



**Studying
Made
Easy!**

ease
Studies

The Idea:

- To make studying easy...
 - MIS 121
Networking for Business
- Tested on Questions generated from the book!
- Midterm Exam
 - 100%



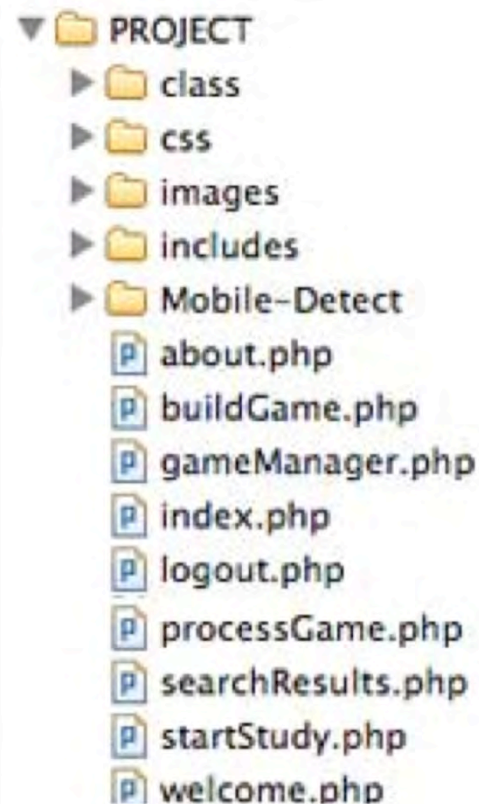


Requirements:



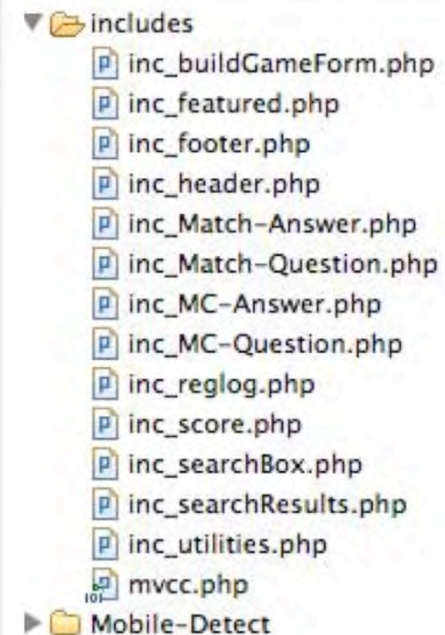
Requirements:

- o File Structure
- o Pages (8)
- o Scripts (9)
- o Includes (13)



PROJECT

- class
- css
- images
- includes
- Mobile-Detect
- about.php
- buildGame.php
- gameManager.php
- index.php
- logout.php
- processGame.php
- searchResults.php
- startStudy.php
- welcome.php

