

Contents

1	Avstämning mot kravspecifikation	1
1.1	Medlemmar och organisation	1
1.2	Ekonomi	1
1.3	Kommunikation	2
1.4	Externa aktörer	2
1.5	Slutligen	2
2	Om det här dokumentet	2
3	Filsystemet	3
3.1	/	3
3.2	bin	3
3.3	build	3
3.4	cli	3
3.5	db	4
3.6	docs	4
3.7	libs	4
3.8	src	4
3.9	tests	4
3.10	var	5
3.11	www	5
4	Serverkonfiguration	5
4.1	Grundläggande krav	5
4.2	Beroenden	6

5	Tester och kodanalys	7
5.1	Unit-tester	7
5.2	Installera Phing	7
5.3	Kör tester	8
5.4	Deploy	8
6	Loggar	8
7	Databasen	8
7.1	Lagrade funktioner	8
7.2	Events	9
7.3	Triggers	9
7.4	Prefix i kolumnnamn	9
7.5	Tabeller	10
7.5.1	dir__Member	10
7.5.2	dir__Faction	11
7.5.3	dir__Workplace	11
7.5.4	‘xref__’ tabeller	12
8	REST-gränssnittet	13
8.1	Media-typer	13
8.2	Resurser	13
8.3	URLer	15
8.3.1	Resurser	15
8.3.2	Kontrollers	17
8.4	Länktyper	19

9	Webbsäkerhet	19
9.1	Konfiguration av webserver	19
9.2	Lösenord	20
9.3	Spärrade användare	20
9.4	Cross-Site Scripting (XSS)	21
9.5	Cross-Site Request Forgeries (CSRF)	21
9.6	Session hijacking	22
9.7	Säkerhet hos användaren	22
10	Användarmanual webbklient	23
10.1	Webbläsare	23
10.2	Listor	23
10.2.1	Arbetslistor	23
10.2.2	Att arbeta med urklipp	23
10.3	Sökningar	24
10.3.1	Avancerade sökningar	25
10.3.2	Medlemssök	26
10.3.3	Söka efter fakturor	26
10.4	Poster	27
10.5	Kortkommandon	27
10.6	Struliga medlemmar	27
10.7	Typer av data i registret	28
10.7.1	Personnummer	28
10.7.2	Bankkontonummer	28
10.7.3	Plusgirokonton	28
10.7.4	Bankgirokonton	29
10.7.5	Inkomst och avgiftsklass	29

10.7.6	Betalningsvillkor	29
10.8	Kontaktuppgifter	30
10.8.1	Telefonnummer	30
10.8.2	Mailadresser	31
10.8.3	Postadresser	31
10.9	Kända problem	32
11	Ekonomifunktionerna	32
11.1	Bokföringsprogram	33
11.2	Flera kassörer?	33
11.3	Att avskriva fakturor	33
11.4	Kontoplanen	34
11.4.1	Att importera baskontoplanen till VISMA	34
11.5	Avgiftstabellen	35
11.6	Bokföringskanaler	36
11.7	Konteringsmallar	36
11.7.1	Hur konteringsmallarna fungerar	37
11.7.2	Standardvariabler i konteringsmallar	38
11.7.3	Medlemsfodringar	39
11.7.4	Att arbeta med konteringsmallar i VISMA	39

1 Avstämning mot kravspecifikation

Version 0.1.1

Det är svårt för mig att göra en tydlig avstämning mot specifikationerna. Visserligen finns ett par dokument som skrevs i början av projektet, men de är mer eller mindre inaktuella. Mycket har ändrats och tillkommit under arbetets gång, efter mailkonversationer, men framförallt efter personliga samtal.

Det går att bryta ner specifikationerna i fyra huvudgrupper:

1. Medlemmar och organisation.
2. Ekonomi.
3. Kommunikation.
4. Externa aktörer.

1.1 Medlemmar och organisation

Register över medlemmar och organisatoriska grupper. Olika typer av korsorganiseringar, uppdrag, färdigheter med mera. Detta är färdigt att testas.

1.2 Ekonomi

Hantering av medlemsfakturor och export till bokföring. Central rapportering och fakturering mellan olika delar av organisationen.

Fakturahantering och export till bokföring är färdigt att testas. Automatiserad central rapportering saknas.

1.3 Kommunikation

Utskick via mail/post/sms till grupper av medlemmar enligt personliga inställningar. System för detta är skrivet. Ett gemensamt beslut togs att fokusera på ekonomin till alphaversionen. Gränssnitt för att använda kommunikationsmodulen saknas.

1.4 Externa aktörer

Register över arbetsplatser, arbetsgivare, motparter med mera. Detta har lyfts ut ur alphaversionen. Tidigare versioner innehöll ett arbetsplatsregister, men ett beslut togs att separera arbetsplatser från arbetsgivare för att registret bättre ska kunna beskriva verkligheten. Det har inte funnits tid att följa upp denna nya design.

1.5 Slutligen

De delar som finns med i alphaversionen är grundligt testade och utvärderade. Jag är mycket nöjd med resultatet och hoppas att det kommer att uppskattas.

Det har varit ett långt projekt med en otydlig projektplan. Jag har varit både programmerare, designer, projektledare och kommunikatör. Utifrån dessa förutsättningar är jag extra stolt över vad jag har presterat.

Under hela perioden har jag fått sex månadslöner i ersättning. Jag har inte fört bok över hur många timmar jag arbetat, men det är utan problem motsvarande 18 månaders heltidsarbete.

2 Om det här dokumentet

Detta är ett långt och ambivalent dokument. Jag har helt enkelt sammanställt all dokumentation jag skrivit under projektets gång. Vissa delar riktar sig till programmerare som vill förstå och kanske arbeta med systemet. Andra delar riktar sig till styrgrupp och beställare. Ytterligare andra delar riktar sig till användare av systemet. Läs det som intresserar er.

3 Filsystemet

Projektet är organiserat i följande filsystem:

3.1 /

I systemets rot finns

- *build.xml*: se **Tester och kodanalys**
- *composer.json*: se **Serverkonfiguration**
- *php.ini*: de PHP inställningar systemet behöver. Se **Serverkonfiguration**
- *mreg.local*: den apache virtual host som använts under utvecklingen.

3.2 bin

Kompilerade binärer.

- *wkhtmltopdf* används för att generera pdf från html.

3.3 build

Genererad metadata, dokumentation mm. Se **Tester och kodanalys**.

3.4 cli

Cli-skript skript systemet använder sig av. Körs som cron-jobb.

3.5 db

Databasen.

- *DB.sql* En tillfällig kopia av databasen till versionshantering. När systemet går i drift ska denna såklart inte användas längre.
- *EMPTY.sql* Den tomma databasen, vilken såklart inte är en helt tom, utan innehåller en hel del grundläggande data. Använd denna för att sätta upp ett nytt lokalt system.
- *events.sql* De MySQL-events systemet använder sig av (cron-jobb som körs internt i databasen. Finns laddade i *EMPTY.sql*.
- *triggers.py* Python-skript som skapar de MySQL-triggers systemet använder sig av. Finns laddade i *EMPTY.sql*.

