

MOBIL SEGÍTŐTÁRS RENDSZER VAKOK ÉS GYENGÉNLTÓK SZÁMÁRA

Bognár Gábor, bognar@irt.vein.hu
Juhász Zoltán, juhasz@irt.vein.hu
Veszprémi Egyetem, Információs Rendszerek Tanszék
Arató András, arato@sunserv.kfki.hu
KFKI RMKI Beszéd- és Rehabilitáció-Technológiai Osztály
Nagy Zoltán, znagy@speecht.com
Speech Technology Kft.

Kivonat

A vakok és gyengénlátók számára a számítógép használata nehéz feladatot jelent. Ezen az elterjedt ernyőolvasó programok próbálnak enyhíteni, de ezek kezelése sem egyszerű. A kézi számítógépre épülő Mobil Segítőtárs (MOST) kifejlesztésének célja az, hogy segítse a vakok mindennapi feladatainak minél egyszerűbb elvégzését (pl.: hangoskönyv olvasás, magnó hallgatás, e-mailek, sms-ek küldése, fogadása, telefonálás, internet használata, stb.) egy kifejezetten vakok számára készült, grafikus felület nélküli menüvezérelt rendszeren keresztül.

A MOST rendszer első, prototípus verziója megmutatta, hogy elképzelésünk alapján működőképes, használható rendszer hozható létre. Ez a változat azonban nem érte el a termék minőséget. A MOST2 GVOP-AKF projekt keretében a rendszer új, megbízható változata készül el, melyet a vak felhasználók önállóan képesek használni. A kézi számítógép (PDA) vakbarát kezeléséhez elengedhetetlenül szükséges a készülék és a rajta futó operációs rendszer üzeneteinek hallhatóvá tétele (pl.: az akkumulátor lemerülésének jelzése, a hálózati kapcsolat (WIFI) megszakadásának/elérhetőségének jelzése stb.). A rendszer továbbfejlesztése érdekében arra is szükség van, hogy a keretrendszerbe további programok könnyen beilleszthetők legyenek, valamint olyan további programokat kell kifejleszteni, melyek lehetővé teszik pl.: a készülékbe épített vagy bluetooth kapcsolaton keresztül vezérelt telefon kezelését, hangfájlok (mp3) lejátszását és rögzítését, az sms küldést és fogadást.

A projekt célkitűzései közé tartozik egy jóminőségű magyar nyelvű beszédmotor kifejlesztése, valamint a Magyar Elektronikus Könyvtár anyagainak elérhetővé tétele, egy a keretrendszerbe illeszkedő alkalmazás segítségével, és további internet szolgáltatások elérése, mint pl.: on-line zenehallgatás, hírek letöltése.

Előadásunkban bemutatjuk a projekt célkitűzéseit, ismertetjük az eddig elért eredményeket, a kifejlesztett keretrendszert, továbbfejlesztési terveinket, valamint a gyakorlati használat során szerzett vak felhasználók visszajelzéseit.

1. Bevezetés

A vakok és gyengénlátók számára az információs társadalom tagjává válni alapvetően nem az Internet elérés, illetve a web-oldalak olvasásának nehézsége, hanem a számítógép használata miatt nehéz. A látók számára kifejlesztett (bár számukra is gyakran nehézkes), vizuális kiválasztásra és „rámutatásra” épülő grafikus felületek nagymértékben megnehezítik, sok vak felhasználó számára pedig lehetetlenné teszik a gép használatát.

