

スマートフォンを活用した英語教育システムの開発

中嶋 正夫

Abstract

This study reports on the development of a system for teaching English using Smart Phones. The system is based on listening and dictation, with students inputting what they hear and then receiving feedback on their answers. Each student's input is recorded in a database. This is not only a record of which answers are correct or wrong, but a learner corpus and parallel target language corpus which can be analyzed in order to identify those phrases and words which are troublesome for students.

Since the system is designed for mobile phone use, students can practice at any time and at any place.

1. はじめに

筆者は数年にわたって、パーソナルコンピュータによる教育支援システムの開発を行ってきた。コンピュータを教育支援ツールとして活用するメリットとして、一般に以下の点が挙げられる。

- (1) 教材をデジタルデータにすることで、配布や更新などが容易になる。
- (2) 音声、画像、映像などの各種メディアとの親和性が高い。
- (3) 学習者の学習履歴データや、誤り傾向を解析するためのデータを蓄積しやすい。
- (4) 学習指導について、知的なルールを構築することで、高度な学習支援が期待できる。

筆者は、過去に Microsoft Excel をベースとした、マルチメディア英語独習システム¹⁾や、Apple Macintosh の HyperCard によるアルゴリズム教育のためのビジュアルプログラミングシステムの開発²⁾を行った。特に、前者のシステムは、実際に大学の英語授業に活用された。

しかし、独習システムというものは、本来パーソナルユースが基本となるべきである。開発当時は、まだパーソナルコンピュータが本来の意味のパーソナルなものでなく、学習者はいつでもどこでもすぐに学習できる状況に恵まれていなかった。

近年はポータブル型ゲーム機をプラットフォームにした学習ツールが多く開発され市販されている。また、パーソナルコンピュータはもちろんのこと、携帯電話をはじめとした小型情報端末が数多く市場に出回っている。特に携帯電話はデータ通信機能を標準で搭載しているために、教材データを配布したり学習データを収集したりすることが容易となった。

現在、携帯電話や PHS の中には、PDA(Personal Digital Assistant) 機能を内蔵した、いわゆるスマー

トフォンと呼ばれる機種がある。欧米では1997年頃から登場し、日本においても昨年までにほとんどの通信業者から発売されている。このスマートフォンには数社からOSが供給されているが、いずれもアプリケーションソフトを開発するキットが用意されており、誰でもソフトウェアの開発を行うことが可能である。

今回の開発では、その中で Microsoft 社の Windows Mobile 5.0 を搭載した PHS を取り上げた (図 1)。



図 1 スマートフォンの例

これは、Windows Mobile が、日本で販売されているスマートフォンの大部分を占めていることや開発のしやすさなどが主な理由である。また、入力インターフェースとして、フルキーボードやタッチスクリーンによるペン入力、音声などさまざまな手法を活用できるのも特徴である (図 2)。



図 2 フルキーボード搭載機

本研究の目的は、このような携帯ツールによる学習システムを構築することによって、いわゆるいつでも、どこでも学習できるユビキタスな独習システムの開発を行うことである。また、学習効果を高めるためのさまざまな学習形態を模索することである。

なお、本研究は平成 18、19 年度科学研究費補助金の対象の研究の一環として行っている。

2. システム概要

2.1. システムの目的

今回のシステムは、英語のディクテーションを学習するソフトウェアである。題材は、大学生のための英語教材からディクテーション問題を抜粋した。学習の流れは、英語の音声を聴いてから、その英文を入力するというスタイルである。

その際、何度か解答する機会を設け、正答でない場合は、各種のヒントを提示することで、正答へと導く。学習者は、スマートフォンを使うことにより、いつでもどこでも学習することができる。そして、Windows Mobile のアプリケーションプログラムの管理の特性により、パーソナルコンピュータと異なり、気軽に学習プログラムを中断したり、再開したりすることが可能である。そして、正答に至るまでの、試行回数や解答内容、誤り内容などの学習履歴を記録することができる。この履歴をネットワーク等で収集することにより、教員は学生の学習の進捗状況や誤りの傾向などを把握することができる、今後の学習指導や教材作成に役立てることができる。