3.6 docs

Handskriven dokumentation av systemet formaterat med markdown. Bland annat filen du läser nu.

3.7 libs

De externa PHP-bibliotek systemet använder sig av. Hanteras av **Composer**. Se **Serverkonfiguration**.

3.8 src

Källkoden till systemet.

3.9 tests

Unit-tester för koden i */src/mreg*. Se **Tester och kodanalys**.

3.10 var

Data systemet genererar eller använder sig av. Här finns exempelvis logg- och cachefiler, översättningar och standardfiler.

3.11 www

Den del av systemet som är synlig via http. Webbservern ska konfigureras att använda denna katalog som rot. Här finns bland annat:

- *bootstrap.php* Grundläggande steg som behöver tas för att köra systemet. Returnerar en Pimple dependency injection container med de objekt systemet kräver. (Används av både http och cli.)
- *gateway.php* Accesspunkt för all HTTP-åtkomst. Systemet använder **apache** med **mod_rewrite** för att skicka alla url:er till gateway. Se *mreg.local*.
- *routes.php* Returnerar en Aura route-map med definierade vägar för alla url:er systemet använder sig av.
- *jsclient* JavaScript-klienten. Servern ska ej använda *mod_rewrite* för denna katalog.

- *static* Statiska filer klienten kan hämta över HTTP. Servern ska ej använda *mod_rewrite* för denna katalog.

4 Serverkonfiguration

4.1 Grundläggande krav

1. Webbservern **Apache 2** med **mod_rewrite**
2. **MySQL >= 5.1.6** med följande inställningar:
`sql-mode="STRICT_TRANS_TABLES,ERROR_FOR_DIVISION_BY_ZERO"`
`event_scheduler="ON"`
3. **PHP >= 5.3.3** som apache-modul
 - **PDO** samt **PDO_MYSQL** för databaskopplingar
 - **HASH** med stöd för *MD5*, *sha512* samt *crc32*
 - **bcmath** för flyttalsaritmetik
4. Önskade PHP-inställningar finns i *php.ini* i rot-katalogen.
5. Svensk lokalisering **sv_SE.utf8** eller motsvarande. Används bland annat för att formatera ekonomiska summor. Med en felaktig lokalisering finns risken att ekonomiska beräkningar avrundas felaktigt.

4.2 Beroenden

Registret använder *composer* för att hantera PHP-bibliotek. För att installera beroenden använd *Phing* (se **Tester och kodanalys**) eller *composer* direkt.

```
curl -s http://getcomposer.org/installer | php
php composer.phar install
```

Följande beroenden installeras:

1. [itbz/httpio](#)
2. [itbz/Cache](#)
3. [itbz/Utils](#)
4. [itbz/DataMapper](#)
5. [itbz/libmergepdf](#)
6. [itbz/phplibphone](#)
7. [itbz/phplibaddress](#)
8. [itbz/stb](#)
9. [itbz/Mobilizr](#)
10. [monolog/monolog](#)
11. [emberlabs/gravatarlib](#)
12. [twig/twig](#)
13. [swiftmailer/swiftmailer](#)
14. [Aura/Cli](#)
15. [Aura/Router](#)
16. [AddedBytes/EmailAddressValidator](#)
17. [pimple/pimple](#)
18. [rych/phpass](#)

5 Tester och kodanalys

5.1 Unit-tester

Systemets unit-tester finns i */tests*.

PHPUnit används för testerna. Ställ dig i */tests* och skriv *phpunit*.

OBS! För att köra testerna krävs att de bibliotek registret använder finns installerade. Se **Serverkonfiguration**.

Testerna kan även köras via *Phing* för att automatiskt generera fina rapporter. Då installeras även beroenden automatiskt.

5.2 Installera Phing

Under utvecklingen har **Phing** använts för att köra unit-tester, generera dokumentation med mera. Phing installeras enklast med PEAR.

```
pear config-set preferred_state alpha
pear install --alldeps phing/phing
pear config-set preferred_state stable
```

5.3 Kör tester

Ställ dig i rot-katalogen och skriv *phing* för att köra alla tester. En ny katalog *build* skapas med test-resultat, dokumentation och resultatet från en del andra analysverktyg. De flesta rapporter är html-formaterade, så peka din webbläsare mot *build* för att granska resultatet.

5.4 Deploy

Phing används även för deploy till testserver. Se *build.xml*. Deploy-rutinen är lite rörig på grund av att testservern inte är dedikerad till projektet.

6 Loggar

Systemet använder **Monolog** för att logga händelser.

- Intressanta händelser (loggnivå *info*) loggas till databastabellen *sys_Log*. Det kan exempelvis vara att en användare loggar ut eller att ett felaktigt personnummer försökte sparas.
- Fel (loggnivå *error*) loggas även till fil med diverse extra information. */var/log/mreg.log*
- Allvarliga fel (loggnivå *alert*) skickas även med mail.

Dessa loggar ska ses som komplement till PHPs vanliga errorlog och webbserverns loggar.

7 Databasen

7.1 Lagrade funktioner

Databasen använder lagrade funktioner för att hantera rättigheter. Tabeller med åtkomst-restriktioner definierar följande kolumner:

- *owner*: namn på användare som äger raden
- *group*: namn på användargrupp som äger raden
- *mode*: åtkomstregler för ägare, grupp samt övriga

När servern läser data från dessa tabeller utvärderas åtkomst direkt i databasen med hjälp av lagrade funktioner. Detta betyder att stora mängder av data en användare inte har rätt att se ej behöver kopieras till applikationen för utvärdering. Det betyder också att applikationen ej har tillgång till information om hur många rader åtkomst nekades för. Dessa rader behandlas helt enkelt inte.

7.2 Events

Events är SQL funktioner som automatiskt körs av databasservern vid givna tidpunkter. Liknar cron-jobb i *nix system. Mreg kör följande events:

- Sessioner

Var femte minut scannar servern alla aktiva användarsessioner. Sessioner som varit inaktiva över 30 minuter loggas ut. Sessioner som varit inaktiva över 90 minuter raderas.

7.3 Triggers

Triggers (utlösare) är funktioner som automatiskt körs av databasservern när data ändras. Registret använder ett stort antal utlösare för att automatisera vissa typer av data-underhåll. Dessa redovisas under respektive tabell.

7.4 Prefix i kolumnnamn

- *n* används som prefix för nummer i fall där det kan vara tvetydigt. *groups* skulle till exempel kunna vara en lista över grupper, men *nGroups* är alltid en räknare av antal grupper.
- *t* används som prefix för unix timestamps. Tider sparas generellt som timestamps i vanligt heltalskolumner. *t* används i dessa fall för att förtydliga att det inte är en siffra utan en tidsangivelse.

7.5 Tabeller

Databasen innehåller tabeller av olika art, varje tabells art kan härledas från dess prefix:

Prefix	Beskrivning
aux_	Information relaterad till annat objekt: Adresser, ändringar osv. Ska läsas tillsammans med respektive huvudrad.
dir_	Grundläggande objekt i registret: Medlemmar, grupper, osv.
eco_	Ekonomisk data: Fakturor, avgiftstabeller osv.
lookup_	Statisk data registret använder för att genomföra olika uppgifter.
search_	Sök-tabeller
sys_	Information systemet behöver för att fungera: Användare, loggar..
xref_	Referens-data. Länkar poster från olika tabeller med varandra.

