

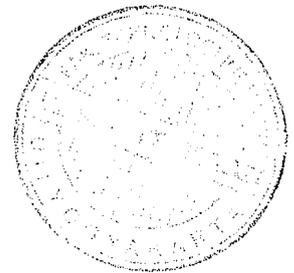
Rancang Bangun Aplikasi TOEFL Untuk Pocket PC Dengan

Teknologi .Net

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana

Teknik Informatika



disusun oleh:

Nama : Siti Nurul Jannah

No. Mahasiswa : 03 523 207

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2007

LEMBAR PENGESAHAN PEMBIMBING

**Rancang Bangun Aplikasi TOEFL Untuk Pocket PC Dengan
Teknologi .Net**

Tugas Akhir

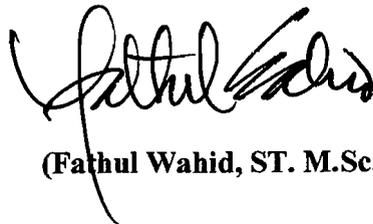
Oleh :

Nama : Siti Nurul Jannah

No. Mahasiswa : 03 523 207

Yogyakarta, 12 April 2007

Pembimbing



(Fathul Wahid, ST. M.Sc.)

LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR

Yang bertanda tangan di bawah ini

Nama : Siti Nurul Jannah

NIM : 03 523 207

Tugas Akhir dengan judul :

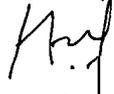
Rancang Bangun Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .Net

Dengan ini saya menyatakan dengan sesungguhnya bahwa dalam Tugas Akhir ini tidak terdapat keseluruhan tulisan atau karya yang saya ambil dengan menyalin, meniru dalam bentuk rangkaian kalimat atau simbol atau algoritma atau program yang menunjukkan gagasan atau pendapat atau pemikiran orang lain, yang saya aku seolah-olah sebagai tulisan atau karya saya sendiri.

Apabila saya melakukan hal tersebut di atas, baik sengaja atau tidak, dengan ini saya menyatakan menarik Tugas Akhir yang saya ajukan sebagai hasil karya saya sendiri ini. Bila di kemudian hari terbukti bahwa saya melakukan tindakan di atas, gelar dan ijazah yang telah diberikan oleh Universitas Islam Indonesia batal saya terima.

Yogyakarta, 30 April 2007

Yang Membuat Pernyataan



Siti Nurul Jannah

Saksi-saksi

Fathul Wahid, ST.,M.Sc.

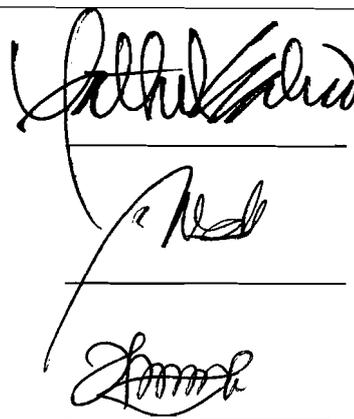
Saksi I

Yudi Prayudi, SSI.,M.Kom

Saksi II

Hendrik, ST.

Saksi III



LEMBAR PENGESAHAN PENGUJI

Rancang Bangun Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .Net

TUGAS AKHIR

Oleh:

Nama : Siti Nurul Jannah
NIM : 03 523 207

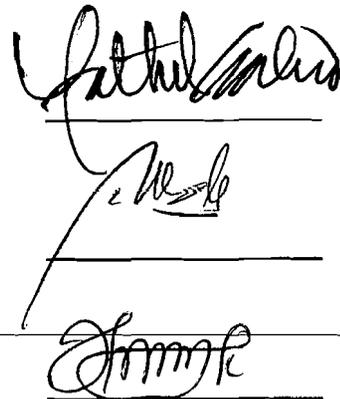
Telah Dipertahankan di Depan Sidang Penguji Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia
Yogyakarta, 30 April 2007

Tim Penguji

Fathul Wahid, ST.,M.Sc.
Ketua

Yudi Prayudi, SSi.,M.Kom
Anggota I

Hendrik, ST.
Anggota II



Mengetahui,

Ketua Jurusan Teknik Informatika
Universitas Islam Indonesia
Yudi Prayudi, SSi.,M.Kom



PERSEMBAHAN

Rasa puji syukur aku panjatkan kepada Allah SWT atas karunia dan rahmat-Nya aku bisa menyelesaikan tugas akhir ini dengan baik dan semoga akan dapat bermanfaat dikemudian hari atau bagi orang lain.

Ucapan terima kasih juga tak lupa aku ucapkan kepada kedua orang tua ku, special to ibuku sebagai sosok wanita yang selalu berkorban untuk kesuksesan putra-putrinya, selalu berjuang, berjuang, dan berjuang tanpa kenal lelah, kakak-kakak ku juga keluargaku yang selalu memberikan semangat.

Buat Ardhan yang selalu menemani disaat senang maupun duka, terima kasih Buat semua dorongan moral dan materil semoga cepat selesai S2nya, tercapai semua harapan, cita-citamu, dan tidak ada hambatan apapun...tetap setia menunggu tidak terbatas ruang dan waktu.

Buat Dika, sahabat di semua suasana, the best partner, our friendship is everafter. Sorry aku diluan...semoga persahabatan kita tetap ada selamanya. Kamu benar: Semangat datang dari diri kita sendiri.. Semoga kamu cepat nemuin seseorang yang bisa merubah hidupmu jadi indah. Sukses, Dik! Ga ada loe ga rame...

Buat Ari, Rofi, Duwi, Egi , cepetan kelar...biar bisa jalan-jalan lagi, trus makan-makan juga. Thank's, girls untuk kebersamaan yang indah dari awal semester sampai sekarang...banyak kenangan indah yang terukir diantara kita. Semoga kita masih punya banyak waktu untuk merajut kembali kebersamaan ini.

Buat Ririx, lama aja seh...buruan dunk. Nggak usah ngeceng torus di kampus. Selesin tu skripsinya. Makasih ya supportnya.

MOTTO

“ Life Is A Highway ”

“ masa lalu adalah sebuah pengalaman, tapi di dalam masa lalu tidak pernah ada masa depan, jangan pernah menoleh ke masa lalu untuk mengingat kenangan suram, lihatlah pengalamannya ”

“Sesungguhnya Allah S.W.T akan membantu orang-orang yang berusaha, sekalipun ia tidak memiliki kekuatan dan kemampuan, melainkan kemauan yang kuat serta niat yang tulus dan ikhlas ”

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamu'alaikum Wr. Wb

Alhamdulillah, segala puji syukur hanyalah kepada Allah SWT dan semoga sholawat serta salam dilimpahkan kepada junjungan kita Nabi Muhammad SAW, keluarga, sahabat, dan pengikut-pengikut beliau (Amin). Sehingga penulisan laporan tugas akhir yang berjudul: **Rancang Bangun Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .Net** dapat penulis selesaikan dengan baik.

Laporan tugas akhir ini disusun untuk melengkapi salah satu syarat guna memperoleh gelar Sarjana Teknik Informatika pada Universitas Islam Indonesia dan atas apa yang telah diajarkan selama perkuliahan baik teori maupun praktek, disamping laporan itu sendiri yang merupakan rangkaian kegiatan yang harus dilakukan setelah tugas akhir ini selesai.

Untuk itu saya menyampaikan ucapan terimakasih dan penghargaan yang setinggi-tingginya kepada:

1. Ayah, Ibu dan keluarga tercinta, atas kasih sayang, do'a dan dukungannya.
2. Bapak Fathul Wahid, ST. M.Sc. selaku dosen pembimbing dan Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.

3. Bapak Yudi Prayudi, S.Si.,M.kom., selaku Ketua Jurusan Teknik Informatika.
Terimakasih atas segala bantuan, dukungan, dan pengetahuannya, serta kemudahan yang telah diberikan.
4. Teman-teman yang telah membantu memberikan ide, kritik dan masukan dalam pembuatan tugas akhir ini.
5. Teman-teman Informatika 2003 dan seluruh pihak yang tidak dapat saya sebutkan satu persatu.

Semoga apa yang telah mereka berikan dengan keikhlasan, mendapat pahala yang setimpal dari Allah SWT. Penulis menyadari dalam penulisan laporan tugas akhir ini masih jauh dari sempurna, karena keterbatasan kemampuan dan pengalaman. Penulis mengharapkan saran dan kritik yang bersifat membangun untuk memperbaiki tugas akhir ini semoga dapat bermanfaat bagi penulis khususnya dan pembaca pada umumnya.

Wassalamu'alaikum Wr. Wb

Yogyakarta, 12 April 2007

Siti Nurul Jannah

ABSTRAKSI

TOEFL (*Test of English as a Foreign Language*) adalah tes Bahasa Inggris yang digunakan untuk mengukur kecakapan (*proficiency*) berbahasa Inggris seseorang dalam konteks akademis seperti, memasuki institusi tertentu, perusahaan, persyaratan beasiswa, dan alat ukur atau evaluasi atas kemampuan berbahasa Inggris seseorang. Dengan adanya *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .Net* ini diharapkan dapat membantu dalam mempelajari hal-hal tersebut diatas.

Rancang Bangun Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .Net ini dibangun dengan menggunakan bahasa pemrograman VB.NET dan menggunakan *Sql Server 2005* sebagai *database* nya. Pada *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .Net* ini *user* dapat mengetahui jumlah *score* TOEFL yang diperoleh sehingga *user* dapat mengukur dan melatih kemampuannya.

Kata kunci: TOEFL

DAFTAR ISI

JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING	ii
LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR	iii
LEMBAR PENGESAHAN PENGUJI	iv
PERSEMBAHAN	v
MOTTO	vi
KATA PENGANTAR	vii
ABSTRAKSI	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
BAB I PENDAHULUAN	15
1.1 Latar Belakang	15
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metode Penelitian	3
1.6.1 Metode Pengumpulan Data	3
1.6.2 Metode Pengembangan Sistem	4
1.7 Sistematika Penulisan	5
BAB II LANDASAN TEORI	7
2.1 TOEFL (Test of English as a Foreign Language)	7
2.1.1 Perhitungan Score TOEFL	10
2.2 Personal Digital Assistant (PDA) dan Pocket PC	12
2.3 Windows CE	13
2.4 Teknologi .NET	14
2.5 Pemrograman Visual Basic.NET	16
BAB III ANALISIS KEBUTUHAN PERANGKAT LUNAK	18
3.1 Metode Analisis	18
3.1.1 Metode pengumpulan data	19
3.2 Hasil Analisis	19
3.2.1 Masukan Sistem	20
3.2.2 Keluaran Sistem	20
3.2.3 Proses Sistem	21
3.2.4 Antarmuka Sistem	21
BAB IV PERANCANGAN PERANGKAT LUNAK	23
4.1 Metode Perancangan	23
4.2 Hasil Perancangan Perangkat Lunak	24
4.2.1 Perancangan sistem dengan diagram konteks	24
4.2.2 Perancangan model dengan <i>Data Flow Diagram</i> (DFD level 1.0)	25
4.2.3 DFD level 2	26

4.3	Perancangan Basis Data	28
4.3.1	Perancangan tabel	28
4.4	Perhitungan score	31
4.5	Perancangan Antar Muka (<i>Interface</i>).....	33
BAB V IMPLEMENTASI PERANGKAT LUNAK.....		42
5.1	Batasan Implementasi	42
5.1.1	Perangkat Keras	42
5.1.2	Perangkat Lunak	43
5.2	Implementasi Sistem	43
BAB VI ANALISIS KINERJA PERANGKAT LUNAK.....		71
6.1	Pengujian Program	71
6.2	Pengujian Dan Analisis.....	71
6.2.1	Pengujian Normal	71
6.2.2	Pengujian Tidak Normal.....	75
6.2.3	Hasil Analisis	76
6.3	Pembahasan Sistem.....	77
BAB VII PENUTUP.....		78
7.1	Simpulan	78
7.2	Saran.....	78
DAFTAR PUSTAKA		80

DAFTAR GAMBAR

Gambar 4.1 Diagram konteks	24
Gambar 4.2 DFD level 1	25
Gambar 4.3 DFD level 2 proses pengerjaan test.....	26
Gambar 4.4 DFD level 2 proses penampilan score.....	27
Gambar 4.5 Rancangan halaman utama.....	34
Gambar 4.6 Rancangan halaman Menu	35
Gambar 4.7 Rancangan halaman Introduction.....	36
Gambar 4.8 Rancangan halaman Structure.....	37
Gambar 4.9 Rancangan halaman Listening	38
Gambar 4.10 Rancangan halaman Reading	39
Gambar 4.11 Rancangan halaman Full Test	40
Gambar 4.12 Rancangan halaman About	41
Gambar 5.1 Halaman Utama.....	44
Gambar 5.2 Halaman Menu	45
Gambar 5.3 Halaman Introduction.....	46
Gambar 5.4 Halaman About	47
Gambar 5.5 Halaman Listening	48
Gambar 5.6 Halaman Listening Sound	49
Gambar 5.7 Halaman Listening Soal	50
Gambar 5.8 Halaman Listening Score	52
Gambar 5.9 Halaman Structure Utama	53
Gambar 5.10. Halaman Structure Soal.....	54
Gambar 5.11 Halaman Structure Score.....	56
Gambar 5.12 Halaman Structure Explanation	57
Gambar 5.13. Halaman Reading Utama	58
Gambar 5.14 Halaman Reading Soal	59
Gambar 5.15 Halaman Reading Score	61
Gambar 5.16 Halaman Full Test Utama	62
Gambar 5.17 Halaman Full Test Narration.....	63
Gambar 5.18 Halaman Full Test Soal Listening.....	64
Gambar 5.19 Halaman Full Test Soal Structure	65
Gambar 5.20 Halaman Full Test Soal Reading.....	66
Gambar 5.21 Halaman Full Test Score.....	67
Gambar 6.1 Halaman Menu	72
Gambar 6.2 Halaman Structure.....	73
Gambar 6.2 Halaman Structure Soal.....	74
Gambar 6.3 Halaman Structure Score.....	75
Gambar 6.4 Kotak Peringatan.....	76
Gambar 6.5 Kotak Hasil.....	76

DAFTAR TABEL

Tabel 4.1 Tabel Soal Structure.....	29
Tabel 4.2 Tabel Soal Listening	29
Tabel 4.3 Tabel Soal Reading.....	30
Tabel 4.4 Tabel benarL.....	30
Tabel 4.5 Tabel benarS	31
Tabel 4.6 Tabel benarR.....	31
Tabel 4.7 Tabel Score	33

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini *pocket PC* sudah banyak digunakan oleh masyarakat, khususnya para pekerja profesional. Hal ini disebabkan oleh kebutuhan masyarakat akan perangkat digital sebagai alat bantu kerja cukup tinggi. Sehingga dibutuhkan aplikasi-aplikasi tambahan dalam *pocket PC* untuk membantu pekerjaan dari penggunaannya. Kemampuan *pocket PC* yang mengesankan untuk ukurannya yang cukup kecil menantang para pembuat *software* untuk mengembangkan aplikasi-aplikasi pendukung yang dijalankan pada perangkat tersebut. Tentu saja teknik pemrograman tidak dapat langsung diadopsi dari cara-cara pembuatan program yang selama ini dijalankan pada komputer. Para pembuat program harus memperhitungkan kemampuan dan fitur dari perangkat portabel itu, misalnya ukuran layar yang jauh lebih kecil, kecepatan prosessor, kapasitas memori yang terbatas, dan sebagainya [DJU02].

TOEFL merupakan suatu syarat dalam menghadapi dunia kerja sekarang ini. Seseorang dituntut untuk mencapai score TOEFL tertentu untuk dapat lulus suatu institusi ataupun memasuki perusahaan besar, sehingga dengan adanya latihan TOEFL di dalam perangkat mobile yang berbasis Pocket PC ini diharapkan dapat membantu mereka yang kurang memiliki waktu untuk mengikuti kursus TOEFL, serta dapat mempermudah dalam mempelajari dan melatih kemampuan TOEFL dimanapun dan kapanpun mereka memerlukannya.

1.2 Perumusan Masalah

Bagaimana merancang dan mengimplementasikan suatu aplikasi perangkat lunak untuk membantu seseorang dalam mempelajari dan melatih kemampuan TOEFL secara praktis sekaligus mengetahui kemampuan TOEFL seseorang dan score yang diperoleh dan hasil latihan tersebut yang dapat diaplikasikan pada *pocket PC*.

1.3 Batasan Masalah

Pada penelitian agar tidak meluas pada permasalahan lain dan lebih terarah sebagaimana tujuan, penulis membatasi penelitian agar memperoleh suatu solusi yang diinginkan. Batasan tersebut adalah:

1. Perangkat lunak yang akan digunakan untuk membangun aplikasi ini adalah *Visual Basic.Net*.
2. TOEFL yang ditampilkan adalah Listening Comprehension, Structure, Reading Comprehension, dan Full Test.
3. Pembacaan (suara) test hanya pada test Listening.
4. Menampilkan pembahasan materi pada kunci jawaban test tertentu.
5. Menampilkan score dari test yang telah dilakukan pengguna.
6. Jumlah soal untuk Full test adalah 70 soal meliputi, structure test 20 soal, listening test 25 soal, dan reading test 25 soal.
7. Output dan program berupa test latihan TOEFL, yang dapat sekaligus menghitung score TOEFL dan hasil latihan tersebut, serta pembahasan jawaban dari jenis test TOEFL tertentu.

1.4 Tujuan Penelitian

Penelitian yang dilakukan bertujuan untuk menghasilkan program aplikasi TOEFL sehingga pengguna dapat melatih kemampuan TOEFL secara mudah dan praktis serta memberikan solusi untuk keterbatasan waktu dibandingkan dengan mengikuti kursus TOEFL di lembaga pelatihan. Serta membantu dalam persiapan menghadapi test TOEFL yang sebenarnya di lembaga pelatihan resmi.

1.5 Manfaat Penelitian

Dengan adanya penelitian ini diharapkan dapat memberikan beberapa manfaat antara lain:

1. Memberikan solusi alternatif yang bisa membantu seseorang dalam mempelajari dan melatih kemampuan TOEFL secara praktis.
2. Aplikasi yang dibangun dapat dijadikan sebagai panduan tambahan dalam memahami soal-soal TOEFL.
3. Penelitian yang dibuat dapat menjadi acuan bagi pengembangan sistem aplikasi yang lebih baik dan lebih lengkap.

1.6 Metode Penelitian

Adapun metode-metode yang akan digunakan untuk mencapai hasil yang baik dalam penelitian ini adalah sebagai berikut:

1.6.1 Metode Pengumpulan Data

Metode pengumpulan data dilakukan dengan cara mengadakan studi

kepuustakaan, yaitu pengumpulan data dengan cara melakukan studi, analisis dan dokumentasi literatur, dan sumber catatan lain yang berkaitan dengan permasalahan yang dibahas.