2.2. システム構成

本研究の実行システムとして、Microsoft の Windows Mobile 5.0 を搭載したシャープの W-Zero3 SH007SH を採用した。スマートフォンに採用されている OS には、前述のもの以外に Symbian などもあるが、前述のように日本国内の各通信業者で広く採用されているのは Windows Mobile だからである。もちろん、他社の Windows Mobile 機でも動作する。

開発環境としては、Microsoft の Visual Studio 2005 の Visual Basic 2005 を使用した。これは、基本的には Windows アプリケーション等を開発する環境であるが、Windows Mobile 5.0 SDK for Pocket PC³⁾ や Localized Windows Mobile 5.0 Pocket PC Emulator Images(Japanese)⁴⁾ をインストールすることにより、Windows Mobile 5.0 のアプリケーションを開発することができる。Windows Mobile のエミュレータも包含しているので、デバッグや試験運用はパーソナルコンピュータのみで行える。

また、通常の Windows アプリケーション用のコマンドと下位互換ながら共通な部分が多いので、過去に開発したプログラムからの移行も比較的楽に行える。今回はプログラムと教材データを作成し、インストーラプログラムとして学習者に配布する。学習者はそれを、自分の Windows Mobile 機にインストール作業を行うことで、本学習システムを実行することができる。

2.3. 学習の構成

学習システムは、まずインストールされたスマートフォンに登録されているオーナー情報から、学習者を特定する。そして、過去の学習履歴をもとに教材を実行する。学習の流れを以下に示す。まず、学習者は学習システムのプログラムのアイコンをダブルタップし、起動させる。起動画面では、図 3 のように、スマートフォンに登録されている学生の名前が表示される。

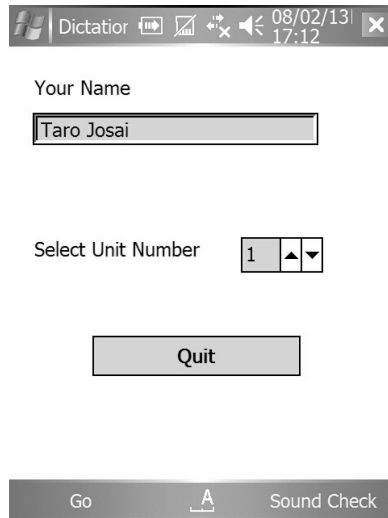


図3 ディクテーションシステムの初期画面

ディクテーションの教材なので、スマートフォンにイヤホンなどを接続し学習を行う。そのために、事前に画面右下のソフトウェアキー [Sound Check] をタップし、サンプルの音声を聴きながら、音量を調節する。多くのスマートフォンは本体に備え付けられた「ソフトキー」にこのソフトウェアキーが割り当てられており、画面をタップしなくても、この操作を行うことができる（図4）。

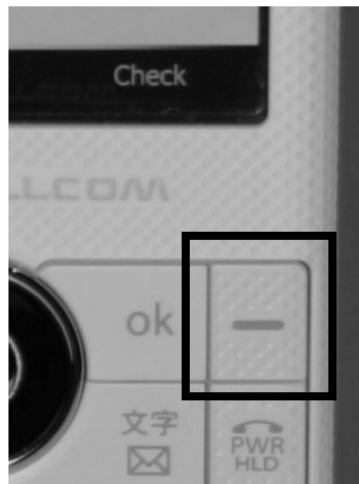


図4 ソフトキー

次に、この初期画面で学習するユニットを選択する。教材は16問で1ユニットが構成されており、自分が勉強したいユニットをタップするか、方向キーなどで選択することができる。ユニットを選択したら、左下のソフトウェアキーの [Go] ボタンをタップするか、ソフトキーを押下する。