7.5.1 dir_Member

Varje medlem motsvaras av en post i denna tabell.

Icke-normaliserat innehåll För att snabba upp läsningar från databasen cachas namn på medlemmens nuvarande LS. Kopierat från **xref_Faction_Member**.

Observera att detta inte i alla lägen är uppdaterat. Om en medlem har lämnat ett LS, men inte gått med i något nytt, kommer det cachade värdet ej att ändras. Detta är endast kosmetiska problem då dessa värden ej används av systemet.

Triggers

- ETags beräknas innan inserts och updates.
- Ändringar sparas efter updates.

7.5.2 dir_Faction

Varje organisatorisk grupp motsvaras av en post i denna tabell.

Typer av grupper Följande typer finns definierade: ‘ANNAN’, ‘DISTRIKT’, ‘LS’, ‘ORTSSEKTION’, ‘FEDERATION’, ‘SYNDIKAT’, ‘DRIFTSSEKTION’, ‘AVDELNING’, ‘KONCERNFACK’, ‘TEMP’ samt ‘TEKNISK’. Här följer några korta kommentarer för några av de mindre uppenbara grupperna:

- ‘ANNAN’ Detta är en slaskgrupp att använda när ingen annan passar. Undvik att använda den i största möjligaste mån.
- ‘TEMP’ Temporära grupper. Grupper som skapas för ett specifikt, ofta kortsiktigt, ändamål. För att sedan slängas när arbetet är slutfört.
- ‘TEKNISK’ Tekniska grupper för exempelvis de som vill ha Syndikalis-ten hemskickad, eller de som inte fått ett första utskick från LS.

Triggers

- ETags beräknas innan inserts och updates.
- Ändringar sparas efter updates.

7.5.3 dir__Workplace

Varje arbetsplats motsvaras av en post i denna tabell.

Triggers

- ETags beräknas innan inserts och updates.
- Ändringar sparas efter updates.

7.5.4 ‘xref__’ tabeller

Alla xref tabeller har samma struktur. De innehåller:

- Datum när relationen skapades.

- Datum för då relationen avslutades. Detta fält ska endast läsas ifall postens status ej är OK.
- Relationens status. Olika xref tabeller definierar olika möjliga statusvärden. Gemensamt är att OK alltid betyder en aktiv (ej avslutad) relation, samt att annat värde än OK betyder att relationen ej är aktiv (är avslutad). **xref_Faction_Member** definierar exempelvis *OK*, *UTTRÄDD*, *UTESLUTEN*, *ÖVERGÅNG*, *AVLIDEN*, *PENSION* samt *ANNAT* som möjliga statusvärden.
- Då det är nödvändigt kan även en kommentar för att närmare beskriva relationens status sparas. För **xref_Faction_Member** sparas här anledning till utträde eller uteslutning, nytt LS vid övergång samt beskrivning för status *ANNAT*.

Registret kan innehålla flera relationer mellan samma objekt, men endast en relation med status OK för varje objektspar.

Triggers

- Datum för avslutad relation sätts efter update om status ändras till icke OK och datum ej är satt.
- **xref_Faction_Member** uppdaterar **dir_Member** cache för LS efter update och insert.

8 REST-gränssnittet

8.1 Media-typer

- *application/x-sie* är bokföringsdata i **SIE-formatet**. Detta är inte en standardiserad mediatyp.
- *application/x-kml* är konteringsmallar i VISMAS kml-format. Detta är en inte standardiserad mediatyp.
- Om inte annat anges är typ i fortsättningen en subtyp av *application*

8.2 Resurser

HTTP-metoder: G = GET, PU = PUT, D = DELETE, PO = POST

Namn	Mediatyp	Metoder	Beskrivning
Adress	json	G,PU,D	En adress
Telefonnummer	json	G,PU,D	Ett telefonnummer
Mail	json	G,PU,D	En mailadress
Medlem	json	G,PU,D	En medlem
Grupp	json	G,PU,D	En organisatorisk grupp, exempelvis ett LS
Arbetsplats	json	G,PU,D	En arbetsplats
Användare	json	G,PU,D	En användare av registret
Systemgrupp	json	G,PU,D	En systemgrupp <i>användare</i> kan vara medlemmar av
Faktura	json	G,PU,D	En faktura
Verifikat	x-sie	G	0 eller flera verifikat för bokföring
Konteringsmall	x-kml	G,PO	En eller flera konteringsmallar
Kontoplan	x-sie	G,PO	Kontoplan
Bokhållare	json	G,PU	En sammansatt resurs av konteringsmallar, kontoplan, avgiftstabell och bokföringskanaler
Ändringar	json	G	Beskriver ändringar utförda på originalresurs

Namn	Mediatyp	Metoder	Beskrivning
Samling av telefon.	json	G,PO	0 eller fler <i>telefonnummer</i>
Samling av mail	json	G,PO	0 eller fler <i>mail</i>
Samling av adresser	json	G,PO	0 eller fler <i>adresser</i>
Samling av medlemmar	json	G,PO	0 eller fler <i>medlemmar</i>
Samling av grupper	json	G,PO	0 eller fler <i>grupper</i>
Samling av arbetspl.	json	G,PO	0 eller fler <i>arbetsplatser</i>
Samling av systemgrp	json	G,PO	0 eller fler <i>systemgrupper</i>
Samling av användare	json	G,PO	0 eller fler <i>användare</i>
Samling av fakturor	json	G,PO	0 eller fler <i>fakturor</i>
Relaterad grupp	json	D,PU	En relation till en <i>grupp</i>
Relaterad medlem	json	D,PU	En relation till en <i>medlem</i>

8.3 URLer

8.3.1 Resurser

Resurs	URL
Sökning	/search?q={query}&startPage={sida}&itemsPerPage={antal}
Samling av användare	/users
Användare	/users/{namn}
Samling av systemgrupper	/sys_groups
Systemgrupp	/sys_groups/{namn}

Resurs	URL
Samling av grupper	/factions
Grupp	/factions/{id}
Samling av adresser	/factions/{id}/addresses
Adress	/factions/{id}/addresses/{addrId}
Samling av telefon.	/factions/{id}/phones
Telefonnummer	/factions/{id}/phones/{phoneId}
Samling av mail	/factions/{id}/mails
Mail	/factions/{id}/mails/{mailId}
Samling av grupper	/factions/{id}/factions
Relaterad grupp	/factions/{id}/factions/{facId}
Samling av grupper (medlemmar av)	/factions/{id}/member-factions
Relaterad grupp	/factions/{id}/member-factions/{facId}
Samling av grupper (avslutade)	/factions/{id}/history
Samling av medlemmar	/factions/{id}/members
Relaterad medlem	/factions/{id}/members/{memId}

