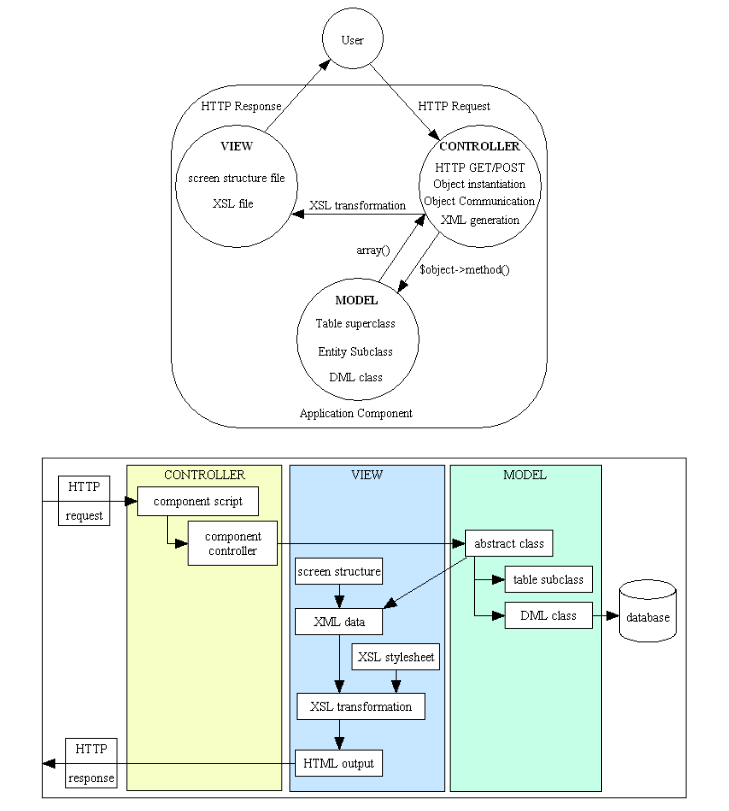
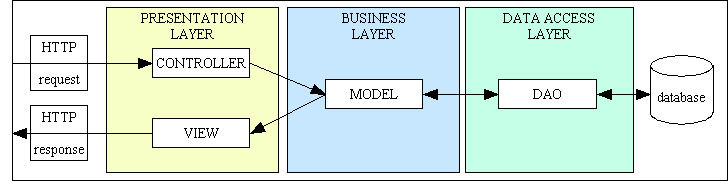
**Exercise 2: MVC Model Framework using XML**



**Requirements: develop a use case for user management (including add, delete, update and search) for administrators.**

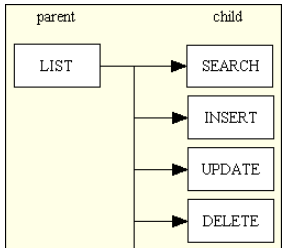
# Request/Response cycle with MVC



* The Controller receives the HTTP request from the client
* The Controller calls the appropriate mythos on the Models and waits for them to finish. Remark that the results contain only data, no HTML.
* If the Model wishes to touch the database it does so by communicating with the DAO
* The Controller calls the View with the results from the Models in order to transform the raw data into HTML
* The Controller sends the HTML response back to the client

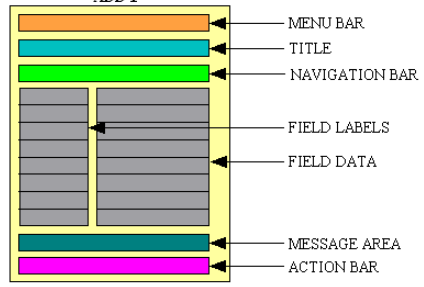
# Usecase for the user management

* A user transaction (which may or may not include a database transaction) is a distinct unit of work (also known as "Use Case") that may be selected by the user from an option on a screen.
* The figure below shows a basic family of forms that perform the basic maintenance tasks on a database table