1.6.2 Metode Pengembangan Sistem

Metode pengembangan sistem disusun berdasarkan hasil dari data yang sudah diperoleh. Metode ini meliputi:

1. Analisa Data

Analisa ini dilakukan untuk mengolah data yang sudah didapat dan mengelompokkan data sesuai dengan kebutuhan perancangan.

2. Desain

Tahap ini merupakan tahap perancangan sistem, yaitu mendefinisikan kebutuhan yang ada, menggambarkan bagaimana sistem dibentuk dan persiapan untuk rancang bangun aplikasi.

3. Pengkodean

Tahap ini adalah penerjemahan rancangan dalam tahap desain ke dalam bahasa pemrograman komputer yang telah ditentukan sebelumnya.

4. Pengujian

Setelah aplikasi selesai dibuat, maka pada tahap ini merupakan uji coba terhadap program tersebut. Sehingga analisis hasil implementasi yang didapat dari sistem disesuaikan dengan kebutuhan sistem tersebut. Jika penerapan sistem sudah berjalan dengan lancar, maka sistem dapat diimplementasikan.

1.7 Sistematika Penulisan

Untuk memudahkan dalam memahami laporan Tugas Akhir ini dikemukakan sistematika penulisan agar menjadi satu kesatuan yang runtun. Adapun sistematika penulisan laporan Tugas Akhir ini adalah sebagai berikut:

Bab I merupakan pendahuluan berisi latar belakang diambilnya permasalahan yang terjadi pada obyek yang diambil, rumusan masalah, batasan masalah, tujuan yang dikehendaki dalam pemecahan masalah, manfaat penelitian, metodologi penelitian, dan sistematika penulisan laporan.

Bab II merupakan bagian yang menjadi landasan teori yang digunakan dalam memecahkan masalah dan membahas masalah yang ada. Teori-teori yang dicantumkan dalam bab ini yaitu, penjelasan tentang TOEFL, jenis test di dalam TOEFL, penjelasan tentang *pocket PC*, .Net Framework, Windows CE dan *Microsoft Visual Studio*.

Bab III merupakan analisis kebutuhan perangkat lunak berisi tentang analisis kebutuhan perangkat lunak yang akan dipakai dalam perancangan aplikasi TOEFL. Bab ini meliputi metode analisis, hasil analisis, masukan sistem, keluaran sistem, proses sistem, dan antarmuka yang diinginkan dalam pembuatan sistem.

Bab IV merupakan perancangan perangkat lunak yang membahas tentang metode perancangan dan hasil perancangan perangkat lunak yang digunakan untuk pembuatan sistem, perancangan basis data serta perancangan antarmuka sistem.

Bab V merupakan implementasi perangkat lunak. Bagian ini memuat batasan implementasi perangkat keras dan perangkat lunak yang dibutuhkan, hasil implementasi yang dibutuhkan meliputi prosedur-prosedur antarmuka.

Bab VI merupakan analisis kinerja perangkat lunak. Bagian ini memuat dokumentasi pengujian terhadap kinerja perangkat lunak yang telah dibuat meliputi data hasil uji, hasil pengujian, dan analisis hasil pengujian.

Bab VII merupakan bagian penutup. Pada bab ini membahas mengenai kesimpulan yang dapat diambil dari seluruh langkah dan tahapan-tahapan yang sudah dilakukan dan beberapa saran yang akan bermanfaat bagi pihak-pihak yang berkepentingan maupun bagi penelitian lebih lanjut.

BAB II

LANDASAN TEORI

2.1 TOEFL (Test of English as a Foreign Language)

Genderang perang globalisasi telah ditabuh dengan keikutsertaan Indonesia dalam World Trade Area (AFTA) dan Asia Pacific Economic Cooperation (APEC). Hakekat globalisasi adalah kerjasama dalam persaingan. Bagi suatu negara yang sudah menyiapkan infrastruktur kuat, globalisasi adalah peluang. Tetapi bagi negara yang belum memiliki infrastruktur kuat, globalisasi adalah ancaman. Dalam kaitannya dengan globalisasi pasar kerja, Indonesia termasuk negara dalam kelompok kedua [<http://www.brawijaya.ac.id/main/news>].

Globalisasi membawa dampak ganda bagi tenaga kerja Indonesia. Jutaan sarjana dan lulusan sekolah menengah kejuruan mendapat kesempatan kerja baik di dalam maupun luar Indonesia. Namun terdapat persaingan yang tajam dan ketat karena arus barang maupun mobilitas tenaga kerja asing tidak dapat dibendung masuk ke Indonesia. Oleh karena itu tantangan utama sekarang adalah peningkatan daya saing dan keunggulan kompetitif dan sektor sumber daya manusia. Bahkan Badan Litbang Dan Informasi Depnakertrans, telah menetapkan rencana strategis dalam bidang pembangunan SDM, yaitu peningkatan kualitas dan daya saing tenaga kerja sesuai keahlian, profesionalitas dan kompetensi melalui pelatihan yang bersifat strategis sesuai kebutuhan pasar serta standarisasi dan sertifikasi profesi untuk menjamin kualitas tenaga kerja.

Jelaslah bahwa globalisasi menciptakan atmosfer kompetitif yang menuntut kemampuan individu untuk bersaing secara sehat. Salah satu aspek penentunya ialah kompetensi komunikasi dalam bahasa Inggris, yang merupakan hal vital dalam komunikasi global. Kenyataannya, kemampuan berbahasa Inggris tenaga kerja Indonesia lebih rendah dibanding dengan negara-negara Asia lainnya. Akibatnya, daya saing tenaga kerja Indonesia menjadi rendah. Perawat Filipina, contohnya, lebih dipilih di Singapura bukan karena profesionalitas, melainkan karena kemampuan berbahasa Inggris yang lebih baik. Bila tidak ada upaya peningkatan kualitas dan standarisasi bahasa Inggris, maka niscaya globalisasi benar-benar akan merupakan ancaman bagi bangsa Indonesia.

TOEFL adalah tes Bahasa Inggris yang dikembangkan oleh Educational Testing Service (ETS) di New Jersey , Amerika Serikat sejak 1963. Tes ini mengukur kecakapan (*proficiency*) berbahasa Inggris seorang penutur asing dalam konteks akademis, dan dengan demikian tes ini digunakan untuk tujuan akademis seperti persyaratan penerimaan mahasiswa (terutama di Amerika dan Kanada) atau *screening* beasiswa. Jenis TOEFL yang paling dikenal adalah *Paper-Based TOEFL* dengan skala nilai 320-660. TOEFL-ITP (*Institutional Testing Program*), skala nilai 310-677, dikenalkan pada tahun 1965. ETS memberi kesempatan pada beberapa universitas untuk menyelenggarakan TOEFL-ITP dengan biaya US \$17.50 untuk keperluan internal institusi tersebut, dan oleh karenanya bukan tes internasional. Pada tahun 1998 tes ini dikomputerisasi dan dinamai *Computer-Based TOEFL* atau TOEFL-CBT, nilainya 0-300, dengan biaya US \$110.

Pada kenyataannya di berbagai tempat, di Indonesia dan Cina khususnya, menjamur berbagai lembaga yang menawarkan tes TOEFL-LIKE/EQUIVALENT dengan biaya sangat murah, Rp 25.000-Rp 75.000, dan soal dicopy sendiri dan beberapa sumber. Tentu saja mereka bukan agen resmi ETS. *Validitas* dan *kredibilitas* nilai dan sertifikat yang dikeluarkannya sering dipertanyakan. Selain untuk memasukkan kecakapan *Speaking* dan mengatasi masalah pembajakan ini, ETS secara bertahap menghapus *Paper-Based TOEFL* dan TOEFL-CBT dan menggantinya dengan NG-TOEFL (*New Generation TOEFL*) yang hanya bisa diakses langsung ke markas ETS melalui Internet di *center* resmi ETS dengan biaya US \$140, dan oleh karenanya juga disebut *Internet-Based Test TOEFL* (IBT TOEFL), dengan skala nilai 0-120.

TOEFL dimaksudkan sebagai alat ukur atau evaluasi atas kemampuan berbahasa Inggris seseorang. TOEFL sendiri ada tiga jenis, yaitu TOEFL International, TOEFL ITP, dan TOEFL Prediction. TOEFL Internasional sering juga disebut sebagai computer-based TOEFL(CBT) karena bentuk tesnya megunakan komputer. TOEFL ITP berupa tes tertulis, dan sering disebut paper-based TOEFL (PBT). Sedangkan TOEFL Prediction merupakan bentuk tes yang mirip TOEFL, tetapi tidak dikeluarkan oleh ETS [<http://www.cdc.eng.ui.ac.id/news>].

Kini, CBT sudah banyak digunakan sejumlah lembaga pendidikan dan kantor untuk mengetahui kemampuan bahasa Inggris calon mahasiswa atau karyawan. Seperti tes lainnya, CBT mengandung empat unsur, yaitu menyimak

(listening), berbicara (speaking) yang biasanya diganti dengan gramatika (structure), membaca (reading), serta menulis (writing).

Listening dimaksudkan untuk mengukur kemampuan mendengar dan menyimak bahasa Inggris, structure untuk mengukur kemampuan berbahasa Inggris dengan gramatika baku, reading untuk mengukur kemampuan memahami bacaan, dan writing sebagai sarana untuk mengukur kemampuan mengungkapkan gagasan dalam tulisan berbahasa Inggris. Tes TOEFL ini memakan waktu sekitar 2-3 jam.

Dengan adanya *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET* ini diharapkan dapat mempermudah dalam pembelajaran pengetahuan tentang sistem bahasa Inggris (Grammar & Structure), meningkatkan dan memperdalam kemampuan maupun keterampilan dalam menyimak teks-teks baik lisan maupun tertulis. Serta membantu dalam mempersiapkan diri menghadapi TOEFL Test yang diharapkan kemudian dapat diterapkan dalam latihan-latihan soal dan Tes TOEFL yang sesungguhnya.

2.1.1 Perhitungan Score TOEFL

Ketika kita mengikuti test TOEFL, kita akan mendapat score antara 20-68 dari setiap section. Dimana di dalam test TOEFL ini terdapat tiga section, yaitu Listening Comprehension, Structure, dan Reading Comprehension. Berikut ini adalah table estimasi nilai yang diperoleh beserta cara perhitungan TOEFL:

Number Correct	Converted Score Section 1	Converted Score Section 2	Converted Score Section 3
25	68		67
24	66		65
23	63		61
22	61		59
21	59		57
20	57	68	55
19	56	65	54
18	54	61	52
17	53	58	51
16	52	56	49
15	51	54	48
14	49	52	46
13	48	50	45
12	47	48	43
11	46	46	42
10	45	44	40
9	43	42	38
8	41	40	36
7	37	38	34
6	37	36	31
5	33	33	29
4	32	29	28
3	30	26	26
2	28	23	24
1	26	21	23
0	24	20	21

Sumber: Lembaga Pendidikan Indonesia Amerika (LPIA)

Dari tabel diatas dapat dihitung perolehan score. Sebagai contoh berikut ini:

	Section 1	Section 2	Section 3
Number correct	15	10	12
Converted score	51	44	43

Kemudian, didapat perhitungan score dengan cara:

1. Tambahkan ketiga converted score: $51+44+43$
2. Jumlah yang diperoleh dibagi 3: $138/3 = 46$
3. Kemudian hasil pembagian dikali 10: $46 \times 10 = 460$

Score TOEFL yang diperoleh adalah 460.

2.2 Personal Digital Assistant (PDA) dan Pocket PC

PDA adalah komputer berukuran kecil yang mampu menyimpan dan mengelola informasi. Pada umumnya PDA bekerja dengan sistem operasi *Windows CE* atau *Palm OS*. Pada awalnya PDA ditujukan sebagai kalender digital sederhana, namun pada perkembangannya PDA digunakan untuk kerja sehari-hari, bermain game, menjalankan musik dan mendownload informasi dan internet.

Pada dasarnya PDA dibedakan menjadi 2 jenis, yaitu Pocket PC dan Handheld PC. Perbedaan utamanya terletak pada ukuran, tampilan dan cara pengoperasiannya. Dibandingkan Pocket PC, Handheld PC cenderung lebih besar dan berat. Tampilannya lebih lebar dan untuk memasukkan data digunakan miniatur papan ketik atau layar sentuh [DJU02]. Bagian-bagian standar sebuah PDA adalah:

- a. Mikroprosesor
- b. Memory
- c. Layar
- d. Baterai
- e. Perangkat input

Pocket PC dikemas dengan berbagai program aplikasi yang didesain untuk mendukung produktivitas dan hiburan, seperti aplikasi:

- a. *Contacts, Calendar dan Notes* (yang biasa dinamakan *Pocket Outlook*)
- b. *Pocket Word*
- c. *Pocket Excel*
- d. *Pocket Internet Explorer*
- e. *Windows Media Player For Pocket PC*
- f. *Microsoft Reader*

Fungsi-fungsi dasar yang dapat dilakukan oleh PDA adalah menyimpan daftar alamat dan nomor telepon, jadwal dan kalender pribadi, dan membuat catatan kecil. PDA yang lebih canggih dapat menjalankan program aplikasi pengolah kata, spreadsheet, buku elektronik, bahkan email dan akses internet. Beberapa PDA dijual dengan beberapa aplikasi yang sudah terinstal di dalamnya, sementara program-program aplikasi lain bisa diinstal kemudian. PDA juga memungkinkan untuk bertukar informasi dengan Personal Computer (PC).

2.3 Windows CE

Windows CE adalah sebuah sistem operasi 32 bit yang bersifat modular dan real-time dengan ukuran yang kecil sehingga cocok diimplementasikan pada perangkat elektronik dengan komputer didalamnya. *Windows CE* mengkombinasikan keunggulan dan kompatibilitas windows serta pengembangan tahap lanjut yang mendukung berbagai arsitektur prosesor dan kemampuan komunikasi dan dukungan jaringan komputer sehingga menjadikannya landasan yang terbuka, skalabel dan lengkap untuk diimplementasikan pada berbagai

produk berdasarkan *Windows CE* seperti perangkat kontrol pada industri, telepon, kamera hingga perangkat hiburan.

Pada tahun 2000, *Microsoft* meluncurkan sistem operasi *Windows CE* generasi baru dan diberi nama *Microsoft Windows for Pocket PC* (disingkat *Pocket PC*). Sistem operasi ini dibuat berdasarkan teknologi windows CE 3.0. pengembangan dilakukan pada *kernel*, dukungan *driver* dan *hardware*, protocol komunikasi dan sekuriti, sehingga pada akhirnya sistem operasi ini menarik perhatian para pembuat komputer seperti Hewlett-Packard, Casio, dan Compaq. *Pocket PC* tidak dimaksudkan untuk menggantikan *Windows CE*. *Windows CE* adalah sistem operasi yang akan terus dikembangkan untuk berbagai komputer portabel.

2.4 Teknologi .NET

.NET Platform merupakan satu set kumpulan teknologi yang memungkinkan teknologi internet ditransformasikan ke dalam platform *distributed computing* dengan skalabilitas dan komabilitas tinggi. Secara teknikal, *.NET Platform* menyediakan konsep pemrograman dengan *library* dan modul-modul baru yang konsisten, terlepas dari jenis bahasa pemrograman yang digunakan.

.NET Platform menyediakan hal-hal berikut bagi para developer:

- a. *Language independent*, dengan programming model yang konsisten di semua *tier* aplikasi yang dibangun.
- b. *Interoperability* dan komabilitas antar aplikasi.

- c. Kemudahan migrasi teknologi yang ada saat ini.
- d. Dukungan penuh terhadap berbagai teknologi standar yang digunakan dalam platform internet, antara lain HTTP, XML, SOAP, dan HTML.

Teknologi inti .NET secara umum terdiri dari 4 area pokok, yaitu:

a. NET Framework

.NET Framework adalah teknologi inti yang menyediakan berbagai library untuk digunakan oleh aplikasi di atasnya. Komponen inti *.NET Framework* adalah *Common Language Runtime (CLR)* yang menyediakan *run time environment* untuk aplikasi yang dibangun menggunakan *Visual Studio .NET*, terlepas dari jenis bahasa pemrogramannya

b. NET Building Block Services

Building Block merupakan sekumpulan servis yang bersifat *programmable*, yang dapat diakses secara *offline* maupun *online*. Servis tersebut merupakan modul-modul yang terdapat di suatu komputer, *server* dalam jaringan, maupun di suatu *server* di internet.

c. Visual Studio .NET

Visual Studio .NET menyediakan *tools* bagi para developer untuk membangun aplikasi yang berjalan di *.NET Framework*. *Visual Studio .NET* membawa perubahan besar dalam gaya pemrograman, karena setiap *programmer* dituntut untuk memahami *.NET object model* dan *Object Oriented Programming* dengan baik, jika tidak ingin menghasilkan dengan performa rendah. *Visual Basic .NET* juga semakin mempertipis jarak antara *Windows Programmer* dengan *Web Programmer*.

d. NET Enterprise Server

Bagian ini merupakan sekumpulan *server based technology* yang digunakan untuk mendukung teknologi .NET, yang mencakup sistem operasi, *database*, *messaging*, maupun manajemen *e-commerce*. Teknologi yang disediakan antara lain: *Windows 2000 Server*, *SQL Server*, *Exchange*, *ISA Server*, dan *BiZTalk Server*.

2.5 Pemrograman Visual Basic.NET

Hampir semua orang, terutama para pemrogram, mengenal dengan baik Visual Basic, yaitu bahasa pemrograman berbasis *Graphical User Interface* (GUI) buatan Microsoft yang mampu membuat setiap pekerjaan menjadi lebih mudah serta mampu meningkatkan produktivitas pemrogram. Seiring dengan semakin meningkatnya kebutuhan didalam pengembangan perangkat lunak, pengembang Visual Basic melihat ada beberapa kelemahan pada Visual Basic seperti tidak memiliki kemampuan *multi-threading*, yakni kemampuan yang memungkinkan suatu tugas dapat dijalankan pada *thread* terpisah. Untuk memperbaiki kekurangan dan Visual Basic, akhirnya microsoft mengeluarkan Visual Basic .NET. Bersama dengan C++ dan C#, Visual Basic .NET merupakan bahasa pemrograman yang sudah termasuk didalam paket Microsoft Visual Studio.NET [AKB05].

Visual Studio .NET sendiri menyediakan lingkungan pengembangan high-level untuk membangun aplikasi-aplikasi pada .NET Framework. Di lingkungan

inilah kita akan merasakan teknologi yang mampu menyediakan pembuatan dan penyebaran aplikasi. Selain itu, Visual Studio .NET juga menawarkan generasi baru aplikasi berbasis Windows dengan fitur-fitur yang tersedia melalui .NET Framework.