Resurs	URL
Samling av medlemmar	/members
Medlem	/members/{id}
Samling av adresser	/members/{id}/addresses
Adress	/members/{id}/addresses/{addrId}
Samling av telefon.	/members/{id}/phones
Telefonnummer	/members/{id}/phones/{phoneId}
Samling av mail	/members/{id}/mails
Mail	/members/{id}/mails/{mailId}
Samling av grupper	/members/{id}/factions
Relaterad grupp	/members/{id}/factions/{facId}
Samling av grupper (avslutade)	/members/{id}/history
Faktura	/member-invoices/{id}
Bokhållare	/accountant
Konteringsmall	/accountant/kml
Kontoplan	/accountant/accounts

8.3.2 Kontrollers

Om inte annat anges används HTTP POST

Utför	URL
Fakturera medlemmar	/accountant/bill
Skriv ut medlemsfakturer	/accountant/print
Exportera verifikationer	/accoutnant/export
Ladda upp en fil för behandling	/files/upload
Ladda ner fil (GET)	/files/download/{file}
Logga ut session	/controllers/service-logout
Töm servers cache-minne	/controllers/clear-cache
Skapa bokhållare	/controllers/create-accountant

8.4 Länktyper

Namn	Beskrivning
<code>/mreg/rels/members</code>	<i>Samling av medlemmar</i> som är medlemmar av originalresursen (en <i>grupp</i>)
<code>/mreg/rels/factions</code>	<i>Samling av grupper</i> originalresursen är medlem av
<code>/mreg/rels/member-factions</code>	<i>Samling av grupper</i> som är medlemmar av originalresursen (en <i>grupp</i>)
<code>/mreg/rels/history</code>	<i>Samling av grupper</i> originalresursen har varit, men ej längre är, medlem av
<code>/mreg/rels/workplaces</code>	<i>Samling av arbetsplatser</i> originalresursen är länkad till
<code>/mreg/rels/sys_groups</code>	<i>Samling av systemgrupper</i> originalresurs (en <i>användare</i>) är medlem av
<code>/mreg/rels/invoices</code>	<i>Samling av fakturor</i> knutna till originalresursen
<code>/mreg/rels/mails</code>	<i>Samling av mail</i> , lägg till mail
<code>/mreg/rels/addresses</code>	<i>Samling av adresser</i> , lägg till adress
<code>/mreg/rels/phones</code>	<i>Samling av telefon.</i> , lägg till nummer

9 Websäkerhet

9.1 Konfiguration av webbserver

Ett grundläggande säkerhetskrav är att all trafik använder krypterad http (https). Vilka algoritmer och rutiner som används för att säkerställa detta krav är en fråga för systemadministratören. Jag kommer inte att säga mer om detta här.

Det samma gäller hur Apache webserver installeras och konfigureras.

9.2 Lösenord

Lösenord måste vara starka för att godkännas av registret. För att beräkna styrka används en algoritm från [Wolfram Alpha](#).

För att godkännas krävs 100 poäng (kan enkelt ändras), vilket motsvarar cirka 90 bitars entropi.

Utöver detta måste lösenord bytas minst var 6:e månad.

9.3 Spärrade användare

Användare kommer att (mot bättre vetande)

- skriva sina lösenord på lappar som de glömmer på bussen
- glömma bort att logga ut när de går från kontoret
- låta gamla inloggningsuppgifter som egentligen inte behövs vara kvar

För att minimera dessa problem har följande säkerhetsåtgärder vidtagits:

- En användare som försöker logga in efter att inte ha loggat in på över tre månader spärras och uppmanas att kontakta admin.
- En användare som skrivit fel lösenord tre gånger på rad spärras och uppmanas att kontakta admin.
- En användare som inte kommunicerat med servern på 30 min loggas ut.
- Varje användare kan endast ha en aktiv session åt gången. Det är meningen att användare ska vara personliga. Om samma användare försöker starta två sessioner betyder det antingen att samma person öppnat två olika webbläsare, eller att någon utomstående kommit över inloggningsuppgifter. I detta läge loggas alla redan inloggade användare ut, och uppmanas kontakta admin om de misstänker att deras lösenord stulits.

9.4 Cross-Site Scripting (XSS)

XSS attacker innebär att en användare av systemet lyckas skriva data till systemet som när det laddas ner av andra användare körs som ett script i webbläsaren. Om detta lyckas kan cookies och annan data som identifierar den legitima användaren göras tillgänglig för attackeraren.

XSS attacker är möjliga först då en webbapplikation vill att användare ska kunna fylla i olika former av HTML eller HTML-liknande taggar. I all data som skickas till registret filtreras därför alla taggar bort (alla tecken inneslutna av ett < och ett > tecken).

9.5 Cross-Site Request Forgeries (CSRF)

CSRF är när en inloggad användare surfar till en helt annan site, vilken i sin tur innehåller en länk till registret. Registret använder cookies med speciella sessions-nycklar för att identifiera inloggade användare. När användarens webbläsare ber om en resurs från vår server kommer vår identifierande cookie automatiskt att inkluderas i transaktionen. När den inloggade användaren surfar till tredje-parts sidan, som innehåller en länk till registret (en attackeraren skulle till exempel kunna skapa en tråd på ett forum och lägga in en fejkad bild som egentligen är en länk till en viss resurs hos oss), kommer webbläsaren att inkludera rätt cookie automatiskt och vi riskerar att lämna ut känslig information.

Följande åtgärder har vidtagits för att försvara mot CSRF attacker:

- Auktoriserade sessioner identifieras ej enbart med sessions-nycklar i cookies. En extra nyckel (fingerprint) genereras och måste skickas med alla förfrågningar för att de ska gå igenom. För att en CSRF attack ska fungera måste attackeraren på något sätt komma över både cookie och fingerprint.
- Registret nekar alla förfrågningar med en HTTP referer header från någon annan domän.

9.6 Session hijacking

Session hijacking är när en attackerare på något sätt kommit över ett giltigt sessions-id, och därför kan använda siten som om hon loggat in. Sessions-id kan kommas över på en mängd olika sätt, bland annat genom att utnyttja säkerhetshål i webbläsare eller genom att tjuvlyssna på trafik till siten.

Följande åtgärder har vidtagits för att stänga ute även de attackerare som lyckats komma över ett giltigt sessions-id: (Vissa av åtgärderna kan tyckas vara svaga, förhoppningen är att de sammantaget ska utgöra ett starkt försvar.)

- Alla sessions-nycklar (sessions-id) skickas med http cookies, vilket anses vara säkrare än att skicka nycklar på andra sätt.
- Förutom att kräva en giltig sessions-nyckel för att identifiera en användare kräver systemet även ett giltigt fingerprint (se ovan). Fingerprints skickas ej i cookies.
- Vid varje lyckad autentisering genereras en ny sessions-nyckel samt fingerprint.
- Vilken webbläsare (User-Agent) som används sparas vid autentisering och måste vara konstant över hela sessionen.

