Philip Gregorius 920703-4290
Elias Ulén 920826-4688
Alexander Sjöman 890821-6271
Ebba Gullstrand 920629-5744

Bakgrund

Som en del i utbildningen på det systemvetenskapliga programmet vid lunds universitet har vi fått i uppgift att utveckla webservices och klienter som konsumerar dessas funktioner. Projektet ingår som en del av den fjärde terminen.

Syfte

Utforma webservices och klienter som i första uppgiften skulle kunna hämta och läsa filer som ligger i webservice projektet. Webservicen ska utöver kunna läsa av filerna och skicka tillbaka innehållet kunna hämta data från en tidigare egen designad databas.

I Erp-integration skall webservicen hantera cronus databas och kunna ändra, hämta och ta bort ur databasen från en klient.

Projektbeskrivning

Projektet består av klienter skrivna i java och c# dessa pratar med varsin av två olika webservices skrivna i c# .

Systembeskrivning

Projektet innefattar två webservices som konsumeras av klienter, där den första kan ta emot filnamn och skicka tillbaka filens innehåll till den anropande klienten. Samt hämta information från vår egen designade databas tillhörande fitness-systemet igualis fitness buddy. Databasen innehåller två tabeller Account och schedule.

Denna webservice används av tre klienter. En java applikation med ett grafiskt gränssnitt, en c# konsol applikation med anslutning genom wsdl.exe och en c# applikation med grafiskt användargränssnitt.

Den andra webservicen behandlar information från Cronus-databasen ur tabellen employee. Webservicen konsumeras av två klienter med grafiskt användargränssnitt skrivna i java och c#. Dessa kan hämta, uppdatera, infoga och ta bort användare samt hämta metadata om tabellen.

Philip Gregorius 920703-4290
Elias Ulén 920826-4688
Alexander Sjöman 890821-6271
Ebba Gullstrand 920629-5744

Kravspecifikationer

Webbservices

- Konsolapplikation där proxyn är skapad med hjälp av wsdl.exe
- Windows-applikation
- Java-applikation
- Ska kunna skicka fil som in parameter och läsa ut innehållet på klient sidan.
- Ska kunna hämta data från en databas skapad tidigare under terminen och skriva ut en collection från den.

Erp-Integration

Webservice som innehåller följande funktioner select,insert,update och delete. För att konsumera dessa metoder ska de skrivas två klienter.

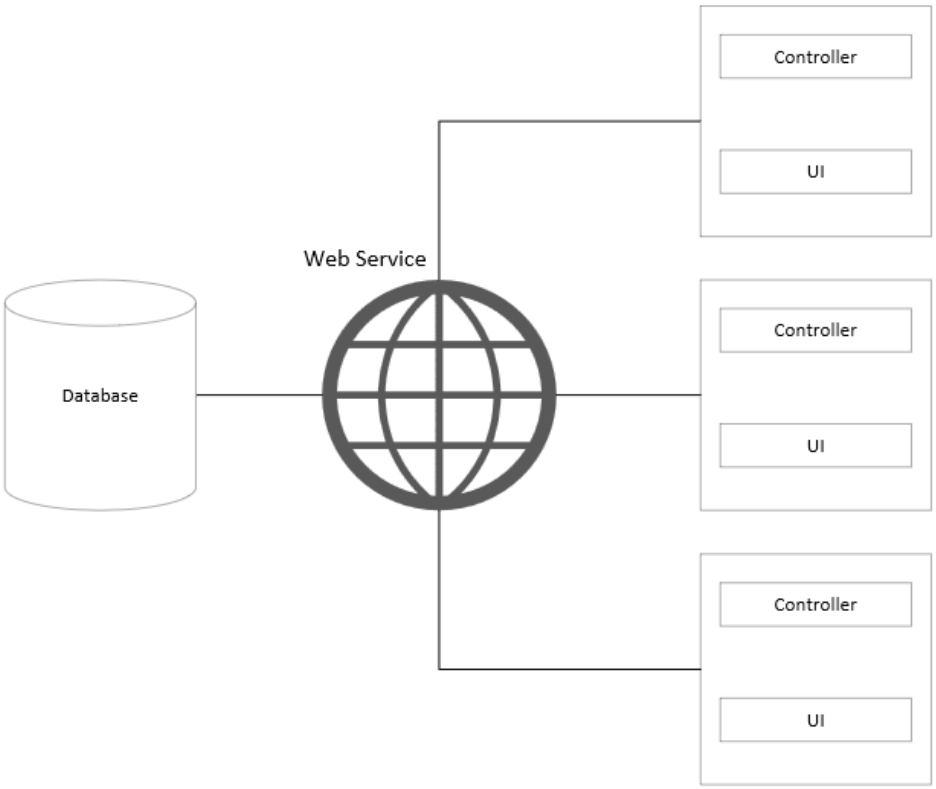
- Windows-applikation
- Java-applikation

Dessa klienter ska förutom att kunna konsumera de tidigare nämnda funktionerna, kunna anropa webservicen och skriva ut följande information.