Kelebihan yang dimiliki bahasa pemrograman *Visual Basic .Net*, yaitu:

1. Dalam beberapa hal kotak dialog yang muncul, nama menu, tampilannya dibuat menarik, dan mudah dalam pemakaiannya.
2. Adanya Emulator PocketPC yang memudahkan untuk mencoba penggunaan aplikasi dari sistem yang dibuat.

BAB III

ANALISIS KEBUTUHAN PERANGKAT LUNAK

3.1 Metode Analisis

Metode analisis merupakan suatu cara dalam menguraikan sebuah komputer melalui identifikasi, perancangan, dan implementasi suatu perangkat lunak. Melalui penelitian terhadap keadaan suatu komputer dengan tujuan untuk merancang, menyempurnakan, atau memperbaharui komputer yang telah ada. Tahap analisis juga mengevaluasi permasalahan-permasalahan dan hambatan-hambatan yang terjadi didalam membangun suatu komputer sehingga dapat dilakukan perbaikan.

Analisis merupakan tahap yang paling penting, karena kesalahan pada tahap ini akan menyebabkan kesalahan pada tahap berikutnya. Proses pengembangan suatu informasi dengan analisis proses yang telah ada, berguna untuk mengetahui seberapa jauh hal tersebut telah mencapai sasaran yang diinginkan.

Metode yang digunakan dalam proses analisis *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET* adalah metode pengumpulan data. Pengumpulan data adalah metode yang digunakan untuk mengumpulkan data yang diperlukan selama penelitian berlangsung. Dalam penelitian ini metode pengumpulan data yang digunakan adalah studi literatur. Metode tersebut dilakukan dikarenakan sistem yang akan dibangun merupakan aplikasi TOEFL, sehingga membutuhkan data-data TOEFL.

3.1.1 Metode pengumpulan data

Metode pengumpulan data merupakan cara pertama yang digunakan dalam melakukan penelitian untuk membangun *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET*. Metode pengumpulan data terdiri dari:

a. Wawancara

Pengumpulan data yang dilakukan dalam proses pembuatan *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET* adalah dengan wawancara secara langsung tentang cara perhitungan score TOEFL.

b. Library research

Mengumpulkan dari buku-buku referensi, dan artikel-artikel yang sesuai dengan penelitian. Buku referensi untuk pembuatan *database* dalam *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET* adalah *BARRON'S : How To Prepare For The TOEFL*, *BARRON'S: TOEFL iBT*, dan buku-buku lain yang mendukung penelitian.

3.2 Hasil Analisis

Berdasarkan analisis yang telah dilakukan maka dapat diketahui apa saja yang akan menjadi masukan sistem, keluaran sistem, fungsi atau metode yang digunakan oleh sistem, kebutuhan perangkat keras, kebutuhan perangkat lunak serta antarmuka sistem yang akan dibuat, sehingga sistem yang dibangun sesuai dengan apa yang diharapkan.

3.2.1 Masukan Sistem

Kebutuhan input (masukan) berupa data dari programmer yang ditujukan untuk memenuhi kebutuhan dalam implementasi *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi.NET* ini, antara lain:

1. Panduan TOEFL (Introduction TOEFL, panduan test Structure, panduan test Listening, panduan test Reading).
2. Masukan Test Structure yang terdiri dari ID soal, soal, pilihan jawaban, jawaban soal, pembahasan soal.
3. Masukan Test Reading yang terdiri dari ID soal, soal, pilihan jawaban, jawaban soal.
4. Masukan Test Listening yang terdiri dari ID soal, soal, pilihan jawaban, jawaban soal, dan suara
5. Masukan Test Scoring yaitu perhitungan jumlah score yang dicapai pengguna yang didapat dari test yang telah dilakukan user.

3.2.2 Keluaran Sistem

Kebutuhan *output* berupa data hasil dari proses yang terdapat dalam implementasi komputer, antara lain:

1. Tampilan data keseluruhan dari test latihan TOEFL.
2. Score yang telah dicapai
3. Pembahasan soal test tertentu



3.2.3 Proses Sistem

Kebutuhan proses meliputi input yang dibutuhkan dalam proses yang ada dalam komputer yaitu proses mengelola TOEFL. Proses ini memiliki dua anak proses yaitu proses pengaksesan/pengerjaan test TOEFL, proses penampilan hasil score TOEFL. Pada proses ini untuk pengaksesan *user* memilih latihan test yang akan dicoba dan keluaran yang dihasilkan berupa data soal test TOEFL yang terdapat dalam database. Setelah *user* memilih jenis test yang diinginkan, maka *user* dapat mengerjakan test, setelah *user* selesai, system akan mencocokkan jawaban *user* dengan jawaban yang benar di dalam database, kemudian melakukan perhitungan jumlah score, dan menampilkannya kepada *user*.

3.2.4 Antarmuka Sistem

Antarmuka atau *interface* merupakan suatu sarana yang memungkinkan terjadinya interaksi antara manusia dan komputer. Oleh karena itu antarmuka dari sistem yang akan dibangun harus *user friendly* agar pengguna dapat dengan mudah menggunakannya tanpa harus mempelajarinya terlebih dahulu. Antarmuka yang dibutuhkan dalam pembuatan sistem ini antara lain:

1. Halaman utama yang memuat menu-menu
2. Antarmuka untuk Introduction
3. Antarmuka untuk Listening Test
4. Antarmuka untuk Structure Test
5. Antarmuka untuk Listening Test
6. Antarmuka untuk Reading Test

7. Antarmuka untuk Full Test
8. Antarmuka untuk Scoring
9. Antarmuka untuk Pembahasan Soal
10. Antarmuka untuk Tentang program

BAB IV

PERANCANGAN PERANGKAT LUNAK

4.1 Metode Perancangan

Metode perancangan yang digunakan dalam membangun *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET* ini adalah metode perancangan terstruktur (*Structured Design Method*). Metode perancangan ini diawali dengan perancangan diagram konteks yang sifatnya lebih global. Yang bertujuan untuk membuat model sistem yang baru secara logik dan konseptual. Kemudian dilanjutkan dengan perancangan *Data Flow Diagram* yang lebih detail dari tiap-tiap bagian sistem. *Data Flow Diagram* merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur. *Data Flow Diagram* juga merupakan diagram yang menggunakan notasi-notasi untuk menggambarkan arus data dari sistem secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut disimpan.

Selain menggunakan *Data Flow Diagram*, pada perancangan sistem ini juga menggunakan tabel, struktur pada *database* dan desain *interface*. Perancangan yang disebutkan diatas ditujukan untuk memberikan kemudahan kepada *user* dalam mendapatkan informasi serta ikut berinteraksi dalam memberikan kontribusi data.

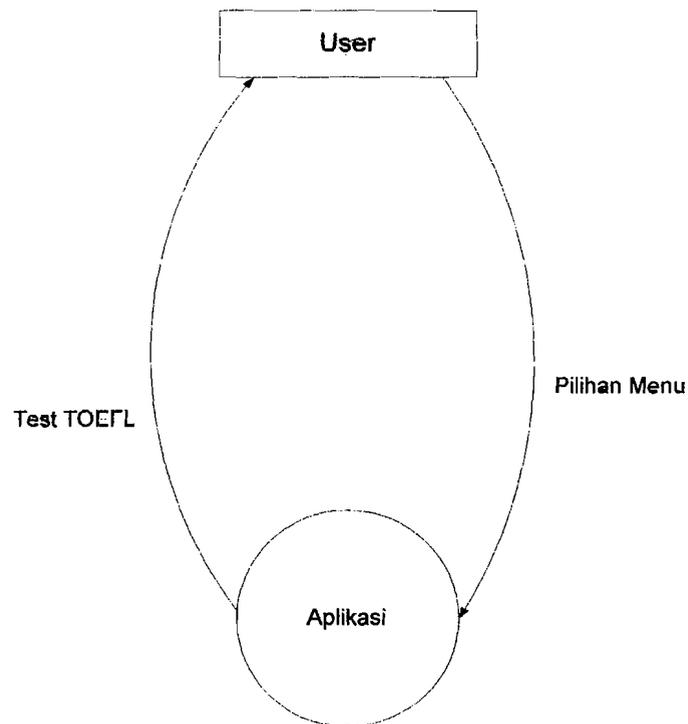
4.2 Hasil Perancangan Perangkat Lunak

Hasil perancangan sistem harus sesuai dengan penerapan metode-metode dalam perancangan yaitu metode perancangan terstruktur (*Structured Design Method*), antara lain:

4.2.1 Perancangan sistem dengan diagram konteks

Diagram konteks menggambarkan *input output* antara sistem dengan satuan luarnya. Perancangan dari diagram konteks meliputi proses dari keseluruhan sistem.

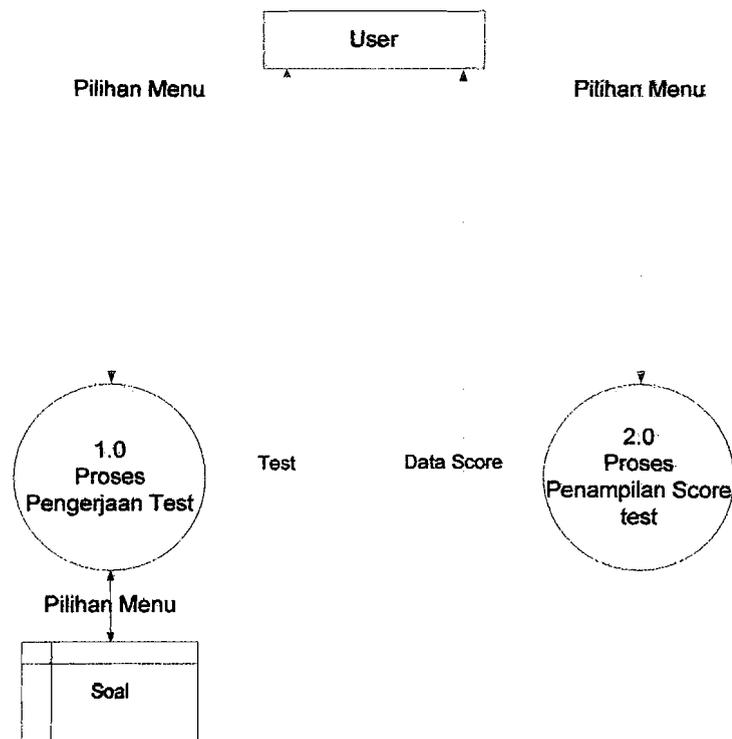
Bentuk dari diagram konteks pada *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET* ini dapat dilihat pada gambar 4.1 berikut ini:



Gambar 4.1 Diagram konteks

4.2.2 Perancangan model dengan *Data Flow Diagram* (DFD level 1.0)

DFD level 1.0 merupakan pengembangan atau penjelasan secara detail bagian per bagian dari diagram konteks, yang dijabarkan menjadi beberapa perancangan proses yang saling terkait. Sehingga lebih memperjelas sistem. Berikut ini bentuk DFD level 1.0 yang dapat dilihat pada gambar 4.2 berikut ini:



Gambar 4.2 DFD level 1

Ket: Data dimasukan oleh *programmer*.

Dari DFD level 1 proses pengerjaan test diatas diperoleh dua anak proses sebagai berikut:

1.0 Proses pengerjaan test.

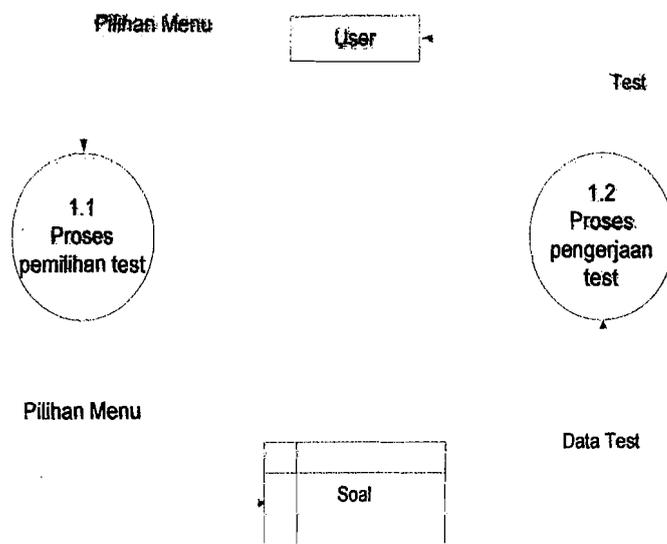
Proses pengerjaan ini akan dijalankan pada saat *user* memilih jenis test TOEFL yang tersedia dan memulai mengerjakan test yang telah dipilihnya.

2.0 Proses penampilan score.

Proses ini akan menampilkan hasil score dari test yang telah dilakukan oleh *user*.

4.2.3 DFD level 2

DFD level 2 merupakan penyederhanaan dari DFD level 1, seperti yang terlihat pada gambar 4.3. dan gambar 4.4.



Gambar 4.3 DFD level 2 proses pengerjaan test

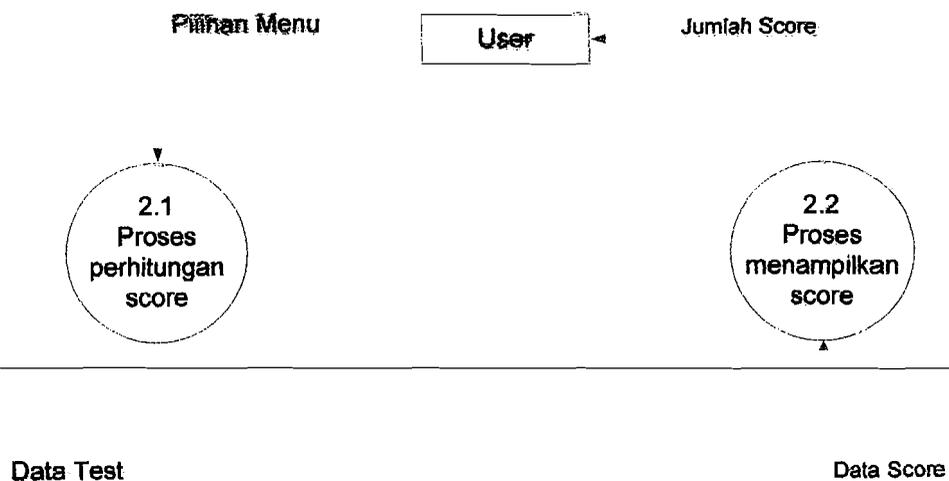
Dari DFD level 2 proses pengerjaan test diatas diperoleh dua anak proses sebagai berikut:

1.1 Proses pemilihan test.

Proses ini akan dijalankan pada saat *user* memilih jenis test yang disediakan oleh aplikasi. *User* akan mengakses jenis test yang telah dipilihnya.

1.2 Proses pengerjaan test

Proses ini dilakukan oleh *user* setelah memilih jenis test, maka *user* akan mulai mengerjakan soal-soal test yang ditampilkan oleh aplikasi.



Gambar 4.4 DFD level 2 proses penampilan score

2.1 Proses perhitungan score.

Proses ini dilakukan oleh aplikasi setelah user selesai melakukan suatu test, maka aplikasi akan mencocokkan jawaban dari *user* dengan jawaban yang benar pada database dan melakukan perhitungan jumlah score yang dicapai oleh *user*.

2.2 Proses menampilkan score

Proses ini dilakukan oleh aplikasi setelah perhitungan score test diperoleh, maka aplikasi akan menampilkan jumlah score kepada user.

4.3 Perancangan Basis Data

Basis data (*database*) merupakan basis penyedia informasi bagi para *user*. Semua data yang dimasukkan oleh *admin* atau pembuat aplikasi ini disimpan berupa file yang disebut *database*. Berikut ini perancangan *database* pada *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET*.

4.3.1 Perancangan tabel

Dalam *database* data terorganisir dalam beberapa komponen yang terdiri dari satu atau lebih tabel. Tabel merupakan fasilitas yang tersedia dalam *database* yang berfungsi untuk menyimpan data, terdiri dari baris dan kolom, dimana kolom merupakan *field* dan baris merupakan jumlah data yang terdapat dalam tabel atau disebut juga dengan *record*. Berikut ini adalah tabel yang digunakan dalam *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET*.

a. Tabel Soal1 (tabel soal structure)

Tabel Soal structure (Tabel 4.1) digunakan untuk menampung ID soal, soal, pilihan jawaban, jawaban, dan pembahasan. Struktur tabelnya dapat dilihat pada tabel 4.1.

Tabel 4.1 Tabel Soal Structure

No	Field	Type	Keterangan
1	ID	int	Nomor soal TOEFL
2	soal	nvarchar	Soal TOEFL structure
3	rbA	nvarchar	Pilihan jawaban A
4	rbB	nvarchar	Pilihan jawaban B
5	rbC	nvarchar	Pilihan jawaban C
6	rbD	nvarchar	Pilihan jawaban D
7	jawaban	nvarchar	Jawaban soal
8	pembahasan	nvarchar	pembahasan jawaban

b. Tabel Soal2 (tabel soal listening)

Tabel Soal listening (Tabel 4.2) digunakan untuk menampung ID soal, soal, pilihan jawaban, jawaban. Struktur tabelnya dapat dilihat pada tabel 4.2.

Tabel 4.2 Tabel Soal Listening

No	Field	Type	Keterangan
1	ID	int	Nomor soal TOEFL
2	soal	nvarchar	Soal TOEFL listening
3	rbA	nvarchar	Pilihan jawaban A
4	rbB	nvarchar	Pilihan jawaban B
5	rbC	nvarchar	Pilihan jawaban C

6	rbD	nvarchar	Pilihan jawaban D
7	jawaban	nvarchar	Jawaban soal

c. Tabel Soal3 (tabel soal Reading)

Tabel Soal reading (Tabel 4.3) digunakan untuk menampung ID soal, soal, pilihan jawaban, jawaban. Struktur tabelnya dapat dilihat pada tabel 4.3.

Tabel 4.3 Tabel Soal Reading

No	Field	Type	Keterangan
1	ID	int	Nomor soal TOEFL
2	soal	nvarchar	Soal TOEFL reading
3	rbA	nvarchar	Pilihan jawaban A
4	rbB	nvarchar	Pilihan jawaban B
5	rbC	nvarchar	Pilihan jawaban C
6	rbD	nvarchar	Pilihan jawaban D
7	jawaban	nvarchar	Jawaban soal

d. Tabel benarL

Tabel benarL (Tabel 4.4) digunakan untuk mencocokkan jumlah benar soal Listening dengan nilai konversi. Struktur tabelnya dapat dilihat pada tabel 4.4.

Tabel 4.4 Tabel benarL

No	Field	Type	Keterangan
1	Benar	int	Jumlah benar jawaban soal Listening
2	converted	int	Nilai konversi jumlah benar

e. Tabel benarS

Tabel benarS (Tabel 4.5) digunakan untuk mencocokkan jumlah benar soal Structure dengan nilai konversi. Struktur tabelnya dapat dilihat pada tabel 4.5.

