

# План автоматизации тестирования для сайта American Eagle

(<https://www.ae.com/us/en/>)

---

## 1. Цели автоматизации

Проблемы, которые решаются:

- Ускорение проверки критически важных пользовательских сценариев
- Повышение стабильности основного функционала (навигация, поиск, корзина)
- Своевременное обнаружение регрессионных багов после изменений в коде
- Минимизация человеческого фактора и ошибок при ручном тестировании

Преимущества:

- Существенное сокращение времени на проверку повторяющихся сценариев
  - Возможность регулярного запуска тестов по расписанию (например, smoke-набор при CI)
  - Повышение уверенности в стабильности основных пользовательских флоу
- 

## 2. Области автоматизированного тестирования

Crucial — Ключевой функционал (гость):

UI, extended tests:

- Добавление товара в корзину
- Изменение количества товаров в корзине
- Удаление товара из корзины
- Проверка соответствия цены товара в каталоге и корзине
- Проверка получения бесплатной доставки при достижении минимальной суммы

UI, smoke tests:

- Доступность ключевых страниц (главная, каталог, Men → View All)
- Доступность кнопки корзины на всех страницах

API, extended tests:

- Добавление, редактирование, удаление товара в корзине
- Сравнение цен: каталог vs. корзина
- Проверка условия бесплатной доставки

API, smoke tests:

- Добавление товара в корзину

- Получение содержимого корзины
  - Получение информации о товаре
  - Редактирование и удаление товара
- 

**Important — Ключевой функционал (зарегистрированный пользователь):**

**UI, extended tests:**

- Регистрация нового пользователя и вход
- Добавление, редактирование, удаление товара в корзине
- Сравнение цен: каталог vs. корзина
- Проверка условия бесплатной доставки

**UI, smoke tests:**

- Доступность кнопки "Вход"
- Доступность кнопки "Регистрация"

**API, extended tests:**

- То же, что и у гостя, но для авторизованного пользователя

**API, smoke tests:**

- Аналогично, для зарегистрированного пользователя
- 

**Nice-To-Have — Дополнительные проверки UI:**

**UI, smoke tests:**

- Проверка доступности кнопки поиска
  - Проверка кнопки "Избранное"
  - Отображение количества товаров на иконке корзины
- 

**3. Тестировщик: Илья Сидорычев.**

---

**4. Технологический стек**

- **Язык:** Java 17
- **Тестовый фреймворк:** JUnit 5
- **Сборка:** Gradle
- **UI тесты:** Selenium WebDriver

- **API тесты:** REST Assured
- **Отчеты:** Allure Report
- **Owner** (properties)
- **Модели данных:** Lombok
- **CI/CD:** GitHub Actions

---

## 5. Подходы к автоматизации

- Приоритизация: сначала smoke, затем extended
- Разделение по ролям: гость / зарегистрированный пользователь
- Разделение по уровням: UI / API
- CI/CD интеграция:
  - Запуск smoke-набора на каждый pull request
  - Полный регресс — отдельный джоб в pipeline

---

## 6. Ограничения и риски

- Отсутствие доступа к базе данных
- Тестирование ограничено до оформления заказа (без покупки)
- Ограниченные возможности тестирования некоторых критических сценариев из-за ограничений, накладываемых системой защиты от ботов веб-сайта (Akamai), которая может блокировать или препятствовать выполнению автоматизированных тестов
- Возможен дефицит времени/ресурсов на полное покрытие тестами
- Узкая команда: выход тестировщика из строя ставит под угрозу весь процесс

---

## 7. Сроки выполнения

Дата	Этап
<b>23.06 – 27.06.25</b>	Составление и утверждение плана тестирования
<b>30.06 – 06.07.25</b>	Разработка и запуск smoke и extended API тестов
<b>07.07 – 16.07.25</b>	Разработка и запуск smoke и extended UI тестов
<b>17.07 – 25.07.25</b>	Интеграция в CI/CD, оптимизация тестов, устранение багов и финальный запуск