



CHALMERS

Utveckling av ett webbaserat bokningsystem

Examensarbete inom Högscoleingenjörsprogrammet i Datateknik

JIMMY SVENSSON

VICTOR EVERTSSON

Utveckling av ett webbaserat bokningssystem

Jimmy Svensson, Victor Evertsson

© JIMMY SVENSSON, VICTOR EVERTSSON, 2014

Institutionen för data- och informationsteknik
Chalmers tekniska högskola
412 96 Göteborg
Tel: 031-772 1000
Fax: 031-772 3663

Institutionen för data- och informationsteknik
Göteborg, 2014

Innehållsförteckning

1	Inledning.....	1
1.1	Syfte	1
1.2	Mål	1
1.3	Avgränsningar	1
1.4	Frågeställningar	1
2	Arbetsmetodik.....	2
3	Teknisk bakgrund.....	3
3.1	MySQL.....	3
3.2	MSSQL.....	3
3.3	PHP.....	3
3.4	ASP.NET.....	4
3.5	Eclipse	4
3.6	Dreamweaver	5
3.7	JavaScript	5
3.8	JQuery	5
3.9	XAMPP	5
3.10	WAMP	6
3.11	SSL.....	6
4	Genomförande	7
4.1	Kravspecifikation	7
4.1.1	Funktionsval.....	8
4.2	Analys.....	9
4.2.1	MySQL tillsammans med PHP	9
4.2.2	MSSQL tillsammans med ASP.NET.....	9
4.2.3	Val av databas och kommunikationsspråk.....	10
4.2.4	XAMPP eller WAMP?	10
4.2.5	Eclipse eller Dreamweaver?	10
4.2.6	Valda verktyg	11
4.3	Design.....	12
4.3.1	Webbsidans layout	12
4.3.2	Inloggningsidans layout	13

4.3.3	Kodstruktur	13
4.4	Implementation.....	15
4.4.1	Databas.....	15
4.4.2	Säkerhet.....	17
4.4.3	Inloggning.....	17
4.4.4	Registrera användare.....	18
4.4.5	Lagra maskin.....	19
4.4.6	Lagra tillbehör.....	20
4.4.7	Bokning.....	21
4.4.8	Mina bokningar.....	26
4.4.9	Sökfunktion.....	27
4.4.10	Maskinöversikt.....	28
4.4.11	Inställningar	30
4.4.12	Raderafunktioner.....	31
4.5	Testning.....	33
5	Resultat.....	34
5.1	Resultat av produkt.....	34
6	Slutsats.....	36
6.1	Resumé.....	36
6.2	Kritisk diskussion.....	36
6.2.1	Webbsidans funktionalitet	36
6.2.2	Utvärdering av verktyg	37
6.2.3	Kodstruktur	38
6.2.4	Säkerhetsaspekter.....	38
6.2.5	Tidsplan och verklighet.....	39
6.2.6	Metod och verklighet	39
6.3	Teknikens roll i samhället	39
6.4	Vidareutveckling	40
	Referenser.....	41
A	Appendix.....	44

Abstract

This thesis deals with a web-based reservation system that facilitates management by advance of construction machinery.

The report describes the implementation of the functions of a website and how a database was involved. It describes design decisions and how usability has been in focus when the system was created. During the development, expandability has been kept in mind.

The results of the thesis became a well-functioning reservation system that follows the company's requirement of functionality and additional functions that were considered to be useful for the reservation system.

This report is written in Swedish

Sammanfattning

Detta examensarbete behandlar ett webbaserat bokningssystem som underlättar hanteringen vid förhandsbeställning av entreprenadmaskiner.

Rapporten redogör för implementeringen av funktioner på en webbsida och hur en databas involverades. Det beskrivs om designbeslut och hur användarvänlighet har varit i fokus när systemet skapades. Vid utvecklingen har utbyggbarhet funnits i åtanke.

Resultatet av examensarbetet blev ett väl fungerande bokningssystem som följer företagets krav av funktionalitet och ytterligare funktioner som ansågs vara användbara för bokningssystemet.

Förord

Detta examensarbete är skrivit på Chalmers dataingenjörsprogram (180 högskolepoäng) och är det slutgiltiga steget för att få ut en högskoleingenjörsexamen. Det omfattar 15 högskolepoäng och görs under 10 veckor.

Vi vill tacka vår handledare och examinator Christer Carlsson för bra handledning och bra respons om rapporten samt att han alltid varit tillgänglig under projektets gång.

Slutligen vill vi tacka de anställda på Wangeskog Hyrcenter AB som gjort detta examensarbete möjligt och för att de varit väldigt tydliga med den funktionalitet som de ville ha med i bokningssystemet.

Definitioner

Databas	En samling data som är strukturerad på ett speciellt sätt
Entreprenadmaskin	Maskin avsedd för arbeten inom t.ex. vägunderhåll, byggen, rivning, markarbeten.
HTTP	Ett kommunikationsprotokoll för att överföra webbsidor till klienten.
HTTPS	Ett kommunikationsprotokoll där transporten av data från HTTP-protokollet krypteras.
Klient	Motsvarar en användare som utför förfrågningar eller tar emot en tjänst.
Kod-editor	Program för att redigera textbaserad kod.
Kryptografisk hashfunktion	En funktion som genererar ett värde, vilket försvårar möjligheten att hitta det ursprungliga värdet.
Plattform	En befintlig miljö för mjukvara att köras inuti och lyda bestämda begränsningar.
Plug-in	Mjukvarukomponent som utökar en specifik funktionalitet till ett mjukvaruprogram.
Product backlog	Del av Scrum-processen, där det bestäms vilka uppgifter som skall utföras under produktutvecklingen.
Server	Ett system som besvarar förfrågningar över ett datornätverk för att förses med en nätverkstjänst.
Sprint backlog	Del av Scrum-processen, där det bestäms vilka uppgifter som skall utföras under en viss tidsperiod framöver.

1 Inledning

Wangeskog Hyrcenter AB är ett maskinuthyrningsföretag lokaliserat på fyra platser i Sverige, två i Göteborg, ett i Stockholm samt ett i Malmö.

Företaget har införskaffat dyra entreprenadmaskiner. För att göra inköpen lönsamma är det viktigt att maskinerna är uthyrda till kunder större delen av året. Maskinerna kommer därför att transporteras mellan de olika maskinuthyrningsplatserna när det är kostnadseffektivt.

I dagsläget hanteras förhandsbokningen av maskiner manuellt. Detta gör administrationen av maskinerna mindre effektiv.

1.1 Syfte

Syftet med detta examensarbete är att göra ett datorbaserat system för att underlätta hantering och administration av förhandsbokningar av entreprenadmaskiner inom företaget.

1.2 Mål

Målet med examensarbetet är att skapa ett webbaserat system där företaget kan logga in för att verifiera entreprenadmaskinernas tillgänglighet samt boka dem.

1.3 Avgränsningar

En avgränsning är att endast de anställda kommer att använda bokningssystemet. Det finns alltså ingen möjlighet för kunder att boka en maskin via detta system.

En annan avgränsning är att det inte kommer att implementeras någon beräkning av kostnaden för de bokade maskinerna. Detta eftersom företaget redan använder ett debiteringsprogram.

1.4 Frågeställningar

- Hur skapas ett webbaserat bokningssystem? (se kapitel 4)
- Vilka programmeringsspråk är bäst för detta examensarbete? (se kapitel 4.2.3)
- Vilka verktyg lämpar sig bäst vid användning av de valda programmeringsspråken? (se kapitel 4.2.6)
- Hur görs ett webbaserat system säkert? (se kapitel 4.4.2)

2 Arbetsmetodik

Scrum [1] är den metodik som kommer att användas för att förenkla arbetets tillvägagångssätt. Det blir en förenklad version av Scrum eftersom alla metoder inte kommer att användas, dock medför Scrum att examensarbetet blir mer strukturerat och lättöverskådligt.

En förstudie skall göras för att identifiera vilka verktyg som kan användas i projektet. Faktainsamlingen kommer att bestå av webb- och litteratursökningar. En fördjupad analys kommer att utföras på de lämpligaste verktygen. Detta för att komma fram till ett beslut om vilka verktyg som skall användas.

Tidigt i projektet kommer en enkel uppbyggnad av webbsidan att skapas. Därefter skall en databas konfigureras. Detta för att kunna sammankoppla webbsidan och databasen. När det fungerar, kommer utvecklingen av webbsidan och databasen att ske parallellt.

Tidsplan finns i Appendix.

3 Teknisk bakgrund

I detta kapitel redogörs om de verktyg och språk samt den teori som ligger till grund för projektet. Utifrån denna teori fås en djupare förståelse i skapandet av en användbar och säker webbsida.

3.1 MySQL

MySQL är ett databashanteringssystem ursprungligen utvecklad av det svenska företaget MySQL AB men ägs nu av Oracle Corporation. MySQL är det mest använda databashanteringssystemet med öppen källkod. MySQL använder sig av SQL (Structured Query Language) och är det vanligaste standardspråket för att få tillgång till en databas [2].

MySQL är en relationsdatabas vilket betyder att istället för att lägga all data i en stor tabell så separeras data i mindre tabeller som har relation till varandra. Databasen använder objekt såsom kolumner, rader, vyer och tabeller som gör det flexibelt att programmera databasen. Det går att lägga till regler mellan olika tabeller i databasen som t.ex. unika, en-till-en, en-till-många, måsten eller valfria relationer. Dessa regler gör att man aldrig får duplicerad, inkonsekvent, inaktuell eller saknad data om designen på databasen är bra [2].

3.2 MSSQL

MSSQL står för Microsoft SQL Server och är ett databashanteringssystem utvecklat av Microsoft för att fungera tillsammans med ASP.NET. Databasen är en relationsdatabas och är den mest använda databasen på Windows-servrar. Källkoden är stängd och har därför en kostnad [3].

3.3 PHP

PHP står för hypertext preprocessor och är ett skriptspråk för att skapa dynamiska webbsidor [4]. Detta uppnås genom att PHP körs på en server som utför PHP-operationer och sedan skickar resultatet till klienten [5]. Med hjälp av PHP är det möjligt att kommunicera med en MySQL-databas och därmed kunna lagra samt hämta information från tabeller i en databas [6]. Eftersom PHP är ett skriptspråk tolkas och exekveras koden i ett annat språk t.ex. C/C++ och gör att koden inte behöver kompileras. Detta medför att kompileringstiden undviks och att webbsida laddas lika snabbt varje gång [5].

När man jobbar med webbsidor som hanterar databaser finns det ett antal säkerhetsaspekter man bör ta hänsyn till för att skydda informationen som är sparad.