Tabel 4.5 Tabel benarS

No	Field	Type	Keterangan
1	Benar	int	Jumlah benar jawaban soal Structure
2	converted	int	Nilai konversi jumlah benar

f. Tabel benarR

Tabel benarR (Tabel 4.6) digunakan untuk mencocokkan jumlah benar soal Reading dengan nilai konversi. Struktur tabelnya dapat dilihat pada tabel 4.5.

Tabel 4.6 Tabel benarR

No	Field	Type	Keterangan
1	Benar	int	Jumlah benar jawaban soal Reading
2	converted	int	Nilai konversi jumlah benar

4.4 Perhitungan score

Score test didapat setelah *user* selesai mengerjakan soal dari test yang telah dipilihnya. Algoritma untuk perhitungan soal ini adalah sebagai berikut:

1. Hitung semua jawaban yang benar untuk tiap kategori soal. Jumlah jawaban yang benar tersebut disebut dengan score mentah (raw score).

2. Ubah score mentah tersebut dengan score ubahan (converted score) yang dapat dilihat pada tabel konversi nilai.
3. Setelah didapat masing-masing score ubahan, hitung seluruh jumlah score dengan cara menjumlahkan semua score tersebut.
4. untuk mendapatkan nilai akhir, bagi jumlah keseluruhan yang telah diperoleh tersebut dengan 3 kemudian dikali 10. Misal,

	Section 1	Section 2	Section 3
Number correct	15	10	12
Converted score	51	44	43

Kemudian, didapat perhitungan score dengan cara:

1. Tambahkan ketiga converted score: $51+44+43= 138$
2. Jumlah yang diperoleh dibagi 3: $138/3 = 46$
3. Kemudian hasil pembagian dikali 10: $46 \times 10 = 460$
4. Score TOEFL yang diperoleh adalah 460.

Hasil jawaban yang benar dapat dicocokkan pada tabel konversi. Tabel konversi dapat dilihat pada tabel 4.7.

Tabel 4.7 Tabel Score

Number Correct	Converted Score Section 1	Converted Score Section 2	Converted Score Section 3
25	68		67
24	66		65
23	63		61
22	61		59
21	59		57
20	57	68	55
19	56	65	54
18	54	61	52
17	53	58	51
16	52	56	49
15	51	54	48
14	49	52	46
13	48	50	45
12	47	48	43
11	46	46	42
10	45	44	40
9	43	42	38
8	41	40	36
7	37	38	34
6	37	36	31
5	33	33	29
4	32	29	28
3	30	26	26
2	28	23	24
1	26	21	23
0	24	20	21

Sumber: Lembaga Pendidikan Indonesia Amerika (LPIA)

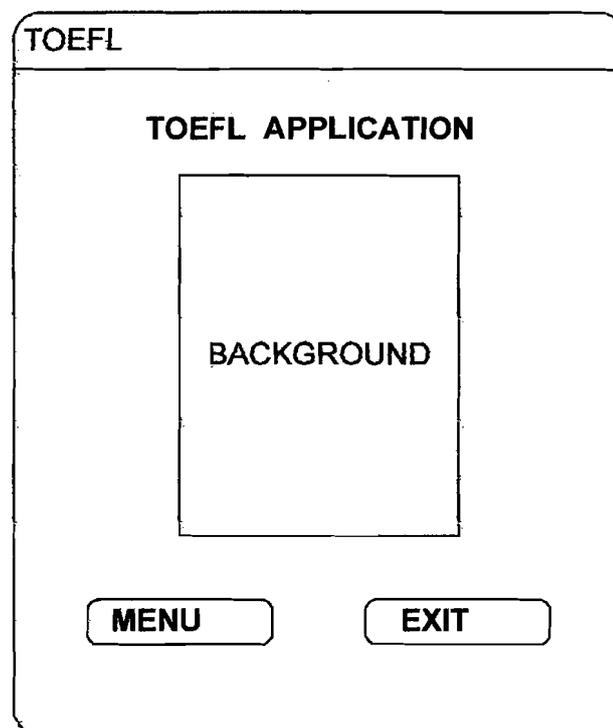
4.5 Perancangan Antar Muka (*Interface*)

Desain antar muka (*interface*) merupakan hasil implementasi dari kebutuhan *user* dengan Pocket PC. Desain *interface* yang utama ditujukan kepada *user*, dimana *interface* didesain sedemikian rupa untuk memudahkan penggunaan sistem aplikasi ini. Desain sederhana aplikasi ini akan memberikan kenyamanan

penggunaan oleh *user*. Berikut desain *interface* pada *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET*

a. Halaman Utama

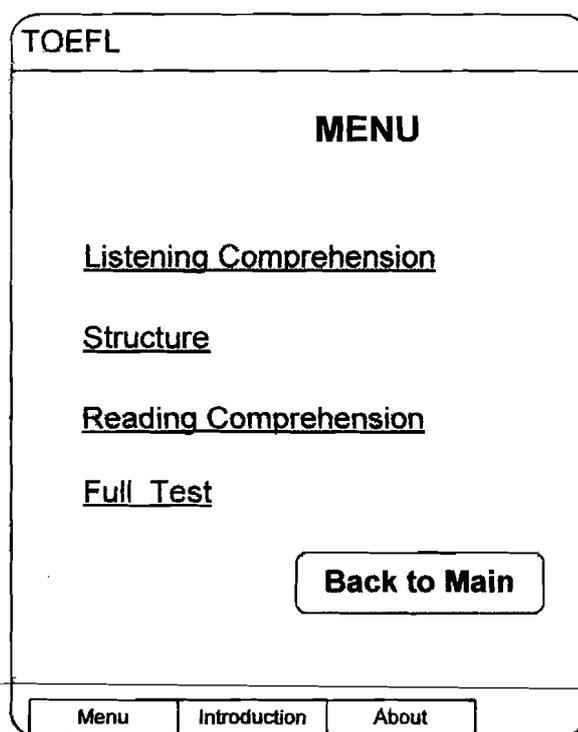
Halaman utama merupakan halaman depan dari *Aplikasi TOEFL untuk Pocket PC dengan Teknologi .NET*. Dan halaman ini didesain sebaik mungkin agar *user* lebih mudah menjalankan atau agar lebih bersifat *user friendly*. Pada halaman ini menampilkan fitur-fitur yang ada pada sistem. Rancangan dari halaman utama ada pada gambar 4.5.



Gambar 4.5 Rancangan halaman utama

b. Halaman Menu TOEFL

Halaman menu merupakan halaman yang menampilkan pilihan-pilihan menu yang dapat digunakan user untuk ditampilkan dan juga terdapat tombol yang digunakan untuk kembali ke halaman utama. Rancangan dari halaman menu ada pada gambar 4.6.

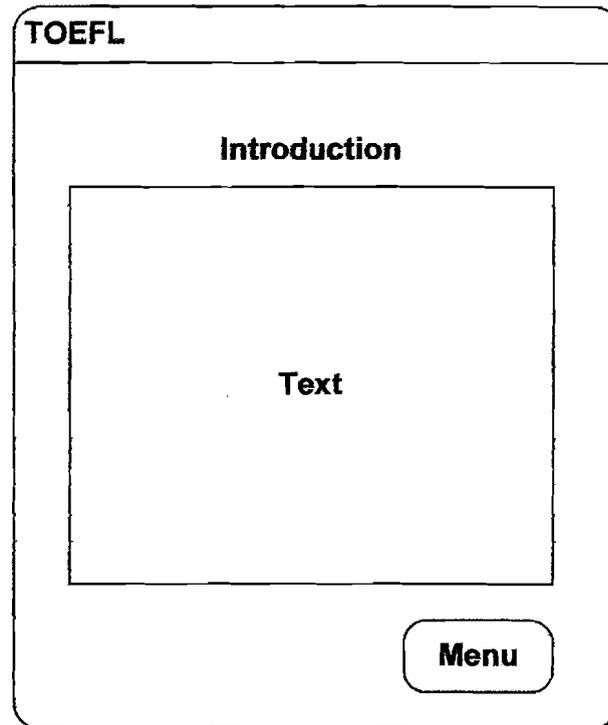


Gambar 4.6 Rancangan halaman Menu

c. Halaman Introduction

Halaman Introduction merupakan halaman yang menampilkan informasi seputar TOEFL, test-test yang terdapat aplikasi TOEFL, panduan umum tentang

TOEFL yang dapat diakses oleh *user*, dan terdapat tombol untuk kembali ke halaman Menu. Rancangan dari halaman introduction ada pada gambar 4.7.

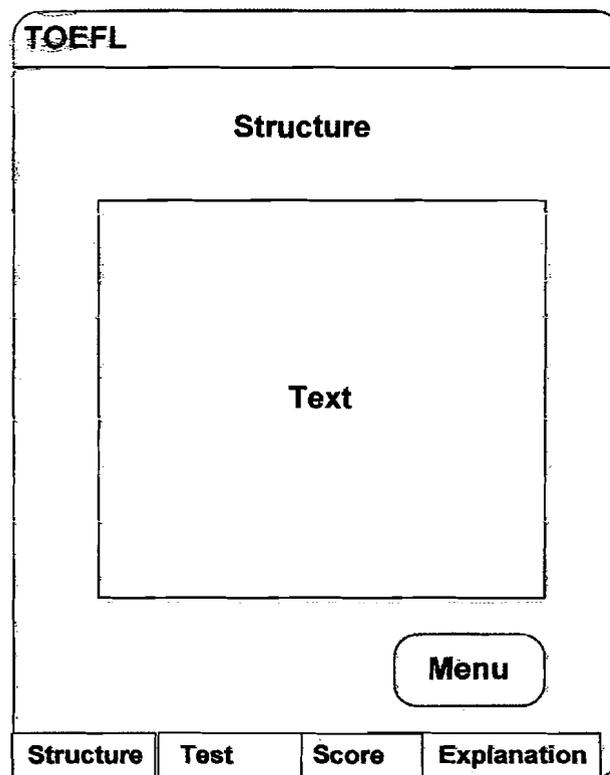


Gambar 4.7 Rancangan halaman Introduction

d. Halaman Structure

Halaman Structure merupakan halaman yang menampilkan jenis test TOEFL yang berupa structure. Pada halaman pertama menampilkan informasi tentang test structure, panduan tentang jenis soal structure yang ditampilkan oleh aplikasi. Halaman kedua *user* akan masuk pada test Structure yang menampilkan soal-soal structure, halaman ketiga menampilkan score yang dicapai *user* setelah melakukan test, halaman keempat menampilkan explanation/pembahasan soal

structure, dan terdapat tombol untuk kembali ke halaman Menu. Rancangan dari halaman Structure ada pada gambar 4.8.

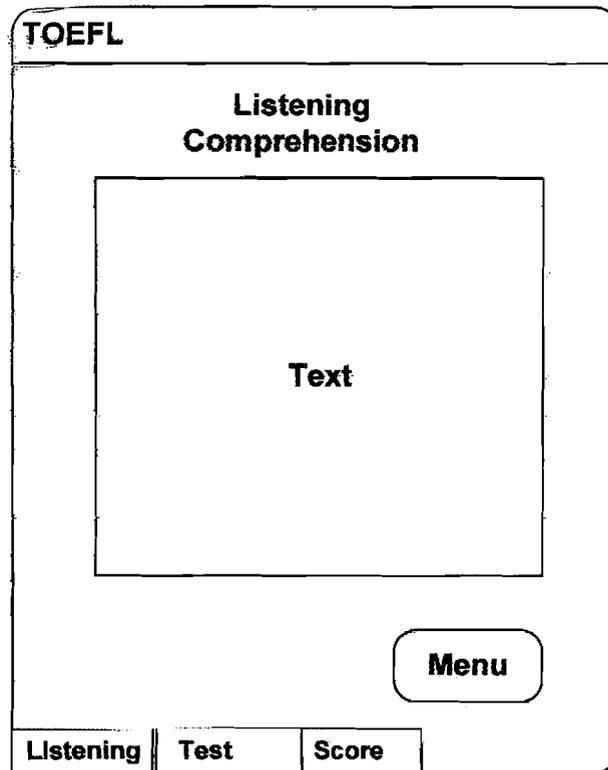


Gambar 4.8 Rancangan halaman Structure

e. Halaman Listening Comprehension

Halaman Listening merupakan halaman yang menampilkan jenis test TOEFL yang berupa Listening. Pada halaman pertama menampilkan informasi tentang test Listening, panduan tentang soal Listening yang ditampilkan oleh aplikasi. Halaman kedua *user* akan masuk pada test Listening yang menampilkan soal-soal Listening, halaman ketiga menampilkan score yang dicapai *user* setelah

melakukan test, dan terdapat tombol untuk kembali ke halaman Menu. Rancangan dari halaman Listening ada pada gambar 4.9.

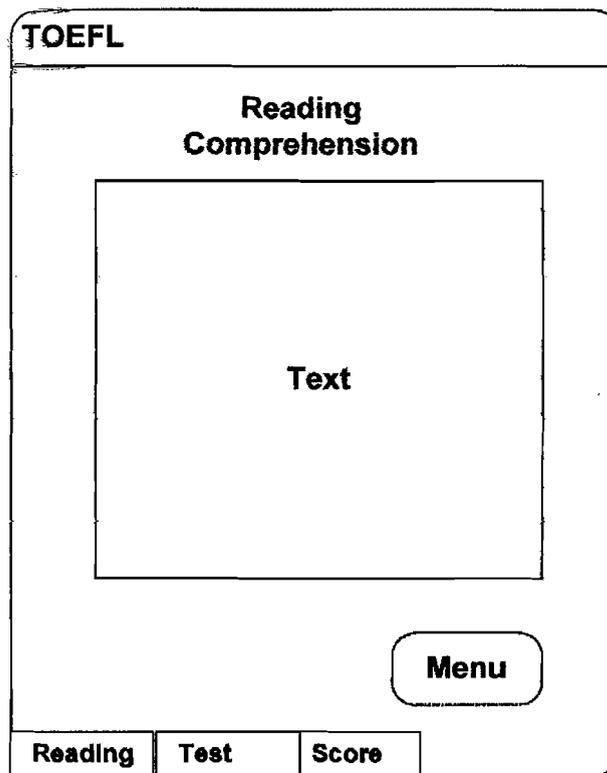


Gambar 4.9 Rancangan halaman Listening

f. Halaman Reading Comprehension

Halaman Reading merupakan halaman yang menampilkan jenis test TOEFL yang berupa Reading. Pada halaman pertama menampilkan informasi tentang test Reading, panduan tentang soal Reading yang ditampilkan oleh aplikasi. Halaman kedua *user* akan masuk pada test Reading yang menampilkan soal-soal Reading, halaman ketiga menampilkan score yang dicapai *user* setelah

melakukan test, dan terdapat tombol untuk kembali ke halaman Menu. Rancangan dari halaman Reading ada pada gambar 4.10.

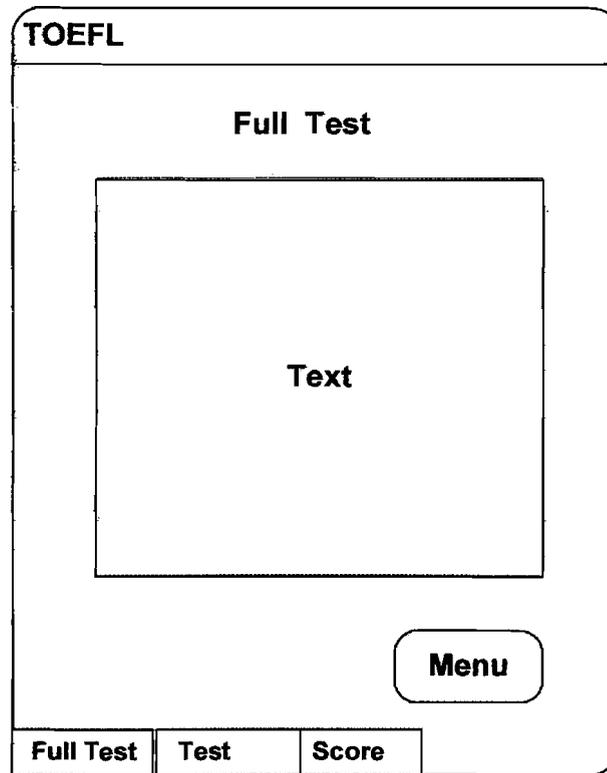


Gambar 4.10 Rancangan halaman Reading

g. Halaman Full Test

Halaman Full Test merupakan halaman yang menampilkan jenis test TOEFL secara keseluruhan, meliputi Structure, Listening Comprehension, dan Reading Comprehension. Pada halaman pertama menampilkan informasi tentang test, panduan tentang soal Full Test yang ditampilkan oleh aplikasi. Halaman kedua *user* akan masuk pada test Full Test yang menampilkan soal-soal Full Test,

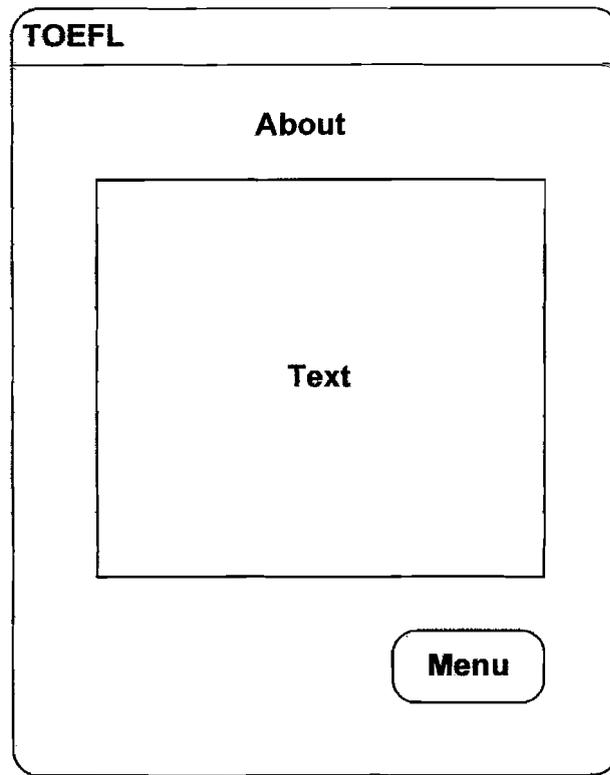
halaman ketiga menampilkan score yang dicapai *user* setelah melakukan test, dan terdapat tombol untuk kembali ke halaman Menu. Rancangan dari halaman Full Test ada pada gambar 4.11.



Gambar 4.11 Rancangan halaman Full Test

h. Halaman About

Halaman About merupakan halaman yang menampilkan informasi tentang versi perangkat lunak, pembuat perangkat lunak. Rancangan dari halaman About ada pada gambar 4.12.