- Information om Personal och deras släktingar (Personalanhörig)
- Information om anställda som har varit borta pga sjukdom år 2004 (Personalfrånvaro)
- First name på anstallda som har varit mest sjuka
- Innehållet och metadata i Employee (Personal) och relaterade tabeller
- Alla nycklar
- Alla indexes
- Alla table_constraints
- Alla tabeller i databasen (2 lösningar)
- Alla kolumner i tabellen Employee (2 lösningar)

Philip Gregorius 920703-4290
Elias Ulén 920826-4688
Alexander Sjöman 890821-6271
Ebba Gullstrand 920629-5744

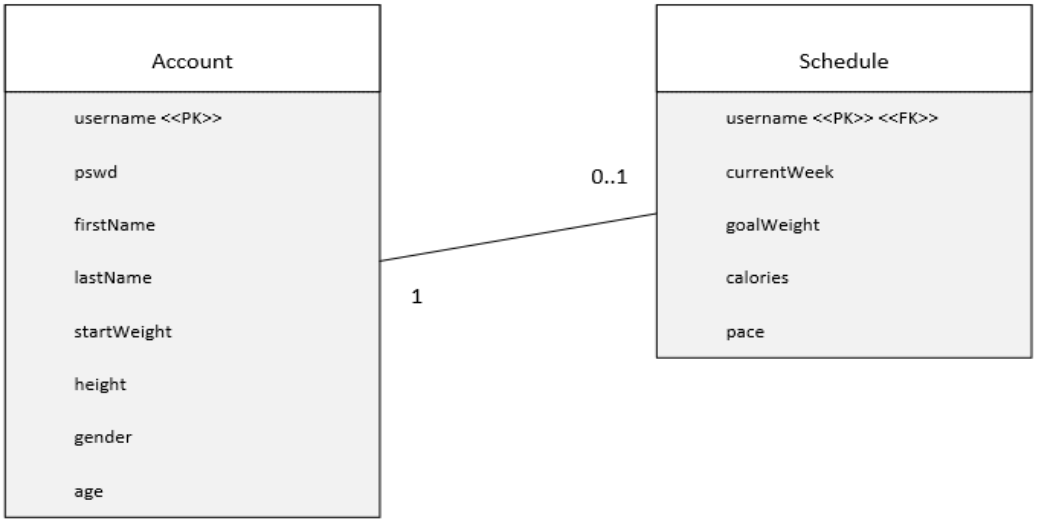
Modeller



Grundläggande modell över webservice

De olika applikationerna kan prata med webservicen och komma åt dess funktioner. Webbservicen har en sammankoppling med en databas, alla applikationer delar således denna databas för servicens funktioner. Klienterna med grafiskgränssnitt och konsol applikationen ett user-interface och till det en controller kopplad som hanterar anrop till och från webservicen.

Philip Gregorius 920703-4290
Elias Ulén 920826-4688
Alexander Sjöman 890821-6271
Ebba Gullstrand 920629-5744

