**演習 1: Bot Builder SDK for Node.js による初めての "おうむ返し" ボットの作成**

**概要**

この演習では、Bot Builder SDK for Node.js を使用してボットを構築し、それを Bot Framework Emulator でテストする方法を示します。

[このフォルダー](https://github.com/GeekTrainer/help-desk-bot-lab/blob/develop/Node/exercise1-EchoBot)の中には、ソリューションと、この演習のステップで作成するコードが入っています。このソリューションは、演習を進めるにあたってさらにヒントが必要な場合に、ガイダンスとして使用できます。ソリューションを使用する前に必ず、npm install を実行してください。

**前提条件**

この演習を完了するには、以下のソフトウェアが必要です。

* [最新の Node.js と NPM](https://nodejs.org/en/download)
* [Visual Studio Code](https://code.visualstudio.com/download) (推奨) や Visual Studio 2017 Community 以上などのコード エディター
* [Bot Framework Emulator](https://emulator.botframework.com/) (ボットのテストに使用するクライアント)

**タスク 1: アプリを初期化し Bot Builder SDK をインストールする**

Bot Builder SDK for Node.js は、Node.js 開発者にとってなじみのある方法でボットを記述する手段を提供する、強力で使いやすいフレームワークです。Express や Restify のようなフレームワークを利用して、JavaScript 開発者にとってなじみのある方法でボットを記述する手段を提供します。

1. Bot Builder SDK とその依存関係をインストールするには、まず、ボット用のフォルダーを作成し、コンソール ウィンドウを開いてそこに移動し、以下の npm コマンドを実行します。app.js をエントリ ポイントとして使用し、残りはそのままにします。
2. npm init
3. 次に、以下の npm コマンドを実行して、[Bot Builder SDK](https://dev.botframework.com/)、[Restify](http://restify.com/)、および [Dotenv](https://github.com/motdotla/dotenv) モジュールをインストールします。
4. npm install --save botbuilder restify dotenv

Bot Builder は Bot Framework の一部で、ボットの作成に使用しますが、Restify は、ボットをホストする Web アプリケーションへのサービス提供に使用されます。Bot Builder SDK は、ご使用の Web フレームワークからは独立していることに注意してください。このハンズオン ラボでは Restify を使用しますが、Express や Koa など別のものも使用できます。Dotenv は、独立したファイルですべての構成設定を簡単に維持するために使用されます。

1. [Nodemon](https://nodemon.io/) の dev 依存関係をインストールします。これは、アプリケーションをホストし、JavaScript に変更が加えられるたびに更新します。
2. npm install --save-dev nodemon

**タスク 2: ボットを作成する**

1. プロジェクトのルート ディレクトリに、以下の内容を含む .env という名前のファイルを作成します。このファイルを使用して、ボットを構成します。
2. PORT=3978
3. MICROSOFT\_APP\_ID=
4. MICROSOFT\_APP\_PASSWORD=
5. ルート ディレクトリに app.js という名前のファイルを作成します。ルート ディレクトリはアプリケーションおよびボットのルートとなります。ボットは Restify フレームワークを使用して、既定でポート 3978 をリッスンします。Restify フレームワークは、ボットを開発する際の標準となっています。

以下のコードには主なセクションが 3 つあります。

* + ChatConnector クラスを使用してチャット コネクターを作成する。
  + Restify ルートでこのコネクターを使用して、メッセージをリッスンする。
  + UniversalBot クラスを使用してコードを追加して、ユーザーに返信する。

Bot Builder SDK for Node.js は、Bot Framework Connector を使用してボットでのメッセージの送受信を構成するための UniversalBot クラスおよび ChatConnector クラスを提供します。UniversalBot クラスはボットの頭脳を形成し、ボットとユーザーのすべての会話の管理を担います。ChatConnector はボットを Bot Framework Connector サービスに接続します。Bot Framework Connector はさらに、ボットがチャネルに送信するメッセージを正規化し、プラットフォームを指定せずにボットを開発できるようにします。これにより開発者は、ユーザーが使用するかもしれない最終的なチャネルよりも、ビジネス ロジックに注意を集中できます。

以下のコードを app.js に追加します。

require('dotenv').config();

const restify = require('restify');

const builder = require('botbuilder');

// Setup Restify Server

var server = restify.createServer();

server.listen(process.env.port || process.env.PORT || 3978, () => {

console.log('%s listening to %s', server.name, server.url);

});

// Create chat connector for communicating with the Bot Framework Service

var connector = new builder.ChatConnector({

appId: process.env.MICROSOFT\_APP\_ID,

appPassword: process.env.MICROSOFT\_APP\_PASSWORD

});

// Listen for messages from users

server.post('/api/messages', connector.listen());

