



АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ ИГРОВЫХ ПРИЛОЖЕНИЙ

Сергей Парамонов
компания «Тортуга»



Зачем?



Игровые приложения



Есть же люди!



Есть же люди

- 4 проекта в активной разработки
- 9 платформ(социальные сети, AppStore, Android Store)
- 10 языков



Есть же люди

- 117 регрессионных тестов
- Среднее время теста прогона всех тестов 4 часа
- Среднее время таска 3 часа
- Цена исправления



Мечты

- Тестировать на каждый коммит
- Тестировать на всех устройствах
- Тестировать каждую ночь
- Нефункциональное тестирование



Особенности наших приложений

- Клиент-серверная архитектура
- Многопользовательность
- «Толстый» клиент написанный на разных языках



Нельзя просто так взять

**и внедрить автоматизированное
тестирование**



Уровни тестирования

- Модульное
- Компонентное
- Системное
- Приемочное



Уровни тестирования

- Модульное
- Приемочное



Проблемы модульного тестирования

- Писать тесты надо уметь
- Тесты пишут разработчики
- Код должен быть тестируемый



Модульное тестирование

- Inverse of Control
- Создание фейковых служб и сервисов
- Нет тестов нет таска(драконовские методы по внедрению)



Проблемы приемочного тестирования

- Отсутствие инструментов
- Сложность написания и поддержки тестов
- Хрупкость
- Долгое прохождение тестов



Кто эти люди?



Кто эти люди?

- Разработчики
- Тестировщики
- Автоматизаторы
- Директора



Как помочь тестировщикам?



Инструменты

- Appium
- Espresso
- Selenium
- WebDriver
- Homebrew



Конвейер развертывания

- Maven
- Jenkins
- Nexus
- Пакетный менеджер



Тестовое окружение

- Тестовое окружение должно быть максимально похоже на боевое(ОС, База Данных и т.д.)
- Разделение сборки(кода) и конфигурации
- Автоматизированные способы установки (скрипты, пакетные менеджеры)
- Эмуляторы и виртуалки



Фикстуры

- Плейсхолдеры в конфигах
- Не работать с источниками данных напрямую
- Использовать то же api что и разработчики
- Не использовать в качестве фикстуры другой тест



Adept АСТ

- Интеграция с TestNg
- Язык тестов - Java
- Клиент-серверная архитектура
- Легкая кастомизация



Adept АСТ - ВОЗМОЖНОСТИ

- Запуск приложения на разных платформах
- Нахождение видимых элементов по пути / тегу
- Эмулирование клика на объекте
- Эмулирование жестов
- Парсинг логов
- Проверка публичных свойств объектов



Пример кода

```
@Test
@Guice(modules = {TestModule.class, AvatarModule.class, CoreModule.class, HouseModule.class, InventoryModule.class,
LocationModule.class, PlayerModule.class, MainModule.class, FixtureModule.class})
public class AvatarTest extends BaseAvatariaTest {

    private static final String TEST_USER_ID = "1";
    @Inject
    private PlayerFixture playerFixture;
    private AcceptanceTestServer testServer;
    private DeviceDriver serverDriver;
    private DeviceDriver driver;
    private TestUser testUser;

    @Test
    public void testRoom() throws InterruptedException, ExecutionException {
        Assert.assertTrue(true);
        serverDriver.start();
        Fetcher fetcher = driver.connect();
        login(fetcher, testUser);

        Avatar avatar = fetcher.findCustomById(testUser.getAvatarGameObjectId(), Avatar.class);
        Assert.assertNotNull(avatar);
        avatar.select();
        Element actionButton = fetcher.findById("txActionsIcon@ui");
        Assert.assertNotNull(actionButton);
        actionButton.click();

        Element cryButton = fetcher.findById("#avatarActionCry");
        Assert.assertNotNull(cryButton);
        cryButton.click();
        avatar = fetcher.findCustomById(testUser.getAvatarGameObjectId(), Avatar.class);
        Assert.assertEquals(avatar.getCurrentAction(), "AvatarCryAction");
        driver.checkError();
        driver.disconnect();
        serverDriver.checkError();
    }

    @BeforeClass
    public void setUp() {
        testUser = Users.getTestUser(TEST_USER_ID);
        playerFixture.createPlayer(testUser);
        testServer = new AcceptanceTestServer();
        testServer.runServer();
        ProjectOptions serverOptions = new ProjectOptions(AvatariaTestConfig.getInstance().getServerFileName(),
AvatariaTestConfig.getInstance().getServerPath());
        serverDriver = testServer.getDriver(DeviceType.SERVER, true, serverOptions);
        ProjectOptions clientOptions = new ProjectOptions(AvatariaTestConfig.getInstance().getClientFileName(),
AvatariaTestConfig.getInstance().getClientPath());
        driver = testServer.getDriver(DeviceType.UNITY_EDITOR, true, clientOptions);
        testUser = Users.getTestUser(TEST_USER_ID);
    }

    @AfterClass
    public void tearDown() {
        testServer.releaseDriver(serverDriver);
        testServer.releaseDriver(driver);
        testServer.destroy();
    }
}
```

