

Campus Party 2018

Manual do Desenvolvedor



Ford Desafio 2018

South America Mobility Product Solutions



Conteúdo

1. Desafio.....	3
a. Objetivo	3
b. Informações sobre o Desafio.....	3
c. Exemplo de aplicações com AppLink.....	3
2. Plataforma de Desenvolvimento	4
a. Criando um novo Projeto	4
b. Carregando Código Exemplo.....	7
c. Testando o Projeto	9

1. Desafio

a. Objetivo

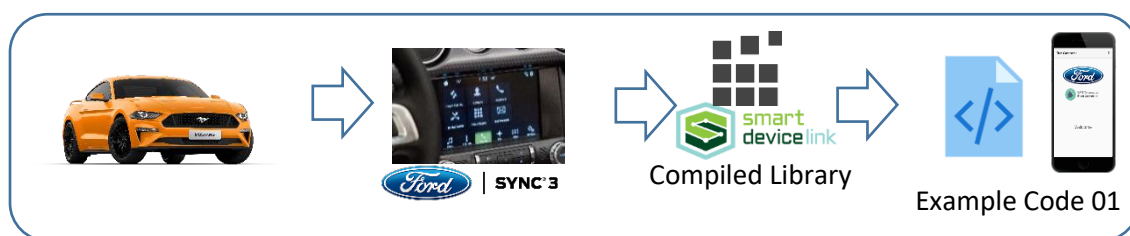
Criar uma aplicação de tema livre com foco em Mobilidade utilizando informações de telemetria de veículos Ford.

Para mais informações, consulte os termos do desafio

b. Informações sobre o Desafio

Neste desafio, disponibilizaremos um código fonte android de exemplo que busca diversas informações do carro em tempo real. Com base neste código, o participante poderá desenvolver diversas aplicações voltadas a mobilidade sem se preocupar com a complexidade do desenvolvimento da comunicação relativa a coleta de informações do veículo.

Abaixo segue uma ilustração simples de como é feita a coleta de dados:



https://www.youtube.com/watch?v=V_i63mDp9A

c. Exemplo de aplicações com AppLink

Abaixo há alguns exemplos de aplicações já existentes para o SYNC.

Glympse	Glympse é a forma mais fácil de compartilhar a sua localização com alguém em tempo real, com segurança. Escolhe quem pode vê-lo e durante quanto tempo.
Filho sem fila	Com este aplicativo, pais de alunos podem, com seus smartphones, avisar à portaria que estão próximos ao Colégio.
Onde Parar	Localizar estacionamentos próximos
Bradesco Exclusive	Localizar agências Bradesco próximos

A lista completa dos apps já disponíveis no SYNC você encontra no site da Ford (<https://www.ford.com.br/servico-ao-cliente/applink/catalogo-de-apps/>)

2. Plataforma de Desenvolvimento

a. Criando um novo Projeto

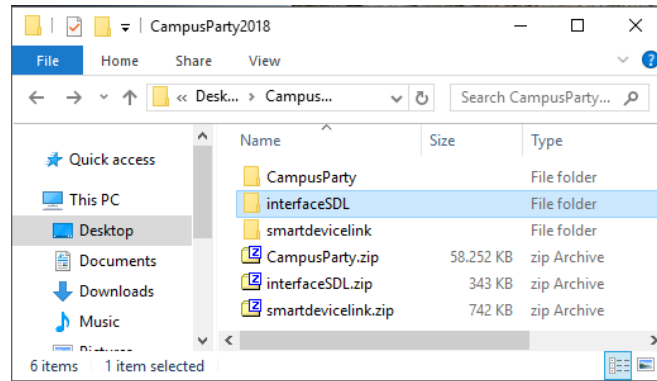
I. Fazer o download dos Arquivos

a. Acessar o GitHub:

<https://github.com/ericbasques/CampusParty2018.git>

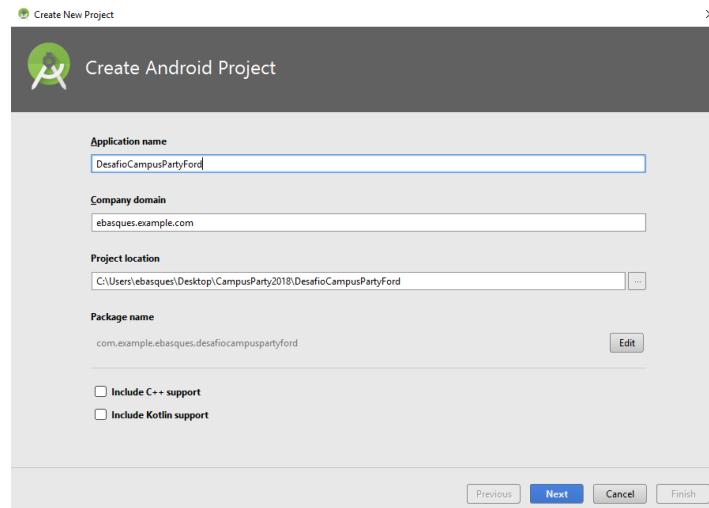
b. Baixar os arquivos “interfaceSDL.zip” e “smartdevicelink.zip” para uma pasta pessoal

c. Descompactar os arquivos



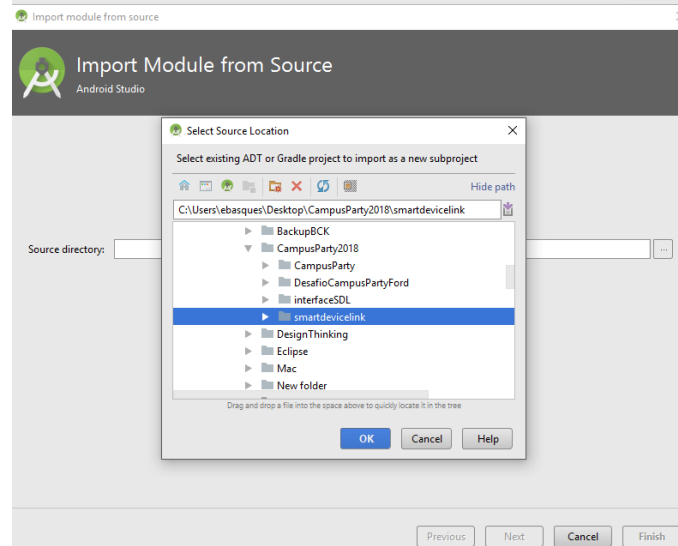
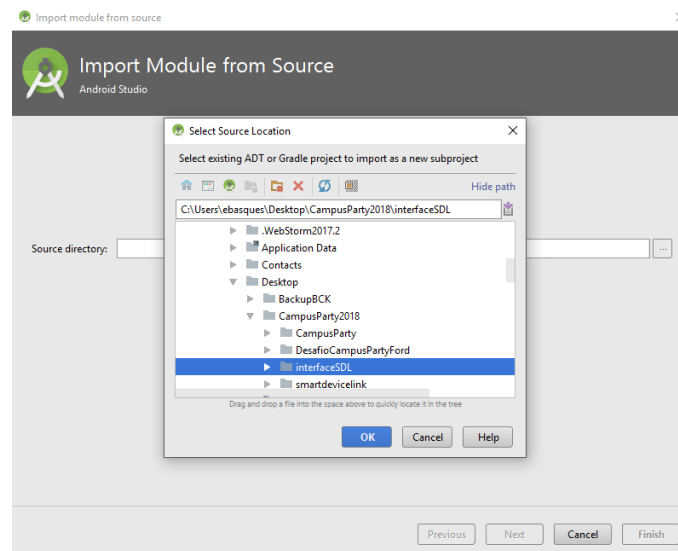
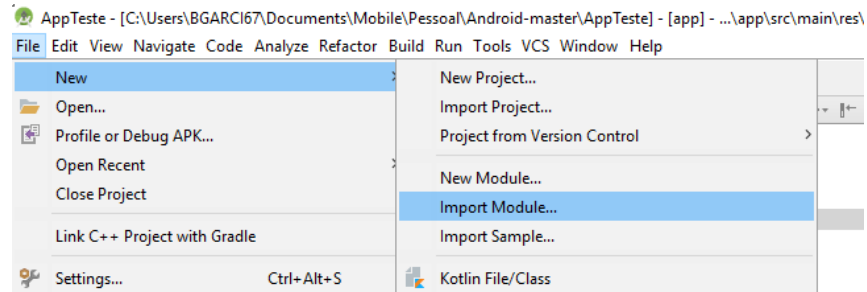
d. Abrir o Android Studio

