LOYOLA UNIVERSITY CHICAGO

CENTER FOR TEXTUAL STUDIES AND DIGITAL HUMANITIES

DIGH 402 - Introduction to Digital Humanities Design and Programming

Spring Semester 2015

Week 13

Dev Week(s) - Wednesday 25th March to Wednesday 8th April

- opportunity to test and work through concepts for final project framework
- develop a ‘view’ for a new format (current ones in /frame/view/…)
  - eg: audio, video, maps…
- develop a ‘plugin’ for a format
  - either the existing text or image
  - or for the new format (eg: to help with audio, video etc)
- either work individually or in teams
- code for ‘view’ and ‘plugin’ may be used in the final project framework
- brief presentation of work to class on Wednesday 8th April
- post updates to class’ Trello organisation
  - Dev week board
  - email me with Trello username & personal/team project title for a list on the board

Final Presentation - Wednesday 22nd April 2015

- create DH project resource
    - text, images, audio, video, maps...
- individual or group
    - max 2 per group
    - if group, each member must clearly identify work and role to gain credit
- 10-15 minutes in length plus questions
    - 10 minutes individual and 15 minutes group
    - peer-reviewed (your answers and assessment will only be shared with myself)
- offer a practical demonstration of their online resource
- answer questions on its development and publication
    - taxonomy, metadata, content structure, design choices, code additions…
- teach the class about a chosen resource and subject material
    - tests weekly reading, understanding of taxonomy, metadata, content structure…
    - content can be age specific or open - any subject, genre…
        - eg: history, literature, culture, music, video etc
        - you may use 3rd party APIs to populate content (good idea for a plugin,,,)

Object Oriented Programming

402framework - v0.6

- update framework to v0.6 - available on GitHub
  - https://github.com/dighteach/402framework
  - https://github.com/dighteach/source/tree/master/2015/DIGH402/402framework/v0.6

- update database SQL to week 13 - available on GitHub
  - https://github.com/dighteach/source/tree/master/2015/DIGH402/402framework/sql/week13/402framework.sql

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

organisation of site content - taxonomy structure

- new DB tables added to handle taxonomy and content structuring
    - **content_group** table= organisation for taxonomy subsets eg: gallery, work, catalogue…
    - **taxonomy** table = definition of taxa names within our overall taxonomy eg: devon, cornwall...
    - **taxonomy_lookup** table = taxa_id and taxa_parent_id

      eg: ‘Brixham’ with parent taxa set to ‘Devon’
      or ‘Time Passes’ with parent taxa set to ‘Uniform Edition’

    NB: both taxa come from taxonomy table...

- two new columns in content_lookup table for ‘content_group_id’ and ‘taxa_id’
    - each item of content can be assigned to a taxa and group
    - eg: ‘gallery’ belonging to ‘Brixham’

- updated constants for new taxonomy DB tables

GitHub SQL

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

updated Loader class - handling ‘taxonomy’

- load_controller() method updated to include taxonomy check based upon specified ‘id’ parameter in URI

- taxonomy details are now retrieved from the DB using the new TaxonomyController class

- after checking for ‘group’ parameter taxonomy details are set in $content_meta
    - shows title and description for specified taxonomy in URI
    - these are output instead of format, eg: image or text, title and description
    - output and rendered in sidebar

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

simple new class for handling ‘taxonomy’

- standard getter method for retrieving taxonomy row query results from DB

- taxonomy_row_query() method queries the DB to retrieve all taxonomy results for a specified taxonomy id

- taxonomy id is passed as a parameter from the now standard router URI
    - eg: content/image/gallery&id=3
    - **id=3** is the taxonomy id we are searching for in the DB

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

updated Router class - handling ‘group’

- Router class can now handle additional user selected option in URI
    - eg: content/image/**gallery**&id=3
    - group option is the third part of the URI eg: gallery

- params[‘id’] now refers to taxa id due to group in URI
    - eg: taxa_id = 3 in the DB table content_lookup

- value of ‘group’ is retrieved using a getter method in auto_load_controller() method in Loader class

- additional ‘group’ parameter now passed to load_controller() method in Loader class

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

updated Loader class - handling ‘group’

- use updated auto_load_controller() method to set $group property

- load_controller() method updated with $group parameter
    - new group view is set based on group specified in URI eg: gallery
    - group class is loaded eg: gallery.php
    - group and taxonomy attributes are passed to new get_group_content() method in group class eg: in gallery.php
    - formatted group content is returned and passed as parameter to standard draw_theme() method

- plugins can also be passed to draw_theme() method for groups...

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

new gallery class - building our image gallery as an example of building a 'group'

- designed to handle passed 'content' from selected taxa id in URI router
    - eg: content/image/gallery&id=3

- group and taxonomy attributes passed as parameters for building gallery HTML

- new methods to format_gallery_view() and format_gallery_layout()
    - format overall structure of a gallery group page
    - format each image within our gallery page
        - create a container for each thumbnail using our grid_3 class
        - add image, title with link, and associated attributes
        - return all gallery images and links as $gallery_content

- $gallery_content is then output as standard content in View class

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

new gallery ‘group’ - additional functionality and design using plugins

- plugins for gallery group
    - checks DB plugins_lookup for content_group column
    - loads required plugin for group, eg: gallery

- checks for required folder in plugins directory and loads javascript

- first gallery plugin is for a modal window to preview gallery images

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

adding a pager for group results - simple pager plugin in js

- load pager plugin for selected groups via plugins_lookup
    - currently gallery

- load plugin js to add pager functionality to group results
    - calculate and output total number of results returned for gallery etc
    - append new pager html to content div
    - calculate number of pages required based on page size (default 12) and total number of results per query
    - divide total results by page size and round up
    - append page link options to pager and add event handler for clicks on page links

- modifying to create single page/image pager?
    - an option for an additional/modified plugin...

