

Eindwerk: Toegangscontrole van een beschermde ruimte



Auteur: Sigurd Geerts

Promotor: Toon Blondeel

Schooljaar: 2006 – 2007

Jaar: 3^e jaar Toegepaste Informatica (Industriële Informatica)

School: Erasmushogeschool Brussel

Synopsis

Zoals de titel 'Toegangscontrole van een beschermde ruimte' al aangeeft, was dit eindwerk vooral gericht op het beveiligen van de toegang van een ruimte. Het moest vanaf nul worden opgebouwd.

Het eerste deel van het eindwerk bestond erin een printplaat te ontwikkelen. Hierop wordt gebruik gemaakt van een microprocessor (PIC 16F877A), die geprogrammeerd werd in PIC BASIC PRO. Daarnaast werden op deze printplaat ook een LCD en een keypad aangesloten.

Vanaf deze printplaat worden de ingevoerde codes via RS232 doorgestuurd naar een vaste PC. De RS232-communicatie tussen de printplaat (de microprocessor) en de vaste PC wordt gerealiseerd door gebruik te maken van de Nport Express van MOXA (de MOXA module).

Er werd een Windows Service ontwikkeld die ervoor zorgt dat de codes worden gecontroleerd en verder verwerkt. Hiernaast voert deze nog tal van andere functies uit, zoals het sturen van e-mails of het aansturen van een Internet camera of webcam die ervoor zorgt dat er beelden worden gemaakt van de personen die de ruimte betreden. De printplaat en de Windows Service zijn de belangrijkste onderdelen van het eindwerk.

Om te bepalen wie de ruimte mag betreden en om na te gaan door wie de ruimte betreden is, werd er ook nog een webapplicatie ontwikkeld in ASP.NET. Met behulp van deze webapplicatie kunnen gebruikers worden toegevoegd die toegang mogen hebben en kan men een overzicht krijgen van de personen die in de ruimte geweest zijn.

Het eindwerk werd in de hierboven beschreven stappen uitgevoerd. De uitwerking liep niet altijd van een leien dakje; zo weigerden sommige apparaten wel eens om mee te werken.

Vele van deze stappen moesten op een compleet verschillende manier worden geïmplementeerd. Dit maakte de ontwikkeling van het project zeer boeiend, mede omdat er van sommige delen nog helemaal geen informatie voorhanden was en er eerst veel opzoekwerk moest gebeuren. Hierdoor kan echter zeker worden gesteld dat het eindwerk zeer leerrijk was.

Inhoudstafel

INLEIDING	5
1. VOOROPGESTELDE ARCHITECTUUR VAN HET EINDWERK	6
2. EERSTE FASE: DE PRINTPLAAT	8
2.1 DE MICROCONTROLLER	8
2.1.1 <i>Wat is een microcontroller?</i>	8
2.1.2 <i>Hoe de microcontroller programmeren?</i>	9
2.1.3 <i>PIC BASIC PRO Compiler</i>	10
2.2 HET LCD	11
2.2.1 <i>Het type LCD</i>	11
2.2.2 <i>LCDOUT</i>	11
2.3 HET TOETSENBORD (KEYPAD)	12
2.3.1 <i>Het type toetsenbord</i>	12
2.3.2 <i>Op welke toets gedrukt?</i>	12
2.4 RS232 COMMUNICATIE	13
2.4.1 <i>Wat is RS232?</i>	13
2.4.2 <i>Waarom RS232?</i>	13
2.4.3 <i>De MAX232 level converter</i>	14
2.4.4 <i>RS232 communicatie in PIC BASIC PRO</i>	14
2.5 DE PRINTPLAAT	15
2.5.1 <i>Testen op prikbordjes</i>	15
2.5.2 <i>De printplaat maken</i>	15
2.5.3 <i>Problemen met de printplaat</i>	15
2.5.4 <i>Het einde van de eerste fase</i>	15
3. TWEDE FASE: DE MOXA MODULE	17
3.1 DE MOXA MODULE	17
3.1.1 <i>Wat is de Nport Express ?</i>	17
3.1.2 <i>De Nport Express configureren</i>	18
3.1.3 <i>Hoe werkt de Nport Express?</i>	19
3.2 HET TESTEN VAN DE NPORT EXPRESS	20
3.2.1 <i>Het VB.NET testprogramma</i>	20
3.2.2 <i>Einde van de tweede fase</i>	20
4. DERDE FASE: DE WINDOWS SERVICE	22
4.1 DE WINDOWS SERVICE	22
4.1.1 <i>Wat is een Windows Service?</i>	22
4.1.2 <i>Hoe een Windows Service maken en installeren?</i>	22
4.1.3 <i>De functies in de Windows Service</i>	23
4.1.4 <i>Einde van de derde fase</i>	24
5. VIERDE FASE: DE INTERNET CAMERA	25
5.1 DE INTERNET CAMERA	25
5.1.1 <i>Problemen met de Internet camera</i>	25
5.2 DE WEBCAMERA	26
5.2.1 <i>De Quickcam Pro 5000 van Logitech</i>	26
5.3 WINDOWS IMAGE ACQUISITION	27
5.3.1 <i>Wat is Windows Image Acquisition?</i>	27
5.3.2 <i>WIA gebruiken in VB.NET</i>	27
5.3.3 <i>Extra functie voor de Windows Service</i>	28
5.3.4 <i>Einde van de vierde fase</i>	28
6. VIJFDE FASE: DE WEBAPPLICATIE	29
6.1 EHB SECURITY SYSTEM	29
6.1.1 <i>Hoe webapplicatie ontwikkelen?</i>	29

6.1.2 De verschillende webpagina's	29
6.1.2.1 De Login pagina.....	29
6.1.2.2 De Main Menu pagina.....	29
6.1.2.3 De Gebruikers pagina.....	30
6.1.2.4 De Overzicht Gebruikers pagina.....	30
6.1.2.5 De Nieuwe Gebruiker pagina.....	30
6.1.2.6 De Delete Gebruikers pagina.....	31
6.1.2.7 De Pas Gebruiker Aan pagina	31
6.1.2.8 De Toegang pagina	32
6.1.2.9 De Alarm pagina.....	33
6.1.3 Het navigatieschema.....	34
6.1.4 Online plaatsen van de webapplicatie	35
6.1.5 Einde van de vijfde fase.....	35
7. ARCHITECTUUR VAN HET EINDRESULTAAT	36
HET BESLUIT	37
HET DANKWOORD	38
DE LITERATUURLIJST	39
Internetsites:.....	39
Boeken en tijdschriften:.....	39
BIJLAGEN	40
BIJLAGE 1: PIN DIAGRAM PIC 16F877A + MAX 232	40
PIC 16F877A.....	40
MAX 232.....	40
BIJLAGE 2 : PIC BASIC PRO CODE FRAGMENT : TOETSENBORD UITLEZEN	41
BIJLAGE 3 : SCHAKELINGEN PRINTPLAAT	42
Schakeling op de printplaat.....	42
Elektronische schakeling.....	43
BIJLAGE 4 : SCREENSHOTS WEBAPPLICATIE EHB SECURITY SYSTEM.....	44
De Login pagina	44
De Main Menu pagina	45
De Gebruikers pagina	46
De Overzicht Gebruikers pagina.....	47
De Nieuwe Gebruiker pagina.....	48
Delete Gebruikers pagina	49
De Pas Gebruiker Aan pagina (1).....	50
De Pas Gebruiker Aan pagina (2).....	51
De Toegangs pagina (1)	52
De Toegangs pagina (2)	53
De Alarm pagina	54
De Alarm pagina (in geval van alarm).....	55
Voorbeeld van een gestuurde email	56

Inleiding

Dit verslag is een beschrijving van het eindwerk van Sigurd Geerts, een derdejaarsstudent toegepaste informatica (specialisatie Industriële Informatica) aan de Erasmushogeschool te Brussel.

De titel van het eindwerk is 'Toegangscontrole van een beschermde ruimte'. Het moest helemaal zelf worden ontwikkeld, zonder dat er op een basis werd voortgebouwd. Er werden aan de student heel weinig verplichtingen opgelegd; er werd meer gespeeld op zijn originele ideeën om tot een geslaagd eindwerk te komen.

De reden waarom de student voor dit eindwerk had gekozen, was dat er een groot aantal verschillende fasen in moest worden uitgewerkt. Zo had men een fase waar er een elektronische schakeling gemaakt moest worden, dan weer fasen waar er diende te worden geprogrammeerd, enz. De verscheidenheid binnen dit eindwerk sprak de student enorm aan.

Eén van de weinige verplichting die werd opgelegd aan de student, was dat hij voor de codegedeeltes van programma's binnen het eindwerk gebruik moest maken van Visual Basic of C#. Bijkomend waren er enkele vereisten, waaraan het project diende te voldoen.

Na een eerste gesprek met de promotor kreeg de student een aantal apparaten ter beschikking (zoals een MOXA module, een Internet camera, een keypad en een display). Deze dienden te worden gebruikt in de ontwikkeling van het eindwerk; door het bekomen van deze apparaten kreeg de student al een beter zicht op het project.

Nu was het de bedoeling dat hij eerst voldoende informatie opzocht over de apparaten die hem ter beschikking werden gesteld. Nadat de student voldoende informatie bij elkaar gezocht had, kon hij de verschillende fasen bepalen om het toegangscontrole-systeem te ontwikkelen.

In de eerste fase moest een printplaat worden ontwikkeld waarmee een keypad en een LCD aangestuurd werden. Er moest ook een manier gevonden worden om de ingevoerde codes te kunnen doorsturen naar een PC, waar deze dan gecontroleerd moeten worden.

In de tweede fase diende de MOXA module in het eindwerk te worden geïntegreerd. Met deze MOXA module is het de bedoeling dat er aan RS232 communicatie wordt gedaan. Dit is dus de ideale manier om de printplaat te laten communiceren met de PC.

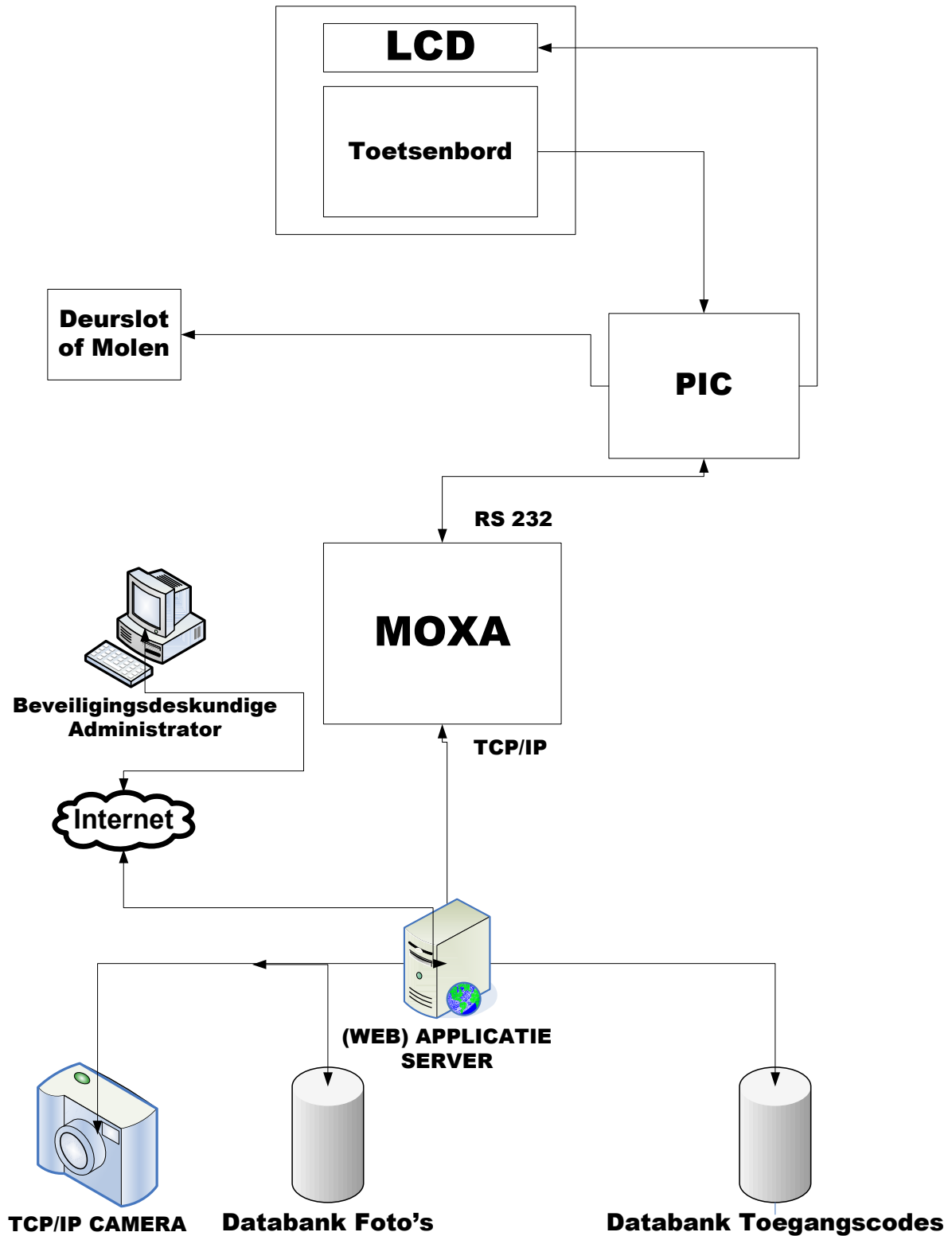
In de derde fase moest een Internet camera aan het eindwerk worden toegevoegd. Het doel van deze Internet camera is om foto's te maken van de personen die de beschermde ruimte betreden hebben.

In een vierde en laatste fase was het de bedoeling een webapplicatie te ontwikkelen waarmee men kan nagaan wie de beschermde ruimte heeft betreden. Tevens behoorde het mogelijk te zijn personen in te voeren in het systeem, zodat deze de beveiligde ruimte kunnen betreden. Deze webapplicatie moest ook communiceren via RS232 met de MOXA module en de codes controleren in de databank.

In dit verslag zal u kunnen lezen hoe de verschillende fasen stap voor stap werden uitgewerkt. Eventuele problemen die optraden tijdens het ontwikkelen van het eindwerk worden ook besproken. Daarnaast wordt aan het einde van dit verslag het eindresultaat besproken en zal de student ook zijn bevindingen geven over het project.

1. Vooropgestelde architectuur van het eindwerk

Na een tweede gesprek met de promotor werd de opstelling van het eindwerk al goed zichtbaar. Hieronder ziet u de architectuur die werd vooropgesteld om het eindwerk succesvol te kunnen realiseren.



Men kan in deze architectuur de verschillende fasen herkennen, zoals reeds besproken in de inleiding. Zo heeft men bovenaan het LCD en het toetsenbord, die worden bestuurd door een PIC (microcontroller). Deze microcontroller zal ook een deurslot aansturen, die open zal gaan wanneer een persoon een correcte code heeft ingetoetst.

Nadat een persoon zijn code heeft ingevoerd, moet deze via RS232 communicatie worden doorgestuurd naar een MOXA module (tweede fase). Deze module moet er dan voor zorgen dat de code via Ethernet (TCP/IP) doorgestuurd wordt naar de webapplicatie. Hier moet de ingevoerde code gecontroleerd worden, door deze te vergelijken met de codes die in de databank zitten.

Als de ingetoetste code teruggevonden wordt, moet de Internet camera aangesproken worden (derde fase). Deze moet een foto maken van de persoon die de ruimte is binnen gegaan, waarop de foto wordt opgeslagen in de databank 'databank foto's'.

Naast het controleren van de ingetoetste code, moet de webapplicatie een administrator eveneens een overzicht kunnen geven van de personen die de ruimte hebben betreden. Met deze webapplicatie moet het ook mogelijk zijn om voor een bepaalde persoon een code aan te maken, zodat deze persoon de beveiligde ruimte kan betreden nadat hij zijn code heeft ingegeven. Ook moet het mogelijk zijn een persoon te verwijderen of zijn gegevens aan te passen (vierde fase).

Het LCD, de PIC, het toetsenbord (keypad) en de Internet camera werden door de school ter beschikking gesteld aan de student.