II. Criar um novo Projeto



III. Importar os módulos

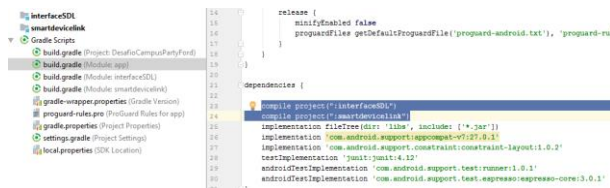
Para importar os módulos (interfaceSDL e smartdevicelink) basta entrar em file>New>Import Module:



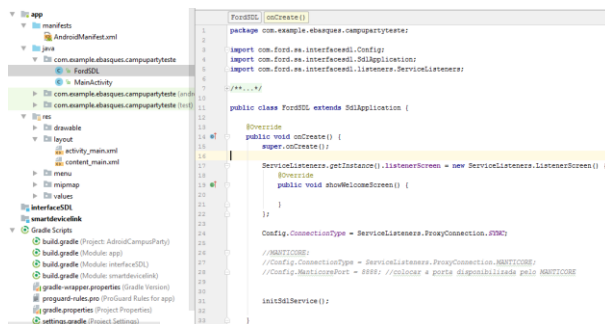
IV. Configurações Iniciais

Para dar continuidade ao processo de criação de um novo projeto, agora é preciso configurar os “imports” no gradle e configurar o “Manifest” Android para adicionar permissões de bluetooth e de classes criadas.

- No build.gradle, incluir as linhas: ‘compile project(“:interfaceSDL”)’ e ‘compile project(“:smartdevicelink”)’, após sincronizar (clicar em “sync”)



- Criar uma classe que herde os atributos da classe SdlApplication (extends com.ford.sa.interfacesdl.SdlApplication). Exemplo:



- No arquivo AndroidManifest.xml :
 - Permissão de Bluetooth:

```
<uses-permission android:name="android.permission.BLUETOOTH" />
```

- Na tag “application” incluir o atributo “android:name” com o nome da sua classe que herda de SdlApplication:

```
android:name=".SdlApplication"
```

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.ebasques.campupartyteste">
    <uses-permission android:name="android.permission.BLUETOOTH" />
    <application
        android:name=".FordSDL"
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="CampuPartyTeste"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="CampuPartyTeste"
            android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

V. Métodos disponíveis:

Confira abaixo uma lista com todos os métodos e listeners disponibilizados pela Ford para a construção do Projeto.

Em caso de dúvidas, recomendamos carregar o código de exemplo que já possui diversas chamadas que podem ser aproveitadas, dando mais agilidade na criação do seu projeto.

