

# Rapport: PocketPolitics

*Utveckling av en Android-app*

## **Gruppmedlemmar:**

Björn Bergqvist

Johan Hasselqvist

Leif Schelin

## Sammandrag

I en tid då väljare släpper sina politiska traditioner för att se sig om och fundera på vilket parti som representerar dem bäst mister de klassiska medierna åter sitt grepp till förmån för den nya tekniken. Liksom radion och TV:n på sin tid har smartphones i nutiden revolutionerat världen.

Behovet för en ny informationskanal som kan visa användare vad som händer i Riksdagen uppfylls av detta projekt. Det utvecklade resultatet är en prototyp för en fullständig app som visar att det är möjligt.

# Innehållsförteckning

## [Sammandrag](#)

### [1 Inledning](#)

#### [1.1 Bakgrund](#)

#### [1.2 Syfte](#)

#### [1.3 Avgränsning](#)

### [2 Teori](#)

#### [2.1 Android API](#)

#### [2.2 XML parsing](#)

#### [2.3 Json och Gson](#)

#### [2.4 Relationsdatabas, SQL och PHP](#)

#### [2.5 Hashning med salt](#)

#### [2.6 Riksdagen](#)

#### [2.7 Övriga begrepp](#)

### [3 Metod](#)

#### [3.1 Kravframställning och RAD](#)

#### [3.2 Domänanalys och SDD](#)

#### [3.3 Implementationprinciper](#)

#### [3.4 Testning](#)

#### [3.5 Strukturanalys](#)

### [4 Resultat](#)

#### [4.1 Från ett användarperspektiv](#)

#### [4.2 Från ett utvecklarperspektiv](#)

### [5 Diskussion](#)

#### [5.1 Projektets gång](#)

#### [5.2 Projektets framtid](#)

### [6 Slutsats](#)

## [Källförteckning](#)

## [Bilaga: UML-diagram](#)

### [Paketdiagram](#)

#### [Klassdiagram: model.riksdag](#)

#### [Klassdiagram: model.user](#)

#### [Klassdiagram: view](#)

#### [Klassdiagram: control](#)

#### [Klassdiagram: net.riksdag](#)

#### [Klassdiagram: net.server](#)

#### [Sekvensdiagram: registrera sig](#)

#### [Sekvensdiagram: öppna en artikel](#)

# 1 Inledning

## 1.1 Bakgrund

“All offentlig makt i Sverige utgår från folket och riksdagen är folkets främsta företrädare” (Riksdagen, 1974).

Undersökningar visar ett tydligt mönster hur röstare har förändrats genom årtiondena. Tidigare var det vanligare att motivera sin röst med arbetsklass, vana och en känsla av tillhörighet eller identifiering med ett parti (Oscarsson & Holmberg, 2011). Idag har det blivit betydligt vanligare att växla mellan att rösta på olika partier. Alltför många röstare motiverar också sitt val med anledningar som kompetenta partiledare och bra ideologi. Detta visar att människor idag är öppnare för nya tankar och idéer vilket ger media en viktig roll.

Opinionsbildningen idag går i huvudsak till genom media. Många tidningar återspeglar sin ideologiska tillhörighet i sin text och det spekuleras i hur folket kommer påverkas av det alltmer koncentrerade mediaägandet (Ohlsson, 2010). Framför allt påverkar kanske inte tidningarna innehållet med hur de skriver något utan om vad de väljer att skriva (Roosvall, 2005). Tidningar skriver om det som är aktuellt och politik är alltid mer aktuellt under valperioder än i övriga tider. Detta ger ingen helhetsbild av Riksdagens arbete som fattar lika mycket beslut vare sig det är valperiod eller inte. För att kunna få en korrekt uppfattning av vad de olika partierna genomför, inte bara vad de lovar inför valet, så behövs det en kontinuerlig och politiskt oberoende källa vilket inte alltid kan sägas om de privata medierna.

Som kontrast till de privata medierna finns myndigheters egna officiella informationskanaler. Dessa ska vara objektiva och kontinuerliga men kan ha svårt att nå ut till alla människor som inte tar sig tid till att gå in på riksdagen.se eller motsvarande myndighets sida för att läsa senaste nytt. Dessutom innehåller de enormt mycket information, och det kan som privatperson vara svårt att sätta sig in i dess helhet, så man skulle behöva ett lager mellan dessa informationskanaler och väljarna som sällar ut det viktiga och presenterar det på ett lämpligt sätt.

Många i Sverige idag äger en smartphone, särskilt unga människor (Mobilsverige, 2012). Dessa spenderar generellt mindre tid till att läsa tidningar och lyssna på radio. Majoriteten av smartphone-ägarna i Sverige använder Android (GlobalWebIndex, 2012). För att nå ut till den nya generationen behövs en ny modern informationskanal.

## 1.2 Syfte

Projektets syfte är att göra det enkelt för väljare att se vad som händer politiskt i Riksdagen och vad olika partier genomför istället för vad de lovar. Väljarna ska också själva kunna ta ställning till Riksdagens beslut, diskutera dessa med andra användare, samt se statistik över vilket parti som överensstämmer bäst med väljarens egen åsikt. Målet är att skapa en app till Android som tillhandahåller detta med officiell information som hämtas kontinuerligt från Riksdagen.

Rapportens syfte är att redogöra för appens utveckling, funktionalitet och bakomliggande teknologi. Krav och aspekter som tagits i betänkande utvärderas också.

### **1.3 Avgränsning**

Applikationen är endast byggd för Riksdagens kanaler. Andra politiska organ som exempelvis kommun, landsting eller EU är inte medtagna. Svenska är det enda språket som stöds av appen.

Appen utvecklas endast till Android-telefoner. Appen kräver API nivå 14 (Android 4.0: Ice Cream Sandwich) eller högre.

## 2 Teori

Detta kapitel informerar läsaren om grundläggande begrepp och teknologier som används i projektet.

### 2.1 Android API

Utveckling till Android sker i Java och XML (Android Developer, 2013b). XML-filerna innehåller bland annat grafiska komponenter medan Java-klasserna innehåller kod som hanterar och visar de grafiska komponenterna.

Androids bibliotek innehåller inte bara grafiska komponenter utan också annan viktig funktionalitet. En sådan är möjligheten att genomföra operationer asynkront (se Kap 2.7) på ett enkelt sätt genom klassen AsyncTask (Android Developer, 2013c). Multitrådning är särskilt viktigt för tidskrävande metoder som nätverksoperationer så att GUI:t alltid upplevs responsivt.

