Saucelabs:

https://saucelabs.com/sign-up

```
public class SaucelabsConnect implements SauceOnDemandSessionIdProvider,

SauceOnDemandAuthenticationProvider {

          public String username = System.getenv("SAUCE_USERNAME");
          public String accesskey = System.getenv("SAUCE_ACCESS_KEY");
          TestAutomationDriver tad = new TestAutomationDriver();


          public SauceOnDemandAuthentication authentication = new
SauceOnDemandAuthentication(username, accesskey);


   @Rule public TestName name = new TestName() {
      public String getMethodName() {
            return String.format("%s : (%s %s %s)", tad.currentTestcase,  tad.platform,
tad.browser, tad.browserVersion);
      };
};

private WebDriver driver;
private String sessionId;
private String deviceName = null;

public String sauceRemoteURL() {

   String remoteURL = "http://" + authentication.getUsername() + ":" +
authentication.getAccessKey() +   "@ondemand.saucelabs.com:80/wd/hub";

          return remoteURL;
}

public WebDriver setUp(String currentTestCase,Browser browser, String browserVersion,
                        Platform platform, String deviceName, String deviceOrientation) throws
Exception {

   DesiredCapabilities capabilities = new DesiredCapabilities();
```

```

  capabilities.setCapability(CapabilityType.BROWSER_NAME, browser);
  capabilities.setCapability(CapabilityType.VERSION, browserVersion);
  capabilities.setCapability(CapabilityType.PLATFORM, tad.platform);
  capabilities.setCapability("deviceName", deviceName);
  capabilities.setCapability("device-orientation", deviceOrientation);

   driver = new RemoteWebDriver(
         new URL("http://" + authentication.getUsername() + ":" + authentication.getAccessKey()
+      "@ondemand.saucelabs.com:80/wd/hub"),capabilities);
            return driver;

}
```