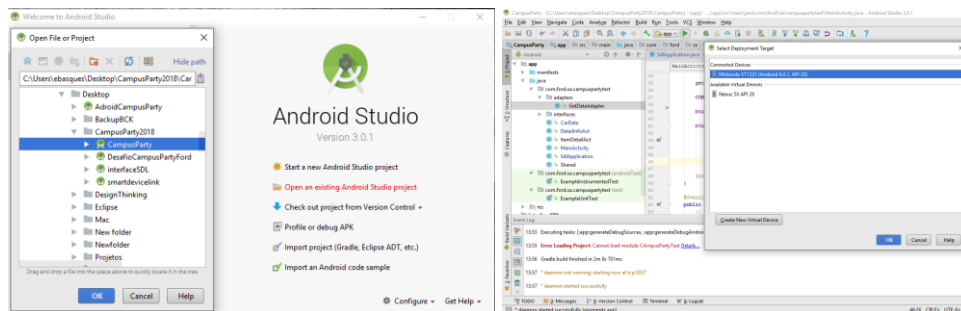
Listner	Método	Descrição
ServiceListeners.getInstance().listenerScreen	showWelcomeScreen()	Método chamado quando a conexão com o SYNC ou Manticore é feita com sucesso.
ServiceListeners.getInstance().listenerHMISatus	onHMIFull(OnHMISatusnotification) onHMIBackground(OnHMISatusnotification) onHMILimited(OnHMISatusnotification) onHMILimited(OnHMISatusnotification)	Métodos chamados quando há alguma alteração de contexto do SYNC
ServiceListeners.getInstance().listenerGetData	onGetVehicleDataResponse(Context ctx, GetVehicleDataResponse response)	Método chamado no retorno das informações do GETDATA
ServiceListeners.getInstance().listenerSubscribeData	onVehicleData(OnVehicleData notification)	Método chamado no retorno das informações do SUBSCRIBE
ServiceListeners.getInstance().listenerOnCommand	onOnButtonPress(OnOnButtonPress notification)	Método chamado quando é efetuado algum click nos botões na tela do SYNC
ServiceListeners.getInstance().listenerLockScreenEvents	onDisposeSyncProxy()	Método chamado quando o SYNC é desconectado
ServiceListeners.getInstance().listenerLockScreenEvents	onLockScreenNotification(OnLockScreenStatus notification)	Método chamado quando há alguma alteração no status da LockScreen
ServiceListeners.getInstance().listenerLockScreenEvents	onCreate(Activity activity, Bundle bundle)	Método chamado na criação da LockScreen (pode ser utilizada para fazer uma LockScreen customizada)

b. Carregando Código Exemplo

Dentro da pasta compartilhada pela Ford, há um projeto android chamado “Campus Party”. Nele poderemos encontrar exemplos mais detalhados de como capturar as informações dos veículos.

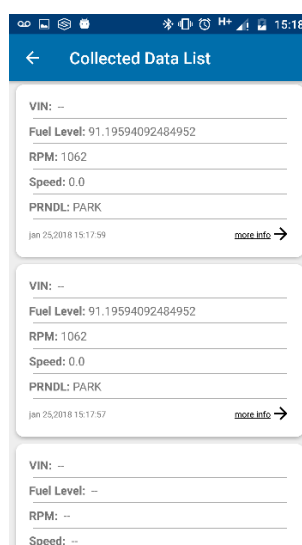
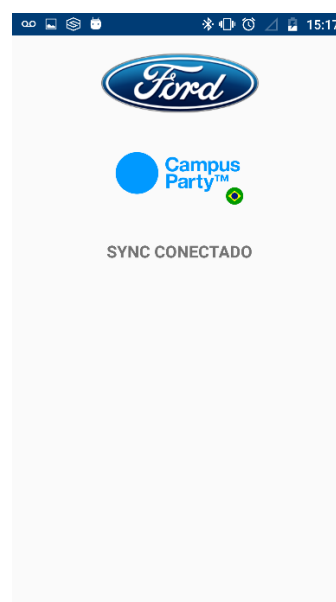
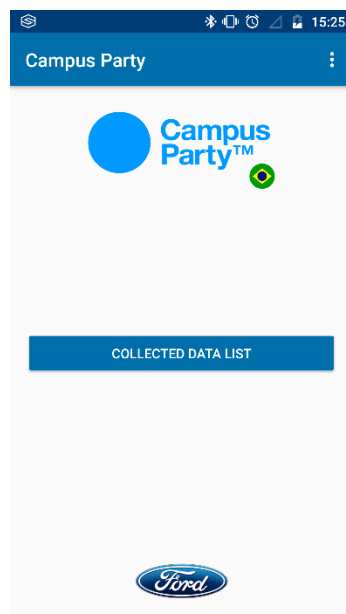
Neste projeto também temos exemplos de como manipular a tela do SYNC. Criar botões, submenus e inserir textos em tempo real na tela.

