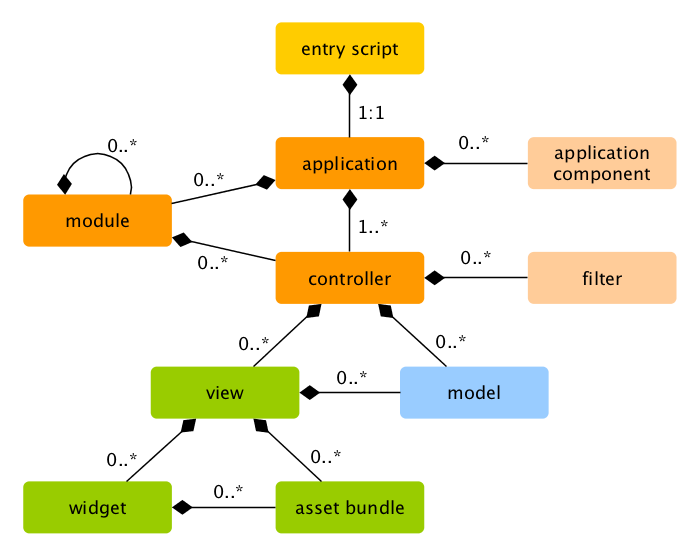
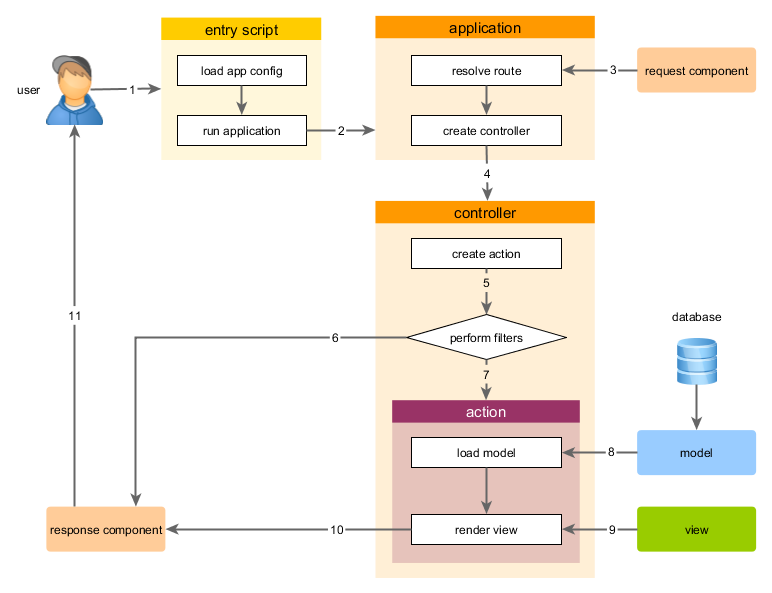
**PROJECT TECHNICAL DOCUMENT**

1. **Environment**
   1. OS: Centos 7
   2. Webserver: Apache version 2.4
      1. mod rewrite
      2. mod\_php5
      3. …
   3. Database: Oracle version 12.1.0.1.0 Enterprise Edition
2. **Program language**
   1. PHP 5.3.2
   2. HTML: use for UI (button, textbox, sheet
   3. Java script: Check validate, script (client)
   4. Jquery: user for menu, calendar
   5. Jwplayer: use for play movie
3. **Framework**
   1. Yii 2.0
4. **Architecture**
   1. MVC
      * + Model
        + View
        + Control



1. **Model**





1. **Componel in Framework**
   1. Working with Forms

To achieve this goal, besides creating an action and two views, you will also create a model.

* Create a model to represent the data entered by a user through a form
* Declare rules to validate the data entered
* Build an HTML form in a view
  1. Working with Databases

# Using Yii with Oracle through PDO

# Information for connection:

# IP server db, username, password, port

Example conntect to oracle:

'db'=>array

(

'class'=>'CDbConnection',

'connectionString'=>'oci:dbname=10.10.4.106:1521/orcl;charset=UTF8'

'username'=>'###',

'password'=>'###',

),

* 1. Modules

Modules are self-contained software units that consist of models, views, controllers, and other supporting components. End users can access the controllers of a module when it is installed in application. For these reasons, modules are often viewed as mini-applications. Modules differ from applications in that modules cannot be deployed alone and must reside within applications.