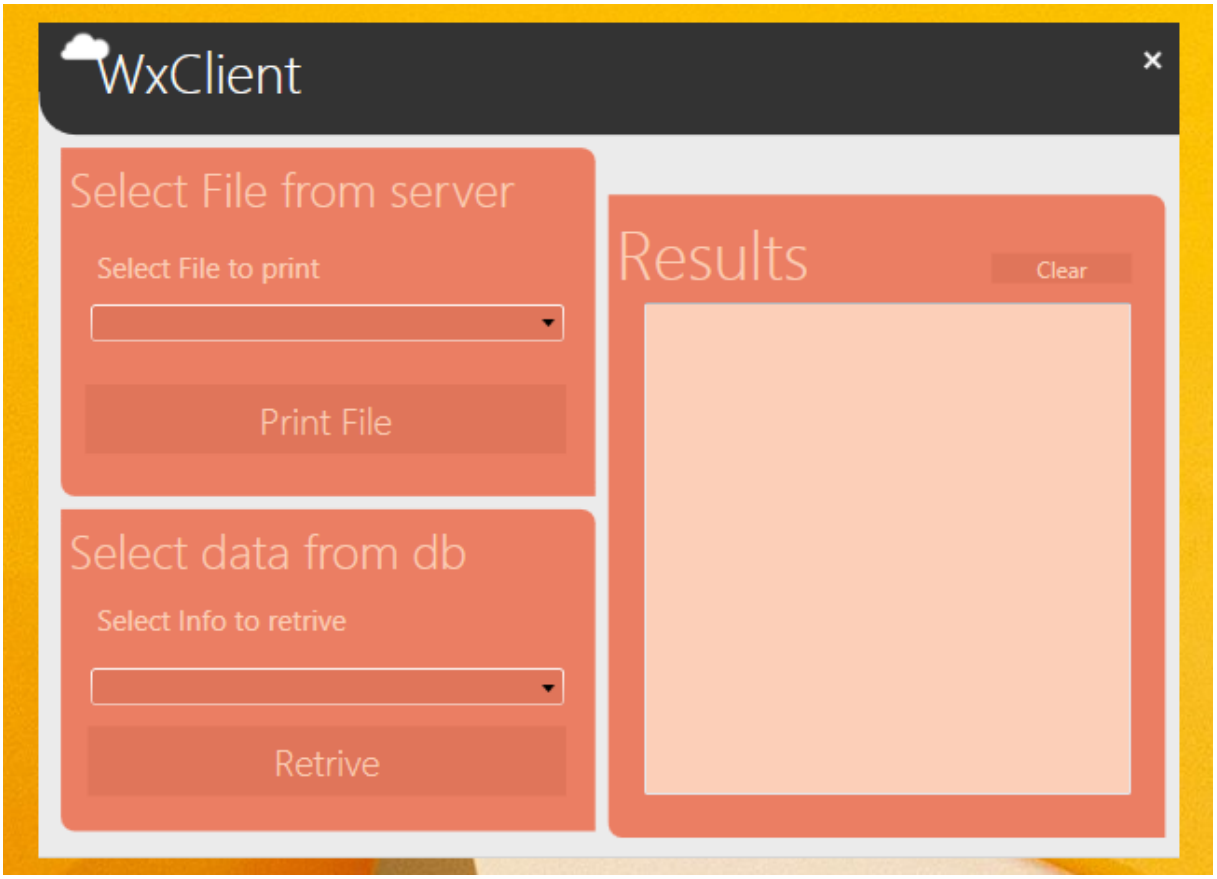


Egen designad databas

Account kan ha ett schedule men ett schedule måste ha ett account.

Philip Gregorius 920703-4290
Elias Ulén 920826-4688
Alexander Sjöman 890821-6271
Ebba Gullstrand 920629-5744

