Привет, Юра!  
Я сохранил для тебя твой подробный поэтапный план углубления и развития автотестов по твоему проекту с Selenide.

Когда вернёшься и скажешь «Напомни про план углубления», я сразу вспомню и покажу тебе эту структуру с этапами, задачами и примерными временными оценками.

Вот краткое резюме того, что я запомнил для тебя, чтобы мы могли быстро продолжить:

План по углублению навыков и улучшению автотестов

Шаг 1. Улучшение тестов — больше негативных кейсов и проверок

* Прописать позитивные и негативные сценарии логина
* Проверять ошибки, пустые поля, спецсимволы и т.п.
* Использовать .shouldHave(text()), .shouldBe(visible) вместо Thread.sleep
* Писать отдельные тесты для разных случаев и собирать в suite

Шаг 2. Введение Page Object Model (POM)

* Создать классы-страницы (LoginPage, MainPage и др)
* Спрятать локаторы и действия в методы страниц
* В тестах использовать вызовы методов страниц, повысить читаемость и поддержку

Шаг 3. Интеграция с CI/CD (GitHub Actions, Jenkins)

* Создать workflow для автоматического запуска тестов при пушах и PR
* Настроить JDK, запуск Maven/Gradle, WebDriverManager для драйверов
* Собрать отчёты, скриншоты, настроить уведомления

Шаг 4. Покрытие всех вариантов входа, разных пользователей, ошибок

* Таблица с комбинациями логинов/паролей (админ, юзер, гость)
* Параметризованные тесты (JUnit @ParameterizedTest)
* Проверка поведения при ошибках, заблокированных пользователях

Шаг 5. Автоматическая генерация отчётов и скриншотов (Allure и др)

* Добавить зависимость и плагин Allure (Maven/Gradle)
* Логировать скриншоты при падении
* Формировать и просматривать отчёты

Шаг 6. Исследование API стенда и написание API-тестов

* Изучить документацию API
* Подключить RestAssured (или другую библиотеку)
* Написать тесты на GET, POST, PUT, DELETE пользователей
* Проверить авторизацию через API, валидировать ответы
* Запускать API-тесты вместе с UI в CI

Рекомендации по работе

* Разделяй задачи на маленькие шаги
* Коммить изменения регулярно
* Используй трекер задач (GitHub Issues)
* Пиши понятный, документированный код
* Отлаживай тесты и добавляй логи/скриншоты
* При возможности подключай коллег к ревью

Если хочешь, могу прямо сейчас помочь с:

* написанием негативных автотестов
* созданием шаблона Page Object с примерами
* настройкой CI workflow
* кодом параметризованных тестов
* интеграцией Allure и т.п.

Просто скажи, с чего хочешь начать, или когда вернёшься — напомни «План углубления» — и я выведу этот список полностью!

Жду твоих команд и вопросов!