



WORDPRESS MARKETING FRONTEND

Extending the Course
Discovery Service

Zachary Trabookis

EducateWorkforce, Clemson University

OPENedX 2019



Course
Discovery

DELIVER COURSE MARKETING METADATA

This session will cover a technical roadmap and challenges faced when extending the Open edX `course-discovery` service for publishing to a new marketing site service endpoint.

Drupal™

Course Discovery 




 WORDPRESS





ALL COURSES



Fluid Lines and Fittings

In Collaboration With
Clemson University Center for Workforce Development


Pacing Type
Self-Paced

📊 Intermediate

- DETAILS

Designed to partially fulfill the curriculum requirements for Aviation Maintenance Technician Schools established by the Federal Aviation Regulations.

LEARN MORE



Workforce Fundamentals

In Collaboration With
Clemson University Center for Workforce Development

Pacing Type
Self-Paced


Skills Covered
Analytical and Research Skills,
Basic Computer Skills,
Communication Skills, Networking

📊 Beginner

- DETAILS

Introduces students to the skills needed to build a successful career in today's modern job market, emphasizing the importance of professionalism in all work environments and jobs.

LEARN MORE



[Courses](#) | [Features](#) | [Technical Colleges](#) | [Resources](#) | [Sign In](#) | [Start Learning](#)

Fluid Lines and Fittings

This course is designed to partially fulfill the curriculum requirements set out in Appendix B Part 147 General Curriculum Subjects for Aviation Maintenance Technician Schools established by the Federal Aviation Regulations. This course will help the learner to identify the common materials, applications, preparation, installation and repair of rigid and flexible fluid lines used in the various pressurized fluid systems in an airplane. Learners will learn about the proper material and fitting selection as well as proper preparation methods for fabrication and repair of rigid fluid lines. Furthermore, learners will also gain an understanding of the various aspects of proper rigid and flexible hose assembly installation. This knowledge is necessary for the technician's ability to properly inspect, maintain, and repair of an airplane's various fluid lines to ensure reliable operation of the various fluid systems during flight.

ENROLL NOW

ESTIMATED TIME

Approximately
2 Hours 58 Minutes

RECOMMENDED PREREQUISITES

Basic Reading Comprehension, Basic Computer Skills, Basic Internet Browsing Skills

📊 Intermediate

What You'll Learn

COURSE MODULE LISTING


- Rigid Fluid Lines
- Module Introduction
- Introducing Rigid Fluid Lines
- Rigid Fluid Lines Materials
- Fabricating Rigid Fluid Lines 📺

ROADMAP


EducateWorkforce Marketing Overview
LEARNING ENHANCED



GET	/api/v1/catalogs/
POST	/api/v1/catalogs/
DELETE	/api/v1/catalogs/{id}/
GET	/api/v1/catalogs/{id}/
PATCH	/api/v1/catalogs/{id}/
PUT	/api/v1/catalogs/{id}/
GET	/api/v1/catalogs/{id}/contains/


 **Course Discovery**
Extend REST API to account for WordPress frontend

ACF PRO


 **Custom Fields**
Extend REST API to account for additional course meta fields

theme fusion

Avada:

 **Theming**
Handle custom field display on template course pages

Search & Filter Pro


 **Search**
Provide a mechanism for exploring course meta



COURSE DISCOVERY

EducateWorkforce Marketing Overview
LEARNING ENHANCED



- Management command pulls in external LMS course api meta information to local store.
- Limited pipeline loaders specific for our sites needs.
- Ingesting of LMS course api meta now includes block level types like Chapters and Sequentials.
- Publisher app was not utilized  for our sites needs.

refresh_course_metadata.py

```

pipeline = (
    # (
    #     (SubjectMarketingSiteDataLoader, partner.marketing_site_url_root, max_wor
    #     (SchoolMarketingSiteDataLoader, partner.marketing_site_url_root, max_work
    #     (SponsorMarketingSiteDataLoader, partner.marketing_site_url_root, max_wor
    #     (PersonMarketingSiteDataLoader, partner.marketing_site_url_root, max_work
    # ),
    (
        # (CourseMarketingSiteDataLoader, partner.marketing_site_url_root, max_work
        (OrganizationsApiDataLoader, partner.organizations_api_url, max_workers),
    ),
    (
        (CoursesApiDataLoader, partner.courses_api_url, max_workers),
    ),
    # (
    #     (EcommerceApiDataLoader, partner.ecommerce_api_url, 1),