// Receive messages from the user and respond by echoing each message back (prefixed with 'You said:')

var bot = new builder.UniversalBot(connector, [

(session, args, next) => {

session.send('You said: ' + session.message.text + ' which was ' + session.message.text.length + ' characters');

}

]);

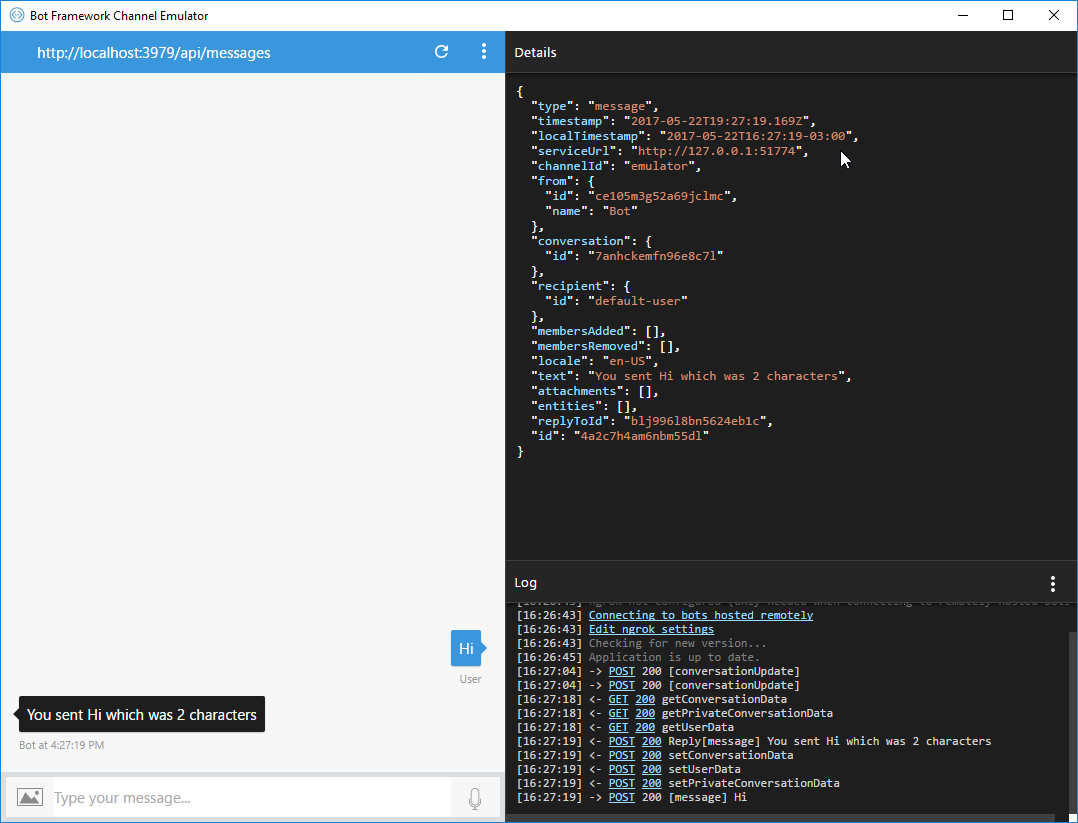
**タスク 3: ボットをテストする**

次に、Bot Framework Emulator を使用してボットをテストし、動作の様子を見てみましょう。このエミュレーターは、localhost 上のボット、またはトンネルを通じてリモートで実行しているボットをテストおよびデバッグできる、デスクトップ アプリケーションです。エミュレーターは、Web チャットの UI に表示されるとおりにメッセージを表示し、JSON 要求をログに記録し、ユーザーがボットとメッセージをやり取りするとおりに応答します。

1. 以下のコマンドを使用して、コンソール ウィンドウでボットを起動します。この時点で、ボットはローカルに実行されています。
2. nodemon app.js

**注:** Windows ファイアウォールの警告が表示される場合は、[アクセスを許可]をクリックします。また、EADDRINUSE エラーが発生する場合は、既定のポートを 3979 または同様のポートに変更します。

1. 次に、Bot Framework Emulator を起動し、ボットに接続します。アドレス バーに http://localhost:3978/api/messages と入力します。これは、ボットがローカルにホストされたときにリッスンする既定のエンドポイントです。
2. [ロケール]を en-US に設定し、[接続]をクリックします。ボットをローカルに実行しているので、[Microsoft App ID]と [Microsoft App Password]を指定する必要はありません。これらのフィールドは、今のところ空白のままにしてかまいません。この情報は、演習 5 で Bot Framework Portal にボットを登録する際に取得します。
3. 送信した各メッセージに対して、メッセージの先頭に "You said"、末尾に "which was ## characters" (## はユーザーのメッセージの文字数) のテキストを付けて、おうむ返しにボットが応答するのを確認します。

[](https://github.com/GeekTrainer/help-desk-bot-lab/blob/develop/Node/images/exercise1-echo-bot.png)