Erre a problémára próbálnak megoldást nyújtani a képernyőolvasó programok [1], melyek a program ablakokban megjelenő szöveget, illetve a grafikus elemeket (pl. lista,

gomb, stb.) reprezentáló információt egy beszéd szintetizátor segítségével (Text-to-Speech, TTS) hanggá alakítják. E módszernek azonban több hátránya is van: (i) valójában csak olyan alkalmazások esetén működik jól, ahol szöveget kell hanggá alakítani, nincs vizuálisan kódolt információ (pl. táblázatok, képek, web oldalak beágyazott programrészekkel, stb.), amit értelemszerűen nehéz hanggá transzformálni, (ii) nem oldja meg az alapproblémát, nevezetesen a számítógép használatának egyszerűsítését. Ez utóbbi különösen fontos lenne idős kori látásromlásban, vakságban szenvedőknek, akik számítástechnikai ismerete nem teszi lehetővé a felolvasóprogramok használatát. [2]

Ezen problémák miatt a prototípus kifejlesztése során kidolgoztunk egy kifejezetten vakok számára készült, nem grafikus felhasználói felületet, amely a programok egységes kezelését teszi lehetővé egy menürendszeren keresztül. Ez a vezérlési elv a BraiLab PC során került kifejlesztésre [3]. A menürendszerben a kézi számítógép navigáló gombjának segítségével mozoghatunk. A programok hierarchikus struktúrába vannak szervezve, melyben a fel-le nyilak segítségével választható ki egy-egy menüelem, a jobbra nyíl segítségével lehet a kiválasztott menübe lépni, illetve a struktúra végén a kiválasztott funkciót végrehajtani. A balra nyíl visszatér az előző szintre (előző menübe vagy programba). A menürendszer fa szerkezete miatt a navigáció mindig egyértelmű, nem lehet a menüben „eltévedni” (a balra nyíl ismételt nyomogatásával a főmenübe kerül a felhasználó).

A kézi számítógépek kétféle módon juthatnak felhasználói adathoz: vagy az érintőképernyőre írással illetve egy virtuális billentyűzet gombjainak megnyomásával lehet gépelni, vagy a géphez kapcsolható külső billentyűzettel. Mivel a vak felhasználók gyakran használnak Braille írógépet vagy Braille táblát (kis fém sablon, mely segítségével a megfelelő helyeken kilyukaszthatják a papírt amelyre írnak) természetesnek tűnt a Braille gépelés lehetőségének kialakítása. A készülék érintőképernyőjét felhasználva kilenc virtuális gombot definiáltunk, melyből hat a Braille pontoknak felel meg, a további három a végrehajtó, törlés és szökő gomb. Mivel az érintőképernyőn egyszerre egy pont lehet lenyomva, ezért a bevinni kívánt karakternek a Braille pontjait egymás után kell megnyomni majd a végrehajtó gombbal jelezni a karakter bevitelének végét. A bevitel „rövidírás” módban is történhet jelentősen növelve ezzel a bevitel sebességét.

A MOST rendszer első, prototípus verziója megmutatta, hogy elképzelésünk alapján működőképes, használható rendszer hozható létre. Ez a változat azonban nem érte el a termék minőséget. A MOST2 GVOP-AKF projekt keretében a rendszer új, megbízható változata készül el, melyet a vak felhasználók önállóan képesek használni.

2. A MOST2 rendszer felépítése és működése

A Mobil Segítőtárs rendszer a PDA hardver eseményeinek kezelésére és az operációs rendszer funkciók elérésére szolgáló C nyelven írt függvénykönyvtárak kivételével, Java nyelven íródott. A Java platform-függetlenségének köszönhetően a PDA-ra kifejlesztett változat minimális módosításokkal (C modulok cseréje után) PC-n is azonnal futtatható, továbbá a felhasználót nem köti egyetlen operációs rendszerhez, valamint a későbbiekben Java-ra épülő szolgáltatások (Jini [5]) elérésére lehetősége nyílik. A hardver illetve operációs rendszer specifikus eszközök, API-k kezeléséhez, eléréséhez C-ben írt függvénykönyvtárakat használunk, melyek függvényeit a keretrendszer JNI (Java Native Interface [4]) segítségével éri el, valamint elrejtí a keretrendszerre épülő alkalmazások elől, és Java környezetben szokásos szolgáltatásokként jeleníti meg azokat (pl.: listener mechanizmus). Ily módon elértük, hogy a keretrendszerbe illeszkedő alkalmazások értesítést kaphatnak a PDA állapotáról

(pl.: akkumulátor töltöttségi szintje, hálózati/akkumulátoros üzem), illetve befolyásolhatják azt (pl.: készülék „felébresztése” adott időpontban, háttérvilágítás lekapcsolása, automatikus kikapcsolási idő beállítása).