SQL-injection är en av de vanligaste attackerna i applikationslagret och används främst för att stjäla eller ta bort data från organisationernas databaser. Den metod som används vid SQL-injections för att göra intrång i databasen är att det skrivs ett SQL-kommando i ett inmatningsfält. Detta kan leda till att någon utför farliga operationer och därmed modifierar innehållet i databasen [7] [8].

Det går att skydda sig mot denna typ av attack genom att använda begränsade behörigheter för varje användare, kontrollera att varje inmatning som användaren gör är av rätt datatyp och använda så kallade "prepared statements" med bundna variabler. Prepared statements betyder att SQL-koden separeras från användarens inmatningar. Detta gör det lättare att urskilja vad som är SQL-kod och vad som är användarens inmatade text [9].

Användarens inmatningar kan man aldrig lita på oberoende om det är från checkboxar eller gömda fält [9]. För att förhindra felaktiga inmatningar kan man kontrollera att inmatningen t.ex. ett användarnamn endast består av alfanumeriska tecken (bokstäverna A–Z och siffrorna 0–9). Det bör också förhindras att flera SQL-förfrågningar sker i samma inmatningsfält. Syntaxfel från databasen bör inte heller visas på webbsidan [10].

3.4 ASP.NET

ASP.NET står för Active Server Pages som ingår i .NET-ramverket. ASP.NET är en modell utvecklat av Microsoft för att skapa dynamiska webbsidor. Dessa webbsidor programmeras i språk som är kompatibla med .NET-ramverket t.ex. Microsoft Visual Basic och C#. Modellen möjliggör kommunikation med en Microsoft SQL Server för att lagra och hämta information. ASP.NET-kod kompileras första gången den laddas på en webbsida och sparas i cacheminnet. Detta innebär att sidan finns tillgänglig snabbare nästa gång den laddas [11] [12] [13].

3.5 Eclipse

Eclipse är en Javabaserad plattform med öppen källkod utvecklad för att skapa integrerade applikationer och webbverktyg. Eclipse gör det möjligt för utvecklare att använda sig av speciella plug-in [14].

För att utveckla och programmera i språk som PHP, JavaScript och HTML finns ett plug-in som heter PDT (PHP Development Tools) och det används tillsammans med Eclipse. PDT-pluginet gör att man enkelt kan se om det uppstår fel när man programmerar i något av de kompatibla språken. Detta medförs att man undviker testköra kod med uppenbara fel [15].

3.6 Dreamweaver

Dreamweaver är en kod-editor för utveckling av webbapplikationer. Verktøget är ursprungligen utvecklat av Macromedia som nu ägs av Adobe. Editorn stödjer bl.a. HTML, C#, Visual Basic, CSS, ASP och XML. Dreamweaver har också många olika funktioner såsom visuell CSS-editor och automatiska kodgeneratorer. Detta verktyg använder dock inte öppen källkod och har därför en fast månadskostnad [16].

3.7 JavaScript

JavaScript är ett av de populäraste programmeringsspråken i världen och används i samband med skapandet av webbsidor. JavaScript är ett objektorienterat skriptspråk utvecklat av Netscape. Språket används med taggar inuti HTML-kod för att skapa funktioner som gör webbsidan mer lättanvänd [17].

Syntaxen för att skriva JavaScript liknar C++ och Java, vilket gör det lättare för utvecklare med dessa kunskaper att börja använda språket. Java och JavaScript är dock helt olika typer av programmeringsspråk och skall inte förväxlas [17].

3.8 JQuery

JQuery är ett objektorienterat ramverk till JavaScript som gör det lättare att skapa den kod som behövs för att skapa användbara funktioner till en webbsida [18].

Andelen webbsidor som använder JQuery har ökat för varje år. Av de 100 000 högst rankade webbsidorna använder 61,9% JQuery. Detta är en ökning från förra året (2013) med 10 %. JQuery har stöd för Internet Explorer 6 och uppåt samt alla andra webbläsare [18].

3.9 XAMPP

XAMPP står för "X ("cross"-platform), Apache HTTP Server, MySQL, PHP och Perl" [19]. Det är ett verktyg för att använda en lokal Apache-webbserver där man har möjlighet att tolka skriptspråken PHP och Pearl samt har tillgång till en MySQL-databas. Verktøget stödjer ett antal andra funktioner som OpenSSL, vilket kan användas för att kryptera anslutningen mellan klient och server [20]. Med hjälp av detta verktyg kan man enkelt testköra PHP-kod via den lokala servern. Man kan då kontrollera koden samt databasen innan den läggs upp på en global webbserver [21].

3.10 WAMP

WAMP står för Windows, Apache, MySQL och PHP. WAMP är likt XAMPP ett verktyg för att använda en lokal Apache-webbserver där man kan använda PHP och MySQL. WAMP fungerar endast på Windows system [22].

3.11 SSL

SSL står för Secure Sockets Layer och är ett standardsätt för att skapa en krypterad anslutning mellan en server och en klient (t.ex. mellan en webbsida och en webbläsare) [23].

Genom användning av SSL kan känslig information överföras säkert över Internet. Om en webbsida är SSL-krypterad finns en lås-ikon eller liknande i URL-fältet. Om man trycker på symbolen visas information om den krypterande anslutningen [23].

För att skapa en säker anslutning måste ett SSL-certifikat användas. Certifikatet använder en publik och en privat nyckel tillsammans med en identitet av webbsidans ägare för att åstadkomma den säkra anslutningen [24].

För att utfärda ett certifikat måste en Certificate Signing Request (CSR) skapas. CSR gör en privat nyckel och en CSR-datafil. Datafilen skickas till en certifikatutfärdare. Utfärdaren använder datafilen för att skapa ett certifikat och en publik nyckel som matchar den privata utan att den privata nyckeln behöver skickas till utfärdaren. SSL-certifikatet skickas sedan till webbsidans ägare som kan installera certifikatet på webbservern. Den publika nyckeln användas därefter för att skapa en säker anslutning mellan klienterna och webbservern [23].

4 Genomförande

4.1 Kravspecifikation

Företaget efterfrågade ett webbaserat bokningssystem för entreprenadmaskiner. Det skall användas internt på företaget för att verifiera maskinernas tillgänglighet samt boka dem. Det fanns även ett antal funktioner företaget ville ha implementerat:

- Inloggning
- Registrering av användare
 - Namn
 - E-postadress
 - Användarnamn
 - Lösenord
- Tre olika användarbehörigheter
 - Administratör
 - Uthyrare
 - Observatör
- Lagring av maskin/tillbehör
 - Artikelnummer
 - Tillverkningsnummer
 - Pris
 - Maskintyp
 - Inköpsort
- Bokning av maskin
 - Artikelnummer
 - Kund
 - Bokningsperiod
- Tillbehör till maskiner skall kunna bokas
- Redigera en bokning
- Visa en bokning

Då bokningssystemet skall vara internt är tanken att registreringsfunktionen placeras efter att inloggningen skett. När en ny användare skapas skall namn, e-postadress, användarnamn och lösenord lagras.

Företaget efterfrågade tre olika användarbehörigheter: administratör, uthyrare och observatör. Dessa får tillgång till olika funktioner på webbsidan. En administratör skall ha tillgång till all funktionalitet, en uthyrare skall kunna boka maskiner samt verifiera maskinernas tillgänglighet och en observatör skall endast kunna kontrollera maskinernas tillgänglighet.

Kravet vid lagring av en maskin eller ett tillbehör är att artikelnummer, tillverkningsnummer, pris, maskintyp och inköpsort (den egna uthyrningsplatsen som köpt maskinen) skall inkluderas. Vid en bokning skall artikelnummer, kund samt bokningsperioden för maskinen lagras. En bokning skall även kunna visas och redigeras.

4.1.1 Funktionsval

Efter funderingar och övervägningar med företaget implementerades även följande funktioner:

- Visualisering av egna bokningar
- Redigerbara användaruppgifter
- En sökfunktion
- Radering av användare, maskin och tillbehör.

Visualiseringen av de egna bokningarna implementerades för att kunna verifiera att innehållet i bokningarna är korrekt samt att det ger en överblick över gjorda bokningar.

Redigerbara användaruppgifter implementerades ifall namn, e-postadress eller liknande ändras.

Sökfunktionen implementerades för att ge ett alternativt sätt att kunna boka maskinerna samt verifiera deras tillgänglighet utan att veta all information om dem.

Radera användare implementerades ifall någon anställd på företaget slutar och därmed inte skall ha tillgång till bokningssystemet längre.

Radera maskin och radera tillbehör implementerades ifall de säljs eller skrotas.

4.2 Analys

För att utveckla en webbsida och en databas finns det många olika språk, verktyg och system som kan användas. De språk som analyserades för databashantering samt databaskommunikation var: MySQL tillsammans med PHP samt MSSQL tillsammans med ASP.NET. Detta eftersom PHP har stöd för MySQL-operationer utan extra konfigurering och ASP.NET har stöd för MSSQL-operationer utan extra konfigurering.

4.2.1 MySQL tillsammans med PHP

MySQL är en av världens mest använda databaser som använder öppen källkod. Detta gör att många webbhotell använder sig av denna typ av databas och därmed blir utbudet större. Det finns också många gratis webbhotell som använder MySQL, vilket kan vara användbart om man inte har stora prestandakrav på sin databas. Detta eftersom gratisversionerna ofta har lagrings- och trafikbegränsningar.

MySQL har ett välutvecklat verktyg som gör det snabbare och enklare att skapa tabeller i databasen. Verktøget heter phpMyAdmin [25] och används ofta i webbhotell för att hantera en MySQL-databas. Programmet har en grafisk uppbyggnad som enkelt ger en överblick av alla tabeller och sparade datavärden. Det går att lägga till tabeller grafiskt eller med SQL-kod beroende på vad användaren föredrar. SQL-koden med innehållet i databastabellerna kan exporteras, vilket gör att man enkelt kan säkerhetskopiera sin databas.

PHP passar bra ihop med MySQL och har inbyggt stöd för att utföra MySQL-operationer. Detta gör att man inte behöver extra drivrutiner och kan därför börja använda databasen snabbare [26].

4.2.2 MSSQL tillsammans med ASP.NET

MSSQL är en populär databas med stängd källkod. Detta gör att de flesta webbhotell som erbjuder MSSQL kräver en månadsavgift.

Verktyg för att utveckla MSSQL är bl.a. SQL Server Management Studio och det används vid konfigurering, administration och hantering av SQL-servern. Verktøget kombinerar olika grafiska verktyg med skript-editors beroende på vad användaren föredrar [27]. En viktig funktion som användaren får tillgång till är Object Explorer som används för att hantera de olika objekten inom servern. Object Explorer har en hierarkisk vy i tabellform där användaren kan välja, bläddra och utföra olika funktioner [28].