```

Django Admin > Publisher

PUBLISHER

Course run states

+ Add  Change

Course runs

+ Add  Change

Course states

+ Add  Change

Course user roles

+ Add  Change





COURSE DISCOVERY

- Define a way to separate different marketing site services
- Handled within each Partner Core configuration

EducateWorkforce Marketing Overview
LEARNING ENHANCED

```
class MarketingService(TimeStampedModel):
    name = models.CharField(max_length=128, unique=True, null=False, blank=False)
    short_description = models.CharField(max_length=255, default=None, null=True)
    course_run_publisher = models.CharField(
        max_length=15, unique=True, null=False, blank=False, verbose_name=_('Course Run Publisher'),
        help_text=_('Used to identify publisher for the Partner (e.g. for Course Run)'))

    def __str__(self):
        return '[name] [code]' % format(name=self.name, code=self.course_run_publisher)
```

Django Admin > Core > Marketing Services

Home > Core > Marketing Services

NAME	SHORT DESCRIPTION	COURSE RUN PUBLISHER
Drupal	https://www.drupal.org/	drupal
Wordpress	https://wordpress.org/	wordpress

Django Admin > Core > Partner

Marketing Site Configuration

Configure the marketing site URLs that will be used to retrieve data and create URLs.

Marketing site service: Wordpress - wordpress ✎ + ✕

Marketing Site URL: Currently: https://educateworkforce.com
Change: https://educateworkforce.com

Marketing Site API URL: Currently: https://educateworkforce.com/wp-json/
Change: https://educateworkforce.com/wp-json/

```
class Partner(TimeStampedModel):
    name = models.CharField(max_length=128, unique=True, null=False, blank=False)
    short_code = models.CharField(
        max_length=8, unique=True, null=False, blank=False, verbose_name=_('Short Code'),
        help_text=_('Convenient code/slug used to identify this Partner (e.g. for Courses)'))
    courses_api_url = models.URLField(max_length=255, null=True, blank=True)
    ecommerce_api_url = models.URLField(max_length=255, null=True, blank=True)
    organizations_api_url = models.URLField(max_length=255, null=True, blank=True)
    programs_api_url = models.URLField(max_length=255, null=True, blank=True)
    marketing_site_service = models.OneToOneField(MarketingService, null=True, blank=True)
    marketing_site_api_url = models.URLField(max_length=255, null=True, blank=True)
    marketing_site_url_root = models.URLField(max_length=255, null=True, blank=True)
```




COURSE DISCOVERY

- Switch between different course run publishers defined in Core Marketing Services using `CourseRun._locate_publisher`



apps /course_metadata / models.py : CourseRun

```
def _locate_publisher(self, partner):
    """ Locates the correct Marketing Service for the Partner"""
    switcher = {
        "drupal": None,
        "wordpress": ChapterMarketingSiteWordpressPublisher
    }

    publisher_class = switcher.get(partner.marketing_site_service.course_run_publisher)

    # Throw error if the class for handling the Marketing Service is not defined in th
    if publisher_class is None:
        raise MarketingSitePublisherException("Cannot locate publisher for marketing s

    return publisher_class(partner)

def save(self, *args, **kwargs):
    ...

    is_publishable = (
        ...
    )

    ...

    if is_publishable:
        publisher = self._locate_publisher(self.course_run.course.partner)
        previous_obj = Chapter.objects.get(id=self.id) if self.id else None

        with transaction.atomic():
            super(Chapter, self).save(*args, **kwargs)
            publisher.publish_obj(self, previous_obj=previous_obj, ignore_tag_creation
```



COURSE DISCOVERY

- Created WordPress publisher to handle this service's endpoint specific calls.
- REST Client session initiated in base publisher class to be used through calls with WordPress API.
- WordPress REST Client endpoints are mapped to publisher classes to handle updates needed for marketing service.



