	NUL	NUL

I will now show the collection creation workflow.

The screenshot shows the 'Your Collections' page on the aiRhive platform. At the top, there's a search bar and navigation links for 'Publications', 'Collections', and 'FR'. Below the header, a purple button says '+ New Collection'. The main area is titled 'DEFAULT COLLECTION' and contains a card for a collection named 'Saved'. The card includes the word 'COLLECTION' and 'Default' in small blue boxes, the title 'Saved', and the text 'Created Aug 9, 2025'. Below this, a section titled 'YOUR COLLECTIONS' displays a message: 'You haven't created any collections yet.'

The screenshot shows a MySQL command-line interface. A query is being run: '1 • SELECT * FROM airchive.collection WHERE account_id = 31;'. The results are displayed in a table titled 'Result Grid'. The table has columns: collection_id, account_id, title, description, is_default, is_public, and created_at. There is one row of data: collection_id 45, account_id 31, title Saved, description Your saved publications, is_default 1, is_public 0, and created_at 2025-08-09 21:15:25. The table also shows NULL values for other fields.

collection_id	account_id	title	description	is_default	is_public	created_at
45	31	Saved	Your saved publications	1	0	2025-08-09 21:15:25
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Now I will create a new collection with the new publication we created previously and a couple others.

The screenshot shows the 'Create Collection' page on the airRhive platform. At the top, there's a navigation bar with 'airRhive', a search bar, and links for 'Publications', 'Collections', and a language switch ('FR'). Below the header, a back-link '← Back to My Collections' and a title 'Create Collection' are visible.

The main form fields include:

- Title:** 'Final Report Test Collection' (input field)
- Description:** 'A cool new collection for our final report.' (text area with rich text editor icons like bold, italic, etc.)
- Visibility:** 'Public: Visible to everyone' (dropdown menu)
- Publications:** A section showing three items added to the collection:
 - 'trustworthy'
 - 'Added to Collection (3)'
 - 'Lie algebroids, quantum Poisson algebroids, and Lie algebroid connections'
 - 'Our Schema'
 - 'Trustworthy Distributed AI Systems: Robustness, Privacy, and Governance'

At the bottom right is a purple 'Create Collection' button, and at the bottom left is a 'Cancel' button.

airChive

Search

Publications ▾ Collections FR

Final Report Test Collection

A cool new collection for our final report.

Publications:

BLOG	PAPER	PAPER
Our Schema AV Aekn Admal • Aug 3, 2025 83 2	Trustworthy Distributed AI Systems: Robustness, Privacy... PRIV AI Governance Distributed AI Wengi Wei • Feb 10, 2025 146 4	Lie algebroids, quantum Poisson algebroids, and Lie... KG CL IT Satyendra Kumar Mishra • Aug 7, 2025 1 0

1 • `SELECT * FROM airchive.collection WHERE account_id = 31;`

Result Grid

collection_id	account_id	title	description	is_default	is_public	created_at
45	31	Saved	Your saved publications	1	0	2025-08-09 21:15:25
46	31	Final Report Test Collection	<p>A cool new collection for...</p>	0	1	2025-08-09 22:25:01
NULL	NULL	NULL	NULL	NULL	NULL	NULL

1 • `SELECT * FROM airchive.collection_item WHERE collection_id = 46;`

Result Grid

collection_id	pub_id	added_at
46	3	2025-08-09 22:25:02
46	7	2025-08-09 22:25:02
46	20	2025-08-09 22:25:01
NULL	NULL	NULL

As you can see it shows up on the explore collections page because it was set to public.

The screenshot shows the 'Explore All Collections' page on the aiRhive platform. The interface has a dark theme with white text and light-colored cards for each collection. At the top, there's a header with the aiRhive logo, a search bar, and navigation links for 'Publications', 'Collections', and a language switcher ('FR'). Below the header, the title 'Explore All Collections' is displayed, followed by a subtitle 'Discover curated collections from across the community'. Six collection cards are listed in two rows of three:

- Collection of Various ML Publications** (Created Aug 5, 2025)
- Final Report Test Collection** (Created Aug 9, 2025)
- bla bla's first collection** (Created Aug 9, 2025)
- Ozzy Luvs Ai** (Created Aug 4, 2025)
- some interesting computer vision pubs** (Created Aug 5, 2025)
- Some Random Stuff** (Created Aug 5, 2025)

Administrator Functionalities

As I showed before, an administrator can approve author requests.