9.7 Säkerhet hos användaren

Slutligen är det självfallet viktigt att registret används i en generellt säker miljö. Det vill säga att våra användare

- sitter bakom brandväggar
- ej sitter på nätverk det går att ansluta till via öppet WLAN
- eller via trådlösa nätverk med WEP kryptering
- använder senaste versionen av sin webbläsare och kontinuerligt installerar säkerhetsuppdateringar

10 Användarmanual webbklient

10.1 Webbläsare

Det finns en mängd webbläsare som alla skiljer sig åt. Registret har testats med *Google chrome* och *Firefox*. Stöd för andra webbläsare kan inte garanteras.

10.2 Listor

Alla grundläggande typer av poster i registret (grupper, medlemmar, arbetsplatser osv.) kan visas i listor. En lista kan vara en sökning, en virtuell arbetslista, alla medlemmar i ett LS osv. Poster kan flyttas mellan listor genom att kopiera till urklipp (mer om det nedan).

Att lägga till en post till listan över ett LS medlemmar är semantiskt samma sak som att göra posten till medlem av LS.

Poster kan raderas från listor med hjälp av knappen *ta bort*.

Att ta bort en post från listan över ett LS medlemmar är semantiskt samma sak som att avskriva medlemmen.

10.2.1 Arbetslistor

Arbetslistor existerar endast i webbklienten. Det vill säga ingen motsvarande samling existerar i registret. Med hjälp av arbetslistor kan en administratör samla poster som en del i arbetsflödet. Exempelvis för att skriva ut etiketter, kontroller betalningar osv.

10.2.2 Att arbeta med urklipp

Urklipp är en global lista som alltid existerar i bakgrunden. Klicka på knappen *visa urklipp* för att arbeta med den direkt.

När en eller flera poster från vilken lista som helst *kopieras* så hamnar de i urklipp.

När knappen *lägg till* aktiveras på någon lista så *flyttas* samtliga poster från urklipp till denna lista. På detta sätt kan poster flyttas mellan listor.

Vissa listor kan bara innehålla vissa typer av poster. Det är exempelvis inte möjligt att lägga till en grupp till en lista över ett LS medlemmar.

10.3 Sökningar

Sökningar görs mot ett speciellt sök-index. För att ny eller uppdaterad data ska vara sökbar måste det indexeras. Eftersom det är en tidskrävande uppgift görs det inte efter varje uppdatering. Istället indexeras databasen automatiskt med ett visst tidsintervall.

Det betyder att ny eller ändrad data inte är sökbar innan en ny indexering utförts.

I nuläget omskapas indexet varje natt. Om det visar sig vara för sällan kan vi korta intervallet.

För sökningar används följande syntax:

- Varje sökord eftersöks *i sin helhet* i databasen. Det vill säga *malm* matchar INTE *malmö*.
- * används som wildcard i slutet av sökord. Det matchar vad som helst. *malm** matchar exempelvis *malmö*.
- * kan inte användas i början av ord.
- + i början av ett sökord gör att endast sökningar som innehåller ordet returneras (sökord utan + gör att poster med detta ord värderas högre än andra poster, men de utesluts inte).
- - i början av ett sökord gör att inga poster som matchar sökordet returneras.
- Parenteser kan användas för att logiskt gruppera sökord.

- Fraser inneslutna av ” eftersöks i sin helhet, det vill säga ej som enskilda ord.

För den tekniskt intresserade finns mer information på mysql.com.

10.3.1 Avancerade sökningar

Det finns för närvarande inte något gränssnitt för avancerade sökningar. Följande syntax används tills vidare.

- **+type_member** söker efter endast medlemmar
- **+type_faction_*** söker efter endast grupper
- **+type_faction_ls** söker efter endast LS (alternativt använd *distrikt*, *ortssektion*, *federation*, *syndikat*, *driftssektion*, *avdelning*, *koncernfack*, *temp* eller *teknisk*).
- **+id_1** söker efter alla poster med id 1 (av ospecificerad typ)

Det går även att söka efter länkad data.

- **+xref_faction_1_ok** söker efter alla poster som är medlemmar av grupp nummer 1 (Stockholms LS).
- **+xref_faction_1_*** söker efter alla poster som är eller har varit medlemmar av grupp nummer 1.
- **+xref_faction_1_* -xref_faction_1_ok** söker efter alla poster som har varit men inte längre är medlemmar av grupp nummer 1.
- **+xref_member_1_ok** söker efter alla poster som medlem nummer 1 är medlem av.

Allt kan kombineras

- **+type_member +xref_faction_1_ok** söker efter alla medlemmar som är medlemmar av grupp nummer 1.

10.3.2 Medlemssök

- **+sex_f** söker efter kvinnor (alternativt *m* för män eller *o* för annat kön).
- **+debitclass_a** söker efter medlemmar i avgiftsklass A.
- **+paymenttype_ag** söker efter medlemmar som betalar via autogiro.
- **+personalid_NNNNNN-NNNN** söker efter personnummer.
- **+workcondition_anstalld** söker efter medlemmar med arbetsvillkor anställd.
- **+invoiceflag_1** söker efter medlemmar med förfallna fakturor.

10.3.3 Söka efter fakturor

- **+type_invoice_member** söker efter medlemsfakturor
- **+invoice_1** söker efter fakturanummer 1
- **+invoice_ocr_133** söker efter faktura med ocr-nummer 133
- **+invoice_paid_*** söker efter betalda fakturor
- **+invoice_paid_pg** söker efter fakturor som betalats via plusgiro (även *ag*, *bg*, och *k* för autogiro, bankgiro och kontant).
- **+invoice_printed** söker efter utskrivna fakturor
- **+invoice_expired** söker efter förfallna fakturor
- **+invoice_locked** söker efter låsta fakturor
- **+invoice_blanco** söker efter blanco-fakturor
- **+invoice_exported** söker efter fakturor som exporterats till bokföring
- **+invoice_ag** söker efter autogiro-fakturor
- byt *+* mot *-* för att utesluta kriterier. Exempelvis **+type_invoice_member +xref_faction_1 -invoice_paid_*** söker efter obetalda fakturor till grupp nummer 1

10.4 Poster

Poster öppnas på höger sida av arbetsytan. Längst ut till höger av den översta knappraden finns en uppsättning knappar för att arbeta med poster. Dessa knappar arbetar alltid på den post som för tillfället är öppen.

- **Redigera** öppnar en dialogruta för att redigera aktiv post.
- **Uppdatera** hämtar aktiv post från servern. Detta är användbart för att hämta ändringar som gjort på posten sedan den öppnades. Den är inte möjligt att spara en post om det finns ändringar som gjorts efter det att posten öppnades.
- **Kopiera** kopierar posten till urklipp.
- **Radera** tar bort posten från registret. Detta är betydelsefullt att posten verkligen raderas och inte kan återskapas. Var mycket restriktiv med detta.

10.5 Kortkommandon

- **Ctrl+F** öppna ny sökning
- **Alt+Ctrl+N** skapa ny post
- **Ctrl+E** öppna aktiv post för redigering
- **Alt+Ctrl+R** hämta senaste versionen av aktiv post
- **Alt+Ctrl+B** öppnar bokhållare

10.6 Struliga medlemmar

Medlemmar med förfallna fakturor flaggas för att underlätta uppföljning. Under Ekonomi > Status visas om en medlem ligger efter med betalningar. En röd prick betyder att förfallna fakturor finns, en grön prick betyder det motsatta.