ASP.NET passar bra ihop med MSSQL och har stöd för alla Microsofts funktioner.

4.2.3 Val av databas och kommunikationsspråk

En jämförelse mellan de analyserade systemen gjordes. Det visade sig att PHP/MySQL har en liten fördel i prestanda när man hämtar data från databasen och skriver ut den på webbsidan jämfört med MSSQL/ASP.NET, men skillnaden är i princip inte märkbar. Utvecklingstiden kan ta längre tid med ASP.NET, eftersom mer kod behövs för komplex funktionalitet. Dessutom behöver koden kompileras varje gång någon ändring har kommit till vilket undviks med PHP [29].

En annan fördel är att vi redan har viss erfarenhet av PHP tillsammans med MySQL.

Slutsatsen blev att PHP tillsammans med MySQL valdes för att genomföra detta examensarbete.

4.2.4 XAMPP eller WAMP?

En jämförelse mellan verktygen XAMPP och WAMP genomfördes. Skillnaden mellan verktygen är liten men XAMPP har stöd för t.ex. Perl och SSL, vilket WAMP saknar. Detta är användbart eftersom det planeras att använda SSL. En annan skillnad är att WAMP bara finns för Windows medan XAMPP finns för de flesta operativsystem. Detta vägde dock inte så tungt i analysen eftersom webbsidan ska skapa på Windows datorer.

Slutsatsen blev att använda XAMPP, eftersom det har extra funktionalitet som bedömdes vara nödvändig för detta examensarbete.

4.2.5 Eclipse eller Dreamweaver?

Dreamweaver har många fördelar som förenklar arbetet med designen på webbsidor, eftersom man kan programmera och ändra design samtidigt. Det finns också många kodgeneratorer via verktyget. Dock skapar de ofta onödigt mycket kod vilket är överflödigt. För att använda Dreamweaver behöver man betala en fast månadskostnad.

Eclipse med PDT-plugin är mer fokuserad på programmeringskod och har förslag på PHP-funktioner när man använder det. Eclipse har även öppen källkod, vilket innebär att det är gratis att använda. Utifrån dessa anledningar valdes det att använda Eclipse [30] med PDT-plugin [31].

4.2.6 Valda verktyg

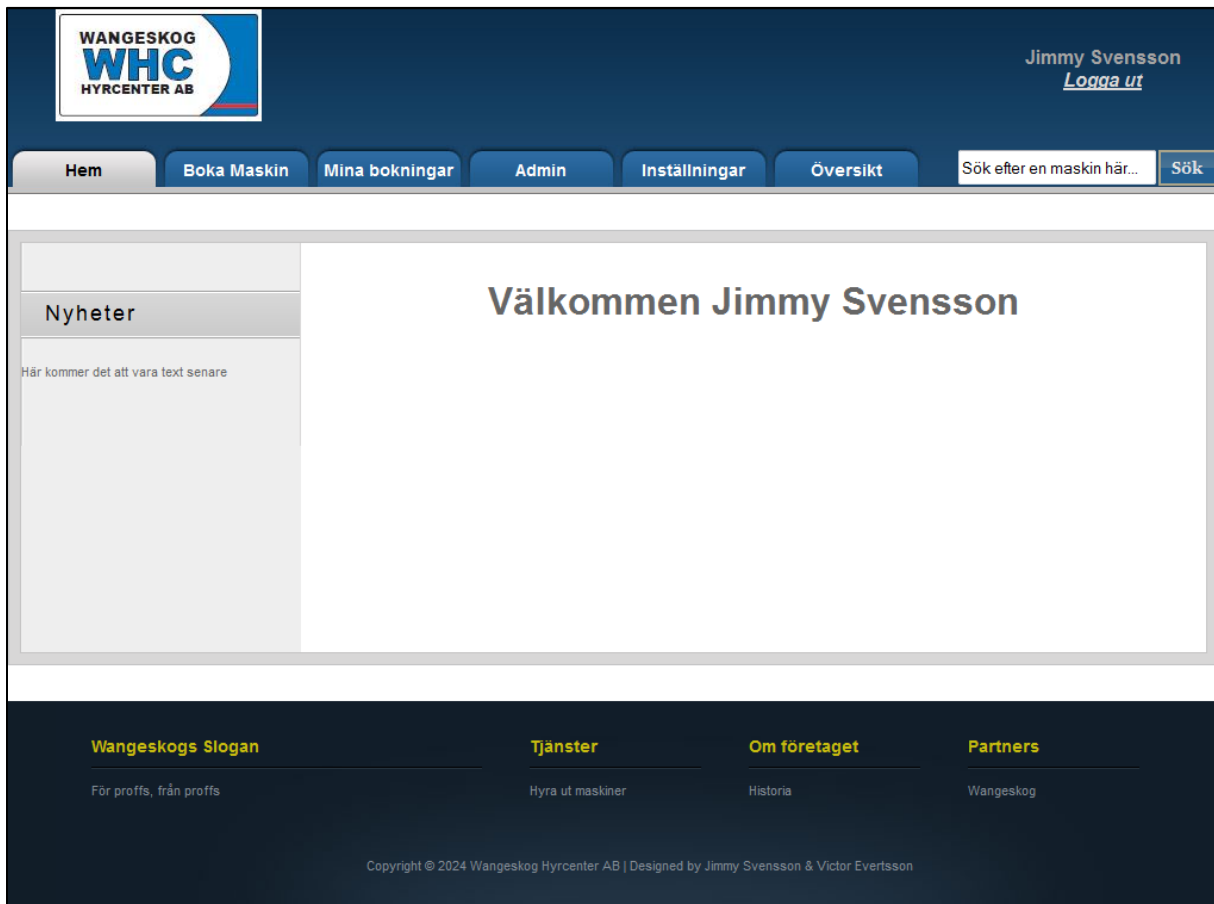
Eclipse kommer att hantera HTML-, JavaScript-, PHP- och JQuery-kod under projektet. Eftersom Eclipse med PDT-pluginet inte har något bra sätt att visualisera CSS-kod valdes det att använda Stylizer 5 [32] för detta. I programmet kan man enkelt se vad ändringar i CSS-koden gör med utseendet på webbsidan, vilket effektiviserar utvecklingstiden.

För att testköra all kod kommer XAMPP [21] att användas lokalt medan webbhotellet Host-ed [33] används globalt. Det verktyg som kommer att hantera databasen är phpMyAdmin. Lokalt sker testningen via phpMyAdmin tillsammans med programmet XAMPP och globalt sker det tillsammans med webbhotellet Host-ed.

4.3 Design

4.3.1 Webbssidans layout

Hela webbsidans layout syns i Figur 4.1. Den delades in i tre olika delar: en övre, en undre och en huvuddel.



Figur 4.1 Webbssidans layout för en administratör.

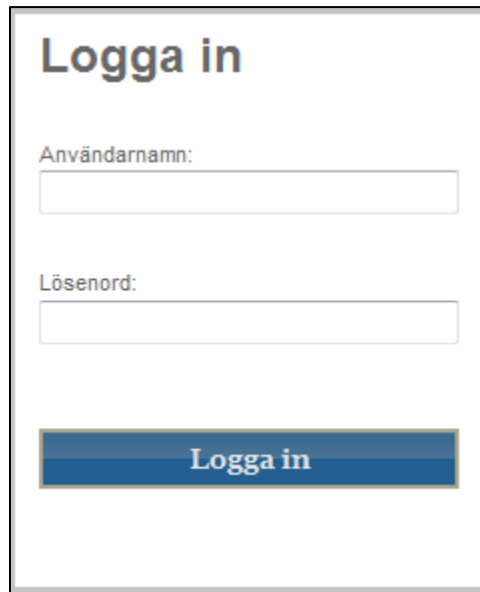
Den övre delen innehåller menyflikarna, loggan, sökfunktionen och utloggningfunktionen. Menyflikarna skapades för att enkelt kunna besöka de mest efterfrågade funktionerna. Placeringen av funktionerna och loggan är gjorda för att göra sidan användarvänlig.

Huvuddelen innehåller de olika huvudfunktionerna t.ex. bokningen. På vissa vyer finns en sidomeny till vänster som ger användaren fler funktioner att välja mellan. På resterande delen visas funktionsinnehållet.

Den nedre delen innehåller företagets slogan samt information om företaget. Detta gör det möjligt att visa extra information om t.ex. bokningssystemet.

4.3.2 Inloggningssidans layout

Designen på inloggningssidan (se Figur 4.2) skiljer sig från webbsidans design. Den gjordes enkel eftersom den endast är till för att säkerhetsställa att ingen utöver företags anställda kan få åtkomst till informationen i bokningssystemet.



The image shows a simple login form with a white background and a thin black border. At the top, the text 'Logga in' is displayed in a large, bold, dark blue font. Below this, there are two input fields. The first is labeled 'Användarnamn:' in a small, grey font, and the second is labeled 'Lösenord:' in the same font. Both fields are empty and have a light grey border. At the bottom of the form, there is a prominent blue button with the text 'Logga in' in white, centered on the button.

Figur 4.2 Funktion för inloggning på webbsidan.

4.3.3 Kodstruktur

Det mesta av HTML-, PHP-, JavaScript- och JQuery-koden placerades i PHP-filer eftersom PHP-kod annars inte kan exekveras.

Från början placerades databasanropen i de olika PHP-filerna. Kod-designen för bokningssystemet förbättrades genom att skapa en klass där alla anrop till databasen sker. Databasklassen inkluderades i varje fil som behöver åtkomst till databasen.

I databasklassen finns en funktion som heter getInstance (se Figur 4.3). Funktionen gör det möjligt för andra PHP-filer att använda en statisk referens till databasen (se Figur 4.4). Med hjälp av denna referens kan filerna utföra funktioner i databasklassen. Funktionerna utför en SQL-förfrågning till databasen som sedan returnerar resultatet till filen som anropade databasklassen (se Figur 4.5).

```
public static function getInstance() {  
    if (!self::$instance instanceof self) {  
        self::$instance = new self;  
    }  
    return self::$instance;  
}
```

Figur 4.3 Funktion för att returnera en referens till databasklassen.

```
require_once("conn.php")
$result = WishDB::getInstance()->get_reservation($_GET['articleNr']);
```

Figur 4.4 Använder en statisk referens av databasklassen för att utföra en SQL-förfrågning.