### 2.2 XML parsing

XML är ett av flera internationella standardformat för att lagra och överföra data på internet (Walsh, 1998). XML är ett strukturerat format med hierarkiska nivåer.

Parsing innebär att man tolkar text-baserad information och omsätter informationen till objekt. Tillgängligt för Java finns flera bibliotek som läser XML. Det finns flera bibliotek som Android rekommenderar varav ett är kXML (Android Developer, 2013a). En fördel med just kXML är att det är implementerat med målet att göra operationerna så minnessnåla som möjligt (Haustein, 2005). Detta kan vara viktigt för smartphones som i regel har mycket mindre RAM-minne än andra datorer.

### 2.3 Json och Gson

Json utvecklades som en ny standard för effektiv textbaserad dataöverföring (JSON, 2013). Den är jämförbar med XML men lämpar sig bättre för enklare data (Megginson, 2007).

Gson är ett bibliotek utvecklat av Google för att förenkla skapandet och tolkningen av Json-objekt till Java (Google Code, 2013). Med en enda rad kod kan man använda Gson för att omvandla Json-objekt till och från Java-objekt.

### 2.4 Relationsdatabas, SQL och PHP

En relationsdatabas är ett vanligt sätt att lagra data i form av tabeller på en server. Då databastypen bygger på tabeller och index är det lätt att hitta och manipulera relevant data.

SQL (Structured Query Language) är ett språk som används för att hämta, lägga till eller modifiera data i en relationsdatabas med hjälp av SQL-anrop. Dessa anrop skapas och skickas av andra språk som exempelvis PHP.

PHP är ett serverspråk, vilket innebär att det utför alla operationer på en server, som till stor del

är dold för användare. Det innebär att känslig information, exempelvis lösenord, som hanteras här är mycket säkrare än information som hanteras direkt på en användarklient.

## 2.5 Hashning med salt

Känslig information som lösenord bör aldrig lagras som de är, eftersom de flesta personer använder samma lösenord på många olika ställen (InfoWorld, 2011). Därför är det viktigt att man vid lagring av dessa känsliga data modifierar dem så att det inte går att ta reda på vad de ursprungligen var. Detta görs med olika hash-funktioner.

En bra hash-funktion omvandlar en sträng (en sekvens av tecken) till en helt annan sträng som alltid är lika lång, oavsett ursprunglig textlängd och på ett sådant sätt att det inte går att matematiskt ta fram den ursprungliga sekvensen av tecken (Defuse Security, 2013). Dock så finns det långa listor med lösenord och deras hashade motsvarighet så att det ofta, beroende på vad för lösenord man har, går att jämföra det hashade lösenordet mot listan och hitta det ursprungliga lösenordet. För att öka säkerheten använder man sig utav ett salt.

Ett salt är en sträng, som bör vara unik för varje användare, som läggs till lösenordet innan man hashar det. Det går då inte att jämföra det med andra hashade lösenord, eftersom de inte har samma salt.

## 2.6 Riksdagen

Alla förslag från regeringen (s.k. propositioner) och alla förslag från ledamöter (s.k. motioner) ska beredas innan de tas upp för röstning i kammaren, där alla ledamöter sitter (Riksdagen, 2011). Beredningen utförs av utskott som tar ställning till förslaget innan de presenterar det för Riksdagen. Utskottet producerar en eller flera utskottsförslag som vardera innehåller en kollektiv rekommendation om att avslå eller godta en mängd förslag. Dessa utskottsförslag presenteras för kammaren i ett s.k. betänkande, som alltså kan innehålla flera utskottsförslag.

För varje utskottsförslag ska ett beslut fattas. Ifall de flesta ledamöter verkar eniga med utskottets rekommendation fattas beslutet utan röstning, detta kallas för en acklamation. Vid oenighet uppstår debatt följt av omröstning.

## 2.7 Övriga begrepp

- Asynkron process - en process som körs på en egen tråd utan synkronisering med huvudtråden förrän processen gått färdigt.
- GUI - grafiskt användargränssnitt.
- MVC - Model-View-Controller: ett arkitekturmönster som separerar domänmodell, visningsklasser och kontroller till olika avgränsade paket.
- Package - ett paket av relaterade Java-klasser.
- Sträng - en sekvens av tecken.
- UML - ett standardiserat sätt att rita diagram över kod. Detta visualiserar framför allt dess struktur på klass- och paket-nivå.

## 3 Metod

Projektet har följt en agil utvecklingsprocess med flera iterationer, en iteration per vecka. I varje iteration ingick två gruppmöten och ett handledarmöte. Detta kapitel presenteras olika delar av processen.

### 3.1 Kravframställning och RAD

För varje ny funktionalitet har ett mål formulerats i projektets RAD: Requirements and Analysis Document. Det är viktigt att alla projektmedlemmar och kunden är eniga med vad som ska utvecklas och hur det ska fungera.

### 3.2 Domänanalys och SDD

Genom att ständigt rita UML-diagram och motivera designvalen i systemarkitekturen finslipas också programstrukturen. Skrivandet kräver viss analys som identifierar mer optimala sätt att strukturera klasserna. Det kan handla om att flytta upp funktionalitet till basklasser eller dela upp paket i underpaket.

### 3.3 Implementationprinciper

Skrien nämner 10 principer för elegant programvara (Skrien, 2007). I alla designbeslut under projektet som rör arkitekturen har målet varit att följa dessa principer. Särskilt noggrant har analys skett med avseende på extensibility: öppenheten för att utöka programmet senare.

### 3.4 Testning

Att testa all funktionalitet genom användarinterfacet är någorlunda problematiskt. Till exempel kan man inte veta om det är datan eller representationen av datan som är felaktig. Det är också tidskrävande att genomföra alla test för hand då en ny funktion lagts till. Ett sätt att göra automatiserade tester på i Android-projekt är att använda verktyget Android JUnit. I projektet användes detta framför allt till att testa och felsöka nätverksoperationer.

### 3.5 Strukturanalys

Det objektorienterade paradigmet följer en hållbar struktur på klass-nivå med inkapsling av data och polymorfism. För att ett projekt ska vara smidigt att vidareutveckla måste även dess arkitektur vara robust (Microsoft, 2013). Det måste alltså finnas en god struktur för hur klasserna förhåller sig till varandra på paket- och projekt-nivå.

