



Bill Yu, Chun Wai Chan, Patrick Kowalski, Sahra Dilmaghanyan, Sami Jashromi


<http://code.google.com/p/comp1920-shoppingcart/>



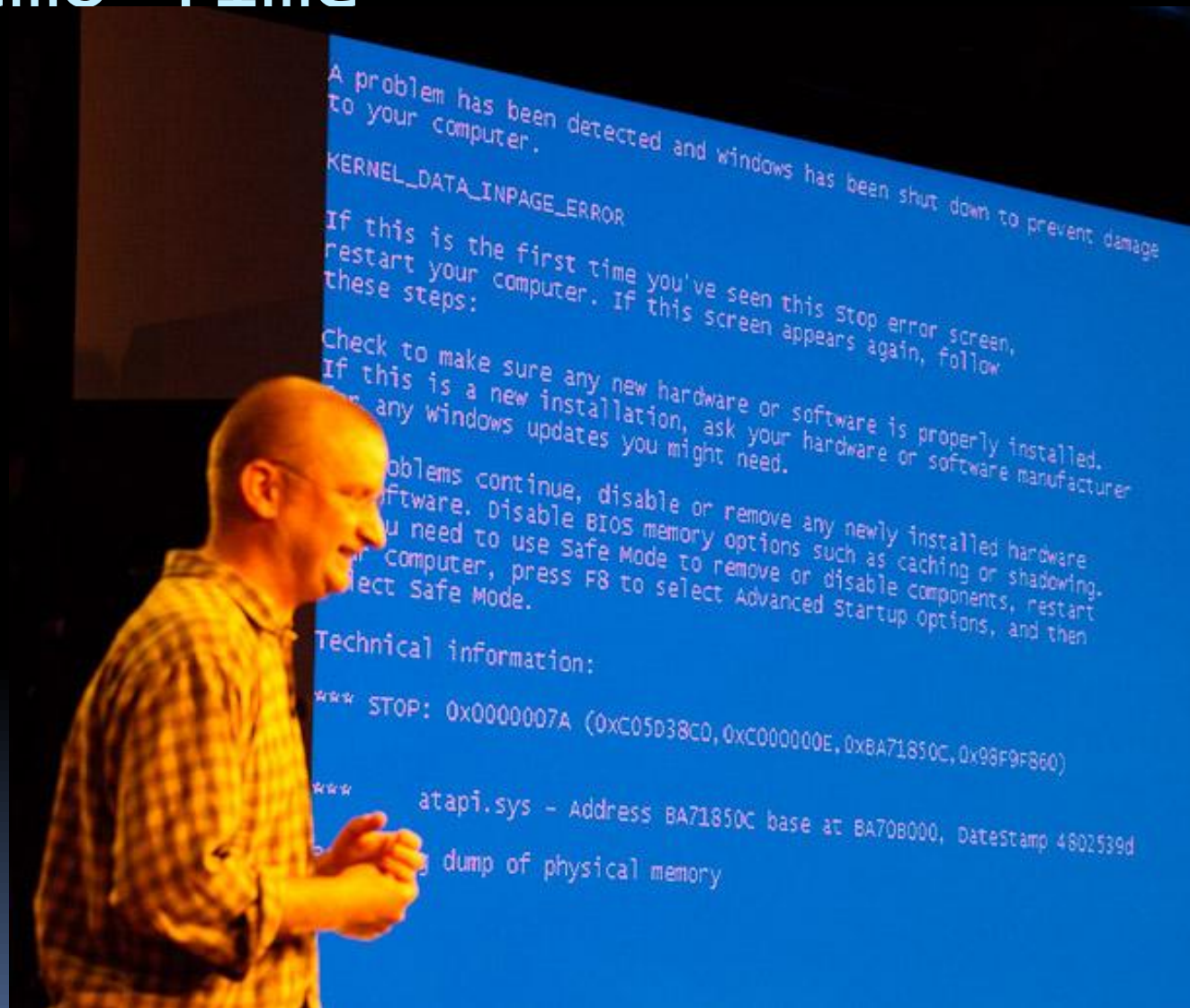
# COMP1920-SHOPPING CART



# Agenda

- Demo (3 mins)
  - Technology Overview (7 mins)
  - Session Shopping Cart (10 mins)
  - Class Exercise (10 mins)
  - Q&A
- 