2.1. Felépítés

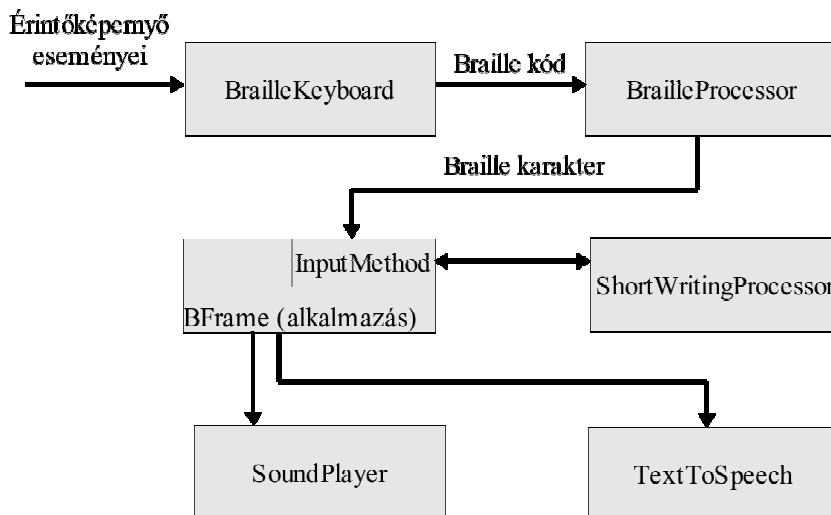
A keretrendszer lehetővé teszi az előre elkészített programok mellett külső fejlesztők által írt alkalmazások számára is – plugin rendszeren keresztül – a különböző rendszerfunkciók elérését. Ezen funkciók, komponensek a TTS, az energiakezelés (*PowerManagement*), a hanglejátszás (wav fájlok), telefon kezelése, SMS küldése fogadása, a program beállításainak mentése, valamint az alkalmazás értesítést kaphat a PDA eseményiről.

A felhasználói felület felépítéséhez a keretrendszer a grafikus felhasználói felület szokásos építőelemeit nyújtja pl.: *BFrame* az alkalmazás ablakának „megjelenítéséhez”, *MenuItem*, *BList*, *BCheckbox* a menük és listák felépítéséhez. Ezek természetesen nem jelennek meg a képernyőn, hanem a leírt módon a menürendszerbe illeszkednek, azokat a felhasználók számára a TTS „jeleníti meg”. Ily módon az alkalmazások fejlesztőinek, a megszokotthoz hasonló elemek felhasználásával, egyszerű az új alkalmazások felhasználói felületének tervezése, fejlesztése.

A keretrendszer moduláris felépítésű, indulásakor egy konfigurációs fájl (property fájl) alapján építi fel, konfigurálja a modulokat (tölti be a megfelelő osztályokat). A moduloknak egy adott interfésznek kell megfelelniük, így egy-egy modul funkciója, szolgáltatása rögzített, implementációja viszont nem kötött. Ezáltal könnyen megvalósítható, hogy a különböző modulokat lecserélhessük, pl.: a magyar nyelvű beszédszintetizátor helyett angol nyelvűt illesszünk a rendszerbe.

A rendszer legfontosabb moduljai az alábbiak.