The screenshot shows the 'Administration' section of the aiRhive platform. At the top, there is a search bar and navigation links for 'Publications', 'Collections', and a user icon. Below the header, the title 'Administration' is displayed, followed by the subtitle 'Manage author requests and topics'. A navigation bar at the top of the main content area includes 'Author Requests' and 'Topic Management' buttons. The main content is a table listing author requests:

EMAIL	REQUESTED AT	
blabla2@gmail.com	Aug 9, 2025, 5:13 AM	<button>Approve</button>
user@gmail.com	Aug 5, 2025, 1:26 PM	<button>Approve</button>
ash@gmail.com	Aug 5, 2025, 12:15 PM	<button>Approve</button>
thing3@gmail.com	Aug 2, 2025, 11:14 PM	<button>Approve</button>
thing2@gmail.com	Aug 2, 2025, 10:57 PM	<button>Approve</button>
user2@gmail.com	Aug 2, 2025, 10:47 PM	<button>Approve</button>

They can also create new topics.

The screenshot shows the airRhive Administration interface. At the top, there is a search bar and navigation links for 'Publications', 'Collections', and 'AA'. Below the header, the title 'Administration' is displayed, followed by the subtitle 'Manage author requests and topics'. A button '+ Create Topic' is visible. Below these, there are two tabs: 'Author Requests' and 'Topic Management', with 'Topic Management' being the active tab. The main content is a table with columns 'CODE' and 'NAME', listing various AI-related topics. Each row has a small icon next to it. The topics listed are: AGT (AI Agents), ETH (AI Ethics and Fairness), GOV (AI Governance), ROB (AI in Robotics), AGI (Artificial General Intelligence), AI (Artificial Intelligence), AV (Autonomous Vehicles), BENCH (Benchmarks and Evaluation), CAUS (Causality), CL (Computation and Language), CSS (Computational Social Systems), CV (Computer Vision), CTRL (Control and Dynamics), DM (Data Mining), DATA (Data Quality and Labeling), DL (Deep Learning), and DAI (Distributed AI).

CODE	NAME
AGT	AI Agents
ETH	AI Ethics and Fairness
GOV	AI Governance
ROB	AI in Robotics
AGI	Artificial General Intelligence
AI	Artificial Intelligence
AV	Autonomous Vehicles
BENCH	Benchmarks and Evaluation
CAUS	Causality
CL	Computation and Language
CSS	Computational Social Systems
CV	Computer Vision
CTRL	Control and Dynamics
DM	Data Mining
DATA	Data Quality and Labeling
DL	Deep Learning
DAI	Distributed AI

I will create a new topic with the code *test*. As you can see this currently does not exist within the database.

The screenshot shows a MySQL command-line interface. The command entered is: `1 • SELECT * FROM airrchive.topic WHERE code = 'test';`. Below the command, the output shows a result grid with three columns: 'topic_id', 'code', and 'full_name'. All three columns contain 'NULL' values, indicating that no rows were found for the specified query.

topic_id	code	full_name
NULL	NULL	NULL

The screenshot shows a dark-themed database interface. At the top, a modal window titled "Create New Topic" is open, containing fields for "Topic Code" (set to "test") and "Full Name" (set to "this is a test"), with a "Create Topic" button at the bottom. Below the modal, a SQL query is entered in the command line:

```
1 • SELECT * FROM airchive.topic WHERE code = 'test';
```

The interface includes a zoom level indicator (100%), a search bar, and various navigation and export/import buttons. A results grid displays the following data:

topic_id	code	full_name
48	TEST	this is a test
	NUL	NUL

A user can also delete a topic, I will delete this *test* topic to show that it can be deleted.