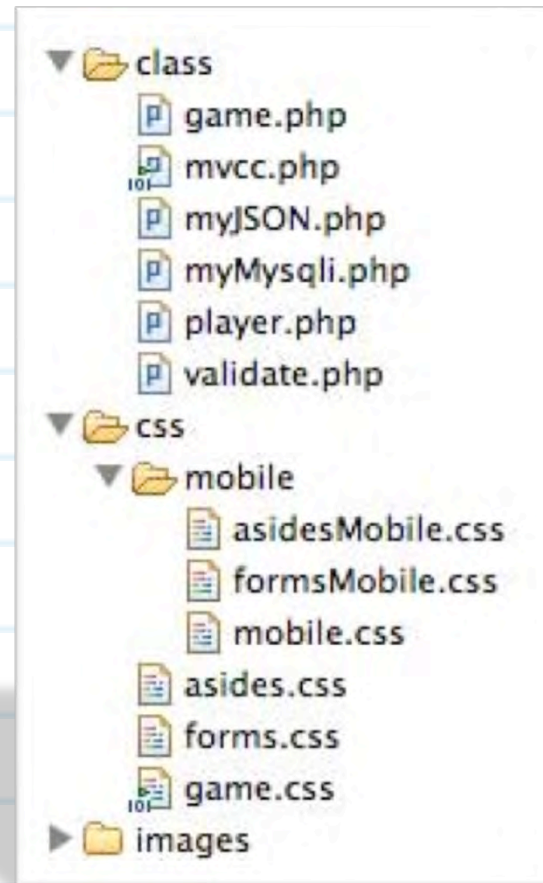


includes

- inc_buildGameForm.php
- inc_featured.php
- inc_footer.php
- inc_header.php
- inc_Match-Answer.php
- inc_Match-Question.php
- inc_MC-Answer.php
- inc_MC-Question.php
- inc_reglog.php
- inc_score.php
- inc_searchBox.php
- inc_searchResults.php
- inc_utilities.php
- mvcc.php
- Mobile-Detect

Requirements:

- o Classes
 - o Custom Classes (4)
 - o Interface (1)
- o Cascading Style Sheets
 - o Screen (3)
 - o Mobile (3)
 - o Validated



Requirements:

- Database
 - Tables (3)
 - se_users
 - se_games
 - se_highscores
- Questions Tables
 - MIS 121 (2)

Field	Type
<u>userID</u>	int(4)
username	
name	
email	
password	

	Field	Type
☐	<u>gameID</u>	int(11)
☐	<u>playerID</u>	
☐	title	
☐	score	

	Field	Type
☐	<u>gameID</u>	int(11)
☐	title	varchar(250)
☐	<u>playerID</u>	int(11)
☐	sergame	longtext

	Field	Type
☐	<u>questionID</u>	int(5)
☐	chapter	varchar(150)
☐	question	text
☐	choiceA	varchar(100)
☐	choiceB	varchar(100)
☐	choiceC	varchar(100)
☐	choiceD	varchar(100)
☐	answer	varchar(100)

Requirements:

- Database (cont.)

- Data UPDATE

- Serialize Game Objects

- LONGTEXT** Holds a string with a maximum length of 4,294,967,295 characters

- Game High-Scores

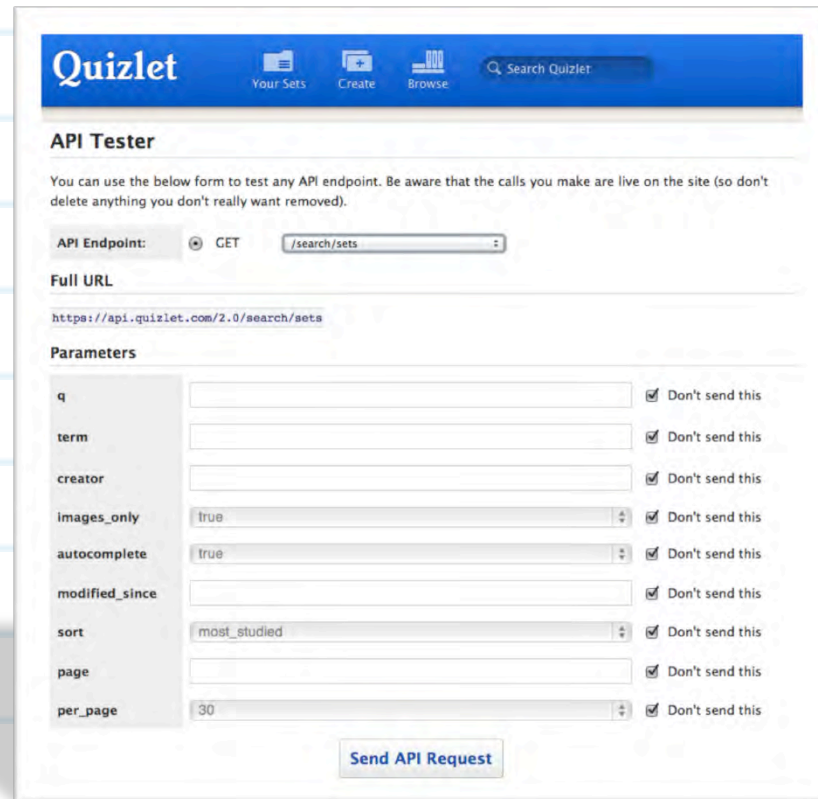
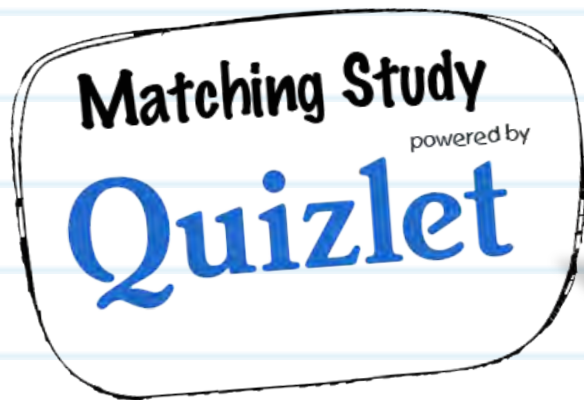
- Composite Primary Keys
 - 1 High-Score per User/ per Game