- A *BrailleKeyboard* a Braille billentyűzet kezelője, az érintőképernyő virtuális gombjaiért felelős, az érintőképernyőről érkező eseményeket alakítja át a Braille pontoknak megfelelő kódokká.
- A *BrailleProcessor* a Braille kódot alakítja át Braille karakterré.
- A *ShortWritingProcessor* feldolgozza a Braille karaktereket és ASCII karakterekké alakítja át (pl.: számjel után az „a” -betű az 1-es számnak felel meg), illetve a rövidírás feldolgozását végzi.
- A *SoundPlayer* hangfájlok lejátszását teszi lehetővé
- A *TextToSpeech* modul a program szöveges üzeneteit alakítja hanggá,



1. ábra Modulok kapcsolata

- A *BFrame* az alkalmazások alapjául szolgáló modul, feladatai az alkalmazás menürendszerében a navigáció megvalósítása, a beviteli események biztosítása (*InputMethod*) és az alkalmazás beállításainak mentése, betöltése.

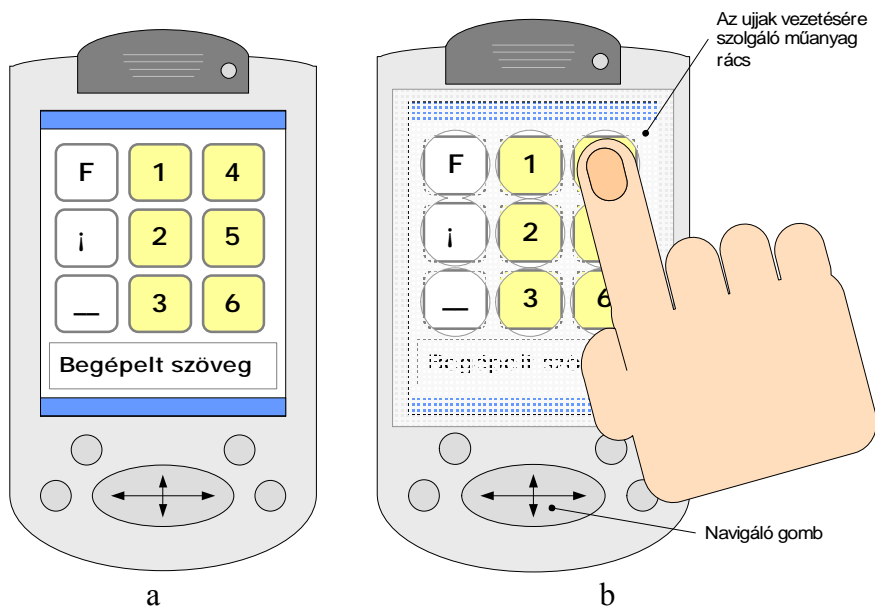
2.2. Működés

Ebben a fejezetben bemutatjuk a készülék működését a felhasználó szemszögéből, illetve a végrehajtás során az egyes modulok együttműködését.

Szövegbevitel

A kéziszámitógép érintőképernyőjét felhasználva kilenc érintési pontot hozunk létre. A 2.a ábrán látható hogy, az 1-6 gombok a Braille pontok megfelelői, míg a további három gomb a karakter elfogadó gomb (F), valamint a törlés (←) és szóköz (_) gombok. A felhasználó az adott karaktert reprezentáló pontokat egymás után megnyomva, majd elfogadva (F) gépeli be a karaktert. Természetesen ezt az érintőképernyő rendszert sem tudnák a vakok ebben a formában használni, hiszen nem tudnák hol helyezkednek el a pontok. Erre a megoldást egy egyszerű kiegészítő jelenti. A 2.b ábrán látható módon az érintőképernyő felett elhelyezünk egy műanyag lapot, mely a pontok felett lyukas. Ez a lap egyrészt az érintési pontok felé vezeti a felhasználó ujját, valamint meggátolja, hogy a képernyőt más ponton nyomják meg. Mivel a képernyőt csak adatbevitelre, és minden program esetén azonos módon, használjuk, a vezető sablon univerzális.[2]