* 1. Filters

Filters are objects that run before and/or after controller actions. For example, an access control filter may run before actions to ensure that they are allowed to be accessed by particular end users; a content compression filter may run after actions to compress the response content before sending them out to end users.

* 1. Widgets

Widgets are reusable building blocks used in views to create complex and configurable user interface elements in an object-oriented fashion. For example, a date picker widget may generate a fancy date picker that allows users to pick a date as their input.

* 1. Routing and URL Creation

When a Yii application starts processing a requested URL, the first step it takes is to parse the URL into a route. The route is then used to instantiate the corresponding controller action to handle the request. This whole process is called routing

* 1. Payment

Can use method payment:

* Connect to baokim.vn
* Via visa
* Telephone card
* Via Vietnam ATM
* Via SMS
  1. Play video
* Postion video in server
* File name video

1. **Security**
   1. Authentication
   2. Authorization
   3. Working with Passwords
   4. Auth Clients
   5. Security best practices
   6. Avoiding SQL injections
   7. Avoiding file exposure
   8. Error Handling
2. **Caching**
   1. Data Caching
   2. Fragment Caching
   3. Page Caching
   4. HTTP Caching
3. **Customization JW Player**
   1. Configuration Options Reference

Configuration options tell a player instance which media to play and how to layout and behave on your page

* 1. Making JW Player Responsive

When responsive, players scale to the width of their container element, maintaining the video aspect ratio

* 1. HD Quality Toggling

HD quality toggle to your player embeds, allowing users to switch between multiple qualities of your videos

* 1. Adding Closed Captions

JW Player 6 supports the rendering of closed captions or subtitles in the video display. These captions can be selected by viewers through the CC button

* 1. Using JW Player Skins

JW Player skins change the appearance of the player, adding a custom layer to your player embeds. All visual components of the player (the controlbar, display, tooltip, dock and playlist) can be skinned. Skins are displayed on desktop and mobile browsers

* 1. Display a Playlist Sidebar

The playlist bar can display a title, description and image for every video. It can be rendered at the *right* or *bottom*of the video screen

* 1. [Adding Preview Thumbnails](http://support.jwplayer.com/customer/portal/articles/1407439-adding-preview-thumbnails)

JW Player 6 supports the loading of preview thumbnails for individual shots or scenes in a video. These thumbnails are displayed in a tooltip when a viewer hovers the controlbar. This allows for quick scanning and navigating of longer-form content

* 1. Adding Chapter Markers

JW Player 6 supports the rendering of chapter markers within the timeslider. These work particularly well for long form video, to segment it into discreet sections. A viewer can quickly hover over the cue point and can see the title of the chapter or section. Clicking on the cue point navigates you directly to that point of the video. These markers are rendered by the JW Player when a standard WebVTT text track is provided in the player embed code

* 1. Social Sharing Overlay

The sharing overlay of JW Player provides a dialog window that can contain:

* An embed code field, displaying the video embed code.
* A video link field, displaying the URL of the page that embeds the video.
* Buttons for sharing through Facebook, Twitter and Email.
  1. Display Related Videos

Through which viewers can find and watch additional videos. The related videos functionality is activated by setting the related configuration block. Inside this block, the following options can be set

* 1. Styling Captions for FCC Compliance

It is possible to change the styling of the captions using the **captions** configuration block. This is primarily relevant in the context of accessibility requirements, such as those mandated by the FCC in the United States

* 1. Building JW Player Skins

JW Player skins change the appearance of the player. No coding skills are required to build a skin, but proficiency with a graphics editor is (e.g. Photoshop or Illustrator).

* 1. Skin XML Reference

A JW Player skin is essentially an XML file containing settings and image elements. These settings and elements are grouped into 5 components.

This reference provides an elaborate overview of all settings and elements these five components support. For all settings, default values are listed between braces.

* Controlbar
* Controlbar timeSlider
* Controlbar Volumeslider
* Display
* Dock
* The Playlist
* The Tooltip
  1. Custom Text Reference

1. Website process
   1. Analysis user requirement list
   2. Design structure of the project
   3. Design main page
   4. Get confirmation from customer and modify
   5. Design function page
      1. Have all screen detail by picture
      2. Have all layout
   6. Get confirmation from customer and modify
   7. Render picture design to html and css
   8. Develop Management module (Admin page)
   9. Develop function page
      1. Get code for all screen detail
   10. Demo
   11. Test function
   12. Test security
   13. Launch website