STAN är ett verktyg för att analysera ett projekts kodstruktur på flera nivåer från enskilda attribut och metoder upp till hela projektet. STAN räknar ut många olika värden som kan ge utvecklarna fingervisning om var systemarkitekturen behöver underbyggas för att vara hållbar eller bekräfta en god design. Typiska exempel är upptäckten av cykliska beroenden, komplicerad logik eller långa metoder. Ibland kan detta vara motiverat men STAN hittar dem åt utvecklarna som därefter kan fatta intelligenta beslut.



## 4 Resultat

Detta kapitel beskriver den färdiga produkten, både hur den ter sig för en användare och hur koden är.

### 4.1 Från ett användarperspektiv

Den färdiga produkten är uppdelad i tre hierarkiska nivåer förutom inloggning och registrering. Den översta nivån är en scrollningsbar lista utav betänkanden vars primära funktion är navigation. Nästa nivå ger en detaljerad bild utav ett betänkande samt listar dess utskottsförlsag med motioner och propositioner (se Kap 2.6). Den sista nivån visar en motion eller proposition som användaren kan läsa i dess helhet. Genom hela hierarkin kan man använda Back- eller Up-knappen för att återvända till den föregående vyn.

Det första man möts av då man startar appen är en väldesignad inloggningsida där man kan logga in eller gå till en annan vy för att registrera sig. Icke-modal specifik feedback ges om lösenordet är för kort eller användarnamnet redan finns. Då programmet laddar något syns det idiomatiska snurrande hjulet.

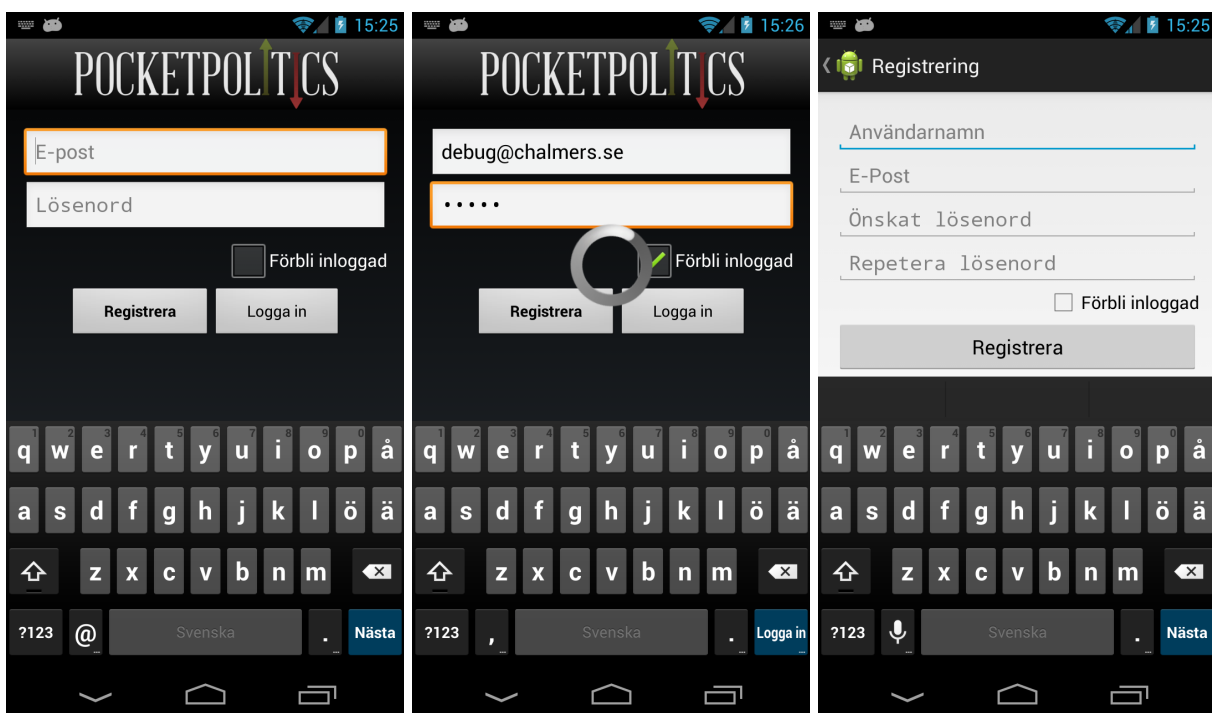


Fig 1: inlogging och registrering.

Applikationens huvudsida består utav en lista som visar olika betänkande ifrån Riksdagen, se fig 2. Listan laddas med de femton nyaste beteckningarna och sorteras med nyast överst. Varje objekt går att interagera med så att man byter vy till just det betänkandet.

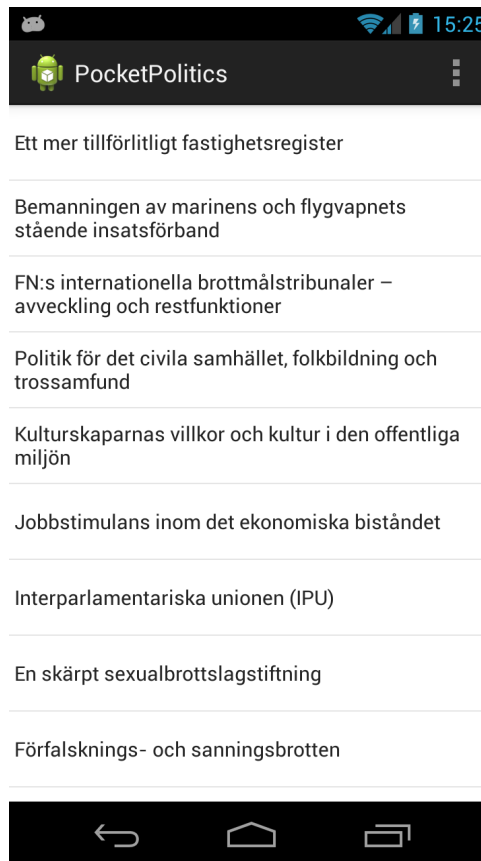


Fig 2: Applikationens huvudsida.

Betänkandevyn innehåller titel, vilket utskott som lagt betänkande och en kort sammanfattning utav betänkandet i fråga. Under texten finns en lista som innehåller olika utskottsförslag som vardera har knappar för att navigera till sina respektive motioner och propositioner.

Motioner och propositioner innehåller titel, författare och fullständig text.

