

# Лаб 13 – Автомат тест үүсгэлт (FastAPI Todo API)

## Зорилго

Search-based (Pynguin), property-based (Hypothesis), AI/LLM-д суурилсан тест үүсгэлтийн аргуудыг нэг төсөл дээр туршиж харьцуулах.

Coverage хэмжих, илэрсэн асуудлыг тайлагнах, автоматжуулалтын үр ашгийг дүгнэх.

## Төслийн сонголт

Сонгосон төсөл: `blog_tutorials/FastAPI_Todo_App/ToDo_Part2`

Шалтгаан: жижиг CRUD гадаргуу, FastAPI + Pydantic стек, in-memory өгөгдөлтэй тул тестлэх, рефактор хийхэд хурдан.

## Онолын товч тайлбар

Search-based testing (Pynguin): Генетик хайлтаар API/функцийн оролт, дуудагдал хослолыг судалж тест үүсгэнэ. Framework-н хамааралгүй цэвэр Python логик дээр хамгийн үр дүнтэй.

Property-based testing (Hypothesis): Олон янзын санамсаргүй оролтыг property-аар тодорхойлж, инвариант/алдааны замыг системтэй илрүүлнэ. Shrinking механизмаар бага жишээгээр алдааг тайлбарлах давуу талтай.

LLM-based testing: Том хэлний загварууд (Copilot/ChatGPT гэх мэт) кодын утгыг тайлбарлан pytest бүтэц, mock/fixture санал болгодог. Хурдан эхлэл өгөх ч хүний баталгаажуулалт шаардлагатай.

## Хийгдсэн ажил

Логикуудыг `service.py` файлд салгаж, FastAPI-г `main.py`-д нимгэлж Pynguin-д нийцтэй болгосон.

Pynguin ашиглан `service.py` дээр smoke тестүүд үүсгэсэн (`tests_pynguin/`).

Hypothesis ашиглан 6 property бичиж CRUD инвариант, алдааны замыг шалгасан (`tests_hypothesis/`).

LLM prompt ашиглан TestClient-тэй API тестүүд бичиж CRUD болон 404 замуудыг хамарсан (`tests_ai/`).

Coverage HTML тайлангуудыг coverage\_penguin/html, coverage\_hypothesis/html, coverage\_ai/html замд хадгалсан.

Лаб тайланг report\_Ovog\_Ner.docx (өөрийн нэрээр солих) болон README-г монголоор бичсэн.

## Тестүүд ба үр дүн

Penguin: service-д ~36% line coverage (smoke).

Hypothesis: service-д ~96% line coverage; KeyError, UUID/ID инвариант, required keys, unique ID.

AI (LLM): main 94%, service 100%, нийт ~96% line coverage; CRUD + 404 замууд

TestClient-ээр.

## Алдаа, асуудал, шийдэл

Penguin FastAPI/Pydantic дээр pickling алдаа өгсөн → логикийг цэвэр Python service.py болгож салгасан.

In-memory state-ийн нөлөөг барасгахын тулд тест бүрт reset\_todos fixture ашигласан.

Windows long-path алдаа гарч болзошгүй → шаардлагатай багцыг салгаж суулгах эсвэл long-path support идэвхжүүлэх.

## Ажилгааны хэсгээс

The screenshot shows a REST client interface with the following sections:

- GET /todos/** Get All Todos (selected)
- POST /todos/** Create Todo
- Parameters**: No parameters
- Request body** required: application/json (selected)
- Example Value**:

```
{  "title": "Do home work",  "description": "Complete all my works of University",  "completed": false}
```
- Responses**
- Curl**:

```
curl -X 'POST' \  'http://127.0.0.1:8000/todos/' \  -H 'accept: application/json' \  -H 'Content-Type: application/json' \  -d '{  "title": "Do home work",  "description": "Complete all my works of University",  "completed": false  }'
```
- Request URL**: http://127.0.0.1:8000/todos/
- Server response**: Code Details

Code	Details	
200	<div>Response body</div> <div><pre>{   "id": "e1bce5c5-b330-4a79-af8f-87de37d50736",   "title": "Do home work",   "description": "Complete all my works of Univercity",   "completed": false,   "created_at": "2025-12-12T22:02:55.807105" }</pre></div> <div>Response headers</div> <div><pre>content-length: 180 content-type: application/json date: Fri, 12 Dec 2025 14:02:55 GMT server: uvicorn</pre></div>	
Responses		
Code	Description	Links
200	Successful Response	No links
Media type		
application/json		
Controls Accept header.		
Example Value   Schema		
<pre>{   "id": "string",   "title": "string",   "description": "string",   "completed": true,   "created_at": "2025-12-12T14:03:11.1212" }</pre>		
422	Validation Error	No links
Media type		
application/json		
Example Value   Schema		
<pre>{   "detail": [     {       "loc": [</pre>		

Name	Description
todo_id <span>required</span>	e1bce5c5-b330-4a79-af8f-87de37d50736
string	
(path)	
Execute Clear	
Responses	
Curl	
<pre>curl -X 'GET' \   'http://127.0.0.1:8000/todos/e1bce5c5-b330-4a79-af8f-87de37d50736' \   -H 'accept: application/json'</pre>	
Request URL	
http://127.0.0.1:8000/todos/e1bce5c5-b330-4a79-af8f-87de37d50736	
Server response	
Code	Details
200	<div>Response body</div> <div><pre>{   "id": "e1bce5c5-b330-4a79-af8f-87de37d50736",   "title": "Do home work",   "description": "Complete all my works of Univercity",   "completed": false,   "created_at": "2025-12-12T22:02:55.807105" }</pre></div> <div>Response headers</div> <div><pre>content-length: 180 content-type: application/json date: Fri, 12 Dec 2025 14:04:00 GMT server: uvicorn</pre></div>
Responses	
Code	Description
	Links

## Дүгнэлт

Property-based (Hypothesis) ба AI тестүүд өндөр coverage, алдааны замуудыг илүү сайн илрүүлсэн.

Search-based (Pynguin) нь framework-салгасан логик дээр л ашигтай; архитектурын салгалт чухал.

Ирээдүйд: core логикийг framework-гүй байлгах, property-based шалгалтыг AI-санаа авч өргөжүүлэх, search-based хэрэгслийг туслах байдлаар ашиглах.\*\*\*