The image shows two screenshots of the aiRhive application. The top screenshot displays a list of topics in a dark-themed interface. The topics are listed in rows with columns for code and full name. A green message at the top right says 'Topic deleted'. The bottom screenshot shows a MySQL query editor with a dark-themed interface. It displays a single query: '1 • SELECT * FROM archive.topic WHERE code = 'test';'. Below the query is a result grid showing one row of data with columns for topic_id, code, and full_name, all containing 'NULL'.

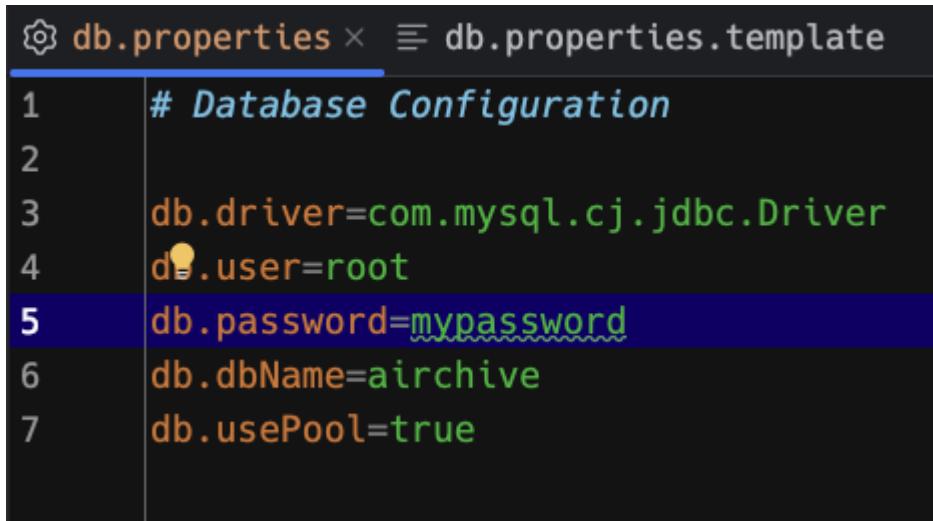
topic_id	code	full_name
NULL	NULL	NULL

That's more or less the main functionalities. There are slight variations depending on if you create a paper or blog publication, etc. But most of the functionalities follow the steps of these.

3.4 System Setup Instructions

1. We used Java 17, Node.js 24.2, MySQL 8.0.42, Maven, and Tomcat 9.0.42. So make sure that these tools are installed.

- a. <https://nodejs.org/en/download/current>
2. Create a new MySQL database named airchive
3. Import either airchive_schema.sql for just the schema or airchive_full.sql for the complete dump.
4. Copy server/src/main/resources/db.properties.template to db.properties in the same directory and set your configurations and credentials. Set db.usePool blank or to false if you don't want to use HikariCP connection pooling.



