

BACHELOROPPGÅVE

Sjølvetjent
Campingplass

CamPLet

Automatisk innsjekkingsautomat for campingplass

av

(12) Roy Arve Hetle
(4) Jens Førsumd
(9) Eivind N. Inderøy
(10) Sander Falkenstein

Avdeling for ingeniør og naturfag
Ingeniørfag - elektro. automatiseringsteknikk
H02-300
Mai 2015

 HØGSKULEN I
SOGN OG FJORDANE

1.0 Referanseside

TITTEL H02-300 Bacheloroppgåve	RAPPORTNR. 1	DATO 22.05.2015
PROSJEKTTITTEL CampLett - Automatisk innsjekking	TILGJENGE Open	TAL SIDER 73
FORFATTARAR Roy Arve Hetle Eivind N. Inderøy Jens Førstund Sander Falkenstein	ANSVARLEGE RETTLEIARAR Joar Sande	
OPPDRAGSGJEVAR Høgskulen i Sogn og Fjordane		
SAMANDRAG Denne rapporten viser utviklinga av ein prototype for automatisk innsjekking på ein campingplass. Prosjektet er basert på ein idé frå eit av gruppemedlemene. Målet var å lage ein funksjonell og brukarvennleg innsjekkingsstasjon, der det skulle være mogleg for besøkande å sjekke inn sjølv og utføre betaling. Rapporten omhandlar undersøkingar og utvikling av prototypen og prosjektadministrativt arbeid som risikoanalyse. Resultatet er ein fungerande prototype som besøkande kan bruke til å sjekke inn på når det ikkje er betjening til stades.		
SUMMARY This rapport shows the development of a prototype for automatic check-in at a camping site. The project is based on an idea from one of the group members. The subject was to develop a functional and user friendly program where a simple check- in could be performed. This rapport shows investigations and development of the prototype and project administrative work such as risk-analysis. The result is a functional check-in system which customers can use when the camping site is unmanned.		
EMNEORD Java, JavaFX, MySQL-database, RS-232,berøringsskjerm, bankterminal, camping, automat, sjølvbetjent		

2.0 Forord

I løpet av sjette semester ved automatiseringslinja ved Høgskulen i Sogn og Fjordane (HiSF), skal studentane gjennomføre hovudprosjekt i faget *HO2-300-Bacheloroppgåve vår 2015*.

Prosjektet skal vere praktisk og/eller teoretisk. Det er eit krav at prosjektet skal dokumenterast i prosjektbeskriving, forprosjektrapport, plakat, pressemelding, munnleg presentasjon og sluttrapport. I tillegg skal det opprettast ei nettside der sluttrapporten skal leggjast ut.

Hovudprosjektet er ei vidareføring av eit prosjekt gruppa hadde femte semester. Det vart då undersøkt interesse hos eigarar av campingplassar for å opprette sjølvbetjent innsjekking. Interessa for eit slikt produkt var til stades.

Vi vil rette takk til Høgskulen i Sogn og Fjordane for støtte til innkjøp av nødvendige utstyr. Vidare rettar vi takk til Kjørnes og Haugen camping for innspel til utvikling av prototypen og Nets Norge for god rettleiing av betalingsløysing.

Vi vil rette ei stor takk til Ingebjørg H. Førstund og Annette Hetle for at dei har lest korrektur på rapporten. Avsluttande vil vi takke vår rettleiar, Joar Sande for rettleiing under arbeidet med prosjektet.

Førde 22.05.2015

Roy A. Hetle

Jens Førstund

Eivind N. Inderøy

Sander Falkenstein

3.0 Samandrag

Dette prosjektet er utført av fire studentar ved Høgskulen i Sogn og Fjordane våren 2015. Prosjektet har namnet CampLett, og målet er å prosjektere og utvikle ein programvare for bruk i sjølvbetjent innsjekkingsautomat.

Utvikling av programmet i prototypen er gjort i programmeringspråket Java, med NetBeans IDE som plattform. Databasen er oppretta i MySQL med Workbench som utviklingsplattform.

Fokuset har vore å utvikle ein brukarvenleg innsjekkingsautomat. Det er mellom anna laga fire forskjellige språkval. I tillegg er det laga figurar som skal vere logiske, slik at ein kan bruke programmet utan at ein beherskar nokon av språka. Dette gjer at det dei fleste campinggjestene kan bruke systemet.

Java-programmet behandlar informasjonen mellom det grafiske grensesnittet og databasen. Alle data som blir fylt inn av gjesten blir lagra i databasen, og samtidig blir data som pris og ledig plassar sendt til det grafiske grensesnittet. Som betalingsløysing er det implementert ein minibankterminal som kommuniserer med programmet. Gjesten må få godkjent kjøp på terminalen før innsjekkinga blir fullført.

Gruppa såg på moglegheita for å bygge prototypen etter regelverket for betalingsautomatar, for å sjå om det var mogleg å få den godkjent for kommersiell bruk. Det blei konkludert med at det ville bli for dyrt og for tidkrevjande å gjennomføre.

Det er utført ein test av programmet ved Høgskulen, og tilbakemeldingane frå studentar og tilsette var gode, under testinga vart det gjort endringar etter kvart som det kom tilbakemeldingar. I tillegg til dette har Kjørnes camping fått testa programmet og gitt det topp tilbakemeldingar.

Resultatet av prosjektet er ein fungerande prototype som gjestar kan nytte til innsjekking når betjeninga ikkje er til stade. Gjesten vel sjølv kor han skal bu, transaksjonen blir gjennomført med minibankkort, namn, lengde på opphald og betaling blir registrert i databasen.

4.0 Innhald

1.0 Referanseside	2
2.0 Forord	3
3.0 Samandrag.....	4
4.0 Innhald.....	5
5.0 Innleiing.....	8
5.1 Bakgrunn for oppgåva.....	8
5.2 Mål	10
5.3 Fagord	11
6.0 Samarbeid med næringslivet	14
6.1 Kjørnes Camping	14
6.2. Haugen Camping.....	14
6.3 Akutten studentkro.....	14
6.4 Microlog	14
6.5 Nets	14
6.6 Trucknor Firda.....	15
7.0 Utviklingsverktøy som er nytta	16
7.1 Java.....	16
7.2 JavaFX.....	16
7.3 MySQL.....	16
7.4 Bankterminal.....	17
7.5 RS-232 Standard.....	17
7.6 Panelpc.....	17
8.0 Teststasjon	18
9.0 Teknisk beskriving av prototypen til CampLett.....	19
9.1 Oppbygging av Java-basert logikk i CampLett-programmet.....	19
9.1.1 Java-klassen MySqlConnection	19
9.1.2 Java-klassen Bestilling.....	19
9.1.3 Java-klassen Bestillingsdata.....	20

9.1.4	Java-klassen HashMap.....	20
9.1.5	Java-klassen MainClass.....	22
9.1.6	Språkpakke	24
9.1.7	Skjermtastatur	25
9.2	Grafisk design i CampLett-programmet.....	26
9.2.1	Cascading Style Sheets	26
9.2.2	FXML og Controller -filer	27
9.2.3	Java-Klassen Alarm	28
9.2.4	Rammeverk for grafikk	28
9.2.5	Oppbygging av grafiske skjermbilde.....	30
9.3	Implementere bankterminal i CampLett-programmet.....	35
9.3.1	Java-biblioteket jbaxi.jar.....	35
9.3.2	Konfigurasjonsfila baxi.ini.....	35
9.3.3	Bruk av JBAXI i CampLett-programmet	35
9.4	Database	39
10.0	Utvikling av CampLett-programmet.....	41
10.1	Utvikling av grafisk design.....	42
10.2	Utvikling av Java-basert logikk i CampLett-programmet	45
10.2.1	Kommunikasjon med database	45
10.2.2	Språkpakke	45
10.2.3	Logikk for det grafiske	45
10.2.4	Skjermtastatur	46
10.3	Utvikling av databasesystemet til CampLett.....	48
10.4	Tilkopling mellom bankterminal og Java	49
10.4.1	Undersøking av kommunikasjon	49
10.4.2	Kommunikasjonstest med hyperterminalar.....	51
10.4.3	Alternativ 1 - Javacard	52
10.4.4	Alternativ 2 - BAXI.....	52
10.5	Test av prototype.....	54
10.5.1	Testresultat.....	56

10.5.2 Styrkar og svakheiter rundt testen.....	56
11.0 Refleksjon rundt vidareføring av produkt.....	57
12.0 Konklusjon.....	59
13.0 Prosjektadministrasjon.....	60
13.1 Organisering av prosjektgruppe.....	60
13.1.1 Styringsgruppe.....	60
13.2 Ressursbruk.....	61
13.2.1 Økonomi.....	61
13.2.2 Timebruk i prosjektet.....	62
13.3 Gjennomføring av prosjektet etter tidsplan.....	63
13.3.1 Møte.....	64
13.4 Risikostyring.....	65
13.4.1 Risikovurdering.....	65
13.4.2 SWOT-Analyse.....	66
13.5 Nettside.....	67
13.6 Prosjektevaluering.....	68
14.0 Figurliste.....	69
15.0 Kjelder.....	70
16.0 Vedlegg.....	73

5.0 Innleiing

CampLett er eit prosjekt som er utført av fire studentar ved Høgskulen i Sogn og Fjordane våren 2015. Prosjektet går ut på å lage ein prototype til eit sjølvbetent system for å sjekke inn på ein campingplass.

På mange campingplassar i dag er det ikkje betjening på ettermiddagar og kveldar. Ved å gjere innsjekkinga sjølvbetjent kan ein gjere kvardagen enklare for både kunde og eigar. Interesse og behovet for eit slikt produkt vart undersøkt Hausten 2014 i samband med eit studentprosjekt.

5.1 Bakgrunn for oppgåva

Grunnen til at prosjektgruppa valde å utvikle eit automatisert innsjekkingssystem for campingplassar var at eit av gruppemedlemene hadde vore på campingferie og var frustrert over at det var vanskeleg å finne ut om det var ledige plassar. I tillegg var det ofte ikkje bemanning tilgjengelig når ein kom til campingplassen. På bakgrunn av dette vart det undersøkt korleis campingplassane i vår region, som er Sogn og Fjordane, løyser slike utfordringar.

Gjennom prosjektet hausten 2014 hadde gruppa kontakt med mange ulike campingplassar. Dei fekk forklart at det vart jobba med ein ide om eit program som gjer at kundar kan sjekke inn sjølve.

Undersøkinga viste at dei fleste campingplassane i Sogn og Fjordane har personell på jobb om dagen, og eigarane er tilgjengelege på telefon utanfor arbeidstida. Mange stilte seg positive til ideen om å gjere dette automatisert, sidan dei ville unngå å måtte reise fram og tilbake mellom campingplassen og sin eigen bustad kvar gong det kjem nye kundar. Dei som var skeptiske til ideen, frykta at dei ville miste kontakten med kundane, og dei følte at det ikkje var noko belastning å vere tilgjengelig på telefon.

Etter undersøkingane tok gruppa kontakt med Kjørnes og Haugen camping som hadde vist stor interesse for å gjere sin arbeidskvardag enklare. Haugen camping har brukt store summer på å automatisere mykje av arbeidet frå før. Der har alle kundar eige nøkkelkort som gjev tilgang til campingplassen og sanitærbygg. Ynskja deira var å redusere behovet for

bemanning og heller bruke tid på å reise innom for å sjekke korleis gjestane har det og for å ta utvask.

Grappa besøkte Kjørnes og Haugen camping for å få innspel på korleis dei ønska at eit sjølvbetjent innsjekkingssystem skulle fungere. I tillegg ble det tatt telefonar til andre campingplassar, utifrå desse tilbakemeldingane såg vi at det var behov i marknaden. Men ein kritisk faktor var kostnaden.

På grunnlag av dette konkluderte vi med å jobbe vidare med ideen som hovudprosjekt. Sjølv om det var interesse i marknaden for å automatisere innsjekking, var eigarane skeptisk til å vere med på å utvikle dette. Grunnen var at dei følte dei ikkje hadde tid til å bidra, og dei var skeptiske på å investere i noko dei ikkje heilt veit kva er. Men dei var veldig interesserte i å vere med på å gje innspel og å sjå det ferdige produktet.

5.2 Mål

Prosjektfasen vart starta med å utforme eit forprosjekt, og her bestemte gruppa kva mål og avgrensingar prosjektet skulle ha. HiSF har tildelt gruppa 500 timar pr. student til å planlegge, gjennomføre og dokumentere heile prosjektet.

Hovudmål

Hovudmålet for prosjektet er å prosjektere og utvikle ein programvare for bruk i sjølvbetjent innsjekkingsautomat på campingplass.

Delmål

- Utvikle ein programvare for prototypen
- Gjere programvaren enkel å oppdatere, slik at nye funksjonar enkelt kan leggst til
- Lage eit brukarvenleg grensesnitt. Det skal brukast mest mogleg logiske og/eller standardiserte symbol i tillegg til forklarande tekst som er enkel å forstå
- Prototypen skal ha funksjonar for bestilling direkte på campingplassen
- Utvikle systemet basert på tilbakemeldingar frå dei campingplassane vi har samarbeid med tidlegare
- Lage ein teststasjon for utprøving av programvaren
- Brukargrensesnittet skal vere på fire ulike språk: tysk, engelsk, norsk og polsk
- Halde fokus på risikostyring
- Som ein del av prosjektet skal gruppa opprette ei nettside, der ein held den ved like med informasjon om prosjektet

Avgrensing

Produktet vert avgrensa til ein prototype for programvaren. Det vil sei at det er utvikling av programvaren som står i hovudfokus. Teststasjonen treng ikkje å lagast for å stå utandørs utan tilsyn, slik at ferdig produkt ville vere.

5.3 Fagord

HiSF	Høgskulen i Sogn og Fjordane
HiSF-AIN	Avdeling for Ingeniørfag og Naturfag ved Høgskulen i Sogn og Fjordane
Open kjeldekode	Programvare der kjeldekoden er offentlig tilgjengelig
Database	Strukturert samling av data
Bankterminal	Terminal som behandlar transaksjonar med minibankkort
Panel Pc	Ein pc med berøringsskjerm, der alt er bygd i ei eining
Berøringsskjerm	Ein skjerm som mekar berøring og kvar på skjermen berøringa er
Algoritme	Beskriving av ein endelig serie av operasjonar
Grensesnitt	Gjer at ulike program/menneske kan kommunisere med kvarandre
Administrator	Person som har særskilt tilgang og overordna ansvar
Universell utforming	Retningslinjer for at samfunnet skal vere tilpassa personer med funksjonsnedsettelse
Plattform	Eit miljø der forskjellige typar programvare kan bli køyrt innanfor
Prototype	Modell som blir laga før ein startar produksjon
HMI	Eit grensnitt mellom menneske og maskin også kalla brukargrensesnitt
Universell utforming	Krav til at ei løysing skal kunne brukast av alle
Smartcard	Smartkort, kort med innebygd elektronisk chip.
Protokoll	Standardiserte reglar for overføring, som kan brukast mellom ulike program på ulike maskiner i same nettverk
Hyperterminal	Ein programvare som kan overføre kommunikasjon via com-port
Databit / bit	Den elementære eininga i digital kommunikasjon. Kan berre ha verdi 0 eller 1
Serielle Bits	Ein bit blir sendt om gangen med enten verdi 0 eller 1
Domenenamn	Ein string som brukast til adressering på internett
Kjeldekode	Programmeringskode som definerer korleis eit dataprogram verkar
SWOT	Analyseverktøy for å kartleggje sterke og svake sider ved eit prosjekt
Primærnøkkel	Nøkkelverdi for å kople saman databasar
Bueining	Campingvogn, telt eller hytte på in campingplass
Handshake	Automatisk utveksling av parameter før datakommunikasjon starter
Java	Programmeringspråk
JavaFX	Grafikk og media –pakkar for Java
NetBeans IDE	Programmeringsplattform for blant anna Java
SceneBuilder	Verktøy i Netbeans for å designe applikasjonar til JavaFX
Oracle	Utviklar av Netbeans IDE og utgivar av Java
MySQL	Databaseadministrasjonsverktøy
MySQL Workbench	Plattform for utvikling av MySQL
WordPress	Verktøy for utforming av Nettside
TapTip	Skjermstatur som ligg i Windows 7 og 8
FTP-portal	Ein lagringsplass på internett for global tilgang til filer
Google docks	Verktøy for å behandle dokument på internett
CMD	Program på Windows for å sende kommandoar til OS
Excel	Rekneprogram som ligg i Microsoft Office

RS232	Ein standard for seriell kommunikasjon
USB	Universal Serial Bus, standard for å kople einingar til ei datamaskin
jbaxi.jar	JAR-fil, frå Nets med metodar for å kommunisere med bankterminal
baxi.ini	Konfigurasjonsfil frå Nets, for å kommunisere med bankterminal
Mini USB	Standard for kommunikasjon, og strømforsyning til mobile einingar
Ingenico ICT-220	Minibankterminal (merkenamn)
Com-port	Ein kommunikasjonsport på pc for seriell overføring av data
XAMPP	Verktøy for å teste og utvikle nettapplikasjonar
HTML	Utviklingspråk for nettside
PHP	Utviklingspråk for dynamiske nettsider
Windows	Eit operativsystem frå Microsoft
XML	Eit språk for deling av strukturerte data
PostMessage	Metode for å sende data i Windows
Compiler / Compiler	Program som konverterer frå kjeldekode til eit køyrbart dataprogram
DFS-13	Protokoll for å kommunisere med bankterminal
BAXI /JBAXI	Protokoll for å kommunisere med bankterminal
Javacard	Plattform for kommunisere med smartkort
Klasse	Ei samling av metodar
Bibliotek	Samling av klassar og variablar som kan brukast til programutvikling
Metode	Gruppe av programkommandoar som kan kallast på etter behov
Konstruktør	Metode som ligg i ein klasse, blir kjørt fyrste gong klassen eller objekt av den blir brukt.
Objekt	Ein instans av ein klasse som vert oppretta når programmet køyrer
Collection	Ei samling av objekt i Java-kode
HashMap	Ein type Collection som har både nøkkel og verdi
null-verdi	En tom verdi, den verdien ein variabel vil ha før den har blitt definert
Set-metode	Metode for å gi ein verdi til ein privat variabel
Get-metode	Metode for å få ein verdi frå ein privat variabel
Eventlister	Ein metode som overvakar eit parameter og kan utføre kommandoar ved endring i parameteren
Arv	Å få eigenskapar som variablar og metodar frå en anna klasse
StackPane	Klasse i JavaFX for å behandle bilde
Scene	JavaFX API for å presentere bilde på skjermen
Interface	Abstrakt Java-klasse, med metodar, utan innhald
CSS	Cascading Style Sheet, språk som kan bestemme utsjånad på HTML eller XML - filer
FXML	Språk utvikla frå XML, for å definere grafisk utsjånad i JavaFX
Controller-fil	Java-fil som inneheld logikk bak den grafiske utsjånaden
Parameter/variabel	Ein haldar for ein verdi som kan endre seg i programmet
String	Variabel som inneheld bokstavar som ord eller setningar
Varchar	Variabeltype som inneheld bokstavar, brukt i MySQL
Integer	Variabeltype i Java som inneheld heiltal
ECR	Electronic Cash Register, kassesystem.

EMV	Europay Mastercard and Visa, Standard for å gjere betaling med kort sikrare
OS	Operativ System
API	Applikasjonsprogrammeringsgrensesnitt, grensesnitt som gjer at eit program kan køyre delar av eit anna program
JDK	Java Development Kit, Java-versjon for å utvikle program
JRE	Java Runtime Environment, Java-versjon for å køyre program
JNA	Java Native Access, Java-bibliotek med metodar for å kommunisere med OS
DIFI	Direktoratet for forvaltning og IKT
OOP	Objekt orientert programmering
DE-9	Plugg som blir brukt som Com-port på pc
NDA	Non-disclosure agreement, er ein avtale om teieplikt

6.0 Samarbeid med næringslivet

I dette kapitlet fortel vi om ulike bedrifter vi har hatt kontakt med, og kva dei har bidrege med i prosjektet.

6.1 Kjørnes Camping

Kjørnes camping ligger 4 kilometer utanfor Sogndal i retning Kaupanger. Eigarane her er tilgjengeleg for kundane døgnet rundt og er avhengig av å vere i nærleiken av campingplassen til ei kvar tid. Løysinga Kjørnes har i dag er i høgste grad basert på tillit. Når personale ikkje er til stades, er det mogleg for besøkande å skrive si eiga kvittering og legge igjen kontantar for opphaldet i ei postkasse.

Når vi tok kontakt, var dei veldig positive til ei form for automatisering av innsjekkinga, då spesielt i skuldersesongen på våren og hausten. Dei har bidrege med gode råd om kva funksjonar ein bør ha i eit sjølvbetjent innsjekkingssystem.

6.2. Haugen Camping

Denne campingplassen er lokalisert på Skei i Jølster. Her har dei automatisert delar av campingdrifta allereie. Grunnen til at vi besøkte dei var for å sjå kva dei hadde gjort, og kva dei meinte vi burde ha med i vårt system. Her fekk vi mange gode tips om kva funksjonar dei som eigarar ynskjer seg.

6.3 Akutten studentkro

Akutten er den lokale studentkroa ved HiSF, avdeling Førde. Der fekk gruppa ein avtale om lån av minibankterminal utanom opningstidene deira.

6.4 Microlog

Microlog leverer betalingsløysingar i form av bankterminalar og automatar. Vi hadde fleire samtaler med dei for å finne ut kva krav som blei stilte til betalingsystem med tanke på å få vårt system godkjent.

6.5 Nets

Nets er leiande leverandør av betalingsløysingar i Norden. Den terminalen vi lånar av Akutten studentkro er levert av Nets. Vi har samarbeida med dei for å finne ut korleis vi skal få kommunikasjon mellom bankterminalen og vårt program.

For å få tilgang til konfigurasjonsfiler og Java-bibliotek som Nets har utvikla, har CampLett inngått ei NDA-avtale med Nets. Dette er ei avtale som sikrar at informasjon som blir delt mellom partane HiSF og Nets ikkje blir delt utanfor dei vilkåra som er bestemt av avtalen. sjå vedlegg 5.

Med grunnlag i punkt 4. i NDA-avtalen, og samtale med Nets, sjå vedlegg 4. Har vi komme til at sensor og fagleiar kan få innsyns i Nets sine filer. Kjeldekodane er derfor levert inn i si heilheit. Rapporten kan ikkje gå grundigare inn på Nets sitt filsystem enn den gjer, då den skal ut i offentlegheita.

6.6 Trucknor Firda

Trucknor Firda er ein lokalt verkstad for tunge kjøretøy i Førde. Her har vi fått låne utstyr til å bygge haldar for prototypen vår.

7.0 Utviklingsverktøy som er nytta

I dette kapittelet vil vi gå nærare inn på kva produkt og program som er brukt i prosjektet.

7.1 Java

NetBeans IDE 8.0 er brukt som programmeringsplattform for å utvikle CampLett-programmet. Programmet er skriva i programmeringsspråket Java. Dette er eit objektorientert programmeringsspråk [1]. Grunnane til at Java er brukt er fleire. Det er ingen kostnader ved å bruke verken utviklingsverktøya eller biblioteka, og det finst mange opne kjeldekodar på internett. Java er også eit av dei mest brukte programmeringsspråka i verda [2]. Alle i prosjektgruppa har også god kjennskap til Java frå før.

Det finst to versjonar av Java, Java Development Kit (JDK) og Java Runtime Environment (JRE). For å utvikle Java-program, er det nødvendig å ha JDK. Denne pakken gir tilgang til alle standard Java-bibliotek i tillegg til ein del nødvendige utviklarverktøy. [3]

Teststasjonen til CampLett skal ikkje brukast til utvikling av program, derfor er det nok å installere JRE for å køyre ferdige program. JRE inneheld dei same biblioteka som JDK, men utviklarverktøya følgjer ikkje med. Det er heller ikkje nødvendig å installere NetBeans eller andre plattformer for å køyre programmet, då JRE ordnar det sjølv.

7.2 JavaFX

JavaFX er utvikla for at programutviklarar skal kunne lage gode grafisk grensesnitt som fungerer likt på ulike plattformer. Dette biblioteket inneheld ei stor mengd klassar som kan nyttast til å lage grafikk og animasjon. JavaFX gjer det mogeleg å lage design i FXML-kode eller å bruke Scene Builder til å konstruere FXML-filer, med ein «drag and drop» -funksjon [4]. Dette gjer at ein kan skilje brukargrensesnittet og logikken på ein enkel og oversiktleg måte.

7.3 MySQL

For å ha kontroll over plassar og gjester, treng programmet ein database. Gjennom undervisning i tidlegare delar av studiet har gruppa fått kjennskap til MySQL. Dette er verdens mest brukte databaseverktøy [5]. Ein annan stor fordel med MySQL er at det er eit gratis system. For å opprette databasen, er det brukt MySQL Workbench. Det er enkelt å kommunisere med MySQL frå dei fleste programmeringsspråk, deriblant Java som er brukt i dette prosjektet.

7.4 Bankterminal

Eit av måla var å ha ei betalingsløyning. For å få til det, var det behov for å opprette kommunikasjon mellom ein bankterminal og vår programvare. Det blei gjort ein avtale med Akutten studentkro om å få låne bankterminalen deira. Dette var ein Ingenico ICT220 terminal som er levert av Nets som leverer betalingskort og informasjonsløyningar i dei nordiske landa. [6]

7.5 RS-232 Standard

RS-232 er ein standard for seriell samankopling og kommunikasjon. Her har ein eit val mellom to ulike pluggar, DB25 eller DE-9. Dette er en standard som kom på 60-talet [7]. RS-232 er ein enkel kommunikasjonsprotokoll der ein sender enkle databits, 0 eller 1, over ledarar i ein RS-kabel. Det er også mogleg å sende RS-232-signal gjennom ein USB-kabel slik vi har gjort i dette prosjektet.

7.6 Panelpc

Pc-løysinga er levert av firmaet Elektronix Com-Scan AS. Den er av merket Aplex og har ein resistiv berøringsskjerm. Maskina har fire COM-portar som ein treng for å kunne kople til eksterne einingar som bankterminal, mus og tastatur [8].

Når det skulle veljast leverandør, fokuserte gruppa på at produktet skulle være av høg kvalitet. Panelpc-en skulle i tillegg vere enkel å frakte med seg. Sidan den har integrert harddisk og prosessor, krev den liten plass og er lett å ta med seg til demonstrasjon hos potensielle kundar.

8.0 Teststasjon

For å utvikle eit produkt som skal ut i marknaden, er det eit stort arbeid å setje seg inn i kvar einiskild standard og regelverk. Sidan prosjektet går ut på å byggje ein prototype, er ein ikkje avhengig av å bygge den etter standarden. Grappa ville likevel sjå om det var mogleg å byggje prototypen etter reglane som er basert på lova om universell utforming [9].