画面は、学習ウィンドウに切り替わる。このとき、学習履歴をもとにして、選択したユニットの未

学習の問題から開始される。ユニット自体初めて学習する場合は問題番号1から開始される（図5）。

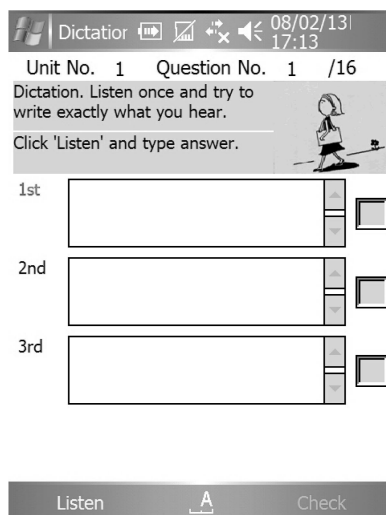


図5 デイクテーション問題画面

デイクテーションでは、1つの問題について、3回解答する機会が与えられる。1回目は、最初 [Listen] ボタンをタップし、英文の音声を聴く。音声が終わると、入力欄「1st」が入力可能となるので、スマートフォンのフルキーボードで入力を行う（図6）。

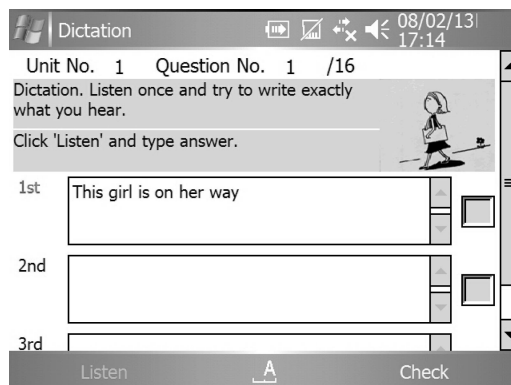


図6 解答の入力例

入力が完了したら、[Check] ボタンをタップし、正誤判定を行う。もちろんこれらのボタンはソフトキーを押下することでも選択できる。正解であれば次の問題に移り、誤っていれば2回目の解答を行う（図7）。

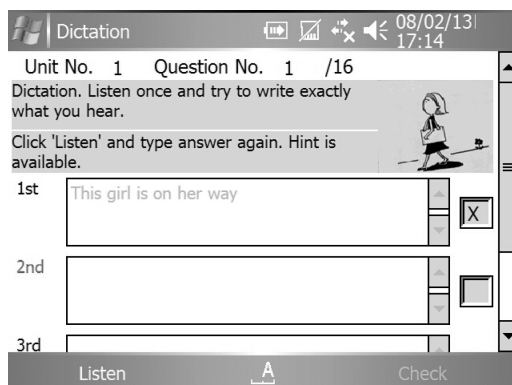


図7 1回目の解答が不正解の場合

2回目の解答では、ウィンドウ上の [Hint] ボタンが利用できるようになる。このボタンをタップすると、ヒント表示欄に出題文の日本語訳が表示される（図8）。解答の入力は1回目と同様で、フルキーボードを使って入力する。