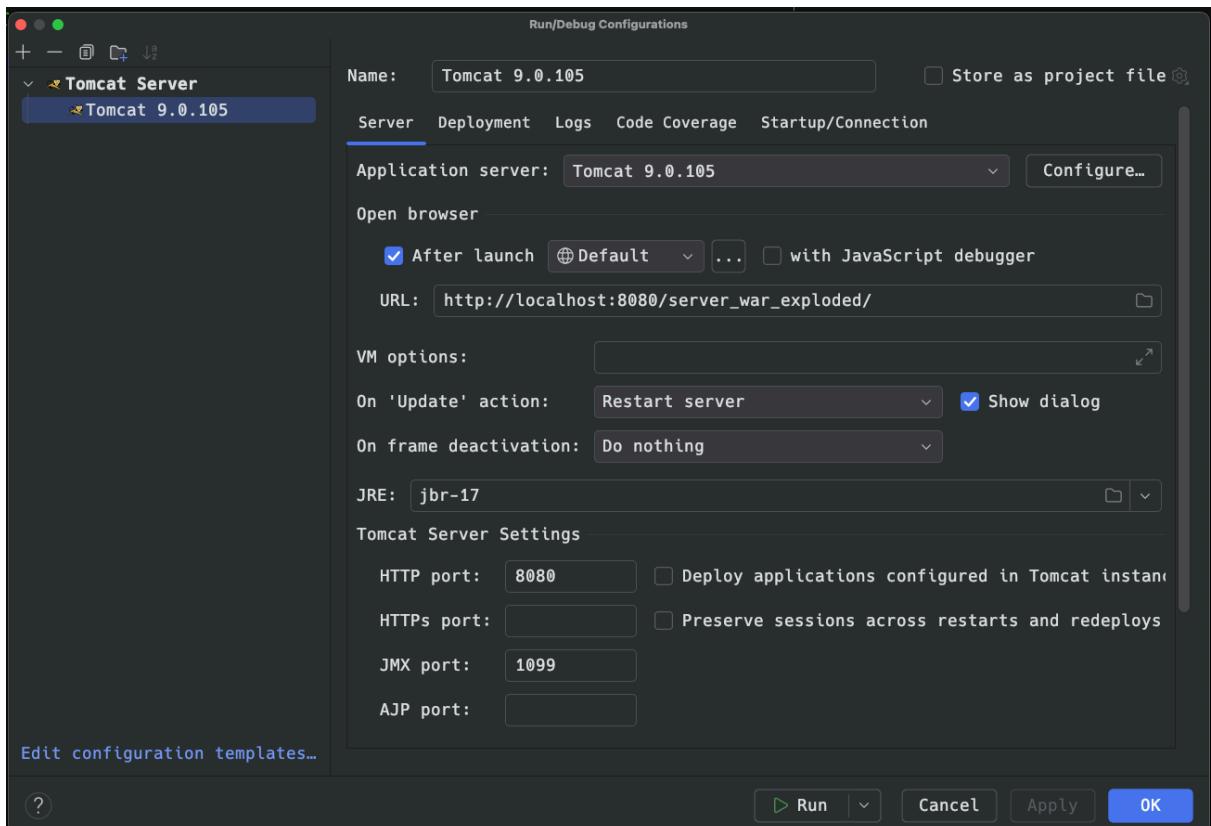
```
db.properties x  db.properties.template
# Database Configuration
db.driver=com.mysql.cj.jdbc.Driver
db.user=root
db.password=mypassword
db.dbName=airchive
db.usePool=true
```

5. Go to the server/ directory and run `mvn clean package`. Copy the generated `.war` into your Tomcat `webapps/` folder, or just set up your IDE to deploy the target package on Tomcat. I'm just deploying and running via my IDE.

```

~/ws_sjsu/ws_cs157a/airchive_dev/server
(zsh) ➔ mvn clean package
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::staticFieldBase has been called by com.google.inject.internal.aop.HiddenClassDefiner (file:/usr/local/Cellar/maven/3.9.10/libexec/lib/guice-5.1.0-classes.jar)
WARNING: Please consider reporting this to the maintainers of class com.google.inject.internal.aop.HiddenClassDefiner
WARNING: sun.misc.Unsafe::staticFieldBase will be removed in a future release
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.airchive:server >-----
[INFO] Building airchive Web Application 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ war ]-----
[INFO]
[INFO] --- clean:3.2.0:clean (default-clean) @ server ---
[INFO] Deleting /Users/aeknadmal/ws_sjsu/ws_cs157a/airchive_dev/server/target
[INFO]
[INFO] --- resources:3.3.1:resources (default-resources) @ server ---
[INFO] Copying 2 resources from src/main/resources to target/classes
[INFO]
[INFO] --- compiler:3.11.0:compile (default-compile) @ server ---
[INFO] Changes detected - recompiling the module! :source
[INFO] Compiling 66 source files with javac [debug target 17] to target/classes
[WARNING] location of system modules is not set in conjunction with -source 17
    not setting the location of system modules may lead to class files that cannot run on JDK 17
    --release 17 is recommended instead of -source 17 -target 17 because it sets the location of system modules automatically
[INFO]
[INFO] --- resources:3.3.1:testResources (default-testResources) @ server ---
[INFO] skip non existing resourceDirectory /Users/aeknadmal/ws_sjsu/ws_cs157a/airchive_dev/server/src/test/resources
[INFO]
[INFO] --- compiler:3.11.0:testCompile (default-testCompile) @ server ---
[INFO] No sources to compile
[INFO]
[INFO] --- surefire:3.2.5:test (default-test) @ server ---
[INFO] No tests to run.
[INFO]
[INFO] --- war:3.4.0:war (default-war) @ server ---
[INFO] Packaging webapp
[INFO] Assembling webapp [server] in [/Users/aeknadmal/ws_sjsu/ws_cs157a/airchive_dev/server/target/airchive]
[INFO] Processing war project
[INFO] Copying webapp resources [/Users/aeknadmal/ws_sjsu/ws_cs157a/airchive_dev/server/src/main/webapp]
[INFO] Building war: /Users/aeknadmal/ws_sjsu/ws_cs157a/airchive_dev/server/target/airchive.war
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time:  5.788 s
[INFO] Finished at: 2025-08-09T22:50:42-07:00
[INFO]
[INFO] -----
~/ws_sjsu/ws_cs157a/airchive_dev/server
(zsh) ➔