Både Vegdirektoratet og DIFI har utarbeida dokument som beskriv kva krav som blir sett til ein automatisert betalingsautomat. DIFI har laga oversikt over kva standardar automatar skal samsvare med. [10] Det einaste som er godkjent på prototypen er bankterminalen, den er EMV sertifisert [11]. Sidan innkjøp av så mange standardar var kostbart, blei dei ikkje kjøpt inn. Ein baserte seg på *Vegdirektoratet sin vegleder*, som set krav til plass rundt automaten og betjeningshøgde [12]. Det er også krav til innsyn, dvs. lysstyrken på skjermen og kva vinklar ein skal kunne lese innhaldet frå [13].

Frå blindeforbundet er det anbefalt å ikkje bruke berøringsskjerm. Det skal og vere mogleg å ha lyd for å informere, og tastatur godkjent for blinde og synshemma [14].

Etter å ha sett oss inn i regelverket, blei det konkludert med at ein måtte sjå vekk frå ein del av anbefalingane sidan dei ikkje var relevante for ein prototype.

Under bygginga av teststasjonen, blei fokuset å lage ein haldar som gjer det enkelt å ta med seg produktet ut til potensielle kundar for demonstrasjon. Det blei derfor bygd ei ramme som panelpc-en kan setjast inn i. Det vart valt å sjå vekk frå kravet til høgde og plassering sidan det ikkje er aktuelt å setje den opp på ein fast plass. Dei fleste gjestar ved campingplassar brukar bil for å komme seg til campingplassen og dermed tilfredsstillar dei krava til syn.

9.0 Teknisk beskriving av prototypen til CampLett

Denne delen av rapporten går djupare inn i programmeringa, program og komponentar som er brukt for å utvikle CampLett. Kapittelet er ei beskriving på korleis CampLett-programmet er bygd opp. I vedlegg nr 9 kan ein sjå blokkdiagram og i vedlegg 10 flytskjema for heile CampLett-programmet.

9.1 Oppbygging av Java-basert logikk i CampLett-programmet

CampLett-programmet er i stor grad basert på objektorientert programmering (OOP). OOP handlar om å dele opp programmet i klassar som har sine sett av variablar og metodar for å utføre spesifikke oppgåver. Eit godt eksempel på det er klassen «MySQLConnection», som er nærare beskrive i følgjande underkapittel. Eit objekt er ein instans eller ein kopi av ein klasse som kan brukast når som helst. Det innehar alle variablar og metodar frå klassen det er objekt av. Det kan opprettast så mange uavhengige objekt av ein klasse som ein ynskjer [1].

Vidare i dette kapittelet vert det beskrive korleis den logiske delen som ligg bak CampLett-programmet er bygd opp og fungerer.

9.1.1 Java-klassen MySQLConnection

Dette er ein klasse som behandlar all kommunikasjon mellom Java og MySQL-databasen. Her er det metodar for å setje bestilling, slette bestilling og lese kva plassar som er opptatt i bestillingsdatabasen. Det finst også ein metode for å lese informasjon som pris, størrelse og type på dei ulike bueiningane.

9.1.2 Java-klassen Bestilling

Klassen «Bestilling» inneheld variablar som stemmer overeins med kolonnane for bestillingar i databasen. På den måten kan eit objekt av denne klassen representere ei bestilling i databasen, sjå figur 1 og 2. Klassen har «set og get -metodar» for å setje inn og hente informasjon i objekta av klassen.

```

public class Bestilling {

    private int nr;
    private Date inskjekk;
    private Date utskjekk;
    private String fornavn;
    private String etternavn;
    private double totalsum;
    private boolean betalt;
}

```

Figur 1 Utsnitt av klassen bestilling, variablene stemmer overeins med databasen

Nr	innsjekk	utsjekk	Fornavn	Etternavn	Total_Sum	Betlaing_godkjent
1	NULL	NULL	NULL	NULL	0	0
2	NULL	NULL	NULL	NULL	0	0
3	NULL	NULL	NULL	NULL	0	0
4	NULL	NULL	NULL	NULL	0	0
5	NULL	NULL	NULL	NULL	0	0
6	NULL	NULL	NULL	NULL	0	0
7	NULL	NULL	NULL	NULL	0	0
8	NULL	NULL	NULL	NULL	0	0
9	NULL	NULL	NULL	NULL	0	0
10	NULL	NULL	NULL	NULL	0	0

Figur 2 Utsnitt av databasen, variablene stemmer overeins med klassen bestilling

9.1.3 Java-klassen Bestillingsdata

På same måte som «Bestilling», inneheld «Bestillingsdata» variablar som stemmer overeins med kolonnane i den andre tabellen i databasen. På den måten kan eit objekt av denne klassen representere ei bueining i databasen.

Klassen har «get-metodar» for å hente informasjon i objekta av klassen. Alle parameter vert sett i konstruktøren. Dette er fordi informasjonen berre skal setjast inn når den vert henta frå databasen. Etter dette skal ein berre ha mogelegheit til å lese frå objektet.

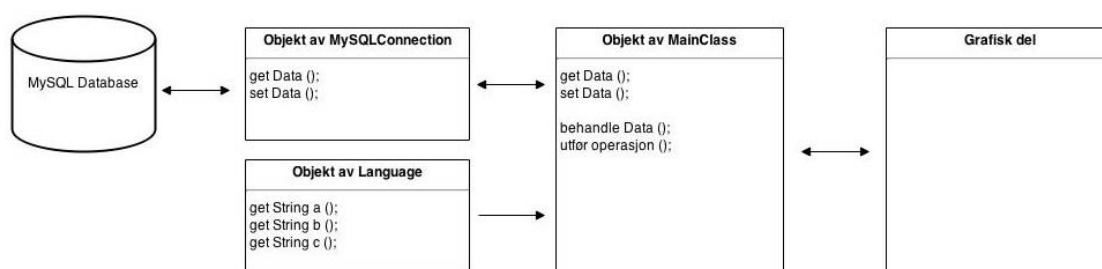
9.1.4 Java-klassen HashMap

I fleire tilfelle er det nødvendig å flytte mange objekt i programmet på same tid. Eksempel på dette er når «MySQLConnection» blir beden om å hente alle bestillingar frå databasen, og kvar bestilling blir representert av eit objekt av «Bestilling». Objekta vert samla i ein «HashMap», som er ein «Collection» eller samling av data [15].

Grunnen til at «HashMap» er brukt, er at den har to inngangsparameter, nøkkel og verdi [16]. Alle bueiningane på campingplassen har unike nummer, og dei er også primærnøkkel i databasen. Ved å bruke dette nummeret som nøkkel i «HashMap» vert det enkelt å hente ut ei aktuell bestilling frå ei større samling.

9.1.5 Java-klassen MainClass

I grove trekk er all databehandling og berekningar avskilt frå den grafiske delen av programmet. Denne organiseringa er gjort for å lage god oversikt, og for å lette utviding og feilretting av programmet. Den grafiske delen brukar eit objekt av klassen «MainClass» som inneheld metodar for databehandling og nødvendige objekt av andre klassar. Figur 3 viser eit utsnitt av dette systemet. Skjema over heile systemet kan ein sjå i vedlegg 9.



Figur 3 Utsnitt av samansettinga av klassane i CampLett-programmet

Metodane i «MainClass» er konstruert for å berekne og hente den informasjonen den grafiske delen har behov for, for å vise relevant informasjon på skjermen. Eksempel på dette er å sortere ut dei ledige hyttene frå alle bueiningane, eller å vise eller skjule tastatur på skjermen.

Vidare vert nokre av dei mest omfattande metodane i «MainClass» forklart djupare.

Setje bestilling

Den grafiske delen av programmet vil i løpet av bestillingsprosessen be brukaren om å taste inn alle opplysningar som krevjast for å utføre ei bestilling. Dette er informasjon om kva spesifikk bueining som er ønska og lengde på opphaldet. Informasjon som pris og om betalinga er godkjend vert berekna i «MainClass».

I «MainClass» finnes det eit objekt av «Bestilling». Dette er i utgangspunktet tomt og er tilgjengelig under heile bestillingsprosessen. Etter kvart som informasjon vert tilgjengeleg vert den sett inn i dette objektet. Når betalinga er godkjend, vert objektet sendt til MySQL-databasen og lagra som ei ny godkjend bestilling.

Omstart

«MainClass» har en metode som bli kjørt kvar gong ei bestilling er godkjend eller blir avbroten, det vil seie kvar gong «velkommenbildet» viser på skjermen. Metoden heiter «cleanSystem», og namnet reflekterer funksjonen godt. Meininga er å setje parameter i «MainClass» tilbake til utgangspunktet, for eksempel å setje bestiller sitt namn og lengde på opphald tilbake til null.

Tabellen med informasjon om bualternativa vert lasta inn frå MySQL i denne metoden. Programmet er sett opp til å laste dette i «cleanSystem» i staden for kvar gong ein treng informasjon. Fordelen med dette er at kommunikasjonen ut av Java vert avgrensa til ein gong for kvar bestilling. Slik sparar ein bruken av prosessor og minne på maskina.

Databasen med bestillingar vert og lasta inn her. Dette er for å kontrollere om utleigetida på nokon av dei aktive bestillingane er utgått. Er den det, vert bestillinga teken ut av databasen.

I «cleanSystem» vert det valte språket sett til engelsk sjølv om det fyrste skjermbildet ber kunden velje språk før språket vert nytta til noko i programmet. Grunnen til at det likevel vert sett til engelsk i «cleanSystem», er ei sikring mot at det skal oppstå null-verdiar ved eventuelle feil.

«cleanSystem» har ein siste sikkerheitsfunksjon. Om ikkje Java får kontakt med databasen vert det sendt opp ei feilmelding i eit eige vindauge på skjermen. Det informerer om nettopp dette og ber kunden ta kontakt med personell på campingplassen.

Loggfil

Når dei ulike delane av programmet vert laga i NetBeans og Java Development Kit(JDK), kan programmet køyrast med tilgang til eit vindauge som viser informasjon om prosessen. Ein kan også skrive ut verdiar for å fylgje betre med ved feilsøking. Dette er derimot ikkje mogeleg når ein køyrer programmet utanfor NetBeans. Derfor har «MainClass» ein metode som kan skrive ut stringar til ei loggfil som legg seg lokalt på pc-en. Her kan systemansvarleg gå inn å lese om det har komme feilmeldingar og liknande.

Interface

«MainClass» implementerer ein «Interface». «Interface» er ein abstrakt klasse som berre inneheld tomme metodar og har ingen anna funksjon en å vere eit skjelett for «MainClass». Ved å implementere denne «Interfacen» får ein feilmeldingar om ein prøver å køyre «MainClass» utan at den inneheld metodar med tilsvarande namn og parameter som «Interfacen» har [17]. Dette er gjort i «MainClass» for det fyrste for å vere trygg på at ikkje vitale delar av programmet manglar, men også for å skape oversikt over alle metodane som trengst.

9.1.6 Språkpakke

CampLett-programmet er oversett til fire forskjellige språk: norsk, engelsk, tysk og polsk.

Språkpakken er bygd opp av klassen «Language», og ein klasse for kvart språk. Denne klassen har ein tom String-variabel for kvar av stringane som vert nytta i heile den grafiske delen av programmet. Med andre ord alle dei synlege setningane eller enkeltord som i eit eller anna tilfelle skal visast på skjermen. Kvar av desse variablane har ein «get-metode», og alle variablane vert sett i konstruktøren til klassen.

For kvart av dei aktuelle språka finst det ein eigen klasse som arvar (extends) frå «Language». Denne arva medfører tilgang til alle String-variablane. Det medfører også arv av konstruktøren, og det er her vi fyller inn med setningane og ord omsett til det aktuelle språket.

I hovudprogrammet ligg det eit objekt av klassen «Language» som heiter «valgtSprak». Når ein kunde vel kva språk han vil bruke, vil dette objektet bli sett likt den klassen som inneheld det aktuelle språket. Dette er mogeleg fordi språkklassane har arva strukturen til «language».

Når den grafiske delen av programmet skal vise ein tekst hentast den aktuelle setninga frå objektet «valgtSprak» via get-metoden til variabelen. Dette vil alltid komme på riktig språk fordi programmet først set objektet til eit språk og alltid refererer til dette objektet.

9.1.7 Skjermtastatur

Programmet vårt gjer seg nytte av «TabTip», det innebygde skjermtastaturet i Windows 7. Innstillingane for dette tastaturet kan setjast på førehand og vil bli brukt kvar gong tastaturet vert opna. Vi har sett tastaturet til å dekke halve skjermen og vere forankra i botnen.

For å opne tastaturet ved behov brukar vi metoden `Runtime` frå Java-biblioteket [18]. Den gjer det mogeleg å sende startkommandoar til Windows. For å fjerne tastaturet frå skjermen igjen, nyttar vi eit eksternt bibliotek, `Java Native Access (JNA)`. Dette biblioteket inneheld metodar som gjer oss i stand til å kommunisere med operativsystemet på tvers av plattform og språk [19]. Kommandoen som vert sendt for å skjule «TabTip» er ein «`PostMessage`». Det er same type kommando som Windows brukar for kommunisere internt [20].

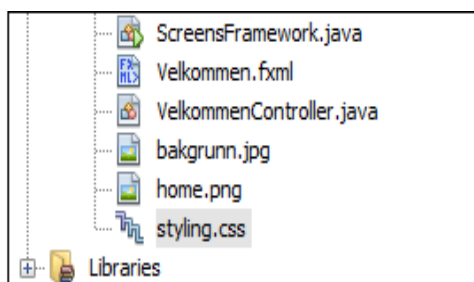
9.2 Grafisk design i CampLett-programmet

Grensesnittet i CampLett er kommunikasjonen mellom datamaskin og menneske. For å kunne forklare korleis det grafiske grensesnittet er bygd opp, blir det i dette kapittelet ei gjennomgåing av dei ulike verktøya og metodane som er nytta for å lage dei grafiske bilda og for å få dei til å kommunisere med Java-koden.

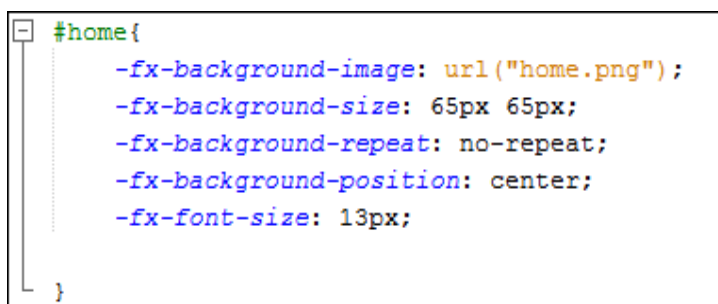
9.2.1 Cascading Style Sheets

Den grafiske delen av CampLett-programmet er i si heilheit designa ved hjelp av Cascading Style Sheets (CSS). Dette er ein enkel mekanisme for å legge til stilar som skrift, farge og størrelse på web-dokument. Det er eit språk som vert brukt til å definere utsjånaden på filer skrivne i HTML eller XML. Ei CSS-fil kan peike på og styre eit uavgrensa tal filer, eksempel på CSS-fil ser ein i figur nr 5.

Ved å endre CSS-fila, vil ein endre fargebruk, bakgrunnsbilde osv. i alle dokument som fila peikar på. CSS overkøyrer FXML-kodar når det gjelde utsjånad. Figur nr. 4 viser eit døme på dette. Der er bakgrunnsbildet sett, og størrelsen på skrifta er endra [21].



Figur 5 viser CSS-fila styling.css

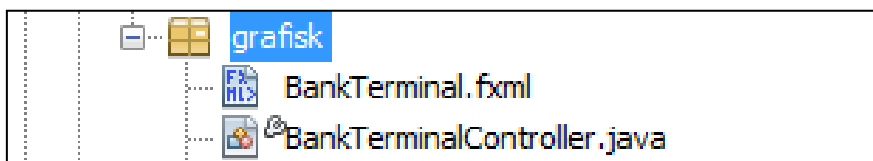


Figur 4 Viser utsnitt av CSS-fil

9.2.2 FXML og Controller -filer

CampLett-programmet sin grafiske del er i hovudsak bygd opp av eit rammeverk, sjå kapittel 9.2.4, og eit sett av FXML-filer og Controller-filer. Kvart skjermbilde har eit slikt sett.

I figur 6 ser ein Controller-fila «BankTerminalController.java» som er ei Java-fil og inneheld logikken bak skjermbildet. FXML-Fila «BankTerminal.fxml» inneheld grensesnitt, struktur og element i XML form, med andre ord utsjånaden på skjermbildet. FXML-fila inneheld i tillegg til utsjånaden på skjermbildet også informasjon om kva Controller-fil og CSS-fil som høyrer til skjermbildet slik at desse automatisk vert køyrt samtidig som FXML-fila.



Figur 6 FXML og Java -fil

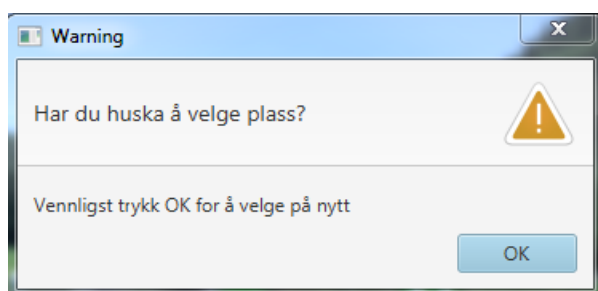
Netbeans autogenererer ein heil del for oss etter FXML-fila er laga, blant anna eit variabelnavn i Java-fila for kvart element i FXML-fila. Slik kan vi styre FXML-elementet frå Java-kode.

Det blir også autogenerert ein «EventListener» som tar hand om det som skal skje når ein trykkjer på ein knapp eller flyttar på ein slider. «Eventlistenaren» er linka til ein metode i Java-fila som vert køyrt når noko skjer på skjermen. Slik kan element på skjermen styre ting som skal skje i Java-koden.

9.2.3 Java-Klassen Alarm

For å varsle kunden om eventuelle feil i programmet, eller ved tilfelle der kunden har gløymt å fylle inn nødvendig informasjon for å fullføre bestillinga er det nytta ein klasse som heier «Alert». Denne klassen ligg i JavaFX-biblioteket og kom med oppdateringa «8u40» [22].

Som ein ser av figur 7, sprett det opp eit standardisert vindauge på skjermen. Dette legg seg over det eksisterande skjermbildet og viser den nødvendige informasjonen. Vindauget vil forsvinne om ein trykker «OK».



Figur 7 Alarm på skjerm

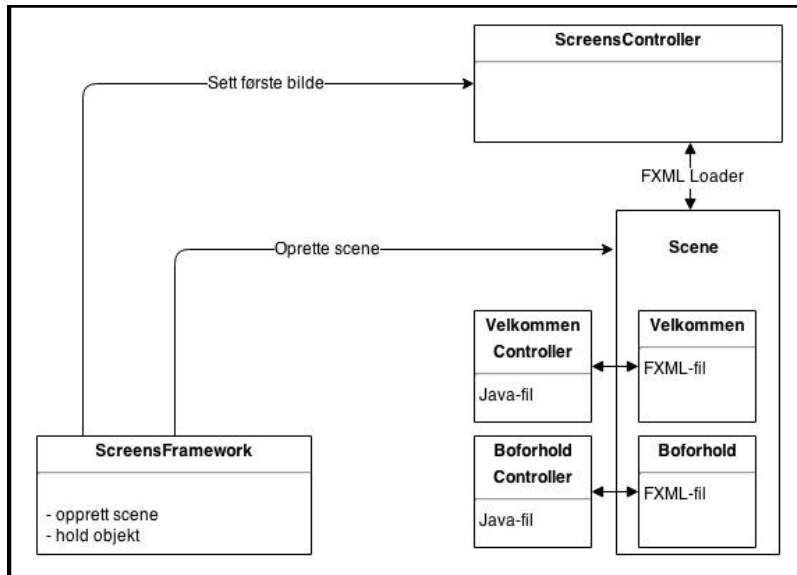
9.2.4 Rammeverk for grafikk

Programmet har åtte ulike skjermbilde som skal visast. Vi vil her forklare korleis organiseringa av dette er gjort i JavaFX. Denne delen av programmet er bygd på ein artikkel av «Angela Ceicedo» publisert 8. februar 2013 [23].

«ScreensFramework» er den klassen i CampLett-programmet som sett i gong programmet. I oppstartinga vert det grafiske bilde eller scena oppretta. Det er også her vi bestemmer at bildet skal visast i fullskjerm og at standard menyramma skal fjernast. Programmet har no eit grafisk bilde, men det er tomt og inneheld berre ein kvit bakgrunn.

«ScreensController» arvar i frå klassen «StackPane» som har eigenskapar som lagar eit fint miljø for å vise skjermbilda våre, mellom anna ved å ta seg av tilpassing av størrelsen på bildet i forhold til tilgjengelig skjermstørrelse. «StackPane» kan også legge fleire skjermbilder oppå kvarandre i ein stabel, og ein kan endre rekkefølgjene etter kva bilde ein vil ha synleg [24]. I CampLett-programmet er det berre eit bilde i stabelen om gongen. Dette er for at grafikken på datamaskina skal få mindre å arbeide med, og på denne måten minimerar vi sjansen for at bilde frys eller at det hakkar.

«ScreensController» inneheld metodar for å organisere kva bilde som vert vist på skjermen. Hovudmetoden i klassen må vite kva skjermbilde som skal køyrast for å starte. Den krev namnet på FXML-fila i String-form som inngangsparameter.



Figur 8 Utsnitt av det grafiske rammeverket

Figur 8 viser blokkskjema av den grafiske delen av programmet, sjå resten av systemet i vedlegg 9. Når programmet startar, vert det fyrste skjermbildet lagt i «StackPane», og ved hjelp av ein «FXMML Loader» vert det lasta inn i den grafiske scena . Kvar gong grafikken tilseier at det skal bytast skjermbilde vert hovudmetoden i «ScreensController» køyrt med referanse til det nye skjermbildet. Denne hovudmetoden vil også køyre ein «fade inn / fade ut» animasjon av skjermbilde på 300 millisekund for å få ein mjuk overgang.

9.2.5 Oppbygging av grafiske skjermbilde

Språkalternativa i hovudmenyen er laga ved å ta i bruk fire knappar og fire «imageview» (digitale bilderammer). Knappen og «imageview» står plassert ved sida av kvarandre, som vist på figur 9. Det er gjort på same måte for alle språkalternativa under kvarandre.



Figur 9 Utsnitt frå første skjermbilde

Knappen og den tilhøyrande «imageview» har fått den same FXML-ID-en. Det vil seie at om ein klikkar på knappen med teksten «Norsk» eller det norske flagget vil den same handlinga bli utført i controller-fila.



Figur 10 Skjerm bilde "Boforhold"

Figur 10 viser skjerm bildet «Boforhold», og det er her i programmet bualternativet blir valt. Her er det plassert knappar, «imageview», sirkler og labelar. Det er gjort på same måte som på førre skjerm bilde, knapp og «imageview» side om side har same FXML-ID. Og ved å klikke på ein av dei blir det utført ei handling basert på valet.

Det er plassert to sirkler med tilhøyrande tekst som viser kva dei forskjellige fargane rød og grøn indikerer. Dei er berre til informasjon og utfører ingen handling om dei vert trykt på.

Dei tre sirklane ved bualternativa viser om det er ledig plass der eller ikkje. Når skjerm bildet blir lasta vert det henta informasjon frå databasen for å sjekke kva plassar som er ledige. Sirklane vil heile tida vise fargen grøn om det er minst ein ledig plass. Prisen per døgn er henta frå databasen og er oppgitt i norsk valuta. Stringen «Dagar» er henta frå språkklassen og kjem derfor på språket som tidlegare er valt.



Figur 13 Oversikt over hytter



Figur 12 Oversikt over telt plassar



Figur 11 Oversikt over bubil plassar

I neste skjermbilde kjem ein til kart over campingplassen for den bueinga ein har valt. Det er laga tre forskjellige skjermbilde for å dekke førre val, sjå figur 11, 12 og 13. Desse skjermbilda er bygd opp med ein sirkel for kvar plass som ein skal trykke på for å velje plassen. Alle sju plassane er raude i utgangspunktet, men etter programmet har sjekka om dei er ledige, vert dei sett grøne. Dette skjer før skjermbilde blir presentert på skjermen.

Trykker ein på ein knapp som er raud, vil det bli køyrt ein metode som viser ein alarm med beskjed om å velje ein annan plass, sjå kap. 9.2.3. Vert det trykt på ein grøn sirkel, vil den skifte til raud farge og lagre ID-en for vidare bestilling.



Figur 14 Registrering av namn og val av overnattingsdøgn

Figur 14 viser eit av dei mest omfattande skjermbilda i programmet. Tekstfeltet er til for å laste inn namnet til kunden. Tekstfeltet er modifisert med metodar i Java som gjer at det automatisk blir skriven stor forbokstav. Ved å bruke ein metode «getText()» i Java sitt bibliotek, hentar ein det som er skriven inn i tekstfeltet, og det bli brukt som parameter for å setje bestillinga i databasen.

I Figur 14 ser ein også slideren som blir brukt for å velje lengde på opphaldet. I prototypen har vi sett ei nedre grense på ein dag og øvre grense på ti dagar. Dette er mulig å endre etter behov. Slideren fungerer slik at ved å dra den fram eller tilbake, vil den sende ein verdi kvar gong den endrar seg. Java-programmet vil gjere ei utrekning, og totalsummen vil heile tida bli vist i ein «label» under slideren.



Figur 15 Bankterminalbilde med progressbar

Neste skjermbilde er basert på kontakt med bankterminal og heiter «BankTerminal», sjå figur 15. Her går det ein framdriftsindikator eller «progress bar» rundt og rundt for å indikere at CampLett-programmet ventar på at det skal bli betalt i bankterminalen. Dette er ein animasjon som er henta frå JavaFX.

I alle skjermbilda er det dei same knappane nedst på skjermen, som vist i figur 14. Det er ein «Home-knapp» som skal ta kunden tilbake til startside. Kanskje blir kunden usikker på noko, eller har komme til ei halvferdig bestilling på eit framandspråk. Det er også ein knapp for å gå tilbake til førre skjermbilde og ein for å gå vidare til neste på dei fleste skjermbilda.

9.3 Implementere bankterminal i CampLett-programmet

Nets har utvikla ein API for tilkopling av bankterminal via RS-232. Denne blir referert til som JBAXI og består mellom anna av filene jbaxi.jar og baxi.ini. For å få tilgang til dette har CampLett måtte inngå ei NDA-avtale med Nets, sjå vedlegg 5. Dette er ei avtale som sikrar at informasjon som blir delt mellom partane HiSF og Nets ikkje blir delt utanfor dei vilkåra som er bestemt av avtalen. Derfor blir det ikkje gått grundigare inn på denne API-en, enn eit lett oversyn.