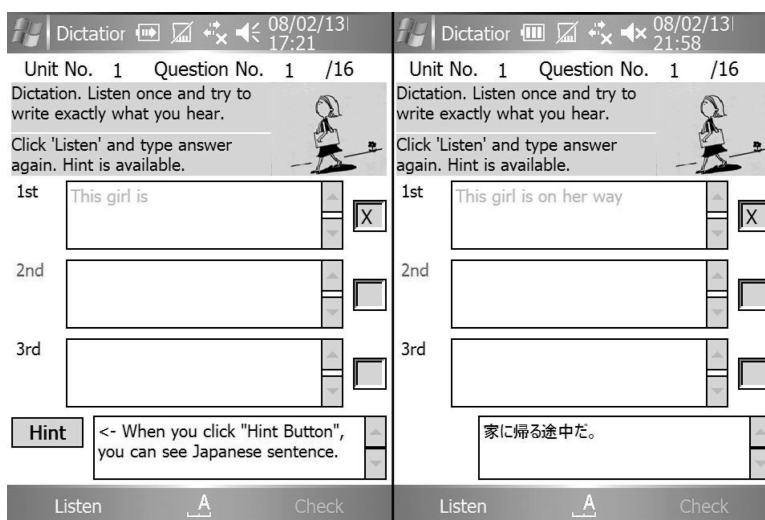


図8 ヒント情報の表示

2回目の解答でも不正解の場合は、3回目の入力に切り替わる。このとき [Hint] ボタンをタップすると、今度はウィンドウ右下に、問題で使われている単語のリストがランダムに、一覧形式で表示される。ここで、英単語を入力するには、上下の方向キーを使って、リストから該当する単語を探し、決定ボタンで入力欄に単語を挿入する（図9）。

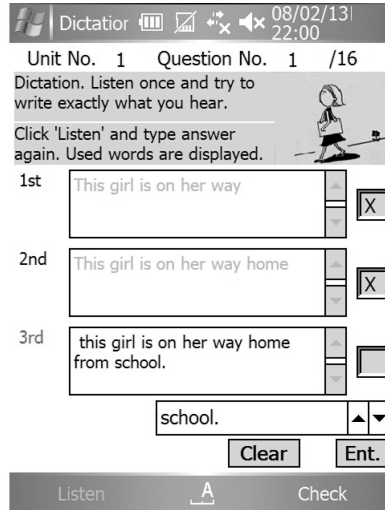


図 9 単語リストからの入力例

基本的には、これを繰り返すことにより、解答の英文を作ることができる。文の途中に単語を挿入したい場合は、入力欄の挿入ポイントにカーソルを表示させ、ボタン [Ent.] をタップする。また、[Clear] ボタンをタップすることで、入力欄に作成した文を全文削除することができる。

この3回目の入力の時には、フルキーボードを使わなくても、解答の英文を完成させることができるのが特徴である。あたかも一般的な携帯電話でメールを作成するようなスタイルで入力できる。

このように、3回の解答まで行った場合と、それ以前の試行で正答が導き出されたときには、正解までにかかった時間や、各試行での学習者の解答データ、正誤の判別などを学習履歴ファイルに保存する。そして、システムはディクテーションの問題を継続するか、終了するかを尋ねる。ここで、継続を選択すれば、次の問題文が表示され、終了を選択すればプログラムは終了する (図 10)。

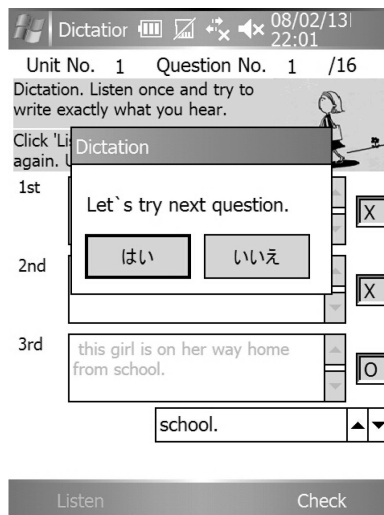


図 10 解答終了時の画面

ただし、Windows Mobile では、基本的にはアプリケーションは完全に終了せず、メモリ上に残っている。再びアプリケーションを起動しようとする時、実際にはプログラムが再開し、以前に中断した問題から学習が再開される。Windows Mobile では、メインメモリに余裕がなくなると、自動的に現在使われていないアプリケーションを完全に終了し、メモリを解放する。この場合、再度アプリケーションを起動すると、初期設定のウィンドウが表示され、ユニットの選択を行うことになるが、これも学習履歴データを読み込むことによって、学習の続きから再開することができる。