## 4.2 Från ett utvecklarperspektiv

Resultatet är en applikation som hämtar och tolkar data asynkront från Riksdagens officiella öppna databas. Riksdagens databas returnerar XML-formaterad information som tolkas med hjälp av det resurssnåla biblioteket kXML (se Kap 2.2). För att inte överbelasta telefonens minne, batteri eller uppkoppling hämtas informationen i flera nivåer. Endast det användaren begär att se hämtas.

Appen ansluter till en dedikerad SQL-databas via ett interface av PHP på en server som kontrolleras av gruppledarna. Denna databas lagrar all användar-specifik data exempelvis email och lösenord. Lösenord hanteras med saltad hast vilket håller dem säkra även om databasen skulle infiltreras (se Kap 2.5). Servern returnerar alltid ett Json-objekt som tolkas med hjälp av biblioteket Gson (se Kap 2.3).

Koden är strukturerad i två robusta lager: ett för MVC och ett för nätverksoperationer. Nätverksoperationerna är helt oberoende av MVC och kan användas till vad som helst. Själva klasserna i sig är oåtkomliga utifrån, all kommunikation styrs av en interface-klass mot MVC-lagret. Alla nätverksoperationer sker asynkront på en egen tråd för att inte GUI:t ska störas och användaren bli lidande. Multitrådningen sköts av Android biblioteket med hjälp av AsyncTask (se Kap 2.1). Det finns mycket stöd för att lägga till nya nätverksklasser vid behov eller byta ut de nuvarande i och med att de hanteras genom interface-klassen.

Viss funktionalitet är fullständigt implementerad med undantag för dess grafiska representation. Det finns stöd i kontroller- och nät-klasserna för både filtrering och sökning men det saknas i GUI:t. Likaså finns det stöd för att kommentera betänkanden, gilla och ogilla betänkanden, motioner samt tidigare kommentarer.

## 5 Diskussion

### 5.1 Projektets gång

Projektets mål har behövts justeras under applikationens utveckling. Ökad domänkunskap om hur Riksdagen arbetade och hur dess databas var uppbyggd ledde till vissa förändringar i programmet. Projektets avgränsning har också omformulerats flera gånger på grund av tidsbrist, till stor del beroende på att arbetskraften var mindre än beräknad, och ökad insikt i vad som är viktigast. Dessa saker kunde inte ha planerats i förväg utan kom som en följd av en ökad förståelse för problemet. Därmed fick utvecklingen stor fördel av att använda en agil process (se Kap 3) jämfört med vattenfallsmetoden då allt är planerat på förhand. I och med iterationerna och den agila processen är också de viktigaste funktionerna implementerade medan flera av de ursprungliga tänkta men mindre viktiga funktionerna enbart har stöd eller lämnats helt oimplementerade.

Att hålla regelbundna möten med protokoll visade sig vara väldigt givande. Detta gjorde att de närvarande gruppmedlemmarna kunde enas om viktiga beslut och hålla ett gemensamt fokus så alla drar åt samma håll. Mötena skapade också en god enighet bland de närvarande. Utöver mötena hölls också regelbunden kommunikation via sociala medier.

Tyvärr var en gruppmedlem ofta frånvarande och deltog knappt i någon annan kommunikation med de övriga gruppmedlemmarna. Detta skapade en barriär mellan de närvarande och den frånvarande vilket ledde till missförstånd och stor frustration. Den frånvarande kunde därför inte vara insatt i projektets utveckling och de ansträngningar som utfördes av denne var således sällan i harmoni med gruppens arbete.

### 5.2 Projektets framtid

Koden är flexibel och utökningsbar till att uppfylla all funktionalitet som data.riksdagen.se delar ut, detta kan till exempel röra sig om webb-tv, fullständig visning av debatt eller prenumeration på enskilda politiker. Dessvärre kommer applikationen slutligen att begränsas av mängden information som finns på riksdagens databas. Genom att införa egna index i SQL-databasen kan man utöka denna ytterligare något med hjälp av egna sökfilter så länge det finns data att indexera.

En annan begränsning som Riksdagen medför ligger i domänen: det är inte säkert att andra regeringar har samma struktur på sina ärenden. Detta innebär att det inte bara går att återanvända samma GUI till andra länder eller nivåer som kommunfullmäktige. Därmed är det inte sagt att det är omöjligt, men det skulle kräva mycket ny kod trots designen kring extensibility (se Kap 3.3).

Ett annat utökningsområde är det sociala fältet: mer stöd åt kommentarer, användarstatistik etc som är vanliga på internetforum. Mycket av detta var från början tänkt att finnas med men avgränsades senare på grund av tidsbrist.

## 6 Slutsats

Projektets mål, en fullständig applikation, uppfyller behovet som presenteras i inledningen. Den utvecklade appen är en prototyp av målet med grundläggande funktionalitet. Projektet har visat att tekniken och informationen finns. Med mer tid skulle applikationen kunna bli ett centralt verktyg för individer att ta del av aktuella beslut som just de är intresserade av och samla statistik på vilket parti och vilken enskild politiker som bäst förtjänar deras röst i Riksdagen.

## Källförteckning

Android Developer (2013a) Parsing XML Data

<http://developer.android.com/training/basics/network-ops/xml.html>

(2013-05-20)

Android Developer (2013b) Application Fundamentals

<http://developer.android.com/guide/components/fundamentals.html>

(2013-05-21)

Android Developer (2013c) Processes and Threads

<http://developer.android.com/guide/components/processes-and-threads.html>

(2013-05-21)

Defuse Security (2013) Salted Password Hashing - Doing it Right

<http://crackstation.net/hashing-security.htm>

(2013-05-20)

GlobalWebIndex (2012) Global Mobile Adoption: Android 54% of global mobile users

<http://www.globalwebindex.net/global-mobile-adoption-android-54-of-global-mobile-users/>

(2013-14-19)

Google Code (2013) google-gson: A Java library to convert JSON to Java objects and vice-versa

<http://code.google.com/p/google-gson/>

(2013-05-21)

Haustein, Stefan (2005) About kXML

<http://www.kobjects.org/kxml/index.php>

(2013-05-20)

InfoWorld (2011) Study finds high rate of password reuse among users

<http://www.infoworld.com/t/data-security/study-finds-high-rate-password-reuse-among-users-188>

(2013-05-20)

JSON (2013) JSON: The Fat-Free Alternative to XML

<http://www.json.org/xml.html>

(2013-05-21)

Meggison, David (2007) All markup ends up looking like XML

<http://quoderat.meggison.com/2007/01/03/all-markup-ends-up-looking-like-xml/>

(2007-01-03)

Microsoft (2013) Chapter 1: What is Software Architecture?