9.3.1 Java-biblioteket jbaxi.jar

For å kunne lese og sende data må programmet kommunisere gjennom biblioteket jbaxi.jar. Programmet kan då lese hendingar ved hjelp av ein «eventlistener» som kan brukast til å bestemme kva informasjon som skal sendast tilbake til terminalen.

9.3.2 Konfigurasjonsfila baxi.ini

Sidan bankterminalar behandlar data som skal vere sikre og omhandlar pengesummar, må alt loggast. Baxi.ini fungerer hovudsakleg som ei konfigurasjonsfil der ein kan stille hastigheit på sending, type port, osv., men også som ein logg for programmet. Den lagrar informasjon om alt som er sendt og lese mellom program og terminal.

9.3.3 Bruk av JBAXI i CampLett-programmet

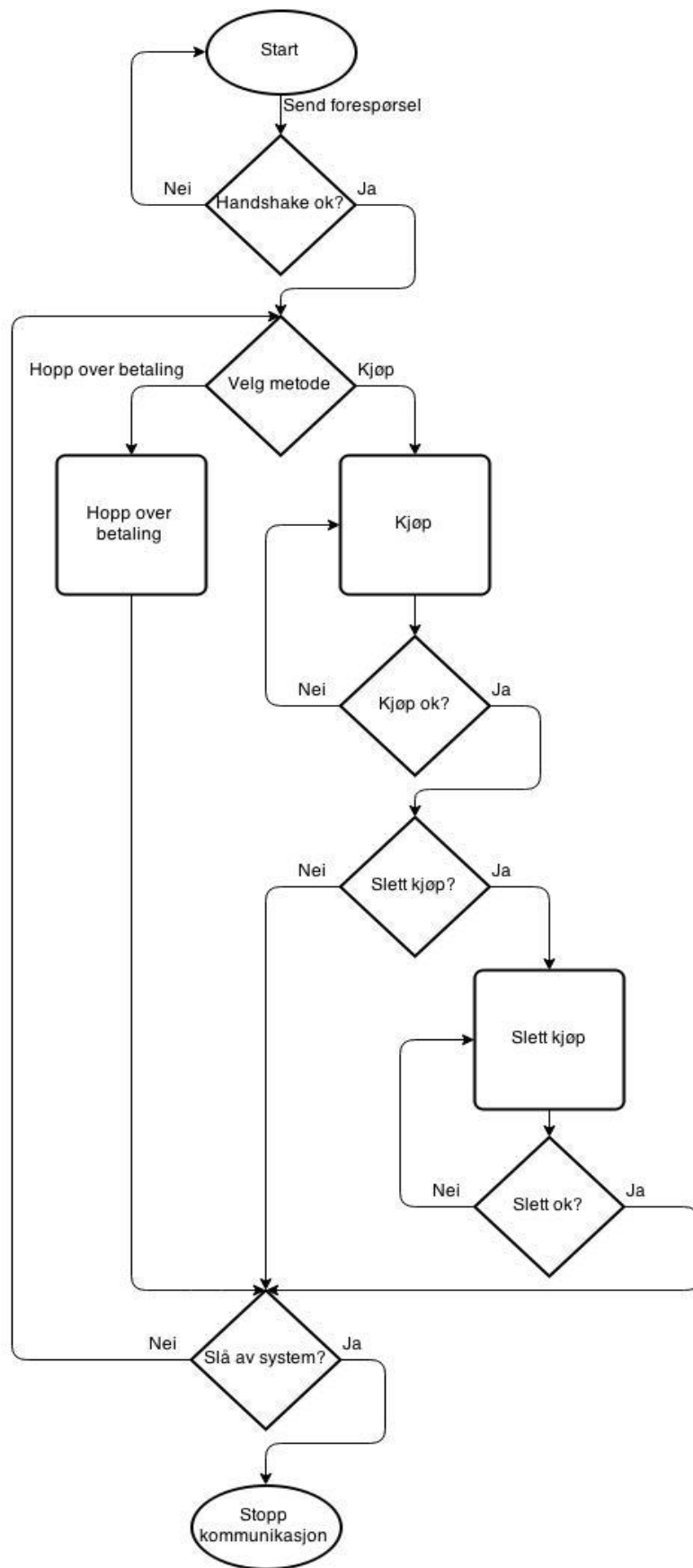
Protokollen til terminalen tilseier at det må sendast ulike «Integer og String» -verdiar, der «Integer» er eit heiltall og String er ein tekst, i ulike metodar for å signalisere kva type operasjon som skal køyrast. Dei ulike kombinasjonane av verdiar gjer at det kan køyrast ulike operasjonar ved hjelp av same metode. Til dømes kan det køyrast «kjøp» og «slett kjøp» i same metode, så lenge det vert sendt riktige verdiar til terminalen.

I CampLett-programmet er det ein klasse som handterer kommunikasjon mellom pc og terminal. Dette er den einaste klassen i programmet som vert køyrt frå det grafiske grensesnittet utan å gå via «MainClass», sidan metodane frå denne klassen blir kalla direkte ifrå «BankTerminal-Controller».

Det er seks metodar som er laga for terminalen, med mulighet for utviding:

1. Starte kommunikasjon med terminal
2. Slå av kommunikasjon med terminal
3. Kjøp
4. Slett kjøp
5. Avbryt kjøp
6. Avstemming

Med desse metodane kan CampLett-programmet gjennomføre og slette sal. Det er fleire alternativ ein kan legge til om ein ynskjer å utvide programmet. Til dømes kan ein lagre all informasjon i ein database som berre administrator har tilgang til.



Figur 16 Algoritmen til betalingsdelen av CamPLett-programmet

Figur 16 viser korleis prosessen til terminalen fungerer i programmet. Det blir fyrst sendt ein førespurnad til terminal ved start av CampLett-programmet. Om førespurnaden blir godkjent, vil den vente til programmet køyrer ein av dei seks metodane.

Når kunden har valt plass og skrive inn alle kriterier for å gå vidare, sender programmet metoden «kjøp» til terminal, og kunden kan betale i terminalen. Når kjøpet er fullført, er det mogeleg å eliminere kjøpet slik at summen ikkje blir trekt frå bankkonto. Under betalingsprosessen er det også mogeleg å hoppe over betaling for å gå direkte til kvittering. Kjøpet vil då bli avbrote og terminalen går tilbake til å vente på metode. Når administrator lukkar programmet, vil kommunikasjon mellom terminal og CampLett-programmet stoppe.

9.4 Database

CampLett-programmet er bygd rundt databasen, og det er der informasjon blir henta og lagra. Databasen består av to tabellar. Den eine tabellen blir brukt for å hente informasjon slik som pris på overnatting, medan den andre tabellen lagrar data om bestillinga til kunden. Desse to tabellane blir knytt saman med ein primærnøkkel på kvar av kolumnenummera med namnet «Nr». Kundane blir registrert i tabellen etter kva nummer dei vel under innsjekking.

Tabell 1 Utsnitt av tabell for campingplassdata

Nr	Boenhet_id	Boenhet	St_relse	Pris
1	1	Hytte	4	1350
2	1	Hytte	4	1350
3	1	Hytte	4	1350
4	1	Hytte	4	1350
5	1	Hytte	4	1350
6	1	Hytte	4	1350
7	1	Hytte	4	1350
8	2	Telt	NULL	100
9	2	Telt	NULL	100

Tabell 1 inneheld informasjon om kva type plassar ein kan leige. Vi har tre val som er campingvogn, telt og hytte. I den fyrste tabellen ligg og informasjon om kor mange ein har av kvar eining og kva prisen er i døgnnet for leige. Denne informasjonen les Java-programmet og brukar for å vise riktig informasjon på dei grafiske skjermbildra.

Tabell 2 Utsnitt av tabell for kundedata

Nr	innsjekk	utsjekk	Fornavn	Etternavn	Total_Sum	Betlaing_godkjent
1	NULL	NULL	NULL	NULL	NULL	NULL
2	NULL	NULL	NULL	NULL	NULL	NULL
3	NULL	NULL	NULL	NULL	NULL	NULL
4	NULL	NULL	NULL	NULL	NULL	NULL
5	NULL	NULL	NULL	NULL	NULL	NULL
6	NULL	NULL	NULL	NULL	NULL	NULL
7	NULL	NULL	NULL	NULL	NULL	NULL

Java-programmet brukar Tabell 2 til å lagre informasjon etter ei bestilling er fullført. Det blir registrert namn, lengde på opphald, totalsum og om betalinga er godkjent.

9.5 Installasjon av CampLett-programmet

Når CampLett-produktet skal installerast, må ein serviceteknikar installere MySQL, databasar og CampLett-program på ein PC med Windows operativsystem. Ein pc med berøringsskjerm er å anbefale.

For å starte programmet, må personell først starte MySQL som administrator og deretter starte CampLett. I oppstarten av programmet vil det komme ein alarm dersom personell har gløymt å starte databaseprogrammet. For fullstendig brukarmanual sjå vedlegg 1.

10.0 Utvikling av CampLett-programmet

Målet med arbeidet var å lage ein prototype av eit innsjekkingsystem til campingplass. Frå undersøkingsperioden hausten 2014 saman med forprosjektet hadde vi ein plan på korleis systemet skulle sjå ut. Dette var ei relativt grov skisse beståande av stort sett det grafiske oppsettet. Programmeringspråket, databasesystem og maskinvare var også planlagt i forprosjektet.

Dette kapittelet går nærmare inn på korleis gruppa har løyst utviklinga av programvaren til prototypen. Det vil bli forklart korleis vi har løyst dei ulike delane i programvaren frå planlegging til testing og feilretting.

Vi delte programvaren i fire hovuddelar, slik at kvar av dei fire gruppemedlemene fekk sitt eige hovudansvar. Dette viste seg etter kvart å verte ei naturleg inndeling, sjølv om nokre av delane vart meir omfattande en andre.

Grafikk

Design den grafiske utsjånaden til programmet.

Logikk

Lage logikk for kommunikasjon mellom delane, kalkuleringar og berekningar av nødvendige verdiar for å hente nødvendig informasjon.

Database

Lage ein database i MySQL som passar til ein tenkt campingplass.

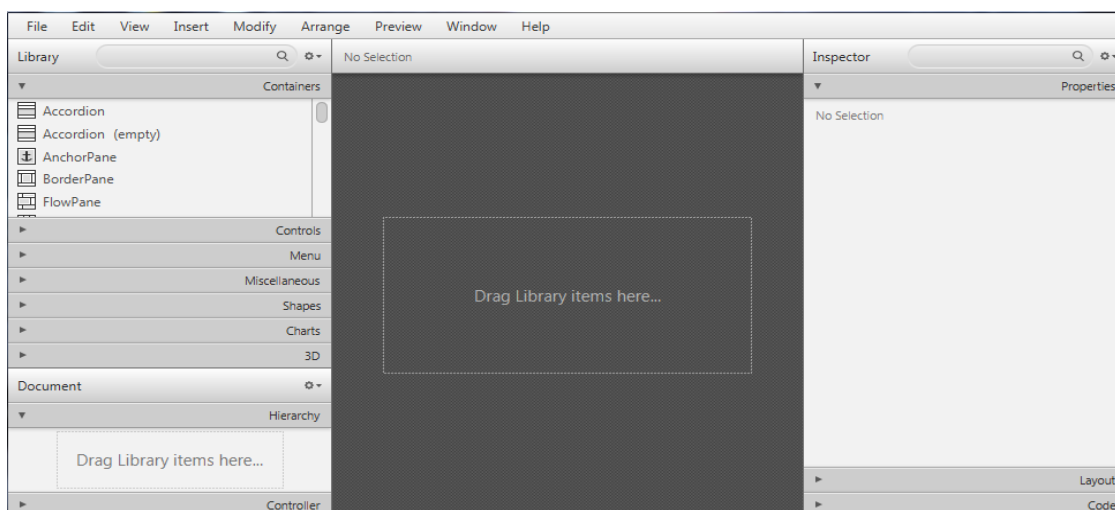
Tilkopling til bankterminal

Få kontakt mellom eit Java-program og bankterminalen, og eventuelt sjå på andre betalingsløysingar.

10.1 Utvikling av grafisk design

Det vart tidleg bestemt at den grafiske delen skulle lagast i JavaFX og ved hjelp av FXML-filer som vi ville konstruere i Scene Builder. Dette hadde gruppa god kjennskap til frå tidligare undervisning, lab-arbeid og utforsking.

Scene Builder er eit visuelt verktøy som gjer det raskt for brukaren å utforme det grafiske miljøet. Der kan ein hente grafiske element med musa og plassere dei der ein vil ha dei i bildet. Programmet stiller med eit stort kategorisert lager av grafiske komponentar som står Java til rådighet. Figur 17 viser eit arbeidsområde som brukaren kan dra og sleppe komponentar på, og samstundes endre deira eigenskapar. FXML kodane blir automatisk generert i bakgrunnen [25].



Figur 17 Arbeidsbenken i Scene Builder

Det første som vart nytt for oss, var at vi hadde behov for å byte ut FXML-filene på skjermen etter kva skjermbilde som skulle visast. Vi undersøkte dette temaet på internetforum som www.stackoverflow.com [26] og liknande. Vi fann fleire meiningar om korleis det burde gjerast.

Vi fann eit godt eksempel på Oracle-bloggen til Anglea Cacedo. Ho har utvikla eit kasino i JavaFX med fokus på fin grafikk og transaksjonar mellom skjermbilder [23], akkurat slik vi hadde tenkt oss. Ho hadde også lagt ut alle kjeldekodane sine for fri bruk. Dei var relativt enkle å sette seg inn i og fungerte bra. Vi brukte dette som eit utgangspunkt for det grafiske rammeverket.

Etter kvart såg vi eit problem med rammeverket. Vi fekk problem med å laste inn sanntidsinformasjon til dei grafiske bilda. Dette såg vi når ein verdi vart sett i det eine skjermbilde og skulle visast i neste skjermbilde. Då fekk vi den gamle null-verdien. Etter feilsøking konkluderte vi med at rammeverket lasta FXML-filene og tilhøyrande Control-filer i starten av programmet. Det medførte også at all informasjonen vart lasta inn i starten i staden for når skjermbilde viser på skjermen. Dette var motseiande i forhold til det vi hadde forstått frå dokumentasjonen til Cacedo.

Vi såg for oss å løyse dette enten ved å bygge om rammeverket, eller å bruke ei form for «eventlistener» som kunne merke at det aktuelle bildet vart vist og deretter laste inn nødvendig informasjon.

Løysinga vart å bygge om rammeverket slik at skjermbilda vert tekne heilt vekk frå systemet og lasta inn på nytt når dei skal visast. Denne modifikasjonen gjorde at vi fekk lasta inn ny informasjon når eit skjermbilde vart vist på skjermen.

Dei grafiske skjermbilda vart laga i JavaFX Scene Builder og kopla saman med Control-file i Java. Dette var ein tidkrevjande prosess, samtidig som det var her dei fleste endringane måtte gjerast etter testperioden.

Etter testen hadde vi mange tilbakemeldingar på at det burde være mogeleg å trykke på både flagget og knappen når ein skal velje språk i starten av innsjekkinga. Det var også eit liknande ynskje i det tilfelle at ein skal velje bualternativ. Her er det også eit bilde som burde utføre den same kommandoen som knappen. Vi løyste dette ganske enkelt ved å gi bilda den same FXML-ID-en som den tilhøyrande knappen. Slik vil den same metoden bli køyrd, og ein slepp å skrive Java-kodane på nytt.

Eit problem som oppstod undervegs, var val av plass i kartet. Her skal plassane som viser opptatt i databasen bli markert med fargen raud, og samtidig skal dei plassane som er ledige gå frå grønne til raude når dei verte trykt på. Om kunden ombestemmer seg og heller vil ha ein anna plass, vil fortsatt den valte knappen vise raudt, og det vil være uråd å vite kva plass ein eigentleg har valt.

Løysinga på det vart at den metoden som i utgangspunktet lastar inn alle dei ledige plassane og sett dei grønne når skjermbildet vart lasta vert køyrd også kvar gang ein knapp blir trykt på.

På den måten vil skjermbildet oppdatere seg i forhold til databasen kvar gong ein knapp blir trykt på og ikkje berre i starten.

10.2 Utvikling av Java-basert logikk i CampLett-programmet

Dette kapittelet tar for seg utfordringar, problemstillingar og løysingane vi kom fram til under arbeidet med logikken bak CampLett-programmet.

10.2.1 Kommunikasjon med database

Vi laga ein eigen klasse for kommunikasjon med MySQL- databasen. Denne klassen inneheld både metodar for å kople til databasen og for å hente ut eller sette informasjon i databasen. Vi hadde gjort liknande ting i faget Industriell-IKT. Kodar som vi hadde frå før kunne brukast med enkle endringar og fungerte nesten med ein gong.

10.2.2 Språkpakke

For å presentere programmet på fleire språk, måtte all skrift som skulle synast på skjermen samlast i variablar. Dei aktuelle variablane måtte hentast inn i grafikken etter kva språk som var valt. Grappa sin første ide var å ha eit sett String-variablar liggande i ein klasse og når språket vert valt skulle kvar av variablane settast til det valte språket. Det ville då bli ei omfattande prosedyre akkurat når ein vel språket, men enkelt å referere til frå den grafiske delen.

Etter kvart kom det fram ei løysing med å lage ein klasse som inneheldt alle dei aktuelle variablane med konstruktør, get-metodar og klassar for kvart språk. Desse arva eigenskapar frå den første klassen. Sjå Kap.9.1.6. Dette gav oss eit enkelt og ryddig system. Bakdelen var at det vart litt komplisert å legge til nye setningar etter kvart, då dei måtte leggest til fleire plassar. På ei anna side fekk vi fordelen av feilmelding frå «compilatoren» i NetBeans om ikkje variablane stemte overeins. Dette kjem av arvestrukturen og gjorde at vi kunne vere sikre på at vi hadde med omsetjing for alle språk.

10.2.3 Logikk for det grafiske

Vi bestemte at den grafiske delen av programmet skal gjere minst mogleg arbeid. Med dette meinte vi at alle operasjonar den grafiske delen treng å utføre skal ligge i ein klasse som alle Control-filene har tilgang til. På den måten kan metodane brukast fleire stadar, og vi lagar eit ryddig miljø, sjå dette nærmare i kap. 9.1.5. Etter kvart såg vi at det var ein del operasjonar som var hensiktsmessig å lage direkte i Control-filene, men i hovudsak vert alle større operasjonar utført i objekt av «MainClass».

Vi bestemte oss for å ha ein Interface på «MainClass» for å halde oversikt over alle metodane. Utgangspunktet for denne «interfacen» vart skriven ved å gå gjennom planane frå forprosjektrapporten. Når vi følte vi hadde med det som var nødvendig i «interfacen», kunne vi enkelt autogenerere alle metodane i «MainClass» og skrive logikken i dei etter kvart.

Vi var førebudde på at det kom til å komme opp ting vi ikkje hadde tenkt på etter kvart. Det vart både lagt til, fjerna og endra på metodar undervegs. Blant anna fann vi ut at vi måtte lage ein eigen metode for å konvertere dato frå det Java gav oss og til ein verdi som MySQL kunne forstå.

10.2.4 Skjermtastatur

Vi har hatt store utfordringar med å styre tastaturet. Det var eit ynskje å kunne bruke eit tastatur på skjermen for å sleppe å ha noko fysisk. På same måte som eit nettbrett eller ein mobiltelefon ville vi at det skulle komme fram når det var behov for det.

Det første vi undersøkte var om det var mogleg å styre tastaturet «osk.exe» som ligg i Windows som standard. Vi fann ut at vi kunne bruke klassen «Runtime» for å køyre program utanfor Java. «Runtime» kan også køyre «CMD-kommandoar» [18], og vi trudde derfor at vi kunne bruke kommandoen «taskkill» for å lukke tastaturet. Dette viste seg ikkje å fungere. Grunnen til det var at «osk.exe» vert køyrd frå systemfilene til Windows og krev derfor administratortilgang for å bli avslutta. Java-programmet vårt har ikkje administratortilgang.

Vi lasta ned eit anna tastatur og installerte det på maskina. Dette kunne vi styre utan administratortilgang. I forhold til design såg ikkje dette tastaturet bra ut, og det vart derfor bestemt å jobbe vidare for å finne ei betre løysing.

Vi fann etter kvart endå eit tastatur på Windows som heiter «TabTip.exe». Dette tastaturet hadde fint grensesnitt. Det hadde også andre fordelar som at vi kunne justere inn størrelsen på det og flytte det rundt på skjermen. Etter at vi avslutta og starta det igjen, hadde det fortsatt den valte plasseringa.

Å avslutte «tabtip.exe» viste seg å vere like vanskelig som for «osk.exe». På www.stackoverflow.com [27] og andre forum var dette diskutert mykje. Det virka som mange hadde det same behovet som oss, og det flørte mange meiningar og forslag men

ingen svar. Vi fann ut at Windows sender noko som heiter «PostMessage», og desse meldingane blir brukte til å styre det meste i Windows. Det skulle også vere mogeleg for oss å generere slike meldingar. Vi fann mange eksempel på det i språket C# og C++, men sidan Java er eit språk som er uavhengig av operativsystem, er det ikkje klargjort for å kommunisere med Windows på den måten. Etter kvart fann vi biblioteket Java Native Access (JNA) som gjorde det mogeleg for oss å kommunisere med Windows via PostMessage [19].

PostMessage krev forskjellige parameter for å vite kva program som skal behandlast og kva som skal gjerast med det. Ved hjelp av ein metode i JNA kalla «Findwindow», går programmet gjennom vindauga som køyrer i bakgrunnen og samanliknar dei med det du søker etter. Denne metoden krev at du sender eit klassenamn på det vindauge du søker for så å samanlikne. Klassenamnet på «TabTip.exe» fant vi ved å bruke eit gratis program som heiter «WinID» [28].

Vi vart i stand til å avslutte dei fleste program på maskina via Java. «Tabtip.exe» var ikkje eit av dei og vi forsto ikkje kvifor. Resultata våre stemte igjen med forum, «TabTip.exe» vart blant anna referert til som eit «spøkelse» i Windows.

Vi fann til slutt grunnen til dette. Grunnen var at kommandoen som vi sendte til «Tabtip.exe» er ein kommando om å avslutte vindaug. Problemet er at når ein vel å avslutte «Tabtip.exe», blir vindaug i realiteten ikkje avslutta, det vert gjømt på sida av skjermen slik at det skal være enkelt å hente fram igjen.

Løysninga vart å sende en systemkommando med eit parameter «SC_CLOSE» via «PostMessage» til Windows. Dette betyr i praksis at vi kommanderer Windows til å trykke på «X» i vindaug. Denne parameteren var ikkje tilgjengelig i JNA biblioteket, men det kunne refererast til ved en integerverdi. Denne verdien kunne Google hjelpe oss med å finne [29].

Det vart også diskutert under prosessen å lage eit tastatur sjølve i Java, eller å bruke noko som var ferdig laga. Vi fann ikkje noko som passa med vårt design. Fordelen med å lage tastaturet sjølve hadde vore at vi kunne avgrense det til dei absolutt nødvendige tastane, for eksempel bokstavar. Bakdelen er at det hadde vore vanskelegare å implementere fleire språk. I «Tabtip.exe» har vi moglegheit til å velje språk. Gruppa sin grafikkansvarleg var også overarbeidd på dette tidspunktet.

10.3 Utvikling av databasesystemet til CampLett

For å strukturere dei data CampLett-programmet skal bruke, blei det oppretta ein database i MySQL med Workbench som plattform. I dette kapitlet blir det forklart korleis databasen er bygd opp, og kva hjelpemiddel som er brukt i den samanheng.

Bakgrunnen for valet av MySQL som databaseverktøy er at dette er den mest brukte databasen i verda og programvaren er gratis [5]. Tidlegare i studiet har vi fått kjennskap til MySQL og vi har valt å bygge vidare på den kompetansen.

I dette prosjektet vart det brukt ein lokal database som inneber at databasen er lagra lokalt på maskina. Bakgrunnen for å bruke ein lokal database er at i dette prosjektet er det berre ei maskin som skal ha tilgang til databasen, og det hadde sett strengare krav til vern av databasen om den låg på nett.

Databasen er utvikla frå eit Excel-dokument som blei eksportert til MySQL. Det er ulike måtar å gjere det på, og vi valde å bruke eit nettbasert program som heiter «database import scriptor» [30]. Programmet konverterar cellene i det valde Excel-dokumentet til MySQL kodar. Grunnen til at vi valde å bruke denne metoden, var at det gjorde jobben raskare, og ein slapp å kode alt i tabellen.

Når MySQL-kodane var generert, vart alle variablane konvertert til integer. Det vart då nødvendig å endre desse til passande variablar, for eksempel må «namn» ha variabel «varchar». Denne løysinga fungerte for vår del, og det blei derfor ikkje brukt tid på å sjå på andre metodar.

10.4 Tilkopling mellom bankterminal og Java

For å finne ut av kva reglement som gjaldt for å implementere ein bankterminal i vår prototype, tok vi hausten 2014 kontakt med Microlog, sjå vedlegg 9. Dei fortalte oss at så lenge terminalen er sertifisert, så treng ein ikkje å halde seg til noko meir enn godkjent innsynsbeskyttelse samt krava i «Universell utforming» når den skal implementerast. Sidan terminalen er produsert av Ingenico og allereie er EMV-sertifisert, vart det fokusert på å finne ut av kommunikasjon mellom programmet og terminalen.

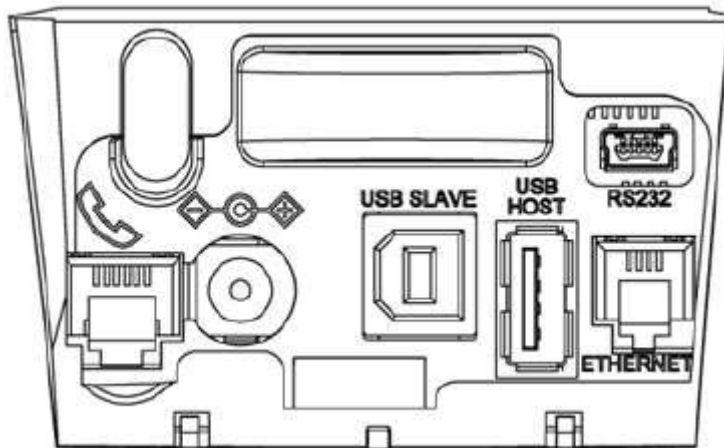
Dette kapittelet tek for seg to ulike løysingar som vi kom fram til, Javacard-plattform i Netbeans og JBAXI-protokoll som vi fekk tilgang til av Nets.