# Demo Time





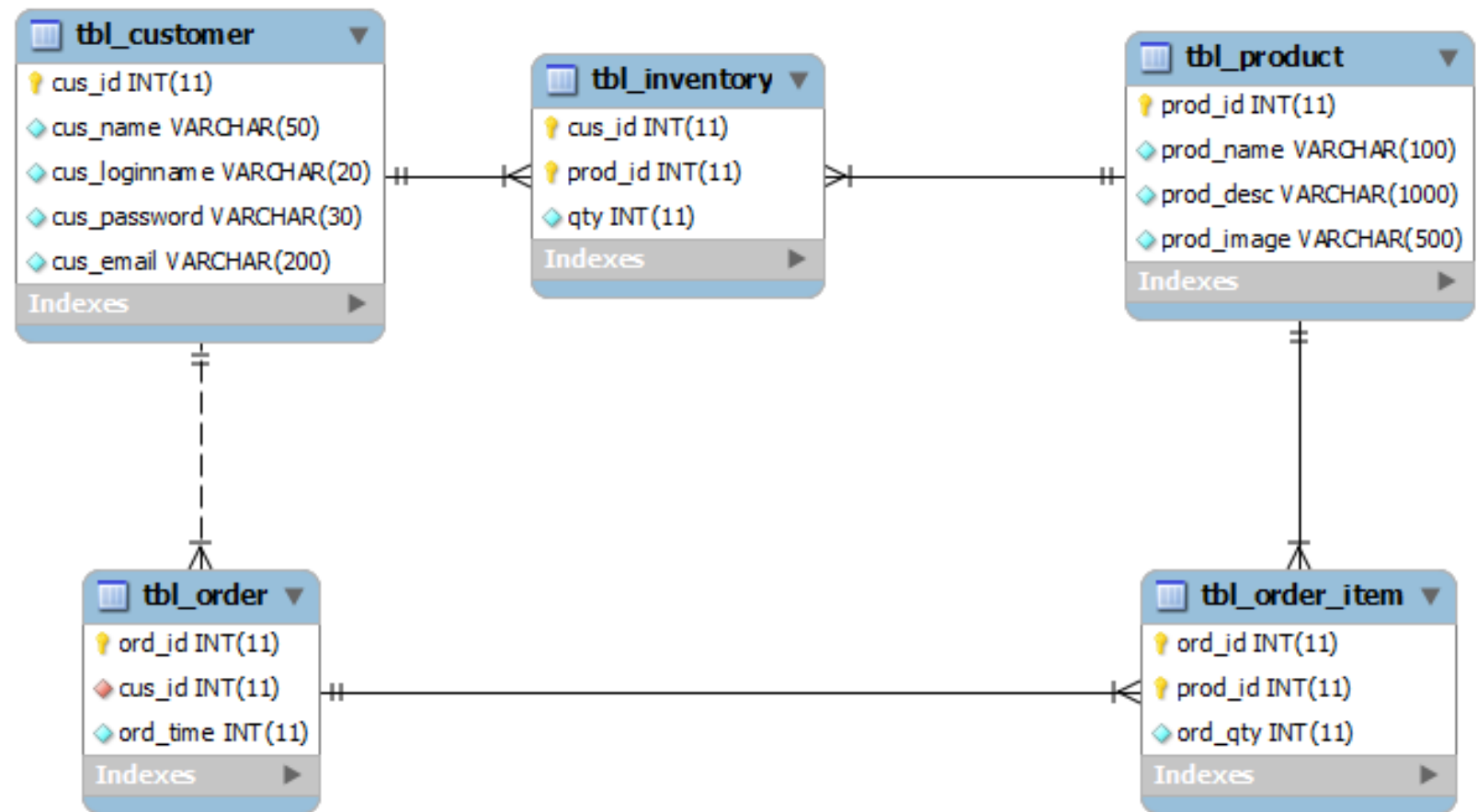
# Technology Overview

- CSS
  - MySQL
  - JQuery & Ajax & JSON & PHP
- 

# CSS



# Schema



# Ajax & JQuery

- This:

- `$("#age").load("./getage.php");`

- Or:

```
$.getJSON('./getageJson.php', function(resp) {  
    $("#age").html(resp)  
});
```

- VS:
- ```
function ajaxFunction(){ var ajaxRequest; // The variable that makes Ajax possible! try{ // Opera 8.0+, Firefox, Safari ajaxRequest = new XMLHttpRequest(); } catch (e){ // Internet Explorer Browsers try{ ajaxRequest = new ActiveXObject("Msxml2.XMLHTTP"); } catch (e) { try{ ajaxRequest = new ActiveXObject("Microsoft.XMLHTTP"); } catch (e){ // Something went wrong alert("Your browser broke!"); return false; } } } // Create a function that will receive data sent from the server ajaxRequest.onreadystatechange = function(){ if(ajaxRequest.readyState == 4){ var ajaxDisplay = document.getElementById('ajaxDiv'); ajaxDisplay.innerHTML = ajaxRequest.responseText; } } var age = document.getElementById('age').value; var wpm = document.getElementById('wpm').value; var sex = document.getElementById('sex').value; var queryString = "?age=" + age + "&wpm=" + wpm + "&sex=" + sex; ajaxRequest.open("GET", "ajax-example.php" + queryString, true); ajaxRequest.send(null); }
```



