TESTING MANUAL:

The application uses in-memory h2 database. For the testing purposes only, the database data can be accessed on /h2-console using jdbc:h2:mem:testdb as JDBC URL. No authentication is required. All the data (existing users and classes) are being initialized in data.sql script. For easier understanding of application endpoints and models, default Swagger 2 documentation has been implemented and is accessible on /swagger-ui.html . The correctness of implementation can be shown with Postman calls.

Authentication has been implemented using spring boot default basic authentication. So far three users exist (martin, lenart, tina). Password is "test" for all of them. User and password can be changed in the script.

List all available classes can be done using GET request on /classes while the basic search of classes was implemented using query parameter: /classes?name="searchvalue"

Student can enrol, cancel enrollment or get all his enrolled classes using POST, DELETE, GET call on /user/enrollment. It can be done for multiple classes simultaneously. For enrolling or cancelling enrollment from classes a JSON representation of the list of course objects should be provided in the request body. Then API only validates ids of provided courses (class name is being ignored). When enrolling to class/classes if one of the provided ids is not found, CourseNotFoundResponse is thrown. On the other hand, a student enrolling to a class that he has already enrolled in or cancelling enrollment from a class that he is not enrolled to does not throw an exception. The argumentation was that the request should only guarantee that the student is enrolled/not enrolled in the class. Successful POST/DELETE request returns a list of all classes that the student is currently being enrolled to.