

Extraction Report: 07_auth_service

Property	Value
Domain	Code
Format	.ts
Text length	2,246 chars
Sentences	30

Entities

Spacy NER: 15 entities (504ms)

Entity	Label	Score
JWT	ORG	
TTL	ORG	
3600 seconds	TIME	
one	CARDINAL	
AuthResult	ORG	
AuthError	ORG	
max 5	LAW	
1	CARDINAL	
Validate	ORG	
Invalid	PERSON	
2	CARDINAL	
3	CARDINAL	
3600	CARDINAL	
Database	PERSON	

Gliner NER: 96 entities (418ms)

Entity	Label	Score
AuthService	system_component	0.93
authentication	action	0.36
Database	system_component	0.80
database	database_table	0.48
TokenService	system_component	0.88
./tokens	system_component	0.51
lifecycle	event_type	0.42
State machine	state	0.89
idle	state	0.56
authenticating	action	0.82
authenticated	action	0.41
failed	constraint	0.37
Precondition	constraint	0.81
connection	requirement	0.34
calling	action	0.40
authenticate	action	0.82
JWT	protocol	0.59
TTL	constraint	0.49
At most one active session	constraint	0.68
state	state	0.92
authenticated	state	0.94

Entity	Label	Score
failed	state	0.91
Authenticate	action	0.43
credentials	requirement	0.83

Stanza NER: 11 entities (1237ms)

Entity	Label	Score
AuthService	ORG	
JWT	ORG	
3600 seconds	TIME	
one	CARDINAL	
5	CARDINAL	
1	CARDINAL	
2	CARDINAL	
3	CARDINAL	
3600	CARDINAL	
TokenService	ORG	
CryptoModule	ORG	

Flair NER: 1 entities (170ms)

Entity	Label	Score
JWT	MISC	1.00

Relations (SRL)

30 SRL frames (84ms)

Verb	Agent	Patient	Other
manage	AuthService	authentication lifecycle	
authenticate			failed

•

```

| | | establish | Database connection | | ARGM-MOD=must | | call | | authenticate | | | return | valid JWT | |
ARGM-LOC=On success, ARG2=with TTL of 36 | | authenticate | | | | | fail | * / export class AuthService { pri |
= | | | authenticate | | user with credentials | | | be | * @precondition username and password | non - empty strings
* | ARGM-MOD=must, ARGM-ADV=with token | | return | @postcondition | AuthResult | | | be | credentials
| | | | @throw | AuthError | | | | exceed | rate limit | | | | throw | Promise < AuthResult > { if ( ! us | new
Error('Credentials required | | | fail | if ( ! user || ! await this.verifyPasswo | | | | return | | { success : | | | revoke
| if ( ! user || ! await this.verifyPasswo | existing sessions | | | create | | new one await this.tokenService.re | |
| generate | Step 3 : | new token // Side effect | | | write | | session record to database const | | | await | = |
this.tokenService.generate(user.id , | | | authenticate | | = await this.tokenService.generate(user. | | | invalidate
Logout | current session | | | be | User | | ARGM-MOD=must, ARGM-LOC=in ' authenticat | | revoke | token | | |

```

Enriched Extraction (batch-enrich)

Single GPU call: 96 entities, 30 roles, 30 SRL frames (878ms)

Latency Comparison

Method	Latency
Separate (GLiNER 418ms + GLiClass 395ms + SRL 84ms)	897ms
batch-enrich (unified)	878ms
Speedup	1.0x

Per-Sentence Enriched Results (sample)

Sentence	Entities	Role	SRL Frames
// AuthService — Handles user authentication and session man *	9	State (0.18)	1
* State machine: idle → authenticating → authenticated		5	State Change (0.44)
Precondition:	1	Condition (0.38)	0
Database connection must be established before calling authe	3	Condition (0.31)	2
* Postcondition: On success, a valid JWT is returned with TT	2	Postcondition (0.63)	1
* Invariant: At most one active session per user at any time	1	Invariant (0.98)	0
*/			

```
export class AuthService { private state: 'idle' | 'aut | 7 | State (0.66) | 2 | | ** | 0 | Effect (0.47) | 0 | | * | 0 | Effect (0.46) | 0 | | Authenticate a user with credentials. | 2 | Condition (0.19) | 1 | | *
```

- @precondition username and password must be non-empty | 6 | Postcondition (0.78) | 3 | | valid
- @throws AuthError | 0 | Exception (0.83) | 1 | | if rate limit exceeded (max 5 attempts per minute) */ a | 4 | Action (0.25) | 1 | | string, password: string): | 0 | Condition (0.17) | 0 | | Promise { if (!username || !password) throw | 5 | State (0.86) | 1 |

Entity Type Distribution (enriched)

Label	Count
action	26
state	19
system_component	15
constraint	11
requirement	9
database_column	6
database_table	5
data_structure	2
event_type	1
protocol	1
metric	1

Role Distribution (enriched)

Role	Count
State (State)	8
Condition (State)	6
Effect (Event)	6
Postcondition (Programming)	3
Action (Event)	2
State Change (Other)	1

Role	Count
Invariant (Programming)	1
Exception (Discourse)	1
Object (Event)	1
Side Effect (Other)	1

QLang Sentences

Gliclass: 30 classifications (395ms)

Discourse (1)

- [Exception] (0.83) valid
 - @throws AuthError...

Event (9)

- [Effect] (0.47) **...
- [Effect] (0.46) *...
- [Action] (0.25) if rate limit exceeded (max 5 attempts per minute) */ async authenticate(us...
- [Object] (0.47) = 'authenticating';
 - // Step 1: Validate credentials against database co...
- [Effect] (0.46) *...
- [Effect] (0.46) *...
- [Action] (0.19) */ async logout(token: string):...
- [Effect] (0.26) Dependency:...