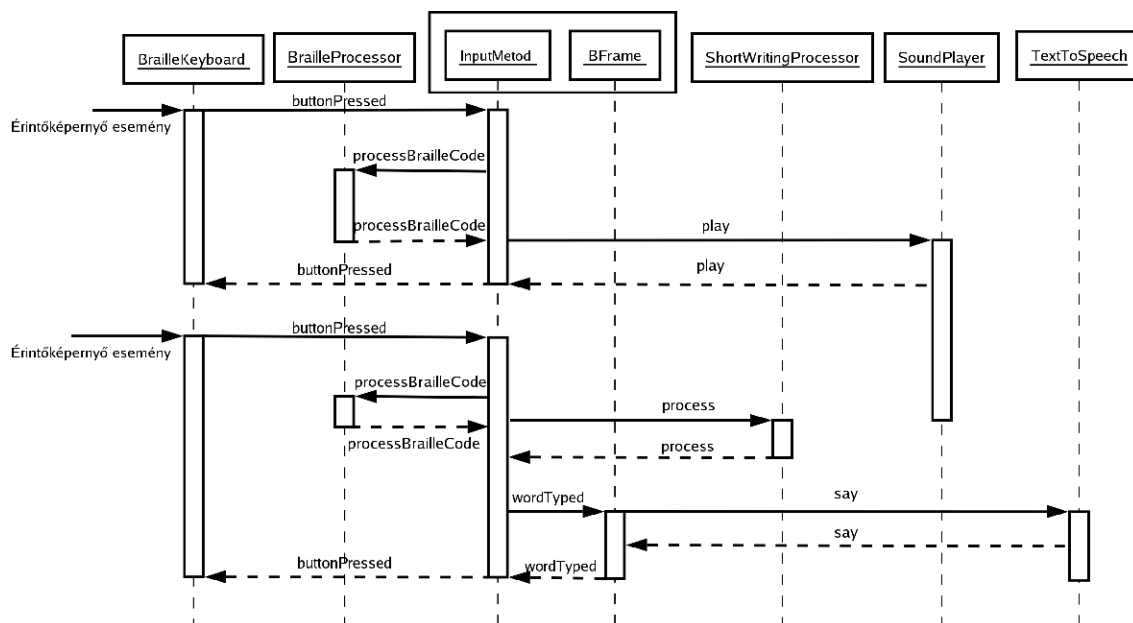
A keretrendszer kialakítása lehetővé teszi, hogy a Braille bevitel mellett, illetve



2:ábra a) A virtuális Braille billentyűzet sematikus rajza.
b) A billentyűzet és a vezető sablon

helyett, a kéziszámitógéphez kapcsolt külső billentyűzetet (esetleg a PDA beépített billentyűzetét) használjuk bevitelre. Ekkor a *BrailleKeyboard* és a *BrailleProcessor* modulok „megkerülésével” a begéptelt karakterek közvetlenül az alkalmazáshoz kerülnek.

Egy „a” betű begépeléséhez a Braille bevittet használva a felhasználó megnyomja az 1-es Braille pontnak megfelelő virtuális gombot, amit egy hangjelzés kísér, majd az F (elfogadó) gombot megnyomva a bevitt karakter („a”) hallható, emberi felvett hang lejátszásával, vagy a *TextToSpeech* modul segítségével (a keretrendszerben beállítható echótól függően). Határoló karakter (szóköz, vessző, pont, felkiáltójel, kérdőjel, pontosvessző) bevitele után a TTS felolvassa a begépelte szót (szintén állítható a beállításokban a szó echó).



3. Példa a modulok közti kommunikációra

Az „a” betű bevitelekor a modulok közti kommunikációt a 3. ábra szemlélteti. A *BrailleKeyboard* modul az elfogadó gomb lenyomásának hatására (miután az „a” betű pontját már előzőleg bevittük) elküldi a *buttonPressed* üzenetet az *InputMethod* modulnak, ami a *BrailleProcessor* segítségével feldolgozza. A példában feltételezzük, hogy a karakter echó felvett hang, a szó echó pedig be van kapcsolva, ezért a *SoundPlayer* modul segítségével lejátszunk az „a” betűt felvett emberi hangon. A következő lépésben egy szó határoló karaktert vitt be a felhasználó (pl.: szóköz). Ekkor az *InputMethod* a *ShortWritingProcessor* modul segítségével a Braille karaktereket ASCII karakterré alakítja, valamint, ha a rövidítés funkció aktiválva van akkor a rövidítés „kifejtését” is elvégzi.