2. Eerste fase: de printplaat

In de eerste fase van het eindwerk stond de ontwikkeling van een printplaat centraal. Het primaire doel van deze printplaat was een LCD en een toetsenbord (keypad) aan te sturen. Op deze printplaat diende dus een elektronische schakeling komen.

Maar hoe stuurt men nu een LCD en een toetsenbord aan vanuit dit soort elektronische schakeling? Hiervoor moet gebruik worden gemaakt van een microcontroller (PIC).

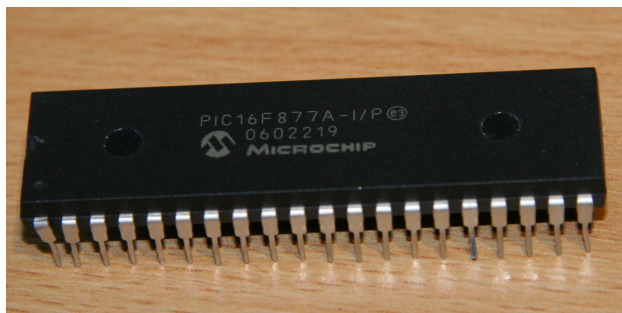
2.1 De microcontroller

2.1.1 Wat is een microcontroller?

Een microcontroller is een digitaal IC dat volledig zelfstandig taken kan verrichten. Men kan een microcontroller het beste vergelijken met een kleine computer, omdat een microcontroller ook zijn eigen (kleine) geheugen heeft waarin men data - het programma - kan laden.

Het is zo dat een microcontroller gevoed moet worden door een spanningsbron van vijf volt om het programma uit te voeren dat in zijn geheugen zit. Eén van de voordelen is dat het geheugen van de microcontroller niet vrij wordt gemaakt wanneer het geen voedingsspanning ontvangt. Met andere woorden verliezen we het programma dat we in het geheugen van de microcontroller geladen hebben niet telkens wanneer het toestel geen spanning ontvangt.

Aan een microcontroller zijn verscheidene pinnen bevestigd, die in aantal van achttien tot veertig variëren; dit hangt af van het type toestel. Er zijn microcontrollers waarvan de pinnen enkel kunnen gebruikt worden als normale I/O poorten, maar bij andere kunnen de pinnen gebruikt worden als A/D converters of als RS232 poorten.



Er werd gekozen om op de printplaat voor het eindwerk gebruik te maken van de PIC 16F877A van Microchip (waarvan u hierboven een afbeelding ziet). Aangezien er zeker een twintigtal I/O poorten nodig zijn om het LCD en het toetsenbord aan te sturen, was een microcontroller met veertig pinnen geen overbodige luxe. Van deze veertig pinnen waren het grootste deel gewone I/O poorten.

Er waren ook twee poorten die als RS232 poorten gebruikt konden worden, maar omdat op de printplaat enkel gewerkt werd met I/O poorten, was dit in principe overbodig.

Het volledige pindigram van de PIC 16F877A kan u terugvinden in de bijlagen (Bijlage 1).

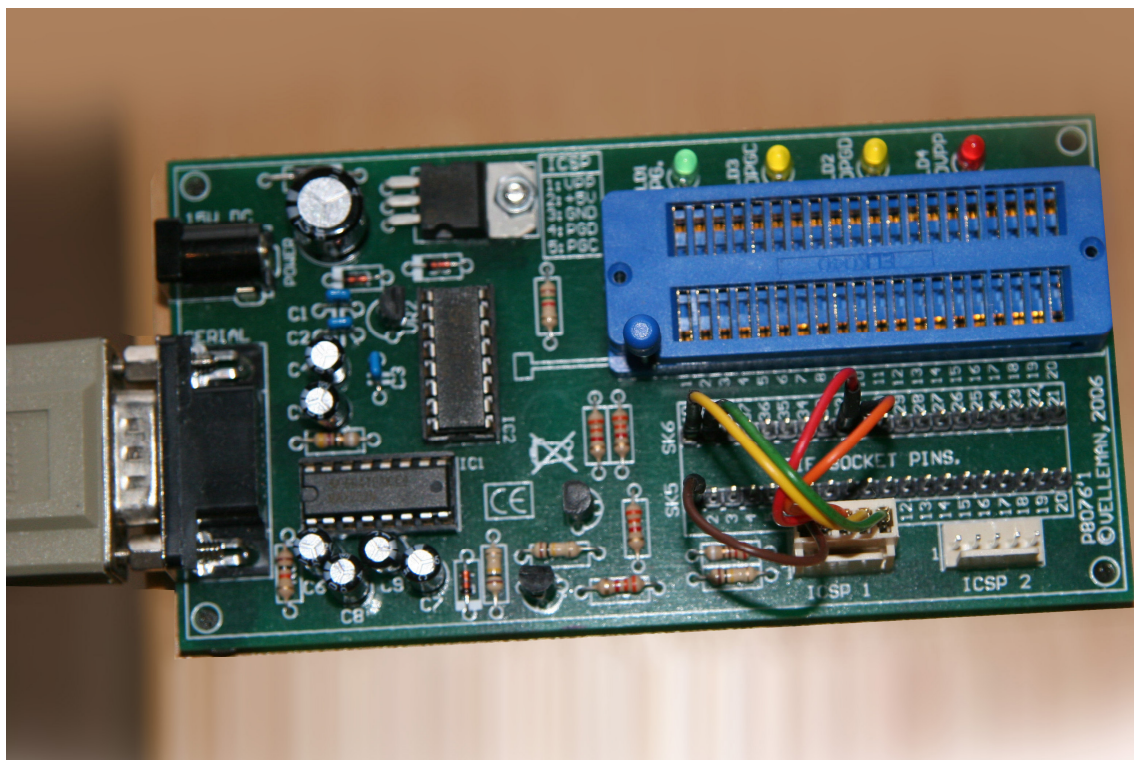
2.1.2 Hoe de microcontroller programmeren?

Zoals reeds beschreven, kan men in het geheugen van de microcontroller een programma laden zodat de microcontroller dit programma uitvoert. De vraag is nu hoe men het programma dat men geschreven heeft in het geheugen van de microcontroller kan laden.

Dit kan heel simpel gebeuren door gebruik te maken van een PIC-programmer. Een PIC-programmer is te vergelijken met een printplaat waarop men de microcontroller plaatst die men wenst te programmeren. Deze printplaat wordt dan verbonden met de seriële poort van de PC.

In het tweede jaar hadden wij reeds een PIC-programmer ontwikkeld. Het probleem van deze programmer was dat deze alleen microcontrollers met achttien pinnen kon programmeren; de microcontroller die gebruikt zou worden in het eindwerk telde veertig pinnen. Dit kon worden opgelost door bepaalde pinnen van de microcontroller te verbinden met bepaalde pinnen van de IC voet op de PIC-programmer.

Omdat dit niet al te gemakkelijk was en er soms conflicten optraden, werd er besloten gebruik te maken van een PIC-programmer van Velleman, aangezien deze microcontrollers van veertig pinnen kan programmeren. Deze moest nog worden gemonteerd, maar dit ging vrij vlot. Hieronder ziet u een afbeelding van de Velleman PIC-programmer.



Op de blauwe socket kan men de gewenste microcontroller plaatsen; deze mag tot veertig pinnen hebben.

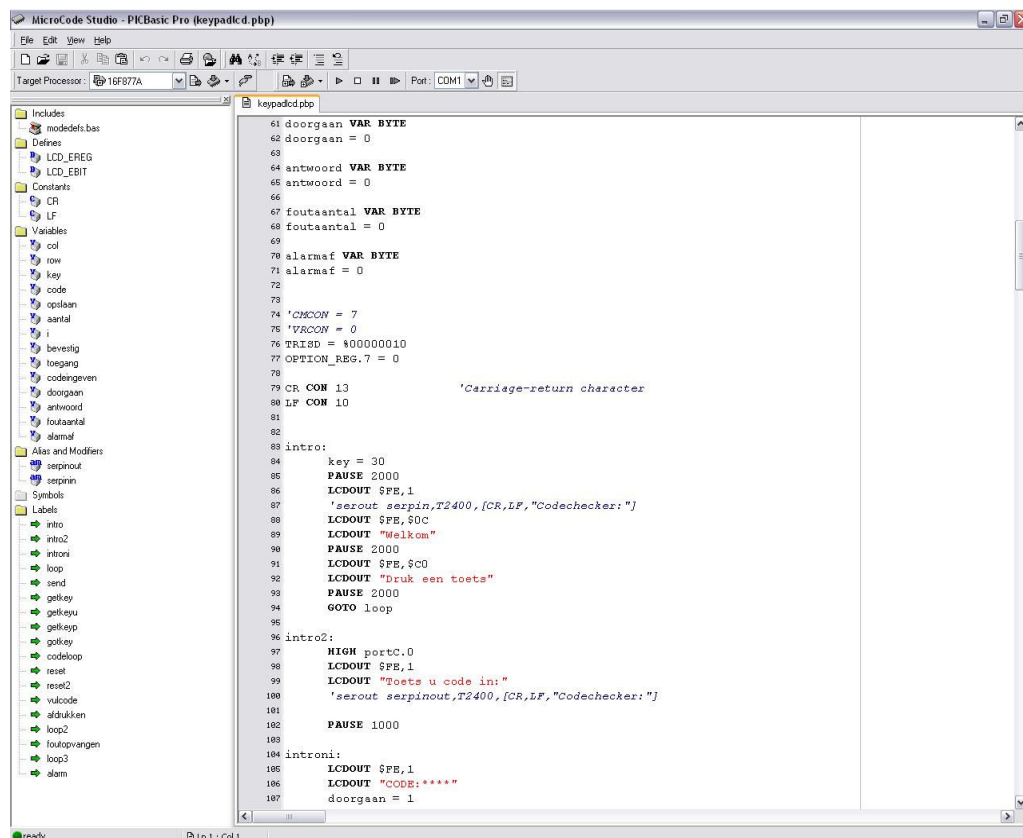
Bij de Velleman-kit werd ook een programma geleverd, PicProg 2006, waarmee men de data in de microcontroller kon laden. Er kunnen enkel gecompileerde programma's (HEX files) in de microcontroller geladen worden, waardoor het programma dat in de microcontroller geladen diende te worden eerst in een andere taal geschreven moest worden. Hierna kon het gecompileerd worden tot een HEX file, dat in de microcontroller geladen kon worden.

2.1.3 PIC BASIC PRO Compiler

De meeste programma's die in een microcontroller geladen worden, zijn geschreven in assembler. Voor het eindwerk leek het echter onverstandig het programma in assembler te schrijven.

Bij het zoeken naar een compiler voor de PIC 16F877A, werd eerst gezocht naar een C-compiler, maar deze werd niet gevonden. Daarom besloot men de PIC BASIC PRO Compiler te gebruiken. Deze compiler was een uitbreiding van de vorige Basic compiler; er waren reeds functies in geïntegreerd die RS232 communicatie en het aansturen een LCD van mogelijk maakten.

We hadden al een compiler, maar nu moest er nog een programma gezocht worden waarin deze compiler gebruikt kon worden. MicroCode Studio bleek het soort programma te zijn waar de PIC BASIC PRO Compiler zonder probleem gebruikt kon worden. Hieronder ziet u een screenshot van het programma MicroCode Studio.



```
MicroCode Studio - PICBasic Pro (keypad.cbp)
File Edit View Help
Target Processor: 16F877A Port: COM1

Includes
- modeldefs.bas
Defines
- LCD_EREG
- LCD_ERBIT
Constants
- CR
- LF
Variables
- col
- row
- key
- code
- opslaan
- aantal
- i
- bevestig
- leesging
- codergeven
- doorgaan
- antwoord
- foutaantal
- alarmaf
Alias and Modifiers
- serpinout
- serpinin
Symbols
Labels
- intro
- intro2
- introni
- loop
- send
- getkey
- getkeyu
- getkeyp
- getkey
- codeleop
- reset
- reset2
- valcode
- afdruksien
- loop2
- foutopvangen
- loop3
- alarm

61 doorgaan VAR BYTE
62 doorgaan = 0
63
64 antwoord VAR BYTE
65 antwoord = 0
66
67 foutaantal VAR BYTE
68 foutaantal = 0
69
70 alarmaf VAR BYTE
71 alarmaf = 0
72
73
74 'CHECKON = 7
75 'TRCOON = 0
76 TRISD = 800000010
77 OPTION_REG.7 = 0
78
79 CR CON 13           'Carriage-return character
80 LF CON 10
81
82
83 intro:
84     key = 30
85     PAUSE 2000
86     LCDOUT $FE,1
87     'serout serpin,T2400,[CR,LF,"Codechecker:"]
88     LCDOUT $FE,$OC
89     LCDOUT "Welkom"
90     PAUSE 2000
91     LCDOUT $FE,$C0
92     LCDOUT "Druk een toets"
93     PAUSE 2000
94     GOTO loop
95
96 intro2:
97     HIGH portC.0
98     LCDOUT $FE,1
99     LCDOUT "Toets u code in:"
100    'serout serpinout,T2400,[CR,LF,"Codechecker:"]
101
102    PAUSE 1000
103
104 introni:
105    LCDOUT $FE,1
106    LCDOUT "CODE:****"
107    doorgaan = 1
```

Nadat het programma volledig geschreven was in PIC BASIC PRO, werd het door de compiler omgezet naar een HEX file, die dan in de microcontroller werd geladen via PicProg 2006 en de Velleman-programmer.

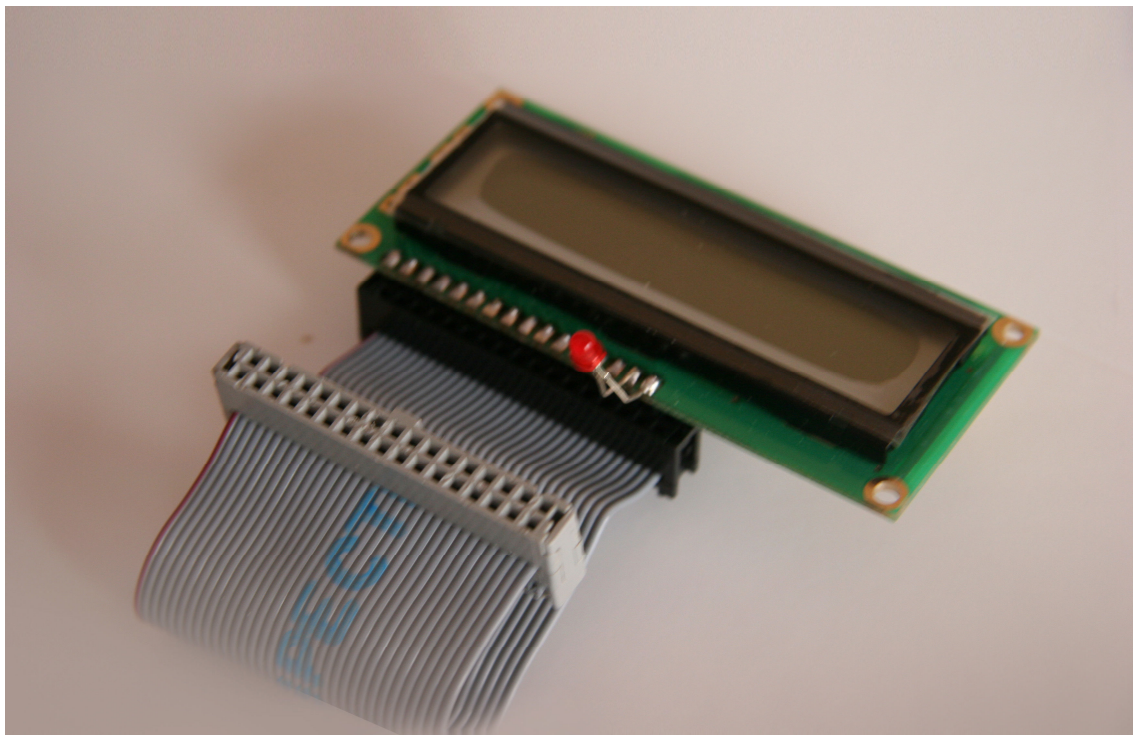
2.2 Het LCD

2.2.1 Het type LCD

Het LCD (Liquid Cristal Display) dat door de school ter beschikking werd gesteld, was een LCD met een chip van het type HD44780 van Hitachi. Het bleek om een LCD te gaan met zestien pinnen.

