

PatientGenerator: an R package to create synthetic test patients using LLMs

Cesar Barboza¹, Adam Black¹, Ger Inberg and Ross Williams¹

¹Erasmus Medical Center, Rotterdam

Software demo: <https://github.com/OHDSI/PatientGenerator>

Background

PatientGenerator is an R package designed to create synthetic, patient-level test datasets in the OMOP Common Data Model (CDM) using LLMs. This approach allows data scientists and epidemiologists to effectively create specific test cases for unit and integration testing in pharmacoepidemiologic studies. Users can generate detailed patient profiles and clinical events via natural language prompts, producing a structured response from an LLM that can then be pushed directly to an OMOP-CDM test database using TestGenerator. Additionally, PatientGenerator includes an interactive Shiny application to visually inspect and modify the synthetic data output from the LLM.

Methods

PatientGenerator leverages the ability of LLM providers to emit structured responses through a JSON schema and the Ellmer package that allows a seamless connection to this type of platforms. The tool utilizes a schema that represents a complete specification of the OMOP-CDM v5.4. From a single class instance, the user opens a chat connection to an LLM provider and send detailed prompts specifying the logic and details of the desired synthetic population. This tool also allows users to chain prompts to modify the population through the LLM in a single session. PatientGenerator also is programmed to be highly integrated with TestGenerator and the testthat framework. PatientDesigner is a Shiny application that uses the D3 JavaScript library for interactive visualizations. Additionally, it integrates with Hecate, a semantic search tool that employs AI to query specific concepts.

```
patientGenerator$prompt(  
  "Population (person table):  
    - 10 adult patients  
    - 5 female  
    - 5 male  
  
  Observation Period:  
    - Start date between date of birth and 2025-12-31  
  
  Condition Occurrence:  
    - All patients must have Diabetes (condition_concept_id: 201826)  
    - Start date between 2015-01-01 and 2020-12-31  
  
  Drug Exposure:  
    - All patients must have Semaglutide (drug_concept_id: 19079450)  
    - Exposure within 30 days post-index date  
  
  Measurement:  
    - All patients must have Fasting glucose (measurement_concept_id: 3018251)  
  
  Procedure Occurrence:  
    - 50% of patients must have Amputation of toe (procedure_concept_id: 4159766)  
  
  Output Requirements:  
    - Populate only the tables specified in this prompt"  
)
```

Figure 1. PatientGenerator prompt interface

Results

PatientGenerator is actively used by data scientists and epidemiologists at the DARWIN EU project. It has been used in both off-the-shelf and complex studies to facilitate the creation of specific test scenarios for these types of studies. In conjunction with PatientDesigner, researchers can fine-tune precise test sets with less effort through an interactive visual interface. The prompts allow for the creation of test sets using natural language, and the visual support decreases the amount of time invested in creating and verifying the validity of a test set, especially when the researcher must create complex or unique edge cases. Moreover, the inclusion of tools such as Hecate for concept searches makes it easier to find codes in real time.

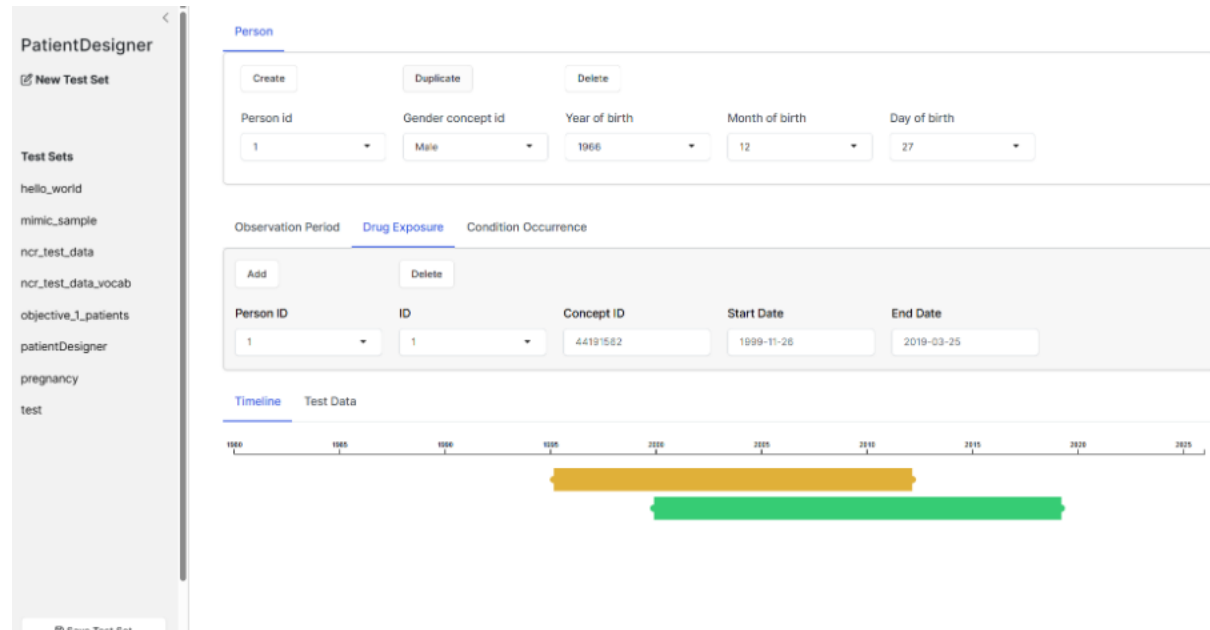


Figure 2. PatientDesigner interface, the user can move, elongate and zoom bars that represent records in tables of the OMOP-CDM

Conclusion

PatientGenerator contributes to the integration of unit testing in data projects that use standard OMOP-CDM tables, leveraging the mainstream adoption of AI tools within the OHDSI ecosystem. Furthermore, it provides a graphical interface that allows users with no programming background to define the specific patient profiles required for a particular study. In addition, its integration with AI tools offers the opportunity to experiment with and evaluate these new technologies within the OMOP-CDM context.

References

1. Bostock M, Ogievetsky V, Heer J. D3: Data-driven documents. IEEE Trans Vis Comput Graph. 2011;17(12):2301-2309. doi: 10.1109/TVCG.2011.185.
2. Darwin EU. TestGenerator: Integration unit tests for studies on the OMOP-CDM [Internet]. GitHub. [cited 2026 Jun 5]. Available from: <https://github.com/darwin-eu-dev/TestGenerator>
3. OpenAI. Structured outputs [Internet]. OpenAI API Documentation. [cited 2026 Jun 5]. Available from: <https://developers.openai.com/api/docs/guides/structured-outputs>
4. Pantheon HDS. Hecate: AI-driven semantic search for OMOP-CDM [Internet]. [cited 2026 Jun 5]. Available from: <https://hecate.pantheon-hds.com/>

5. Wickham H. testthat: Get started with testing. R J [Internet]. 2011 [cited 2026 Jun 5];3(1):5-10. Available from: https://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf
6. Wickham H, Cheng J, Jacobs A, Aden-Buie G, Schloerke B. ellmer: Chat with large language models [Internet]. R package version 0.4.1. 2026 [cited 2026 Jun 5]. Available from: <https://ellmer.tidyverse.org>