<http://msdn.microsoft.com/en-us/library/ee658098.aspx>

(2013-05-23)

Mobil Sverige (2012) Mobilens roll i mediamixen H1 2012 - En studie i reklameffekt

<http://www.mobil Sverige.se/>

(2013-14-19)

Ohlsson, Jonas (2010) Spelar medieägandet roll?

[http://www.utbildning.gu.se/digitalAssets/1316/1316416\\_141-154-jonas-ohlsson.pdf](http://www.utbildning.gu.se/digitalAssets/1316/1316416_141-154-jonas-ohlsson.pdf)

Oscarsson, Henrik & Holmberg, Sören (2011) Swedish Voting Behavior

[http://www.valforskning.pol.gu.se/digitalAssets/1357/1357064\\_report-2011-4-swedish-voting-behavior.pdf](http://www.valforskning.pol.gu.se/digitalAssets/1357/1357064_report-2011-4-swedish-voting-behavior.pdf)

Skrien, Dale (2007) Object-Oriented Design using Java, 1st edition. Sid. 5-6.

New York: McGraw-Hill.

ISBN 978-0-07-297416-4

Riksdagen (1974) Kungörelse om beslutad ny regeringsform

[http://www.riksdagen.se/sv/Dokument-Lagar/Lagar/Svenskforfattningssamling/Kungorelse-1974-152-om-beslu\\_sfs-1974-152/](http://www.riksdagen.se/sv/Dokument-Lagar/Lagar/Svenskforfattningssamling/Kungorelse-1974-152-om-beslu_sfs-1974-152/)

(2013-14-19)

Riksdagen (2011) Så arbetar utskotten

<http://www.riksdagen.se/sv/Sa-funkar-riksdagen/Sa-arbetar-utskotten/>

(2013-05-20)

Roosvall, Anna (2005) Utrikesjournalistikens antropologi

<http://su.diva-portal.org/smash/record.jsf?pid=diva2:192743>

<http://su.diva-portal.org/smash/get/diva2:192743/FULLTEXT01>

Walsh, Norman (1998) A Technical Introduction to XML

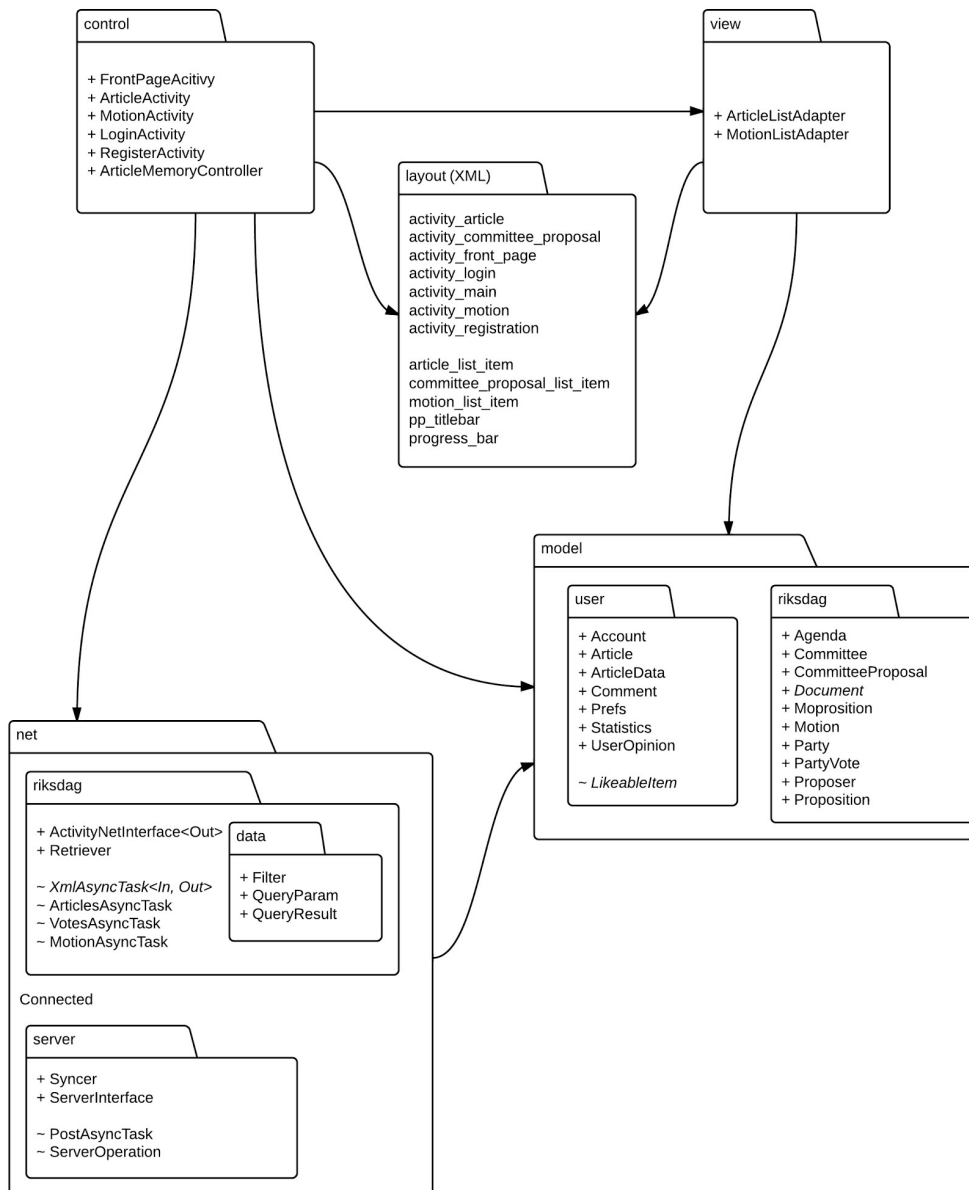
<http://www.xml.com/pub/a/98/10/guide0.html>

(1998-10-03)