Van deze zestien pinnen er acht voorzien als datalijnen. De microcontroller in het project werd echter zo ingesteld dat hij maar vier datalijnen nodig had, aangezien er slechts vier nodig zijn om het LCD aan te sturen. Dit werd aangepast in het PIC BASIC PRO programma.

Op het LCD kunnen maximaal twee regels van zestien karakters geplaatst worden. Dit LCD moest net als de microcontroller gevoed worden door een externe voeding, maar aangezien de vereiste spanningen even groot waren (5V), was dit geen probleem. Hieronder ziet u een afbeelding van het LCD.



2.2.2 LCDOUT

Dankzij de PIC BASIC PRO Compiler was het zeer eenvoudig om tekst naar het LCD te sturen. Men moest enkel het volgende commando gebruiken:

LCDOUT "Welkom"

De functie LCDOUT zorgde ervoor dat de tekst (in dit geval 'Welkom') naar het LCD gestuurd werd, waarop dan de gewenste tekst tevoorschijn kwam.

Men diende er wel voor te zorgen dat het LCD aan de juiste pinnen van de microcontroller aangesloten werden. Of men moest, indien men het LCD aan andere pinnen van de microcontroller aansloot, pinnen definiëren in PIC BASIC PRO, waarna men eveneens de functie LCDOUT kon gebruiken om tekst te laten verschijnen op het LCD.

2.3 Het toetsenbord (keypad)

2.3.1 Het type toetsenbord

Het toetsenbord, dat ook door de school ter beschikking werd gesteld, was opgebouwd uit een standaard vier maal vier matrix. Achteraan op het toetsenbord waren negen pinnen bevestigd, waarvan de eerste vier gebruikt werden voor de kolommen. Daarna volgden dan vier pinnen die gebruikt werden voor de rijen van het toetsenbord. De negende pin van het toetsenbord moest met de massa worden verbonden.

Dit toetsenbord had geen externe voeding nodig, aangezien het geen stroom vereiste. De acht eerste pinnen van het toetsenbord werden verbonden met acht standaard I/O poorten van de microcontroller.

Hieronder ziet u een afbeelding van het toetsenbord.



2.3.2 Op welke toets gedrukt?

Om na te kunnen gaan welke toets van het toetsenbord ingedrukt is, moest er een algoritme ontwikkeld worden. Daarom werd in het programma, dat in de microcontroller werd geladen, een deeltje code geschreven waarbij de vier rijen aanzien werden als ingangspoorten en de vier kolommen werden aanzien als uitgangspoorten.

Alle uitgangspoorten worden laag gezet en daarna worden de rijpinnen van de microcontroller één voor één hoog gezet. Nu wordt elke kolom - binnen de rij waarvan de rijpin hoog gezet werd - afgegaan. Indien er een kolompin van de microcontroller ook hoog staat, weet men perfect welke toets werd ingedrukt.

In de bijlagen vindt u het codefragment waarmee het toetsenbord wordt uitgelezen terug (Bijlage 2).

2.4 RS232 communicatie

2.4.1 Wat is RS232?

RS232 is een standaard die een communicatie methode beschrijft, waarbij informatie bit voor bit over een fysiek kanaal - bijvoorbeeld tussen twee computers - wordt verstuurd. Een andere naam voor RS232 is seriële communicatie.

De informatie, die wordt verzonden, moet opgedeeld zijn in datawoorden (frames). Deze datawoorden bestaan uit een opeenvolging van bits (nullen en enen). Het aantal bits dat in dit soort datawoorden zit, kan ingesteld worden op een PC, maar dit aantal ligt altijd tussen vijf en acht.

Voor een goede overdracht worden extra bits toegevoegd, waardoor synchronisatie en foutdetectie mogelijk worden gemaakt. Het is wel belangrijk dat zowel de zender als de ontvanger ingesteld staan op hetzelfde aantal databits. Indien dit niet zo is, kan de binnenkomende informatie fout worden geïnterpreteerd, of zelfs helemaal niet worden herkend.

Databits worden met een vastgelegde frequentie verzonden, de baudrate. Zowel de zender als de ontvanger moeten geprogrammeerd zijn voor het gebruik van dezelfde bitfrequentie. Nadat de eerste bit ontvangen is, berekent de ontvanger op welke momenten de volgende databits kunnen worden verwacht.

RS232 definieert een asynchrone manier van communiceren. Dit betekent dat het verzenden van een datawoord op elk moment kan beginnen. Dit levert enkele problemen op aan de ontvangtzijde om te detecteren welke van de ontvangen bits de eerste van een datawoord is.

Om dit probleem op te lossen wordt elk datawoord voorafgegaan door een signaleringsbit, ook wel de startbit genoemd. Deze startbit is te herkennen aan zijn laag signaalniveau (nul). Omdat de lijn hoog staat (één) als er geen communicatie is, kan de startbit eenvoudig door de ontvanger worden herkend.

2.4.2 Waarom RS232?

Op de printplaat die voor het eindwerk ontwikkeld moest worden, was zeker RS232 communicatie nodig aangezien de code die door een persoon wordt ingetoetst op het toetsenbord, op de één of andere manier moet kunnen worden doorgestuurd naar een PC. De keuze om hiervoor RS232 communicatie te gebruiken leek bijgevolg logisch.

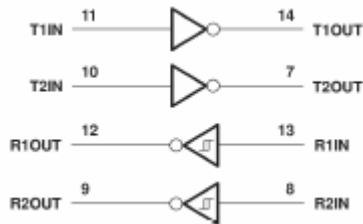
Een eerste voorwaarde om aan RS232 communicatie te doen, is het gebruik van een DB 9 connector. Deze connector werd voorzien op de printplaat en zou er dan voor zorgen dat de printplaat via een seriële kabel verbonden kon worden met een seriële poort. Dit zorgde ervoor dat er gecommuniceerd kon worden tussen de printplaat en een PC.

Er bleek wel een probleem te zijn, men kon namelijk de TX pin (om data te versturen) en de RX pin (om data te ontvangen) van deze DB 9 connector niet rechtstreeks verbinden met twee pinnen van de microcontroller. Dit omdat een seriële poort volgens de RS232 standaard werkt - RS232 werkt met logische niveaus van +15 Volt en -15 Volt - en de microcontroller met TTL niveaus werkt. We waren dus verplicht een omzetter te gebruiken, indien we de RS232 signalen met de microcontroller wilden verbinden.

2.4.3 De MAX232 level converter

De MAX232 is een heel bekend IC, dat vaak gebruikt wordt om de signalen van TTL niveaus om te zetten naar RS232 niveaus.

Het IC is intern opgebouwd uit twee receivers, die de signalen van de seriële poort omzetten naar TTL, en twee drivers, die de TTL niveaus omzetten naar RS232 niveaus. Het omzetten van de TTL niveaus naar RS232 niveaus is mogelijk dankzij de ingebouwde spanningspomp van de MAX232, die zelf spanningen opwekt. Hieronder ziet u het logisch diagram van de MAX232.



Om deze MAX232 level converter te kunnen laten werken, moet deze worden gevoed met een spanning van vijf volt. Dit was echter geen probleem, aangezien de microcontroller ook vijf volt nodig had om te werken.

2.4.4 RS232 communicatie in PIC BASIC PRO

Vervolgens diende een manier gevonden te worden om de RS232 communicatie mogelijk te maken vanuit de microcontroller. Met andere woorden, men moest er op de één of andere manier voor zorgen dat de ingevoerde code vanuit de microcontroller naar een seriële poort kon worden doorgestuurd.

De oplossing hiervoor was snel gevonden, aangezien in de PIC BASIC PRO Compiler reeds functies aanwezig zijn, die RS232 communicatie vanuit een microcontroller mogelijk maken. Deze zijn:

serout serpinout,T9600,[CR,LF,"BINNEN"]

serin serpinin,T9600,alarmaf

Met de eerste functie is het de bedoeling dat er data vanuit de microcontroller naar de seriële poort verstuurd kan worden. Bij deze functie moet men eerst de pin aangeven die TX pin is op de microcontroller. Men geeft dus mee welke pin van de microcontroller gebruikt zal worden om data te versturen (hier is dat serpinout). Daarna wordt dan de baudrate aangegeven (T9600). Als laatste wordt de tekst meegegeven (hier: 'BINNEN', dat vooraf gegaan wordt door een nieuwe lijn).

Met de tweede functie is het de bedoeling dat data die verstuurd werd vanuit de seriële poort ontvangen kan worden door de microcontroller. Bij deze functie moet men eerst meegeven welke pin van de microcontroller de RX pin is (de pin die data gaat ontvangen). Daarna wordt de baudrate aangegeven (T9600) en ten slotte kan er een variabele meegegeven worden, waarin de ontvangen data bewaard moet worden (alarmaf).

Het is ook zo dat de SERIN functie wacht tot er data ontvangen is voor er naar de volgende instructie gesprongen wordt in het geheugen van de microcontroller. Men kan dit wachten eventueel wel omzeilen door gebruik te maken van een timer.

2.5 De printplaat

2.5.1 Testen op prikboardjes

Nadat al het onderzoekwerk in verband met de benodigde componenten achter de rug was, werd er gestart met de ontwikkeling van een elektronische schakeling. Het schema hiervan is terug te vinden in de bijlagen (Bijlage 3). Deze schakeling werd niet meteen op een printplaat gemaakt, maar werd eerst getest op een aantal prikboardjes.

Het voordeel van deze prikboardjes was dat het PIC BASIC programma, dat in de microcontroller zat, ook meteen getest kon worden. Indien er nog fouten in het programma zaten kon men de microcontroller eenvoudig van het prikboard halen en deze herprogrammeren. Er kon eveneens meteen getest worden of alle componenten werkten.

Het testen op deze prikboardjes duurde ongeveer een week, maar de resultaten waren positief.

2.5.2 De printplaat maken

Aangezien alles naar behoren functioneerde op de prikboardjes, werd er besloten de elektronische schakeling op een printplaat te maken. Nadat de printplaat geëtst was (dit houdt in dat alle koperbanen zichtbaar zijn op de printplaat), konden de elektronische componenten van de prikboardjes worden gehaald en op de printplaat gesoldeerd worden.

2.5.3 Problemen met de printplaat

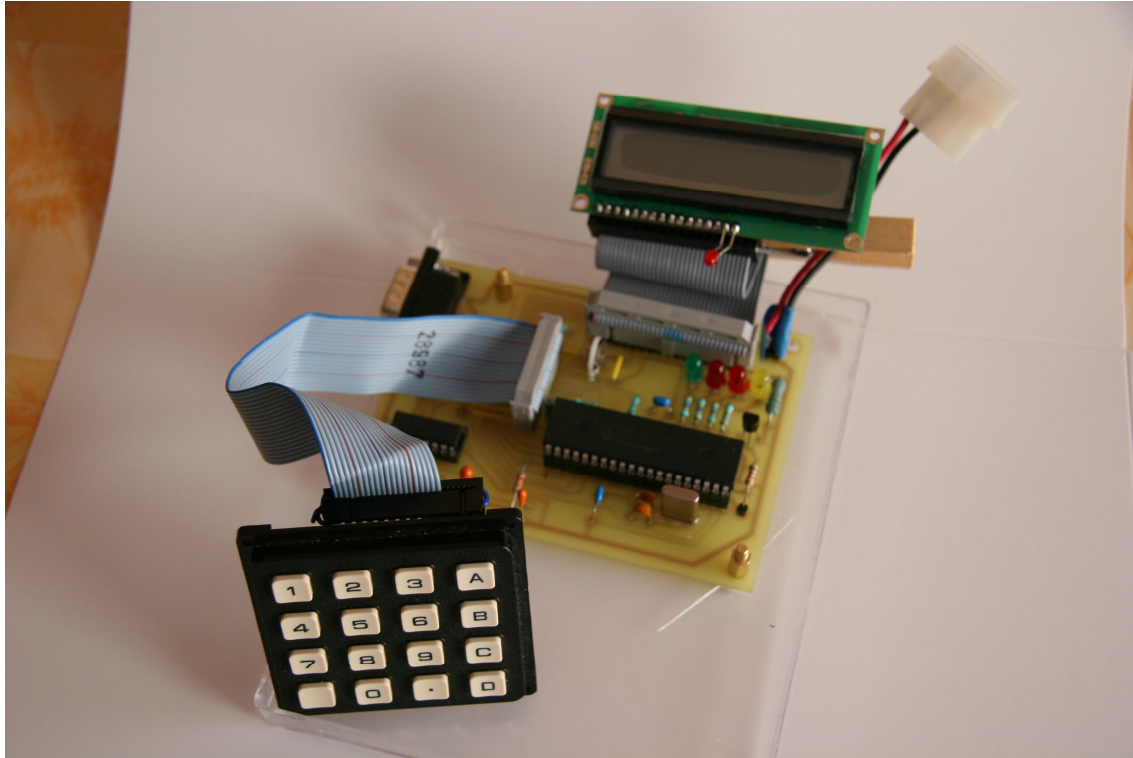
Het werd tijd om de ontwikkelde printplaat te gaan testen, maar bij deze testfase bleek dat er een aantal problemen waren met de printplaat. Er kwamen vreemde tekens op het LCD, in plaats van de de gewenste tekst.

Er werd eerst vanuit gegaan dat het LCD niet meer werkte ; daarom werd een nieuw LCD aan de printplaat gekoppeld. Ook hierna bleven echter vreemde tekens verschijnen op het LCD. De oorzaak van het probleem bleek te liggen bij de koperbanen.

Daarom werd na enige tijd besloten een nieuwe printplaat te ontwikkelen, waarop nieuwe elektronische componenten gesoldeerd werden.

2.5.4 Het einde van de eerste fase

Nadat de elektronische componenten op de nieuwe printplaat geplaatst waren, kon ook deze een eerste maal getest worden. Deze printplaat werkte wel perfect.



Hierboven ziet een afbeelding van de ontwikkelde en werkende printplaat.

We beschikken nu over een printplaat waarop een microcontroller ervoor zorgde dat een LCD aangestuurd werd. Op dit LCD verschijnt een boodschap om de persoon, die de beschermde ruimte binnen wil gaan, te verwelkomen.

Daarna verschijnen er boodschappen die naar de toegangscode van de persoon gaan vragen. De persoon heeft drie kansen om zijn correcte toegangscode in te geven via het toetsenbord. De ingetoetste code wordt via RS232 communicatie doorgestuurd naar een seriële poort, waar de ontvangen code wordt gecontroleerd.

Na de controle stuurt deze seriële poort een antwoord terug, dat via de MAX232 terecht komt bij de microcontroller. Afhankelijk van het gekregen antwoord wordt dan een nieuwe boodschap aan de persoon getoond. Indien de ingegeven code correct is, stuurt de seriële poort een één, waarop de persoon de ruimte kan betreden. Bij een foutieve code zendt de seriële poort een nul en wordt de persoon gevraagd zijn code opnieuw in te toetsen.

Als de persoon na een derde poging opnieuw een verkeerde code ingeeft, treedt het beveiligingssysteem in werking en wordt het alarm geactiveerd. Niemand kan zijn code intoetsen alvorens het alarm uitgeschakeld is. Het uitschakelen van het alarm moet door de verantwoordelijke van de beschermde ruimte gebeuren.

Dit was het einde van de eerste fase.

3. Tweede fase: de MOXA module

In de tweede fase van het eindwerk was het de bedoeling de ter beschikking gestelde MOXA module te integreren in het eindwerk.

3.1 De MOXA module

3.1.1 Wat is de Nport Express ?

De gekozen MOXA module was de Nport Express, geproduceerd door Moxa Technologies. Het doel van dit apparaat is een toestel in staat te stellen via RS232 of RS485 te communiceren met PC's binnen een netwerk. De Nport doet dienst als virtuele com-poort.

Het is een erg handig toestel om oude apparaten van vroeger - en van nu -, die enkel kunnen communiceren via een seriële poort, te laten communiceren met een PC (of verschillende PC's) binnen een netwerk.

Deze Nport Express heeft een seriële ingang, die verbonden wordt met het seriële apparaat (in ons geval de printplaat), en een Ethernet uitgang, die kan verbonden worden met een Hub, Router, Switch, een PC, enz.