以上のような学習の流れを経て、1 ユニット 16 問の解答が終了したら、次のユニットを選択できるようになる。

3. システム開発

3.1. 開発環境

本システムは Microsoft Visual Studio 2005 Professional Edition の Visual Basic 2005 を用いて作成した。これに、Windows Mobile 5.0 SDK for Pocket PC と Localized Windows Mobile 5.0 Pocket PC Emulator Images(Japanese) を別途インストールすることにより、日本語版の Windows Mobile 5.0 のアプリケーションプログラムを作成することができる (図 11)。

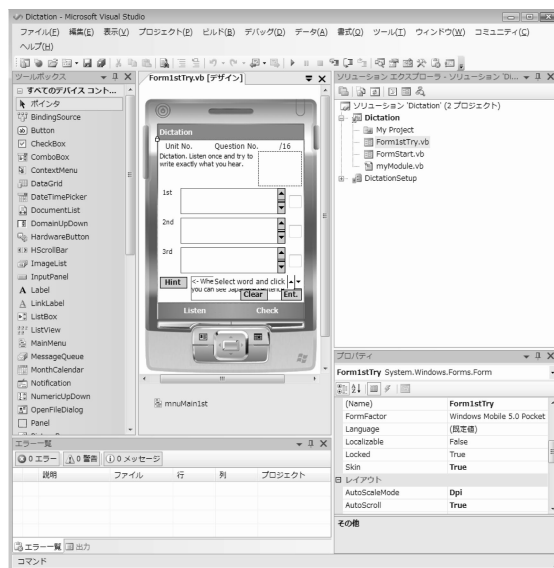


図 11 Microsoft Visual Studio 2005 による開発画面

Windows Mobile 5.0用のアプリケーションプログラムを作成するためには、Visual Basic 2005 のプロジェクト作成ウィンドウで、「スマートデバイス」から「Windows Mobile 5.0 Pocket PC」用のデバイスアプリケーションを選択する（図 12）。

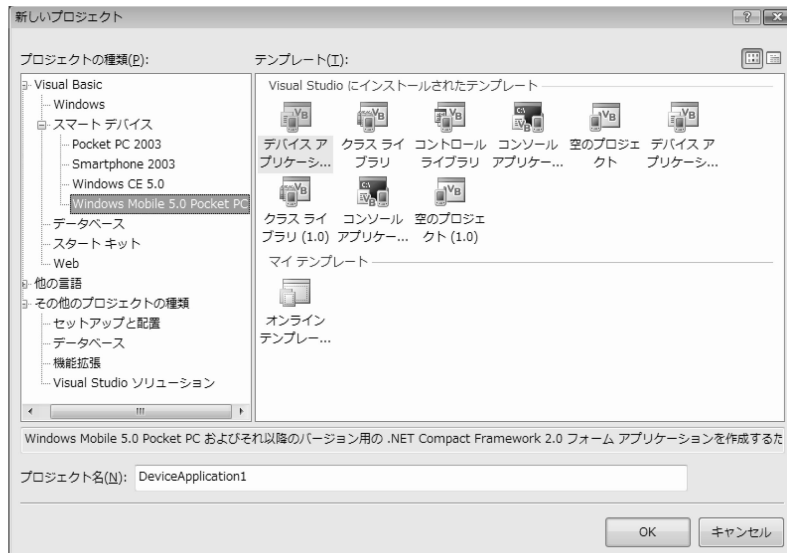


図 12 Windows Mobile 用アプリケーションのプロジェクト

すると、通常のアプリケーションの作成ウィンドウが表示される。このとき、「ツールボックス」には、Windows Mobile 5.0 で利用できるコントロール群（インターフェースなど）が表示される。アプリケーションの作成手順は、ウィンドウズアプリケーションとほとんど変わらない。デバッグの実行は、前述でインストールしたエミュレータか、Visual Studio 2005 をインストールしたパーソナルコンピュータに USB 等で接続した Windows Mobile の実機で行うことができる（図 13）。