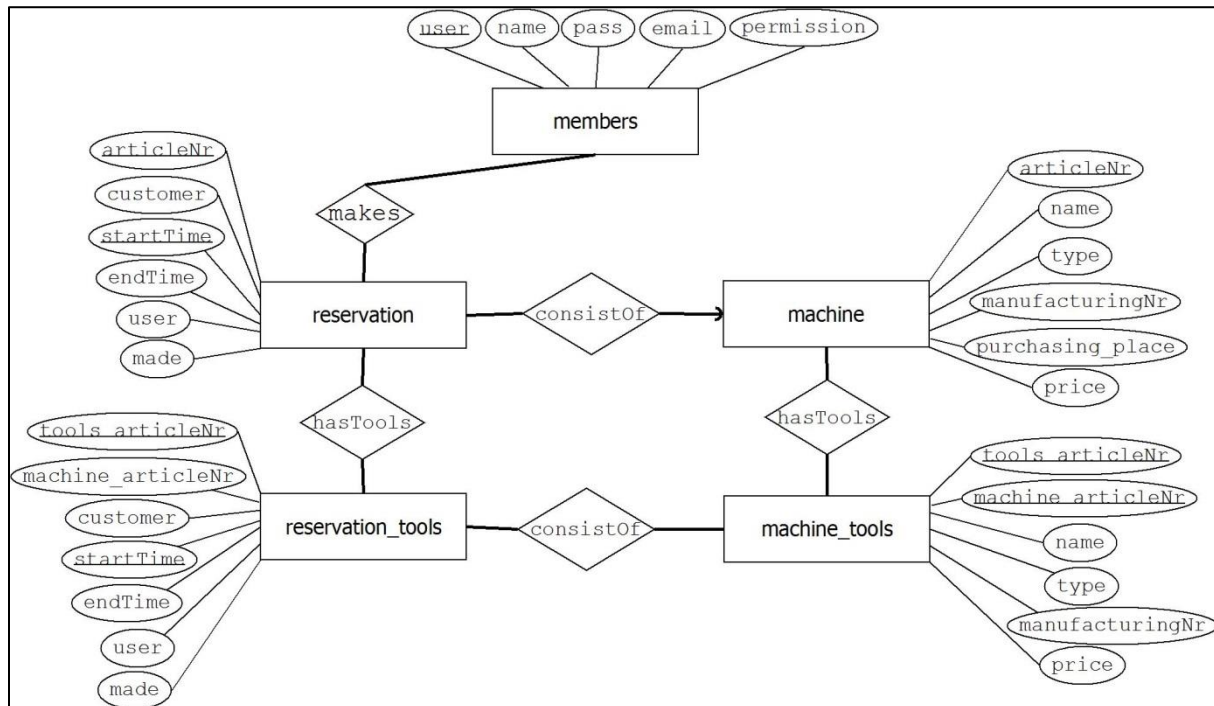
```
public function get_reservation($articleNr){
    if ($stmt = $this->prepare("SELECT * FROM reservation WHERE articleNr= ?")){
        $stmt->bind_param('i', $articleNr);
        $stmt->execute();
        $results = $this->result_to_array($stmt,$stmt->result_metadata());
        $stmt->close();
    }else {
        $results = null;
    }
    return $results;
}
```

Figur 4.5 Funktion för att hämta en bokning.

4.4 Implementation

4.4.1 Databas

Vid implementationen av databasen valdes det att skapa fem olika tabeller. De olika tabellerna är *members*, *reservation*, *reservation_tools*, *machine* och *machine_tools*. I Figur 4.6 visas databasstrukturen samt innehållet i tabellerna.



Figur 4.6 ER-diagram för databasstrukturen.

Tabellen *members* omfattar all information om de registrerade användarna. När en ny användare registreras sparas denna i tabellen *members*. Nyckeln i tabellen är *user* (användarnamnet). Innan registreringen av användaren godkänns, sker en kontroll i PHP-koden att användarnamnet inte finns tidigare.

För att lagra maskinerna, samt se information om dem, skapades tabellen *machine*. Nyckeln i tabellen är *articleNr*. Detta eftersom varje maskin har ett unikt artikelnummer som referens.

Tabellen *machine_tools* innehåller information om de olika tillbehören. I denna tabell är *tools_articleNr* tillsammans med *machine_articleNr* nyckel. Anledningen till att maskinens artikelnummer inte ensamt är nyckel beror på att varje maskin kan ha flera tillbehör koppade till sig samtidigt. Inte heller tillbehörets artikelnummer kan vara nyckel ensamt eftersom tillbehören är universella och kan användas av olika maskiner. Tillsammans utgör de dock en unik nyckel. Det finns en koppling till tabellen *machine* eftersom tillbehören måste kunna referera till maskinerna de tillhör.

Tabellen *reservation* innehåller alla maskiners bokningar. En maskin kan finnas flera gånger i denna tabell om den är bokad mer än en gång. Den kan dock inte vara bokad på samma tid flera gånger. Därför utgör artikelnumret tillsammans med starttiden nyckeln för tabellen. Kopplingen till tabellen *machine* finns eftersom man skall kunna identifiera vilken maskin som bokas och därför sparas maskinens artikelnummer i denna tabell. Det finns också en koppling till tabellen *members* eftersom det skall gå att identifiera vem som utfört bokningen och därför sparas användarnamnet i tabellen.

Det finns separata tabeller för bokning av maskiner respektive tillbehör. Anledningen är att lättare kunna hantera de bokade tillbehören. Tabellen *reservation_tools* innehåller därför information om tillbehörets bokning. Ett tillbehör kan vara bokad flera gånger i denna tabell. Dock kan den inte vara bokad under samma intervall. Därför utgör tillbehörets artikelnummer tillsammans med starttiden nyckeln. Det finns en koppling till tabellen *reservation* för att kunna identifiera vilken maskin tillbehöret är bokad till. Därför behöver maskinens artikelnummer sparas i denna tabell. Det finns också en koppling till tabellen *machine_tools* eftersom det skall gå att identifiera vilket tillbehör som bokas och därför sparas tillbehörets artikelnummer i tabellen.

4.4.2 Säkerhet

För att göra bokningssystemet säkert vidtogs ett antal säkerhetsåtgärder. Det valdes att använda SSL-kryptering samt en kryptografisk hashfunktion. Databasförfrågningarna förbättrades också för att göra säkerheten i systemet bättre.

Eftersom HTTP skickar informationen i klartext från/till webbsidan innebär det att all information (t.ex. användarnamn och lösenord) skickas oskyddat. Genom att avlyssna Internettrafiken kan då känslig data avläsas. Därför är det viktigt att skydda den skickade informationen genom att skapa en krypterad anslutning. Det implementerades en SSL-kryptering som medför att HTTPS-protokollet används istället för HTTP.

Vid registrering av en ny användare skapas ett krypterat lösenord som är baserat på det inskrivna lösenordet. Det krypterade lösenordet genereras med hjälp av en hashfunktion kallad md5. Detta lagras sedan i databasen. När en användare försöker att logga in, krypteras det inskrivna lösenordet och jämförs mot det som finns lagrat i databasen.

För att förhindra SQL-injections används prepared statements med bundna variabler. Prepared statements kontrollerar att användarens inmatade text är av rätt datatyp. Detta säkerställer att inga specialbokstäver finns med i SQL-förfrågningen. Dessa statements implementerades i funktionerna i databasklassen som utför SQL-förfrågningar.

4.4.3 Inloggning

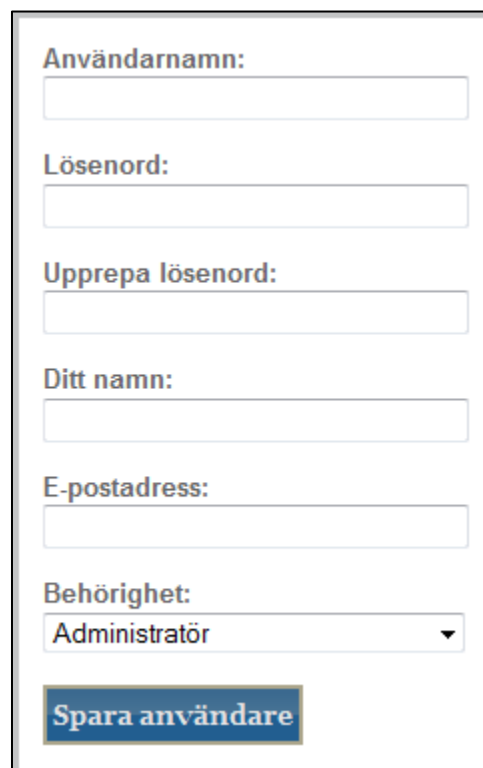
En inloggningsfunktion implementerades för att endast ge berättigade användare tillgång till systemet samt att de kritiska funktionerna begränsas till användarna med administratörsbehörighet.

Användarens inskrivna uppgifter kontrolleras mot de sparade uppgifterna i databasen för att verifiera att inloggningen fungerar korrekt. Detta implementerades med hjälp av HTML- och PHP-kod. Om användarnamnet och lösenordet är korrekt länkas användaren vidare till startsidan, annars skrivs ett felmeddelande ut och användaren blir uppmanad att försöka igen.

4.4.4 Registrera användare

Registreringsfunktionen är till skillnad från många andra applikationer inte placerad vid inloggningsfunktionen, utan istället placerad inuti webbsidan under fliken *Admin* (som endast är tillgänglig för administratörer). Detta eftersom webbsidan endast är till för de anställda och ingen utomstående skall kunna registrera sig.

Vid registrering av en ny användare används fälten i Figur 4.7. Den nya användaren ges en behörighet, som är antingen administratör, uthyrare eller observatör. För att den nya användaren skall sparas korrekt i databastabellen *members* görs ett antal felkontroller på administratörens inmatningar m.h.a. PHP-kod. Fälten och texten skapades m.h.a. HTML-kod.



The image shows a registration form with the following fields and elements:

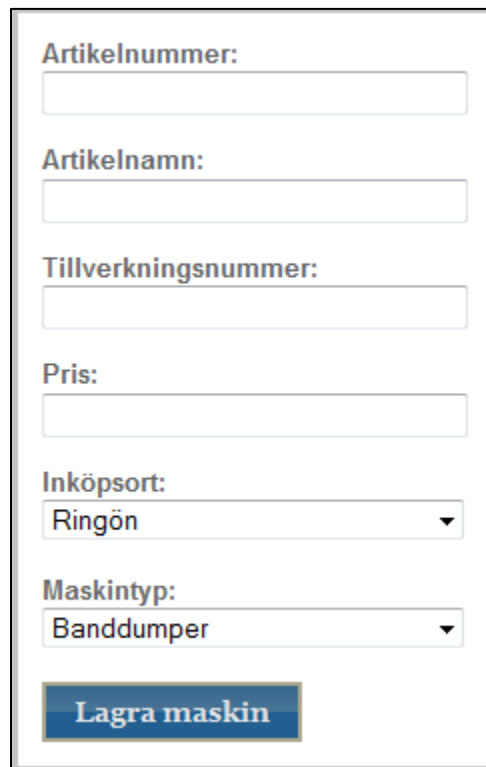
- Användarnamn:** A text input field.
- Lösenord:** A text input field.
- Upprepa lösenord:** A text input field.
- Ditt namn:** A text input field.
- E-postadress:** A text input field.
- Behörighet:** A dropdown menu with "Administratör" selected.
- Spara användare:** A blue button with white text.