Deze Nport van MOXA moest extern gevoed worden door een externe spanningsbron van twaalf Volt.

Op de volgende pagina ziet u een afbeelding van de Nport Express van Moxa Technologies.



3.1.2 De Nport Express configureren

Hoe kon men deze Nport van Moxa gebruiken in het project? Met behulp van de bijgeleverde software van Moxa kon men dit apparaat een eigen IP adres geven, waardoor alle PC's binnen hetzelfde netwerk deze Nport konden aanspreken. Er diende enkel rekening mee te worden gehouden dat de range van de Nport binnen dezelfde range viel als die van het netwerk.

Als de Nport Express een eigen IP-adres heeft, doet het eigenlijk dienst als virtuele com-
poort die kan aangesproken worden door de verschillende PC's binnen het netwerk.

Als men de Nport Express wil gebruiken voor RS232 communicatie, moeten ook het aantal start- en stopbits en de baudrate ingesteld worden. De ingestelde gegevens moeten helemaal gelijk zijn aan deze van het seriële apparaat; indien dit niet zo is, is er geen communicatie mogelijk.

De Nport kon op verschillende manieren worden ingesteld, aangezien met dit toestel zowel RS232 als RS485 communicatie mogelijk gemaakt werd. De printplaat, die zoals eerder besproken ontwikkeld werd in de eerste fase van het eindwerk, werkte met RS232 communicatie. Hierdoor moest de Nport Express zo ingesteld worden dat er RS232

communicatie kon gebeuren, opdat het mogelijk werd de printplaat te laten communiceren met een PC binnen het netwerk. Het instellen van de Nport gebeurde door een aantal schakelaars achteraan de Nport op één te zetten.

3.1.3 Hoe werkt de Nport Express?

De Nport Express zorgt er voor dat er een transparante connectie wordt gelegd tussen de printplaat en een PC binnen het netwerk.

Het apparaat werd ingesteld om aan RS232 communicatie te doen, waardoor het volgende binnen de Nport Express gebeurt. Wanneer de Nport databits ontvangt via de seriële ingang, zullen deze databits verpakt worden in een TCP/IP pakket. Dit TCP/IP pakket wordt dan via de Ethernet poort verstuurd over het netwerk en komt zo bij de correcte PC terecht.

Wanneer een PC binnen het netwerk een antwoord terug stuurt, krijgt de Nport eerst een TCP/IP pakket binnen via de Ethernet poort; in dit geval zal de Nport dit TCP/IP pakket uitpakken. De data die in het TCP/IP pakket zat wordt vervolgens omgezet naar databits, die via de seriële poort van de Nport verstuurd worden. Deze databits komen dan via een seriële kabel terecht bij het seriële apparaat (de printplaat).

Er kan dus in twee richtingen gecommuniceerd worden via de Nport Express.

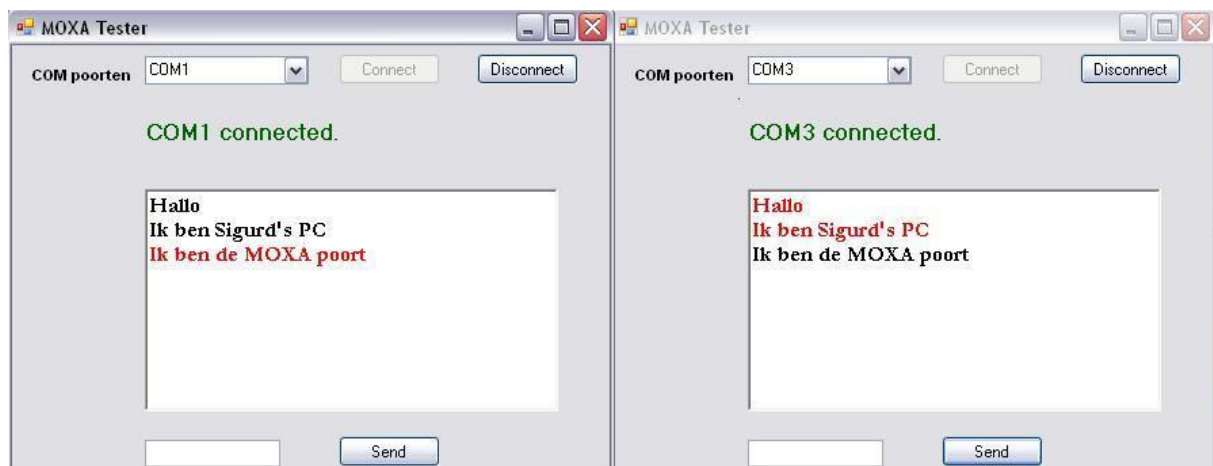
Het is wel zo dat de data, die ontvangen wordt door de Nport Express, opgeslagen wordt in een buffer totdat deze data opgevraagd wordt. Het opvragen van de data kan door een pc of door het seriële apparaat gebeuren.

3.2 Het testen van de Nport Express

3.2.1 Het VB.NET testprogramma

Om na te gaan of de Nport Express goed geconfigureerd was, werd er een programma geschreven in VB.NET. Voordat dit programma gestart werd, sloot men de seriële poort van de Nport Express aan op de seriële poort (COM1) van een PC binnen het netwerk. De Ethernet poort van de Nport Express werd verbonden met een Hub.

Hieronder ziet u een afbeelding van het testprogramma.



De bedoeling van dit testprogramma was berichten te kunnen sturen van de seriële poort van een PC (COM 1) naar de Nport Express (COM 3) en vice versa. In het hierboven getoonde voorbeeld is de rode tekst de tekst die ontvangen werd door de poort. De zwarte tekst is de tekst die door de poort verstuurd werd.

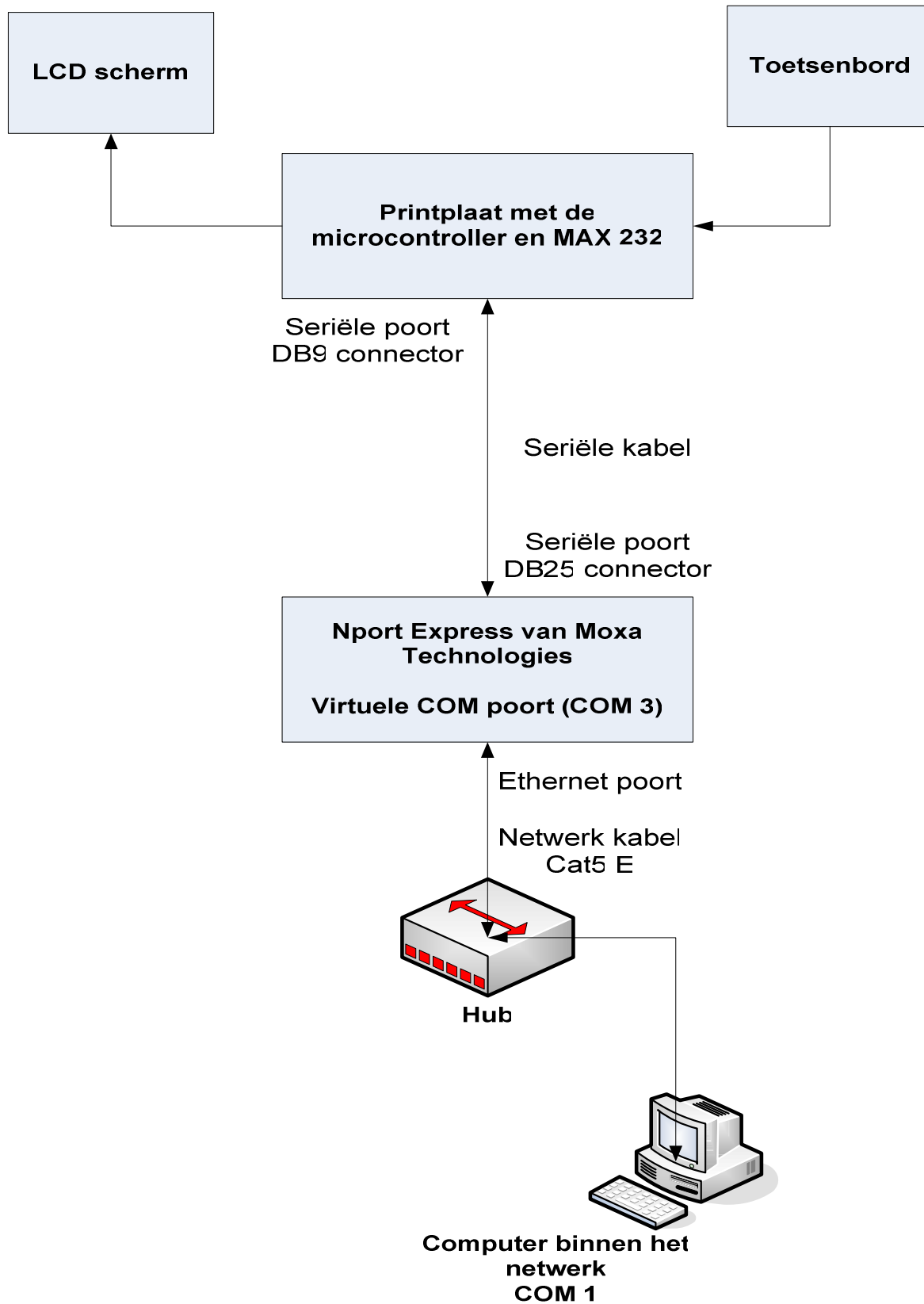
Doordat er berichten van een PC naar de Nport Express konden worden gestuurd en omgekeerd, kon men er zeker van zijn dat de Nport Express zonder problemen werkte.

3.2.2 Einde van de tweede fase

Na de testen tussen een PC binnen het netwerk en de Nport Express, besloot men de printplaat aan te sluiten op de seriële poort van de Nport Express. Wanneer men nu het programma startte in VB.NET, kreeg men de ingetoetste code te zien. Men probeerde eveneens een antwoord terug te sturen naar de printplaat (een nul of een één).

Wanneer de microcontroller op de printplaat een één ontving, verscheen op het LCD dat de code bevestigd was. Wanneer echter een nul naar de printplaat werd gestuurd, verscheen op het LCD de tekst 'foutieve code'. Hieruit kon men besluiten dat de Nport Express van Moxa Technologies succesvol geïntegreerd was in het eindwerk en er kon dankzij de Nport Express gecommuniceerd worden tussen de printplaat en een PC binnen het netwerk.

Op de volgende pagina ziet u de voorlopige architectuur van het eindwerk aan het einde van de tweede fase.



4. Derde fase: de Windows Service

In de derde fase was het de bedoeling de codes, die door de gebruiker werden ingetoetst en vervolgens doorgestuurd werden naar een computer, te gaan controleren in een databank, om daarop een antwoord terug te sturen naar de printplaat via de Nport. De vraag was nu hoe men de ingetoetste codes ging opvangen en controleren.

4.1 De Windows Service

4.1.1 Wat is een Windows Service?

Er werd eerst gedacht alles (het controleren van de codes, overzichten geven van al wie binnen geweest is, enz.) in één webapplicatie te doen. Dit bleek echter niet mogelijk, omdat men in een ASP webapplicatie een seriële poort niet kan blijven af luisteren.

Na enkele dagen naar een geschikte oplossing te hebben gezocht, kwam het idee om te gaan werken met een Windows Service.

Een Windows Service valt te vergelijken met een gewoon VB.NET programma. De enige verschillen zijn dat men bij een Windows Service geen gebruik kan maken van een visuele lay-out en dat het gemaakt is om in de achtergrond van het besturingssysteem te draaien. Het kan ook draaien zonder dat een gebruiker ingelogd heeft.

Dit was de ideale manier om de codes mee te controleren in de databank, aangezien een Windows Service wel de mogelijkheid biedt om een seriële poort (in ons geval de Nport) voortdurend af te luisteren.

4.1.2 Hoe een Windows Service maken en installeren?

Het maken van de Windows Service was niet erg ingewikkeld. Het programma Microsoft Visual Studio 2005 biedt bij het opstarten van het programma namelijk de mogelijkheid een Windows Service te creëren in plaats van een gewone Windows applicatie.

Als men ervoor gekozen heeft een Windows Service te creëren, kan men gewoon het programma schrijven in Visual Basic, zoals men gewend is bij het maken van een gewone Windows applicatie.

Het enige wat men dus niet kan is een mooie lay-out voorzien, maar aangezien Windows Services bedoeld zijn om in de achtergrond te draaien, is dit overbodig.

Nadat men de code van de Windows Service volledig geschreven heeft, dient men nog twee zaken toe te voegen aan de Windows Service. Als eerste moet men een ServiceInstaller en een ServiceProcessInstaller toevoegen. Deze twee installers maken het mogelijk de Windows Service te installeren op het besturingssysteem.

Voordat men echter kan overgaan tot de installatie van de service moet men eerst het volledige project bouwen, zodat men een exe file bekomt.

Zodra men de exe file heeft, kan men overgaan tot de installatie van de Windows Service. Dit doet men door gebruik te maken van de Microsoft Visual Studio 2005 Prompt. Binnen deze prompt gaat men naar de directory waar de .exe file van de Windows Service staat en vervolgens gebruikt men het commando 'installutil naam van de exe file.exe'. Door dit commando uit te voeren, zal de Windows Service op het besturingssysteem worden geïnstalleerd.

Wanneer men nu gaat kijken naar alle services die aanwezig zijn op het besturingssysteem, zal men de ontwikkelde service terug vinden tussen de andere services. Hierop kan men de service zelf starten. De service zal ook gestart worden telkens het besturingssysteem opgestart wordt.

4.1.3 De functies in de Windows Service

Wat voor functies werden geïmplementeerd in de Windows Service?

De eerste functie die geïmplementeerd werd in de service, was dat de virtuele com-poort (Nport) voortdurend moest worden afgeluisterd, om na te gaan of ze data ontvangen had van de printplaat. Indien er data ontvangen werd, moest worden nagegaan wat voor data dit was. Indien het een code was, moest deze vergeleken worden met de codes aanwezig in de tabel 'Gebruikers' van onze databank.

Er waren namelijk drie tabellen aangemaakt in onze Acces databank 'EHBsecurity'. Eén van deze drie was de tabel 'Gebruikers'. In deze tabel werden de voornaam, achternaam, code en een omschrijving van de gebruikers bijgehouden. Er werden twee kolommen bijgevoegd, één om de startdatum bij te houden en een andere kolom voor de einddatum. Deze twee kolommen gaven de periode aan waarbinnen de persoon binnen mocht in de beschermde ruimte. Deze start- en einddatum bevatten niet alleen een datum, maar ook een uur.

Indien de vergelijking in de tabel 'Gebruikers' een positief resultaat opleverde, werd er een positief antwoord gestuurd naar de microprocessor op de printplaat en mocht de persoon de beschermde ruimte binnen. In het andere geval werd een negatief antwoord gestuurd en werd aan de persoon gevraagd zijn code opnieuw in te geven of als het zijn derde poging was treedt het alarm in werking.

Een tweede functie was, dat telkens wanneer een persoon een juiste code ingaf de datum en het uur werden weggeschreven in de kolom 'uurbinnen' van de tabel 'Toegang'. Dit was één van de aangemaakte tabellen in de Acces databank. Deze tabel had als doel een overzicht te geven van de personen die de beschermde ruimte binnen gegaan waren. Deze tabel bestond uit zeven kolommen: voornaam, achternaam, code, datum, uurbinnen, uurbuiten en foto.

Een derde functie die aanwezig was in de service, was een functie om het uur weg te schrijven wanneer de persoon de beschermde ruimte verliet. Dit uur werd weggeschreven in de kolom 'uurbuiten' van de tabel 'Toegang'. Er werd vanuit gegaan dat een persoon nooit langer dan één dag in de beschermde ruimte kan zitten.

Een vierde en voorlopig laatste functie was een functie waarmee een e-mail werd gestuurd naar de administrator, indien het alarm was geactiveerd. Het alarm zal afgaan telkens een persoon de beschermde ruimte binnen wil en drie maal een foutieve code ingeeft. In geval van alarm moest deze vierde functie in werking treden en ervoor zorgen dat er een e-mail naar de administrator werd gestuurd. Via deze e-mail wist de administrator dat er alarm was en kon hij (in combinatie met de ASP webpagina; uitleg hierover volgt later) het alarm uitschakelen.