Flaggningen uppdateras varje natt då systemet söker igenom fakturaregistret. En medlem kan alltså under en kortare tid vara flaggad trots att betalning registrerats. Listan över obetalda fakturor är alltid korrekt.

Det går även att söka efter flaggade medlemmar med **invoiceflag_1**.

10.7 Typer av data i registret

10.7.1 Personnummer

Ange personnummer med 10 siffror (6 siffror, avdelare, 4 siffror). För personer utan svenskt personnummer ersätts de fyra sista siffrorna av xxxx, alternativt xx1x för män eller xx2x för kvinnor. För personer över hundra år används plustecken (+) som avdelare mellan datum och födelsenummer, annars används bindestreck (-).

10.7.2 Bankkontonummer

Används endast med autogiro. Ange konto med clearing- (alltid fyra siffror) och löpnummer skiljda med ett kommatecken (,).

Clearingnummer ska alltid anges med fyra siffror. Vissa banker (Swedbank) använder clearingnummer med fem siffror. Då är den sista siffran en intern kontrollsiffra som inte ingår i det nationella banksystemet. Kontrollsiffran ska vid inmatning i registret utelämnas.

Systemet visar bankens namn. Detta läses från clearingnumret. Om bankens namn är okänt behöver det inte betyda att kontonumret är felaktigt ifyllt (systemet känner inte till alla banker).

10.7.3 Plusgirokonton

Ett Plusgironummer kan bestå av lägst två siffror och högst åtta siffror ($x-x$ till $xxx\ xx\ xx-x$). Den sista siffran är en kontrollsiffra.

10.7.4 Bankgirokonton

Ett bankgironummer består av sju eller åtta siffror (*xxx-xxxx* eller *xxxx-xxxx*). Den sista siffran är en kontrollsiffra.

10.7.5 Inkomst och avgiftsklass

Inkomst anges som månatlig sammanlagd inkomst före skatt (efter sociala avgifter).

Om uppgift om inkomst finns kommer avgiftsklass att beräknas baserat på inkomst (vid fakturering).

Om uppgift om inkomst saknas kan avgiftsklass anges explicit, och kommer i förekommande fall att användas vid fakturering.

Om uppgift om både inkomst och avgiftsklass saknas kommer blanco-fakturer att genereras.

För att en medlem ska övergå till blanco-fakturering måste alltså både uppgift om inkomst och avgiftsklass tömmas.

10.7.6 Betalningsvillkor

Systemet använder sig av följande betalningsvillkor:

- **AG** (autogiro). Medlemmen betalar via autogiro. Systemet prickar av betalningar automatiskt från autogirots rapportfiler.
- **BAG** (begär autogiro). Medlemmen önskar betala via autogiro. Generera autogirofil och skicka till banken för behandling. (Omvandlas automatiskt till BAG-V när autogirofil hämtats från systemet.)
- **BAG-V** (begär autogiro väntar på svar). Begäran om autogiro har skickats till banken. (Omvandlas automatiskt till AG när klartecken kommer från banken.)
- **MAG** (makulera autogiro). Medlemmen önskar inte längre betala via autogiro. (Omvandlas automatiskt till MAG-V när autogirofil hämtats från systemet.)

- **MAG-V** (makulera autogiro väntar på svar). Begäran om att makulera autogiro har skickats till banken. (Omvandlas automatiskt till LS när klartecken kommer från banken.)
- **PAG** (problem med autogiro).
- **LS** Ej autogiro utan betalning direkt till LS.

10.8 Kontaktuppgifter

Varje post i registret kan ha ett obegränsat antal telefonnummer, mail- och vanliga adresser. Varje kontaktuppgift identifieras av ett namn. Detta namn fyller ingen teknisk funktion, utan visas endast för att det skall vara enkelt att identifiera olika kontaktuppgifter. Det finns inget som hindrar att en mailadress kallas för *Tele*, även om det såklart skulle vara olyckligt.

När en kontaktuppgift öppnas för redigering visas när och av vem uppgiften senast sparades. Tidskoden ändras varje gång posten sparas, även om ingenting faktiskt ändrades. Använd denna funktion för att markera att en kontaktuppgift fortfarande är giltig, exempelvis efter ett telefonsamtal till en medlem.

10.8.1 Telefonnummer

Telefonnummer kan matas in i registret i olika format. Följande telefonnummer är identiska från systemets perspektiv:

- +468123456
- +46-8-123 456
- 08-123456
- 08123456

Om landskod fylls i måste den föregås av ett *+*. Om landskod ej fylls i är *46* standard.

Till höger om telefonnummer visas antingen en grön eller en röd prick. Grön prick betyder att numret är ett svenskt mobilnummer. När systemet skickar SMS eller skriver ut listor över SMS-nummer tas endast dessa nummer med.

Röd prick betyder att numret inte är ett giltigt svenskt mobilnummer.

När telefonnummer öppnas för redigering visas ytterligare information om land, område och operatör. Denna information läses från telefonnumret och är ett stöd för att organisatören enkelt ska kunna se att allt ser korrekt ut.

10.8.2 Mailadresser

Till höger om mailadresser visas antingen en grön eller en röd prick. Grön prick betyder att mailadressen är giltig. När systemet skickar mail eller skriver ut listor av mailadresser tas endast dessa adresser med.

Röd prick betyder att adressen inte är en giltig mailadress.

10.8.3 Postadresser

Postadresser sparas i registret nedbrutna i en mängd fält. Den kompletta adressen byggs upp enligt prioriteringsregler från **SIS**. För att det inte ska bli fel är det viktigt att informationen är ifylld på rätt sätt. Kort beskrivning av mindre uppenbara fält:

- **Förmedlande mottagare** Även känt som care-of eller c/o. För care-of-adresser ska *mottagarrolluppgift* vara *c/o* och *mottagare* namnet på den förmedlande personen/institutionen.
- **Administrativ adress** Kan exempelvis vara en postbox eller poste restante. För en boxadress ska *Typ av administrativ adress* vara *box* och *administrativ adress* postboxens nummer.
- **Uppgång** Exempelvis *UH* för uppgång höger eller *U2* för uppgång nummer två.
- **Våningsplan** Exempelvis *NB* för nedre botten eller *2TR* för våning nummer två.

- **Lägenhetsnummer** Lägenhetsnummer som det skrivs i svensk folkbokföring. Endast siffror.
- **Extra uppgift** Ett väldigt fritt fält. Kan exempelvis vara *mittemot nummer 50* eller *50 meter till vänster om huvudingång*.
- **Landskod** Kod enligt **ISO 3166-1**, där exempelvis *SE* står för Sverige.

10.9 Kända problem

Följande begränsningar finns i alpha-versionen:

- Att ladda upp filer fungerar endast med Google Chrome.
- Möjligheterna att redigera bokhållare är begränsade
- Listan över medlemmar hos varje grupp kan endast visa 300 poster. För grupper med fler medlemmar är den kortsiktiga lösningen att söka efter `+type_member +xref_faction_id_ok`, där `id` ersätts med gruppens id.