apps / course_metadata / publishers_wordpress.py

```
class BaseMarketingSiteWordpressPublisher:
    """
    ... Utility for publishing data to a Wordpress marketing site.
    """
    ...

    def __init__(self, partner):
        self.partner = partner

        self.client = MarketingSiteWordpressAPIClient(
            self.partner.marketing_site_api_username,
            self.partner.marketing_site_api_password,
            self.partner.marketing_site_api_url
        )

        self.rest_api_base = '{base}/wp/v2'.format(base=self.client.api_url) # urljoin(self.c
        self.media_api_base = '{base}/media'.format(base=self.rest_api_base)
        self.tag_api_base = '{base}/edx-tag'.format(base=self.rest_api_base)
        self.post_organization_api_base = '{base}/organization'.format(base=self.rest_api_base)
        self.post_course_api_base = '{base}/course'.format(base=self.rest_api_base) #urljoin(
        self.post_module_api_base = '{base}/module'.format(base=self.rest_api_base)
        self.post_lesson_api_base = '{base}/lesson'.format(base=self.rest_api_base)
        self.post_simulation_api_base = '{base}/simulation'.format(base=self.rest_api_base)

        # Define the post_base based on publisher type
        switcher = {
            "OrganizationMarketingSiteWordpressPublisher" : self.post_organization_api_base,
            "CourseRunMarketingSiteWordpressPublisher" : self.post_course_api_base,
            "ChapterMarketingSiteWordpressPublisher" : self.post_module_api_base,
            "SequentialMarketingSiteWordpressPublisher" : self.post_lesson_api_base,
            "SimulationMarketingSiteWordpressPublisher": self.post_simulation_api_base
        }
        self.post_base = switcher.get(self.__class__.__name__, None)
```




COURSE DISCOVERY

EducateWorkforce Marketing Overview
LEARNING ENHANCED

- Client object used to create requests from `course-discovery` service to WordPress REST API.`
- Base Publisher class uses this client's `api_session` method to retrieve the REST API session in order to create and update WordPress posts.
- Bearer token generated when authenticating with JSON Web Tokens security on WordPress REST API.

apps / course_metadata / utils.py

```
class MarketingSiteWordpressAPIClient(object):
    """
    The marketing site API client we can use to communicate with the marketing site (Wordpress)
    """
    ...

    def __init__(self, marketing_site_api_username, marketing_site_api_password, marketing_site_api_url):
        if not (marketing_site_api_username and marketing_site_api_password):
            raise MarketingSiteAPIClientException('Marketing Site API credentials are not properly configured!')
        self.username = marketing_site_api_username
        self.password = marketing_site_api_password
        self.api_url = marketing_site_api_url.strip('/')
        self.bearer_token = BearerToken(self.api_url, self.username, self.password) # get_bearer_token(self._api_url, self

        response = self.api_session.get('{root}/wp/v2/users/'.format(root=self.api_url))
        if response.status_code == 200:
            response_json = response.json()
            ...

    @cached_property
    def init_session(self):
        session = requests.Session()

        login_url = '{root}/jwt-auth/v1/token/validate'.format(root=self.api_url)
        token_headers = self.headers

        response = session.post(login_url, headers=token_headers)
        if not (response.status_code == 200): # and response.url == admin_url
            raise MarketingSiteAPIClientException(
                ...
            )

        return session

    @property
    def api_session(self):
        self.init_session.headers.update(self.headers)
        return self.init_session

    @property
    def headers(self):
        return {
            'Content-Type': 'application/json',
            'Authorization': 'Bearer ' + self.bearer_token.token(),
        }
```





COURSE DISCOVERY

- Meta updates are handled through base publisher methods but uniquely defined within individual types.
- Unique meta to push include the following:
 - Chapter
 - CourseRun
 - Organization
 - Program
 - Sequential
 - Simulation

EducateWorkforce Marketing Overview
LEARNING ENHANCED



BaseMarketingSiteWordpressPublisher

```
class BaseMarketingSiteWordpressPublisher:
    """
    Utility for publishing data to a Wordpress marketing site.
    """
    ...

    def __init__(self, partner):...

    def publish_obj(self, obj, previous_obj=None):...

    def delete_obj(self, obj):...

    def serialize_obj(self, obj):...

    def create_post(self, post_data):...

    def create_media(self, post_data):...

    def create_tag(self, tag_data):...

    def post_id(self, obj):...

    def edit_post(self, post_id, post_data, post_type=""):...

    def delete_post(self, post_id):...
```

```
class SimulationMarketingSiteWordpressPublisher(BaseMarketingSiteWordpressPublisher):...

class SequentialMarketingSiteWordpressPublisher(BaseMarketingSiteWordpressPublisher):...

class ChapterMarketingSiteWordpressPublisher(BaseMarketingSiteWordpressPublisher):...

class CourseRunMarketingSiteWordpressPublisher(BaseMarketingSiteWordpressPublisher):...

class OrganizationMarketingSiteWordpressPublisher(BaseMarketingSiteWordpressPublisher):...
```