De ontwikkelde Windows Service bestond voorlopig uit de vier besproken functies.

4.1.4 Einde van de derde fase

In deze derde fase werd aldus een Windows Service ontwikkeld, die automatisch gestart werd door het besturingssysteem en op de achtergrond draaide. Hierdoor werd het mogelijk de ingevoerde codes te gaan controleren in een databank en de administrator te verwittigen in geval van alarm. Er was bijgevolg al een groot deel van het eindwerk ontwikkeld.

5. Vierde fase: de Internet camera

In de vierde fase moest de Internet camera in het eindwerk worden geïntegreerd.

5.1 De Internet camera

5.1.1 Problemen met de Internet camera

De Internet camera die ter beschikking werd gesteld was van het merk D-Link. Deze camera was een netwerk (Internet) camera, die moest worden voorzien van een eigen IP adres. In een eerste testfase bleek deze camera perfect te werken, maar na enkele weken konden er geen beelden meer worden gemaakt met de camera. De oorzaak hiervan werd gezocht en al snel kwam men ertoe dat er waarschijnlijk een probleem was met de lens.

Men moest dus gaan zoeken naar een ander middel om beelden (foto's) te maken. Daarom werd ervoor gekozen een webcamera te gebruiken in plaats van de Internet camera die niet meer werkte.

5.2 De webcam

5.2.1 De Quickcam Pro 5000 van Logitech



De webcam die gebruikt ging worden, was de Quickcam Pro 5000 van Logitech waarvan u hierboven een afbeelding ziet. Het verschil tussen deze webcam en de Internet camera was dat deze camera enkel en alleen aangesloten moest worden op de USB poort van de computer via de bijgeleverde USB kabel. Hier werd dus niet gewerkt met een IP-adres.

De vraag was nu wel hoe men deze webcam ging integreren in het eindwerk. Met andere woorden: hoe kon men ervoor zorgen dat de webcam aangesproken werd vanuit de Windows Service?

5.3 Windows Image Acquisition

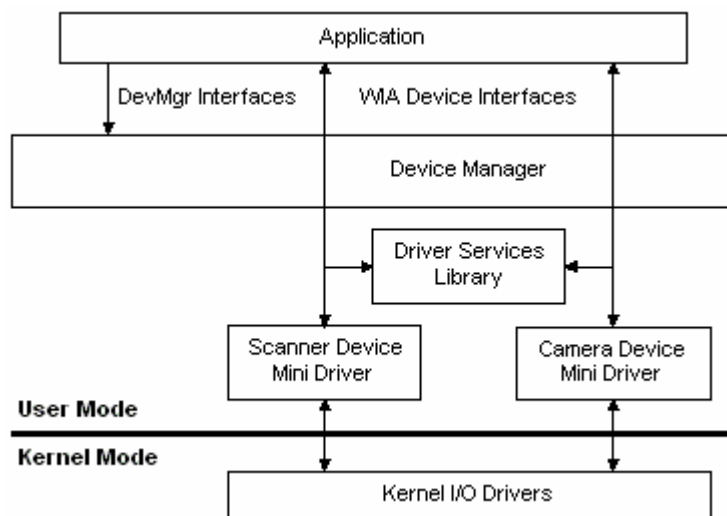
5.3.1 Wat is Windows Image Acquisition?

Windows Image Acquisition (WIA) is een API (Application Programming Interface) die geïntegreerd is in elk modern besturingssysteem van Windows. Deze API maakt het mogelijk dat programma's kunnen communiceren met bijvoorbeeld scanners, digitale camera's, enz.

WIA is opgebouwd uit drie hoofdcomponenten:

- 1) Een Device Manager die gaat zoeken naar aanwezige apparaten (scanners, camera's, enz.) en bovendien ook de eigenschappen van het apparaat ophaalt, 'events' creëert voor de apparaten en ten slotte de objecten voor de aanwezige apparaten aanmaakt.
- 2) Een Minidriver Service Library die al de services voor elk aanwezig apparaat afzonderlijk gaat implementeren.
- 3) Een Device Minidriver die de WIA eigenschappen en commando's naar het specifieke apparaat mapt.

Hieronder wordt de WIA architectuur getoond:



5.3.2 WIA gebruiken in VB.NET

WIA was dus de ideale oplossing om onze webcamera te kunnen aanspreken en hem foto's te laten nemen. Nu moesten we enkel nog de link leggen tussen WIA en onze Windows Service.

Dit kon perfect gebeuren door de library 'Interop.WIA' toe te voegen aan onze Windows Service. Deze library zit niet in de standaard libraries van VB.NET, maar is wel heel snel te vinden op het Internet en gratis te downloaden.

Nadat de 'Interop.WIA' library aan de Windows Service was toegevoegd, konden we in deze service de webcamera aanspreken door gebruik te maken een WIA object dat men creëerde.

Op dit WIA object werd dan de methode 'TakePicture' aangeroepen. En zo kon men een foto nemen met de webcam van de Windows Service. Dit resulteerde in een extra functie voor onze Windows Service.

5.3.3 Extra functie voor de Windows Service

Eén van de vereisten binnen het eindwerk was dat er een foto genomen werd van de persoon die de beschermde ruimte betreden had. Om dit te realiseren moest er dus een extra functie aangemaakt worden binnen de Windows Service.

Deze functie moest er voor zorgen dat aan de webcam de opdracht werd gegeven een foto te maken zodra iemand de beschermde ruimte binnen ging. Deze foto diende te worden bewaard in de databank.

De foto's moesten bijgehouden worden in de kolom 'foto' van de tabel 'Toegang'. Deze foto's worden niet rechtstreeks (in binaire vorm) opgeslagen in de databank, maar bewaard in een map op de harde schijf; de locatie wordt telkens opgeslagen in de kolom 'foto' van de tabel 'Toegang'.

Bij het leegmaken van de tabel 'Toegang' mag men bijgevolg niet vergeten de map, waar de foto's opgeslagen worden, ook leeg te maken.

5.3.4 Einde van de vierde fase

Ondanks de problemen met de Internet camera is men er in deze vierde fase toch in geslaagd een foto te maken van de personen die de beschermde ruimte binnen komen door gebruik te maken van een webcam op USB.

6. Vijfde fase: de webapplicatie

De vijfde en laatste fase van de ontwikkeling van het eindwerk had als doel een webapplicatie te ontwikkelen.

Deze webapplicatie moest een overzicht kunnen geven van de personen die de beschermde ruimte binnen gegaan zijn. Daarnaast diende het ook mogelijk te zijn met de webapplicatie personen in te schrijven in de databank, zodat deze toegang zouden hebben tot de beschermde ruimte. De laatste functie van de webapplicatie is dat men het alarm kan uitschakelen. De webapplicatie is ontwikkeld voor de administrator van de ruimte.

De screenshots van de verschillende webpagina's van de webapplicatie kan u vinden in de bijlagen (Bijlage 4: Screenshots webapplicatie EHB Security System).

6.1 EHB Security System

6.1.1 Hoe webapplicatie ontwikkelen?

Aangezien er al enige kennis van ASP.NET aanwezig was, werd besloten een webapplicatie te ontwikkelen in ASP.Net. Webapplicaties in ASP.NET konden ook in Microsoft Visual Studio 2005 worden gemaakt. De webapplicatie kreeg de toepasselijke naam 'EHB Security System'.

6.1.2 De verschillende webpagina's

6.1.2.1 De Login pagina

De eerste pagina die men te zien zal krijgen wanneer men naar de webapplicatie surft, is de login pagina.

Hier is het de bedoeling dat de persoon die verantwoordelijk is voor de beschermde ruimte zijn gebruikersnaam en paswoord invult. Deze zullen gecontroleerd worden. Indien de gebruikersnaam of het paswoord niet correct zijn, krijgt men een foutmelding te zien en moet men het correcte paswoord of gebruikersnaam opnieuw ingeven. Als de gebruikersnaam en het paswoord beiden correct zijn, komt men op de volgende pagina terecht: de Main Menu pagina.

Het paswoord en de gebruikersnaam zitten in de code verwerkt. Met andere woorden: als men de gebruikersnaam of het paswoord wil wijzigen kan men dit enkel doen in de code zelf.

6.1.2.2 De Main Menu pagina

Nadat men heeft ingelogd, komt men op de Main Menu pagina terecht. Hier kan men kiezen wat men precies wil gaan doen:

- 1) Iets wijzigen in verband met de gebruikers, personen die toegang hebben tot de beschermde ruimte.
- 2) Een overzicht krijgen van iedereen die de beschermde ruimte binnen gegaan is.
- 3) Het e-mailadres aanpassen waarnaar een e-mail moet worden gestuurd in geval van alarm.

Afhankelijk van welke keuze men maakt, komt men op een volgende pagina terecht.

6.1.2.3 De Gebruikers pagina

Als men er voor kiest iets te wijzigen in verband met de gebruikers, komt men op de Gebruikerspagina terecht. Ook hier kan men kiezen wat men wil gaan doen:

- 1) Een overzicht krijgen van de gebruikers die toegang hebben tot de beschermde ruimte.
- 2) Een gebruiker toevoegen.
- 3) Een gebruiker verwijderen (deleten).
- 4) Een gebruiker aanpassen.

Afhankelijk van welke keuze men maakt, komt men op een volgende pagina terecht.

6.1.2.4 De Overzicht Gebruikers pagina

Als men een ervoor kiest een overzicht te krijgen van alle personen die toegang hebben tot de beschermde ruimte, krijgt men tabel te zien.

In deze tabel zijn de voornaam, achternaam en code van de persoon te zien. Ook wordt de startdatum en einddatum aangegeven. Deze begrenzen de periode waarbinnen de persoon in kwestie toegang heeft tot de beschermde ruimte. Als laatste is een beschrijving te zien. Deze beschrijving geeft de functie aan van de persoon en/of waarom hij toegang krijgt tot de beschermde ruimte.

De gegevens die in de tabel te zien zijn, worden opgehaald uit de tabel 'Gebruikers' in onze Acces databank (EHB security).

Onder aan de pagina staan nog twee knoppen, één om terug te keren naar de Main Menu pagina en een andere om terug te keren naar de Gebruikers pagina.

6.1.2.5 De Nieuwe Gebruiker pagina

De Nieuwe Gebruiker pagina krijgt men te zien wanneer men op de Gebruiker pagina ervoor kiest om een gebruiker toe te voegen.

Met deze pagina kan men een nieuwe gebruiker toevoegen aan de databank, zodat deze persoon ook toegang krijgt tot de beschermde ruimte. Vooraleer men een nieuwe gebruiker kan toevoegen, moeten er wel enkele zaken ingevuld worden.

Eerst moet men de voornaam en achternaam van de persoon ingeven. Daarna moet men de persoon een code van vier cijfers geven. Bij het bepalen van deze code kan men ervoor kiezen om zelf een code in te geven; er zal meteen gecontroleerd worden of deze code al aanwezig is in de databank, aangezien er geen twee gebruikers aanwezig mogen zijn met dezelfde code. Indien deze reeds aanwezig is, zal een foutboodschap worden getoond. Men kan eveneens op de knop 'Random Code' klikken en dan wordt de code willekeurig bepaald. Deze code is dan wel meteen uniek.

Nadat men de voornaam, achternaam en code ingegeven heeft, kan men een start- en einddatum voor de nieuwe gebruiker instellen. Dit kan men doen door eerst een datum te kiezen in de kalender en dan het uur, de minuten en de seconden te bepalen. Daarna klikt men op de knop 'startdatum' of 'einddatum'.

Indien men een startdatum ingeeft, is men verplicht ook een einddatum mee te geven. Men kan er ook voor kiezen geen start- en einddatum mee te geven. Dit wil dan zeggen dat de nieuwe gebruiker altijd toegang heeft tot de beschermde ruimte.

Het laatste dat men verplicht moet ingeven is een beschrijving. Wanneer alles is ingevuld kan men ook nog een e-mailadres ingeven. Dit moet het e-mailadres zijn van de nieuwe gebruiker. Nadat men ook dit gedaan heeft, klikt men op de knop 'Voeg Gebruiker Toe'. Als alles correct is ingevuld, wordt een nieuwe gebruiker aangemaakt in de tabel 'Gebruikers' in onze databank en krijgt men de boodschap te zien 'Gebruiker Toegevoegd'. Indien iets niet correct is ingevuld zal men een foutboodschap te zien krijgen.

Nadat de nieuwe gebruiker is toegevoegd aan de databank zal er een e-mail gestuurd worden naar de nieuwe gebruiker. In deze e-mail staat dan van wanneer tot wanneer hij toegang heeft tot de beschermde ruimte en welke code hij zal moeten gebruiken om de beschermde ruimte te betreden.

Onder aan de pagina staan nog twee knoppen, één om terug te keren naar de Main Menu pagina en een andere om terug te keren naar de Gebruikers pagina.

6.1.2.6 De Delete Gebruikers pagina

De Delete Gebruikers pagina is de pagina die men zal te zien krijgen als men ervoor kiest om een gebruiker te verwijderen.

Op deze pagina zal men in tabelvorm een overzicht krijgen van alle gebruikers die toegang hebben tot de beschermde ruimte. Deze gegevens worden opgehaald uit de tabel 'Gebruikers' in de databank. In de laatste kolom van de tabel zal voor elke rij telkens een checkbox te zien zijn.

Rechts onderaan in de tabel staat een knop 'Delete'. Met deze knop zullen de gebruikers verwijderd worden. Dit gebeurt echter alleen indien er gebruikers aangevinkt zijn in de tabel. Als dit het geval is, zullen de aangevinkte gebruikers uit de tabel 'Gebruikers' in de databank worden verwijderd, nadat er op de 'Delete' knop geklikt wordt. Als men op de 'Delete' knop klikt zonder dat er gebruikers geselecteerd zijn in de tabel, dan zal men een foutboodschap te zien krijgen.

Onder aan de pagina staan nog twee knoppen, één om terug te keren naar de Main Menu pagina en een andere om terug te keren naar de Gebruikers pagina.

6.1.2.7 De Pas Gebruiker Aan pagina

Op deze pagina zal men terecht komen nadat men op de Gebruikers pagina heeft gekozen een gebruiker aan te passen.

Er zal in tabelvorm een overzicht worden getoond van alle gebruikers die toegang hebben tot de beschermde ruimte. Deze gegevens worden opgehaald uit de tabel 'Gebruikers' in de databank. In de laatste kolom van de tabel zal voor elke rij telkens een checkbox te zien zijn.

Onderaan rechts in de tabel staat een knop 'Pas Aan'. Men moet eerst de gebruiker kiezen die men wenst aan te passen. Dit doet men door op de checkbox te klikken die overeen komt met de gewenste gebruiker. Nadat dit gebeurd is, moet men op de knop 'Pas Aan' klikken.

Door op deze knop te klikken krijgt men dan de gegevens (voornaam, achternaam, enz.) van de gekozen gebruiker te zien. Telkens zal naast deze gegevens een vakje verschijnen waarin men een nieuw gegeven kan ingeven voor de gebruiker. Dit is niet zo voor de code, want men kan een bestaande gebruiker zijn code niet veranderen.

Ook hier geldt weer de regel dat als er een startdatum ingegeven is, er ook een einddatum ingegeven moet worden (en omgekeerd).

Men is niet verplicht in alle vakjes een nieuwe waarde in te geven. Enkel die vakjes waar men de bestaande gebruiker een nieuw gegeven wil ingeven moeten worden ingevuld (bijvoorbeeld start- en einddatum).

Men kan eveneens opnieuw een e-mailadres invullen. Er zal dan een e-mail gestuurd worden naar de bestaande gebruiker nadat men op de knop 'Pas Gebruiker Aan' geklikt heeft. In deze e-mail zal dan staan dat de gegevens aangepast werden en zullen zijn gegevens (startdatum, einddatum en code) te zien zijn.

Door op de knop 'Pas Gebruiker Aan' te klikken zullen ook de gegevens in de tabel 'Gebruikers' in de database aangepast worden. Automatisch zal men dan op de Overzicht Gebruikers pagina terecht komen, waar men zal zien dat de gegevens van de gekozen gebruiker aangepast zijn.