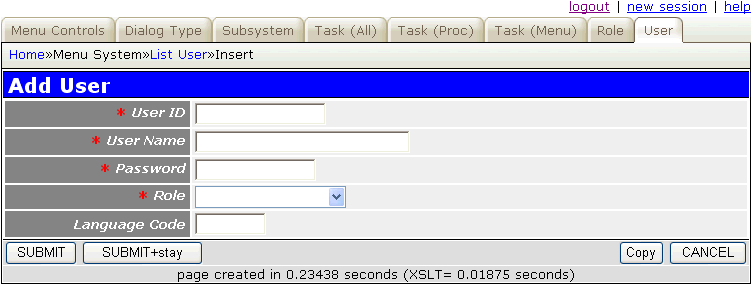


# Add

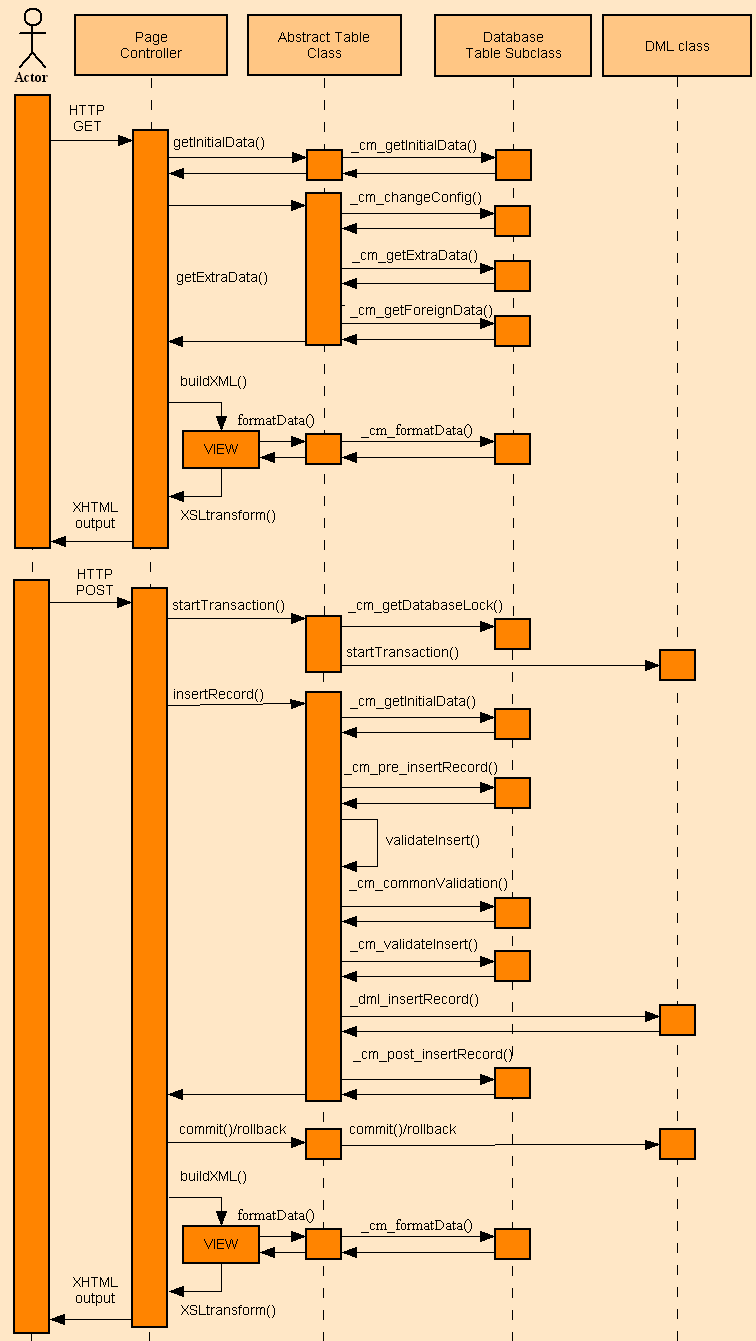
* Purpose: add 1 new row at a time to the database
* Structure



* Explanation
  + Menu bar: show the menu options which led to this screen
  + Title: This is supplied at runtime
  + Navigation bar: This is usually empty for this type of function
  + Data area: As this area is for a single occurrence the data will be shown vertically, one field per line, with data on the right and labels on the left.
  + Message area: any messages or instructions.
  + Action bar:
    - Cancel: terminate the current function without performing any updates and return to the previous function.
    - Submit: submit all data to the database then return to the previous function
    - Submit + stay: submit all data to the database, then initialize the screen ready for new input.
    - Copy: copy the data to memory
    - Paste: fill the screen with data from the memory area.
* Screenshot:

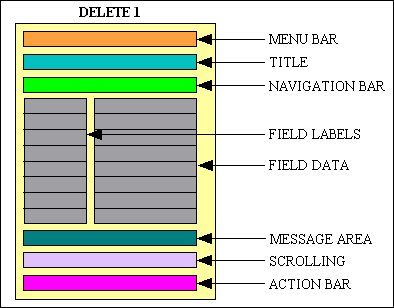


* UML:

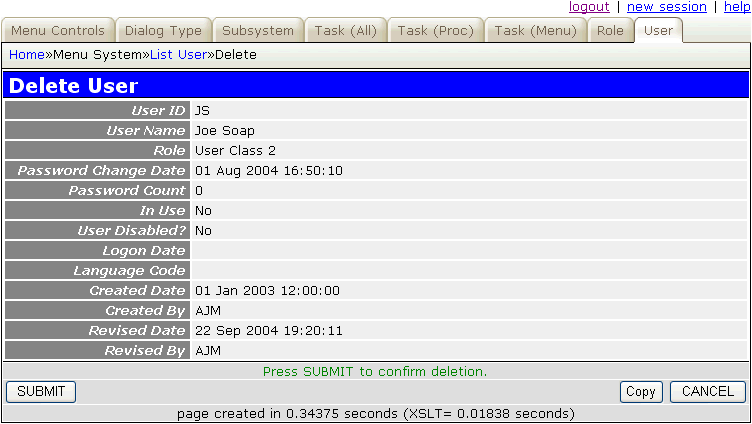


# Delete

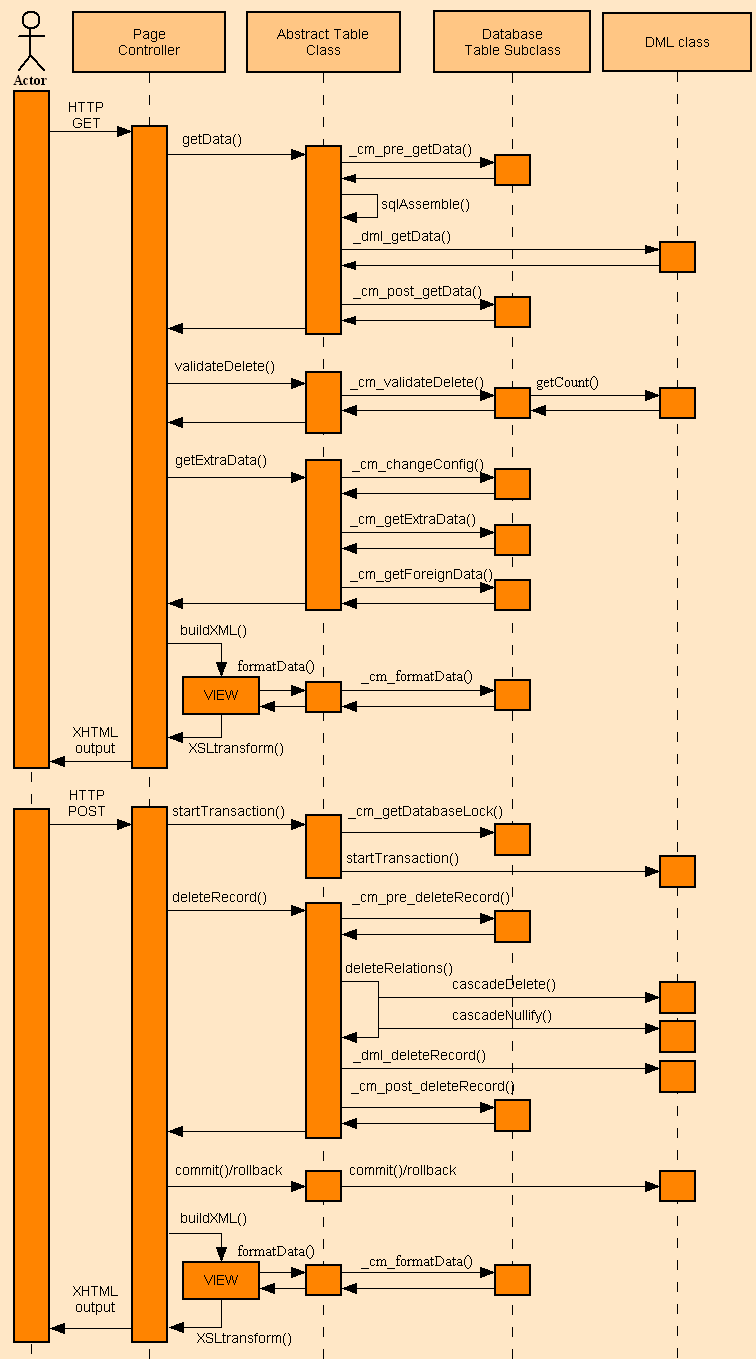
* Purpose:
  + retrieve a record from the database using the primary key(s) passed down by the previous form
  + check if any subordinate entries exist on any related database tables which will prevent the deletion from continuing
  + If there are any such constraints then an error message will be generated.
* Structure



* Detail about button: <http://www.tonymarston.net/php-mysql/dialog-types.html#delete1>
* Screenshot