11 Ekonomifunktionerna

Alla lokala grupper i registret som behöver kunna arbeta med ekonomiska funktioner (generera fakturor, exportera verifikationer till bokföring osv) måste skapa en så kallad *bokhållare*. En bokhållare är ett sammansatt objekt som består av *kontoplan*, *avgiftstabell*, *bokföringskanaler* samt *konteringsmallar*.

Samtliga delar krävs för att ekonomifunktionerna ska fungera. Att skapa dem kan vara lite krångligt, men i gengäld behöver de sällan uppdateras.

Börja med att kontakta den systemansvarige för att skapa en standard-bokhållare.

11.1 Bokföringsprogram

Ekonomidelen av registret är skriven för att användas tillsammans med något fristående bokföringsprogram. Verifikationer exporteras till bokföring i filformatet **SIE**, vilket är branschstandard i Sverige. Alla program med stöd för import av SIE-filer är lämpade för användning tillsammans med registret.

Men...

Extra fokus har lagts på interaktionen med bokföringsprogram från **VISMA**. Speciellt att komma igång underlättas för den som använder något av VISMA:s program. I det följande kommer jag även att ge specifika tips och instruktioner riktade till VISMA-serien.

11.2 Flera kassörer?

Det är rekommenderat att varje lokalgrupp endast har en användare med rättigheter att arbeta med ekonomi. I de fall flera användare behövs är det mycket viktigt att dessa har samma primära system-grupp samt att alla fakturor görs tillgängliga för användare i denna grupp. Om detta inte görs finns risken att den ena kassören skapar fakturor som den andra inte kan läsa, vilket i sin tur leder till att både avprickning mot reskontra och export till bokföring ej kommer att stämma.

Om ni inte till 100% förstår detta: Kontakta sysadmin innan ni skapar flera användare som kan arbeta med ekonomin.

11.3 Att avskriva fakturor

Obetalda fakturor kan raderas från registret. Till exempel om en medlem lämnar organisationen kan inaktuella fakturor behöva tas bort. Ibland kan detta dock ställa till problem.

När registret tar emot en betalning som är lägre än det fakturerade beloppet så bokförs mellanskillnaden som en fodring på medlemmen, och en ny faktura genereras med denna summa. Om den nya fakturan sedan raderas kommer fodringen att hänga kvar i bokföringen. Systemet kan i detta läge inte hjälpa till att avskriva fodringen, utan det måste göras manuellt i bokföringen.

Ett liknande problem kan uppstå om exempelvis en medlem lämnar organisationen med obetalda fakturor och avdelningen väljer att inte kräva in betalning. Ifall det rör sig om gamla fakturor som redan har rapporterats centralt kan de inte längre raderas. I detta läge kan administratören välja att markera fakturan som betald med 0 kr. För att undvika att en ny faktura genereras ska då *fakturera fordran* INTE vara markerad. Fordringen kommer att exporteras till bokföringen och måste där avskrivas manuellt.

11.4 Kontoplanen

För att kunna generera fakturor och verifikationer måste registret innehålla en kontoplan. För att export av verifikationer ska fungera felfritt måste kontoplanen i registret vara *exakt* samma som kontoplanen i bokföringsprogrammet. Det innebär att ett kontos nummer, typ OCH namn måste vara samma.

För att undvika att hamna i en situation där det är oklart vilken kontoplan som gäller är det rekommenderade arbetssättet att endast redigera kontoplanen i bokföringsprogrammet, och sedan exportera den till registret.

I alpha-versionen kommer registret inte heller innehålla stöd för att arbeta direkt med kontoplanen.

Registret kan läsa kontoplaner i ett format utvecklat av VISMA. Detta format är nära knutet till det officiella SIE-formatet, men är inte standard (SIE innehåller egentligen inte något format för kontoplaner). Det är alltså osäkert om andra bokföringsprogram än VISMA kan arbeta med kontoplaner i detta format.

Tillsammans med den här guiden bör ni ha fått en kopia av baskontoplanen. *Baskontoplan.kp*. Det är rekommenderat att utgå ifrån denna när ni utvecklar kontoplanen för er lokalgrupp. Om ni använder baskontoplanen rakt av behöver ingenting exporteras till registret, den används som standard.

11.4.1 Att importera baskontoplanen till VISMA

Att importera en kontoplan i VISMA kan bara göras vid nytt bokföringsår.

Först måste ni kopiera *Baskontoplan.kp* till katalogen för kontoplaner. Vad den ligger kan variera. Min installation av *VISMA Förening* i *Windows XP* använder

C:\Documents and Settings\All Users\Application Data\SPCS\SPCS
Förening\Gemensamma filer\Kpl\

Baskontoplanen baseras på EUBAS97. Som standard söker *VISMA* efter kontoplaner baserade på BAS2010 eller liknande. Ni måste istället välja EUBAS97 för att *VISMA* ska hitta kontoplanen.

När baskontoplanen väl är importerad kan ni omarbета den efter behov och sedan exportera den till registret.

11.5 Avgiftstabellen

Avgiftstabellerna i registret är hierarkiskt ordnade. Högst upp finns en central tabell där centrala avgifter per medlem definieras. Under den centrala tabellen finns distrikten, och under dem lokala avgiftstabeller. Denna ordning innebär att om de centrala avgifterna ändras så ändras de automatiskt i distriktens avgiftstabeller, vilket i sin tur gör att de även ändras lokalt.

På samma sätt ändras distriktens avgifter automatiskt i de lokala avgiftstabellerna. Kontrollera att rätt distrikt ställts in.

I tabellen kan ni sedan skapa en eller flera avsättningar. Med avsättning menas en summa som dras från varje inbetald avgift för att användas till något speciellt ändamål. Vanliga exempel är stridsfond, studiefond och avsättning till annan grupp.

När ni skapar avsättningar får endast *VERSALER* användas, ÅÄÖ är ej tillåtna. Var noga med namnen, de ska senare användas som summa-variabler när ni skapar konteringsmallar.

Slutligen skapar ni de olika avgiftsklasser ni använder er av. Anger inkomstintervall, avgift och storlek på avsättningar. Notera att avgiften som anges är den samlade avgiften. De olika avsättningarna kommer sedan att dras från varje inbetalda avgift, och den summa ni fritt kan röra er med blir alltså avgift - avsättningar. (I extrema fall kan denna summa till och med bli negativ:

exempelvis ifall ni sätter er avgift lägre än den centrala avgiften. Registret innehåller inte några spärrar för detta, det är upp till varje lokalgrupp att hålla koll på sin avgiftstabell.)

För varje avgiftsklass måste även en konteringsmall anges.

11.6 Bokföringskanaler

Med hjälp av bokföringskanaler länkas data från registret till faktiska konton i kontoplanen. En inbetalning till plusgirokonto måste exempelvis översättas till motsvarande bokföringskonto (1920 PlusGiro i baskontoplanen). På samma sätt måste en medlems gruppstillhörigheter länkas till kontoplanens konto för samma dessa grupper.