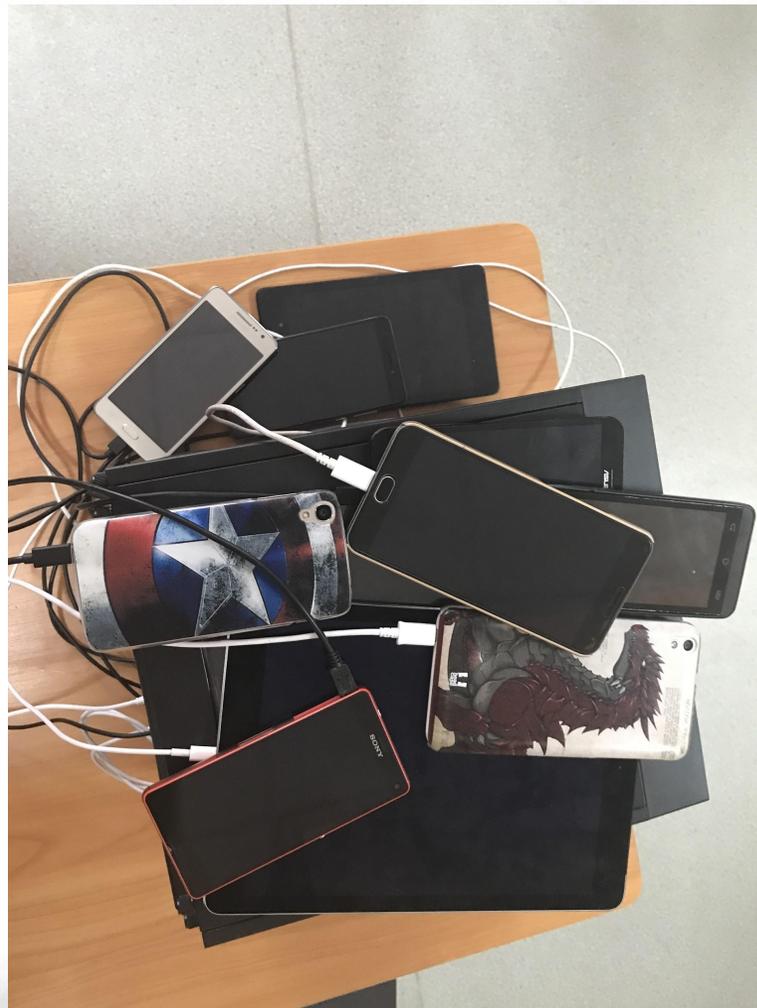


Octopusik

- Загрузка новых сборок
- Запуск приложений
- Управление пулом



Octopusik

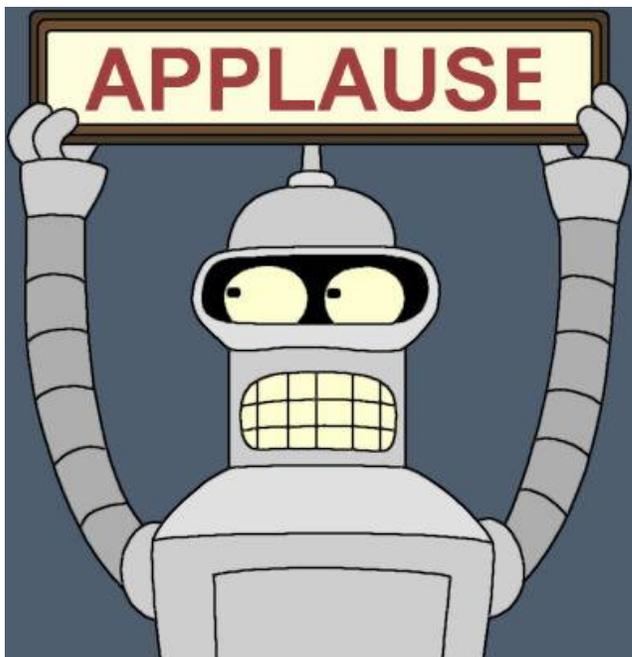


Планы на будущее

- Параллельный запуск тестов
- Клиентская часть для Flash
- Тестирование производительности
- Сравнение изображений



Вопросы?



Сергей Парамонов

Технический директор

ООО «Тортуга»

<http://tortuga.games/>

sergey.paramonov@tortugasocial.com