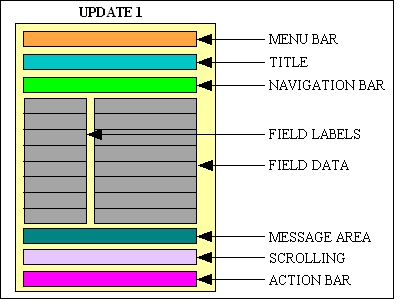


* UML

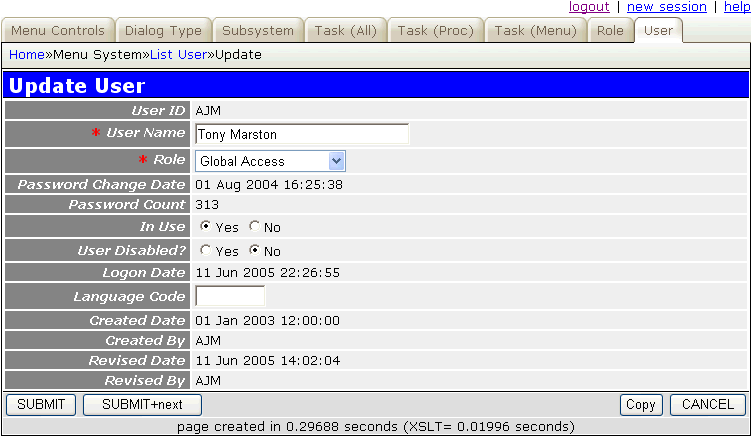


# Update

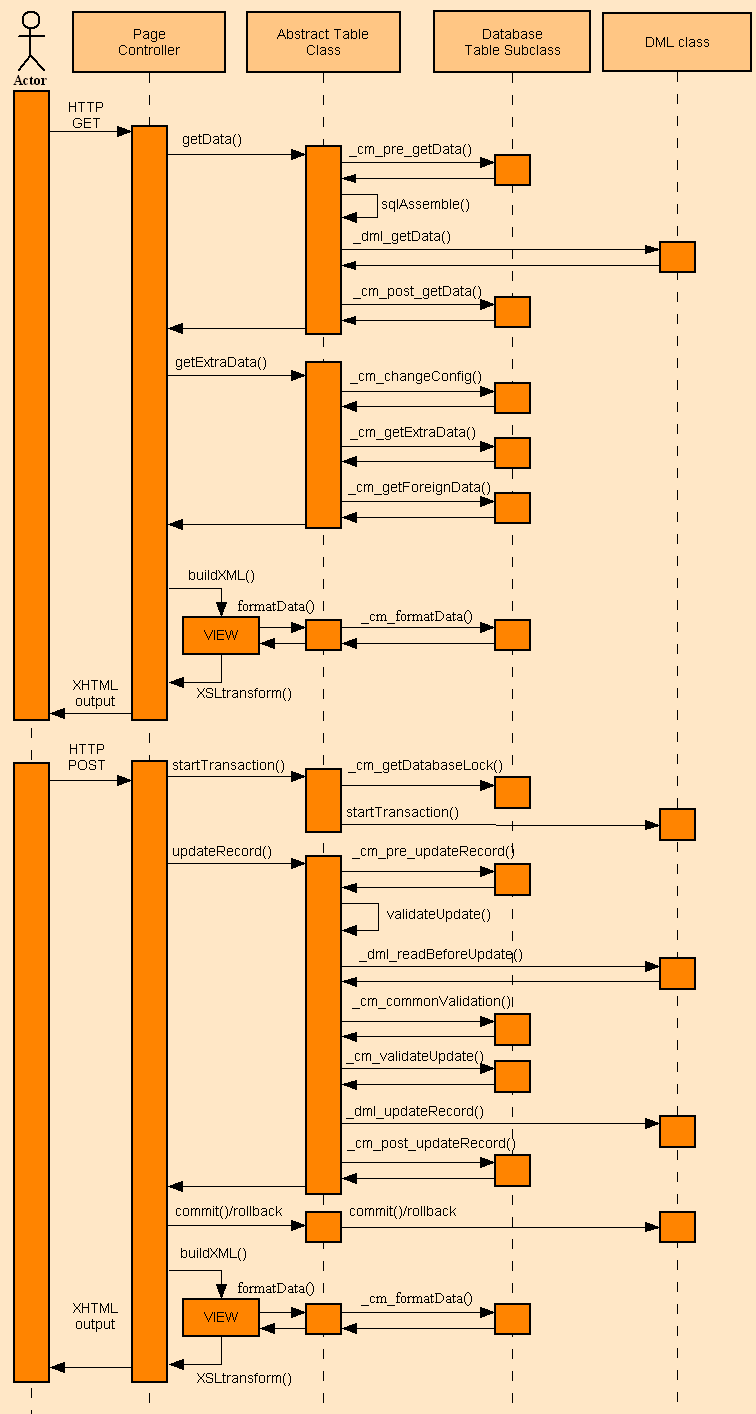
* Purpose
  + Upon initial entry this function will retrieve one or more records from the database using the primary key(s) passed down by the previous form
  + It will display the current values for one record at a time and allow the user to change them.
* Structure



* Explanation: <http://www.tonymarston.net/php-mysql/dialog-types.html#update1>
* Screenshot



* UML



# Search

* Details at: <http://www.tonymarston.net/php-mysql/dialog-types.html#search1>