6.1.2.8 De Toegang pagina

De Toegang pagina is de pagina die men zal te zien krijgen indien men op de Main Menu pagina kiest om een overzicht te krijgen van al de personen die de beschermde ruimte binnen gegaan zijn.

In deze pagina is een zoekmechanisme verwerkt. Dit zoekmechanisme heeft als doel dat er sneller kan gekeken worden of een bepaalde persoon binnen is geweest in de beschermde ruimte.

Zo zijn er vier verschillende manieren om te zoeken:

- 1) Op voornaam
- 2) Op achternaam
- 3) Op code
- 4) Op periode (dan geeft men twee datums op waartussen gezocht moet worden.)

Als men gebruik maakt van het zoekmechanisme worden enkel die gegevens uit de tabel 'Toegang' in de databank opgehaald, die overeenkomen met de opgegeven waarde (voornaam, achternaam of code) waarop gezocht moest worden.

Indien er overeenstemmende gegevens gevonden worden, zal een tabel verschijnen waarin de voornaam, achternaam, code, datum, uur dat de persoon binnen gegaan is, uur dat de persoon buiten gegaan is en als laatste de genomen foto getoond worden. Als er geen overeenstemmende gegevens gevonden worden, zal een foutboodschap worden getoond.

Als men op periode zoekt, zullen enkel die personen worden getoond waarvan het uur, dat ze de beschermde ruimte binnen gegaan ligt tussen de aangegeven periode.

Men heeft ook de mogelijkheid om niet met het zoekmechanisme te werken door op de knop 'Volledig Overzicht' te klikken. Dan worden alle gebruikers getoond die de beschermde ruimte binnen gegaan zijn.

Onder aan de pagina staat nog een knop 'Terug naar Homepage' om terug te keren naar de Main Menu pagina.

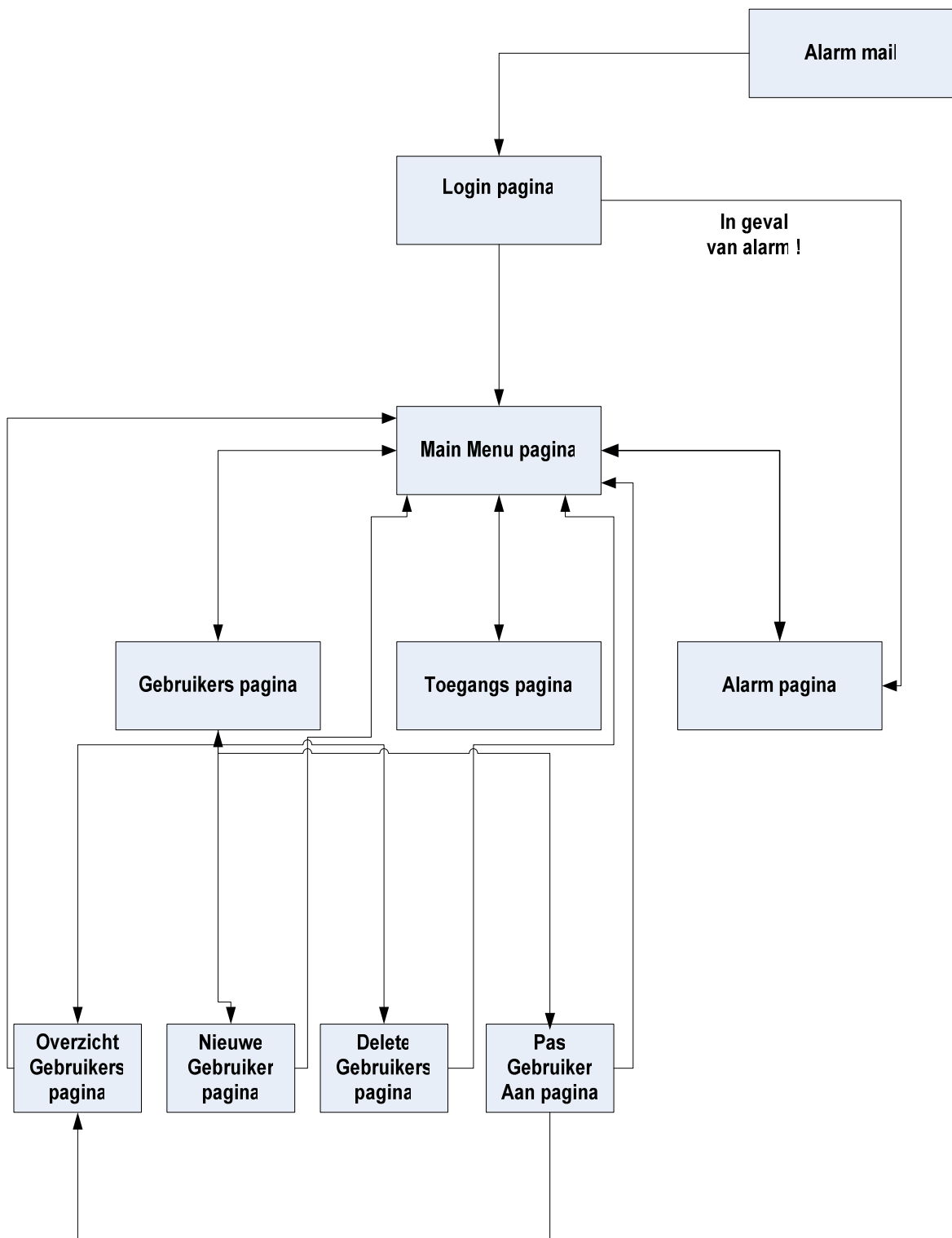
6.1.2.9 De Alarm pagina

De Alarm pagina is de pagina die men zal te zien krijgen wanneer men er op de Main Menu pagina voor kiest het e-mailadres aan te passen waarnaar een e-mail gestuurd dient te worden in geval van alarm.

Op deze pagina zal men dus het e-mailadres van de administrator kunnen veranderen. Deze Alarm pagina zal echter ook gebruikt worden om het alarm uit te schakelen in geval van alarm. Want in de e-mail die naar de persoon die verantwoordelijk is voor de beschermde ruimte gestuurd wordt, staat een link naar de Alarm pagina. Indien op deze link geklikt wordt, zal men eerst moeten inloggen, waarna men rechtstreeks op de Alarm pagina terechtkomt. De Alarm pagina zal dan zo opgebouwd worden dat men het alarm kan uitschakelen door op de 'Alarm uitschakelen' te klikken.

Onder aan de pagina staat nog een knop 'Terug naar de Homepage' waarmee men terug kan keren naar de Main Menu pagina.

6.1.3 Het navigatieschema



6.1.4 Online plaatsen van de webapplicatie

Toen de webapplicatie volledig afgewerkt was, moest deze nog online geplaatst worden. Daarvoor werd gebruik gemaakt van IIS (Internet Information Services). IIS is zeer gebruiksvriendelijk en ideaal om een webapplicatie online te plaatsen binnen een lokaal netwerk.

Dit betekende wel dat de website enkel bereikbaar was vanuit het lokaal netwerk. Indien men er van op een willekeurige PC wenst aan te kunnen, moet men de webapplicatie binnen aangekochte vrije webruimte plaatsen.

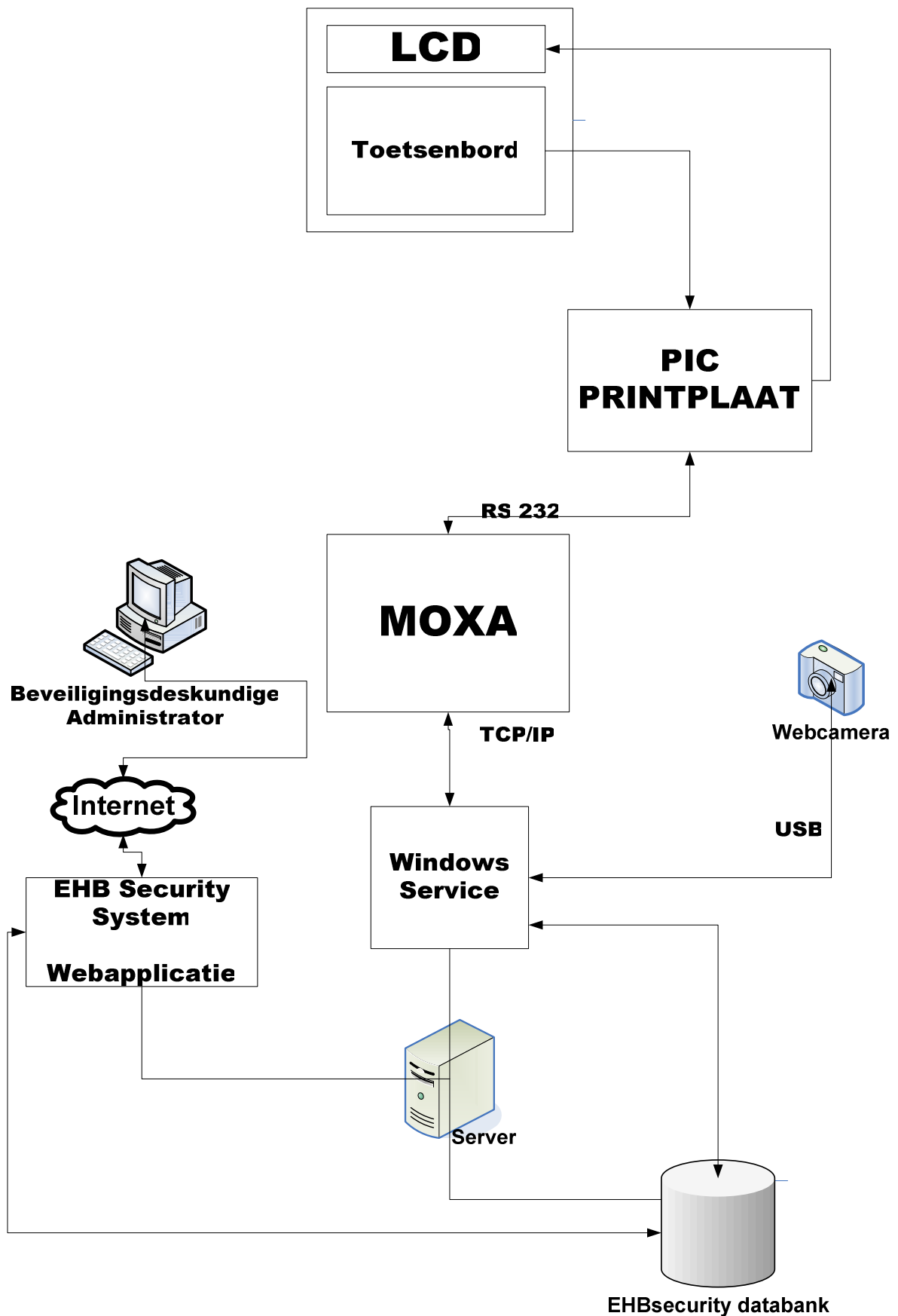
Aangezien dit voor het eindwerk onnodige kosten zouden zijn, werd dit niet gedaan.

6.1.5 Einde van de vijfde fase

Nadat de vijfde en laatste fase achter de rug was, mocht men zeggen dat de ontwikkeling van het eindwerk ten einde was. In deze laatste fase werd het doel bereikt een soort van controle middel te hebben over de beschermde ruimte. Dit controlemiddel moest van op afstand te gebruiken zijn.

Aan deze voorwaarden voldoet de ontwikkelde webapplicatie zeker, want de website is een goede manier om controle te krijgen over de beschermde ruimte en bovendien kan het van op elke PC (binnen het netwerk) worden aangesproken.

7. Architectuur van het eindresultaat



Het besluit

Na een periode van vier maanden kunnen we toch tevreden zijn over het resultaat. Indien men gaat kijken naar de vooropgestelde doelen, kan besloten worden dat deze allemaal gerealiseerd werden.

De ontwikkeling van het eindwerk is zeker niet zonder problemen verlopen. Zo was het falen van de eerste printplaat een moment waarop veel tijd is verloren gegaan. Dit is achteraf ook logisch te verklaren, aangezien de printplaat de basis moest vormen van het project.

Indien men een vergelijking gaat maken tussen de vooropgestelde architectuur van het eindwerk en de architectuur van het eindresultaat, zijn er volgende belangrijke veranderingen.

De eerste verandering is dat men in de vooropgestelde architectuur opteerde te werken met een webapplicatie die ook nog eens de printplaat moest sturen (codes gaan controleren, enz.). In de architectuur van het eindresultaat valt op dat dit opgelost is door wel een webapplicatie te ontwikkelen, maar daarnaast ook een Windows Service te ontwikkelen.

Het voordeel van deze Windows Service is dat deze voortdurend op de achtergrond kan draaien en zo de besturing van de printplaat voor zijn rekening kan nemen. De functies binnen deze Windows Service werden dan wel meer uitgebreid (sturen van e-mail) dan oorspronkelijk besproken was.

Het tweede grote verschil is dat er geen Internet camera gebruikt werd. De verklaring hiervoor is dat er problemen waren met de Internet camera, die plots niet meer werkte. Dit moest echter op de één of andere manier toch gerealiseerd worden, aangezien het nemen van een foto van de degene die de beschermde ruimte betreedt een vereiste was.

Men is hierin geslaagd door gebruik te maken van een webcam. Aangezien deze webcam net dezelfde functie had als de Internet camera, kon deze perfect gebruikt worden.

Indien men mij zou vragen wat ik allemaal heb bijgeleerd bij uitwerken van dit eindwerk, kan ik zeggen dat ik erg veel heb bijgeleerd op verschillende vlakken. Zo heb ik meer kennis opgedaan in verband met het ontwikkelen van een printplaat en het ontwerpen van een elektronische schakeling. Daarnaast is mijn kennis van VB.NET ook weer uitgebreid, aangezien ik nog nooit gehoord had van een Windows Service; nu heb ik zelfs geleerd deze zelf te ontwikkelen. Zo kan ik nog vele andere voorbeelden vinden, maar alles wat ik geleerd heb opsommen is onbegonnen werk.

Om af te sluiten kan ik terugkijken op een geslaagde ontwikkeling van het eindwerk.

Het dankwoord

Ik zou nog een aantal personen willen bedanken, die hebben bijgedragen aan een succesvolle ontwikkeling van het eindwerk.

In de eerste plaats zou ik mijn promotor, de heer Toon Blondeel, willen bedanken. Hij was diegene die het idee, achter het eindwerk, ontwikkelde en altijd klaar stond om mij te helpen als er problemen waren of als ik vragen had.

Ik zou ook nog mevrouw Gitte Vermeiren willen bedanken. Wanneer er vragen waren in verband met VB.NET, kon ik steeds bij haar terecht. Het is mede dankzij haar dat ik op het spoor ben gekomen van de Windows Service.

Ook mevrouw Liliane van Eyck zou ik willen bedanken, aangezien zij de evolutie van mijn eindwerk perfect opgevolgd heeft.

Als allerlaatste zou ik ook mijn medestudenten willen bedanken, want ook bij hen kon ik terecht als er een probleem was.