Mera specifikt måste dessa bokföringskanaler finnas:

1. Alla avgiftsklasser
2. *K*, *PG*, *BG* samt *AG* (Kontant, PlusGiro, Bankgiro samt Autogiro)
3. Alla grupper som ni gör avsättningar till

Kanaler *K*, *PG*, *BG*, *AG* samt en standarduppsättning avgiftsklasser skapas automatiskt när ni skapar bokhållare. Dock kan ni behöva ändra dessa så att de stämmer med kontoplanen. Använder ni baskontoplanen behöver dessa kanaler inte ändras.

Om ni gör avsättningar till andra grupper måste ni alltid lägga till kanaler för detta.

11.7 Konteringsmallar

Det sista och mest komplicerade steget är att skapa de konteringsmallar registret använder för att generera verifikationer. Jag kommer börja med att beskriva konteringsmallarna allmänt, för att sedan specifikt beskriva hur VISMA hanterar mallar.

Delar av er ekonomi kommer att skötas utanför registret. Och det är fullt möjligt att ni kommer att bokföra transaktioner manuellt som liknar de registret hanterar. För att dessa ska bokföras på samma sätt är det önskvärt

att samma konteringsmallar finns i bokföringsprogrammet som i registret. Det rekommenderade arbetssättet är därför att skapa konteringsmallarna i bokföringsprogrammet och sedan exportera dem till registret. Detta är dock i nuläget endast möjligt om ni använder VISMA.

I alpha-versionen kommer registret inte heller innehålla stöd för att arbeta direkt med konteringsmallarna.

Tillsammans med den här guiden bör ni ha fått en fil med standardmallar, *konteringsmallar.kml*. Denna fil innehåller förslag till konteringsmallar i ett format specifikt för VISMA. Om ni har möjlighet att importera dessa mallar så gör det redan nu, då blir det enklare att följa med.

konteringsmallar.kml innehåller följande mallar:

- **MSKULD**. Bokför inbetald fordran på medlem. Mer om detta nedan.
- **MAVG**. Standardmall som kan användas för alla avgiftsklasser som gör avsättningar till grupp och find och använder sig av baskontoplanen. Det rekommenderas att ni använder denna mall som utgångspunkt när ni skapar nya mallar.

11.7.1 Hur konteringsmallarna fungerar

En konteringsmall är en verifikation där vissa delar lämnats tomma. Att omvandla en konteringsmall till en verifikation innebär att fylla alla tomma platser med information.

De tomma platserna kan vara en del av verifikationstexten, ett kontonummer eller en summa.

För att registret ska kunna veta vilken information som ska stoppas in på vilken plats används så kallade variabler. Istället för att skriva en summa skriver ni namnet på en variabel. Registret ersätter sedan variabeln med rätt summa.

Variabelnamn måste alltid skrivas med VERSALER (ÅÄÖ är ej tillåtet) och omslutas av { samt }. Såhär: **{VARIABEL}**

Registret innehåller ett antal fördefinierade variabler som alltid finns tillgängliga. Se tabell.

Ni kan även använda de avsättningarna från avgiftstabellen. Om ni exempelvis gör en avsättning till en stridsfond och har döpt denna till *SF* så kan ni för varje inbetald avgift bokföra avsättningen till stridsfond med hjälp av summa-variabeln **{SF}**.

Det är mycket viktigt att varje konteringsmall bokför summa-variablerna **{REST}** och **{FORDRAN}**. Om de utelämnas kommer det inte att gå att bokföra inbetalningar där den inbetalda summan ej överensstämmer med den förväntade.

Titta gärna på hur standardmallarna är konstruerade.

11.7.2 Standardvariabler i konteringsmallar

Alla variabelnamn består av VERSALER. ÅÄÖ är ej tillåtet.

Variabel	Typ	Beskrivning
AVGIFT	Summa	Förväntad avgift
SUMMA	Summa	Faktiskt inbetald summa
REST	Summa	Del av <i>summa</i> som överskrider <i>avgift</i>
FORDRAN	Summa	Del av <i>summa</i> som underskrider <i>avgift</i>
CENTRAL	Summa	Avsättning centrala avgifter
DISTRIKT	Summa	Avsättning distriktsavgifter
BETKANAL	Kontonummer	Tillgångskonto för den inbetalda summan
AVGIFTSKANAL	Kontonummer	Intäktskonto avgiften ska bokföras på
F-NR	Verifikationstext	Fakturanummer
OCR	Verifikationstext	OCR-nummer
SPEC	Verifikationstext	Fakturans specifikation
M-NR	Verifikationstext	Medlemsnummer

Utöver detta tillkommer summa-variabler för de avsättningar som definierats i avgiftstabellen.

11.7.3 Medlemsfodringar

När en medlem betalar för lite genererar registret automatiskt en ny faktura för att kräva in den del av avgiften som saknas. För att göra det används mallen **MSKULD**. Hur standardversionen av **MSKULD** ser ut i VISMA visas i figur 2.

11.7.4 Att arbeta med konteringsmallar i VISMA

VISMA kan skapa konteringsmallar av olika typer. Vi är endast intresserade av mallar för bokföring. När ni skapar en ny mall, se till att typ av mall är *bokföring*.

Ge varje mall ett unikt ID. Det är detta ID som ska anges när ni länkar avgiftsklasser till mallar.

Namn är en lite längre beskrivning av mallen, har ingen teknisk betydelse.

Verifikationsserie och *interna anteckningar* berör endast VISMA (ej registret). De förinställda värdena fungerar i de flesta fall.

I *verifikationstexten* har ni möjlighet att använda variabler för fakturanummer ,ocr-nummer osv.

Under *konteringar* fyller ni sedan i önskade transaktioner. I de fall summor och kontonummer är kända på förhand använder ni dem direkt, annars använder ni någon av registrets variabler.

Id:

Namn:

Typ av mall

☒ Bokföring
☐ Leverantörsfaktura
☐ Periodisering/Fördelning

Verifikationsserie:

Verifikationstext:

Interna anteckningar:

Konteringar

Leverantörsfaktura

☐ Total
☐ Frakt
☐ Moms

Konto	Resultatenhet	Belopp
{AVGIFTSKANAL}		-{AVGIFT}
4110		{SAC}
2421		-{SAC}
4120		{DISTRIKT}
2422		-{DISTRIKT}
4130		{SF}
2071		-{SF}
4150		{SYNDIKAT}
{SYNDIKATKANAL}		-{SYNDIKAT}
{BETKANAL}		{SUMMA}
3990		-{REST}
1510		{FORDRAN}

Figure 1: MAVG. Standardmall för medlemsavgifter.

Id:

Namn:

Typ av mall

☒ Bokföring

☐ Leverantörsfaktura

☐ Periodisering/Fördelning

Leverantörsfaktura

☐ Total

☐ Frakt

☐ Moms

Verifikationsserie:

Verifikationstext:

Interna anteckningar:

Konteringar

Konto	Resultatenhet	Belopp
1510		-{AVGIFT}
{BETKANAL}		{SUMMA}
3990		-{REST}
1510		{FORDRAN}
<input type="text"/>	<input type="text"/>	<input type="text"/>

Figure 2: **MSKULD**. Inbetalning av skuld från medlem.