10.4.1 Undersøking av kommunikasjon

For å finne ut av kva type kommunikasjon vi måtte bruke mellom programmet og bankterminalen, undersøkte vi kva dokumentasjon leverandørane leverte. Nets og Ingenico har opne produktark og dokumentasjon som omhandlar terminalen, sjå vedlegg 12.

ICT-220 har fleire COM-portar som kan nyttast til ulik bruk. Første alternativ er ein port med mini-USB endestykke i terminalen og DE-9 i kassesystemet. Ein kan også bruke ein av to USB-portar til sending av informasjon. Terminalen har ein master USB-port (USB A) som ein kan kople opp imot skrivar, skannar osv., og då verkar terminalen som ein vert for eksterne einingar. Den har også ein USB slave port (USB-B) som ein kan kople opp mot pc for å sende data. Den krev ein USB-drivar slik at terminalen blir registrert som ein virtuell COM-port.

Gruppa ønska å teste kommunikasjon med dei datamaskinene som programmet vart utvikla på. Dei er ikkje levert med DE-9 portar, så det måtte enten brukast ein adapter med USB til DE-9 kontakt, eller kommunikasjonen måtte gå gjennom slaveporten. Valet landa på å ha kommunikasjon gjennom slaveporten som ein ser midt i bilete på figur 18. Den kan brukast så lenge det er installert ein drivar for eininga på datamaskina. Drivaren for USB tilkopling har Nets lagt ut på si eiga heimeside tilgjengelig for nedlastning [31].



Figur 18 Baksida av ICT-220 terminalen med alle inngangar.

Frå figur nr. 19 ser ein bilete av den kabelen som er nytta mellom slaveporten og panelpc-en i prosjektet. Dette er ein USB-A til USB-B -kabel. RS-232 protokollen for kommunikasjon er nytta sjølv om det er brukt pluggar og portar frå USB-standarden.

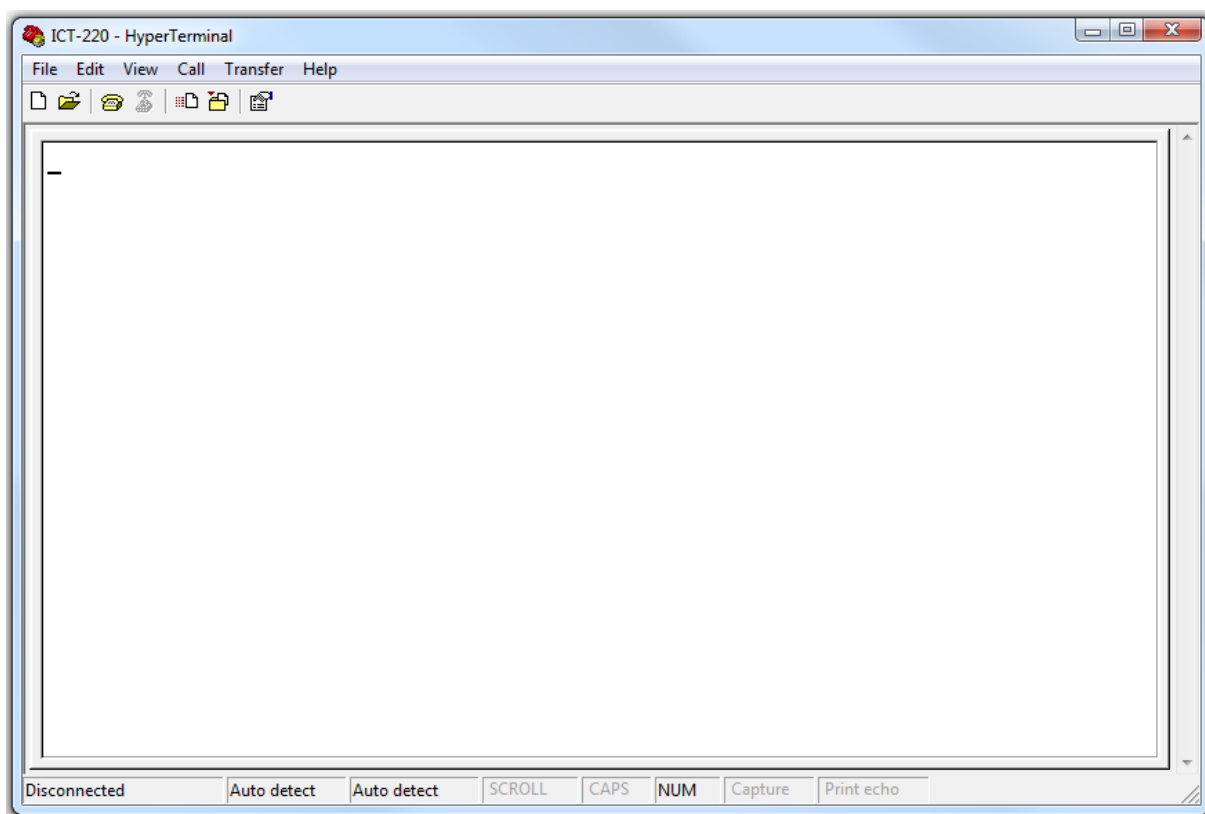


Figur 19 USB B (t.v) til USB A (t.h).

Då terminalen var registrert som ein virtuell COM-port på pc-en, måtte gruppa finne ut kva kommunikasjonsprotokoll som skulle brukast slik at programmet kunne sende den informasjonen ein treng. Ifølge drivarinstallasjonsmanualen til terminalen måtte ein oppdatere til baxi.net-versjon. Dette er filer som er bindeledd mellom terminal og kassesystem. Desse fekk vi tilgang til etter samtale med Nets seinare i undersøkinga. BAXI er ein termologi som ein brukar for heile BAXI-pakken.

10.4.2 Kommunikasjonstest med hyperterminalar

Tidlig i prosjektet var ikkje protokollen for kommunikasjon utlevert frå Nets. Det blei derfor forsøkt å sende ulike data til terminalen gjennom USB-portane frå ei datamaskin for å sjå om terminalen kunne lese enkle bit-signal. Det vart gjort forsøk der det blei brukt to ulike hyperterminalar, samt eit Java-program som var utvikla ved eit tidlegare prosjekt, for å sende serielle signal med ulike verdiar i RS-232 protokoll. Det var ingen respons frå terminal, og gruppa konkluderte med at det mangla ein «handshake». Det vil seie godkjenning frå terminalen slik at den opna for kommunikasjon. Det var tankar om at det kunne vere den tidlegare nemnte BAXI-fila som burde vore implementert. Forsøk nummer to var å kople opp terminalen med ein panelpc som hadde DE-9 port. Forsøket med å sende serielle bits til terminalen frå ein hyperterminal blei gjentatt, men her var det heller ingen respons.



Figur 20 Hyperterminal i Windows

Hyperterminalen som vist i figur 20, er programvare som var installert fast på eldre OS, men er ikkje på Windows 7 og nyare. Det er programvare som ligg gratis på nett for nedlasting slik at nyare OS også kan bruke det. Gruppa har hatt ein del erfaring med det tidlegare frå laboratorieøvingar og prosjekt med RS-232 kommunikasjon. Ved hjelp av dei programma kan ein sende serielle bit, bestemme hastigheit på sending og ein del andre kriterium.

10.4.3 Alternativ 1 - Javacard

Javacard er ei plattform vi har tilgang til i NetBeans, og den har bibliotek for å utvikle program som skal kommunisere med smartkort [32]. Dagens bankkort blir kalla smartkort på grunn av chipen som er installert i dei. Tidlegare hadde alle betalingskort berre ei magnetstripe som ein brukte i betalingssamanheng. Den lagra enkle data som ein må bruke for å fullføre ein transaksjon. Dette var ein usikker måte å lagre data på, så smartkort vart utvikla og levert med ein chip som var mykje sikrere. Ved å utvikle smartkort kom ein ny sikkerheitsprotokoll, gjorde at det ikkje lenger var like enkelt å svindle eigaren av kortet [33].

Parallelt med undersøkinga av BAXI, blei det undersøkt kva som måtte til for å sette opp eit program ved hjelp av Javacard. Det blei konkludert med at den tida som var til rådighet var for lita til at gruppa kunne sette seg inn i to ulike løysingar av kommunikasjon til terminalen, eventuelt kunne det utvidast med Javacard dersom ein hadde hatt tid til overs. Prosjektet vart derfor avgrensa til den informasjonen som Nets gav oss.

10.4.4 Alternativ 2 - BAXI

Intuit developer har lagt ut ein API for ein anna type Ingenico terminal, Ingenico 3070, som omhandlar kommunikasjon rundt terminalen [34]. Den tek for seg spesifikke kodar som tilseier om den skal ta imot data eller sende «handshake». Gruppa fekk bekrefta dette av ulike firma som lagar programvare til bankterminalar via telefon og e-post, sjå vedlegg 4. Tilbakemeldingane sa at Nets kunne gje tilgang til dei rammeverka som trengst for å opprette kommunikasjon mellom terminal og ECR(Electronic Cash Register).

Etter samtale med Nets, fekk gruppa tilgang til to ulike protokollar for kommunikasjon. Den første protokollen, DFS-13, krev mykje programmering og er veldig tidkrevjande. Denne protokollen er hovudsaklig brukt på gamle kassesystem utan skjerm som ikkje har mulighet til å nytte BAXI. Det andre alternativet, som er nemnt tidlegare, er BAXI-protokollen. Den er mykje meir brukarvenleg og enklare å programmere, derfor valde vi den.

Nets har all data for kommunikasjon i ein FTP-portal som er lukka med brukarnamn og passord. For å få tilgang til den måtte gruppa inngå ein NDA-avtale med Nets, sjå vedlegg 5. Dette er ein avtale som sikrar at informasjon som blir delt mellom partane CampLett og Nets, ikkje blir delt utanfor dei vilkåra som er bestemt av avtalen.

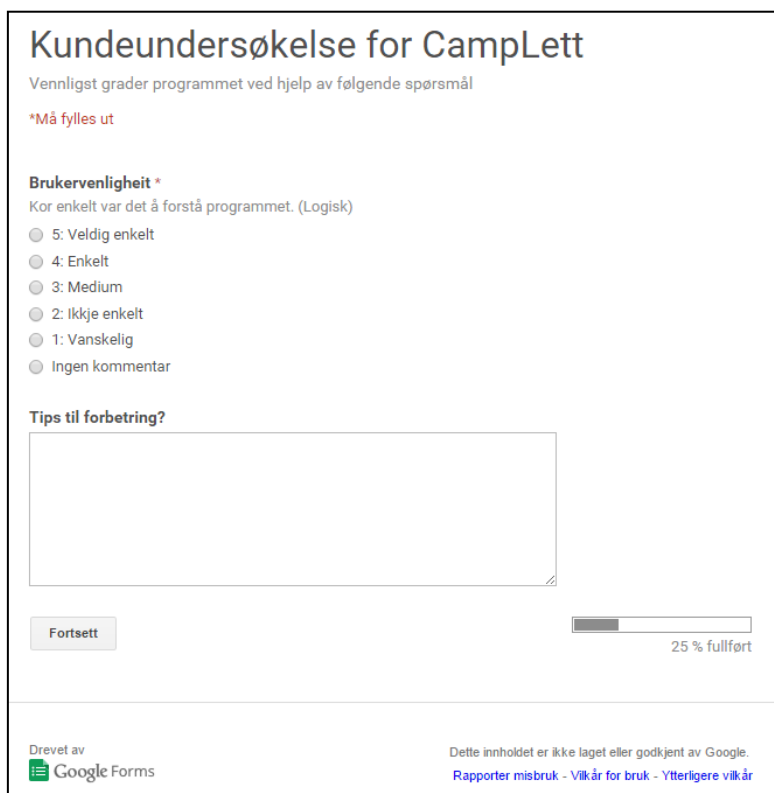
Gruppa hadde dialog med Nets si kundeavdeling som driv med utvikling og feilsøking av programvare opp mot bankterminal. Dei gav tilgang til den lukka FTP-portalen etter at avtalen var underskriven. Her fekk vi tilgang til dei filene og dokumenta som måtte til for å kople saman programvaren vår og bankterminalen. Det innebar eit bibliotek og konfigurasjonsfiler som fungerte som mellomledd mellom terminal og program. Utvikling av metodar og løysingar i Java gjorde gruppa sjølv.

Dei filene vi hadde behov for på FTP-serveren til Nets er som nemnt beskytta av ei NDA-avtale, og vi kan derfor ikkje gå djupare inn i korleis desse filene er bygde opp, men ein kan sjå meir om kva dei er brukte til i kap 9.3.

10.5 Test av prototype

For å teste og få tilbakemelding på programmet, gjennomførte gruppa først ei kundeundersøking. Den blei utført i veke 16 ved HiSF, avdeling Førde. Undersøkinga tok for seg ei fiktiv kundegruppe som bestod av elevar og tilsette ved HiSF, og dei fekk moglegheit til å sjekke inn ved hjelp av CampLett-programmet. Prosjektgruppa har også hatt ein dialog med Kjørnes Camping i Sogndal gjennom prosjektet, for å få idear og tilbakemeldingar om produktet. I veke 19 tok gruppa turen til Kjørnes for å utføre ein slutttest.

Under testperioden vart prototypen sett opp i kantineområdet ved HiSF. Der kunne forbigjerande teste programmet på panelpc-en, for så å gje tilbakemelding i ei kundeundersøking via «Google Docs» som var satt opp på ein pc ved sidan av. Fokuset i testen var å undersøke om programmet var brukarvennleg, så det var viktig at dei som utførte testen klarte å gjennomføre utan hjelp. Målet med programmet er at det skal vere sjølvbetjent.



Kundeundersøkelse for CampLett
Vennligst grader programmet ved hjelp av følgende spørsmål


*Må fylles ut

Brukervenlighet *
Kor enkelt var det å forstå programmet. (Logisk)

- 5: Veldig enkelt
- 4: Enkelt
- 3: Medium
- 2: Ikkje enkelt
- 1: Vanskelig
- Ingen kommentar

Tips til forbetring?

25 % fullført

Drevet av  Google Forms

Dette innholdet er ikke laget eller godkjent av Google.
[Rapporter misbruk](#) - [Vilkår for bruk](#) - [Ytterligere vilkår](#)

Figur 21 Utsnitt av kundeundersøkinga frå Google Docs

Som ein ser av figur 21, kjem brukaren til eit enkelt spørjeskjema som dei fyller ut. Alle svara frå undersøkinga blir automatisk lagra og oppdatert i eit Excel dokument på Google sine serverar, og prosjektgruppa fekk då heile tida tilgang på svar og statistikk frå undersøkinga.

Løysinga vi har valt til testinga samlar alle resultata frå testen og gjer dei lett tilgjengelige. Ved å bruke «Google Docs», sparte gruppa tid og fekk dokumentert alle tilbakemeldingane fortløpande.

Etter kvar undersøking vart det retta opp i dei fleste elementa som var nemnt, slik at den utbetra versjonen av programmet var klar til neste test. Gruppa vart delt i to under testen. Der den eine delen stod for kundeforhold, og den andre stod for programmering og feilretting.

	A	B	C	D	E
1	Tidsmerke	Brukervennlighet	Tips til forbedring?	Design	Tips til forbedring?
20	14.04.2015 kl. 13.19.41	5: Veldig enkelt	Tilbakemelding på trykk, på antall døgn.	5: Bra.	
21	14.04.2015 kl. 13.20.45	4: Enkelt	litt språk vask	5: Bra.	
22	14.04.2015 kl. 13.26.51	4: Enkelt	Forlaring fargekodar opptatt ledig for ein trykker. Plasse	4: Ok. Litt småplukk	Sjå tidlegare
23	14.04.2015 kl. 13.29.02	5: Veldig enkelt		4: Ok. Litt småplukk	
24	14.04.2015 kl. 13.30.44	4: Enkelt	Godta knapp tilgjengelig etter at du har srevet på tastatur	4: Ok. Litt småplukk	
25	14.04.2015 kl. 13.35.07	5: Veldig enkelt	Særs godt	5: Bra.	
26	14.04.2015 kl. 13.41.26	4: Enkelt		4: Ok. Litt småplukk	det tok litt tid før jeg forstod at jeg måtte krysse ut tastaturet.
27	14.04.2015 kl. 13.45.04	5: Veldig enkelt	Når en skal skrive navn, burde pilen komme på fornavn Burde være mulighet for flere dager enn 4. Kanskje sorter etter alfabetet eller pris.	4: Ok. Litt småplukk	

Figur 22 Utsnitt frå Excel dokumentet, med tilbakemeldingar frå kundeundersøking

Excel- skjemaet figur 22. syner når tilbakemeldingane vart registrerte. Sidan det samstundes vart jobba med oppdateringar kunne ein sjå om tilbakemeldingane endra seg etter at det kom ny programvare på teststasjonen.

Statistikken over tilbakemeldingane synte at det var dei same problema eller ønska som gjekk igjen. Eit godt eksempel er tastaturet som var vanskelig å fjerne. I første versjon av programmet måtte ein trykke på «X» i hjørnet på tastaturet for å lukke det før ein kunne gå vidare. Dette var den største forbedringa som var gjort etter kundeundersøkinga, og ein kan lese meir om dette i kap. 10.2.4.

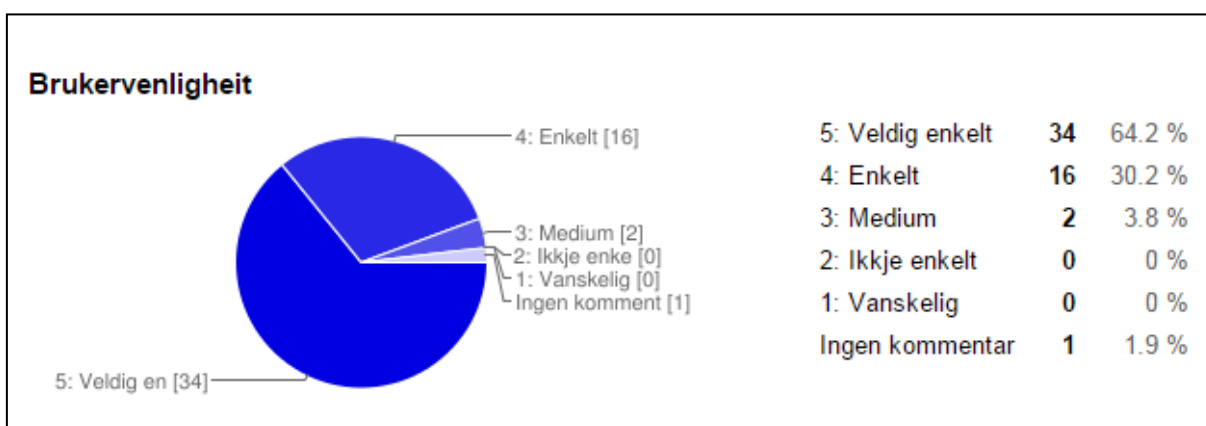
Under testen var panelpc-en heile tida kopl opp mot skulen sitt nettverk, slik at nye versjonar av programmet kunne lastast inn når større endringar var gjort. Teststasjonen vart då fortløpande oppdatert og nye funksjonar vart lagde til.

Etter at kundeundersøkinga og utbetringar var gjennomført, tok gruppa kontakt med Kjørnes Camping for å avtale tid til slutttest. Den vart sett til Torsdag i veke 19, og dei tilsette ved campingplassen skulle få moglegheit til å sjå og prøve ut prototypen.

Ved besøket i Sogndal fekk campingpersonale prøve å gjennomføre ei bestilling for så å gje tilbakemelding på om det var brukarvenleg. Kjørnes var veldig positive til prototypen, og meinte dette var eit program som kunne vore installert på campingplassen. Ved spørsmål om kva dei ville ha hatt med ved ei eventuell utviding, så nemnde dei det å velje tal personar som skal sjekke inn, fleire alternativ på hytter, prisforskjell og avstemming ved dag og månadsslutt.

10.5.1 Testresultat

Målsettinga med testen var å få tilbakemelding på brukarvenlegheita til produktet vårt. Figur 11 er eksempel på ein graf Google Docs kunne autogenerere frå resultatata. Utifrå dette kunne vi konkludere med at «CampLett» var brukarvennleg. Dette sidan 94,4% svarte at det var enkelt eller veldig enkelt å bruke. Av 54 testpersonar var det ingen som meinte det var vanskeleg eller ikkje enkelt.



Figur 11 Utsnitt av statistikken over brukarvenlegheit frå undersøkinga

10.5.2 Styrkar og svakheiter rundt testen

Testen har vore eit bra hjelpemiddel for å forbetre programvaren. Campingbransjen har vore nøgd med produktet. Det vart ikkje gjort noko endringar etter besøket hos dei. Ein svakheit ved at testen blei utført ved ein høgskule er at brukargruppa er forholdsvis lita, dei har tilnærma lik bakgrunn og det er liten aldersforskjell. Ved ein campingplass vil kundegruppa vere meir varierende.

11.0 Refleksjon rundt vidareføring av produkt

Etter tilbakemeldingane på prototypen vår, både frå testpersonar og campingbransjen, er det freistande å vidareføre prosjektet til eit kommersielt produkt. I så fall er det nokre punkt som må utviklast meir.

Bankterminal

I prototypen er det brukt ein Ingenico ICT 220 bankterminal. Denne er i utgangspunktet ikkje berekna på bruk utan personale til stades. Det vil vere naturleg å investere i ein terminal som er berekna på å bli montert fast og kan samanliknast med sjølvbetjent bensinpumpe. Nets leverer noko som heiter «iUN-terminal» som er berekna for sjølvstendige billettautomatar, sjå vedlegg 11. Desse hadde vore eit bra alternativ om prosjektet vert jobba vidare med.

Det kan bli nødvendig å endre BAXI-delen av programmet til å passe andre typar bankterminalar, men dette har vi ikkje undersøkt i dette prosjektet.

Kvittering

Kunden vil forvente å få ei kvittering på kjøpet sitt. I prototypen er kvitteringa synleg på skjermen etter betalinga er godkjend. Den burde bli skriven ut ved hjelp av ein tilkopla printer. Andre løysingar kan vere å sende den på e-post eller SMS til kunden.

Database

Databasen som ligg lokalt på teststasjonen burde flyttast til ein server som personell på campingplassen har tilgang til. Det burde også utviklast eit enkelt administrator-program på denne serveren som personell kan logge inn på for å få oversikt over inntekter, MVA-grunnlag og liknande. Det burde også vere tilgang for å endre pris på bueiningane her.

Administratordelen burde kunne overstyre databasen slik at eventuelle feil kan rettast og manuelle bestillingar kan leggest inn.

Automaten og serveren må vere kopla på eit lokalt nettverk utan internett for å ivareta sikkerheita for data. Om systemet skal koplast mot internett, må tilgangen til MySQL-databasen sikrast vesentleg meir en den er i dag.

Reglement

Lovvekt angående automatar og universell utforming må gjennomgåast på nytt ved eventuelt kommersielt bruk. CampLett-automaten er per no å rekne som ein prototype for å prøve marknaden og interessa, og der ikkje alle krav nødvendigvis er teke omsyn til.

I tillegg til desse punkta vil det vere nødvendig med spesialtilpassing på campingplassen for å tilpasse kart og tal på bueiningar. Det kan også diskuterast om ein skal lage ein enklare universell versjon der karta med spesifikke plassar utgår. Det er også mogleg å lage eit meir avansert administrator-program, der personell får tilgang til å legge inn kart og tal på plassar sjølv.

Under forundersøking hausten 2014 kom vi over eit campingkort. Dette blir brukt over heile Europa og er tatt i bruk på dei fleste campingplassar i Norge. Campingkortet inneheld all informasjon om eigar, slik at campingplassar berre treng å lese dette digitalt for å registrere kunden [35]. For å gjere CampLett endå enklare å bruke ville det vere ein god ide å kunne lese dette inn i vårt system.

Eit nettbasert system vil gje større fridom både for innehavarane av systemet og til brukarar. Det kan utviklast ein HTML-versjon av systemet på ei nettside og ein applikasjon, slik at det kan bestillast opphald på førehand via internett.

12.0 Konklusjon

Oppgåva til CampLett har vore å utvikle eit sjølvbetjent system for innsjekking på campingplassar. Som følgje av kostnadsanalysane for komplett system med bom ved inngang og maskinvare for bruk utandørs, som vart gjort hausten 2014 [36], vart oppgåva avgrensa til å utvikle programvare tilrettelagt for campingplass og ein teststasjon tilkople bankterminal. Desse elementa vart definerte som ein prototype.

Resultatet er ein fullt funksjonell automat som har grensesnitt via ein berøringsskjerm på 15 tommar, og ein Ingenico bankterminal. Skjermen er stor nok til å presentere store og oversiktlege berøringselement som gjer den enkel i bruk. I automaten er det mogeleg å velje hytte- og telt eller vogn –plassen som kunden sjølv ynskjer. Ein vel kor lang tid ein vil leige plassen for og tastar inn namnet sitt. Kunden må betale i bankterminalen med ein gong og får opp ei kvittering på skjermen når transaksjonen er fullført.

Prototypen gjer det mogleg for kunden å hoppe over betalinga eller å annullere den etter den er fullført. Dette er funksjonar som er laga for at testing av systemet skal kunne fullførast utan at testpersonen vert økonomisk lidande. Desse vala vert sjølv sagt fjerna i eit eventuelt kommersielt produkt.

Programmet er skriva i Java og er sett saman med tanke på utviding til eit fullskala produkt. Alle data vert lagra i database som ikkje er avhengig av Java. På den måten kan andre program få tilgang til bestillingsinformasjon. Programmet er laga i fleire delar og er programmert på ein oversiktlege måte, slik kan nye funksjonar leggast til utan at heile programmet nødvendigvis må endrast og testast på nytt.

Prototypen har vore testa over tre dagar i kantina på HiSF avdeling Førde. Her har det vore høgskulestudentar og lærerar innom og gjeve oss konstruktive tilbakemeldingar.

Testperioden var viktig for prosjektet, då produktet vart markant forbetra på få dagar.