Mycket korta beskrivningar av varje moduls funktion



Gränssnitt för webservice, klient i c#

Print File

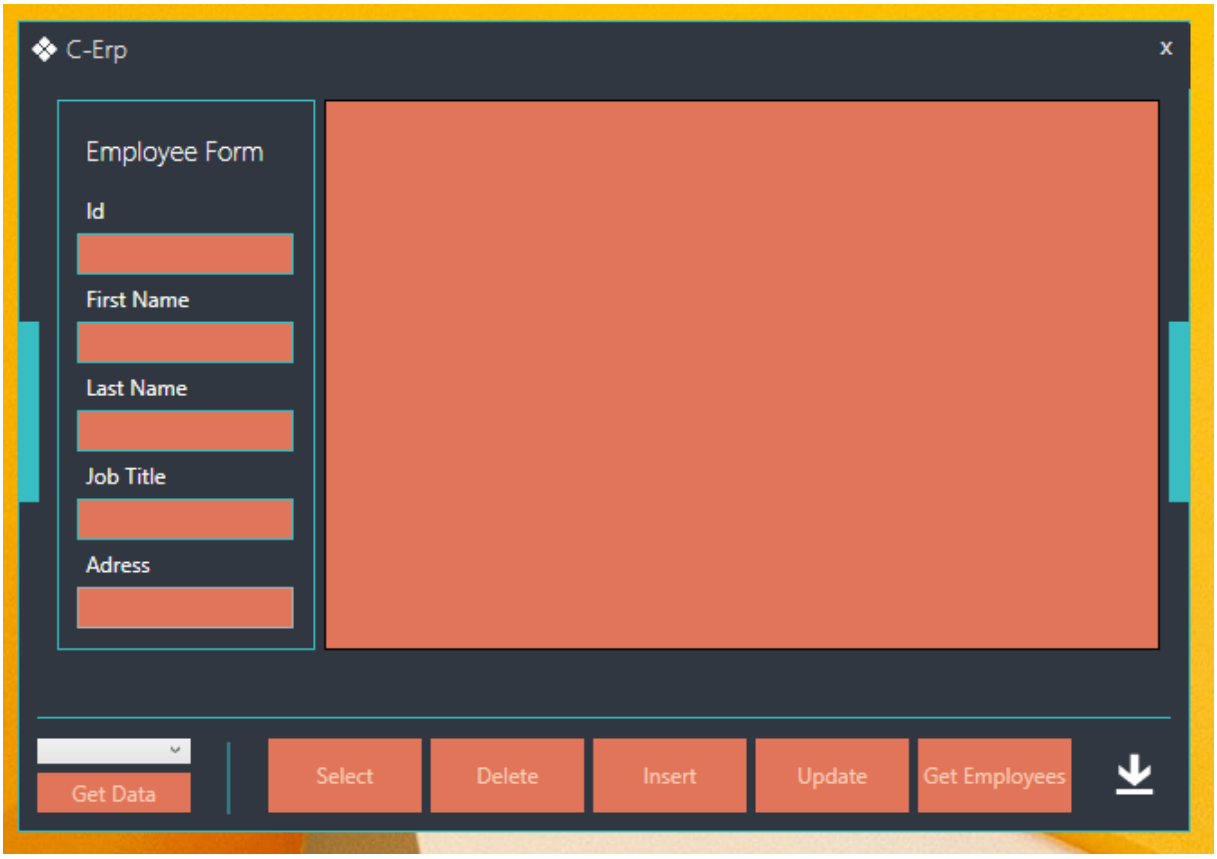
Printar vald fil genom att hämta filnamn från ovanstående combobox, boxen fylls från webservicen genom en funktion som söker igenom vilka filer som finns. Filen tas sen som inparameter till funktion som hämtas från comboboxen. Därefter skrivs innehållet ut under Results.

Retrive

Välja vad som ska hämtas från databasen en collection av resultatet visas under Results.

Clear

Tömmer Results innehåll.



Gränssnitt för Erp-integrations webservice klient i c#

Get Data

Beroende på valt sök värde ur ovan stående combobox väljer knappen vad som ska sökas genom att anropa två metoder från webservicen.

- GetColumnNames(); För att kunna hämta namnen på columnerna.
- Get *** Data(); Beroende på vilken query som man valt.

Select

Kräver att Id är registrerat från Employee Form och skickar detta sen till Select funktionen i webservicen.

Delete

Kräver att Id är registrerat från Employee Form och skickar detta sen till Delete funktionen i webservicen.

Insert

Kräver att alla fält i Employee Form är inmatade och skickar det till webservicen som utför query

Update

Kräver att alla fält i Employee Form är inmatade och skickar det till webservicen som utför query

Get Employee

Hämtar alla employees i databasen genom webservicen.

Gridview

Vid alla anrop som vill ha data tillbaka fylls gridviewn med informationen, detta görs genom funktionen DataGridFill. Den tar emot en lista i lista och bygger en datamodel av detta som sedan sätts som GridViewns datasource.

En beskrivning av den hårdvara och mjukvara som använts i praktikfallet

Alla Webservice och C#-klienter är skrivna på Microsofts .NET plattform genom Visual Studio 2013, på Windowsplattformen Windows 7 och 8 (Primärt Windows 8). Java-klienterna är skrivna med hjälp av eclipse kepler.

Buggar och problem

- När man hämtar alla employees visar timestamp sys.byte.

- IIS låter inte webservicen ansluta till databasen trots försök att skapa ny användare eller ändra inställningar i IIS. Endast testat med windows 8 problemet gäller både uppgift 1 och 2, vanliga funktioner som inte kräver anslutning till databasen funkar dock. Funktionerna funkar dock när man kör servicen genom visual studios temporära server.