De literatuurlijst

Internetsites:

- www.circuitsonline.net
- www.melabs.com/products/pbp.htm
- www.picbasic.nl
- www.thouters.be/downloads/thesis.pdf
- www.moxa.com
- www.msdn.com
- www.msdn2.com
- www.wikipedia.org

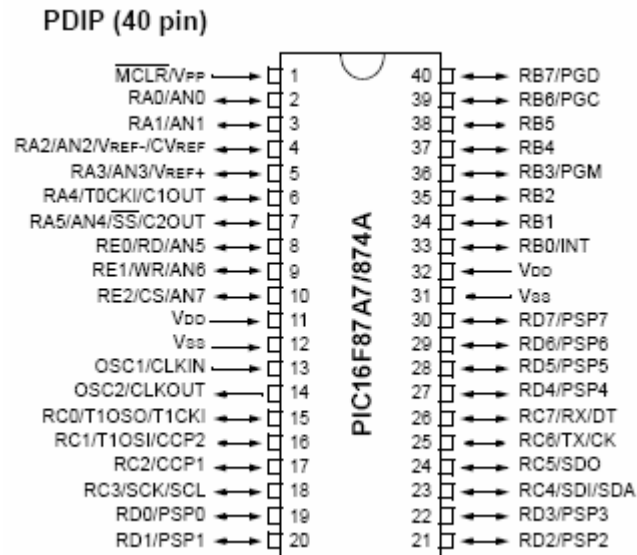
Boeken en tijdschriften:

- pic basic pro manual
- pic basic pro course
- Windows Services en .NET het wordt alleen maar beter | Jan Willem Overbeek| .NET magazine #4 | december 2003

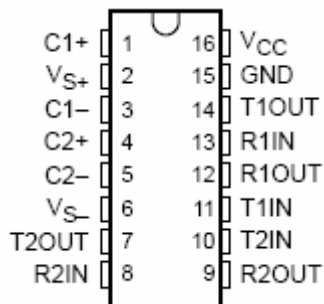
Bijlagen

Bijlage 1: Pin diagram PIC 16F877A + MAX 232

PIC 16F877A



MAX 232



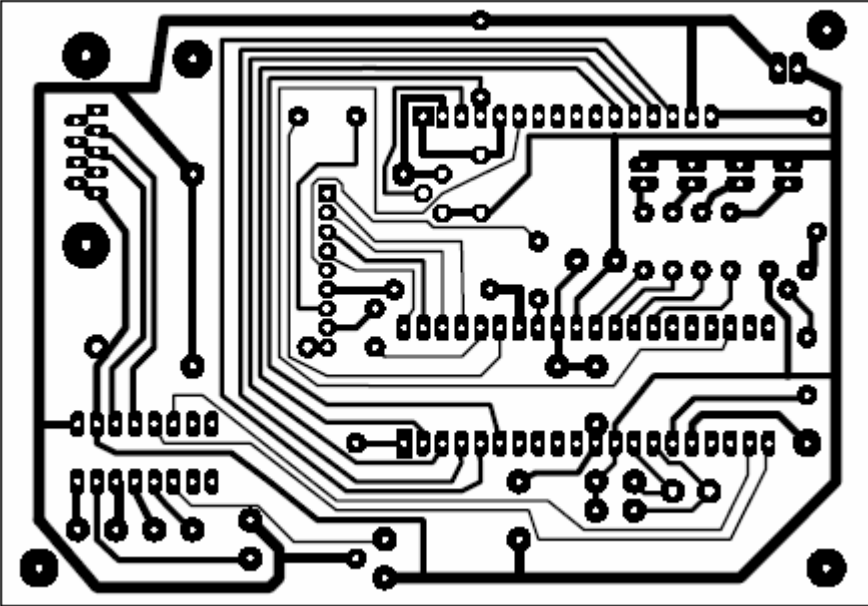
Bijlage 2 : PIC Basic Pro code fragment : Toetsenbord uitlezen

```
loop:
  gosub getkey
send:
  goto loop
getkey:
  pause 50
getkeyu:
  PORTB = 0
  TRISB = $f0
  if((PORTB >> 4)!=$f) then getkeyu
  pause 50
getkeyp:
  for row = 0 to 3
    PORTB = 0
    TRISB = (dcd row)^ $ff
    col = PORTB >> 4
    if col != $f then gotkey
  next row
  goto getkeyp

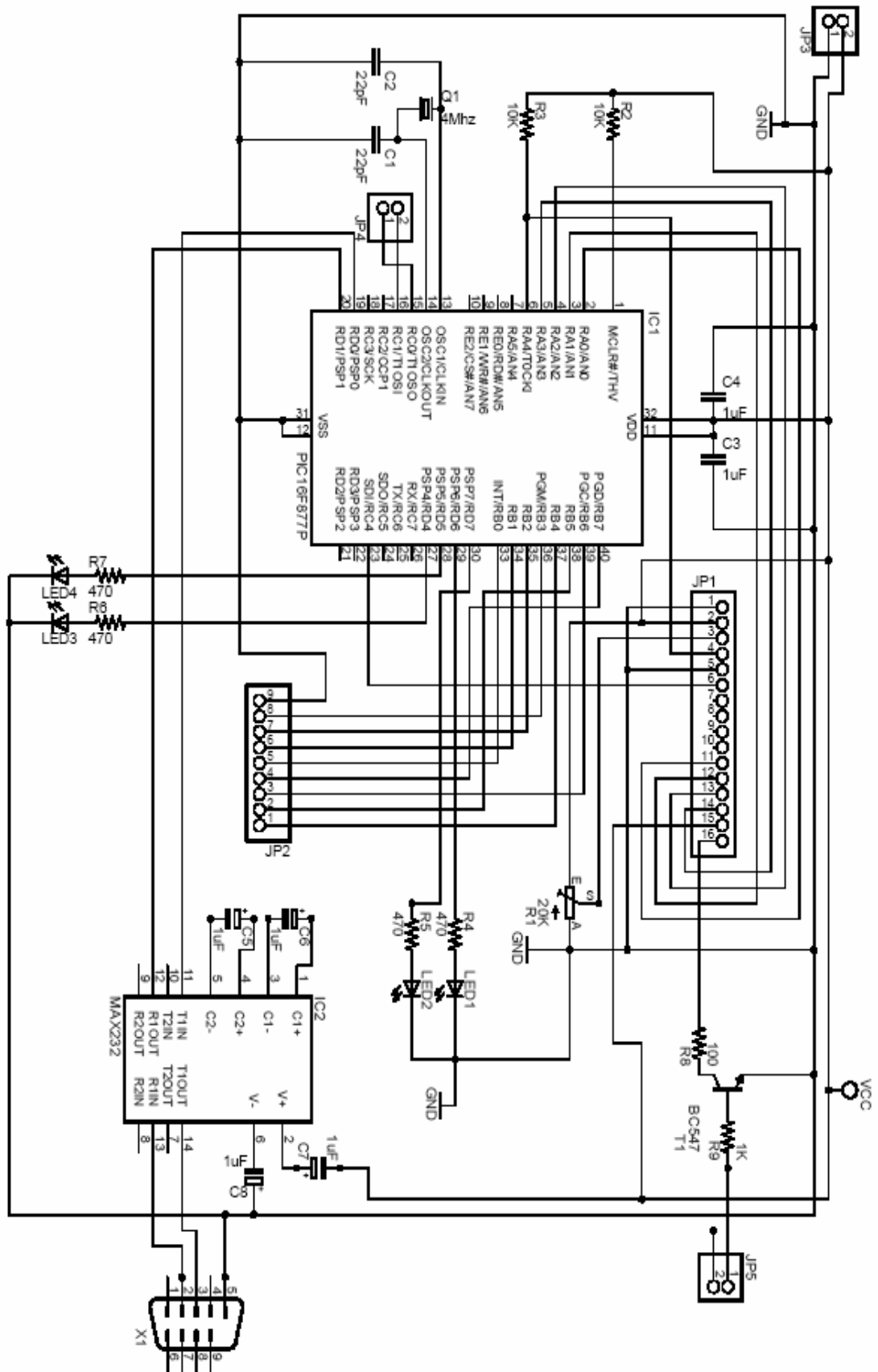
gotkey:
  key = (row *4) + (ncd (col ^ $f))
  codeingeven = 1
```