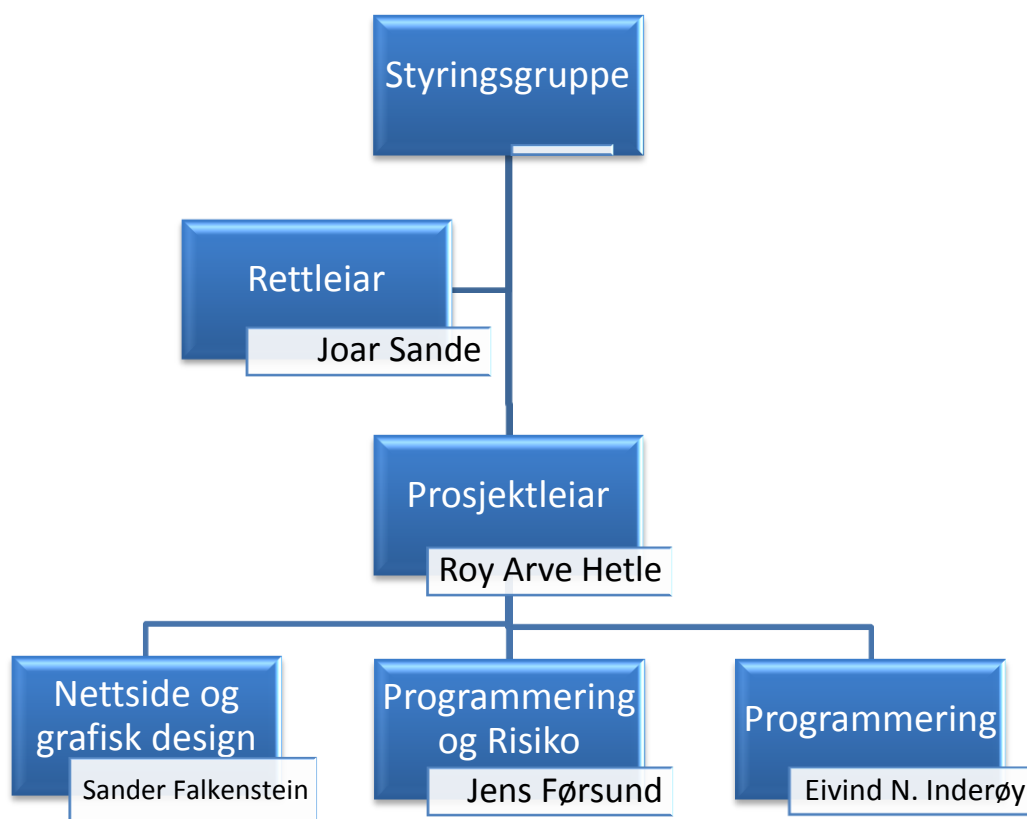
Utgangspunktet for programvaren er basert på marknadsundersøkingane vi gjorde hausten 2014. Vi har hatt kontakt med bransjen, spesielt Kjørnes camping i Sogndal, både i forprosjektet og i avsluttande del. Dei som har fått anledning til å prøve prototypen ser ut til å være imponert over resultatet og glade for at deira meiningar og forslag har vorte høyrde. Med dette konkluderer vi med at alle mål og delmål er nådde. Vi meiner vi har eit godt utgangspunkt for eit produkt til vidare utvikling.

13.0 Prosjektadministrasjon

I dette kapitlet beskriver vi korleis vi har gjennomført prosjektet som gruppe, og korleis vi har fokusert og fordelt arbeid.

13.1 Organisering av prosjektgruppe

I forprosjektet blei arbeidsoppgåvene fordelt i gruppa og organisasjonen blei som vist på figur 23. Gruppa var bevisst på at det kunne bli behov for å fordele om på arbeidet etter som ein fekk meir oversikt over kor tidkrevjande dei ulike delane av prosjektet ville bli. Dette er og teke med som ein del av risikovurderinga for prosjektet. Sjølv om det var mange usikre faktorar med tanke på tid, har det berre vore mindre justeringar i forhold til den opphavlige arbeidsplanen.



Figur 23 Organisasjonskart

13.1.1 Styringsgruppe

Sidan vi gjennomfører eit prosjekt utan oppdragsgjevar, har vi sjølve valt omfanget for prosjektet. Styringsgruppa har bestått av prosjektgruppa ilag med vår rettleiar ved HiSF-AIN, Joar Sande. I utgangspunktet skal styringsgruppa ha møte kvar fjortande dag, men gruppa og rettleiar blei einige om at det ikkje var behov for så hyppige møter.

13.2 Ressursbruk

I denne delen av rapporten beskriver vi korleis vi har disponert ressursane til prosjektet, det vil seie, kostnadane ved prosjektet og timebruken til gruppa.

13.2.1 Økonomi

Gruppa stod utan ekstern oppdragsgivar. Dette til tross for at ein gjennom prosjektet hausten 2014 og forprosjektet prøvde å få inn ein ekstern oppdragsgivar. Grunnen til dette var at campingplassane gruppa hadde kontakt med hadde avgrensa med tid og ressursar. I tillegg ønska dei å sjå ein prototype før dei ville gå vidare. Når gruppa bestemte seg for å ta prosjektet vidare, vart det teke omsyn til at ein fekk tilgang til mindre ressursar. For å halde kostnaden låg vart det fokusert på å utvikle ei programvare. I oppstarten av forprosjektet fekk vi beskjed frå rettleiaren vår at HiSF kunne gi støtte til nødvendige innkjøp. Dette gjorde at vi kunne auke budsjettet og fekk løyve til å kjøpe inn ein panelpc, noko som gjorde at vi kunne lage ein meir brukarvenleg HMI. Til resten av prosjektet har gruppa teke i bruk gratis programvare til utvikling. Til betalingsløyving blei det inngått ein avtale med Akutten studentkro, der vi fekk låne ein minibankterminal.

Tabell 3 Oversikt over rekneskap og budsjett

	Rekneskap		Budsjett		
	Inntekter	Utgifter	Inntekter	Utgifter	
Panelpc		8750		8750	
Operativ system PC		1680		0	
Reise		1122		800	
Støtte	11552		9550		
sum	11552	11552	9550	9550	
Resultat		0		0	
Differanse					2002

Som ein ser av tabell 3, har gruppa brukt 2 002 kr meir enn budsjettet. Dette var avklart med rettleiar før innkjøp vart gjort. Det er to årsaker til at det blei brukt meir enn budsjett. Det eine var at panelpc-en blei levert utan operativsystem og harddisk, og det andre var at

Kjørnes camping ligg forbi Sogndal retning Kaupanger, og dermed blei det meir køyring enn gruppa hadde rekna med.

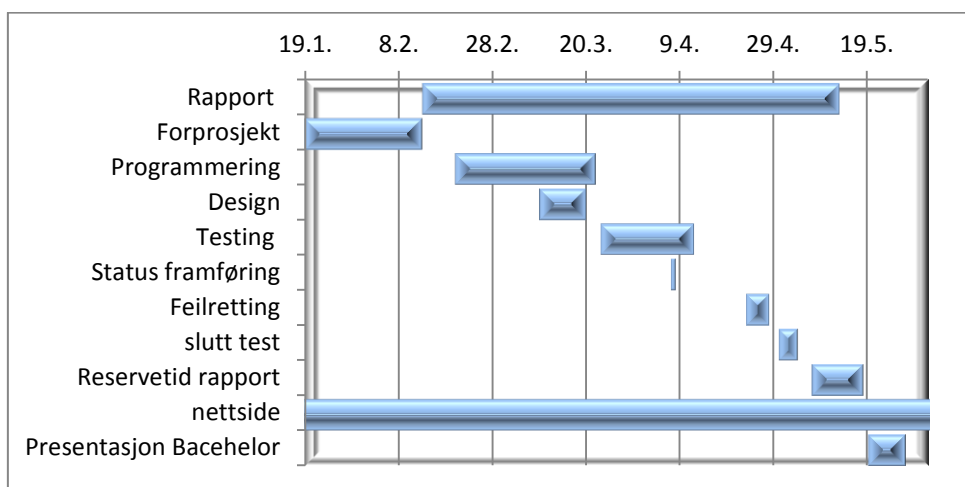
13.2.2 Timebruk i prosjektet

Gjennom prosjektet har gruppa hatt eit felles timedokument, der alle i gruppa har lagt inn kor mange timar som er brukt. Det viser seg ved prosjektslutt at timefordelinga har blitt veldig skeivfordelt. Det er mange mindre grunnar til det, men enkelt personar på gruppa har stor innsikt i programmering og har dermed måtte bidrege til å rettleie andre. Utanom dette har det vore private forhold som har vore til grunn for at ikkje alle har hatt moglegheit til å bidra like mykje.

Grensa som var sett frå skulen si side var 500 timar per student, ei gruppe på fire skulle då ha 2000 timar som maksimal timebruk. I forprosjektet var vi bekymra for å få høgt timetal og sette derfor avgrensingar på oppgåva for å ikkje få for mykje arbeid. Resultatet vi jobba mot blei oppnådd, med færre timar enn berekna. Vi brukte 1 628 timar totalt på gruppa, sjå vedlegg 3. I prosjektet står det igjen arbeid etter at rapporten er ferdig. Når prosjektet er ferdigstilt reknar vi med at gruppa har eit totalt timetal på ca 1 700 timar. På dei timane som står att skal det lagast plakat, leverast rapport, forberede og gjennomføre framføring. Utan om dette må heimesida oppdaterast med informasjon som skulen har satt krav til.

13.3 Gjennomføring av prosjektet etter tidsplan

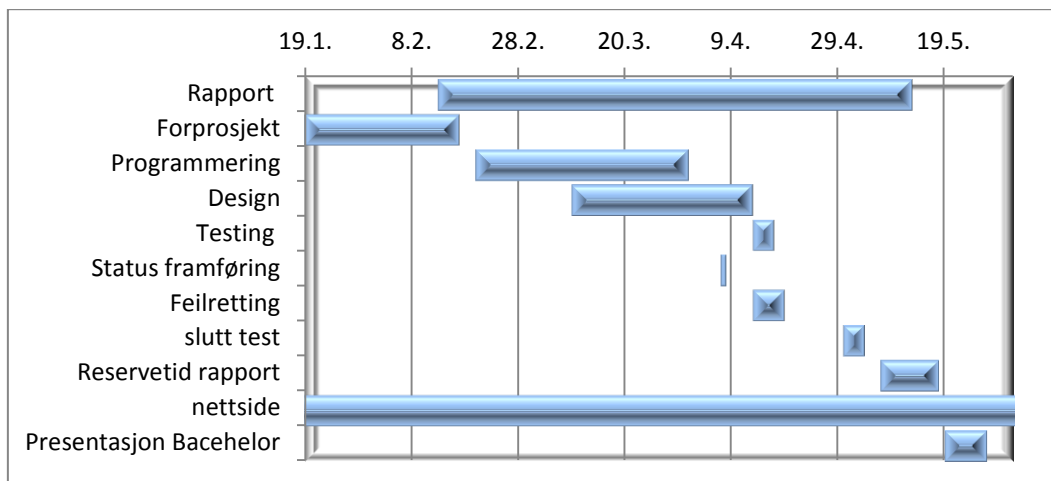
Under forprosjektet blei det sett opp eit gannt-skjema. Her blei det sett start og slutt -dato for viktige oppgåver i prosjektet. Ved å bruke gannt hadde gruppa oversikt over kva oppgåver som skulle vere ferdige til kva dato. Figur 24. viser når gruppa planla å starte med dei ulike delane og når dei skulle vere ferdige. Som ein ser blei det lagt inn tid mellom dei ulike oppgåvene når det nærma seg slutten av prosjektperioden. Grunnen til det var at gruppa ville ha moglegheit for å hente seg inn att om det skulle oppstå uventa problem.



Figur 24 Gannt-skjema fra forprosjekt

Figur 25 viser når vi kom i gang med oppgåvene og når vi blei ferdig med dei. Under utviklinga av programvaren og det grafiske designet hadde vi som mål å kunne teste prototypen i veke 13, men arbeidet med utvikling av program og design tok lenger tid enn planlagt. Det blei derfor teke kontakt med styringsgruppa for å få utsetjing. Det blei bestemt å utsetje testinga til veke 16.

Medan programmet blei testa, sat gruppa og utvikla oppdateringar slik at vi fekk testa dei ut fortløpande. På den måten blei feilrettinga gjort tidlegare enn planlagt og vi henta inn igjen store delar av tida vi hadde tapt på å utsetje testinga.



Figur 25 Gantt-skjema etter prosjektet er ferdig

13.3.1 Møte

For at alle i prosjektgruppa skulle vere oppdatert på kva dei andre jobbar med, vart det kalla inn til møte kvar veke i oppstarten av prosjektet. I periodar av prosjektet varierte behovet for å ha møte.

Styringsgruppa skulle ifølge framdriftsplanen halde møte kvar fjortande dag, men prosjektgruppa og rettleiaren Joar Sande vart einige om at det ikkje var behov for møte anna enn ved behov. Prosjektleiaren fekk ansvar for å kalla inn til møte i styringsgruppa. Med det meinte ein større endringar på prosjekt og/eller større avvik frå framdriftsplanen.

Til alle møte er det innkallingar og møtereferat, desse ligg i vedlegg 6.

13.4 Risikostyring

I dette kapitlet har vi basert ein del av metodane våre på NTNU sin Concept rapport nr. 12 [37]. Mykje av arbeidet er også basert på rapport for Sjølvstyrt camping 2014 [36].

For å få oversikt over dei elementa av usikkerheit eller risiko ved gjennomføring av prosjektet vårt, har vi valt å sette fokus på usikkerheit allereie i forprosjektet. Vi starta med ei idémyldring på farar og fallgruver. Dette var mykje basert på eigen erfaring med tidlegare prosjektarbeid.

13.4.1 Risikovurdering

Vi sette opp farane i ein tabell (sjå tabell 4) med sannsyn for at det inntreff og konsekvens om det skulle inntreffe, begge på ein skala frå 1 til 5. Produktet av dette utgjer kor mykje elementet trugar prosjektet vårt.

Det vart nødvendig å sette ei grense for kva nivå vi skulle sette i gang tiltak og ikkje. Vi valde å sette grensa for produktet på 9, då dette såg ut som ein naturleg inndeling i forhold til dei elementa vi hadde i tabellen vår. For dei elementa med produkt over 9, laga vi tiltak for forbetring og rekna ut eit nytt produkt. I somme tilfelle kan ei løysing rette opp fleire faremoment, medan andre element treng fleire løysingar.

Det er sjølvsagt ikkje alle farar og trugsmål ein veit om og kan førebu seg på. Det er derfor viktig at ein er open for å risikovurdere element som kjem til etter kvart i prosjektarbeidet.

Dei elementa av fare som ikkje utgjorde sum over 9 såg vi ikkje på som reelle farar. Men spesielt dei elementa med høg faktor av konsekvens og lite sannsyn kan vere skumle om dei inntreff. Derfor er det viktig at vi veit om dei på førehand.

Tabell 4 Utdrag frå risikoanalyse

Utfordring	Sans 1-5	Kons 1-5	Sum	Tiltak nr	Ny sans	Ny kons	Ny sum
For lite tid	2	5	10	a	1	5	5
Tekniske problem med innlevering	2	5	10	b	2	3	6
Feil mann/verktøy på oppgåva	3	4	12	c	2	3	6
Økonomisk ugjennomførbart prosjekt	1	4	4				4
Tiltak nr.	Tiltak for forbedring						
a	Begynne tidleg med faste møter og klare arbeidsmål						
b	Sette av tid til å levere tidlegare enn en fristen						
c	Må vere klare på oppgåvene og vere opne for å bytte på roller, hyppige statusmøter						

Tabell 4 viser eit utkast av risikoanalysa vår, og fullstendig analyse kan ein sjå i vedlegg 7. Vi ser av denne risikoanalysa at orden i dokument og god loggføring løyser mange av utfordringane våre. Det var også viktig for oss med profesjonelle møte, og at vi i samarbeid kom fram til klare arbeidsoppgåver og ansvarsområder. Dette hjalp oss til å halde fokus på kva som skulle gjerast. Det var godt å ha milepælar å strekke seg etter slik at vi ikkje utsette ting for mykje. For å få rapporten på eit profesjonelt nivå, var vi nøyddde til å bruke eksterne resursar på korrekturlesing. God planlegging av dette gjorde at korrekturleserane fekk gjort dette til avtalt tid. Vi såg også at føring av dagbok var viktig slik at heile gruppa hadde tilgang og kunne ta over arbeidet når nokon av ulike grunnar ikkje kunne møte.

13.4.2 SWOT-Analyse

Som ein del av produktutviklinga har vi laga ei SWOT-analyse som tek føre seg framtidsutsiktene til produktet, sjå vedlegg 8. Ved å sette opp stikkord i forhold til produktutviklinga i eit slikt skjema, får vi kategorisert kva som er mogelegheiter, svakheiter, styrkar og truslar. Kunnskapen om SWOT-analyse er i stor grad basert på Concept rapport nr. 12 frå NTNU [37].

Etter kvart som utviklinga er i gang gjeld det å fokusere på styrkane våre og utnytte mogelegheitene så mykje som råd er. Truslane må vi prøve å unngå så godt som vi klarar og svakheitene våre må vi prøve å snu eller nøytralisere. Fleire av svakheitene lar seg løyse allereie i risikovurderinga.

Under utviklinga av produktet har vi sett dei reelle utsiktene for eit slikt produkt. Ved å fokusere på mogelegheitene til både prosjektgruppa og produktet, har vi hatt eit positivt fokus heile vegen.

Ved å fokusere på styrkane til prosjektgruppa, har vi utvikla eit fungerande produkt utan å møte på problem som vi ikkje har klart å løyse. Noko av grunnen til det er også at vi har prøvd å unngå eller forbetre oss på dei svakheitene vi hadde i utgangspunktet. Ein av truslane vi såg i marknaden var at vi kunne bli sett på som lite seriøse sidan vi var studentar og ikkje ei reell bedrift. Dette har vi klart å motbevise ved å presentere eit produkt som er fungerande og betre en det som var forventa av bransjen.

13.5 Nettside

Som ein del av prosjektet blei det oppretta ei nettside. Ein i gruppa fekk hovudansvaret for å halde den oppdatert med informasjon om framdrifta i prosjektet. Skulen sett i tillegg krav til at prosjektbeskriving, forprosjektrapport, pressemelding, plakat og sluttrapport skal leggst ut på sida.

Utforminga av nettsida er gjort i «WordPress» som er eit publiseringsverktøy med ferdige malar som har fokus på utsjånad, nettstandardar og brukarvennlegheit [38] Ein av dei store fordelane med «WordPress», er at det ikkje krev kunnskap i PHP eller html -programmering.

Frå tidlegare har gruppa avgrensa erfaring med utvikling av nettside og ville derfor prøve å utvikle den lokalt før den blei lagt ut på nettet. For å utvikle nettsida lokalt, må ein ha ein lokal server. Til dette valde vi å bruke XAMPP sidan gruppa hadde kjennskap til den frå tidlegare. XAMPP er ei samling av programvare og verktøy som kan nyttast fritt for å utvikle nettapplikasjonar og liknande [39].

Vi fekk tildelt eit domenenamn til nettsida og ein plass på serveren til HiSF. Når vi fekk dette, eksporterte vi nettsida frå den lokale serveren på ei XML-fil, og den importerte vi til den nettbaserte kontoen.

13.6 Prosjektevaluering

For gruppa sin del har dette vore eit veldig interessant og spennande prosjekt. Vi har drege nytte av eit breitt spekter av kunnskap og eigenskapar. Vi har lært mykje gjennom heile perioden, blant anna har vi oppretta ei nettside der ein kan lese og gjere seg kjent med prosjektet vårt. Vi har sett oss inn i og nytta fleire typar risikoanalyse, for å ha kontroll over prosjektperioden. Det har berre vore mindre justeringar i forhold til framdriftsplanen.

I dette prosjektet har det vore stort fokus på programvareutvikling. Vi har lært ein heil del nytt om programmering, og ikkje minst om korleis vi får dei forskjellige einingane som bankterminal, Java, MySQL og skjermtastatur til å jobbe saman. Heile gruppa har utvikla seg som programmerarar.

Det er utfordrande å jobbe i prosjektgrupper, men ein lærer mykje undervegs. Sjølv om det er forskjell på timebruken i gruppa så har samarbeidet vore godt, og vi er veldig fornøgd med resultatet og synes prosjektet har vore kjekt å jobbe med.

14.0 Figurliste

Figur 1 Utsnitt av klassen bestilling, variablene stemmer overeins med databasen.....	20
Figur 2 Utsnitt av databasen, variablene stemmer overeins med klassen bestilling.....	20
Figur 3 Utsnitt av samansettinga av klassane i CampLett-programmet	22
Figur 4 Viser utsnitt av CSS-fil.....	26
Figur 5 viser CSS-fila styling.css	26
Figur 6 FXML og Java -fil.....	27
Figur 7 Alarm på skjerm	28
Figur 8 Utsnitt av det grafiske rammeverket	29
Figur 9 Utsnitt frå første skjermbile	30
Figur 10 Skjermbile "Boforhold"	31
Figur 11 Oversikt over bubilplassar	32
Figur 12 Oversikt over teltplassar	32
Figur 13 Oversikt over hytter	32
Figur 14 Registrering av namn og val av overnattingsdøgn	33
Figur 15 Bankterminalbilde med progressbar.....	34
Figur 16 Algoritmen til betalingsdelen av CampLett-programmet	37
Figur 17 Arbeidsbenken i Scene Builder.....	42
Figur 18 Baksida av ICT-220 terminalen med alle inngangar.	50
Figur 19 USB B (t.v) til USB A (t.h).	50
Figur 20 Hyperterminal i Windows	51
Figur 21 Utsnitt av kundeundersøkinga frå Google Docs.....	54
Figur 22 Utsnitt frå Excel dokumentet, med tilbakemeldingar frå kundeundersøking	55
Figur 23 Organisasjonskart.....	60
Figur 24 Gantt-skjema fra forprosjekt.....	63
Figur 25 Gantt-skjema etter prosjektet er ferdig	64
Tabell 1 Utsnitt av tabell for campingplassdata.....	39
Tabell 2 Utsnitt av tabell for kundedata	39
Tabell 3 Oversikt over rekneskap og budsjett.....	61
Tabell 4 Utdrag frå risikoanalyse.....	66

15.0 Kjelder

- [1] T. Budd, «Understanding Object-oriented programming with Java,» i *Understanding Object-oriented programming with Java*, Harlow, Addison-Wesley , 2002, pp. 3-10.
- [2] Java,, «Java,» Oracle, [Internett]. Available: <https://www.java.com/en/about/>. [Funnen 01 april 2015].
- [3] Oracle, «docs.oracle.com,» [Internett]. Available: <http://docs.oracle.com/javase/8/docs/>. [Funnen 1 Mai 2015].
- [4] Oracle, «Oracle,» Oracle, 2013. [Internett]. Available: <http://docs.oracle.com/javase/8/javafx/get-started-tutorial/jfx-overview.htm#JFXST784>. [Funnen 1 april 2015].
- [5] MySQL, «MySQL,» [Internett]. Available: <https://www.mysql.com/products/community/>. [Funnen 7 mai 2015].
- [6] Nets , «nets,» Nets, 31 mars 2015. [Internett]. Available: <http://www.nets.eu/no-nb/om-nets/hvem-vi-er/Pages/WhoWeAre.aspx>. [Funnen 31 mars 2015].
- [7] Universitetet i Oslo,, «UiO,» 1 januar 2000. [Internett]. Available: <http://folk.uio.no/hpv/kj207/komp/node4.html>. [Funnen 10 april 2015].
- [8] Aplex Technology, Aplex Technology, [Internett]. Available: <http://www.aplextec.com/en/productdetail-63.html>. [Funnen 25 januar 2015].
- [9] lovdata, [Internett]. Available: <https://lovdata.no/dokument/SF/forskrift/2013-06-21-732>. [Funnen 3 mai 2015].
- [10] DIFI, «Direktoratet for forvaltning og IKT,» [Internett]. Available: <http://uu.difi.no/veiledning/automater/krav-til-automater>. [Funnen 9 mai 2015].
- [11] Microlog, [Internett]. Available: http://www.microlog.no/emv-eurocard_mastercard_visa.htm#31. [Funnen 1 mai 2015].
- [12] Vegdirektoratet, «Universell utforming av veger og gater,» vegdirektoratet, 2011, pp. 123-126.
- [13] DIFI,, «Direktoratet for forvaltning og IKT,» [Internett]. Available: <http://uu.difi.no/veiledning/automat/uu-skolen/betjening>. [Funnen 2 mai 2015].
- [14] Norges blindforbund, [Internett]. Available: <https://www.blindforbundet.no/internett/universell-utforming/automater>. [Funnen 9 mai 2015].
- [15] oracle,, «Oracle,» [Internett]. Available: <https://docs.oracle.com/javase/8/docs/api/java/util/Collection.html>. [Funnen 15 mars 2015].

- [16] oracle,, «Oracle,» [Internett]. Available:
<https://docs.oracle.com/javase/8/docs/api/java/util/HashMap.html>. [Funnen 1 mai 2015].
- [17] oracle,, «Oracle,» [Internett]. Available:
<https://docs.oracle.com/javase/tutorial/java/landl/createinterface.html>. [Funnen 2 mai 2015].
- [18] Oracle, «Oracle,» [Internett]. Available:
<http://docs.oracle.com/javase/8/docs/api/java/lang/Runtime.html>. [Funnen 6 mai 2015].
- [19] github, «Github,» [Internett]. Available: <https://github.com/twall/jna/blob/master/README.md>.
[Funnen 3 mai 2015].
- [20] K. Gregory, «Using Visual C++,» i *Using Visual C++*, Indianapolis, Que, 1998, p. 639.
- [21] Oracle, [Internett]. Available: <https://docs.oracle.com/javafx/2/api/javafx/scene/doc-files/cssref.html#introscenegraph>. [Funnen 12 mai 2015].
- [22] oracle, «docs.oracle,» [Internett]. Available:
<http://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Alert.AlertType.html>. [Funnen 12 mai 2015].
- [23] A. Caicedo, «Oracle,» [Internett]. Available:
https://blogs.oracle.com/acaicedo/entry/managing_multiple_screens_in_javafx1. [Funnen 3 mai 2015].
- [24] oracle,, «Oracle,» [Internett]. Available:
<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/layout/StackPane.html>. [Funnen 7 mai 2015].
- [25] Oracle, «docs.oracle,» [Internett]. Available: <http://docs.oracle.com/javase/8/scene-builder-2/get-started-tutorial/overview.htm#JSBGS164>. [Funnen 12 mai 2015].
- [26] stackoverflow,, [Internett]. Available: <http://stackoverflow.com/>. [Funnen 13 mai 2015].
- [27] stackoverflow, «webområde for openkjelde,» [Internett]. Available:
<http://stackoverflow.com/questions/10101742/difference-between-wm-close-and-sc-close>.
[Funnen 7 mai 2015].
- [28] Dennis Babkin, [Internett]. Available: <http://dennisbabkin.com/winid/>. [Funnen 7 mai 2015].
- [29] Microsoft, «Microsoft,» Microsoft Windows, [Internett]. Available:
<https://msdn.microsoft.com/en-us/library/windows/desktop/ms646360%28v=vs.85%29.aspx>.
[Funnen 7 mai 2015].
- [30] datase import scriptor, [Internett]. Available: <http://databaseimportscriptor.com/>. [Funnen 7 mai 2015].
- [31] Nets, «Betalingsterminal,» [Internett]. Available:
<http://www.betalingsterminal.no/no/Betalingsterminaler-forside/Kommunikasjon/USB-driver-til-iCT220-og-iPP350/>. [Funnen 7 mai 2015].