## Bilaga: UML-diagram

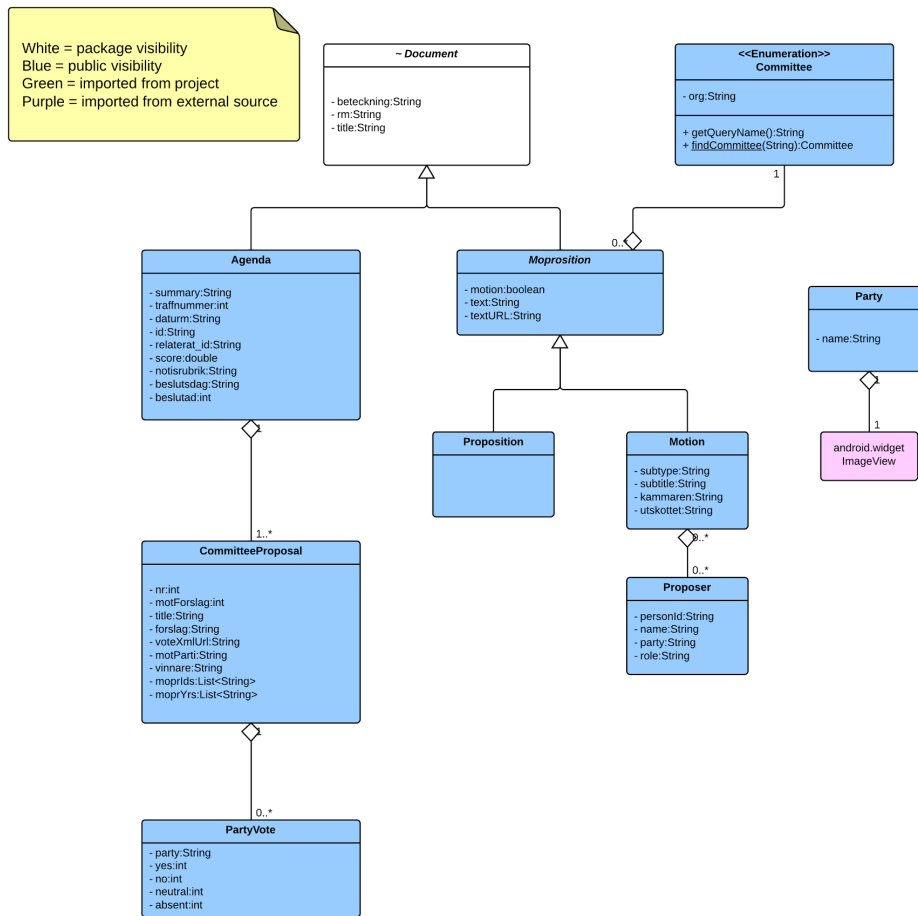
Denna bilaga innehåller diagram över koden.