Figur 4.7 Funktion för att skapa en ny användare.

4.4.5 Lagra maskin

Funktionen för att lagra en maskin gjordes enligt Figur 4.8, där användaren får mata in information om maskinen. För att lagra data korrekt i databasen implementerades felkontroller på samtliga fält. Funktionen för att lagra maskiner är placerad under *Admin*-fliken.

I fälten för *Inköpsort* och *Maskintyp* får administratören välja mellan ett antal alternativ. Alternativen för *Inköpsort* är: *Stockholm*, *Malmö*, *Ringön* och *Högsbo* och för *Maskintyp*: *Banddumper*, *Kompaktgrävare*, *Hjullastare*, *Vält*, *Skylift* och *Övrig*.



The image shows a web form for adding a machine. It consists of the following elements:

- Artikelnummer:** A text input field.
- Artikelnamn:** A text input field.
- Tillverkningsnummer:** A text input field.
- Pris:** A text input field.
- Inköpsort:** A dropdown menu with "Ringön" selected.
- Maskintyp:** A dropdown menu with "Banddumper" selected.
- Lagra maskin:** A blue button with white text.

Figur 4.8 Funktion för att lagra en maskin i databasen via webbsidan.

4.4.6 Lagra tillbehör

När en maskin hyrs finns det ibland olika tillbehör som man även vill boka. För att kunna boka ett tillbehör måste det vara kopplat till minst en maskin som är lagrad i databasen. Vid lagring av ett nytt tillbehör valdes att implementera en funktion som sköter detta genom att fälten i Figur 4.9 fylls i. Funktionen är placerad under *Admin*-fliken.

Vid lagringen av tillbehöret utförs ett antal felkontroller, framförallt sker en kontroll om det finns en maskin med det angivna artikelnumret. Tillbehörets artikelnummer behöver inte vara unikt eftersom det kan lagras till flera maskiner.



Artikelnnummer maskin:

Artikelnnummer tillbehör:

Namn:

Typ:

Tillverkningsnummer:

Pris:

Lagra tillbehör

Figur 4.9 Funktion för att lagra ett tillbehör i databasen via webbsidan.

4.4.7 Bokning

Huvuduppgiften med examensarbetet är bokningsfunktionen. Den implementerades enligt Figur 4.10. Om maskinens artikelnummer är känt sedan tidigare kan användaren fylla i det direkt. Annars kan det i den vänstra sidomenyn väljas en maskintyp och därefter välja att boka maskinen. Maskinens artikelnummer skrivs då in i fältet automatiskt. På båda sätten får man välja mellan vilka datum maskinen skall bokas samt vilken kund som skall hyra maskinen. Om det finns tillbehör till maskinen kan även dessa bokas. För att visualisera en maskins tillgänglighet samt underlätta att datumet skrivs in på rätt sätt samt implementerades en kalender m.h.a. JQuery.

Det första som syns i kalendern är dagens datum ifyllt och de tidigare datumen bortmarkerade (se Figur 4.10, där dagens datum var 12 juni 2014). Det finns alltså ingen möjlighet att av misstag boka en maskin tidigare än dagens datum.

The screenshot shows a web application interface for booking a machine. On the left is a sidebar menu with the following items: Maskiner, Banddumper, Kompaktgrävare, Hjullastare, Vält, Skylift, and Övrigt. The main content area is titled "Boka maskin" and contains a form with the following elements: "Artikelnummer:" with an input field, "Kund:" with an input field, a calendar for June 2014 with the 12th highlighted, "Starttid:" with an input field, "Sluttid:" with an input field, and a blue button labeled "Fortsätt bokning".

Figur 4.10 Funktion för att boka en maskin.

Om en maskin är bokad under ett eller flera tillfällen kommer dessa att bli bortmarkerade i kalendern när man försöker göra sin bokning. I Figur 4.11 ges ett exempel, där en maskin är bokad 5-15 juli samt 20-25 juli.

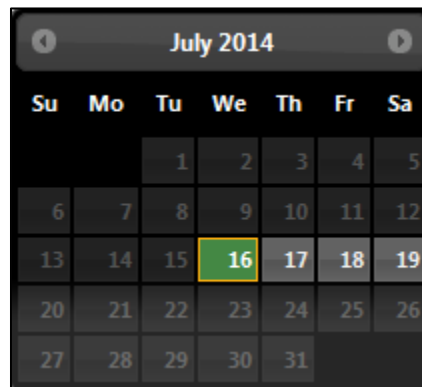


A calendar for July 2014 with days 1 through 31. Days 1-5, 16-19, and 20-25 are shaded grey, indicating they are booked. Days 6-15, 26-31 are white, indicating they are available.

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Figur 4.11 En kalender med lediga dagar för en maskin.

Om man väljer ett startdatum som finns tillgängligt, t.ex. 16 juli, markeras detta och det visas hur lång tid maskinen som längst kan bokas (se Figur 4.12).



A calendar for July 2014 with days 1 through 31. Day 16 is highlighted with a green border, indicating it is the selected start date. Days 1-5, 16-19, and 20-25 are shaded grey, indicating they are booked. Days 6-15, 26-31 are white, indicating they are available.

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Figur 4.12 En kalender med ett startdatum.

Väljer man då att boka maskinen till den 18 juli kommer 16-18 juli att markeras och fälten för start- och slutdatum fylls i automatiskt (se Figur 4.13). Eftersom artikelnumret nu är ifyllt visas de olika tillbehören som kan bokas till maskinen. För att välja ett eller flera tillbehör kryssar man i checkboxarna (se Figur 4.13). Om tillbehöret är bokat under den valda perioden visas ett felmeddelande.

Artikelnummer:

Kund:

July 2014

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Starttid:

Sluttid:

Tillbehör:

Tillbehörets Artikelnummer: 123
 Tillbehörets Artikelnummer: 454

Fortsätt bokning

Figur 4.13 En kalender med start- och slutdatum valda.

När användaren fyllt i fälten korrekt och klickat på *Fortsätt bokning* visas en sida där användaren ser information om den pågående bokningen, information om maskinen samt eventuella tillbehör som håller på att bokas (se Figur 4.14). Trycker man på *Spara bokning* lagras informationen om maskinens bokning i databastabellen *reservation* och tillbehörens bokning i databastabellen *reservation_tools*.

Om bokningen:

Kund:	Test AB
Starttid:	2014-07-16
Sluttid:	2014-07-18

Maskin:

Artikelnummer:	3
Maskin namn:	Kompaktgrävare 1.2 ton
Maskintyp:	kompaktgrävare
Tillverkningsnummer:	325211555
Inköpsplats:	Stockholm
Pris:	3000

Tillbehör:

Artikelnummer:	123
Tillbehörets namn:	Skopa
Maskintyp:	Kompaktgrävare
Tillverkningsnummer:	123456
Pris:	99

[Spara bokning](#)

Figur 4.14 Information om den pågående bokningen, maskinen och tillbehöret.

Efter lagringen har endast administratören möjlighet att redigera bokningen. Detta görs via *Redigera bokning* under *Admin*-fliken. Där visas alla kommande bokningar i systemet samt information om dem. För att redigera eller radera en bokning används knapparna i Figur 4.15. Vid redigering av en bokning länkas man till bokningssidan. Alla uppgifter om bokningen ifyllda och man har möjlighet att redigera allt förutom artikelnumret. Efter att den bokningen har redigerats, ändras bokningen i databasen.

ArtikelNr	Kund	Starttid	Sluttid	Användare	Gjordes	Tillbehör	
3	Test AB	18-06-2014	20-06-2014	Jimmy Svensson	12-06-2014	123	Redigera Radera
5	Test AB	01-07-2014	16-07-2014	Stefan Andersson	12-06-2014	454	Redigera Radera
40	Test AB	23-06-2014	03-07-2014	Gustav Karlsson	12-06-2014	Saknas	Redigera Radera

Figur 4.15. Information om alla kommande bokningar.

4.4.8 Mina bokningar

Det implementerades en flik på webbsidan som listar uppgifter om alla bokningar användaren har gjort. Bokningsuppgifterna hämtas från databastablerna *reservation* och *reservation_tools*. Hämtningen sker m.h.a. PHP-kod där användarens användarnamn jämförs med användarnamnen i databastablerna.

Det krävs administratörsbehörighet för att kunna redigera eller radera bokningar. Om detta är uppfyllt visas information om bokningen och även knapparna *Redigera* och *Radera* (se Figur 4.16). Knapparnas funktionalitet fungerar på samma sätt som i *Redigera bokning*.

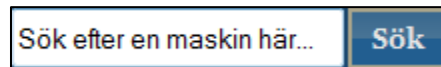
Mina bokade maskiner						
Artikelnummer	Kund	Starttid	Sluttid	Bokningen gjordes	Tillbehörets artikelnummer	
3	Test AB	2014-06-18	2014-06-20	2014-06-12 13:48:15	123	Redigera Radera
5	Test AB	2014-08-04	2014-08-20	2014-06-12 13:53:35		Redigera Radera

Figur 4.16 Funktion för se samt redigera/radera bokningar.

4.4.9 Sökfunktion

Det implementerades en sökfunktion för att underlätta för användaren att hitta en maskin. I sökfunktionen finns det möjlighet att söka på maskinens artikelnummer, artikelnamn, tillverkningsnummer, inköpsort, pris eller maskintyp (se Figur 4.17). Vid sökning av en specifik maskin måste sökningen vara exakt. Det går dock att söka efter ett intervall av artikelnummer t.ex. 100-200. Det går också att söka efter maskiner som innehåller en viss sträng. Detta görs genom att skriva ett procenttecken följt av strängen t.ex. %grävare. Då visas alla maskiner vars artikelnamn innehåller strängen. Anledningen till att sökningen görs m.h.a. ett procenttecken är att företaget använder detta i andra verktyg.

Om en sökning är lyckad (dvs. sökningen hittar minst en maskin) listas information om maskinerna och det ges möjlighet att boka dem samt verifiera deras tillgänglighet.

A search input field with a light blue border and a blue button to its right. The input field contains the text "Sök efter en maskin här..." and the button contains the text "Sök".