Other (2)

- [State Change] (0.44) *
- State machine: idle → authenticating → authenticated | failed
-

*...

- [Side Effect] (0.65) Step 3: Generate new token // Side effect: writes session record to database...

Programming (4)

- [Postcondition] (0.63) * Postcondition: On success, a valid JWT is returned with TTL of 3600 seconds....
- [Invariant] (0.98) * Invariant: At most one active session per user at any time....
- [Postcondition] (0.78) *
 - @precondition username and password must be non-empty strings
 - @postc...
- [Postcondition] (0.51) Postcondition: State transitions to 'idle', token is revoked....

State (14)

- [State] (0.18) // AuthService — Handles user authentication and session management

import { Da...

- [Condition] (0.38) Precondition:...
- [Condition] (0.31) Database connection must be established before calling authenticate()....
- [State] (0.66) */ export class AuthService { private state: 'idle' | 'authenticating' | 'auth...
- [Condition] (0.19) Authenticate a user with credentials....
- [Condition] (0.17) string, password: string):...
- [State] (0.86) Promise { if (!username || !password) throw new Error('Credentia...

- [State] (0.71) if (!user || !await this.verifyPassword(password, user.hash)) { this.state...

Qualtron: 6 classifications (13824ms)

Core (1)

- [Concept] (0.90) *...

Other (1)

- [Code] (0.95) Promise { await this.tokenService.revoke(token); this.state 2 = 'i...

Programming (2)

- [Type] (0.95) // AuthService — Handles user authentication and session management...
- [Invariant] (1.00) Invariant: At most one active session per user at any time....

State (2)

- [Condition] (0.95) @precondition username and password must be non-empty strings...
- [State] (0.95) = 'authenticating';...

QHG Process Models

FSM (2081ms)

Could not parse structured output

None

BPMN (2458ms)

Could not parse structured output

None

DFG (3783ms)

Could not parse structured output

None

KnowledgeState (5240ms)

Could not parse structured output

None

CNL / QNR2 Rules

3 rule patterns detected

Obligation (3)

- must be established before calling authenticate().
- must be non-empty strings
- must be in 'authenticated' state.

Heuristic Facts & Rules

0 facts, 7 rules

Rules (sample)

- [obligation] * Precondition: Database connection must be established before calling authenticate()....
- [obligation] * @precondition username and password must be non-empty strings...
- [conditional] * @postcondition returns AuthResult with token if credentials are valid...
- [conditional] * @throws AuthError if rate limit exceeded (max 5 attempts per minute)...
- [obligation, conditional] if (!username || !password) throw new Error('Credentials required');...
- [conditional] if (!user || !await this.verifyPassword(password, user.hash)) {...
- [obligation] * Precondition: User must be in 'authenticated' state....

Topics (Gensim LDA)

- **Topic 0:** {'word': 'user', 'weight': 0.0811}, {'word': 'this', 'weight': 0.0618}, {'word': 'credentials', 'weight': 0.0425}, {'word': 'await', 'weight': 0.0425}, {'word': 'password', 'weight': 0.0232}, {'word': 'username', 'weight': 0.0232}, {'word': 'success', 'weight': 0.0232}, {'word': 'error', 'weight': 0.0232}
- **Topic 1:** {'word': 'token', 'weight': 0.0636}, {'word': 'state', 'weight': 0.0514}, {'word': 'tokenservice', 'weight': 0.0513}, {'word': 'database', 'weight': 0.0391}, {'word': 'this', 'weight': 0.0391}, {'word': 'idle', 'weight': 0.0269}, {'word': 'revoke', 'weight': 0.0269}, {'word': 'authenticated', 'weight': 0.0269}
- **Topic 2:** {'word': 'idle', 'weight': 0.0526}, {'word': 'authservice', 'weight': 0.0362}, {'word': 'authenticated', 'weight': 0.0362}, {'word': 'state', 'weight': 0.0362}, {'word': 'failed', 'weight': 0.0362}, {'word': 'authenticating', 'weight': 0.0362}, {'word': 'postcondition', 'weight': 0.0197}, {'word': 'authenticate', 'weight': 0.0197}
- **Topic 3:** {'word': 'username', 'weight': 0.0544}, {'word': 'credentials', 'weight': 0.0544}, {'word': 'password', 'weight': 0.0544}, {'word': 'string', 'weight': 0.0374}, {'word': 'authresult', 'weight': 0.0374}, {'word': 'authenticate', 'weight': 0.0374}, {'word': 'async', 'weight': 0.0204}, {'word': 'promise', 'weight': 0.0204}
- **Topic 4:** {'word': 'generate', 'weight': 0.0442}, {'word': 'tokenservice', 'weight': 0.0442}, {'word': 'user', 'weight': 0.0442}, {'word': 'await', 'weight': 0.0442}, {'word': 'step', 'weight': 0.0442}, {'word': 'new', 'weight': 0.0442}, {'word': 'token', 'weight': 0.0442}, {'word': 'this', 'weight': 0.0442}

Summary (Sumy LexRank)

```
// AuthService — Handles user authentication and session management /** * Authenticate a user with credentials. // Step 1: Validate credentials against database const user = await this.db.findUser(username); if (!user || !await this.verifyPassword(password, user.hash)) { this.state = 'failed'; return { success: false, error: 'Invalid credentials' }; } * Precondition: User must be in 'authenticated' state. */ async logout(token: string): Promise { await this.tokenService.revoke(token); this
```