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

book viewer - BookViewer class

- another group view class similar to the GalleryViewer class

- outputs a formatted table view for pages per book selected by the user
    - uses the URI with content/text/book&id=3
    - uses parameter in URI to query DB for taxonomy matched content

- two main methods to full format_book_view and format_book_layout for page details
    - format_book_layout() method outputs a single row per page
    - " " outputs cells for page title, page description, page text snippet, and a link to the full page text

- there is also a help method for table headers
    - NB: could be moved to a generic table builder class…

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

Current framework v0.6 examples

- image page
- text page
- gallery
- book

Object Oriented Programming

402framework - v0.7

- update framework to v0.7 - available on GitHub
  - https://github.com/dighteach/402framework
  - https://github.com/dighteach/source/tree/master/2015/DIGH402/402framework/v0.7
- update database SQL to week 13 v0.7 - available on GitHub
  - https://github.com/dighteach/source/tree/master/2015/DIGH402/402framework/sql/week13/402frameworkv0.7.sql

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

adding a menu - database tables

- two new database tables for menu and menu_lookup

- menu allows us to specify different menus for the framework
    - for example a persistent site-wide menu in the header
    - content menus in the sidebar…

- menu_lookup table allows us to associate a menu with a node and define the menu's tree
    - uses a node set in the new table 'node'
    - eg: parent links and child links
    - no grandchild links at the moment

- node table allows us to specify menu and content nodes to link to static pages
    - links can be to any page with format
    - could be static page with html, image or text etc from DB table content

GitHub SQL

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

adding a main menu

- new main menu rendered to header section of framework template

- load_menu() method added to Loader class
    - new constant is set for known menu ids in DB
    - accepts menu_id as parameter allowing us to build any required site menu
    - loads MenuController class
    - instantiates object for class
    - gets menu using menu_id parameter
    - sets menu output to static property in Loader class

- load_menu() called in draw_theme() method

- property set in load_menu() is now passed to draw_header() method

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

adding a main menu - new MenuController class

- getter method returns structured menu links for menu_id parameter

- menu_links_query() queries the DB for all parent links for the required menu_id
    - then menu_child_query() queries the DB for all child links for specified parent_id

- child links are pushed into an array with parent links
    - parent links and child links, then next parent and child links etc...

- each combined parent and child array is then pushed into the main menu_links array

- we can now use menu_links array to output our menu

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

adding a main menu - updated View class

- draw_menu() method added to View class allows us to abstract menu output
    - uses the menu_links array from MenuController class
    - checks for parent links and then each child link
    - structures the links as <ul> and <li> in HTML

- main menu is drawn from the draw_header() method
    - site wide persistent menu for links such as home, about, contact, help etc...

- css is currently in the frame.css file

- links are based on node link in node table in DB
    - allows creation of holder pages for site content etc based on Router URI standard
        - eg: page for image galleries could list info on all site galleries etc…
    - create link to a specific content item in content table in DB
    - links can be to single items, such as image or text, or holder pages etc…

GitHub Code

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

handling requests for generic controller and params URI

- updated Loader class and load_controller() method to handle controller and params URI
    - eg: content&id=22
      taxonomy&id=3

- new conditional check for controller and params

- loads class for frame/controller (eg: taxonomy, content...)

- then loads applicable frame/view
    - eg: frame/view/taxonomy.php
      frame/view/content.php

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

handling requests for generic controller and params URI - taxonomy.php

- this loads a tabular view of all content for the specific params id in URI

- eg: taxonomy&id=3 will load all content (images, texts etc) for taxa_id=3 from content table using content_lookup

- new taxonomy_content_query() method in TaxonomyController class
    - TaxonomyController class in frame/controller/taxonomy.php

- TaxonomyViewer class is used to render the tabular output for the DB result
    - TaxonomyController class in frame/view/taxonomy.php

GitHub Code 1 | GitHub Code 2

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

handling requests for generic controller and params URI - content.php

- follows same pattern as taxonomy.php for loading content&id=?

- ContentController class has now been updated with get_controller_content() method
    - accepts controller and params

- returns requested content for content and params[‘id’]

- ContentViewer class formats content for output
    - currently outputs split content view to match image and text content options in DB
    - parallel display of image and text
    - can be updated to only handle contentformat (eg: contentimage) where content is available for requested id
    - simply checks DB returned result for values

eg: content&id=15

GitHub Code 1 | GitHub Code 2

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

handling requests for generic controller and params URI - plugins

- plugins are also available for controller and params view

- checks plugin_type in plugins_lookup table for available plugins

- adds available plugins to the rendered view

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

More framework examples

- taxonomy page

- content page

Object Oriented Programming

Object Oriented Programming - Abstract overview of current framework structure

Adding our first sets of data to the framework DB

- content, content_group, content_type and content_lookup

- content_meta

- menu and menu_lookup

- node

- plugins and plugins_lookup

- taxonomy and taxonomy_lookup

- users