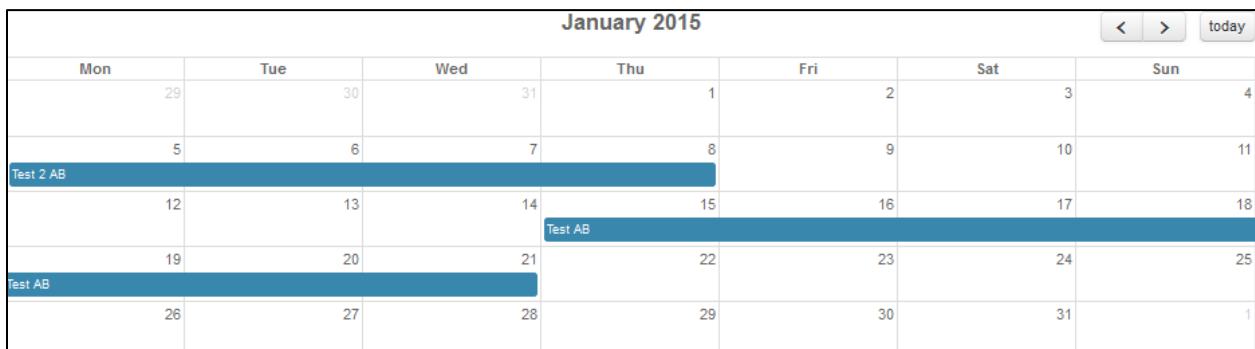
Figur 4.17 Funktion för att söka efter en maskin.

4.4.10 Maskinöversikt

För att enkelt kunna se hur mycket maskinerna varit uthyrda under ett år implementerades ett antal kalendrar som visar detta. De kalendrar som finns representerar enskilda maskiners bokningar samt en som visar alla maskiners bokningar. Dessa implementerades m.h.a. JQuery-kod.

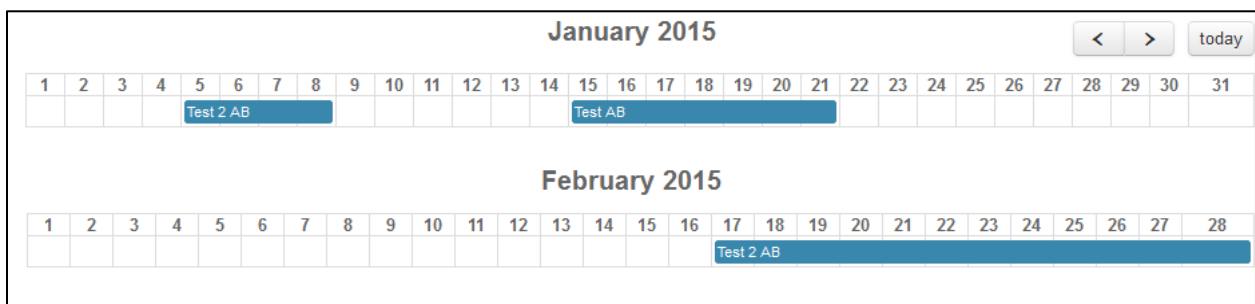
De enskilda kalendrarna visar en månads- respektive årsöversikt. De nås genom att välja en maskintyp i sidomenyn under *Boka Maskin*. De är också tillgängliga via sökfunktionen. I kalendrarna går det att se historik om tidigare och kommande år genom att använda pilarna i övre högra hörnet på kalenderbilderna.

Månadskalendern visar varje vecka på en rad under en månad (se Figur 4.18). Årskalendern visar varje månad på en rad med siffran på dagen ovanför och månadsnamnet överst. I Figur 4.19 visas de två första månaderna på årskalendern, samtliga månader syns i verkligheten men visas inte i figuren.



January 2015							<	>	today
Mon	Tue	Wed	Thu	Fri	Sat	Sun			
29	30	31	1	2	3	4			
5	6	7	8	9	10	11			
Test 2 AB									
12	13	14	15	16	17	18			
			Test AB						
19	20	21	22	23	24	25			
Test AB									
26	27	28	29	30	31	1			

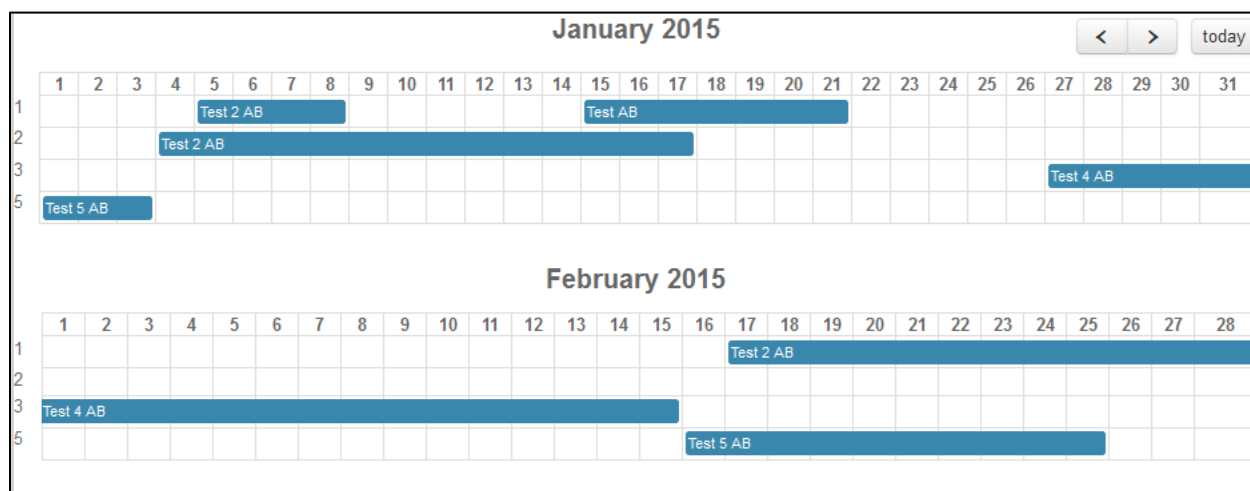
Figur 4.18 Funktion för att se när en maskin är bokad under en månad.



January 2015																															<	>	today																										
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																													
				Test 2 AB										Test AB																																													
February 2015																																																											
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28																																
																Test 2 AB																																											

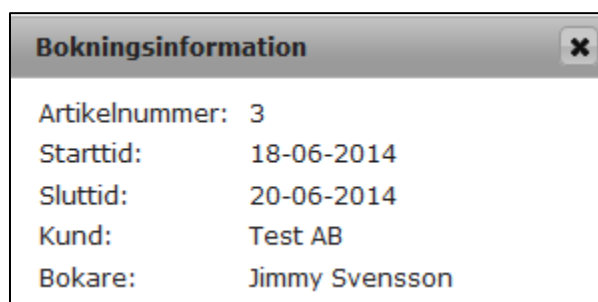
Figur 4.19 Funktion för att se när en maskin är bokad under ett år.

Slutligen finns det en kalender som ger en översikt över alla maskiners bokningar under ett år. Kalendern är placerad under *Översikt*-fliken. Den innehåller bokningarna månadsvis och den listar alla artikelnummer på maskinerna till vänster om kalendern. I Figur 4.20 visas de två första månaderna på året, samtliga månader syns i verkligheten men visas inte i figuren.



Figur 4.20 Funktion för att visa alla maskiners bokningar under ett år.

I kalendrarna är det möjligt att se mer information om bokningarna genom att klicka på den bokning man är intresserad av. Då visas en dialogruta som listar artikelnummer, kund, start- och sluttid samt vem som gjort bokningen (se Figur 4.21). Dialogrutan skapades m.h.a. JQuery-kod.



Figur 4.21 Dialogruta som visar information om bokningen.

4.4.11 Inställningar

Det implementerades en flik för användarinställningar på webbsidan, detta gör att användaren har möjlighet att redigera sina uppgifter (se Figur 4.22). Under fliken listas information om användaren. Informationen hämtas m.h.a. PHP-kod från databastabellen *memebers*. De användaruppgifter som går att redigera är namn, e-postadress och lösenord. För att byta lösenord krävs att det nya lösenordet upprepas korrekt samt att det gamla lösenordet skrivs in korrekt. På samtliga fält utförs det felkontroller. Om det inte hittas några fel i fälten ersätts de gamla uppgifterna med de nya i databasen, annars visas felmeddelanden.



The image shows a web form for editing user settings. It contains the following fields and labels:

- Användarnamn:** A text input field containing the value "user".
- Ditt namn:** A text input field containing the value "Jimmy Svensson".
- E-postadress:** A text input field containing the value "Jimmy@test.se".
- Nytt lösenord:** An empty text input field.
- Upprepa lösenord:** An empty text input field.
- Gammalt lösenord:** An empty text input field.
- Spara uppgifter:** A blue button with white text.

Figur 4.22 Funktion för att ändra sina användaruppgifter.

4.4.12 Raderafunktioner

Det finns tre olika raderafunktioner: radera användare, radera maskiner samt radera tillbehör. I samtliga raderafunktioner finns en separat sökfunktion implementerad. För att använda raderafunktionerna krävs en administratörsbehörighet.

För att radera en användare söker man efter användarens namn (se Figur 4.23). Om det vid sökningen inte finns någon med det angivna namnet i databasen skrivs ett felmeddelande ut. Annars visas information om varje användare tillsammans med en raderaknapp (se Figur 4.24). Efter att klickat på raderaknappen på den användare man vill ta bort, raderas denna ur databasen.



The screenshot shows a search form with the following elements:

- Label: **Namn på personen:**
- Text input field containing: **Jimmy Svensson**
- Submit button: **Sök**

Figur 4.23 Funktion för att söka användare.



The screenshot displays two user profiles, each with a 'Radera' (Delete) button below the information:

User 1:

- Användarnamn: user
- Namn: Jimmy Svensson
- E-postadress: Jimmy@test.se
- Behörighet: Administratör

User 2:

- Användarnamn: Jimmysvensson
- Namn: Jimmy Svensson
- E-postadress: jimmysv@test.se
- Behörighet: Uthyrare

Figur 4.24 Funktion för att visa information om de sökta användarna.

Radering av en maskin görs genom att söka på maskinens artikelnummer. Det visas då information om maskinen samt en raderaknapp. Väljer administratören att radera maskinen tas den bort från databasen och om det finns bokningar på maskinen efter dagens datum eller kopplingar till tillbehör, raderas dessa.

Radera ett tillbehör kan utföras efter att en sökning av tillbehörets artikelnummer gjorts. Därefter visas information om tillbehöret samt vilka maskiner det är kopplat till. Vid radering av tillbehöret försvinner kopplingarna till maskinerna samt de bokningar som finns på tillbehöret framöver.