### Paketdiagram

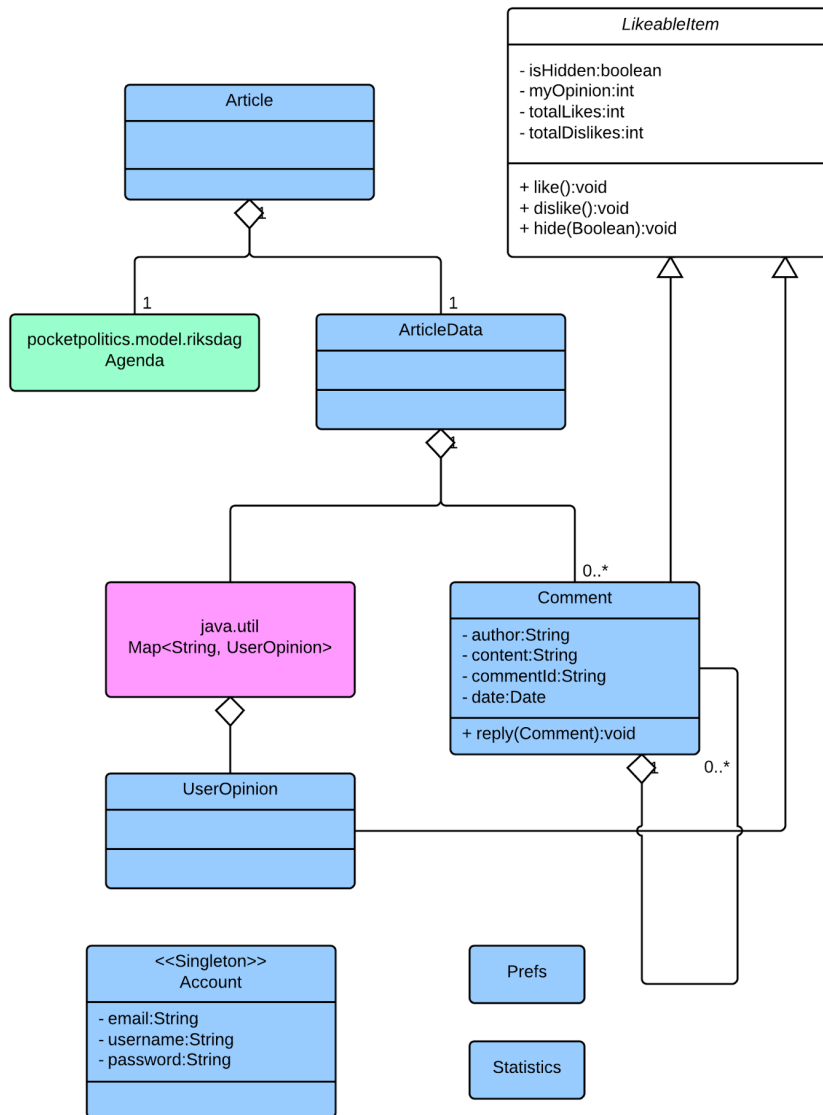




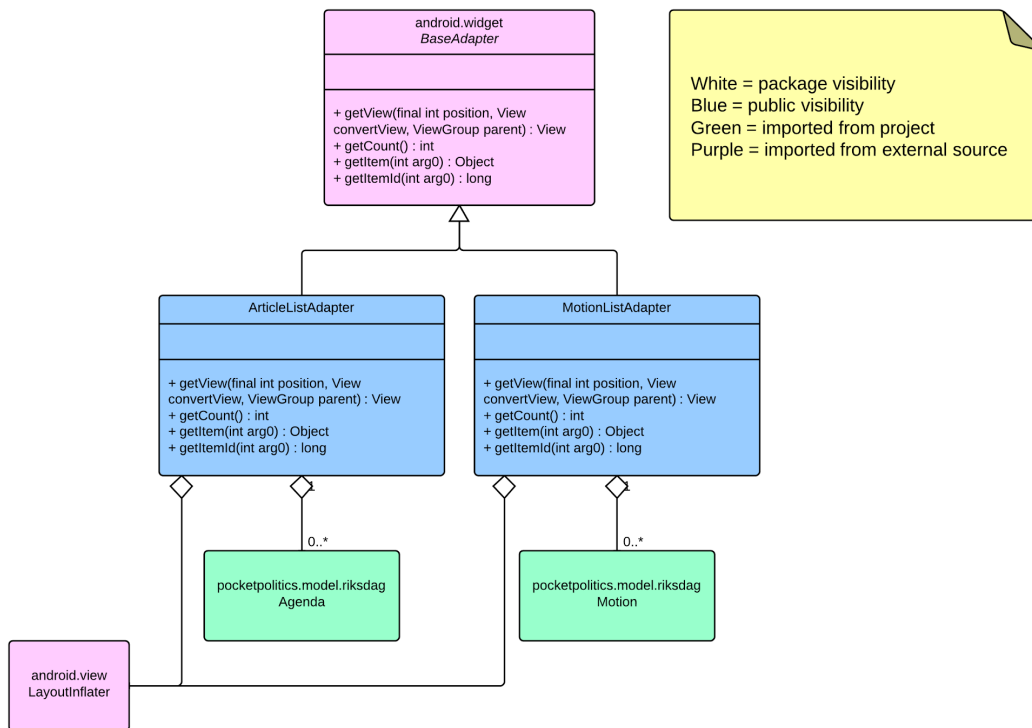
## Klassdiagram: model.riksdag



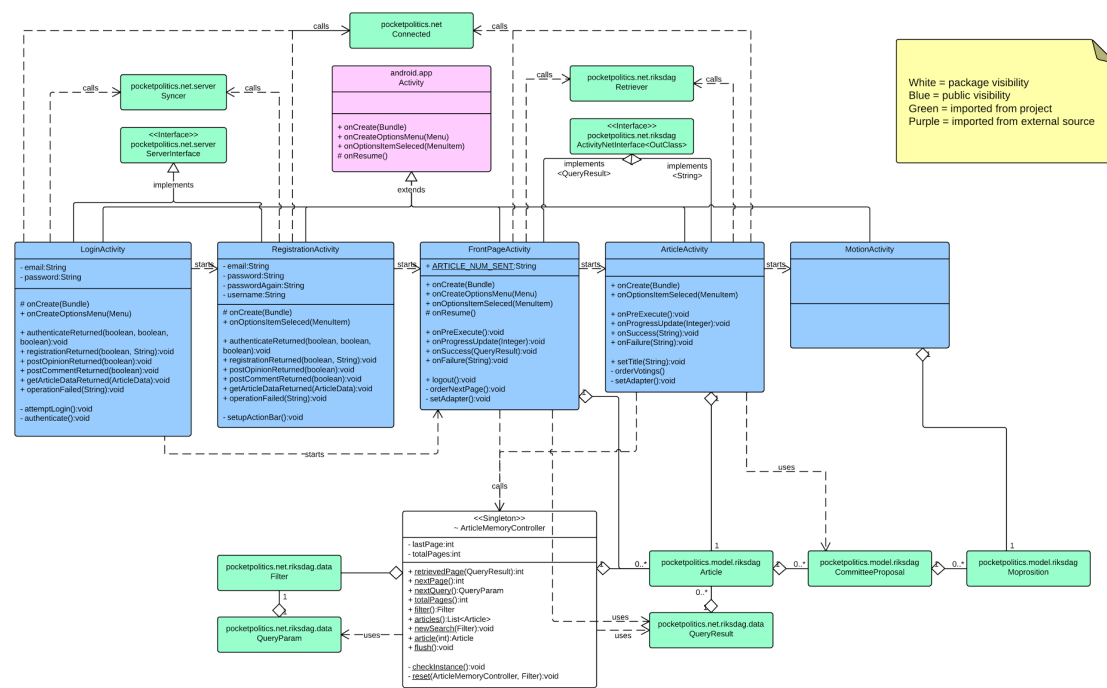
## Klassdiagram: model.user



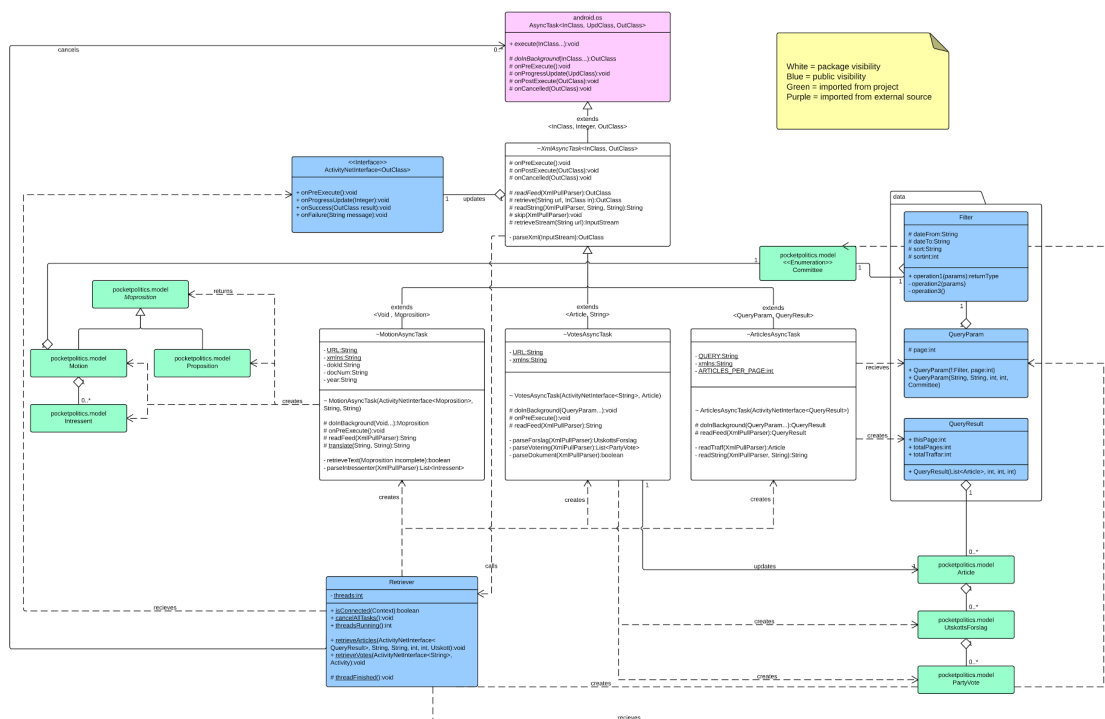