	Field	Type
☐	<u>gameID</u>	int(11)
☐	title	varchar(250)
☐	playerID	int(11)
☐	sergame	longtext

	Field	Type
☐	<u>gameID</u>	int(11)
☐	<u>playerID</u>	int(11)
☐	title	varchar(250)
☐	score	int(11)

Requirements:

- JavaScript
- JSON API



The screenshot shows the Quizlet API Tester interface. At the top is the Quizlet logo and navigation links: 'Your Sets', 'Create', 'Browse', and a search bar. Below this is the 'API Tester' section. It includes a warning: 'You can use the below form to test any API endpoint. Be aware that the calls you make are live on the site (so don't delete anything you don't really want removed).' The 'API Endpoint' is set to 'CET' and the path is '/search/sets'. The 'Full URL' is displayed as 'https://api.quizlet.com/2.0/search/sets'. Under the 'Parameters' section, there is a list of parameters: 'q', 'term', 'creator', 'images_only' (set to 'true'), 'autocomplete' (set to 'true'), 'modified_since', 'sort' (set to 'most_studied'), 'page', and 'per_page' (set to '30'). Each parameter has a checkbox labeled 'Don't send this', all of which are checked. A 'Send API Request' button is at the bottom right.

```
$jsonurl = "https://api.quizlet.com/2.0/search/sets?client_id=pDwgm6UpCD&q=$topic&page={$_GET['page']}";  
$jsonData = file_get_contents($jsonurl);  
$searchResults = json_decode($jsonData, true);
```


Requirements:

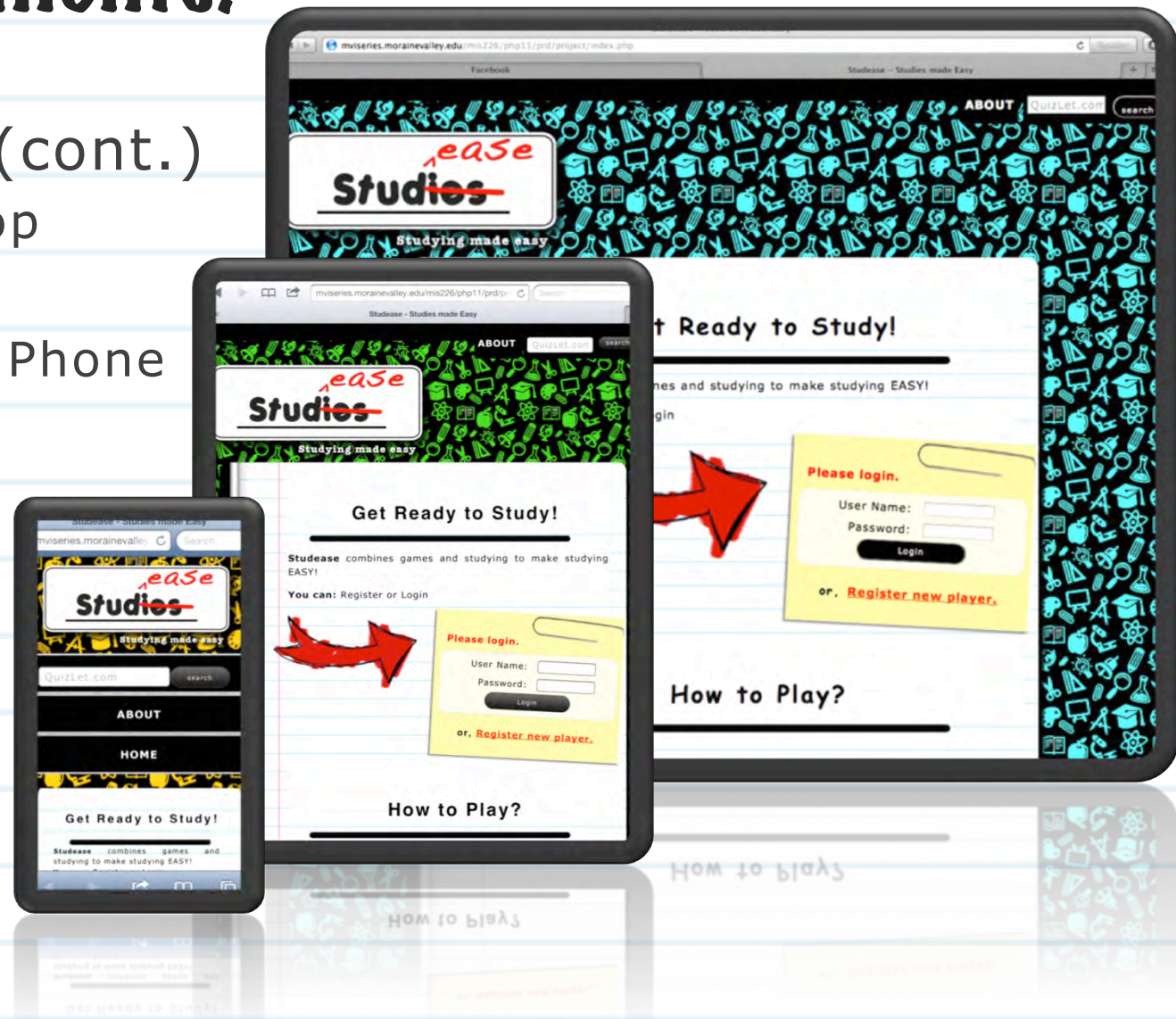
- Mobile
 - Mobile**detect**
 - www.MobileDetect.net



```
<title><?php echo "$title"; ?></title>
<?php
if ($mobileCSS)
    echo " <link rel=\"stylesheet\" href=\"css/mobile/mobile.css\" /> ";
else
    echo " <link rel=\"stylesheet\" href=\"css/game.css\" /> ";
?>
<?php $colors = array("#63fef5", "#ff3000", "#48d622", "", "#ffffff" );
$invert = array("#ff3000", "#11b700", "#0ea7ff", "#0b4ac9", "#ff0000" ); $num = rand(0,4); ?>
<style>
nav ul li:hover { background-color:<?php echo "$invert[$num]" ?>; }
body { background-color:<?php echo "$colors[$num]" ?>; } </style>
</head>
```

Requirements:

- o Mobile (cont.)
- o Desktop
- o Tablet
- o Smart Phone



Learning Experiences & Plan Deviations:



Learning Experiences:

o Design Techniques

o The Composite Pattern

o *Structural type*

o Strategy Pattern

o *Behavioral type*

```
Serializable {  
  
    /* Methods */  
    abstract public string serialize ( void )  
    abstract public void unserialize ( string $serialized )  
}
```

o The Serializable Interface

```
$game = unserialize($_SESSION['current_game']); // unserializes the game object  
$_SESSION['current_game'] = serialize($game); // serialize the game object  
  
$player = unserialize($_SESSION['player']); // unserializes the player object  
$_SESSION['player'] = serialize($player); // serialize the player object
```


Plan Deviations:

- Time effort
 - Estimated: 35 hours
 - Actual: ~40+
- JSON API
 - Brainioes to Quizlet
- Tables, Scripts and Classes
 - Small re-organizational issues

Click!

Demo Time!