Gambar 4.12 Rancangan halaman About

BAB V

IMPLEMENTASI PERANGKAT LUNAK

5.1 Batasan Implementasi

Pada bagian implementasi ini perangkat lunak yang sudah dibangun sebelumnya akan di uji cobakan untuk melihat apakah pembangunan perangkat lunak sudah sesuai dengan fungsi dan tujuan dari di kembangkannya perangkat lunak tersebut.

Pada kenyataannya program *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* ini memiliki beberapa keterbatasan, yaitu:

1. Pengguna hanya dapat memilih menu yang sudah ada dan data-data test TOEFL telah disediakan oleh aplikasi
2. Pengguna tidak dapat melakukan penambahan ataupun pengurangan pada isi program *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET*

Pada pembuatan program *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* soal-soal diambil dari *BARRON'S : How To Prepare For The TOEFL* dan *BARRON'S: TOEFL iBT*. Kedua buku tersebut adalah buku TOEFL yang banyak direferensikan sebagai buku acuan latihan TOEFL.

5.1.1 Perangkat Keras

Perangkat keras yang digunakan pada saat implementasi program adalah:

1. *Pocket PC* HP IPAQ HW6500
2. RAM 128 MB
3. Memori 32 MB

5.1.2 Perangkat Lunak

Perangkat lunak yang digunakan pada saat implementasi *Aplikasi TOEFL*

Untuk Pocket PC Dengan Teknologi .NET adalah:

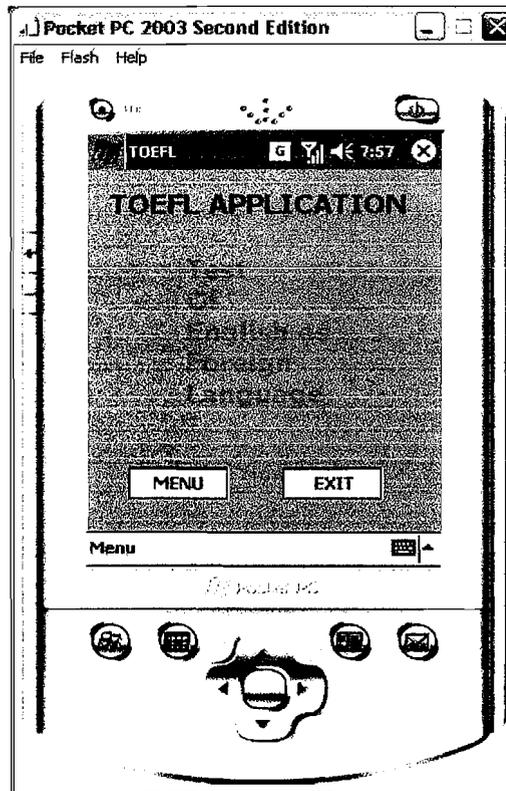
1. *Windows Mobile* 2003
2. *Active Sync* 4.0
3. *SQL Server CE*

5.2 Implementasi Sistem

Hasil dari implementasi dan gambaran dari antarmuka dari *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* adalah sebagai berikut:

a. Halaman Utama

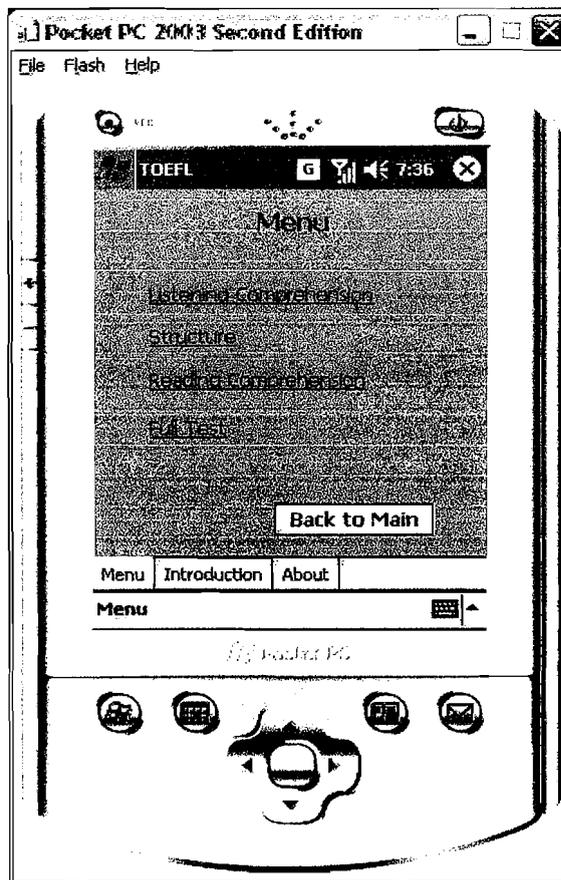
Halaman utama ini merupakan halaman pembuka pada *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET*, gambar untuk halaman pembuka ada pada gambar 5.1.



Gambar 5.1 Halaman Utama

b. Halaman Menu

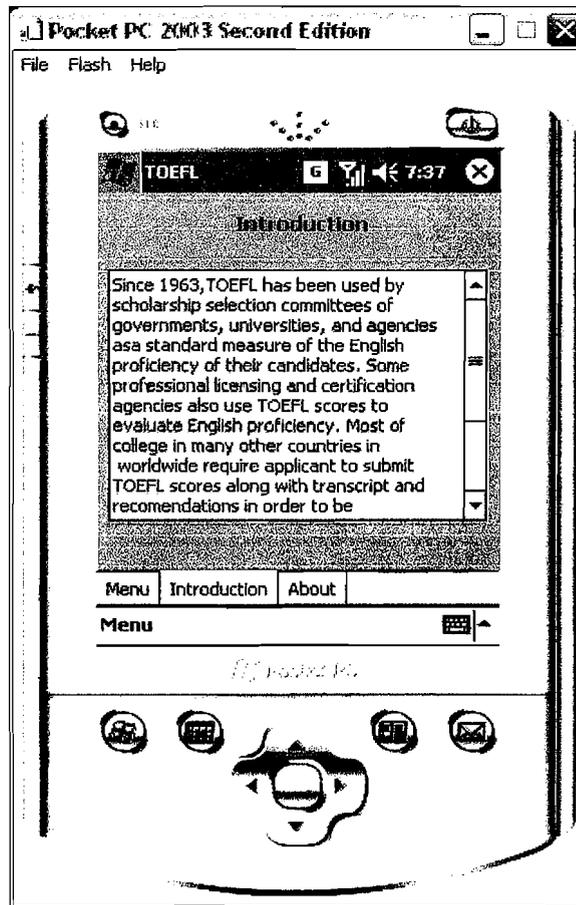
Halaman menu ini digunakan oleh pengguna untuk memilih jenis test apa yang di kehendaki oleh pengguna sehingga test yang tampil sesuai dengan keinginan pengguna, gambar untuk halaman menu ada pada gambar 5.2.



Gambar 5.2 Halaman Menu

c. Halaman Introduction

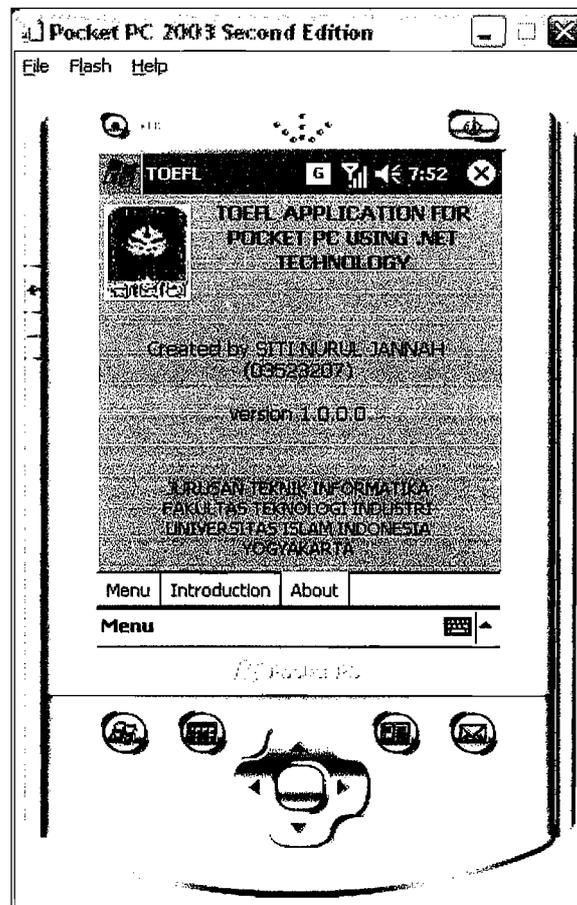
Halaman Introduction menampilkan informasi seputar TOEFL, meliputi sejarah TOEFL beserta kegunaannya, gambar untuk halaman Introduction ada pada gambar 5.3.



Gambar 5.3 Halaman Introduction

d. Halaman About

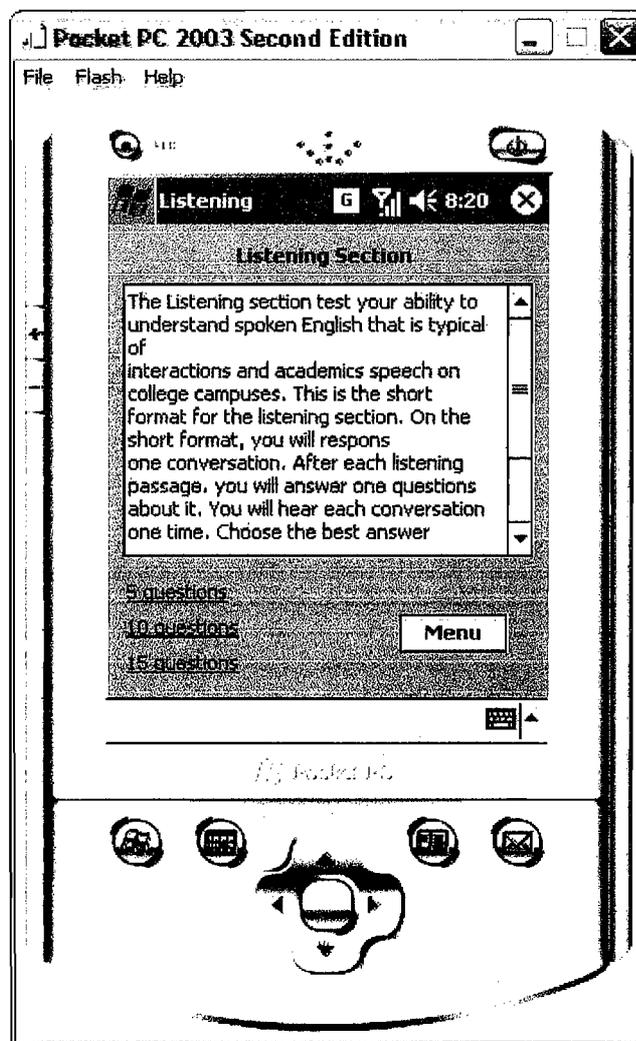
Halaman About berfungsi untuk menampilkan informasi tentang program, meliputi pembuat program dan versi, gambar untuk halaman About ada pada gambar 5.4.



Gambar 5.4 Halaman About

e. Halaman Listening Utama

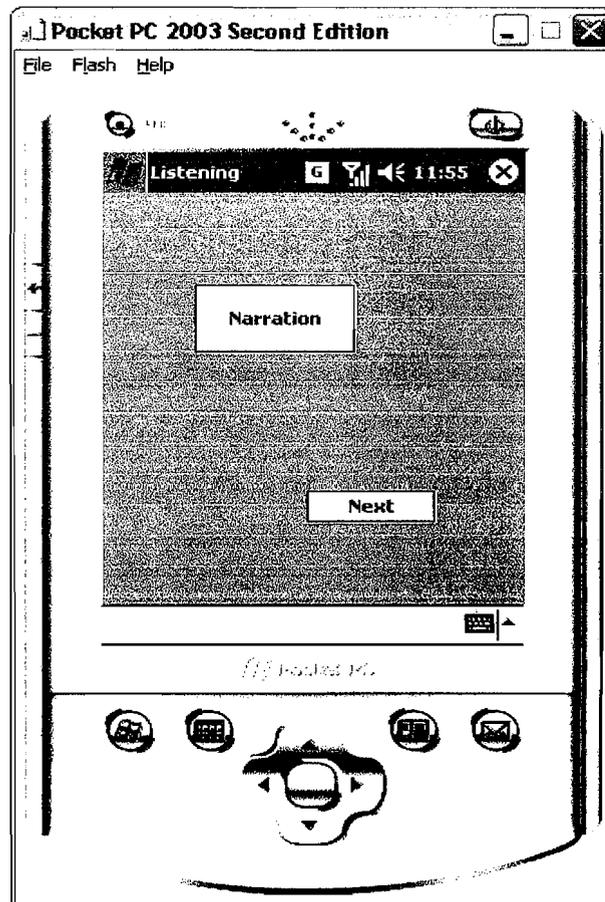
Halaman Listening menampilkan test listening yang akan diakses oleh *user*. Halaman ini akan tampil setelah *user* memilih Listening Comprehension pada halaman Menu, halaman pertama dari Listening Comprehension ini adalah petunjuk pengerjaan test Listening, pada halaman pertama ini *user* dapat memilih 3 jenis jumlah soal yang akan dikerjakan, menyesuaikan dengan kebutuhan dan waktu. Gambar untuk halaman Listening ada pada gambar 5.3.



Gambar 5.5 Halaman Listening

f. Halaman Listening Narration

Pada halaman soal Listening Narration akan menampilkan halaman suara yang akan didengarkan oleh *user*. Halaman suara ini akan tampil setelah *user* memilih jumlah soal yang dikehendaki. *User* mendengarkan soal berupa percakapan setelah menekan button narration. Gambar untuk halaman soal Listening Narration ada pada gambar 5.6



Gambar 5.6 Halaman Listening Sound

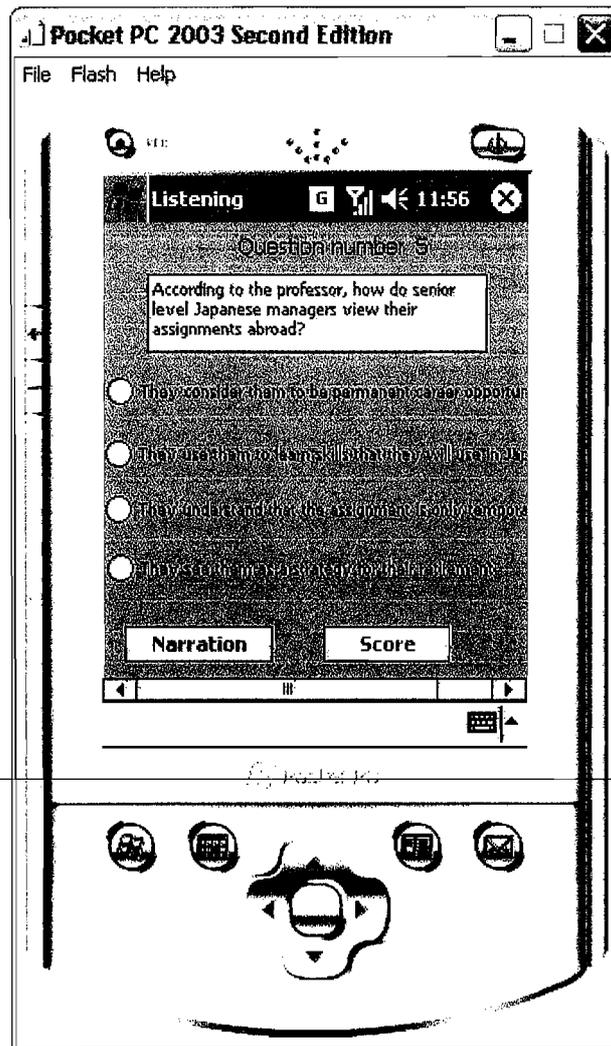
Berikut adalah *source code* untuk menampilkan isi dari halaman Listening berupa suara:

```
'mengambil suara yang sesuai dengan nomor_soal
Dim str As String
str = "Program Files/toefl/soundwav/" +
nomor_soal.ToString + ".wav"
Dim sound As New Sound(str)

sound.Play()
```

g. Halaman Listening Soal

Pada halaman soal Listening Soal akan menampilkan halaman soal beserta pilihan jawaban berdasarkan percakapan suara yang telah didengarkan oleh *user*. Gambar untuk halaman soal Listening Soal ada pada gambar 5.7



Gambar 5.7 Halaman Listening Soal

Berikut adalah *source code* untuk menampilkan isi dari halaman Listening berupa

soal:

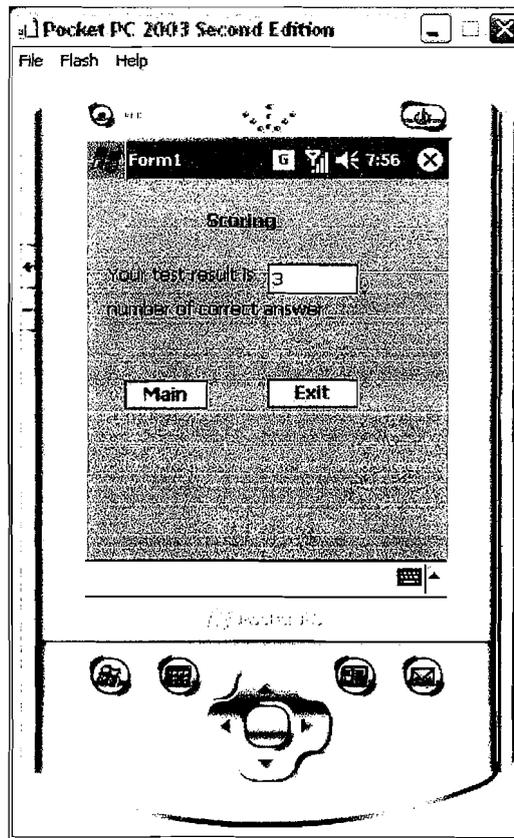
```

Try
' Mengambil data untuk menampilkan ID,soal,pilihan jawaban, dan
jawaban
    cmd = New SqlCeCommand("SELECT
ID,soal,rbA,rbB,rbC,rbD,jawaban FROM soal2 where ID = '" &
nomor_soal & "'", conn)
    conn.Open()
    dtr = cmd.ExecuteReader
' menampilkan data ke dalam Textbox dan radiobutton
    dtr.Read()
    objdataadapter = New SqlCeDataAdapter(cmd)
    objdataadapter.Fill(objdataset, "soal2")
    objdatatable = objdataset.Tables("soal2")
    Dim position As Integer =
BindingContext(objdatatable).Position
    With objdatatable.Rows(position)
        TextBox2.Text = dtr("soal")
        rbA.Text = dtr("rbA")
        rbB.Text = dtr("rbB")
        rbC.Text = dtr("rbC")
        rbD.Text = dtr("rbD")
        jawaban = dtr("jawaban")
    End With
    dtr.Close()
    conn.Close()
Catch ex As Exception
    MsgBox(ex.Message, MsgBoxStyle.Information, "TOEFL")
Exit Sub
End Try