COURSE DISCOVERY

EducateWorkforce Marketing Overview
LEARNING ENHANCED

- Each individual type meta classes include *publish_obj* and *serialize_obj* overrides.
- The *publish_obj* checks to see if a *post_id* exists for the type meta model and if so it performs and 'edit' rather than 'create' call to a WordPress post. 🚀
- Meta values that are passed in the WordPress Publisher REST Client request body are defined in *serialize_obj method*.

🚀 Improves on performance with REST API calls.

ChapterMarketingSiteWordpressPublisher

```
class ChapterMarketingSiteWordpressPublisher(BaseMarketingSiteWordpressPublisher):
    """
    Utility for publishing course run data to a Wordpress marketing site.
    """
    unique_field = 'location'
    post_lookup_field = 'data_locator'
    post_lookup_meta_group = 'open_edx_meta'

    def publish_obj(self, obj, previous_obj=None, ignore_tag_creation=True):
        """Publish a Chapter to the marketing site..."""
        if previous_obj and obj.status != previous_obj.status or previous_obj == None or not obj.wordpress_post_id:
            logger.info('Publishing module [%s] to marketing site (Wordpress) ...', obj.location)

            post_data = self.serialize_obj(obj, ignore_tag_creation)

            try:
                post_id = self.post_id(obj)
                self.edit_post(post_id, post_data)
            except (PostLookupError) as error:
                post_id = self.create_post(post_data)
                ...

            obj.save(suppress_publication=True, is_published=True)
        else:
            logger.info('Not publishing module [%s] to marketing site (Wordpress) since status [%s]'
                        ' of previous obj has not changed ...', obj.location, previous_obj.status if previous_obj

    def serialize_obj(self, obj, ignore_tag_creation=True):
        """Serialize the Chapter instance to be published to Wordpress as custom post type 'module'..."""
        data = super().serialize_obj(obj)
        data['title'] = obj.title
        data['slug'] = obj.slug

        if not ignore_tag_creation:
            ...

        sequentials = []
        for sequential in obj.sequentials.all().order_by('chapter_order'):
            ...

        data['fields']['module_lessons'] = sequentials

        data['fields']['goal'] = obj.goal_override
```




COURSE DISCOVERY

Sample **Chapters** for the *Fluid Lines and Fittings* course

EducateWorkforce Marketing Overview
LEARNING ENHANCED



Django administration

Home > Course Metadata > Chapters

Select chapter to change

4 results (171 total)

Action: ----- 0 of 4 selected

<input type="checkbox"/>	UUID	HIDDEN	WORDPRESS POST ID	MIN EFFORT	MAX EFFORT	TITLE	SLUG	COURSE ORDER	LOCATION
<input type="checkbox"/>	ab469be3-8a24-45e8-991f-323d34aec043	✖	1629	0:00:00	0:00:00	Overview	demo-faa-part-147-flf101-2018_2019-overview-0b7be501d67949ac9f9e5ca9accb75b1	0	block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapte
<input type="checkbox"/>	4173e057-4fd6-43b6-a7cf-6c1f5db5a337	✖	1628	0:00:00	0:00:00	Review	demo-faa-part-147-flf101-2018_2019-review-1bcf3fb9b6b34fa5804c0d896d8e9fcf	3	block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapte
<input type="checkbox"/>	9df79a4d-fb37-42b8-ae19-5bec87d39f77	✖	1630	2:58:48	0:00:00	Rigid Fluid Lines	demo-faa-part-147-flf101-2018_2019-rigid-fluid-lines-bebd7b06958f495b8d2c39722fd8d20f	1	block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapte
<input type="checkbox"/>	7f534723-36e2-4d90-9eed-e459cc2c1d24	✖	1627	0:00:00	0:00:00	Flexible Hose Fluid Lines	demo-faa-part-147-flf101-2018_2019-flexible-hose-fluid-lines-ea9431825d3d46f2bb0d66186ad7d0b0	2	block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapte

4 chapters



COURSE DISCOVERY

Sample **Chapter** within the *Fluid Lines and Fittings* course

EducateWorkforce Marketing Overview
LEARNING ENHANCED



Django administration

Home > Course Metadata > Chapters > block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapter+block@bebd7b06958f495b8d2c39722fd8d20f: Rigid Fluid Lines

Change chapter

Course run:	course-v1:DEMO+FAA-PART-147-FLF101+2018_2019: Fluid Lines and Fittings
Location:	block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapter+block@bebd7b06958f495b8d2c39722fd8d20f
Status:	Published <input type="button" value="v"/>
Title:	<input type="text" value="Rigid Fluid Lines"/>
Lms web url:	Currently: https://courses.educateworkforce.com/courses/course-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapter+block@bebd7b06958f495b8d2c39722fd8d20f Change: <input type="text" value="https://courses.educateworkforce.com/courses/course-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@chapter+block@bebd7b06958f495b8d2c39722fd8d20f"/>
Goal override:	The goal of this module is to prepare the learner to be able to identify the correct construction, and fittings commonly used in aviation fluid systems. This module covers fluid and gaseous systems within aviation that are critical to the safe operation as proper installation, inspection, and repair of the various lines and fittings throughout each system. Correct and proper inspection, installation, repair, and fabrication of these systems is essential to the safe operation of aircraft.

Sequentials:

- ✖ block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@2f0c913f1cab4c06911
- ✖ block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@720ebde8cc4b4856ae
- ✖ block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@8c6732711f2a494188
- ✖ block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@97efd03a5c0c41fab3e
- ✖ block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@99b5d3f502974a1991
- ✖ block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@e4565a261b4b498db4
- ✖ block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@f9b8b6cff5db443ea42

+ Hold down "Control", or "Command" on a Mac, to select more than one.

Min effort:	2:58:48
	Estimated number of hours:minutes:seconds [hh:mm:ss] needed to complete this chapter.
Max effort:	0:00:00
	Average number of hours:minutes:seconds [hh:mm:ss] needed to complete this chapter.
Course order:	<input type="text" value="1"/>



COURSE DISCOVERY

Sample **Sequential** within the *Fluid Lines and Fittings* course

EducateWorkforce Marketing Overview
LEARNING ENHANCED



Django administration

Home > Course Metadata > Sequentials > block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@720ebde8cc4b4856ae3b35b3339d4eff: Fabricating Rigid Fluid Lines

Change sequential

Course run:	course-v1:DEMO+FAA-PART-147-FLF101+2018_2019: Fluid Lines and
Location:	block-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequentia
Status:	Published <input type="button" value="v"/>
Title:	Fabricating Rigid Fluid Lines <input type="button" value="i"/>
Lms web url:	Currently: https://courses.educateworkforce.com/courses/course-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@sequential+block@720ebde8cc4b4856ae3b35b3339d4eff Change: <input type="text" value="https://courses.educateworkforce.com/courses/course-v1:D"/>

Objectives:

- Describe how to create a flareless fitting and demonstrate the process.
- Describe the tube forming process, identify the tools required, and demonstrate a successful tube bend.
- Explain and demonstrate the process for constructing single and double tube flares.
- Explain how to form a bead on a tubing line and demonstrate the process.
- Identify common flaws in tube flaring and the problems these flaws cause.

Hold down "Control", or "Command" on a Mac, to select more than one.

Simulations:

- simulation@fabricating-a-tube-assembly: Fabricating a Tube Assembly

Hold down "Control", or "Command" on a Mac, to select more than one.

Min effort:

Estimated number of hours:minutes:seconds [hh:mm:ss] needed to complete this sequential.

Max effort:





Average number of hours:minutes:seconds [hh:mm:ss] needed to complete this sequential.