Para carregar o projeto basta abrir o Android Studio e carregar o Projeto. Caso ocorra algum erro, certifique-se que você possui a versão de SDK correta para o projeto.



Atenção ! Antes de executar o projeto, vá em “Build>Clean Project” e depois vá em “Build>Rebuild Project” para gerar um novo .APK.

Após executar o aplicativo, o .APK instalado será este abaixo:



c. Testando o Projeto

No processo de teste do aplicativo temos 3 frentes diferentes para teste (Bancada, veículo desbloqueado - Mustang - e máquina virtual do SYNC - Manticore)

i. Bancada/Carro

Para testarmos pela Bancada, basta parear o smartphone com o SYNC e procurar pelo aplicativo em “Aplicativos”>”Conectar Aplicativos”.



Caso não apareça nada, faça uma atualização da lista de aplicativos: (Configurações>Aplicativos>Atualizar Aplicativos)



Se ainda assim não funcionar, tente executar um “master reset” no SYNC e refaça os procedimentos acima novamente.

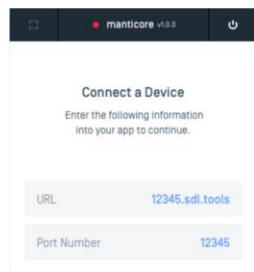
ii. Manticore

O Manticore nada mais é do que uma máquina virtual do SYNC. Quando você se cadastra no site da SDL, automaticamente você tem acesso a esta ferramenta online.

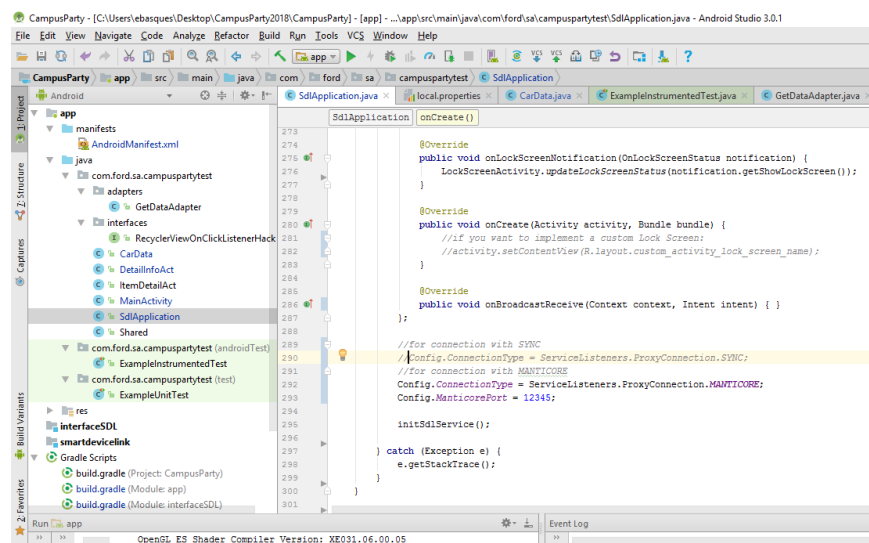
Com o Manticore, além de simular o SYNC, também é possível simular os dados que são imputados no SYNC como velocidade, posição GPS e etc..

Para executar testes através do Manticore, primeiramente você precisa se cadastrar no site da SDL através do link: <https://smartdevicelink.com/sign-in/?next=/resources/manticore/>

Em seguida, você poderá solicitar uma sessão do manticore, na qual irá receber alguns parâmetros para estabelecer conexão.



Tendo o “Port Number” em mãos, basta inserir o número da porta no “Config.ManticorePort”:



Observação 01: É necessário ter conexão à internet para estabelecer a comunicação com o Manticore.

Observação 02: Quando utilizar a conexão pelo Manticore, não esqueça de comentar a conexão tradicional. Faça o mesmo quando deixar de testar pelo Manticore e for usar a bancada/carro.

Observação 03: A execução do Manticore pode apresentar um pouco de lentidão devido ao processo de virtualização.

Mais Detalhes sobre o código para conexão e seu funcionamento estão disponíveis no link: <https://smartdevicelink.com/resources/manticore/>