図 13 エミュレータ画面 (右)

3.2. アプリケーションソフトの作成

本システムは大きく2つのフォーム（ウィンドウ）から構成されている。1つめのフォームは初期画面であり、学習者の学習履歴データを読み取り、既に学習済みのユニットを認識し、次に学習すべきユニットに学習者を導く。また、問題文を聞き取る前に、適正な音量を調節するための [Sound Check] ボタンを用意した。

アプリケーションソフトの開発で使う各種コントロールは、ウィンドウズアプリケーションに準じているが、多くの場合それに適応できるプロパティやメソッドはウィンドウズ用のもののサブセット版であるため、機能に制限がある。

Windows Mobile 5.0 の開発においては、Windows Mobile 5.0 SDK for Pocket PC をインストールすることで、Microsoft .Net Compact Framework 2.0⁹⁾ によるプログラム開発が容易になるのだが、この中には音声データを簡便に再生させるメソッド（命令文）は用意されていない。

この機能を実現させるためには Win32 API を使って、OS の機能を直接呼び出す必要がある。そのため、図 14 のようなコードを記述する。このように、通常のウィンドウズアプリケーションでは、あらかじめ用意されたメソッドで機能が実現できるものでも、Windows Mobile ではクラスやメソッドそのものが用意されていないことがある。同様の機能を実現するための Win32 API を呼び出すステートメントを記述しなければならないため、開発者には負担がかかることが多い。

```
Public Class FormStart
    <DllImport("CoreDll.DLL")>
    Private Shared Function PlaySound(ByVal szSound As String, ByVal hMod As IntPtr, ByVal flags As Integer) As Integer
    End Function
    Const SND_FILENAME As Integer = &H20000 'サウンドデータをファイル名で指定する
    Private Sub mnuSoundCheck_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuSoundCheck.
        PlaySound(pathName & "dictation#check.wav", IntPtr.Zero, SND_FILENAME)
    End Sub
End Class
```

図 14 Win32 API によるサウンド機能のコーディング

2つめのフォームは、実際に問題を解くためのウィンドウである。「2.3. 学習の構成」のように、1つの問題に対して、3回の解答する機会 (attempt) を学習者に与えている。基本的には、音声を聴き、答えを入力し、正誤判定を行うというフローの繰り返しである。

教材は、大きく3つのデータから構成されている。1つめはテキストデータである。これには、問題文の正解である英文と日本語訳の文から構成されている。2つめは、音声データである。これはディクテーションで聴き取るための英文を読んだ音声である。前述のとおり、サウンドの再生には Win32 API を用いるが、これでサポートされているデータ形式は原則的に wav 形式である。mp3 形式など圧縮形式のデータを扱えれば、教材データのコンパクト化が図られるが、今回は見送った。3つめのデータは画像データである。音声だけでなく、その場面に関連した画像を表示することで、学習者へのヒントになったり、学習にアクセントを加えることで学習意欲を維持したりする効果がある (図 15)。

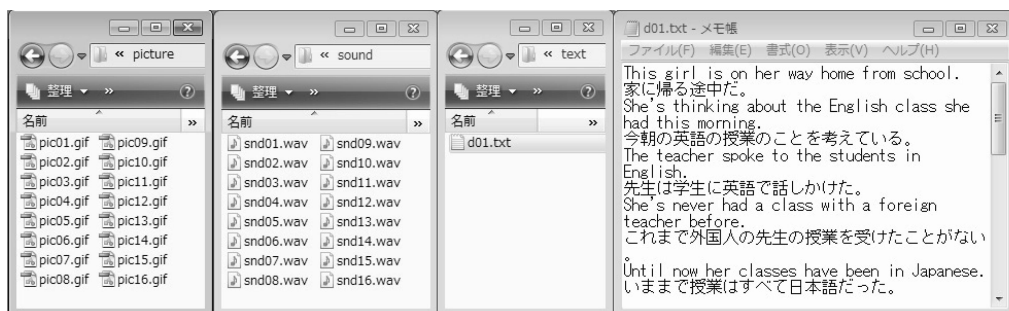


図 15 教材データ (左から、画像、音声、問題文・日本語訳)