4.5 Testning

Under projektet har det utförts tester för att verifiera bokningssystemets innehåll samt struktur. Det gjordes egna tester och användartester.

De egna testerna gjordes för att verifiera designens layout och att funktionaliteten fungerade som planerat. Detta utfördes lokalt och globalt m.h.a. verktyget XAMPP respektive webbhotellet Host-ed. Testerna utfördes kontinuerligt efter att en funktion implementerats eller utvecklats.

Användartesterna utfördes fortlöpande under utvecklingen av bokningssystemet. De verifierade systemets användarvänlighet, funktionalitet samt design. Personerna som testade systemet var företrädesvis anställda på företaget.

Vid testerna på de icke-anställda framkom utvecklingsmöjligheter samt korrigeringar som kunde förbättra systemet. Förslagen diskuterades därefter med företaget och redigerades samt implementerades oftast. Detta har medfört att systemet utvecklats efter användares synpunkter.

Testerna på de anställda gjordes för att anpassa bokningssystemet efter användarna. Det har även här uppkommit funderingar samt förslag på utvecklingar samt ny funktionalitet som gjort systemet påminnande om andra verktyg företaget använder. Detta har medfört att systemet blivit användarvänligt och att det utvecklats efter de anställdas åsikter.

5 Resultat

Arbetet har utvärderat vilka språk som kan användas för att utveckla ett webbaserat bokningssystem. Det framkom att HTML, JavaScript, JQuery samt CSS var nödvändiga för att skapa den funktionalitet och design som systemet behövde. För att bestämma databas samt kommunikationsspråk gjordes en förstudie. I den framkom att PHP tillsammans med MySQL samt ASP.NET tillsammans med MSSQL var de mest lämpade. PHP/MySQL valdes eftersom det redan fanns viss erfarenhet av PHP/MySQL samt att inläringstiden är kortare med PHP.

Baserat på språken gjordes en analys av vilka verktyg som kunde användas. Den resulterade i att Eclipse med PDT-plugin och Dreamweaver jämfördes. Valet blev Eclipse med PDT-plugin eftersom det är kostnadsfritt samt att det är mer fokuserat på PHP-kod. PDT-pluginen har även stöd för HTML-, JavaScript- och JQuery-kod. Det saknar dock stöd för visualisering av CSS-kod, därför användes verktyget Stylizer 5 för detta.

5.1 Resultat av produkt

Bokningssystemet består av PHP-, HTML-, JQuery-, JavaScript- samt CSS-kod. PHP används för att kommunicera mellan webbsidan och databasen. För att göra kommunikationen säker används SSL-kryptering samt skydd mot SQL-injections. En kryptografisk hashfunktion används för att kryptera lösenord innan de skickas till databasen.

Eftersom bokningssystemet endast skall vara tillgängligt för behöriga användare, finns en inloggningsfunktion. Vid inloggningen har användaren en behörighet som används för att begränsa tillgången till de kritiska funktionerna.

Huvudfunktionerna är att kunna boka samt visualisera maskinernas tillgänglighet. Funktionerna använder framförallt HTML och JQuery. HTML visar text samt inmatningsfält och JQuery används för att visualisera maskiners bokningar. Visualiseringen görs med kalendrar för att få en översikt över bokningarna. Bokningsfunktionen (se Figur 5.1) är gjord så att användaren enkelt kan ange rätt information samt boka eventuella tillbehör. Efter att en bokning lagrats kan den redigeras eller raderas. För att kunna boka maskiner och tillbehör måste de vara lagrade i databasen. Därför finns funktioner för att lagra maskiner respektive tillbehör. Radering av maskiner/tillbehör är också möjligt.

WANGESKOG
WHC
HYRCENTER AB

Jimmy Svensson
[Logga ut](#)

Hem Boka Maskin Mina bokningar Admin Inställningar Översikt Sök efter en maskin här... Sök

Boka maskin

Artikelnummer:

Kund:

June 2014

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Starttid:

Sluttid:

[Fortsätt bokning](#)

Maskiner

Banddumper

Kompaktgrävare

Hjullastare

Vält

Skylift

Övrigt

Figur 5.1 Webbsidans layout för en administratör vid bokning.

6 Slutsats

6.1 Resumé

Från början av arbetet fanns funderingar om webbsidans design och funktionalitet. Det fanns förhoppningar om att projektet skulle utöva våra tidigare programmeringskunskaper samt utveckla nya kunskaper inom området.

Den slutgiltiga designen på webbsidan blev över förväntan och det implementerades mer funktionalitet än vad som från början var planerat. Detta eftersom vissa funktioner implementerades snabbare än vad som uppskattades i tidsplaneringen. Den nya funktionaliteten har inneburit att det har blivit en bättre produkt för användarna.

Under projektet har vi fått användning av våra kunskaper från utbildning samt nya kunskaper har utvecklats.

6.2 Kritisk diskussion

6.2.1 Webbsidans funktionalitet

Det går att diskutera hur generella funktionerna är och hur många av dem som kan användas i andra bokningssystem.

I bokningssystemet finns det återanvändbar kod som kan användas i andra system. Ett exempel är kalendern som underlättar för användaren att välja start- och slutdatum. Den hjälper även till genom att endast visa de datum som är tillgängliga.

En annan funktion som kan användas i andra system är årsöversiktskalendern. Den gör det möjligt att presentera samtliga bokningar under ett år. På detta vis kan en överblick av en stor mängd data visas på endast en sida.

Användarbehörigheterna som implementerades i systemet kan också återanvändas. Detta för att endast de med hög tillit skall ha tillgång till de kritiska funktionerna.

Vid de flesta funktionerna på webbsidan används dynamisk uppdatering av data, vilket innebär att data hämtas på nytt från databasen vid varje uppdatering. Detta medför att ändringar av sparad data hämtas på nytt när sidan laddas om. Alternativet hade varit att inte använda en databas. Detta skulle medfört svårigheter i funktioner som t.ex. inloggningen eftersom det inte finns något annat bra sätt att spara användarnas uppgifter. Det hade däremot gått snabbare att ladda in olika sidor. Kraven på funktionerna gjorde dock att det hade blivit svårt att utelägna databasen.

Den funktionalitet som kan förbättras i systemet är bokningen av tillbehören, eftersom det endast skrivs ut ett felmeddelande. Det hade varit bättre att visualisera tillbehörets tillgänglighet på lämpligt vis. Tidsramen för projektet gjorde dock att detta inte implementerades.

6.2.2 Utvärdering av verktyg

För att göra bokningssystemet användarvänligt användes CSS-kod och programmet Stylizer 5 [32], vilket vi rekommenderar varmt. Programmet har många bra funktioner för att redigera och flytta om CSS-kod, vilket effektiviserade utvecklingsprocessen. Designen på en webbsida kan alltid utvecklas men det framkom genom användartester att webbsidan var användarvänlig och att funktionaliteten var lättåtkomlig.

Eclipse har använts för att programmera HTML, PHP, JavaScript och JQuery. I stora delar av projektet har HTML- och PHP-kod använts, vilket Eclipse tillsammans med PDT-pluginen har felkontroller på samt funktionsförslag. Det som saknats i projektet har varit förslag på JavaScript- och JQuery-funktioner samt kompileringskontroller för dessa. För att ersätta kompileringskontrollerna utfördes fler testkörningar på JavaScript- och JQuery-koden, vilket var tidskrävande. Funktionerna behövde dessutom sökas upp separat. Eclipse har dock stöd för det som användes mest i projektet vilket får anses som bra.

Det har fungerat väldigt bra att testköra programmeringsfilerna lokalt med hjälp av XAMPP. Den kompilerar PHP-kod och visar en logg om det finns fel i koden. Det som saknades var att lista JavaScript- och JQuery-fel, men det är inget som verktyget är konstruerat för. Det har också stöd för SSL vilket användes i projektet. Eftersom XAMPP har stöd för det som huvudsakligen utfördes i projektet var det ett väldigt användbart verktyg.

För att skapa databastabellerna har phpMyAdmin använts. Verktyget användes ofta och fungerade utmärkt under hela projektet. Det har stöd för att exportera och importera databastabeller. Detta har varit användbart för att säkerhetskopiera databasen och enkelt kunna testköra olika databasinnehåll.

6.2.3 Kodstruktur

Det skapades en egen PHP-fil för varje vy på webbsidan. I början av projektet hade varje fil en eller flera anrop som utfördes direkt till databasen. Detta gjorde programmeringsstrukturen otydlig. För att förbättra detta skapades en egen klass där alla databasanrop sker. Strukturen blev mycket tydligare och vid redigering i databasen behöver bara en klass ändras.

Vid utvecklingen av webbsidan har det funnits en avsikt att den skall vara utbyggbar. Det skall vara enkelt att implementera nya vyer, funktioner, flikar m.m. Detta medför bra förutsättningar till vidareutveckling av bokningssystemet vid behov.

6.2.4 Säkerhetsaspekter

I början av projektet ansågs det att hashfunktionen md5 var tillräcklig som kryptering. Det framkom vid senare efterforskningar att det fanns fler säkerhetsåtgärder att vidta för att göra bokningssystemet säkrare. Säkerheten för databasanropen förbättrades genom att använda prepared statements i databasklassen. Detta innebär att det blir svårare att göra intrång på data som finns lagrad i databasen.

För att öka webbsidans kommunikationssäkerhet implementerades SSL-kryptering. Detta gör att all kommunikation från och till webbsidan krypteras och skickas på en anslutning som endast är känd av de parter som kommunicerar.

Det finns många sätt att försöka undvika intrång. I bokningssystem finns det skydd mot de vanligaste attackerna eftersom intrång kan ge stora konsekvenser. Det kan finnas ovanliga intrång som systemet inte är skyddat mot. Men den implementerade säkerheten anses som kraftfull eftersom de flesta intrången förhindrats.

Det som kan förbättra säkerheten ytterligare i systemet är att ersätta md5 som kryptering av lösenorden. Det kom fram i slutet av projektet att säkrare hashfunktioner finns tillgängliga som dessutom krypterar lika snabbt som md5.

6.2.5 Tidsplan och verklighet

Då examensarbetarna tidigare inte skapat något i storlek med denna webbsida och att den dessutom skulle vara kopplad till en databas, var det svårt att göra en exakt tidsplanering (se Appendix). Därför blev tidsplaneringen generellt längre tidsmässigt än vad det slutligen tog att genomföra.