- [32] Netbeans , [Internett]. Available: <https://netbeans.org/kb/docs/javame/java-card.html>. [Funnen 9 mai 2015].
- [33] how stuff works , [Internett]. Available: <http://computer.howstuffworks.com/question332.htm>. [Funnen 9 mai 2015].
- [34] intuit, «developer.intuit,» [Internett]. Available: https://developer.intuit.com/docs/030_qbms/0070_advanced_topics/pin_debit. [Funnen 7 mai 2015].
- [35] Camping key Europe, «Camping Key Europe,» [Internett]. Available: <http://www.campingkey.com/en/support/>. [Funnen 9 mai 2015].
- [36] J. F. E. N. I. o. S. F. Roy Arve Hetle, «Sjølvtstyrte Camping,» Høgskulen i Sogn og Fjordane, 2014.
- [37] O. T. J. T. M. V. H. I. J. Kjell Austeng, «Usikkerhetsanalyse og metoder,» Concept-programme,tInstitutt for bygg, anlegg og transport,NTNU, Trondheim , 2005.
- [38] Wordpress,, [Internett]. Available: https://codex.wordpress.org/WordPress_Features. [Funnen 7 mai 2015].
- [39] Apache,, [Internett]. Available: <https://www.apachefriends.org/index.html>. [Funnen 7 mai 2015].

16.0 Vedlegg

1. Brukermanual
2. Microlog
3. Timeliste
4. E-post, leverandører
5. NDA (Nets)
6. Møtereferat
7. Risikovurdering
8. SWOT
9. Blokkdiagram
10. Flytskjema
11. Terminal utviding (iUN)
12. ICT 220
13. Kjeldekodar for CampLett-programmet

Vedlegg 1

BRUKSANVISNING

Camp **Lett**

Prototype av CampLett V1.7

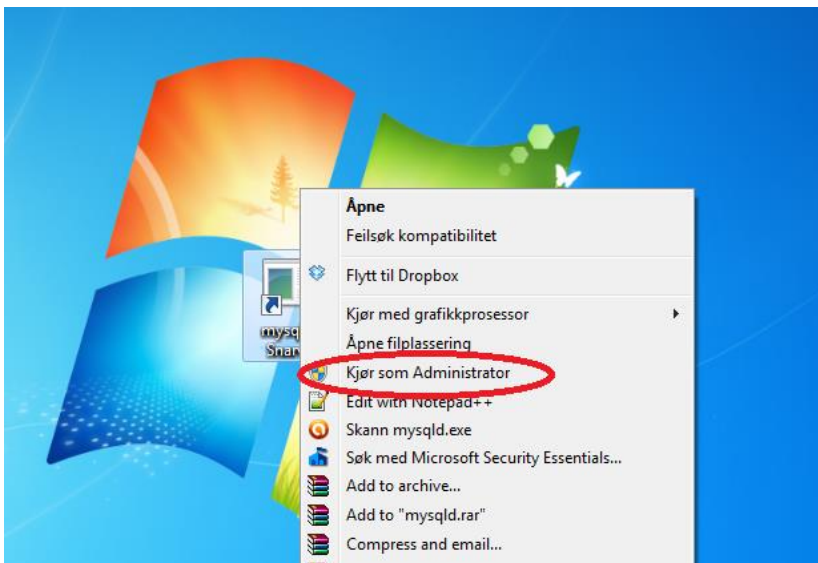
Innhald

Starte database	3
Starte program	4
Velg språk	5
Boforhold.....	5
Velge plass.....	6
Tal dagar	7
Betaling.....	8
Kvittering	8
Alarmer.....	9
Slå av program.....	10
Resett database.....	11

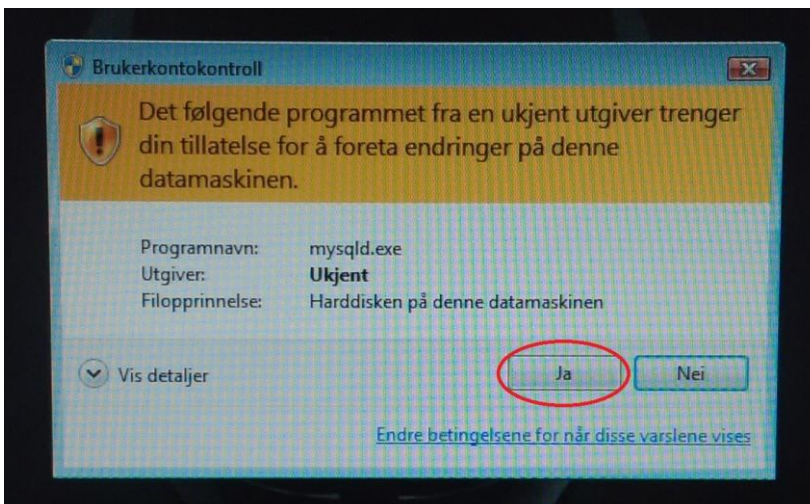
Starte database



Høgreklikk på snarveg for databasar



Trykk på "Kjør som Administrator"



Trykk "Ja"

Starte program



Dobbelklikk på CampLett-ikon

Velg språk



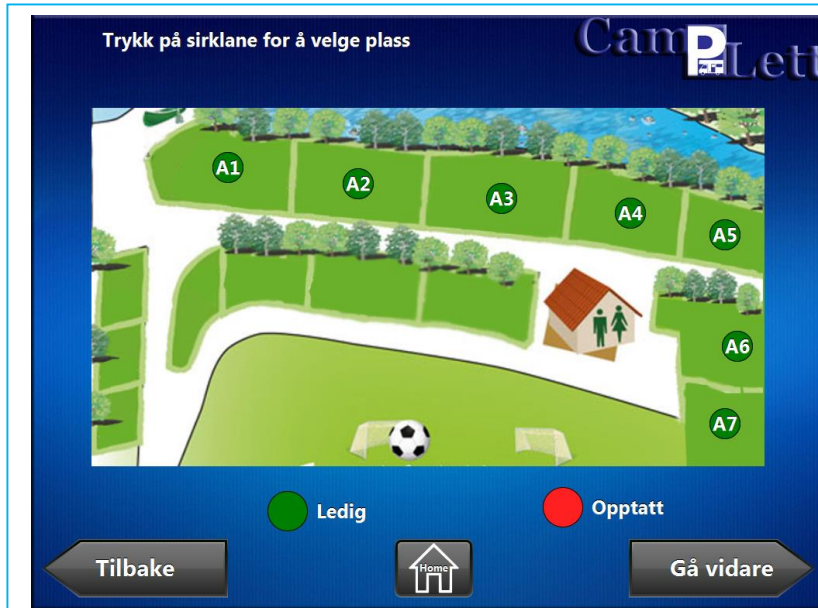
- Trykk på ønska språk, enten ved knapp eller flagg

Boforhold



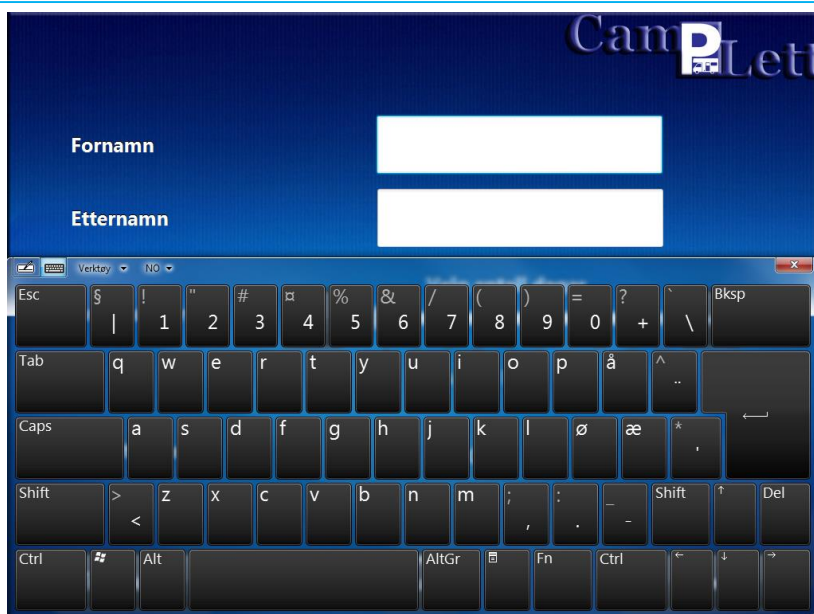
- Trykk på ønska boforhold
- Til høyre er det markert om det er ledige plassar og pris per døgn
- Trykk tilbake for å komme til hovudmeny
- Trykk home for å komme til hovedmeny

Velge plass



- Trykk på sirkel ved ønska plass. (Likt for alle boforhold) Ønska plass blir raud ved valg.
- Raud sirkel er opptatt
- Grøn sirkel er ledig
- Trykk "Gå videre" for å fortsette bestilling.
- Trykk "Tilbake" for å gå tilbake til boforhold
- Trykk Home for å gå til startskjerm

tal dagar

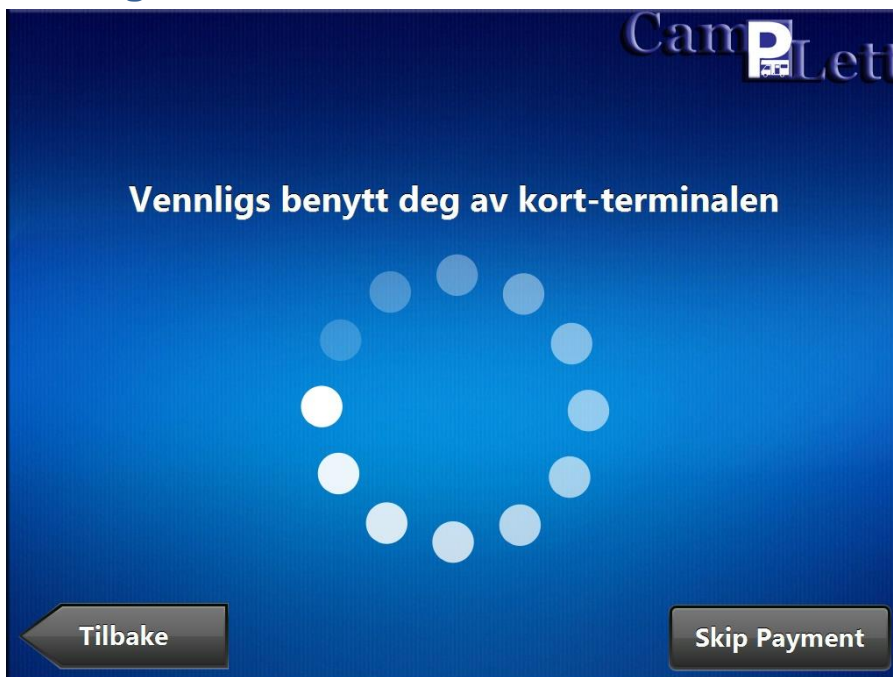


- Tast inn navn i rutene ved hjelp av tastatur.
- Fjern tastatur ved å trykke ENTER i siste rute eller trykk utanfor tastatur.



- Velg antall dager ved slider eller pluss og minus tast.
- Trykk "Godta" for å gå til betaling
- Trykk "Tilbake" for å gå tilbake til plassvalg
- Trykk "Home" for å gå til startskjerm

Betaling



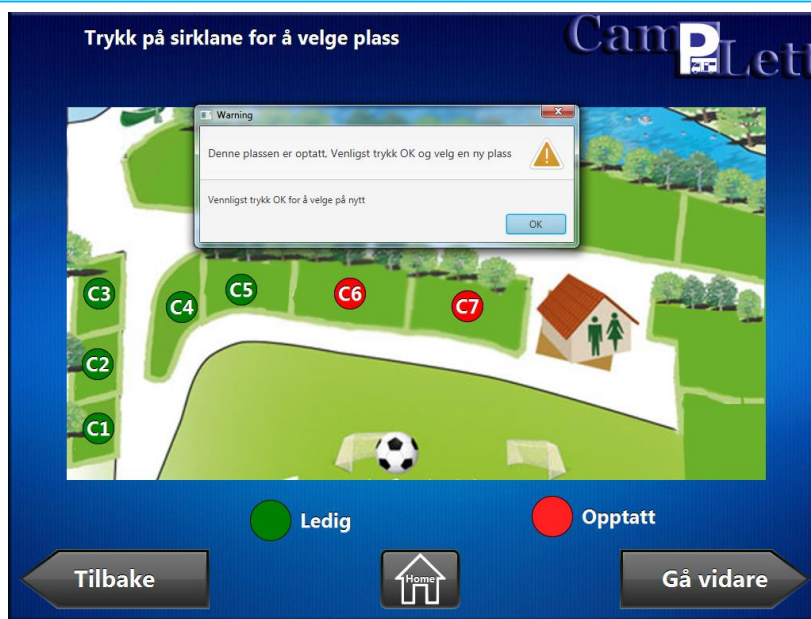
- Sett kort i terminal
- Tast kode+ok
- Betaling utført og program går vidare til kvitteringsskjerm
- Tast "Skip payment" for å hoppe over betaling
- Tast tilbake for å gå til "Antall dager"

Kvittering



- Viser all info som er registrert under bestilling
- Etter 30 sekunder går skjerm automatisk til første skjermbilde
- Tast "Slett kjøp" for å slette transaksjon fra minibank, om det blei utført
- Trykk "Home" for å gå til første skjermbilde med eingang

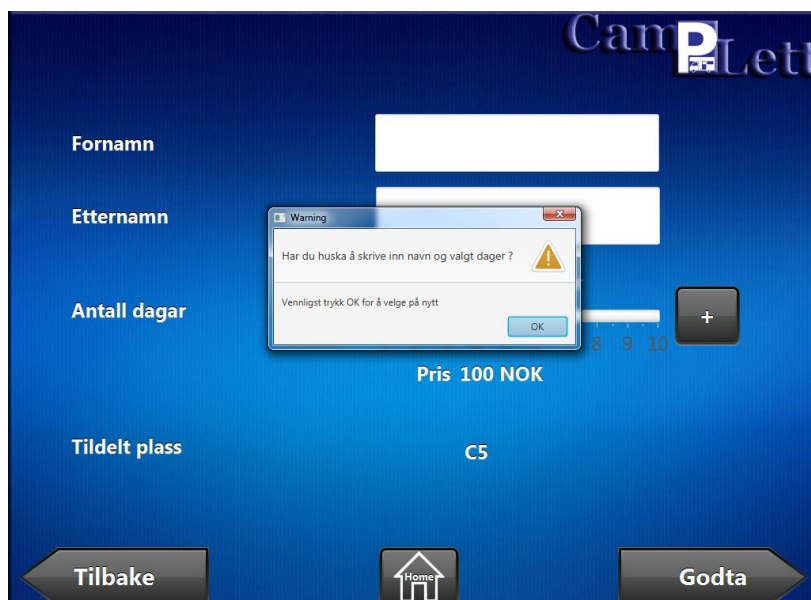
Alarmer



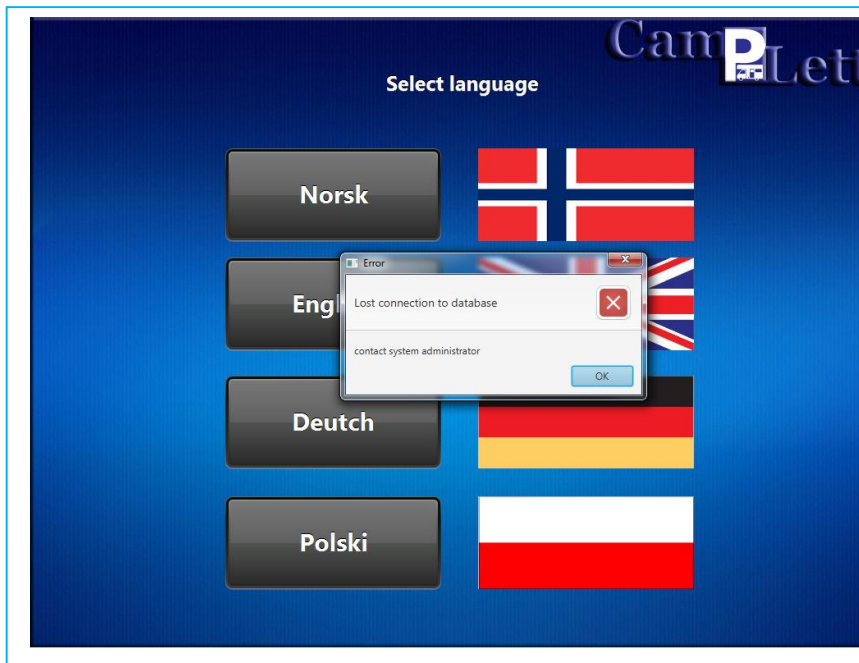
Alarm ved trykk på raud(opptatt) plass. Trykk ok for å prøve på nytt



Alarm ved trykk på gå videre uten ha valt plass. trykk ok for å prøve på nytt



Alarm ved trykk på godta uten å taste inn navn. Trykk ok for å prøve på nytt



Alarm ved oppstart av program,
database manglar.

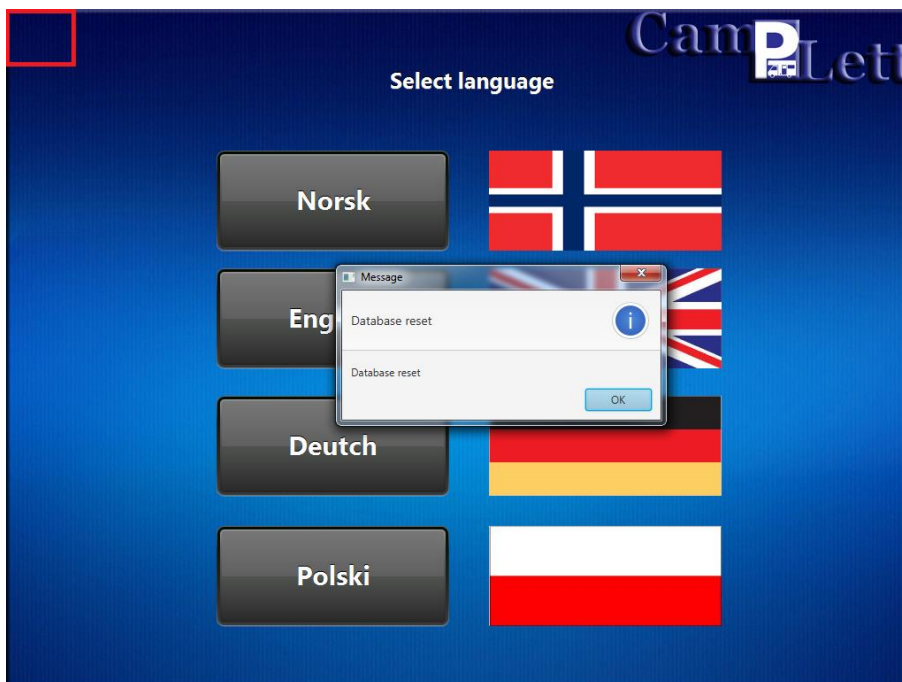
Forsøk å start MySQL eller kontakt
system-administrator.

Slå av program



Trykk på høgre, øvre hjørne i
CampLett-logo for å slå av
programmet(usynlig knapp)

Resett database



Trykk på venstre, øvre hjørnet for å
resette database(usvnlig knapp)

Vedlegg 2 Microlog

ti 11.11.2014

Til: post@microlog.no;

Hei

Eg tar kontakt med dykk i samband med ei prosjektoppgåve på Høgskulen i Førde. Vi held på å lage til eit automatisk innsjekk system og lurte på reglane med EMV-sertifisering. Vi skal lage eigen software som vil operere med billettsystem og betalingssystem.

Så lenge betalingsterminalen er sertifisert, så kan den brukast ilag med anna software? (kassasystem, billettsystem osv)

Mvh

Eivind N. Inderøy

E-post: eivindi@stud.hisf.no

TLF: 94857308

ti 11.11.2014

Til:

Eivind Nordal Inderøy;

Hei,

Det er korrekt. Det som er viktig å ta hensyn til er å få godkjent innsynsbeskyttelse for betalingsterminalen samt kravene i Universell utforming.

Lykke til!

Med vennlig hilsen

Aslak Lie

Daglig leder

Vedlegg 3 Timeliste

4	21.jan	4,50		1,50	0,50	4,00	10,50
4	22.jan	1,50	2,00	3,50	3,50	4,00	14,50
4	23.jan	3,00					3,00
4	24.jan						0,00
5	25.jan						0,00
5	26.jan		3,00				3,00
5	27.jan	6,00	4,00	4,00	3,00		17,00
5	28.jan	1,50	5,00	4,00	4,00	6,00	20,50
5	29.jan						0,00
5	30.jan	5,00	5,00	5,00	5,00		20,00
5	31.jan						0,00
6	01.feb						0,00
6	02.feb	2,00	4,00	2,00	3,50		11,50
6	03.feb	2,00	3,00	2,50		2,00	9,50
6	04.feb	3,50		2,00	0,50	6,00	12,00
6	05.feb		1,00	1,50			2,50
6	06.feb				0,50		0,50
6	07.feb						0,00
7	08.feb		4,00	2,00			6,00
7	09.feb	2,00	4,00	3,00			9,00
7	10.feb	3,00		2,00	1,50		6,50
7	11.feb	1,00	3,00		3,00	3,00	10,00
7	12.feb		2,00	3,00			5,00
7	13.feb						0,00
7	14.feb						0,00
8	15.feb						0,00
8	16.feb	4,00	2,50	4,00			10,50
8	17.feb	2,00	2,00	4,00	4,00		12,00
8	18.feb						0,00
8	19.feb	4,00	3,00	3,00	3,50		13,50
8	20.feb	5,00	5,00	5,00	5,00		20,00
8	21.feb						0,00
9	22.feb	0,50					0,50
9	23.feb	1,50			2,50	8,00	12,00
9	24.feb			3,50	3,00		6,50
9	25.feb						0,00
9	26.feb	3,00					3,00
9	27.feb						0,00
9	28.feb						0,00
10	01.mar	5,00	8,00	8,00	5,00		26,00
10	02.mar	1,50	4,00	2,00	5,00		12,50

Eivind har bursdag

Open dag

Innlevering av forprosjektbeskriv

NY Innlevering av forprosjektbeskriving

Jens, Sander og Eivind er i stryn
Jens, Sander og Eivind er i stryn

10	03.mar	4,50		4,00			8,50
10	04.mar	5,50	3,00	8,00	8,00	3,00	27,50
10	05.mar	2,00	6,00	4,00	1,00		13,00
10	06.mar	6,00	3,00	5,00	4,00		18,00
10	07.mar						0,00
11	08.mar						0,00
11	09.mar	6,00		7,00	5,00		18,00
11	10.mar						0,00
11	11.mar	7,00		7,00	7,50		21,50
11	12.mar	7,00	4,00	5,00	6,00		22,00
11	13.mar	6,00	6,00	4,00			16,00
11	14.mar	9,00	8,00	8,00	8,00		33,00
12	15.mar						0,00
12	16.mar	7,00	8,00	9,00	8,00		32,00
12	17.mar	4,00	1,00	2,00	4,50		11,50
12	18.mar	5,00	4,00	4,00	6,00	4,00	23,00
12	19.mar	6,50	8,00	8,00	7,00		29,50
12	20.mar	6,50	9,00	8,00	10,50		34,00
12	21.mar						0,00
13	22.mar						0,00
13	23.mar	6,00		3,00	6,50		15,50
13	24.mar	4,00	3,00	4,00	3,50		14,50
13	25.mar	7,00	7,00	7,00	7,00	4,00	32,00
13	26.mar	7,00	7,00	6,00	3,00		23,00
13	27.mar	6,00	6,00	6,00	7,50		25,50
13	28.mar						0,00
14	29.mar						0,00
14	30.mar	6,50	7,00	6,00	6,50		26,00
14	31.mar	7,00	7,50	6,50	4,50		25,50
14	01.apr	4,50	5,00	4,50	4,50	6,00	24,50
14	02.apr	4,00	3,50	4,00	4,50		16,00
14	03.apr						0,00
14	04.apr						0,00
15	05.apr						0,00
15	06.apr	5,00		5,00	5,00		15,00
15	07.apr	6,50	4,00	8,00	8,00		26,50
15	08.apr	9,50		9,00	9,50		28,00
15	09.apr	8,50	4,00	8,00	5,50		26,00
15	10.apr	8,50	7,00	7,00	8,00		30,50
15	11.apr			5,00			5,00
16	12.apr			2,00			2,00
16	13.apr		7,00	7,00	5,00		19,00

Roy har bursdag

Innlevering1 matte3 - ny dato

Påskeferie

Påskeferie

Påskeferie

Påskeferie

Påskeferie

Påskeferie

Påskeferie og Jens har bursdag

Påskeferie

Midtvegspresentasjon

16	14.apr	5,00	7,00	7,00	7,00		26,00
16	15.apr	5,00	4,00	7,00	6,00	4,00	26,00
16	16.apr	7,00	2,00	6,00	7,50		22,50
16	17.apr	5,00		5,00	4,50		14,50
16	18.apr						0,00
17	19.apr						0,00
17	20.apr	7,00		5,00	5,00		17,00
17	21.apr	5,50	3,00		5,50		14,00
17	22.apr		4,00	1,50			5,50
17	23.apr			2,00			2,00
17	24.apr			5,00			5,00
17	25.apr	7,00	5,00	6,00	6,00		24,00
18	26.apr	6,00	7,00	7,50	7,00		27,50
18	27.apr	9,00		8,50			17,50
18	28.apr	6,50	3,00	5,00	4,00		18,50
18	29.apr	8,00		6,00	5,50		19,50
18	30.apr	7,00	7,50	7,50	7,50		29,50
18	01.mai	3,50	4,00	3,00	4,00		14,50
18	02.mai	3,00		3,00	3,00		9,00
19	03.mai						0,00
19	04.mai	5,00	5,00	9,00	6,00		25,00
19	05.mai	8,50	5,00	8,00	8,00		29,50
19	06.mai	8,00	6,50	7,00	7,00	4,00	32,50
19	07.mai	12,00	7,50	10,00	12,00		41,50
19	08.mai	9,50	4,00	9,50	9,50		32,50
19	09.mai	6,00	8,00	2,00	2,00		18,00
20	10.mai	4,50	1,00	2,00			7,50
20	11.mai	8,50	7,00	8,00	8,50		32,00
20	12.mai	10,50	5,00	8,00	7,50		31,00
20	13.mai	4,00	7,00	7,00	7,00		25,00
20	14.mai						0,00
20	15.mai	4,00		3,50			7,50
20	16.mai						0,00
21	17.mai						0,00
21	18.mai						0,00
21	19.mai	7,00	8,00	7,50	7,50		30,00
21	20.mai	9,50	5,00	8,00	8,00		30,50
21	21.mai						0,00
21	22.mai						0,00
21	23.mai						0,00
22	24.mai						0,00
22	25.mai						0,00
22	26.mai						0,00
22	27.mai						0,00

Innlevering2 matte3

Hugs å lage til pressemelding og plakat

Innlevering av sluttrapport

Presentasjon m/plakat

22	28.mai						0,00
22	29.mai						0,00
22	30.mai						0,00
23	31.mai						0,00
23	01.jun						0,00
23	02.jun						0,00
23	03.jun						0,00
23	04.jun						0,00
23	05.jun						0,00
23	06.jun						0,00
24	07.jun						0,00

Nettsida ferdigstillt. Rydde på lin

	Roy	Sander	Jens	Eivind	Total fellestid	Sum Alle
	415,00	308,00	421,00	372,50	58,00	1574,50

Vedlegg 4 e-post leverandørar

Mal

Hei

Vi er fire ingeniørstudenter frå Høgskulen i Sogn og Fjordane avd. Førde. Vi skal i vår utarbeide ei bacheloroppgåve der vi lagar software ved hjelp av programmeringspråk i Java og Bibliotek i Java-Netbeans. I vår software skal vi ha kommunikasjon med ein bankterminal av typen Ingenico ICT220 for å kunne betale for tjenesten vi presenterar i vårt program.