```

h. Halaman Listening Score

Pada halaman soal Listening Score akan menampilkan jumlah score jawaban benar yang telah dikerjakan oleh *user*. Gambar untuk halaman soal Listening Score ada pada gambar 5.8



Gambar 5.8 Halaman Listening Score

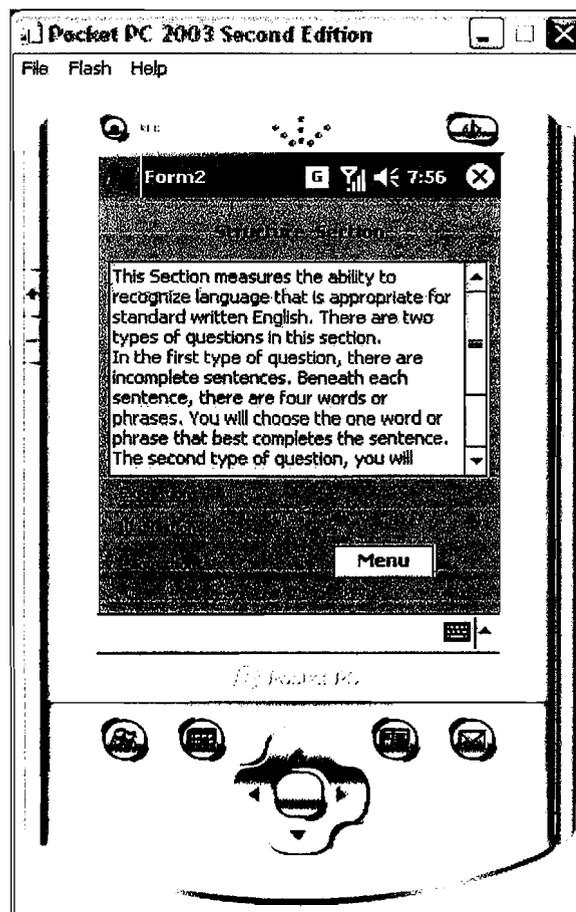
Berikut adalah *source code* untuk menampilkan isi dari halaman Listening berupa score:

```
'menampilkan score dari test yang telah dilakukan user
panelscore.Show()
If (jawaban = "A") And rbA.Checked Then benar = benar + 1
If (jawaban = "B") And rbB.Checked Then benar = benar + 1
If (jawaban = "C") And rbC.Checked Then benar = benar + 1
If (jawaban = "D") And rbD.Checked Then benar = benar + 1

TextBox3.Text = benar.ToString
```

i. Halaman Structure Utama

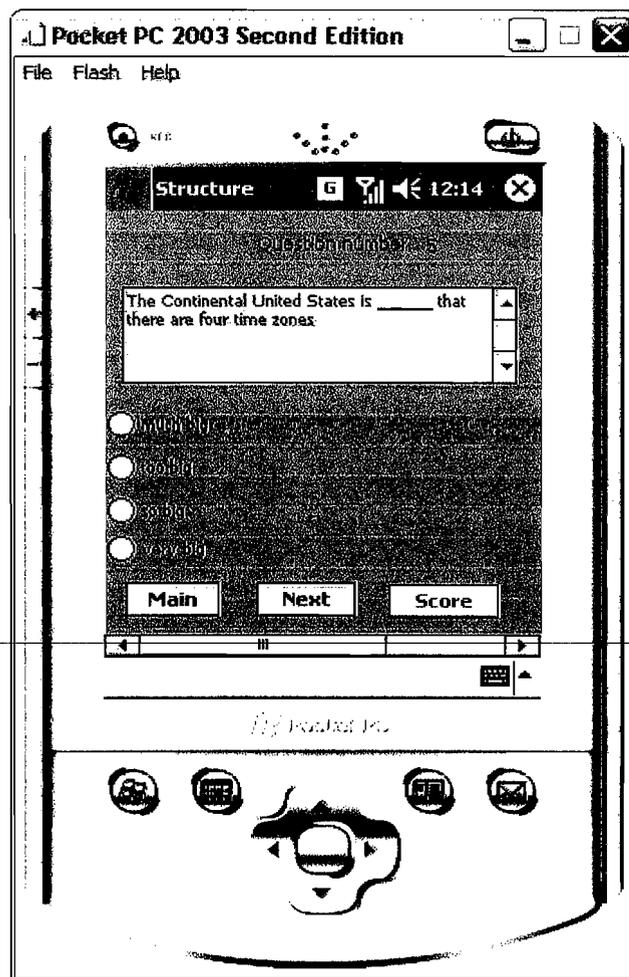
Halaman Structure menampilkan test Structure yang akan diakses oleh *user*. Halaman ini akan tampil setelah *user* memilih Structure pada halaman Menu, halaman pertama dari Structure ini adalah petunjuk pengerjaan test Structure, pada halaman pertama ini *user* dapat memilih 3 jenis jumlah soal yang akan dikerjakan, menyesuaikan dengan kebutuhan dan waktu. Gambar untuk halaman Structure ada pada gambar 5.9.



Gambar 5.9 Halaman Structure Utama

j. Halaman Structure Soal

Pada halaman soal Structure Soal akan menampilkan halaman soal beserta pilihan jawaban yang dapat dikerjakan oleh *user* dengan cara memilih jawaban yang paling benar. Soal akan ditampilkan sesuai dengan pilihan jumlah pertanyaan yang dikehendaki *user* pada halaman Structure Utama. Gambar untuk halaman soal Structure Soal ada pada gambar 5.10.



Gambar 5.10. Halaman Structure Soal

Berikut adalah *source code* untuk menampilkan isi dari halaman Structure berupa

soal:

```

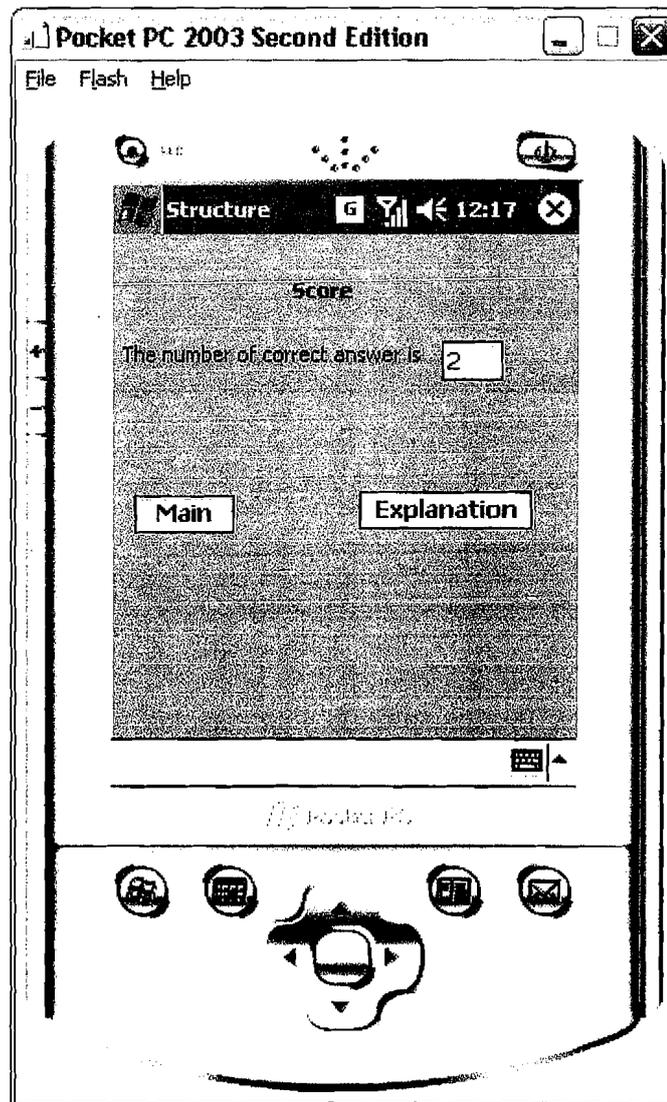
Try
' Mengambil data untuk menampilkan ID,soal,pilihan jawaban, dan
jawaban

        cmd = New SqlCeCommand("SELECT
ID,soal,rbA,rbB,rbC,rbD,jawaban,pembahasan FROM soall where ID =
'" & nomor_soal.ToString & "'", conn)
        conn.Open()
        dtr = cmd.ExecuteReader
' menampilkan data ke dalam Textbox dan radiobutton
        dtr.Read()
        objdataadapter = New SqlCeDataAdapter(cmd)
        objdataadapter.Fill(objdataset, "soall")
        objdatatable = objdataset.Tables("soall")
        Dim position As Integer = BindingContext(objdatatable).Position
        With objdatatable.Rows(position)
            TextBox1.Text = dtr("soal")
            rbA.Text = dtr("rbA")
            rbB.Text = dtr("rbB")
            rbC.Text = dtr("rbC")
            rbD.Text = dtr("rbD")
            jawaban = dtr("jawaban")
'menampilkan pembahasan dari soal yang telah dikerjakan user
        TextBox4.Text = TextBox4.Text + vbCrLf + vbCrLf +dtr("pembahasan")
            dtr.Close()
            conn.Close()
        End With
        Catch ex As Exception
            MsgBox(ex.Message, MsgBoxStyle.Information, "TOEFL")
        Exit Sub
    End Try

```

k. Halaman Structure Score

Pada halaman soal Structure Score akan menampilkan jumlah score jawaban benar yang telah dikerjakan oleh *user*. Gambar untuk halaman soal Structure Score ada pada gambar 5.11.



Gambar 5.11 Halaman Structure Score

Berikut adalah *source code* untuk menampilkan isi dari halaman Structure berupa

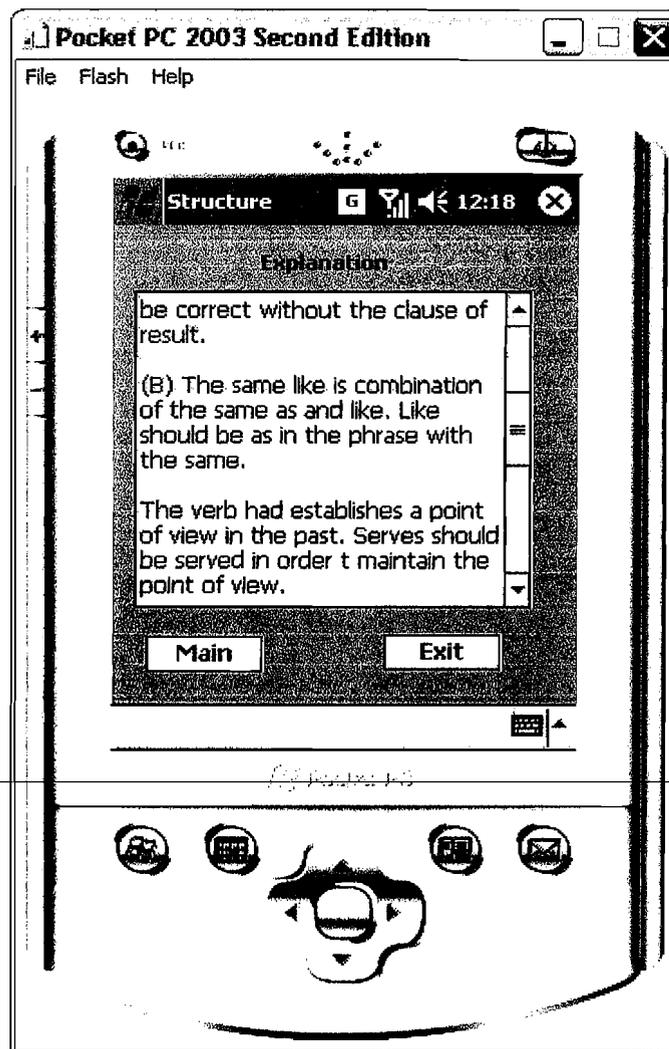
score:

```
'menampilkan score dari test yang telah dilakukan user
  If (jawaban = "A") And rbA.Checked Then benar = benar + 1
  If (jawaban = "B") And rbB.Checked Then benar = benar + 1
  If (jawaban = "C") And rbC.Checked Then benar = benar + 1
  If (jawaban = "D") And rbD.Checked Then benar = benar + 1

  TextBox3.Text = benar.ToString
```

1. Halaman Structure Explanation

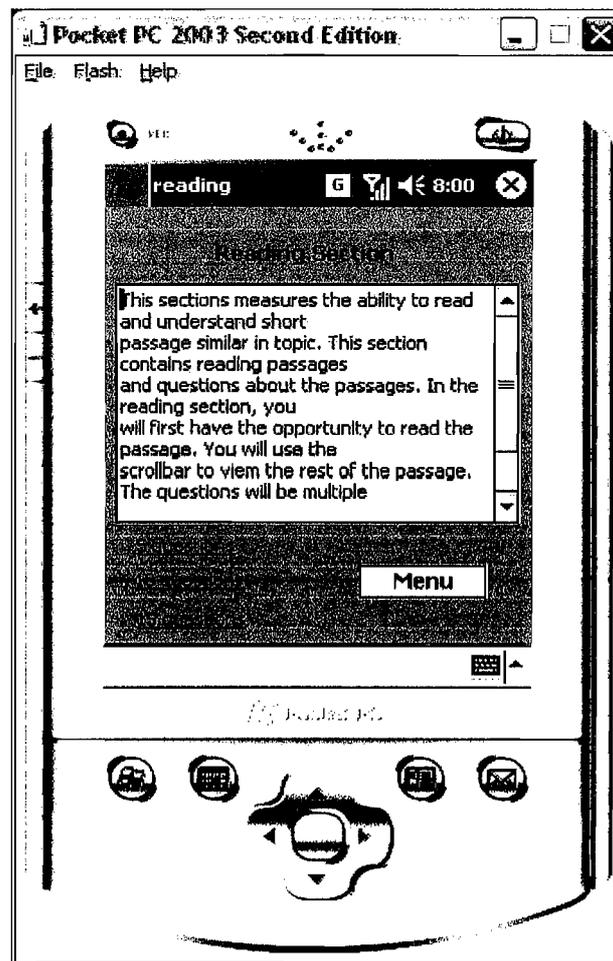
Pada halaman soal Structure Explanation akan menampilkan pembahasan soal yang telah dikerjakan *user*. Gambar untuk halaman soal Structure Explanation ada pada gambar 5.12.



Gambar 5.12 Halaman Structure Explanation

m. Halaman Reading Utama

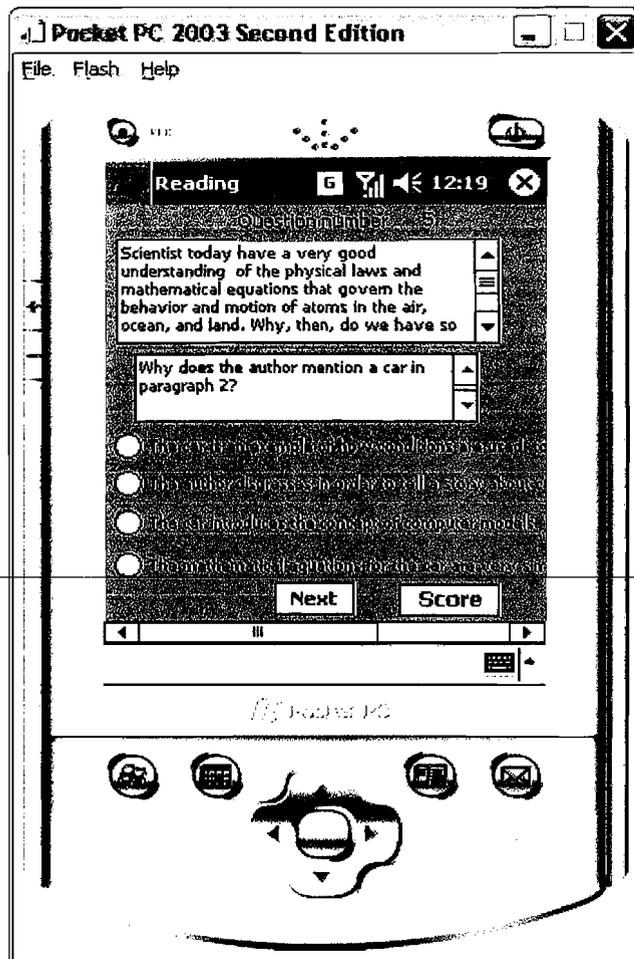
Halaman Reading Utama menampilkan test Reading yang akan diakses oleh *user*. Halaman ini akan tampil setelah *user* memilih Reading pada halaman Menu, halaman pertama dari Reading ini adalah petunjuk pengerjaan test Reading, pada halaman pertama ini *user* dapat memilih 3 jenis jumlah soal yang akan dikerjakan, menyesuaikan dengan kebutuhan dan waktu. Gambar untuk halaman Reading Utama ada pada gambar 5.13.



Gambar 5.13. Halaman Reading Utama

n. Halaman Reading Soal

Pada halaman soal Reading Soal akan menampilkan halaman soal beserta pilihan jawaban yang dapat dikerjakan oleh *user* dengan cara memilih jawaban yang paling benar. *User* akan membaca text reading terlebih dahulu untuk dapat menjawab soal yang berkaitan dengan text tersebut. Soal akan ditampilkan sesuai dengan pilihan jumlah pertanyaan yang dikehendaki *user* pada halaman Reading Utama. Gambar untuk halaman soal Reading Soal ada pada gambar 5.14.



