**Дневник Практики в <epam>**

**Неделя 1**

День 1. Знакомство с рабочим местом, создание учетной записи, ознакомление с документацией компании. Установка всех необходимых для работы программных средств. Прохождение задания на Adaptation Portal компании. Получение материалов необходимых, для выполненый практического задания, а так же получение практического задания.  
  
День 2. Прохождение задания на Adaptation Portal компании. Ознакомление с материалами, выданными руководителем, содействующими выполнению практического задания (Основы Java, Обьекты Java, ООП, Java Collection Framework).

День 3. Окончательное прохождение заданий на Adaptation Portal компании. Ознакомление с материалами, выданными руководителем, содействующими выполнению практического задания (Enum, Exceptions, I/O streams ).

День 4. Ознакомление с материалами, выданными руководителем, содействующими выполнению практического задания (JDBC, XML, JSON, Selenium WebDriver, WebDriver API, XPath).

**Неделя 2**

День 5. Ознакомление с материалами, выданными руководителем, содействующими выполнению практического задания (WebDriver Синхронизация и ожидание, Page Object pattern, Page Factory pattern, Actions, JavaScript Executor, Remote WebDriver, Test Automation Challenge, Framework).

День 6. Ознакомление с материалами, выданными руководителем, содействующими выполнению практического задания (Patterns: Singleton, Static Factory Method, Factory Method, Strategy, Decorator, Composite; BDD (Behavior-driven development)).  
  
День 7. Создание проекта. Добавление библиотек, и добавление Factory pattern для возможности тестировать на различных браузерах. Установка Веб-драйвера для Chrome и Firefox. Создание тестов (#1 Успешная регистрация) и проверка работоспособности на обоих браузерах.

День 8. Добавил confige-файл содержаций данные и позволяющий изменять браузер. Добавил класс, позволяющий считывать данные из config-файла. Создание тестов (#2 Проверка названия Страницы, #3 Проверка количества вкладок и их названий ) и проверка работоспособности на обоих браузерах. Создал файл gitignore. Перезаписал проект в новый репозиторий с gitignore и исправил неполадку с FirefoxDriver.

День 9. Создание тестов (#4 Проверка наличия ссылки в вкладках меню, #5 Проверка перехода по ссылке и существования имени пользователя на странице) и проверка работоспособности на обоих браузерах.

**Неделя 3**

День 10. Создание тестов (#6 Переход на страницу с изображениями, проверка имен изображений) и проверка работоспособности на обоих браузерах.

День 11. Доработка теста (#6). Создание тестов (#7 Нажатие на изображение и на название изображения и проверка что в обоих случаях открылась одна и та же страница) и проверка работоспособности на обоих браузерах. Firefox данный тест не осилил.

День 12. Сменил манитор. Настроил открытие окна браузера в полном размере при запуске теста для обоих браузеров. Создание тестов (#8.1-8.6 Проверка наличия элементов на странице деталей изображения) и проверка работоспособности на обоих браузерах. Понизил версии Firefox и Selenium для работоспособности Теста #6 на Firefox. Разделил тест #3 на два под-теста.

День 13. Создание тестов (#8.7 Проверка наличия элементов на странице деталей изображения, #9 Переход на страницу Автора, #10 Проверка полноты информации Альбомов) и проверка работоспособности на обоих браузерах. Так же доработка теста (#3.2).

День 14. Создание тестов (Собственные #11.1, #11.2, #12.1, #12.2, #13.1, #13.2 Тестирование Галерей, Групп и Поисковой функциональности) и проверка работоспособности на обоих браузерах.

**Неделя 4**

День 15. Доработка тестов (#8.4, #13.2). Прохождение задания на Adaptation Portal компании. Попытка внедрения JavaScript-ожидания, но пока что не совсем удачно.

День 16. Принято решение, о том что написание тестов окончено. Приступление к модификации тестов, что бы их можно было запускать независимо друг от друга. Ознакомление с материалами, содействующими выполнению практического задания (TestNG). Создание файла, для паралельного запуска тестов.

День 17. Создал класс для параллельного запуска тестов, в этом классе каждый тест сделал независимым, реализовал возможность запускать каждый тест поодиночке. В результате при запуске всех независимых тестов в произвольном порядке выскакивали каждый раз новые ошибке, которые я устранял.