Sjølve hovudprogrammet er vi i godt gang med og har laga funksjonar med database til MySQL og den visuelle delen har vi laga i JavaFX med fxml-filer osv.

Det som vi lurar på med denne mailen er sjølve kommunikasjonsdelen mellom Pc(rs-232com) og terminalen. Vi har litt erfaring med RS-232 kommunikasjon og det å sende seriell bits, men vi har ikkje tilstrekkelig informasjon til å kunne kommunisere med terminalen.

Vi har ikkje tilstrekkelig med tid til å sette oss inn i hardkodinga for terminalen sidan prosjektet skal vere avslutta i slutten av april/ begynnelsen av mai.

Problemstilling:

- 1) Finst det opne kildekodar som vi kan bruke.
- 2) Korleis kan vi opprette kommunikasjon mellom Java-netbeans og terminalen.
- 3) Korleis kan vi sende den informasjonen vi vil til terminalen.

Svar frå leverandørar etter utsendt «mal»

13. mars 2015

Fra: Fredrik Bjørn

Til: 'einvindi@stud.hisf.no

Hei Eivind.

Sitter dessverre ikke på slik informasjon.

Ta heller kontakt med NETS kundeservice, eller muligens NETS forhandlersupport:

forhandlersupport@nets.eu

Da de sitter på API etc. for integrasjon mot NETS bankterminaler.

Med vennlig hilsen

Fredrik Bjørn

fr 13.03.2015

Til: Eivind Nordal Inderøy;

Eneste tipset jeg kan komme med her er å sjekke om det er POINT eller NETS (evt andre jeg ikke har erfaring med) som leverer denne terminalen.

De har rammeverk som man bør bruke for å kommunisere med terminal. Kanskje ikke de har rammeverk i java, da må dere evt lage et program som fungerer som mellomledd mellom terminal og java-programmet.

POINT har javastøtte, NETS har ikke hatt det før hvertfall.

Utover dette tar jeg 600 kr timen:)

Mvh

Stian

Svar frå Nets kan ikkje offentliggjeras pga. vedlegg 5.

Vedlegg 5NDA (Nets)

Nets Norway AS, org nr 990 224 978 (**Nets**) and Hisf, org nr 974246724 address Svanehaugvegen 1, 6812 Førde (**Hisf**), have entered into this non-disclosure agreement (**Agreement**) as of the date last written below.

Nets and the Company are hereinafter referred separately as a **Party** and jointly as the **Parties**.

1. For the purpose of evaluating the feasibility of possible commercial relationship regarding integration of an ICT220-terminal into a javaprogram in a bachelor assignment(**the Purpose**), it will be necessary for each Party to give and receive Confidential Information (as defined in section 2) to and from each other. As the Parties wish to maintain the confidentiality and integrity of all Confidential Information, the Parties have entered into this Agreement.
2. In this Agreement confidential information means any information and data furnished by the Disclosing Party to the Receiving Party directly or indirectly hereunder in tangible or intangible form, including but not limited to trade secrets, know how, ideas, copy righted material, software code and software specifications, business methods, business activities, inventions and concepts, technical, marketing, operating, performance, cost and process information as well as customer and supplier information whether it is marked confidential or otherwise should be understood to be confidential (**Confidential Information**). The Confidential Information shall also include the fact that the Parties are negotiating on mutual cooperation and the existence of this Agreement.
3. The Receiving Party shall copy the Confidential Information only to the extent necessary for the Purpose and shall not use the Confidential Information for any other purpose other than the Purpose without the prior written permission of the Disclosing Party.
4. The Receiving Party shall keep all Confidential Information strictly confidential and shall not disclose it to third parties without the prior written consent of the Disclosing Party. Notwithstanding the aforesaid, the Receiving Party may disclose Confidential Information to its employees, consultants, legal advisors and other such representatives who need to know to fulfil the Purpose and provided that the Receiving Party ensures that they comply with similar confidentiality requirements as laid out in this Agreement
5. Nets may disclose the Confidential Information to its Affiliates and subcontractors for use only in connection with the Purpose. Nets shall be responsible for observance of the provisions of this Agreement by its Affiliates and that its subcontractor accepts confidentiality obligations similar to those contained in this Agreement.

For the purpose of this Agreement "**Affiliate**" shall mean any legal entity that is: (a) directly or indirectly controlling Nets, or (b) under the same direct or indirect control as Nets, or (c) directly or indirectly controlled by Nets for so long as such control lasts. "**Control**" means control as existing, now or hereafter, through direct or indirect ownership of fifty per cent (50 %) or more of the nominal value of the issued equity share capital or of fifty per cent (50 %) or more of the

shares entitling the holders to vote for the election of the members of the board of directors or persons performing similar functions.

6. The Receiving Party shall take all reasonable precautions to safeguard any Confidential Information by handling such Confidential Information with at least the same degree of care as the Receiving Party normally employs to safeguard its own information of equivalent importance from unauthorized disclosure, publication, dissemination, or use, never, however, less than with reasonable degree of care.
7. The Receiving Party shall not be restricted from disclosing Confidential Information pursuant to a judicial or governmental order, but any such disclosure shall be made only to the extent so ordered and provided that the Receiving Party:
 - (i) shall as soon as practicable notify the Disclosing Party so that the Disclosing Party may intervene in response to such order; or
 - (ii) if timely notice cannot be given, shall seek to obtain a protective order from the court or government for such information.
8. The obligations and limitations set forth herein regarding Confidential Information shall not apply to information or material which is:
 - (i) at any time in the public domain other than by a breach of this Agreement on the part of the Receiving Party;
 - (ii) at any time rightfully received from a third party which has the right and transmits it to the Receiving Party without any obligation of confidentiality;
 - (iii) rightfully known to the Receiving Party without any limitation on use or disclosure prior to receipt of the same from the Disclosing Party;
 - (iv) independently developed by the Receiving Party without access to Confidential Information received from the Disclosing Party; or
 - (v) generally made available to third parties by the Disclosing Party without any restriction concerning its use or disclosure.
9. This Agreement shall in no way restrict the Parties' customary research and development activities.
10. Title, the right to possess and all other rights in or to Confidential Information of the Disclosing Party as between the Parties shall, except otherwise provided herein, remain with the Disclosing Party.
11. The Receiving Party shall promptly upon receipt of a written demand from the Disclosing Party cease using Confidential Information and:
 - (i) return or destroy all material incorporating any Confidential Information; and
 - (ii) to the extent technically practical expunge or destroy any Confidential Information from any computer, word processor or other device whatsoever into which it was copied, read or programmed by the

Receiving Party or on its behalf other than copies made solely for back-up or archive purposes.

Each Party shall, however, be entitled to retain copies required by law or regulations.

12. This Agreement is not intended to and shall not be construed as creating a joint venture, partnership or other form of business association between the Parties or as establishing a license grant of any kind.
13. Neither this Agreement nor the disclosure of Confidential Information or any other action constitutes any representation, warranty or guarantee, including but not limited to the accuracy, completeness or the merchantability or fitness for a particular purpose of Confidential Information.
14. Neither this Agreement nor disclosure or receipt of Confidential Information shall constitute or imply any promise to make any purchase of products or services by either Party or any commitment by either Party with respect to the present or future marketing of any product or service or any promise or intention to enter into any other business arrangement.
15. Neither Party is entitled to assign or otherwise transfer this Agreement, in whole or in part, without the prior written consent of the other Party. Notwithstanding the aforesaid, Nets is entitled to assign or transfer this Agreement to its Affiliate or in connection with Nets' transfer of relevant business activities to a third party through informing the other Party of the assignment or transfer in writing.
16. The failure or delay by either Party in exercising any right or remedy will not constitute a waiver. No waiver of any term or condition of this Agreement or of any right or remedy arising in connection herewith shall constitute a continuing waiver or a waiver of any right or remedy relating to a subsequent breach of such provision or of any other right or remedy hereunder.
17. This Agreement shall become effective as of the date when signed by duly authorized representatives of both Parties and shall thereafter remain valid and in force for a period of five (5) years. Notwithstanding the preceding sentence, all Confidential Information received during the term of this Agreement shall be subject to the terms of this Agreement for a period of at least three (3) years as from the date of receipt by the Receiving Party of the relevant Confidential Information or longer period if required by mandatory law.
18. This Agreement shall be governed by the laws of Norway, excluding its conflict of law provisions.
19. Any dispute arising out of this contract, or other contracts resulting from it, shall be finally settled by Arbitration pursuant to the Rules of the Arbitration and Dispute Resolution Institute of the Oslo Chamber of Commerce in force at any time.

The number of arbitrators shall be one (1), the language of the arbitration shall be English and seat of the arbitration shall be in Oslo, Norway.

Notwithstanding the foregoing, nothing in this Agreement shall be deemed to limit the Parties' rights to seek interim injunctive relief or to enforce an arbitration award in any court of law.

20. The Parties have negotiated this Agreement and each Party's legal counsel has had the opportunity to review this Agreement. Nets and the Company agree that any rule of construction or interpretation requiring resolution of any ambiguities in this Agreement against the drafting Party will not apply in the construction or interpretation of this Agreement.
21. This Agreement states the entire agreement and supersedes all prior agreements, discussions, representations or other understandings, written or verbal, between the Parties with respect to the subject matter hereof and may not be amended except in writing signed by a duly authorized representative of the respective Parties.

In witness whereof, this Agreement has been signed by the duly authorized representatives of both Parties.

Oslo

Førde 17.03.2015

Nets Norway AS

HiSF

Rune Marcussen
Senior Partner Manager Northern Europe
Sales

Eivind N. Inderøy
Engineering at HiSF

Prosjektmøte

Dato	21.01.2015	Kl. Start	11.40	Kl. Slutt	12.45
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Sakliste

- Forprosjekt -> ferdig 13.02
- Møte med Joar og Preben

Aktuelt

1. Forprosjekt -> ferdig 13.02
 - Nettside
 - Presise mål (Hovud /del)
 - Finne ulike Tekniske løysingar og vurdere dei mot kvarandre ut frå kompleksitet, økonomi og tid.
 - Sette opp tids/resurs –plan
 - Budsjett for hovudprosjekt
 - Konklusjon skal innehalde
 - Presise hovud og del –mål
 - Tittel på prosjekt
2. Etter forprosjekt
 - Møte i styringsgruppa
 - Styringsgruppa tar slutning om godkjenning
 - Viss godkjent forprosjekt starter hovudprosjekt
3. Møte med Joar og Preben, eigne referat av Jens og Roy

Ymse

- Roy Arve er valt som Prosjektleder
- Jens er valt som fast referent av Møter
- Eivind lagar til feles prosjektdagbok for alle på gruppa

Til neste møte

Møtereferat er skriv av **Jens Førstund**.

Referatet vert gjort tilgjengeleg for alle gruppemedlem omgåande og kvar einskild er sjølv ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	22.01.2015	Kl. Start	13.00	Kl. Slutt	14.45
-------------	------------	------------------	-------	------------------	-------

45 min pause

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Sakliste

1. Oppstart forprosjekt
 - a. Sjå på forbetring frå føregåande prosjekt
 - b. Kva kan vi ta med vidare i neste prosjekt
 - c. Fordele arbeidsoppgåver forprosjekt
2. Lage ein arbeidsplan for forprosjekt

Aktuelt

1. Diskutert rundt rammene på prosjektet, konkludert med at vi lagar og prosjekterer kun SW, ikkje HW.
 2. Forslag: Forbetre økonomi analyse, innsnevring av hovudprosjekt med tanke på tid og kostnad
- Finne ut kva vi skal utvikle, med god dokumentasjon
 - Sjå gjennom tilbakemelding sist prosjekt
 - Sette opp forbetningspunkt
 - o Kva kan vi bruke frå sist prosjekt
 - Lage oversikt over tidsbruk for hovudprosjekt

- Setje opp Mål:
 - Hovudmål
 - Delmål

- Sette opp rollene i gruppa
 - Kontakt person Næringsliv
 - Programmering HMI
 - APP/ server
 - Design
- Gjennomføring av analyser risiko og SWOT?

- Budsjett/ økonomi

- Nettside (til informasjon om prosjekt) ein ansvarlig

Ansvarsområde

Eivind	-	Teknisk løysing
Jens	-	Risiko og mål
Sander	-	Nettside
Roy	-	Tidsbruk og økonomi

Ymse

Til neste møte

- Alle skal være godt i gang med sine ansvarsområde
- Være klar til å diskutere ansvarsområde i hovudprosjekt

Møtereferat er skriv av **Jens Førund**.

Referatet vert gjort tilgjengeleg for alle gruppedlem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

[

]

Prosjektmøte

Dato	28.01.2015	Kl. Start	11.10	Kl. Slutt	12.40
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Roy, laga estimert timeoppsett med forskjellige diagram

Jens, endra på mål, laga risikovurdering, vil ha tilbakemelding på det

Eivind, Har laga ei grovskisse ut frå møta

Sander, Har forska en del på HTML-programmering, har funnet eit program Wordpress som fungerer bra

Sakliste

Gjennomgang av kva som er gjort.

Diskutere ansvarsområde i hovudprosjekt

Aktuelt

Gjennomgang av time-estimat, må endre litt på fargane i kakediagram, time-estimat ser bra ut

Mål ser bra ut.

Diskusjon rundt teknisk løysing, kva skal være med og ikkje.

Ansvarsområde:

Prosjektleder Roy Arve Hetle

Risiko Jens

Programmering (design) Jens er hovudansvarlig, Eivind nr 2. alle skal delta

Rapport (økonomi) Roy er hovudansvarlig pga. PL

Nettside Sander er hovudansvarlig

Ymse

Tilbakemelding på risikovurdering

Beskriving av SW til forprosjekt.

Til neste møte

Roy, Sjekk ut google docks, sjå meir på oppbygging av forprosjektrapporten

Sander, fortsett med heimesida

Eivind, fortsetter med teknisk løysing

Jens, fortsette med risikovurdering

Møtoreferat er skriv av **Jens Før Sund**.

Referatet vert gjort tilgjengeleg for alle gruppelemmer omgåande og kvar ein skild er sjølv ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	04.02.2015	Kl. Start	10.10	Kl. Slutt	11.40
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Jens

- Klar med risikovurdering, laga nye element med løysingar osv..
- Skjekk kor mykje vi kan kopiere og lime frå førre rapport, eller om alt må skrivast om
- SWAT, treng meir input
- Tabellen til risikovurdering blir lagt som vedlegg, så setter vi inn eit utsnitt i rapporten

Sander

- Gjort seg godt kjend med wordpress
- Treng anna bilde og namn på sida etc
- Syner fram heimesida på projektor, alle tykkje det ser bra ut

Eivind

- Har fått ned det som har våre diskutert rundt ideen
- Laga opp til fleire teikningar i Paint
- Laga grovskisse på systemet
- Ønskjer tilbakemeldingar på det han har gjort

Roy

- Google docks ser ikkje ut til å være ein god ide, blant anna er det kun retting på bokmål
- Laga organisasjonskart, gant, etc
- Laga eit lite budsjett
- Vil ha tilbakemelding på plan på hovudprosjekt

Sakliste

3. Gjennomgang av kva som skal inn i forprosjektrapport
 - Innleveringsfrist 13.02.2015
 - Kva skal rapport innehalde

- Setje frist for når forprosjektrapport skal vere klar til korrektur

Aktuelt

- Gjennomgang av kva som skal inn i forprosjektrapport
- Innleveringsfrist 13.02.2015
- Kva skal rapport innehalde
 - Nettside **OK må skrivast litt om kva som er valt osv..**
 - Innleiing **begynt**
 - **Samandrag**
 - Plan hovudprosjekt Inkl. tidsplan **ok**
 - Teknisk løysing **på god veg**
 - Estimert timar **OK**
 - Risiko **OK oppsummering mangler**
 - Fordeling arbeidsoppgaver **ok**
 - **Konklusjon**
 - Budsjett økonomi **OK**
 - Mål **ok**
- Setje frist for når forprosjektrapport skal vere klar til korrektur
Mest mulig må være klart til fredag
Alt skal være klart til tysdag utan om konklusjon og samandrag

Ymse

- Alle skriv dagbok og timeliste
- Brainstorming på SWAT til produkt
- Risikoanalysa vart synt fram

Til neste møte

Sander

- Gå på fronter og sjå på dei tidlegare heimesidene for å sjå at vi har med det som er vanlig å ha på heimesida

Roy

- Sander må ha tilgang til fronter

- Gi bekjed til Ole Kristian om at vi må bruek bankterminalen til Akutten

Alle

Alle er forplikta til å settje seg inn i risikodelen

Alle punkt til forprosjekt skal være ferdige

Les gjennom dokument

Møtereferat er skriv av **Jens Før Sund**.

Referatet vert gjort tilgjengeleg for alle gruppedlem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	11.02.2015	Kl. Start	10.15	Kl. Slutt	11.10
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Førstund	<input type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Eivind: Redigert etter korrektur

Sander: Nettside er oppe. Har laga til eit eige avsnitt om den

Roy: Redigert, korrekturlest

Sakliste

1. Status arbeidsoppgåver, frå sist møte
2. Gjennomgang av kva som står att å gjere til forprosjektrapport
 - Innleveringsfrist 13.02.2015.
3. Ymse

Aktuelt

1. Det som står att
 - Samandrag
 - Konklusjon
 - Oppdatere budsjett - blir søkt om støtte frå skulen til panelskjerm
 - Sette saman rapport

Ymse

- Referenten må gje meir tilgang på referatmal i tilfelle han ikkje møter
- Risikoelement nr.34 må få nye tiltak.
- Funnet tiltak for å få ned fråvær. Om nokon er meir enn 20 min. sein til oppsett tid, må han spandere kaffi den dagen. Gruppa har vedtatt det.
- Kva skal ut på heimesida:
 - Alt skal korrekturlesast og godkjennast av gruppa
 - Alle bilder skal undersøkast om dei har opphavsrett og om dei er lov å brukel

Til neste møte

- Rapport skal vere innsendt.
- Oppstart hovudprosjekt.

Møtereferat er skriv av **Eivind Inderøy**.

Referatet vert gjort tilgjengeleg for alle gruppemedlem omgåande og kvar einskild er sjølv ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

Jens: har lese referat og retta opp det som er eg kan gjere 11.02.2015.

Prosjektmøte

Dato	23.02.2015	Kl. Start	15.30	Kl. Slutt	16.00
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Forprosjekt rapport levt og godkjend

Sakliste

1. Oppstart hovedprosjekt:
 - Framdriftsplan
 - oppsummere forprosjekt rapport
 - sette oss mål til neste møte

Aktuelt

1. Oppstart hovedprosjekt:
 - Framdriftsplan
 - 24Feb-23 mars ->programmering
 - 23 mars -> testing
 - 23 april - > feilretting
 - 30 april -> slutttest ->02.mai
 - Rapport -> 6. mai
 - Reservetid -> 18.mai
 - Framføring -> 08 april
 - Hovudframføring -> 27 mai
 - Oppsummering av forprosjekt rapport
 - Møte med styringsgruppa, Joar onsdag 25. etter kl 12.

Ymse

Måte med Joar, kl. 12.30, viss mulig.

Støtte til innkjøp

Tilbakemelding på rapport

Til neste møte

Neste møte 04.03.2015

Grafisk/skjematisk oppstilling av noko lunde kva vi treng å ha med av delar, klassar ol. (ansvar Jens)

- ✓ Fordele kva kvar enkelt skal jobbe med
- ✓ Begynne på algoritme
- ✓ Undersøke tilkopling av bankterminal

Få tilbakemelding på Forprosjektrapporten

Bestille panel-PC, høre med joar korleis fakturering fungerer (ansvar Eivind)

Lage ei form for timeplan, for å holde timebudsjettet (ansvar Roy)

Oppdatere webside (ansvar Sander)

Møtereferat er skriv av **Jens Før Sund**.

Referatet vert gjort tilgjengeleg for alle gruppelemmer omgåande og kvar ein skild er sjølv ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	04.03.2015	Kl. Start	10.00	Kl. Slutt	10.40
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Førstund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Eivind:

Har bestilt panel-PC, Eivind har kontrollen utan HDD og operativsystem.

Bankterminal: Sett litt på Java-card, slått det frå seg, skal sjå meir på tilkopling tcpip et.

Sander:

Gjort lite

Jens:

Har komme i gang med grovskissene av programmet, lage ein del klassar etc.

Roy:

Har laga tabell i excel, prøver å få det inn i ein database på mysql

Har laga plan på timefordeling resten av prosjektet.

Sakliste

1. Status arbeidsoppgåver, frå sist møte.
2. Framdrift i forhold til plan, er 11 veker igjen. Framlegg frå PL om kva gjennomsnittstid pr. veke må vere for å nå 500 timar.
3. Innkjøp, er godkjent av Joar. Må gjennomførast
4. Ymse

Aktuelt

2. Gjennomgang av Roy
3. Alereie gjennomført

Ymse

Spørsmål frå Roy, Skal vi ha forskjellige hyttestørrelsar ?

Lik hyttestørrelse, men kolonna forblir i mysql

Ta priseksempel frå Kjørnes camping.

Til neste møte

Jobb vidare med det som vi har jobba med før

Møtereferat er skriv av **Jens Før Sund**.

Referatet vert gjort tilgjengeleg for alle gruppedlem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	18.03.2015	Kl. Start	10.30	Kl. Slutt	11.30
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Sander har laga dei fleste layouts, meiner sjølv han skal være klar med det meste til neste måndag.

Eivind har hatt problem med å få dokumentasjon på bankterminalen, slit med å kommunisere med den. No har han fått tilgang til ftp-serveren til nets og kontakt med ein anna student som div med noko av det same.

Jens har det meste klart, men ting må syast i hop. Har sett på fleire ting som har dukka opp undervegs. Skjermtastatur, fullskjerm, alarm-poppup.

Roy har jobba med rapport, fått en del dokumentasjon frå Eivind. Godt i gang med rapportskriving og lest andre rapporter, vil gjerne ha seg en prat med Preben etter påske om rapportopsettet.

Sakliste

Status arbeidsoppgåver, frå sist møte

2. Status i forhold til gant og framdriftsplan
3. Legge plan for vidare arbeid

Aktuelt

2. Vi ligg bak skjema, det er ikkje sjans å få det ferdig til førstkommande måndag.
3. Vi utsett testing til 9.april. vi kan teste utan bankterminal, men helst med.

Ymse

-Vi må ha ein plan på teststasjon.

Undergruppe, Roy vil være sjef for dette også.

Vi vil at den skal være fin, laga av metall, eller et anna materiale.

Universell utforming??

-Camplett logo må vi ha, Jens vil være sjef for dette også

-Nettside må oppdaterast av og til, Sander sitt ansvar.

Til neste møte

Jens: Logo, overordna programmerar, hjelpe Eivind

Sander: Gjere ferdig layout

Eivind: Få opp bankterminalen

Roy: Fortsette med rapportarbeid, ansvar for teststasjon

Møterefertat er skriv av **Jens Førsund**.

Referatet vert gjort tilgjengeleg for alle gruppedlem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

Prosjektmøte

Dato	03.03.2015	Kl. Start	12.40	Kl. Slutt	13.40
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Jens: Har laga logo, fletting av programmer, css, hjelpt eivind litt

Eivind: No fungerer bankterminalen. Har laga metodar som kan brukast av resten av programmet.

Sander: Noken ting gjenstår, men det meste er begynt å komme på plass.

Roy: Skreve en del rapport, sett på andre rapportar osv. Har designa teststasjon, skal finne noke materiale å lage den av.

Sakliste

1. Status i forhold til gant og framdriftsplan
2. Legge plan for vidare arbeid
 - a. Planlegge testing

Aktuelt

1. Forsatt bak, tatt litt innpå. Det er urealistisk å bruke 500t /stud
2. Vi skal ha sett samman programmet til eit før 1. april.
Vi skal ha test den 9.april

Ymse

Til neste møte

Fortsette å nå måla.

Møtereferat er skriv av **Jens Før Sund**.

Referatet vert gjort tilgjengeleg for alle gruppelemmer omgåande og kvar einskild er sjølve

ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	01.04.2015	Kl. Start	11.00	Kl. Slutt	12.30
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Førstund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Jens: begynt å dokumentere litt, har konsultert Sander og Eivind med samansying av programmet. Har retta fleire feil som dukka opp under vegs.

Eivind: bankterminalen fungerer simpelt saman med resten av programmet. Skrevet ein god del på rapport og levert til PL.

Roy: arbeida en del med rapport, satt in den informasjonen som kjem frå Eivind, korrekturlest osv. Filer med informasjon til rapport kan sendast til prosjektmappa H.rapport

Sander: Føler seg klart til testing no. Sydd saman programmet. Implementert css osv..

Sakliste

Plan vidare arbeid

- b. Status framføring 08.04.2015
- c. Oppstart testing

Aktuelt

- a. Statusframføring, vi klargjer den på kveld 6 april og 7 april.
- b. Oppstart testing utsatt til måndag 13 april, grunna knapt med tid i forhold til presentasjon. Skal lagast backupløysing eller bruke p-sys sin stand i tilfelle vår ikkje blir klar.