Gambar 5.14 Halaman Reading Soal

Berikut adalah *source code* untuk menampilkan isi dari halaman Reading berupa

soal:

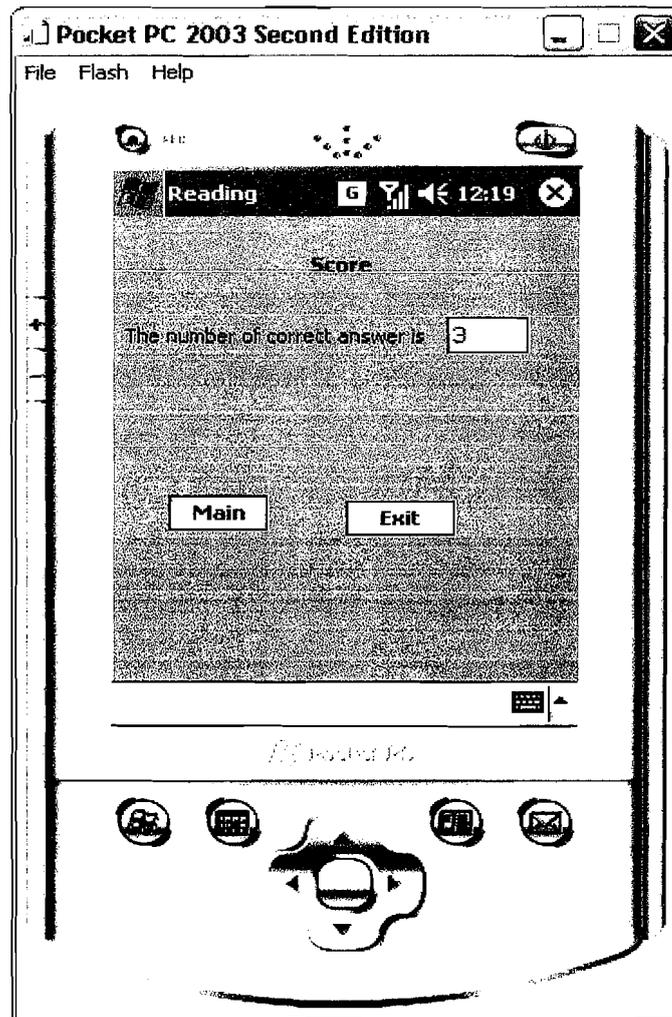
```

Try
' Mengambil data untuk menampilkan ID,soal,pilihan jawaban, dan
jawaban
        cmd = New SqlCommand("SELECT
ID,text,soal,rbA,rbB,rbC,rbD,jawaban FROM soal3 where ID = '" &
Label4.Text & "'", conn)
        conn.Open()
        dtr = cmd.ExecuteReader
' menampilkan data ke dalam Textbox dan radiobutton
        dtr.Read()
        objdataadapter = New SqlDataAdapter(cmd)
        objdataadapter.Fill(objdataset, "soal3")
        objdatatable = objdataset.Tables("soal3")
        Dim position As Integer = BindingContext(objdatatable).Position
        With objdatatable.Rows(position)
            TextBox1.Text = dtr("text")
            TextBox3.Text = dtr("soal")
            rbA.Text = dtr("rbA")
            rbB.Text = dtr("rbB")
            rbC.Text = dtr("rbC")
            rbD.Text = dtr("rbD")
            jawaban = dtr("jawaban")
        End With
        dtr.Close()
        conn.Close()
    Catch ex As Exception
        MsgBox(ex.Message, MsgBoxStyle.Information, "TOEFL")
    Exit Sub
End Try

```

o. Halaman Reading Score

Pada halaman soal Reading Score akan menampilkan jumlah score jawaban benar yang telah dikerjakan oleh *user*. Gambar untuk halaman soal Reading Score ada pada gambar 5.15.



Gambar 5.15 Halaman Reading Score

Berikut adalah *source code* untuk menampilkan isi dari halaman Reading berupa

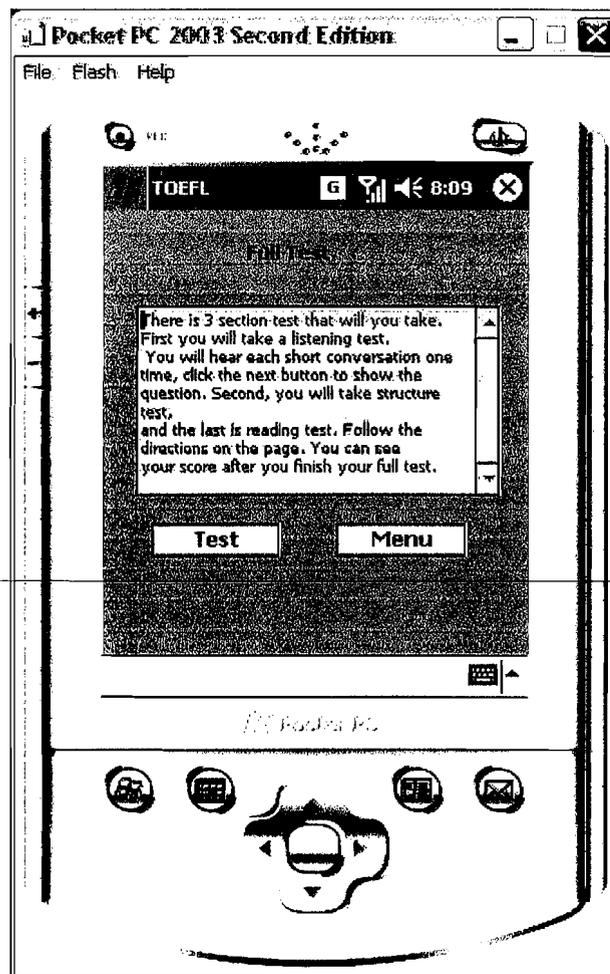
score:

```
'menampilkan score dari test yang telah dilakukan user
If (jawaban = "A") And rbA.Checked Then benar = benar + 1
    If (jawaban = "B") And rbB.Checked Then benar = benar + 1
    If (jawaban = "C") And rbC.Checked Then benar = benar + 1
    If (jawaban = "D") And rbD.Checked Then benar = benar + 1

TextBox4.Text = benar.ToString
```

p. Halaman Full Test Utama

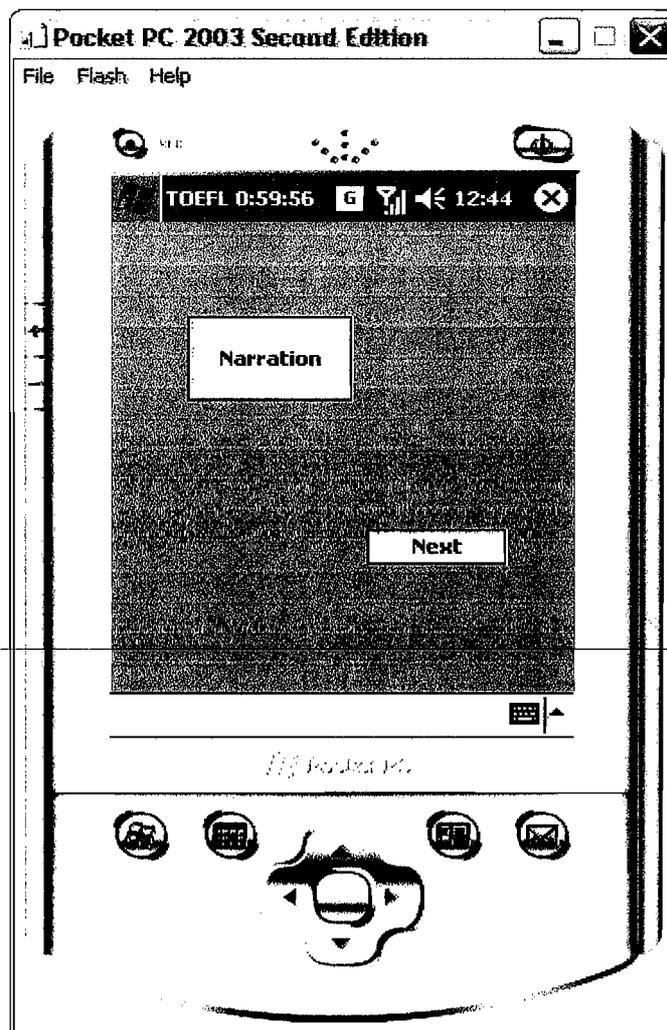
Halaman Full test ini menampilkan petunjuk pengerjaan TOEFL secara keseluruhan. Pada Full Test ini akan ditampilkan 3 section test sekaligus secara bertahap pengerjaannya. Test yang pertama Listening, test kedua Structure, dan yang terakhir Reading. Waktu yang disediakan adalah 1 jam untuk mengerjakan ketiga soal tersebut. *User* dapat melihat jumlah score TOEFL yang diperoleh diakhir pengerjaan. Gambar untuk halaman soal Full Test Utama ada pada gambar 5.16.



Gambar 5.16 Halaman Full Test Utama

q. Halaman Full Test Narration

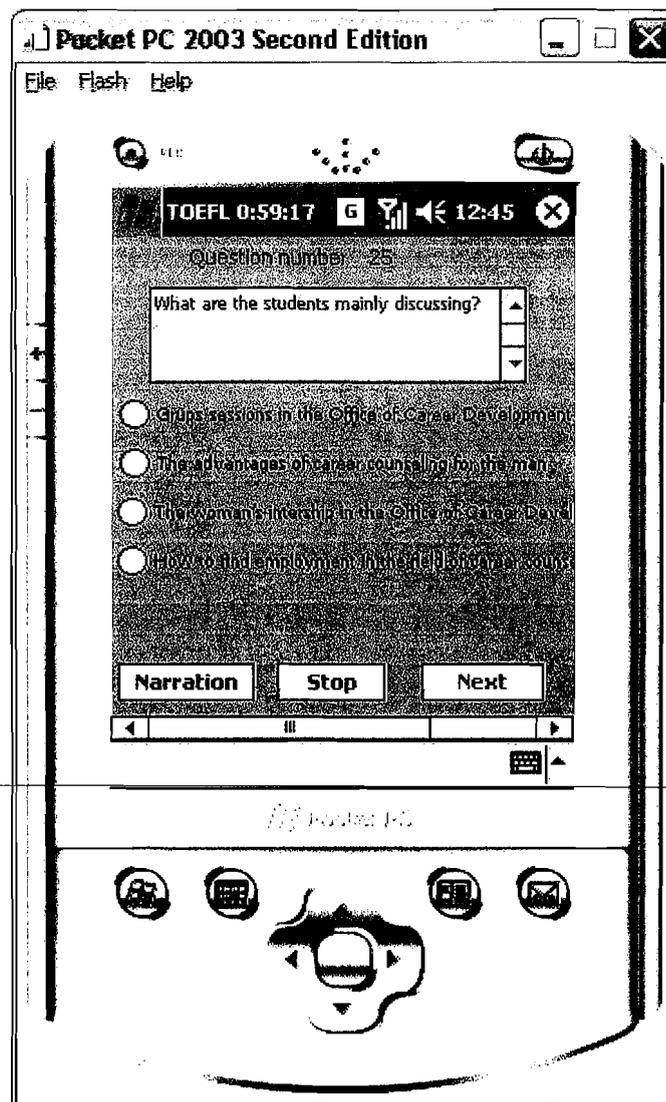
Pada halaman Full Test Narration akan menampilkan halaman suara yang akan didengarkan oleh *user*. Halaman suara ini akan tampil setelah *user* memilih button test pada halaman Full Test Utama. *User* mendengarkan soal berupa percakapan setelah menekan button Narration. Gambar untuk halaman soal Full Test Sound ada pada gambar 5.17



Gambar 5.17 Halaman Full Test Narration

r. Halaman Full Test Soal Listening

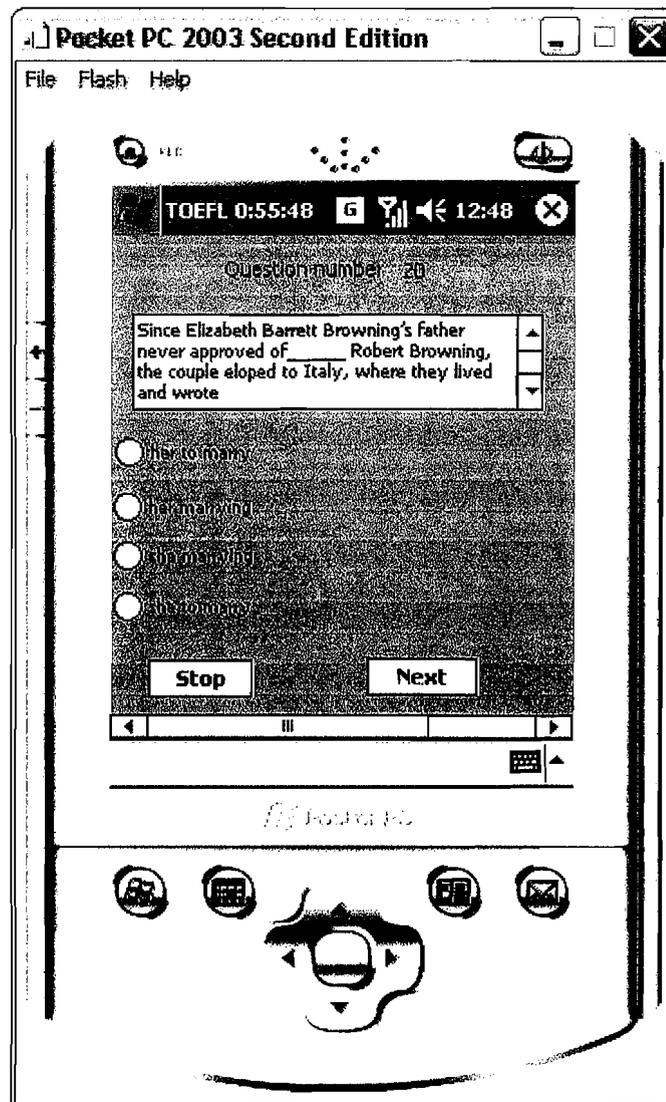
Pada halaman Full Test Soal Listening akan menampilkan soal beserta pilihan jawaban berdasarkan percakapan yang telah didengarkan *user*. Gambar Full Test Soal Listening ada pada gambar 5.18.



Gambar 5.18 Halaman Full Test Soal Listening

s. Halaman Full Test Soal Structure

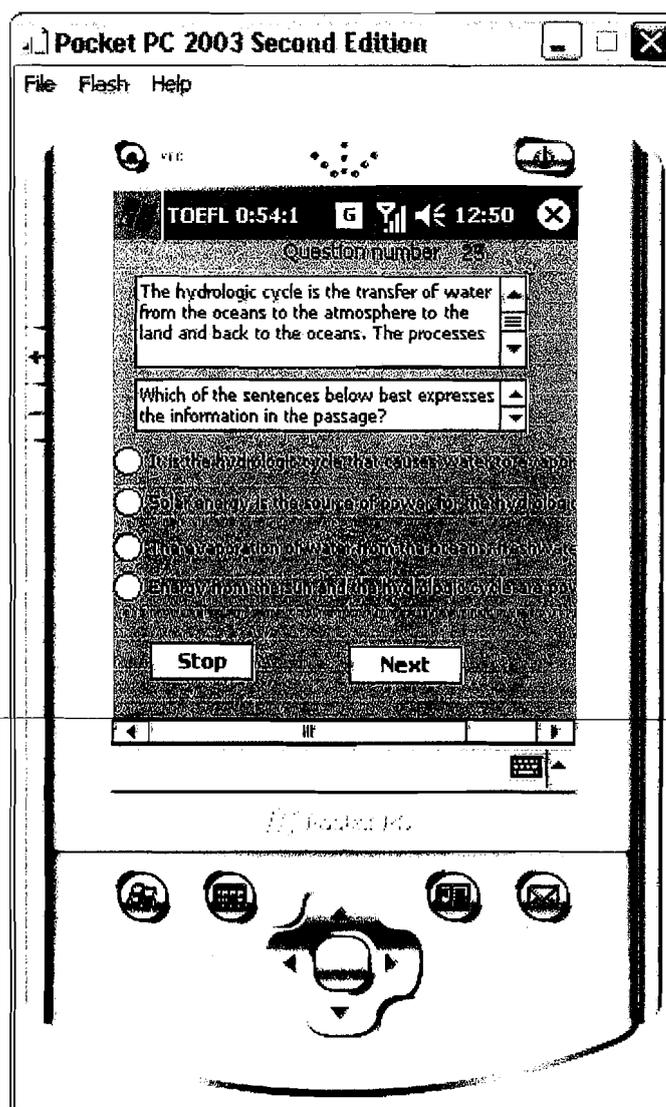
Pada halaman Full Test Soal Structure akan menampilkan soal-soal structure beserta pilihan jawaban. *User* dapat memilih salah satu jawaban yang benar. Gambar Full Test Soal Structure ada pada gambar 5.19.



Gambar 5.19 Halaman Full Test Soal Structure

t. Halaman Full Test Soal Reading

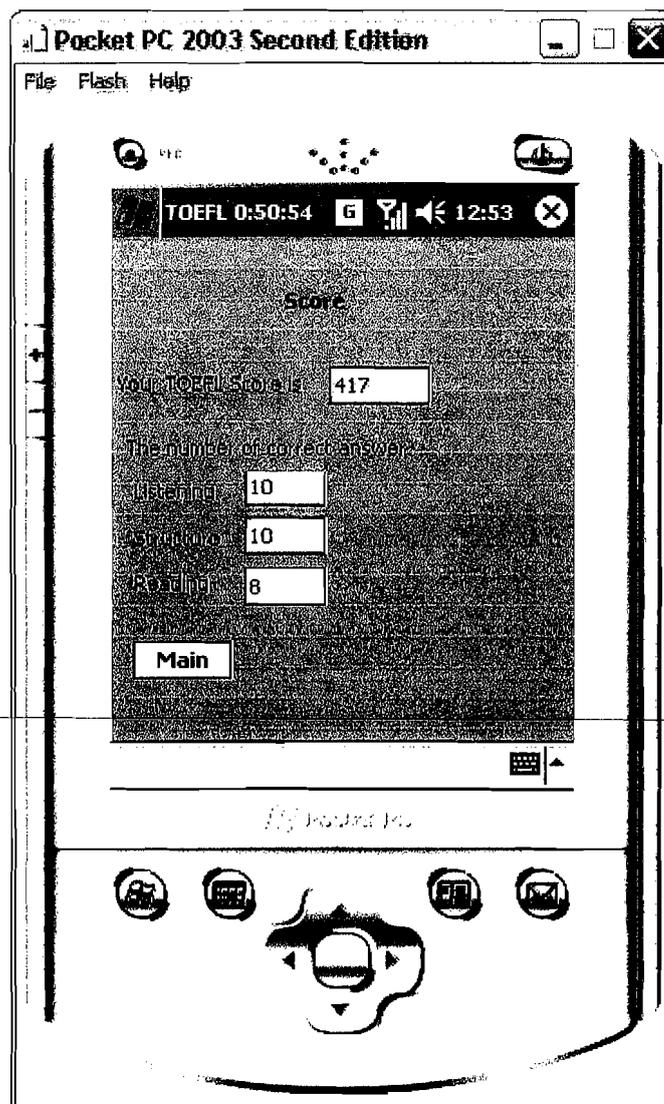
Pada halaman Full Test Soal Reading akan menampilkan soal-soal structure beserta pilihan jawaban. *User* akan membaca text Reading terlebih dahulu sebelum dapat menjawab pertanyaan yang berdasarkan text tersebut. Gambar Full Test Soal Reading ada pada gambar 5.20.



Gambar 5.20 Halaman Full Test Soal Reading

u. Halaman Full Test Score

Halaman Full Test Score menampilkan jumlah score TOEFL yang dicapai *user* setelah mengerjakan secara keseluruhan ketiga section pada Full Test. Juga menampilkan jumlah jawaban benar pada masing-masing section. Gambar Full Test Score ada pada gambar 5.21.