```



```

client - package-lock.json
server - db.properties

09-Aug-2025 22:52:40.793 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_HOME: /Users/aeknadmal/servers/apache-tomcat-9.0.105
09-Aug-2025 22:52:40.795 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --add-opens=java.base/java.lang.invoke=ALL-UNNAMED
09-Aug-2025 22:52:40.795 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --add-opens=java.base/java.lang.reflect=ALL-UNNAMED
09-Aug-2025 22:52:40.796 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --add-opens=java.base/java.util=ALL-UNNAMED
09-Aug-2025 22:52:40.796 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --add-opens=java.rmi=sun.rmi.transport=ALL-UNNAMED
09-Aug-2025 22:52:40.796 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Djava.util.logging.config.file=/Users/aeknadmal/libra
09-Aug-2025 22:52:40.796 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Djava.util.logging.manager=org.apache.juli.ClassLoad
09-Aug-2025 22:52:40.796 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dcom.sun.management.jmxremote=
09-Aug-2025 22:52:40.796 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dcom.sun.management.jmxremote.port=1099
09-Aug-2025 22:52:40.797 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dcom.sun.management.jmxremote.ssl=false
09-Aug-2025 22:52:40.797 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dcom.sun.management.jmxremote.password.file=/Users/ae
09-Aug-2025 22:52:40.797 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dcom.sun.management.jmxremote.access.file=/Users/aekn
09-Aug-2025 22:52:40.797 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Djava.rmi.server.hostname=127.0.0.1
09-Aug-2025 22:52:40.797 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Djdk.tls.ephemeralDHKeySize=2048
09-Aug-2025 22:52:40.797 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Djava.protocol.handler.pkgs=org.apache.catalina.webs
09-Aug-2025 22:52:40.797 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dsun.io.useCanonCaches=false
09-Aug-2025 22:52:40.798 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dorg.apache.catalina.security.SecurityListener.UMASK=
09-Aug-2025 22:52:40.798 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dignore.endorsed.dirs=
09-Aug-2025 22:52:40.798 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dcatalina.base=/Users/aeknadmal/Library/Caches/JetBra
09-Aug-2025 22:52:40.798 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: --Dcatalina.home=/Users/aeknadmal/servers/apache-tomcat
09-Aug-2025 22:52:40.814 INFO [main] org.apache.catalina.core.ApplicationListener.lifecycleEvent The Apache Tomcat Native library which allows using OpenSSL was not fo
09-Aug-2025 22:52:41.115 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["http-nio-8080"]
09-Aug-2025 22:52:41.135 INFO [main] org.apache.catalina.startup.Catalina.load Server initialization in [629] milliseconds
09-Aug-2025 22:52:41.202 INFO [main] org.apache.catalina.core.StandardService.startInternal Starting service [Catalina]
09-Aug-2025 22:52:41.203 INFO [main] org.apache.catalina.core.StandardEngine.startInternal Starting Servlet engine: [Apache Tomcat/9.0.105]
09-Aug-2025 22:52:41.222 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-8080"]
09-Aug-2025 22:52:41.251 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in [115] milliseconds
Connected to server
[2025-08-09 10:52:41,610] Artifact server:war exploded: Artifact is being deployed, please wait...
09-Aug-2025 22:52:43.343 INFO [RMI TCP Connection(2)-127.0.0.1] org.apache.jasper.servlet.TldScanner.scanJars At least one JAR was scanned for TLDs yet contained no TLD
09-Aug-2025 22:52:44.370 WARNING [RMI TCP Connection(2)-127.0.0.1] org.glassfish.jersey.server.wadl.WadlFeature.configure JAXBContext implementation could not be found.
[2025-08-09 10:52:45,269] Artifact server:war exploded: Deploy is deployed successfully
[2025-08-09 10:52:45,269] Artifact server:war exploded: Deploy took 3,659 milliseconds
09-Aug-2025 22:52:45.229 INFO [Catalina-utility-2] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [/Users/aeknadmal/servers/...
09-Aug-2025 22:52:51.753 INFO [Catalina-utility-2] org.apache.jasper.servlet.TldScanner.scanJars At least one JAR was scanned for TLDs yet contained no TLDs. Enable deb
09-Aug-2025 22:52:51.772 INFO [Catalina-utility-2] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [/Users/aeknadmal/serv