Bijlage 3 : Schakelingen printplaat

Schakeling op de printplaat

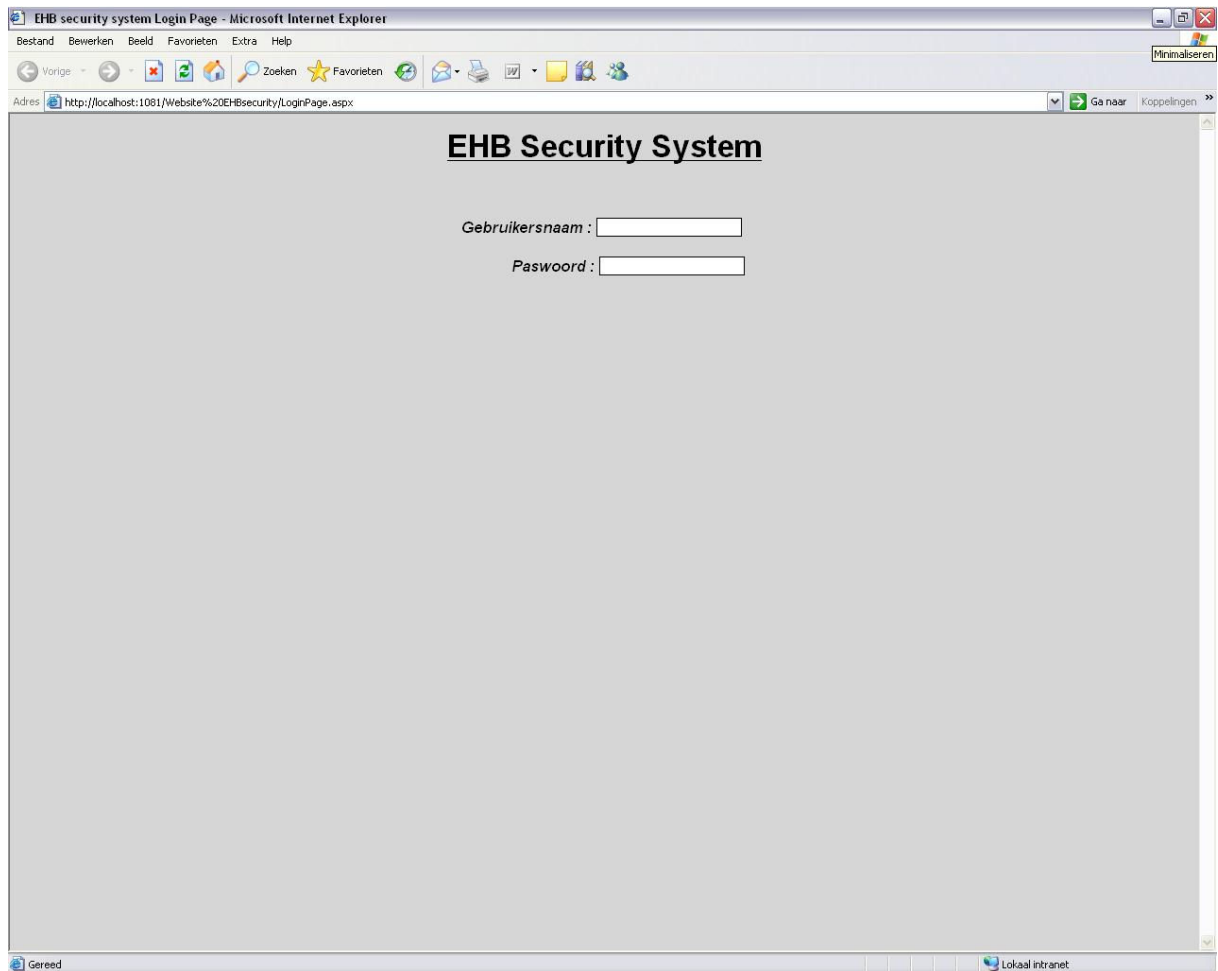


Elektronische schakeling

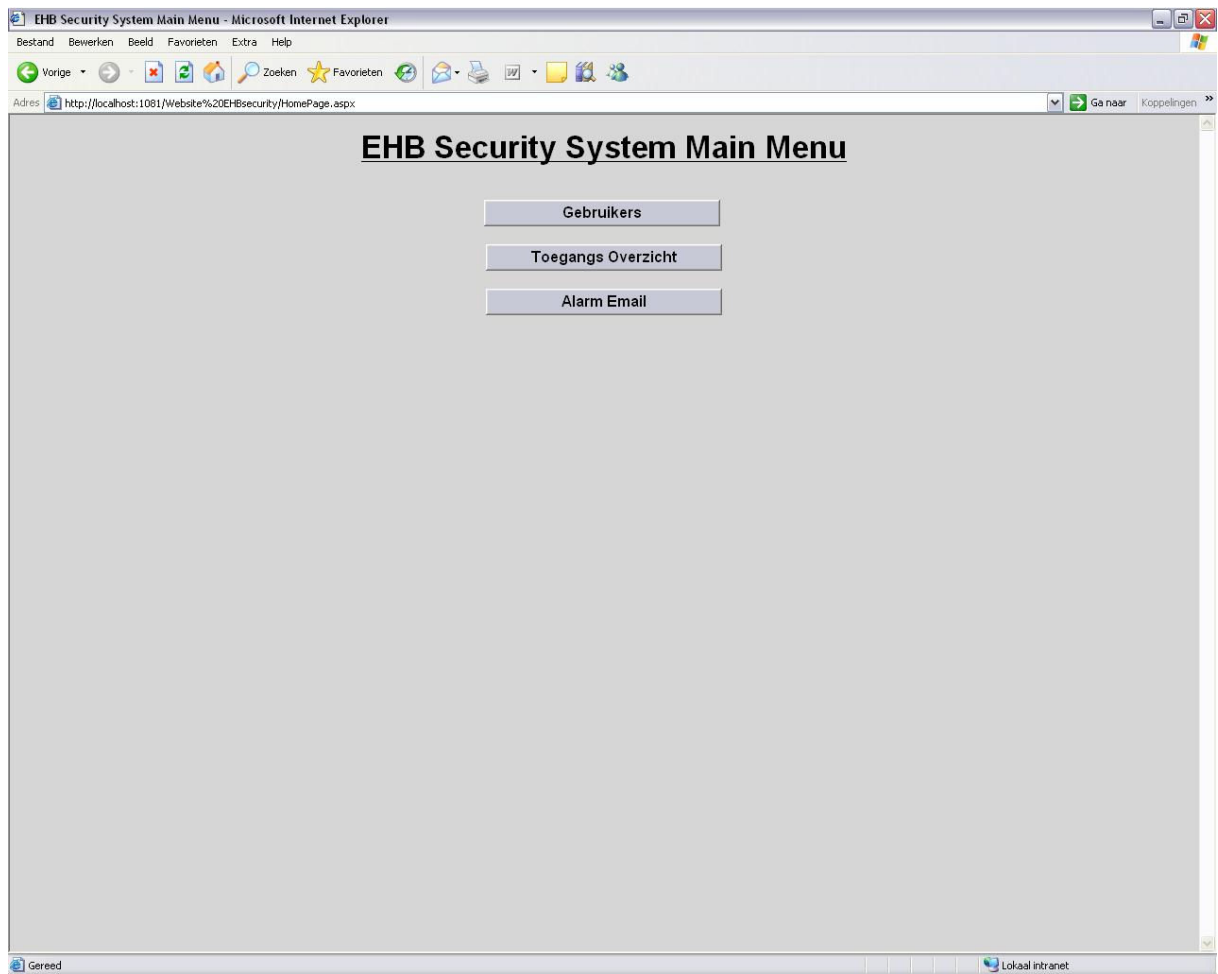


Bijlage 4 : Screenshots webapplicatie EHB Security System

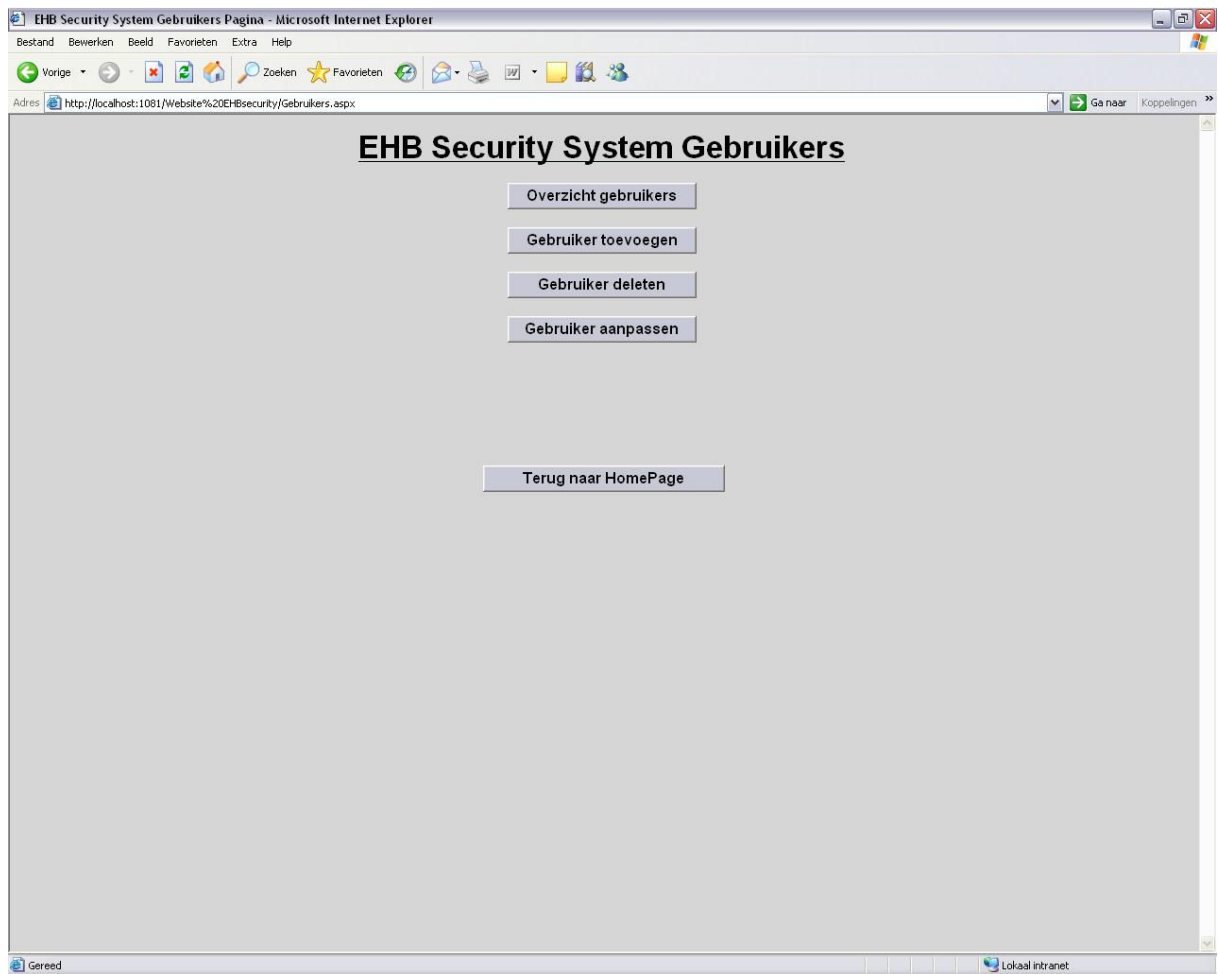
De Login pagina



De Main Menu pagina



De Gebruikers pagina



De Overzicht Gebruikers pagina

The screenshot shows a Microsoft Internet Explorer browser window titled 'Overzicht Gebruikers Pagina - Microsoft Internet Explorer'. The address bar shows the URL: <http://localhost:1081/Website%20EHBsecurity/Overzichtgebruikers.aspx>. The main content area features the heading 'EHB Security System Overzicht Gebruikers' and a table with the following data:

Voornaam	Achternaam	Code	Start	Einde	Beschrijving
Sigurd	Geerts	1234	15/05/2007 13:00:00	15/05/2007 23:00:00	EHB
Marjolein	Bresseleers	4567	15/05/2007 13:00:00	15/05/2007 21:00:00	KUL
Julien	GEERTS	0145			Tester
freddy	geerts	0143	16/05/2007 19:00:00	16/05/2200 19:00:00	tester

Below the table, there are two buttons: 'Terug naar HomePage' and 'Terug naar Gebruikers'. The browser's status bar at the bottom shows 'Gereed' and 'Lokaal intranet'.

De Nieuwe Gebruiker pagina

Nieuwe Gebruiker Pagina - Microsoft Internet Explorer

Bestand Bewerken Beeld Favorieten Extra Help

Vorige Zoeken Favorieten

Adres <http://localhost:1081/Website%20EHBsecurity/Nieuwegebruiker.aspx> Ga naar Koppelingen

EHB Security System Nieuwe Gebruiker

Om de Startdatum of Einddatum te bepalen, kies eerst een datum in de kalender en bepaal dan de tijd door uren, minuten en seconden te bepalen. Druk daarna op de Startdatum of Einddatum knop!

mei 2007						
ma	di	wo	do	vr	za	zo
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Gekozen Datum

Uren Minuten Seconden

Voornaam (dd/mm/yyyy hh:mm:ss) Startdatum

Achternaam (dd/mm/yyyy hh:mm:ss) Einddatum

Code Beschrijving

Emailadress

Gereed Lokaal intranet

Delete Gebruikers pagina

EHB Security System Delete Gebruikers

Voornaam	Achternaam	Code	Start	Einde	Beschrijving	
Sigurd	Geerts	1234	15/05/2007 13:00:00	15/05/2007 23:00:00	EHB	<input type="checkbox"/>
Marjolein	Bresseleers	4567	15/05/2007 13:00:00	15/05/2007 21:00:00	KUL	<input type="checkbox"/>
Julien	GEERTS	0145			Tester	<input type="checkbox"/>
freddy	geerts	0143	16/05/2007 19:00:00	16/05/2200 19:00:00	tester	<input type="checkbox"/>

De Pas Gebruiker Aan pagina (1)

EHB Security System Pas Gebruiker Aan

Voornaam	Achternaam	Code	Start	Einde	Beschrijving	
Sigurd	Geerts	1234	15/05/2007 13:00:00	15/05/2007 23:00:00	EHB	<input type="checkbox"/>
Marjolein	Bresseleers	4567	15/05/2007 13:00:00	15/05/2007 21:00:00	KUL	<input type="checkbox"/>
Julien	GEERTS	0145			Tester	<input type="checkbox"/>
freddy	geerts	0143	16/05/2007 19:00:00	16/05/2200 19:00:00	tester	<input type="checkbox"/>

[Pas aan](#)

[Terug naar HomePage](#) [Terug naar Gebruikers](#)

De Pas Gebruiker Aan pagina (2)

Pas Gebruiker Aan Pagina - Microsoft Internet Explorer

Bestand Bewerken Beeld Favorieten Extra Help

Vorige Zoeken Favorieten Ga naar Koppelingen

Adres <http://localhost:1081/Website%20EHBSecurity/Pasaanindatabank.aspx>

EHB Security System Pas Gebruiker Aan

Voornaam: Sigurd	Nieuwe Voornaam: <input type="text"/>
Achternaam: Geerts	Nieuwe Achternaam: <input type="text"/>
Code: 1234	
Startdatum: 15/05/2007 13:00:00 (dd/mm/yyyy hh:mm:ss)	Nieuwe Startdatum: <input type="text"/>
Einddatum: 15/05/2007 23:00:00 (dd/mm/yyyy hh:mm:ss)	Nieuwe Einddatum: <input type="text"/>
Beschrijving: EHB	Nieuwe Beschrijving: <input type="text"/>
Emailadress: <input type="text"/>	

Gereed Lokaal intranet

De Toegangs pagina (1)

EHB Security System Toegangs Pagina - Microsoft Internet Explorer

Bestand Bewerken Beeld Favorieten Extra Help

Vorige Zoeken Favorieten

Adres <http://localhost:1081/Website%20EHBSecurity/Toegang.aspx> Ga naar Koppelingen

EHB Security System Toegang

Om te zoeken op periode : kies eerst een datum in de kalender en bepaal dan de tijd door het uur, de minuten en de seconden te bepalen. Druk daarna op de Van of Tot knop !

Voornaam:

Achternaam:

Code:

< mei 2007 >

ma	di	wo	do	vr	za	zo
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Uren:

Minuten: Seconden:

Van: Tot:

(dd/mm/yyyy hh:mm:ss) (dd/mm/yyyy hh:mm:ss)

Lokaal intranet

De Toegangspagina (2)

Microsoft Internet Explorer window: EHB Security System Toegangspagina

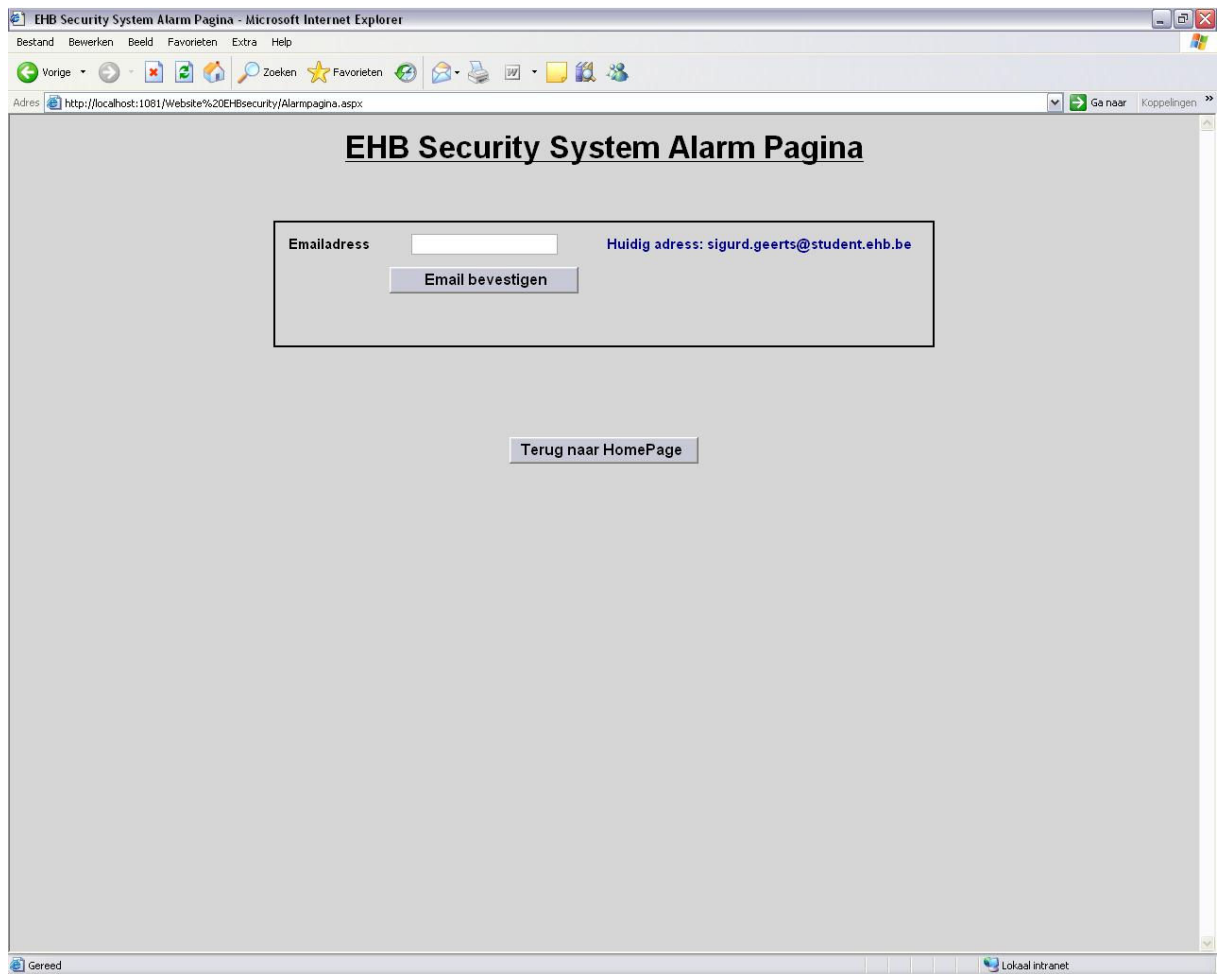
Adres: <http://localhost:1081/Website%20EHBsecurity/Toegang.aspx>

Buttons: Zoek op periode, Volledig Overzicht, Ga naar, Koppelingen

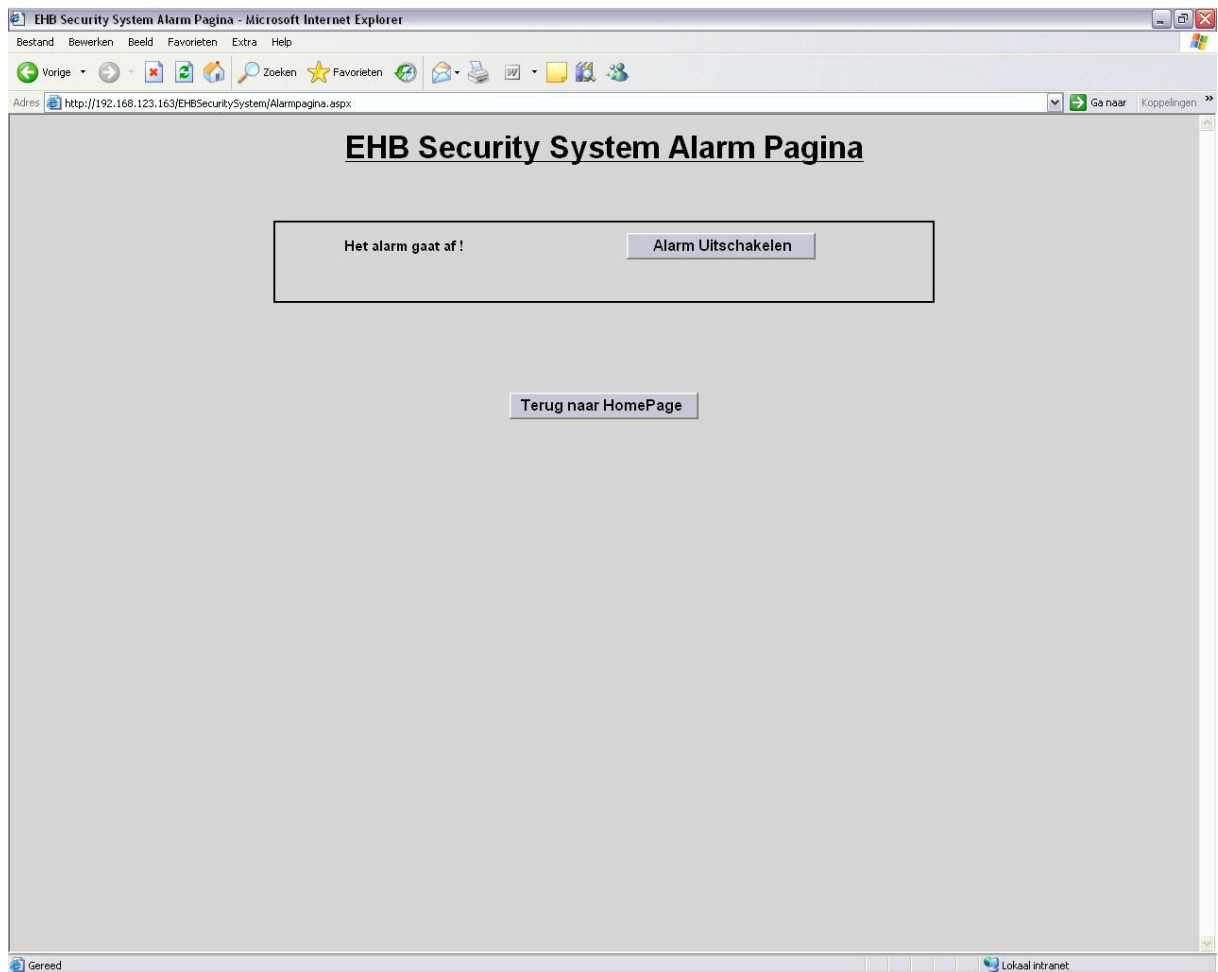
Voornaam	Achternaam	Code	Datum	UurBinnen	UurBuiten	Foto
Sigurd	Geerts	1234	15/05/2007	13:41:25	13:41:49	
Marjolein	Bresseleers	4567	15/05/2007	13:43:48	13:44:15	
Julien	GEERTS	0145	15/05/2007	22:10:05	22:10:35	

Buttons: Terug naar HomePage, Lokaal intranet

De Alarm pagina



De Alarm pagina (in geval van alarm)



Voorbeeld van een gestuurde email

Het bericht is verzonden met urgentie Laag.

Van:  ehbsecuritysystem@telenet.be

Verzonden: zo 13/05/2007 11:58

Aan: sigurd.geerts@student.ehb.be

CC:

Onderwerp: [SPAM] Toegangsgegevens EHB Security System !

Beste Geerts Sigurd,

U hebt toegang van 13/05/2007 10:00:00 tot 14/05/2007 00:00:00 tot de beschermde ruimte !

U moet de volgende toegangscode gebruiken : 3333 .

Opgelet wanneer u de ruimte probeert binnen te gaan voor of na de aangegeven periode zal u dit niet lukken !!!

Met vriendelijke groeten ,

EHB Security System