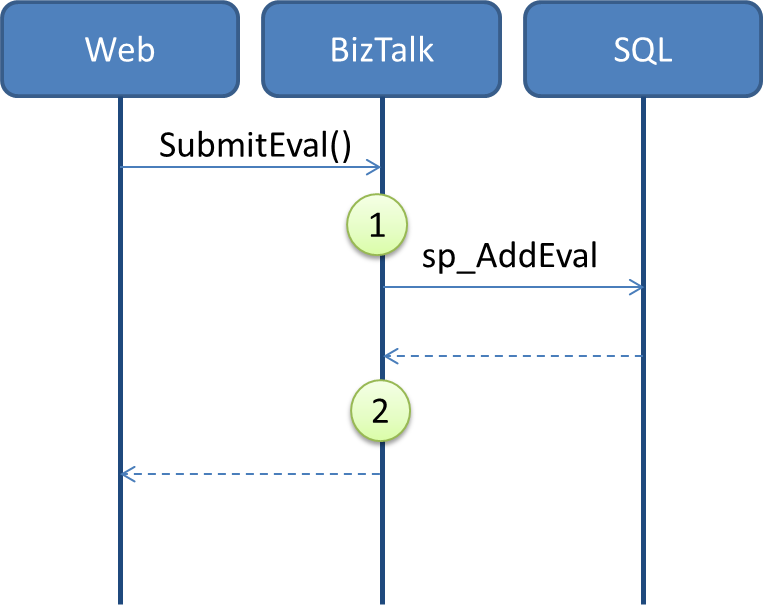
# Course Eval Application

## Sammanfattning

Du ska hjälpa AddSkills med att lösa integrationen för deras nya kursutvärderingssystem. Så här långt in i projektet har man tagit fram en Web applikation, i vilken man kan göra utvärderingar, och en databas dit alla utvärderingar sparas. Din uppgift är att lösa integrationen mellan dessa.



### Verksamhets regler:

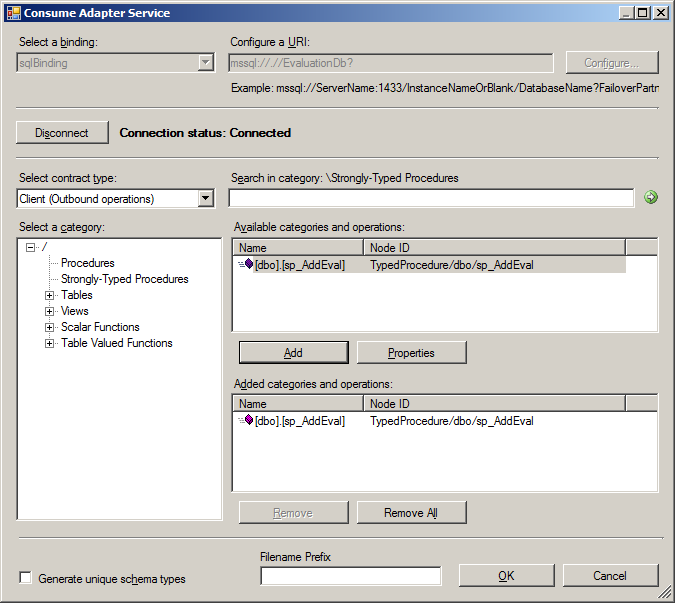
1. Utvärderingar av kurs **A372** ska alltid vara bästa möjliga (5), oavsett vad användaren angivit (vi utgår från att användaren i så fall inte förstod frågan).
2. Svaret tillbaka till web applikationen ska vara ”Tack [studentens *namn*]. Din utvärdering är sparad”.
3. Svaret från den lagrade proceduren som du skall anropa för att spara utvärderingen, behöver inte användas i returen till web applikationen.
4. Lämpliga utökningar eller funktionalitet att lägga till (om du har tid):
   * Komplettera processen med att ett mail skickas till [expenses@bts2010.com](mailto:expenses@bts2010.com) om utvärderingen understiger värdet 3.
   * Utför verksamhetsregel 1 med hjälp av BRE.
   * Tracka lösningen med BAM.
   * För in Scope för felhantering så att man kastar ett fel med throw exception shapen om Rating är 1 och fångar det i ett exception handling block där man skickar ett svar till användaren i form av ”Systemfel, 1 är ett för lågt värde”, och därefter avslutar orkestreringen.