Gambar 5.21 Halaman Full Test Score

Berikut adalah *source code* untuk menampilkan isi dari halaman Full Test

berupa perhitungan score:

```

Private Function hitung_nilai() As Integer
    'perhitungan score TOEFL berdasarkan test yang dilakukan
    user
        Dim nilail As Integer
        Dim nilais As Integer
        Dim nilair As Integer
        Dim jmlnilai As Integer
        Dim total As Integer
        'mengambil nilai konversi untuk test Listening
        Try
            cmd = New SqlCommand("SELECT converted from nilail
where benar = '" & benarL & "'", conn)
            conn.Open()
            dtr = cmd.ExecuteReader
            dtr.Read()
            objdataadapter = New SqlDataAdapter(cmd)
            objdataadapter.Fill(objdataset, "nilail")
            objdatatable = objdataset.Tables("nilail")
            Dim position As Integer =
BindingContext(objdatatable).Position
            With objdatatable.Rows(position)
                nilail = dtr("converted")
                dtr.Close()
                conn.Close()
            End With
        Catch ex As Exception
            MsgBox(ex.Message, MsgBoxStyle.Information, "TOEFL")
            Exit Function
        End Try
        'mengambil nilai konversi untuk test Structure
        Try
            cmd = New SqlCommand("SELECT converted from nilais
where benar = '" & benarS & "'", conn)
            conn.Open()
            dtr = cmd.ExecuteReader
            dtr.Read()
            objdataadapter = New SqlDataAdapter(cmd)
            objdataadapter.Fill(objdataset, "nilais")
            objdatatable = objdataset.Tables("nilais")
            Dim position As Integer =
BindingContext(objdatatable).Position
            With objdatatable.Rows(position)
                nilais = dtr("converted")
                dtr.Close()
                conn.Close()
            End With
        Catch ex As Exception
            MsgBox(ex.Message, MsgBoxStyle.Information, "TOEFL")
            Exit Function
    
```

```

End Try
'mengambil nilai konversi untuk test Reading
Try
    cmd = New SqlCommand("SELECT converted from nilair
where benar = '" & benarR & "'", conn)
    conn.Open()
    dtr = cmd.ExecuteReader
    dtr.Read()
    objdataadapter = New SqlDataAdapter(cmd)
    objdataadapter.Fill(objdataset, "nilair")
    objdatatable = objdataset.Tables("nilair")
    Dim position As Integer =
BindingContext(objdatatable).Position
    With objdatatable.Rows(position)
        nilair = dtr("converted")
    End With
    dtr.Close()
    conn.Close()
End With
Catch ex As Exception
    MsgBox(ex.Message, MsgBoxStyle.Information, "TOEFL")
Exit Function
End Try
'menghitung score keseluruhan
jmlnilai = nilail + nilais + nilair
total = (jmlnilai / 3) * 10
hitung_nilai = total
End Function

```

v. Pengacakan Soal

Pada aplikasi *TOEFL untuk Pocket PC dengan Teknologi .Net* ini penampilan soal dari keseluruhan jenis test menggunakan metode pengacakan soal. Pengacakan bertujuan untuk menghindari soal muncul sama ketika *user* membuka aplikasi kembali. Tetapi dapat pula dalam satu test soal sama muncul lebih dari dua kali, hal ini disebabkan keterbatasan database soal. Berikut ini *source code* untuk pengacakan soal:

```

'mengambil soal secara acak dengan cara melihat data dari record
terakhir
Dim i As Integer
Randomize()
i = 0
While i = 0
    i = Int(Rnd() * getRecCount(soal))
    nomor_soal = i
End While

```

w. Timer

Pada aplikasi *TOEFL untuk Pocket PC dengan Teknologi .Net* untuk jenis test Full Test akan menampilkan waktu untuk pengerjaan testnya. Waktu yang disediakan adalah 1 jam untuk mengerjakan ketiga section test didalam Full Test.

Berikut ini *source code* untuk Timer:

```
' menampilkan waktu untuk pengerjaan Test
  Dim waktu As TimeSpan
  waktu = selesai - Now
  Me.Text = "TOEFL " + waktu.Hours.ToString + ":" +
waktu.Minutes.ToString + ":" + waktu.Seconds.ToString
```

BAB VI

ANALISIS KINERJA PERANGKAT LUNAK

6.1 Pengujian Program

Pengujian program perangkat lunak *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* bertujuan untuk mengetahui apakah perangkat lunak tersebut sudah dapat berjalan sesuai dengan fungsi-fungsi yang diharapkan dan apakah sudah sesuai dengan tujuan dari pembuatan perangkat lunak tersebut.

6.2 Pengujian Dan Analisis

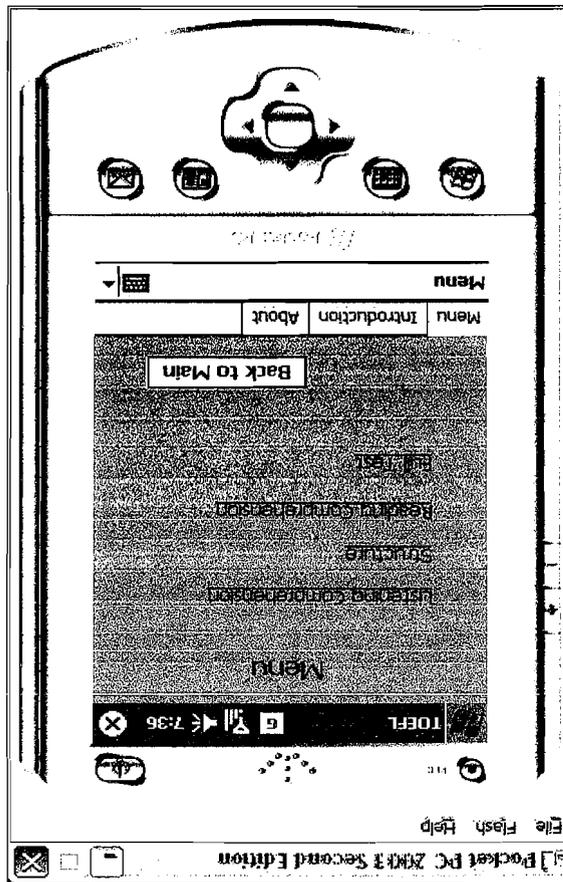
Pada tahap pengujian ini akan dilihat apakah perangkat lunak sudah dapat berjalan sesuai dengan fungsi dan kebutuhan.

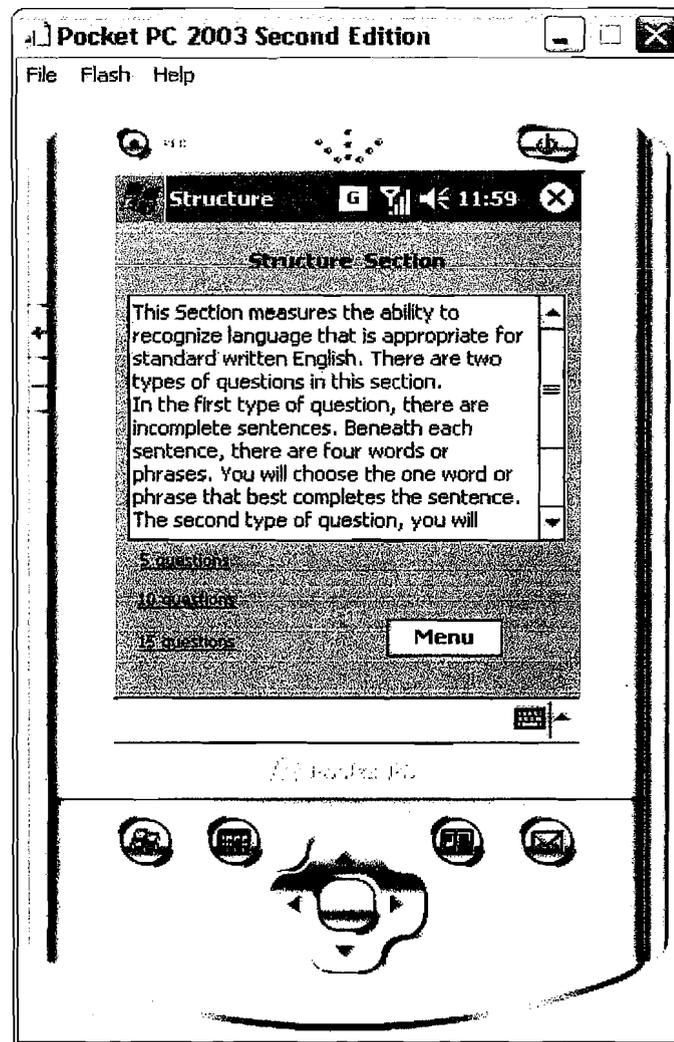
6.2.1 Pengujian Normal

Pada bagian ini perangkat lunak akan diujikan dengan masukan-masukan yang benar sesuai dengan prosedur-prosedur yang sudah ditetapkan.

Pada bagian ini akan diujikan apabila pengguna memilih menu yang telah disediakan, gambar menu dapat dilihat pada gambar 6.1.

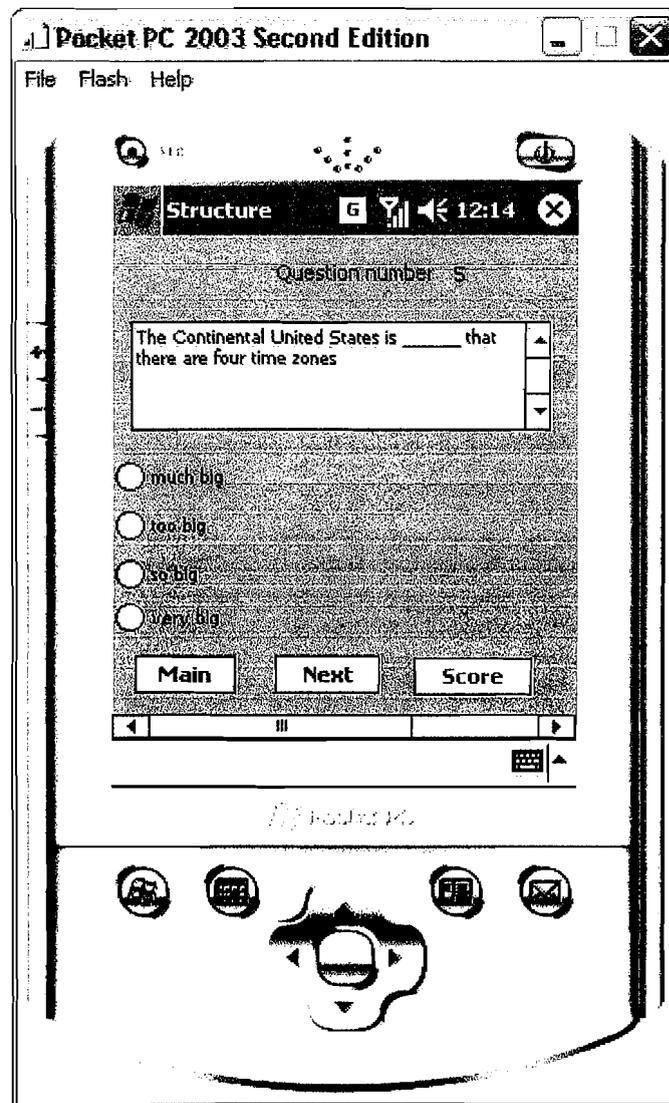
Gambar 6.1 Halaman Menu





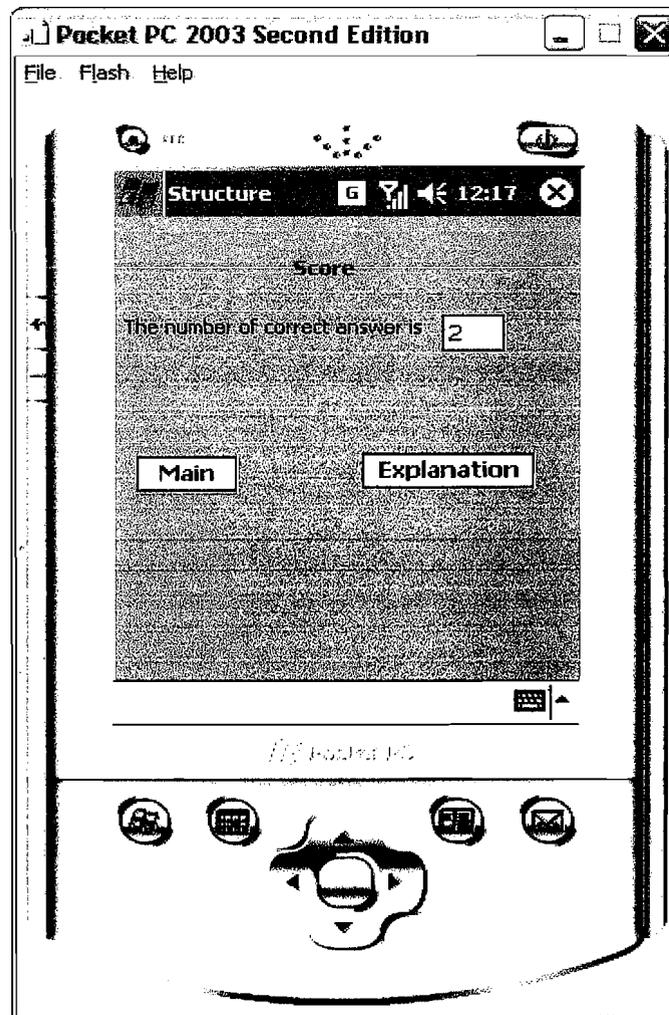
Gambar 6.2 Halaman Structure

Setelah pengguna memilih salah satu menu, yang mana pada bagian ini pengguna memilih Menu Structure, maka akan tampil halaman Structure. Setelah *user* memilih jumlah soal yang akan dilakukan, maka akan tampil halaman soal sesuai jumlah soal yang dipilih. Gambar halaman Structure Soal ada pada dapat dilihat pada gambar 6.2.



Gambar 6.2 Halaman Structure Soal

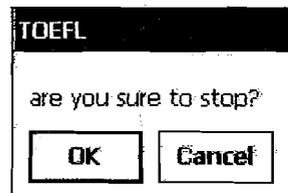
Setelah *user* selesai mengerjakan soal maka *user* dapat mengetahui jumlah score jawaban benar yang telah dikerjakan. Gambar halaman Structure Score ada pada dapat dilihat pada gambar 6.3.



Gambar 6.3 Halaman Structure Score

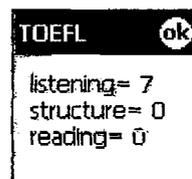
6.2.2 Pengujian Tidak Normal

Pada pengujian tidak normal akan diujikan pada jenis Full Test, apabila *user* menghendaki untuk berhenti dari Test sebelum test selesai dikerjakan, maka akan muncul sebuah kotak peringatan yang dapat dilihat pada gambar 6.4.



Gambar 6.4 Kotak Peringatan

Setelah muncul peringatan tersebut *user* akan memilih 2 opsi tersebut, jika memilih “Cancel”, maka *user* akan kembali meneruskan test hingga selesai. Jika memilih “Yes”, maka *user* akan memperoleh kotak hasil jumlah benar soal yang telah dikerjakan. Kotak Hasil dapat dilihat pada gambar 6.5



Gambar 6.5 Kotak Hasil

6.2.3 Hasil Analisis

Dari hasil pengujian diatas dapat diambil kesimpulan dari kinerja program *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* yaitu:

- a. Program dapat berjalan dengan baik karena pengguna hanya tinggal menggunakan dan memilih menu yang sudah ada.
- b. Program akan memunculkan kotak peringatan apabila *user* menghendaki untuk berhenti dari Test sebelum test selesai dikerjakan berlaku hanya pada jenis Full Test.

6.3 Pembahasan Sistem

Program *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* ini amat mudah untuk digunakan karena pengguna hanya memilih menu-menu yang telah disediakan, dari menu-menu tersebut langsung berhubungan dengan halaman berikutnya yang berisi test yang terkait dengan jenis test yang telah dipilih oleh pengguna.

Kelebihan dari program *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* ini adalah karena program ini terinstall pada perangkat *mobile* sehingga bermanfaat bagi orang-orang yang memiliki keterbatasan waktu untuk melakukan latihan test TOEFL . *User* dapat mengerjakan latihan TOEFL kapanpun dan dimanapun mereka berada selama mereka membawa perangkat *mobile* mereka.

Tetapi program ini juga masih memiliki banyak kekurangan, yaitu seperti jumlah soal yang amat terbatas karena memori Pocket PC yang relatif masih kecil sehingga tidak memungkinkan untuk menyimpan terlalu banyak data di dalam database.

BAB VII

PENUTUP

7.1 Simpulan

Setelah melalui tahapan analisis, desain, pengkodean dan pengujian maka penulis dapat mengambil beberapa kesimpulan sebagai berikut:

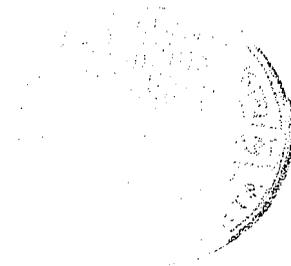
- a. Program *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* dapat membantu pengguna untuk melatih kemampuan TOEFL dengan media yang *portable*.
- b. *User* dapat mengetahui *score* TOEFL yang diperoleh setelah mengerjakan test, sehingga dapat mengukur kemampuannya.
- c. Kekurangan sistem antara lain: jumlah soal yang kurang memadai, masih terbatas jumlahnya

7.2 Saran

Berdasarkan pada pengujian yang telah dilakukan pada perangkat lunak yang dibuat, masih banyak kekurangan dan kelemahan sehingga perlu dikembangkan lagi agar kinerjanya lebih baik, oleh karena itu untuk pengembangan ke depan disarankan:

- a. Jumlah soal pada *Aplikasi TOEFL Untuk Pocket PC Dengan Teknologi .NET* ditambah, seiring dengan bertambahnya memori *Pocket PC*.

- b. Penggunaan animasi untuk desainnya sehingga aplikasi lebih bersifat interaktif dan menarik.



DAFTAR PUSTAKA

- [AKB05] Akbar, A. *Visual Basic .NET Belajar Praktis Melalui Berbagai Tutorial dan Tips*, Bandung: Informatika. 2005
- [DJU02] Djuandi, F. *Pemrograman pada Pocket PC*, Jakarta: Elex Media. 2002.
- [SHA05] Sharpe, PJ. *Barron's How To Prepare For The TOEFL 11th Edition*, Ciputat: Binarupa Aksara. 2005.
- [SHA07] Sharpe, PJ. *Barron's TOEFL Ibt Internet Based Test 12th Edition*, Ciputat: Binarupa Aksara. 2007.