```

6. Go to the client/ directory. Run `npm install` to install necessary dependencies.

And then start the development server `npm run dev`.

```

~/ws_sjsu/ws_cs157a/airchive_dev/server
(zsh) → cd ..
~/ws_sjsu/ws_cs157a/airchive_dev
(zsh) → cd client
~/ws_sjsu/ws_cs157a/airchive_dev/client
(zsh) → npm install

up to date, audited 440 packages in 2s

108 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
~/ws_sjsu/ws_cs157a/airchive_dev/client
(zsh) →

```

```

Found 0 vulnerabilities
~/ws_sjsu/ws_cs157a/airchive_dev/client
(zsh) → npm run dev

> frontend-shadcn@0.0.0 dev
> vite

VITE v7.1.1  ready in 784 ms

→ Local:  http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help

```

7. Visit the app at <http://localhost:5173>
 - a. Currently we have Vite configured to proxy API requests to <http://localhost:8080/airchive/api>.
8. Make sure everything is running including the MySQL server and everything should work.

4 Lessons Learned

Aekn Admal

I learned a lot with this project. First and foremost I learned that it takes several tries to decide on how your code should be structured. Mainly because as you add more features you realize that it's much easier to do things another way rather than the way you are currently implementing them. I also learned a lot about how Java and databases work together through the Connection and PreparedStatement classes. I think all in all this was a good experience for learning how to build a full-stack application. Another big takeaway is that it takes time to improve your project as you only see the need for changes as you continue working on it. I did the initial prototyping and everything related to publications and recommendations. I really enjoyed the project and learned a lot.

Huy Tran

I am in charge of the collection functions, and some of the topic functions of the website. But, since this is also the first time I made a website, everything was challenging. For the collection functions, I find it hard to get used to the SQL stuff, especially when incorporating them into normal Java code on the backend. The collection creation and deletion were the easy ones, the real hard parts are the functions to find and add publications into the collections. There are multiple find functions, like findByAccount, findById, findRecommendedPublic, etc. and they were the bane of my existence for my part of the project.

Eda Koker

User and admin functions were my part of the project. I have never made a website before, so everything was new to me. Since I haven't been in the U.S for long, and I don't use english much in my country, I did have a hard time communicating with my teammates, I had to double check to make sure I didn't misunderstand things. For the project, I learnt how to use packages and collections related to SQL (connection and prepared statement). Overall, I really tried my best for this project, and my teammates helped me a lot too. They made sure that I was on the same page, or close to it, so that I could contribute.