### Arbetsordning

1. Skapa SQL scheman
2. Applicera logiken i din orkestrering (ProcessEvals.odx)
3. Deploya din solution
4. Publicera orkestreringen som en WCF tjänst
5. Testa applikationen genom att trycka F5.

## Skapa SQL schemas och Send Port

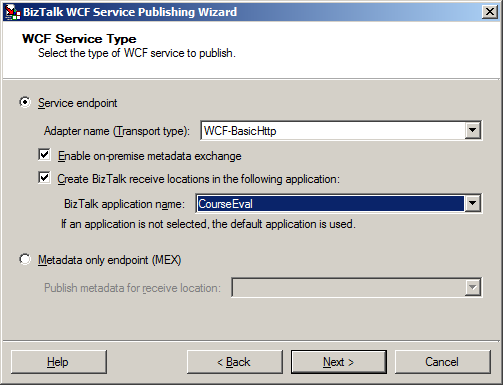
1. Högerklicka på **Eval.Orchestrations** projektet och välj *Add->Add Generated Item*
2. Välj *Consume Adapter Service,* klicka på *Add.*
3. I dropplisten *Select a binding* välj ***sqlBinding****.* Klicka på *Continue*.
4. På Security fliken välj *Windows* som Credential Type.
5. Välj fliken *URI Properties*. Sätt *InitialCatalog* till **EvaluationDb** och *Server* till **localhost**.
6. Klicka *Ok* för att stänga fönstret, och sen på *Connect*.
7. Välj *Strongly-Typed Procedures* i listan av kategorier. Välj sedan *[dbo].[sp\_AddEval]*. Avsluta med att klicka på *Add* och sedan på *Ok.*



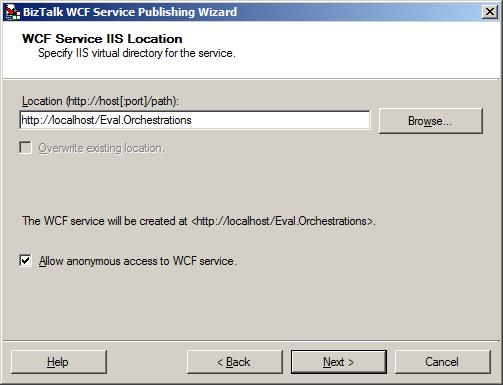
1. Du ska nu ha två nya filer i din solution; en schema fil och en bindings-fil.
2. Bindings filen ska du importera till **CourseEval** applikationen för att skapa och konfigurera din send port (efter att ha deployat din solution). Det kan vara en bra idé att sätta *Retry count* (*Transport Advanced Options)* till 0 under tiden man utvecklar (i annat fall måste du vänta i 3\*5 min om det går fel).
3. Schema filen innehåller request- och response scheman du ska använda i din orchestrering för kommunikation via SQL adaptern.

## Publicera Orkestrering som WCF tjänst

I Lab 8 publicerade du en orkestrering som WCF tjänst. Denna gång ska du göra samma sak fast genom att låta *BizTalk WCF Publishing Wizard* skapa porten och dess location (I Lab 8 skapade du porten manuellt via BizTalk Administration Console, och använde wizarden för att endast skapa WSDL’en). För att göra detta klickar du för ”*Create BizTalk receive loactions in the following application*” enligt skärmdumpen nedan. Välj **basicHttpBinding** som *Transport Type.*



Klicka även för ”*Allow anonomus access to WCF service”…*



När du är klar så har tjänsten skapats i en IIS Application som heter **Eval.Orchestrations**. Glöm inte att byta *Application Pool* till **BAMAppPool** (Öppna IIS Managern och högerklicka på Eval.Orchestrations. Välj *Manage Application* -> *Advanced Settings…)*

Om du gjort allt rätt så ska du kunna nå din WSDL på denna adress:

<http://localhost/Eval.Orchestrations/Eval_Orchestrations_ProcessEvals_EvalService.svc>