Ezután értesíti az alkalmazást (*BFrame*), hogy a felhasználó begépelte egy szót, az alkalmazás pedig – mivel a szó echó bekapcsolt állapotban van –, a *TTS* modul segítségével kimondja a bevitt szöveget.

3. Alkalmazások

A fejlesztés során eddig a következő felhasználói alkalmazások készültek el:

- Óra, ébresztőóra: Lekérdezhető, illetve beállítható az aktuális idő, dátum. Lehetőség van rendszeres időbemondás és ébresztési időpont beállítására, valamint többféle csörgőhang közül választhatunk.
- Szövegszerkesztő: Egyszerű szövegszerkesztő, mellyel a felhasználó létrehozhat szöveges anyagokat, megnyithat már meglévő állományokat, szerkesztheti és elmentheti azokat. A program a beállított felolvasási mód szerint szavanként,

mondatonként, bekezdésenként vagy folyamatosan felolvassa a szöveget, miközben egy virtuális kurzort mozgat a szövegen belül. Így az aktuális felolvasott szövegrész azonnal szerkeszthető. Lehetőség van továbbá szöveges állományok Internetről történő letöltésére is, ebben az esetben az állományra egy URL cím megadásával hivatkozhatunk, valamint a szerkesztett szöveget SMS üzenetként elküldhetjük a beépített telefon segítségével.

- **Névjegyzék:** Ez egy hagyományos címjegyzéki szolgáltatásokat nyújtó alkalmazás. A bejegyzésekben tárolható nevekhez tartozó e-mail cím, mobil és vezetékes telefonszám, levelezési cím. A tárolt adatokban különböző szempontok alapján keresni lehet, és lehetőség van az adatok közvetlen felhasználására is, mint pl. a telefonszám tárcsázása (megfelelő készülék esetén), illetve elektronikus levél vagy SMS küldése.
- **Telefon:** Mobiltelefon funkciókat megvalósító alkalmazás. Lehetővé teszi tetszőleges telefonszám tárcsázását, SMS küldését. Kezeli a küldött és fogadott, olvasott és olvasatlan üzeneteket. A menüből kiválasztott parancsok segítségével lehetőség van a beérkezett üzenet felolvastatására, megválaszolására, törlésére. Az új üzenetek érkezésére hangjelzéssel figyelmeztet. A bejövő hívások a készülék egy hardver gombja segítségével fogadhatók.
- **E-mail kliens:** Elektronikus levelek küldését, fogadását teszi lehetővé, támogatja a POP3 és IMAP levelező szervereket. Az IMAP protokoll esetén a leveleket mappákba rendezhetjük a szerveren.
- **Magnó:** Az SD memória kártyán elhelyezett mp3 fájlok lejátszását teszi lehetővé, a legalapvetőbb magnó funkcionalitással rendelkezik: lejátszás, pillanatállj, előre - hátra tekerés, megállítás.
- **Számológép:** Egyszerű alpműveletes számológép, kiegészítve szögfüggvények számításának lehetőségével.

A telefon és e-mail kliens programok önálló komponensek, így pl. amikor a névjegyzék vagy a szövegszerkesztő programból (illetve bármely más később kifejlesztésre kerülő programból) szeretnénk SMS-t vagy e-mailt küldeni, akkor a megfelelő komponens építi fel dinamikusan a menüt az alkalmazás számára. Ezzel elértük, hogy mindig egységes felületet (ugyanazt a menüstruktúrát) nyújtsunk a felhasználó számára.