Ymse

Database. Uenighet om korleis databsen blir håndtert i forhold til det grafiske.

Design. Alle skal sjå på programmet og prøve slik at vi føler at vi er fornøgd sjølv, Sandet tar ansvar for css

Diskusjon på om vi skal begynne med utviding. Det blir ikkje enda ialfall.

Til neste møte

Lage spørjeskjema, gjere klart til presentasjon, lage ferdig design, sjå om vi har tid til å reise på campingplass.

Sander, design

Roy, prototype

Eivind, spørjeskjema

Jens, sette i hop programmet bistå dei andre

Møtereferat er skriv av **Jens Førstund**.

Referatet vert gjort tilgjengeleg for alle gruppelem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	10.04.2015	Kl. Start	14.00	Kl. Slutt	15.00
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Førstund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Eivind, Ferdig med kundeundersøkelsen, har rette ein del bugs mellom terminalen og systemet

Jens har bistått dei andre , koplå saman programmet, slete leng med touch tastatur..

Sander, har endra design etter ønske frå andre, endra en del funksjonar som ikkje har fungert optimalt

Roy, har laga kasse til standen, skrive rapport, henta dei beste rapportane frå tidlegare og lagt dei på server

Sakliste

3. Status arbeidsoppgåver, frå sist møte
4. Status i forhold til gant og framdriftsplan
5. Legge plan for vidare arbeid
 - a. Planlegge testing
 - b. Rapport
 - c. Holdar Panel PC

Aktuelt

2. i rute i forhold til den utsette fristen.
- 3.

- a. Vi lagar ein knapp for å hoppe over betaling, vi lagar og en knapp for å tømme databasen
Vi må ha plakater og noko gratis, Roy kjøper kjeks, byrja i 10-11 tida.
- b. Skal prate med Marcin, Preben og Kamil. For å få tips osv.
- c. OK

Til neste møte

Kjøre testperiode og same informasjon

Møtoreferat er skriv av **Jens Førstund**.

Referatet vert gjort tilgjengeleg for alle gruppelemmer omgåande og kvar ein skild er sjølv ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	15.04.2015	Kl. Start	10.00	Kl. Slutt	11.00
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Førstund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Jens, lest litt rapportar

Har oppdatert SW med propertyeventlistener, fungre bra, satt inn tøm database knapp, avslutt knapp og hopp over betaling knapp.. testen er i gang.

Sakliste

1. Status frå sist
2. Tilbakemeldinger test så langt
3. Plan vidare arbeid
4. Ymse

Aktuelt

2. tilbakemelding om at programmet er enkel

Knapp bør endre seg når du trykker. Kva betyr ledig rød eller grøn. Innsjekkkdato på kvitteringa. Kr/døgn, pluss fleire ting

3. Testing i dag. Jens og Roy begynner med test, Sander og Eivind begynner på feilretting

Ymse

Til neste møte

Forsette testing og feilretting

Møtereferat er skriv av **Jens Førstund**.

Referatet vert gjort tilgjengeleg for alle gruppemedlem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	06.05.2015	Kl. Start	10.00	Kl. Slutt	11.00
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Førstund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Ferdig med testing på studentar, komme i gang med rapportskriving.

Sakliste

6. Status framdriftsplan
7. Slutt test
8. Pressemelding
9. Rapport
 - a. Kva er klart?
 - b. Korrektur
10. Fordeling arbeidsoppgåver vidare
11. Planlegge tur til Kjørnes

Aktuelt

1. Rapporten skal være ferdig til 13 mai. Etter det skal det være korrektur. Vi må kreve å få korrektur tilbake måndag 18 Mai.
2. Slutttest, blir for oss sjølv og Kjørnes Camping Torsdag 07 mai.
3. Ferdig levert.
4. Mykje er ferdig. Skal begynne å sette saman rapporten, det er valt ut to på teknisk og to på rettskriving til korrektur.
5. Eivind og Roy reiser, Sander og Jens blir igjen.

Ymse

Miljøkoordinator(Jens) utforma eit framlegg på morgonprosedyre.

Det vart bestemt at alle skal være på skulen minimum frå kl 10.00 til 16.00 kvar dag utanom helg fram til 13. mai, vær gjerne lenger.

Til neste møte

Roy Arve begynner å sette saman rapporten. Sander fortsetter med det grafiske, Jens og Eivind gjer ting som trengs å gjerast

Møtereferat er skriv av **Jens Førsund**.

Referatet vert gjort tilgjengeleg for alle gruppedlem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|

|

Prosjektmøte

Dato	12.05.2015	Kl. Start	13.00	Kl. Slutt	14.00
-------------	------------	------------------	-------	------------------	-------

Tilstade			
Jens Før Sund	<input checked="" type="checkbox"/>	Roy Arve Hetle	<input checked="" type="checkbox"/>
Sander Falkenstein	<input checked="" type="checkbox"/>	Eivind N. Inderøy	<input checked="" type="checkbox"/>

Status sidan sist

Tilbakemelding Kjørnes Camping var veldig positiv

Alle har jobba godt med rapport

Sakliste

12. Gjennomgang av oppsett rapport
 - a. Rapport skal til korrektur 13.05 alt må vere klart, slik at det er tid til å sette det inn i rapporten som går ut
13. Oppdatere heimeside
 - a. Må få ut meir info på nettsida
14. Plan vidare arbeid
 - a. Alle kjelde koder må kommenterast, sidan dei skal vere med som vedelegg til rapporten
 - b. Plakat må lagast, skal trykkast på fleten dekor= bør vere klar med design i god tid
15. Gjennomgang, er det noko som er gløymt, har vi kontroll på alt?
 - a. Fristar framover:
 - i. Rapport tilbake frå korrektur, 18mai skal inn 22.mai
 - ii. Plakat skal vere på plass 26.mai
 - iii. Framføring av prosjekt med utspørjing 27.mai
 - iv. All info må vere lagt ut på nettsida før 5.Juni

Aktuelt

1. risikovurdering må inn frå forprosjektrapport, rekneskap, reiserekning.
2. nettsida bør oppdaterast, få ut status på korleis vi ligg ann, tilbakemeldingane frå Kjørnes, pressemelding kan legjast ut.
3.
 - a. Alle kjildekodar må kommenterast, Sander har begynt. Roy høyrer med Joar korleis vi skal levere kjeldekodane

- b. Plakat, alle tenker litt på forslag til design så diskuterar vi det på onsdag-torsdag.
- 4.

Fristar framover:

- i. Rapport tilbake frå korrektur, 18mai skal inn 22.mai
- ii. Plakat skal vere på plass 26.mai
- iii. Framføring av prosjekt med utspørjing 27.mai
- iv. All info må vere lagt ut på nettsida før 5.Juni

Ymse

-Ord frå Java skal skrivast slik dei er i Java med klammer «»

-CampLett om systemet CampLett-programmet om programmet, ikkje bruk programmet vårt, etc

- Ikkje skrive formuleringar som vi og oss og kva vi har gjort utanom kapittelet utvikling.

Til neste møte

Ingen planlagde møter vidare

Møtereferat er skriv av **Jens Før Sund**.

Referatet vert gjort tilgjengeleg for alle gruppemedlem omgåande og kvar einskild er sjølve ansvarlege for å lese gjennom dette. Ein har frist på ei veke for å komme med tilbakemeldingar, om

Opent kommentarfelt til referat

|
|

Vedlegg 7 Risikovurdering

Utfordringar	Sans 1-5	Kons 1-5	Sum	Tiltak	Ny sans	Ny kons	Ny sum
For lite tid	2	5	10	a	1	5	5
Tekniske problem med innlevering	2	5	10	b	2	3	6
Feil mann/verktøy på oppgåva	3	4	12	c	2	3	6
Økonomisk ugjennomførbart prosjekt	1	4	4				4
Ingen ledige grupperom	5	3	15	d	3	2	6
Teknisk svikt, pc osv.	3	4	12	e	3	2	6
Vekkheft av andre	4	3	12	f	2	3	6
Arbeid blir utsatt for lenge	3	4	12	a	1	4	4
Fristar blir glømt/overesett	2	5	10	g	1	5	5
Feil/dårleg tilbakemelding frå styringsgruppa	2	4	8				0
Større utfordringar en venta (java)	3	4	12	h	2	3	6
Manglande element i risikovurdering	3	3	9	i	2	2	4
Feil fokus på oppgåva	2	5	10	a,c	1	4	4
Viktig informasjon blir rota vekk	3	5	15	a,e	1	5	5
Manglar viktige element i rapport	2	5	10	e,k	1	5	5
Dårleg korektur	3	5	15	k	1	5	5
For lite faglig tyngde i rapport	3	5	15	j,l	1	5	5
For dårlig forklart tekniske ting	3	4	12	k	1	4	4
Manglande dokumentasjon på kva vi har gjort	2	4	8	e,j	1	4	4
For høgt fokus på arbeid kontra rapport	3	5	15	j,m	2	4	8
Vanskeleg å få konstruktive tilbakemeldingar frå faglærar	3	4	12	n	2	3	6
Sabotasje frå andre	1	5	5				0
Problem med opphavsrett på bilde ect.	2	3	6				0
Folk lagrar i samme dokument på samme tid	4	4	16	o	2	4	8
Dårleg kommunikasjon i gruppe gjer til at fleire jobber m	3	4	12	c	1	4	4
Tar på oss meir enn vi har tid til	3	4	12	h	2	4	8
For mykje anna skulearbeid	2	4	8				0
Problem med barn	3	3	9	p	3	2	6
Alvorlig Sjukdom i gruppa	1	5	5	p	1	3	3
Alvorlig ulykke i gruppa	1	5	5	p	1	3	3
Frivillig arbeid blir prioritert framfor prosjekt	2	4	8				0
Folk møter ikkje på prosjektmøte	3	3	9	q,r	1	3	3
Sein kveld på akutten	5	1	5				0
For lite dokumentert dagbok/timeliste	4	4	16	j	1	4	4
Sjukdom i nær familie	2	3	6	p	1	3	3
Dødsfall i nær familie	2	3	6	p	1	3	3

Tiltak	Tiltak for forbedring
a	Begynne tidlig med faste møter og klare arbeidsmål
b	Sette av tid til å levere tidelgare en fristen
c	Må være klare på oppgåvene og være åpne for å bytte på roller, hyppige statusmøter
d	Booke rom tidleg, og sjå på andre mulige stader å arbeide
e	Ta backup av dokument ofte, huske å lagre, alt skal lagrast på dropbox
f	Bruke grupperom og kveldar mest mogeleg
g	Gruppeleiar har ansvar for å dagleg halde seg oppdatert på fronter
h	Begrense oppgåva frå starten av og ha muligheter for utviding
i	Risiokovurderinga er open for endring under heile prosjektet
j	Kvar enkelt skal føre timeliste og dagbok for kvar dag noko er gjort
k	Rapporten skal korrekturlesast av fleire uavhengige og utenforstående
l	Sette fokus på det faglege, skrive rapport for å beskrive produktet
m	Satt av tid til rapport, ein person har hovudansvar for raport
n	Gi beskjed til faglærer om kva vi vil ha tilbakemelding, bruke andre en faglærer i tillegg
o	Vi skal sjå på andre muligheter, google docks. Om ikkje må opning av dokument avtalast
p	vere fleksible på tid for møte og arbeid. Førde dagbok og timelister slik at det er enkelt for andre og ta over om det trengs
q	Alle må føle ansvar for prosjektet, PL er ansvarlig for å kalle inn til møter!
r	Om nokon er meir enn 20 min. sein til oppsett tid, må han spandere kaffi den dagen

Vedlegg 8 SWOT

	Indre forhold	Ytre forhold
	Styrker	Muligheter
Positivt	Fleksibilitet i produktet	Behov for produktet
	Mogelegheit for begrensing	Vi kan byggje vidare på andre system som alle
	Produktet skal ha mulighet for å enkelt kunne utvides	Gode tilbakemeldingar frå markedet
	Gratis produktutvikling	Er vi utan konkurranse?
	Vi er ikkje avhengige av økonomisk overskot	Kan vi få til eit krav, eller ønske frå myndighei
	Bygges ved hjelp av gratis programvare (netbeans etc)	Kan vi få økonomisk støtte?
	SW har låge kostnader etter utvikling	Vi ønskjer at kunden kan tene på å kjøpe prod
	Ha tru på det vi gjer, visjonar, vil sjå resultat	mulighet for å kople til online ?
	God planlegging, Hyppige statusmøter, Kvalitetsikring	verda blir meir og meir tekologisk
	Mulighet til å justere prosjektet etter kvart	kan tilpassast til ulike kunde
	God skriftleg dokumentasjon	Tilgang på sterk kompetanse i lærarar
	lettare å selje et produkt når vi har fungerande sw	mulighet for teststasjon
	mykje teknisk kompetanse i gruppa	svak kronekurs, auke av tursitnæring
	gruppa har interesse for produktet	
	drivkraft i karakter	
	fleksibelt produkt	
Godt samarbeid, føler seg trygg på gruppa, nær kontakt på fritida		
one of a kind		
norske folk bak produktet		
fokus på kunde heile tida		
	Svakheter	Trusler
Negativt	Vi kan bli for einspora og låse oss til løysingar som kanskje ikkje er optimale	Markedet ønskjer eit anna produkt en det vi g
	Tidsramma blir for lita for å gjennomføre prosjektet	Lover og reglar jobbar oss imot
	Om vi ikkje klarar å lage ein fungerande prototype	Nokon stel ideen vår
	Underkjent forprosjekt	Negativitet i marknaden
	Feil fokus under produktutvikling, lagar noko vi ikkje treng	Ikkje behov i marknaden
	vanskelig å lage fint design	Eksisterer det eit tilsvarande eller konkurrerar
	vanskelig å bli fornøgd med eget resultat	Behov for support / oppdateringer
	fare for at ting går treigt i SW	useriøst med studenter
	vi har ingen pris på full skala system	hisf domene som ikkje når ut til folk
	lite erfaring	ikkje interessert i å legge så mykje pengar i de
	lite som skal til for å felle bedrifta	blir det for avansert for eldre
	når vi er ferdig med bachelor kjem vi til å gi oss	vi er nødt til å kjøpe en del ting av andre leverer

Ytre forhold

Muligheter

Behov for produktet

Vi kan bygge vidare på andre system som allerede eksisterer

Gode tilbakemeldingar frå markedet

Er vi utan konkurranse?

Kan vi få til eit krav, eller ønske frå myndigheitene om slikt produkt ?

Kan vi få økonomisk støtte?

Vi ønskjer at kunden kan tene på å kjøpe produktet, økonomisk og/eller i form av fritid.

mulighet for å kople til online ?

verda blir meir og meir tekologisk

kan tilpassast til ulike kunde

Tilgang på sterk kompetanse i lærarar

mulighet for teststasjon

svak kronekurs, auke av tursitnæring

Trusler

Markedet ønskjer eit anna produkt en det vi gjer.

Lover og reglar jobbar oss imot

Nokon stel ideen vår

Negativitet i marknaden

Ikkje behov i marknaden

Eksisterer det eit tilsvarande eller konkurrerande produkt ?

Behov for support / oppdateringer

useriøst med studenter

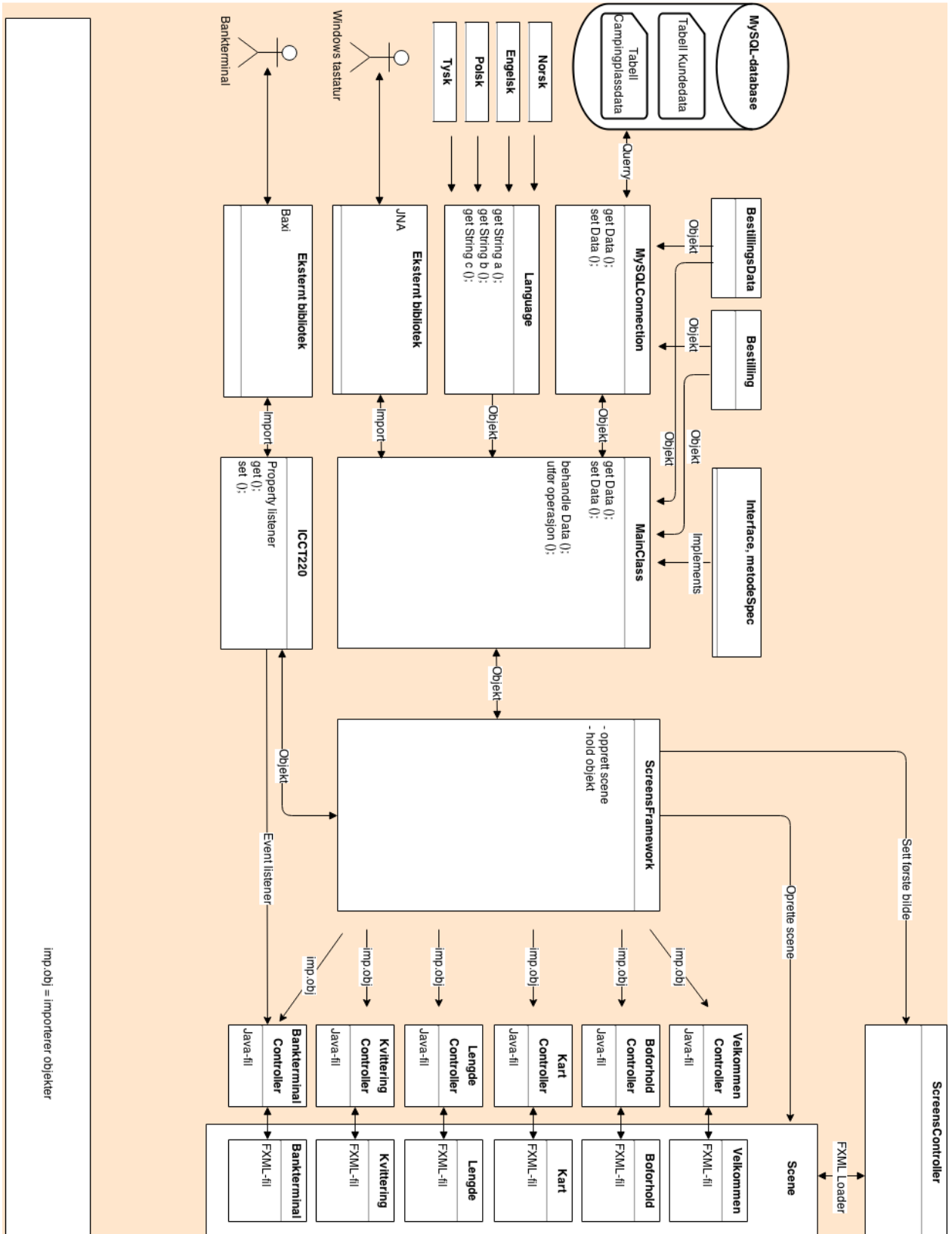
hisf domene som ikkje når ut til folk

ikkje interessert i å legge så mykje pengar i det sjølv

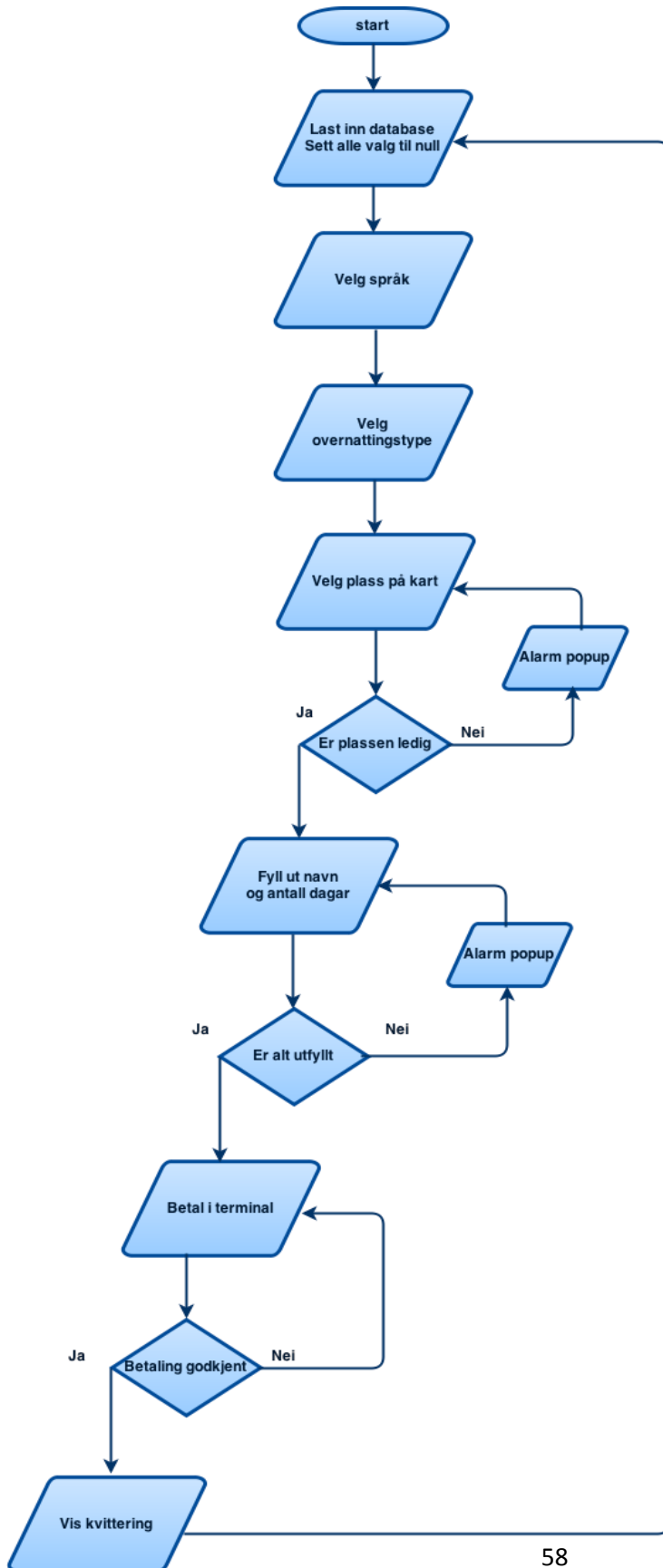
blir det for avansert for eldre

vi er nødt til å kjøpe en del ting av andre leverandører

Vedlegg 9 Klassediagram



Vedlegg 10 Flytskjema



Vedlegg 11 Terminal utviding (iUN)



Unattended payment terminal iUN

iUN is a payment terminal for integration in self-service machines and other unattended points of sale. iUN has modular design, which implies that the machine vendor installs the components needed for desired functionality: Card reader, PIN pad and/or Contactless reader. iUN is a complete, uniform Nordic solution with all necessary security and hardware certifications.

iUN is a flexible, PCI PTS 3.x certified payment solution for all industries. iUN handles all types of card payments in addition to contactless payments by card or mobile phone. iUN is resistant to removal and vandalism, and can withstand cold winter days as well as warm sunny days. iUN has a well proven integration protocol and several communication interfaces can be applied.

The PIN pad (iUP250) with tactile keys and intuitive visual guiding by coloured LED lights makes iUN user friendly also for the visually impaired. iUN's "sleep mode" ensures low power consumption and is beneficial for solar powered or battery powered machines.

EXAMPLES OF APPLICATION AREAS

- Vending machines
- Ticketing: public transport, cinemas etc.
- Parking meters in streets and parking houses
- Hotel self check-in/self check-out
- Petrol machines
- Fast food restaurants; ordering and payments
- Charging machines for tickets
- Car washing machines

BENEFITS WITH IUN

- Nordic solution: Uniform hardware and software across borders
- Used by customer without assistance from staff – enables self-service around the clock
- Always updated and compliant Nets software
- Less manual labour, staff can concentrate on core business
- Customer support in all Nordic countries, in local languages
- No-PIN solution (no CVM) for low amount payments and intangible purchases

IUN HARDWARE DEVICES



PIN PAD IUP250



CARD READER IUR250



CONTACTLESS READER IUC150



CONTACTLESS READER IUC180B



iUN DEVICES, TECHNICAL DATA

X = Included / Yes



iUP250



iUR250



iUC150



iUC180B

	iUP250	iUR250	iUC150	iUC180B
PROCESSOR				
TYPE	Risc 32-bits ARM7 Risc 32-bits ARM9	Risc 32-bits ARM7	Risc 32-bits ARM9	Risc 32-bits ARM9
SPEED	450 MIPS+50 MIPS	50 MIPS	450 MIPS	450 MIPS
MEMORY				
RAM / FLASH	16MB / 128MB	96Kb / 512Kb	8MB / 16MB	16MB / 128MB
COMMUNICATION MODE				
ETHERNET	X			X
RS-232	X			X
CARD READERS				
SMART CARD		X		
MAGSTRIPE		ISO V2/3		
HYBRID		X		
CONTACTLESS			ISO 14443A/B	ISO 14443 A/ B
DISPLAY				
GRAPHIC 128 X 64 PIXELS	X			X
BACKLIT RGB	X			X
BLACK & WHITE	X			X
KEYBOARD				
NUMBER OF KEYS	15+2			
KEY TEXTURE	Metallic			
BUZZER				
BUZZER	X	X	X	X
CONNECTIONS				
RS 232	1			1
USB HOST	4			4
USB SLAVE	1	1	1	1
MDB SLAVE	1			1
POWER SUPPLY				
EXTERNAL POWER SUPPLY	12V-30V DC	5V USB	5V USB	12V-30V DC
STAND BY MODE	X	X	X	X
SIZE (MM)				
OVERALL WxHxD	120x132x45	72x108x140	12x134x53	120x134x62
CUT AREA WxH	107x85	74x62	107x85	107x85
ENVIRONMENT				
OPERATING TEMPERATURE	-20 C to +65 C	-20 C to +65 C	-20 C to +65 C	-20 C to +65 C
STORAGE TEMPERATURE	-20 C to +65 C	-20 C to +65 C	-20 C to +65 C	-20 C to +65 C
RELATIVE HUMIDITY, NON-CONDENSING	90 % HR at + 55 C	90 % HR at + 55 C	90 % HR at + 55 C	90 % HR at + 55 C
IP				
INGRESS PROTECTION	IP65	IP34	IP65	IP65
IK				
SHOCK PROTECTION	IK10	IK10	IK10	IK10
PCI PTS				
3x	X	X	X	

The iUN devices are embedded in their enclosure and integrated according to EVA and IMB market standards (European Vending Association and International Metallic Base). The iUN devices stands up to the most demanding outdoor environments due to being proofed against vandalism and extreme weather.

Nets | Merchant Solutions | www.nets.au



Vedlegg 12 ICT 220

Ligger i Zip-mappe tilgjengelig for sensor og faglærer

Vedlegg 13 Kjeldekodar for CampLett-programmet

Ligger i Zip-mappe tilgjengelig for sensor og faglærer