CUSTOM FIELDS

EducateWorkforce Marketing Overview
LEARNING ENHANCED



- WordPress includes a REST API V2.
<https://developer.wordpress.org/rest-api/>
- Plugins that handle new post types with custom endpoints and fields within the WordPress REST API:
 - Custom Post Type UI (CPT UI)** 
Provides way to create custom posts within WordPress.
(e.g. Organizations, Programs, Courses, Modules, Lessons, Simulations)
<https://wordpress.org/plugins/custom-post-type-ui/>
 - Advanced Custom Fields (ACF) Pro** 
Need Pro for the Repeater field type. Allows creating a set of subfields which can be repeated.
(e.g. Chapters, Sequentials, Objectives)
<https://www.advancedcustomfields.com/pro/>
 - JWT Authentication for WP REST API V2** 
Extends the WP REST API using JSON Web Tokens Authentication as an authentication method.
<https://wordpress.org/plugins/jwt-authentication-for-wp-rest-api/>
 - ACF to REST API** 
Exposes ACF endpoints to the WordPress REST API.
<https://github.com/airesvsg/acf-to-rest-api>



CUSTOM FIELDS

Sample **POST** for the *WordPress ACF Lesson* endpoint

EducateWorkforce Marketing Overview
LEARNING ENHANCED



Method: POST | Request URL: http://ew-localhost.com/wp-json/wp/v2/lesson

Parameters ^

Headers	Authorization	Body										
<p>Toggle source mode Insert headers set</p> <table border="1"> <thead> <tr> <th>Header name</th> <th>Header value</th> </tr> </thead> <tbody> <tr> <td>accept-language</td> <td>en-US,en;q=0.8</td> </tr> <tr> <td>content-type</td> <td>application/json</td> </tr> <tr> <td>user-agent</td> <td>Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/</td> </tr> <tr> <td>Authorization</td> <td>Bearer [redacted]</td> </tr> </tbody> </table>			Header name	Header value	accept-language	en-US,en;q=0.8	content-type	application/json	user-agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/	Authorization	Bearer [redacted]
Header name	Header value											
accept-language	en-US,en;q=0.8											
content-type	application/json											
user-agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/											
Authorization	Bearer [redacted]											

ADD HEADER

Headers are valid

Method: POST | Request URL: http://ew-localhost.com/wp-json/wp/v2/lesson

Parameters ^

Body content type: application/json

FORMAT JSON | MINIFY JSON

```

{
  "slug": "cucwd-faa-part-147-ad101-template-types-of-drawings-19375859352440e6af3e54",
  "fields": {
    "effort": {
      "actual_effort": "00:07:00",
      "estimated_effort": "00:05:00"
    },
    "open_edx_meta": {
      "modified_by": 3,
      "data_locator": "block-v1:CUCWD+FAA-PART-147-AD101+TEMPLATE+type@sequential+blo",
      "date_modified": "2018-10-21 21:29:15.706294",
      "lms_web_url": "http://courses.localhost:8000/courses/course-v1:CUCWD+FAA-PART-
    }
  }
}
    
```



CUSTOM FIELDS

EducateWorkforce Marketing Overview
LEARNING ENHANCED

Edit of Course ACF Field Group

WordPress Dashboard: My Sites | Development Environment | 18 | 0 | + New

Field Groups [Add New](#)

All (5) | Active (5) | Search

<input type="checkbox"/>	Title	...	Field s
<input type="checkbox"/>	Organization		3
<input type="checkbox"/>	Course		6
<input type="checkbox"/>	Simulation		6
<input type="checkbox"/>	Module		5
<input type="checkbox"/>	Lesson		4
<input type="checkbox"/>	Title	...	Field s

Bulk Actions 5 items

WordPress Dashboard: My Sites | Development Environment | 18 | 0 | + New

Edit Field Group [Add New](#)

Course

Order	Label	Name	Type
1	Hero	hero	Group
2	Short Description	short_description	Text Area
3	Content Overview	content_overview	Wysiwyg Editor
4	Modules	course_modules	Relationship
5	Effort	effort	Group
6	Open edX Meta	open_edx_meta	Group

Field Label
This is the name which will appear on the EDIT



CUSTOM FIELDS

EducateWorkforce Marketing Overview
LEARNING ENHANCED



Edit of **Course ACF Field Group** subfield *Invitation Only*

Order	Label	Name	Type
1	Data Locator	data_locator	Text
2	Registration URL	registration_url	Url
3	Card Image URL	card_image_url	Url
4	Pacing Type	pacing_type	Select
5	Mobile Available	mobile_available	True / False
6	Invitation Only	invitation_only	True / False
7	Course Start Date	course_start_date	Date Time Picker
8	Course End Date	course_end_date	Date Time Picker