正誤判定するために、学習者が解答欄に入力した英文とテキストデータにある英文のデータを比較する。正答と比較するためには、学生の解答データと比較的緩いマッチングを行う。具体的には、単語間の余計なスペースを削除したり、両者の文字をすべて小文字に変えたりしている(図 16)。これは、学習者に身につけさせたいスキルが、あくまでもディクテーション能力だからである。この他の英語のスキルを習得させるには、別の形態の問題を出題させればよい。

```
Public Function deleteSpace(ByVal sentence As String) As String
    REM 入力された文字列の余計なスペースを取り除く
    Dim strLength As Integer
    Dim i As Integer
    Dim prevChar As String
    Dim oneChar As String
    Dim addChar As String
    Dim resultSentence As String

    REM 文字列の前にあるスペースをすべて取り除く
    While Mid(sentence, 1, 1) = " "
        sentence = Mid(sentence, 2)
    End While

    REM 文字列の後ろにあるスペースをすべて取り除く
    strLength = sentence.Length
    While Mid(sentence, strLength, 1) = " "
        sentence = Mid(sentence, 1, strLength - 1)
        strLength = sentence.Length
    End While

    REM 単語間のダブったスペースを取り除く
    strLength = sentence.Length
    prevChar = Mid(sentence, 1, 1)
    resultSentence = prevChar

    For i = 2 To strLength
        oneChar = Mid(sentence, i, 1)
        addChar = oneChar
        If prevChar = " " Then
            Select Case oneChar
                Case " "
                    addChar = ""
                Case "."
                    resultSentence = Mid(resultSentence, 1, resultSentence.Length - 1)
                Case ","
                    resultSentence = Mid(resultSentence, 1, resultSentence.Length - 1)
                Case Else
                    resultSentence = resultSentence & addChar
            End Select
        End If
        resultSentence = resultSentence & addChar
        prevChar = oneChar
    Next
    Return resultSentence.ToLower
End Function
```

図 16 解答文字列の前処理

3.3. インストーラの作成

Windows Visual Studio は、通常のアプリケーションだけでなく、スマートフォンへインストールするためのプログラム（インストーラ）を作成することができる。今回はプログラム本体と、教材データを一括で組み込むインストーラを作成した。プログラムのアップデートや教材の追加については、このインストーラを利用することで、電子メール等で配布し実装することが容易になる（図 17）。