A keretrendszer lehetővé teszi a kézi számítógép beállításainak, állapotának lekérdezését, megváltoztatását a menürendszeren keresztül, pl.: lehetőség van a PDA kikapcsolási idejének beállítására, a hangerő szabályozására, az akkumulátor állapotának lekérdezésére. Nem csak a PDA beállításai változtathatók meg, hanem a program is testre szabható, beállítható a TTS relatív hangereje, sebessége, megadható, hogy szeretnénk-e a szavak bevitele után hallani az éppen bevitt szót (szó echó), illetve hogy az egyes karakterek után felvett emberi hangon, vagy TTS segítségével mondja ki a program a karaktert.

A hálózati alkalmazások számára szükséges internet kapcsolat (GPRS, Bluetooth, WLAN) beállítását illetve létrehozását a kézi számítógépen jelenleg manuálisan, előre meg kell adni, a vak felhasználók ezt még nem tudják megtenni a keretrendszer segítségével, azonban folyik a fejlesztés ennek megvalósítására.

4. Eredmények

A prototípus verziót továbbfejlesztve, új keretrendszert hoztunk létre amely megbízhatóbban kezeli a beérkező eseményeket, új elemeket tartalmaz a menü

felépítéséhez (pl.: *BCheckbox*, *BList*). Lehetővé teszi a PDA beállításainak megváltoztatását, állapotának lekérdezését, pl.: figyelmeztet ha az akkumulátor töltöttségi szintje nem megfelelő, ezáltal elértük, hogy a felhasználók egyedül is képesek használni az eszközt. Néhány felhasználó már eddig is tesztelte, használata a készüléket, a jövőben még több tesztelőt szeretnénk bevonni a fejlesztésbe.

A prototípus fejlesztése során elkészült programokat az új keretrendszerbe illesztettük, további szolgáltatásokkal bővítettük azokat. Új alkalmazásokat fejlesztettünk ki: magnó, számológép, telefon, e-mail kliens. Ezen új programokat a felhasználók jelenleg is használják, tesztelik.

5. Továbbfejlesztési tervek

A projekt hátralévő részében szeretnénk a meglévő alkalmazásainkat továbbfejlesztani, illetve további programok kifejlesztésével segíteni a vak felhasználókat a mindennapi tevékenységek elvégzésében.

Szeretnénk egy tudományos számológépet kifejlesztani, folyamatban van egy színelismerő program fejlesztése is. A további célok között szerepel egy külső telefon illetve headset kezelése bluetooth kapcsolaton keresztül. Fejlesztés alatt áll egy web böngésző is.

A Magyar Elektronikus Könyvtár szolgáltatásainak illetve bármely más elektronikus könyvtár (pl.: Neumann -ház) elérésére egy kereső szolgáltatást szeretnénk kifejlesztani, ami lehetővé tenné, a keretrendszerbe épülve, több hazai könyvtár elérését, keresését egy egységes felületen keresztül.

Az elkészült egyszerű szövegszerkesztőt szeretnénk továbbfejlesztani, hogy képes legyen RTF illetve HTML formátumú fájlok kezelésére, valamint lehetőség legyen alapvető editor parancsok elvégzésére (kivágás, beillesztés, keresés).

6. Irodalomjegyzék

- [1] JAWS képernyőolvasó program.
http://www.freedomscientific.com/fs_products/software_jaws.asp
- [2] Dr. Juhász Zoltán, Dr. Arató András, Hevesi Vilmos, Pratser Frigyes: Java alapú hordozható kliens vakok számára hálózati szolgáltatások elérésére, Networkshop Konferencia, Győr, 2004. január – február
- [3] G. Kiss - A. Arató - J. Lukács - J. Sulyán - T. Vaspöri: BraiLab, a Full Hungarian Text-to-speech Microcomputer for the Blind. Proc. World Conference on Phonetics 1987 aug 1-4.
- [4] Java Native Interface Specification
<http://java.sun.com/j2se/1.5.0/docs/guide/jni/spec/jniTOC.html>
- [5] Sun Microsystems, „The Jini Specification”, <http://www.sun.com/jini>