6	invitation Only	invitation_only	True / False
Field Label This is the name which will appear on the EDIT page	<input type="text" value="Invitation Only"/>		
Field Name Single word, no spaces. Underscores and dashes allowed	<input type="text" value="invitation_only"/>		
Field Type	True / False		
Instructions Instructions for authors. Shown when submitting data	<input type="text" value="Whether to restrict enrollment to invitation by the course staff."/>		
Required?	<input type="checkbox"/> No		



CUSTOM FIELDS

Edit CPT **Course** includes *ACF Field Group* subfields

EducateWorkforce Marketing Overview
LEARNING ENHANCED



The screenshot displays the 'Edit Course' interface for a course titled 'Fluid Lines and Fittings'. The interface is divided into several sections:

- Course Title:** 'Fluid Lines and Fittings' with a 'Permalink' field containing the URL: `http://development.educateworkforce.com/course/demo-faa-part-14-...-fittings-course/`.
- Hero Section:** Includes a 'Banner Graphic' field with a recommended dimension of 1200x700. The image shows a transparent aircraft fuselage. Below it is a 'Banner Description' field with the text 'Discover This Course' and a 'Banner Organization' dropdown menu set to 'Clemson University Center for Workforce Development'.
- EdX Skills:** A section for selecting skills, with 'Most Used' skills including:
 - Analytical and Research Skills
 - Aviation Maintenance Technician Schools
 - FAA
 - Introduction to FAA Part 147
 - Part 147
 - Basic Computer Skills
 - Communication Skills

- Effort:** Fields for 'Estimated' (02:58) and 'Actual' (00:00) duration.
- Data Locator:** A field containing the course ID: `course-v1:DEMO+FAA-PART-147-FLF101+2018_2019`.
- Registration URL:** A field containing the registration link: `https://courses.educateworkforce.com/register?course_id=course-v1:DEMO+FAA-PART-147-FLF101+2018_2019`.
- Card Image URL:** A field containing the asset URL: `https://courses.educateworkforce.com/asset-v1:DEMO+FAA-PART-147-FLF101+2018_2019+type@asset+`.
- EdX Tags:** A list of tags including 'assemblies', 'bead', 'damage', 'decals', and 'fitting'.



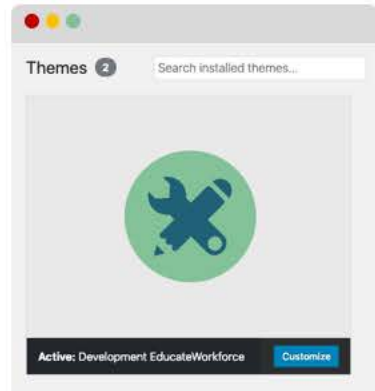
Avada:

THEMING

Branding overrides for the frontend marketing site are specific to what theme you setup.

At EducateWorkforce we use the ThemeFusion Avada theme which includes a page builder.