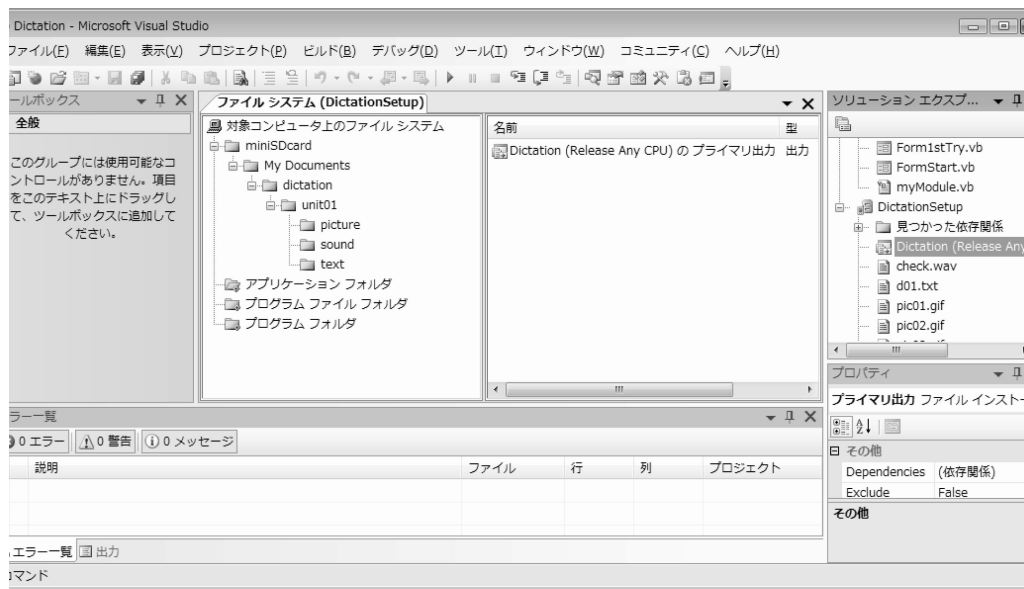


図 17 インストーラの作成画面

3.4. システムの評価

現在、大学生数人によるシステムの評価を始めたところである。今後はシステムを試験運用することで、システムのユーザビリティの検証と改良を行っていく。また、学習履歴を収集、解析することで、本システムの有用性の評価や、今後の教材データの作成に反映させたい。また、今後はディクテーションだけでなく、学生の英語に関する能力を高めるための学習方法の考察や、スマートフォンへの実装を予定している。そして、学生のコーパスを収集するための入力機器としての活用も予定している。

4. おわりに

本研究では、スマートフォンによる英語の学習システムを開発した。これは、モバイル機器による e-Learning の実用性を評価すると同時に、学習者の英語のディクテーションの習熟度を評価するための学習データの収集を目的としている。

システムを構築した結果、スマートフォンのマルチメディアツールとしての可能性を見出した。また、Microsoft Visual Studio という開発環境を活用することによって、ある程度容易にシステムが構築できることが確認できた。

しかし、必ずしも通常の Windows アプリケーションを開発するのと同様に構築できるとは限らな

かった。今回は、Microsoft .Net Compact Framework 2.0 を使用したが、これに含まれるコントロールなどには制限があり、Windows アプリケーションでは、容易に実現できる機能が、Windows Mobile の場合には、Win32 API を直接呼び出すことで、実装する必要があった。

これについては、最近 Microsoft .Net Compact Framework 3.5 が Microsoft で配布が始まり、最新の Microsoft Visual Studio で開発ができるようなので、いくつかの機能については、解決できると思われる。これにより、従来よりも容易に開発可能となることが期待できる。

今後は、よりユーザフレンドリーな学習のインターフェースを構築し、より学習者のモチベーションを高めることを目指す。

たとえば、正誤判定で単純な○×判定だけでなく、内容に応じて獲得ポイントを表示する機能を付け加えることを予定している。

また、スマートフォンの通信機能を活用し、ネットワークを通じてサーバから教材をより簡便に提供する機能や、学習データを収集し教員が容易に分析できるデータベースの構築を目指したい。

【参考文献】

- 1) 中嶋 正夫 他：“英語独習用マルチメディア教材の作成とサポートについて”，私立大学情報教育協会大学教育と情報 93 号，2001.
- 2) 中嶋 正夫 他：“ビジュアルプログラミング環境と問題解決アプローチの定量評価システム”，Navigware フォーラム 1997 年論文集，p.113-121 1997.
- 3) <http://www.microsoft.com/downloads/details.aspx?familyid=83A52AF2-F524-4EC5-9155-717CBE5D25ED&displaylang=en>
- 4) <http://www.microsoft.com/downloads/details.aspx?familyid=eec33ae3-c129-4c25-abaa-18e8e842178f&displaylang=en>
- 5) <http://www.microsoft.com/downloads/details.aspx?FamilyID=aea55f2f-07b5-4a8c-8a44-b4e1b196d5c0&DisplayLang=ja>