## Klassdiagram: view



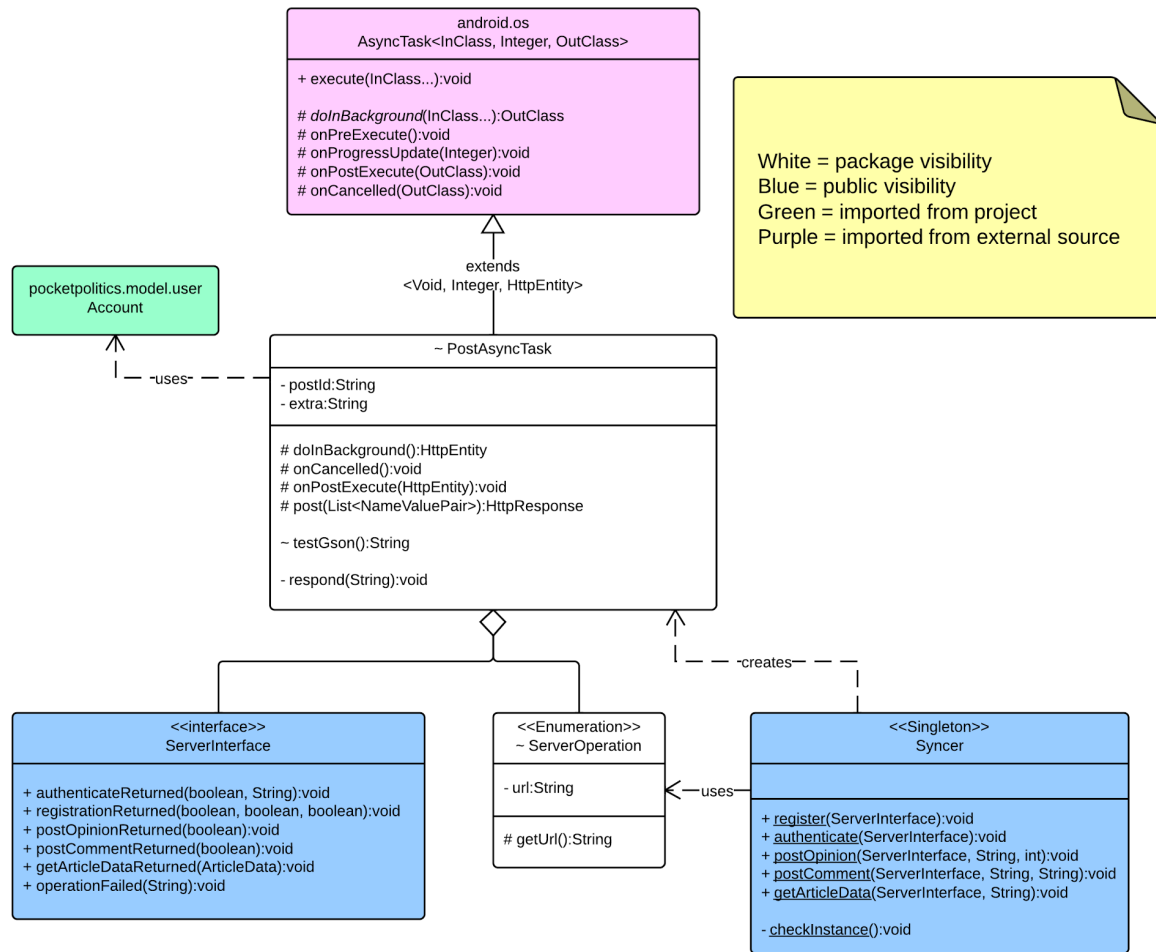
### Klassdiagram: control



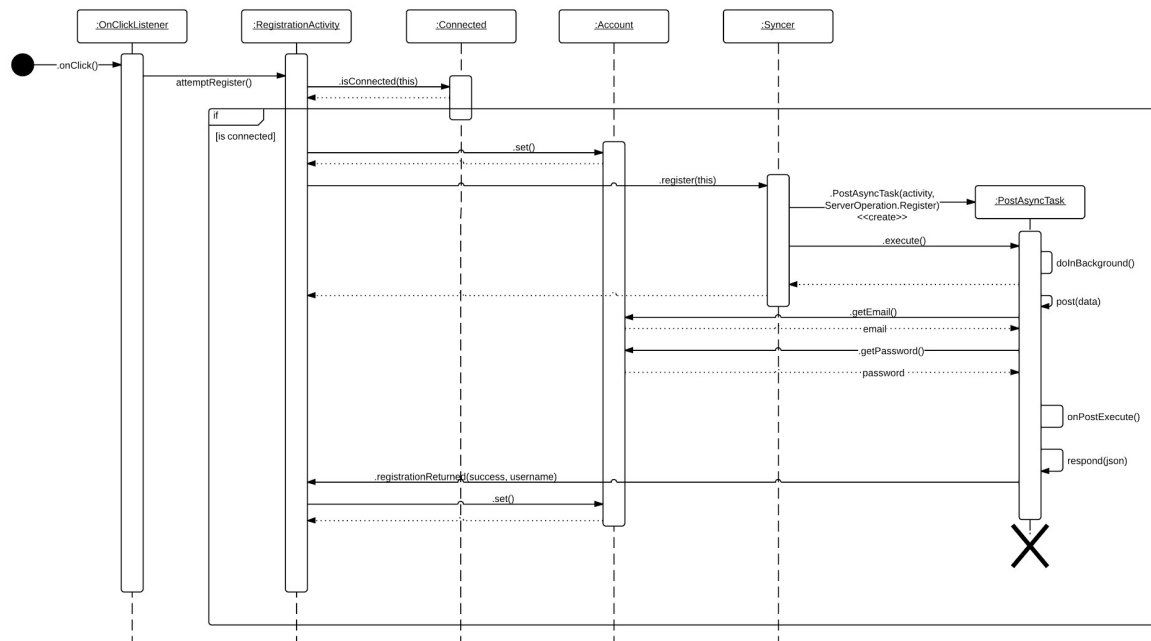
## Klassdiagram: net.riksdag



## Klassdiagram: net.server



## Sekvensdiagram: registrera sig



## Sekvensdiagram: öppna en artikel