De största avvikelserna från tidsplaneringen var konfigurationen av en databas. Detta behövdes inte eftersom det valdes att använda ett webbhotell där databasen redan var konfigurerad. Utvecklingen av databastabellerna tog mindre tid att skapa jämfört med vad som var planerat. Detta eftersom verktyget phpMyAdmin gjorde det mycket lättare att implementera tabellerna i databasen.

6.2.6 Metod och verklighet

En enklare version av Scrum följdes genom hela projektet. I början skrevs en product backlog med alla arbetsuppgifter. Inför varje vecka gjordes en ny sprint backlog med de nya uppgifterna som skulle utföras under veckan. Under projektets gång utökades product backlogen då nya arbetsuppgifter tillkom. När projektet avslutades hade samtliga uppgifter slutförts.

Scrum medförde att projektets arbetsuppgifter blev överskådliga samt att arbetets tillvägagångssätt blev strukturerat.

Förstudien för att identifiera vilka verktyg som skulle användas, blev litteratur- och webbsökningar. Det ansågs vara tillräckligt för att få information om de olika verktygen som kunde användas. Mellan dessa gjordes sedan en analys för att bestämma vilka verktyg som var mest lämpat för projektet.

Det första som skapades i projektet var en webbsida. Denna kopplades till en färdigkonfigurerad databas som fanns på webbhotellet. När sammankopplingen mellan webbsidan och databasen fungerade, arbetades det parallellt på dem.

6.3 Teknikens roll i samhället

Med hjälp av bokningssystemet kan företaget förhandsboka maskinerna. Eftersom man i god tid vet när maskinerna behöver transporteras kan man strukturera och planera transportererna mellan de olika uthyrningsplatserna. Detta innebär att man kommer att boka upp redan befintliga transportföretag för att transportera maskinerna. Detta medför att miljöutsläppen minskar eftersom man då undviker att skicka en separat transport för endast en maskin.

6.4 Vidareutveckling

Vid implementering av ett system finns det alltid möjlighet att vidareutveckla det på ett eller flera sätt. För vårt bokningssystem finns det många möjligheter till fortsatt utveckling.

För att få bokningen av en maskin mer lättöverskådlig kan bilder av de olika maskinerna läggas till. Vid bokningen kan en kostnadsberäkning för den aktuella bokningsperioden beräknas. Det kan också implementeras en funktion där man har möjlighet att se de senaste bokningarna som gjorts i systemet.

Bokningssystemet kan utvecklas så att det inte bara är till för företagets anställda, en möjlighet är att låta kunden boka maskinerna i systemet. Om det då implementerats en kostnadsberäkning kan också en betalningsfunktion utvecklas. En flik med information om företaget är lämpligt om kunden bokar maskinen. Man kan även implementera en funktion för nyheter där företaget kan meddela kunderna viktig information.

Statistik om t.ex. maskinernas bokningar samt vilka kunder som hyr mest maskiner under ett år kan implementeras. Detta för att få en överblick om vilka maskiner som är mest använda samt vilka kunder som hyr mest maskiner.

Referenser

- [1] Scrum, "Create software that delivers more value," Scrum.org, 2014. [Online]. Available: <https://www.scrum.org>. [Använd 11 april 2014].
- [2] Oracle, "What is MySQL?," Oracle Corporation, 2014. [Online]. Available: <http://dev.mysql.com/doc/refman/4.1/en/what-is-mysql.html>. [Använd 29 04 2014].
- [3] Host Shopper , "What is MS SQL?," Host Shopper , 2008. [Online]. Available: <http://www.host-shopper.com/what-is-ms-sql.html>. [Använd 14 05 2014].
- [4] The PHP Group, "General Information," The PHP Group, 2014. [Online]. Available: <http://www.php.net/manual/en/faq.general.php>. [Använd 14 april 2014].
- [5] CodingUnit Programming Tutorials, "PHP Tutorial – Language Introduction," CodingUnit Programming Tutorials, 2014. [Online]. Available: <http://www.codingunit.com/php-tutorial-language-introduction>. [Använd 20 05 2014].
- [6] A. Bradley, "What is PHP Used For?," About.com, 2014. [Online]. Available: http://php.about.com/od/phpbasics/qt/what_is_php_used_for.htm. [Använd 14 05 2014].
- [7] Acunetix, "SQL Injection: What is it?," Acunetix, 2014. [Online]. Available: <https://www.acunetix.com/websitesecurity/sql-injection/>. [Använd 17 april 2014].
- [8] P. MacIntyre, PHP: The Good Parts, US: O'Reilly Media, 2010.
- [9] The PHP Group, "SQL Injection," The PHP Group, 2014. [Online]. Available: <http://www.php.net/manual/en/security.database.sql-injection.php#security.database.sql-injection> . [Använd 17 april 2014].
- [10] L. Ganapathy, "How to Prevent SQL Injection Attack (Explained with an Example)," Ramesh Natarajan, 2012. [Online]. Available: <http://www.thegeekstuff.com/2012/02/sql-injection-attacks/>. [Använd 17 april 2014].
- [11] D. Bolton, "What is ASP.NET?," About.com, 2014. [Online]. Available: <http://cplus.about.com/od/introductiontoprogramming/a/asp.htm>. [Använd 14 05 2014].
- [12] Microsoft, "ASP.NET Overview," Microsoft, 2014. [Online]. Available: <http://msdn.microsoft.com/en-us/library/vstudio/4w3ex9c2%28v=vs.100%29.aspx>. [Använd 20 05 2014].

- [13] J. R. G. Paz, Beginning ASP.NET MVC 4, Berkeley, CA: Apress, 2013.
- [14] Eclipse Kepler , "What is Eclipse?," Eclipse Kepler , 2014. [Online]. Available: http://help.eclipse.org/kepler/index.jsp?topic=%2Forg.eclipse.platform.doc.isv%2Fguide%2Fint_eclipse.htm. [Använd 08 05 2014].
- [15] Zend, "PHP Development Tools," Zend Technologies Ltd, 2013. [Online]. Available: <http://www.zend.com/en/company/community/pdt/>. [Använd 01 april 2014].
- [16] J. Kyrnin, "Adobe Dreamweaver," About.com, 2014. [Online]. Available: <http://webdesign.about.com/od/dreamweaver/fr/dreamweaver-profile.htm>. [Använd 14 05 2014].
- [17] W3Schools, "JavaScript Introduction," W3Schools, 2014. [Online]. Available: http://www.w3schools.com/js/js_intro.asp. [Använd 03 april 2014].
- [18] The jQuery Foundation, "Category Archives: Foundation," The jQuery Foundation, 2014. [Online]. Available: <http://blog.jquery.com/category/foundation/>. [Använd 03 april 2014].
- [19] STANDS4, "What does XAMPP stand for?," STANDS4 LLC, 2014. [Online]. Available: <http://www.abbreviations.com/XAMPP>. [Använd 03 april 2014].
- [20] portableapps, "XAMPP," portableapps, 2014. [Online]. Available: <http://portableapps.com/apps/development/xampp>. [Använd 20 maj 2014].
- [21] Apache Friends, "XAMPP Apache + MySQL + PHP + Perl," Apache Friends, 2014. [Online]. Available: <https://www.apachefriends.org/index.html>. [Använd 01 april 2014].
- [22] Techterms.com, "WAMP," Techterms.com, 2013. [Online]. Available: <http://www.techterms.com/definition/wamp>. [Använd 14 05 2014].
- [23] DigiCert, "What Is SSL (Secure Sockets Layer) and What Are SSL Certificates?," DigiCert, 2014. [Online]. Available: <http://www.digicert.com/ssl.htm>. [Använd 15 05 2014].
- [24] GlobalSign, "What is SSL?," GlobalSign, 2014. [Online]. Available: <https://www.globalsign.eu/ssl-information-center/what-is-ssl.html>. [Använd 14 05 2014].
- [25] SourceForge, "Bringing MySQL to the web," SourceForge.net, 2014. [Online]. Available: www.phpmyadmin.net/. [Använd 11 april 2014].
- [26] The PHP Group, "Requirements," The PHP Group, 2014. [Online]. Available: <http://www.php.net/manual/en/mysql.requirements.php>. [Använd 01 april 2014].

- [27] Microsoft, "Use SQL Server Management Studio," Microsoft, 2014. [Online]. Available: <http://msdn.microsoft.com/en-us/library/ms174173.aspx>. [Använd 14 april 2014].
- [28] Microsoft, "Object Explorer," Microsoft, 2014. [Online]. Available: <http://msdn.microsoft.com/en-us/library/ms173849.aspx>. [Använd 14 april 2014].
- [29] Comentum, "PHP vs ASP.net Comparison," Comentum, 2010. [Online]. Available: <http://www.comentum.com/php-vs-asp.net-comparison.html>. [Använd 11 april 2014].
- [30] Eclipse, "Eclipse," The Eclipse Foundation, 2014. [Online]. Available: <https://www.eclipse.org/>. [Använd 01 april 2014].
- [31] The Eclipse Foundation, "PHP Development Tools," The Eclipse Foundation, 2014. [Online]. Available: <http://projects.eclipse.org/projects/tools.pdt>. [Använd 01 april 2014].
- [32] skybound, "STYLIZER IS REAL-TIME CSS," skybound.ca, 2014. [Online]. Available: <http://www.skybound.ca/>. [Använd 11 april 2014].
- [33] Host-ed, "QUALITY WEB HOSTING," Host-ed.NET , 2014. [Online]. Available: <http://host-ed.net/>. [Använd 11 april 2014].

A Appendix

Moment	Vecka 1	Vecka 2	Vecka 3	Vecka 4	Vecka 5	Vecka 6	Vecka 7	Vecka 8	Vecka 9	Vecka 10
	24 mars - 28 mars	31 mars - 04 april	7 april - 11 april	14 april - 18 april	21 april - 25 april	28 april - 02 maj	05 maj - 09 maj	12 maj - 16 maj	19 maj - 23 maj	26 maj - 30 maj
	M T O T F	M T O T F	M T O T F	M T O T F	M T O T F	M T O T F	M T O T F	M T O T F	M T O T F	M T O T F
Research	█									
Webbsida										
Visuell uppbyggnad		█								
Grundlig struktur			█							
Koppla samman databasen				█						
Inloggningsfunktion				█	█					
Möjlighet att boka maskiner						█	█			
Se bokade maskiner								█	█	
Användarbehörigheter									█	█
Förfining av Layout										█
Databas										
Konfiguration av databasen		█	█							
Stöd för inloggning/behörighet			█	█						
Lagring av maskin information					█	█				
Lagring av bokningsuppgifter					█	█				
Testning & Bugg fix										█
Rapportskrivning		█	█	█	█	█	█	█	█	█

Tabell 1.1. Tidsplanen för examensarbetet