# JSON – Javascript Object Notation

```
{  
  "products" : [  
    {  
      "id" : 1,  
      "name" : "handbag",  
      "imgUrl" : "./img/handbag.jpg"  
    },  
    {  
      "id" : 2,  
      "name" : "handgun",...
```

# XML


```
<products>
  <entry>
    <id>1</id>
    <name>handbag</name>
    <imgUrl>./img/handbag.jpg</imgUrl>
  </entry>
  <entry>
    <id>2</id>
    <name>handgun</name>
```

# PHP

```
echo json_encode(  
    array(  
        array('prodId' => 1, 'name' => 'handbag',  
            'imgUrl' => './img/handbag.jpg'),  
        array('prodId' => 2, 'name' => 'handgun',  
            'imgUrl' => './img/handgun.jpg')  
    ));
```



# Shopping Cart

- It's all about \$\_SESSION!
  - Don't even need to use a DB.
- 



# Cart Item

- Simply a key value pair:
  - `productName => quantity`
- Add item/update quantity:
  - `productName => (quantity +/- some count)`

# A bunch of Items?

- Easy:

```
array( "prodName_o1" => 100,  
      "prodName_o2" => 13,  
      ...);
```



# How about the cart?

- `$_SESSION['cart'] = array();`
- That's our cart.!

# Cart Operations

- Add new item:
  - ▣ `$_SESSION['cart']['newItem'] = 10;`
- Add more:
  - ▣ `$_SESSION['cart']['newItem'] += 5;`
- Remove:
  - ▣ `unset($_SESSION['cart']['newItem']);`





# Get Cart Content

- `foreach ($_SESSION['cart'] as $prodName => $quantity) {`  
    ...  
}
  - That's it!
- 

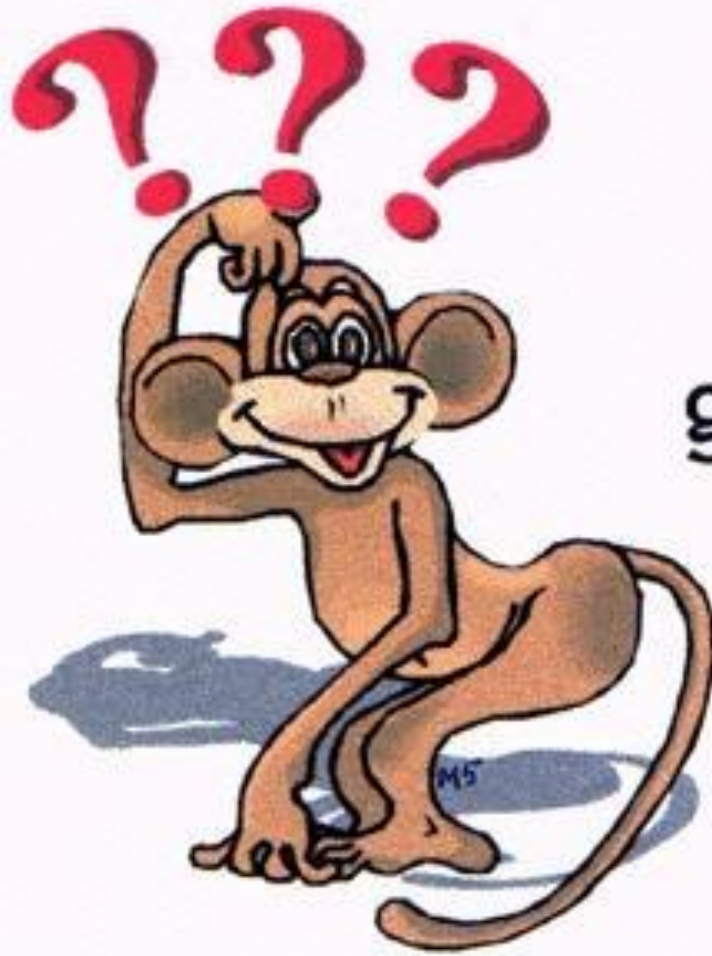
# Class Exercise



# Project Webpage

- <http://code.google.com/p/comp1920-shoppingcart/>
- Just search googlecode
- -> Project Hosting
- -> Search for comp1920
- You can sync down the full source code or read it from the web 😊

# Q & A



Questions  
are  
guaranteed in  
life;  
Answers  
aren't.