ACF Pro field groups values are able to be displayed on your own theme pages.



```

Project
├── course-banner.ai
├── intermediate.svg
├── module-banner.ai
├── search.svg
├── simulation-banner.ai
├── js
│   ├── responsive-background-
│   ├── single-course.js
│   ├── single-module.js
│   ├── single-page.js
│   └── single-simulation.js
├── search-filter
│   ├── 19.php
│   ├── 2894.php
│   ├── results.php
│   └── results-infinite-scroll.php
├── utilities
│   ├── acf_custom_fields
│   │   └── acf-export.json
│   ├── cpt_ui
│   ├── .gitignore
│   ├── functions.php
│   ├── header.php
│   ├── README.md
│   ├── screenshot.png
│   └── single-course.php
├── single-course-2018Q4.php
├── single-module.php
├── single-simulation.php
├── style.css
├── MEEP
├── twentyfifteen
├── twenty nineteen
├── twentyseventeen
├── twentysixteen
├── EducateWorkforce.zip
├── EducateWorkforce-2018Q4.zip
├── index.php
├── MEEP.zip
├── upgrade
├── uploads
│   ├── index.php
├── wp-includes
└── htaccess

single-course.php
32 $open_edx_meta = get_field( selector: 'open_edx_meta');
33
34
35 if ( $open_edx_meta ) {
36     $registration_url = $open_edx_meta['registration_url'];
37 }
38
39 $organization = ( array_key_exists( key: 'banner_organization', get_field( selector: 'he
40 if ( $organization ) {
41     $organization_title = get_the_title( $organization );
42     $organization_logo_banner = wp_get_attachment_url( get_field( selector: 'Logo_banne
43 }
44
45 $course_effort = get_field( selector: 'effort');
46 $course_estimated_effort = $course_effort['estimated_effort'];
47 $course_actual_effort = $course_effort['actual_effort'];
48
49 /* Description fields */
50 $overview = get_field( selector: 'content_overview' );
51
52 /* Lesson fields */
53 $course_modules = get_field( selector: 'course_modules' );
54
55 /* Course Metrics */
56 $edx_subjects = get_the_terms( $post->ID, taxonomy: 'edx_subjects' );
57
58 /* Contact Us */
59 $contact_email = 'support@educateworkforce.com';
60 $contact_subject = "EducateWorkforce%20-%20Requesting%20Course%20Access";
61 $contact_body =
62     'How can I access this course?%0A%0A' .
63     '-----%0D%0D' .
64     'Support Use Only%0D%0D' .
65     'Course Title: ' . $page_title . '%0D' .
66     'Course ID: ' . $open_edx_meta['data_locator'] . '%0D';
67
68
69 <section id="content" <?php Avada()->layout->add_style( 'content_style' ); ?>>
70 <?php while ( have_posts() ) : ?>
71 <?php the_post(); ?>
72 <div id="post-<?php the_ID(); ?>" <?php post_class(); ?>>
73 <?php echo fusion_render_rich_snippets_for_pages(); // WPCS: XSS ok. ?>
74 <?php if ( ! post_password_required( $post->ID ) ) : ?>
75 <?php if ( Avada()->settings->get( 'featured_images_pages' ) ) : ?>
76 <?php if ( 0 < avada_number_of_featured_images() || get_post_me
77 <ul class="fusion-flexslider flexslider post-slideshow">
78 <ul class="slides">
79 <?php if ( get_post_meta( $post->ID, key: 'course_videos

```


Avada:



EducateWorkforce Marketing Overview

LEARNING ENHANCED



THEMING

EducateWorkforce

Courses Features Technical Colleges Resources Sign In

COURSE DETAILS

Workforce Fundamentals

This course introduces students to the skills needed to build a successful career in today's modern job market. This course emphasizes the importance of professionalism in all work environments and jobs. In addition to this, the individual modules of this course go into detail on specific skills that are needed in the workplace such as: reading comprehension, critical thinking, mathematical skills, basic computer skills, and soft skills. Being able to apply these skills to the real world will make students more employable and more successful in their jobs.

Overall, this course provides an opportunity for students to learn about professionalism and how it can affect future employment opportunities, and provides them with the knowledge needed to progress their career in manufacturing or any field.

INVITATION ONLY

RECOMMENDED PREREQUISITES

Basic Reading Comprehension, Basic Computer Skills, Basic Internet Browsing Skills

Beginner

EducateWorkforce

Courses Features Technical Colleges Resources Sign In Start Learning

COURSE DETAILS

Fluid Lines and Fittings

This course is designed to partially fulfill the curriculum requirements set out in Appendix B Part 147 General Curriculum Subjects for Aviation Maintenance Technician Schools established by the Federal Aviation Regulations. This course will help the learner to identify the common materials, applications, preparation, installation and repair of rigid and flexible fluid lines used in the various pressurized fluid systems in an airplane. Learners will learn about the proper material and fitting selection as well as proper preparation methods for fabrication and repair of rigid fluid lines. Furthermore, learners will also gain an understanding of the various aspects of proper rigid and flexible hose assembly installation. This knowledge is necessary for the technician's ability to properly inspect, maintain, and repair of an airplane's various fluid lines to ensure reliable operation of the various fluid systems during flight.

ENROLL NOW

ESTIMATED TIME
Approximately
2 Hours 58 Minutes

RECOMMENDED PREREQUISITES
Basic Reading Comprehension, Basic Computer Skills, Basic Internet Browsing Skills

Intermediate

What You'll Learn

COURSE MODULE LISTING

Rigid Fluid Lines

Module Introduction

Introducing Rigid Fluid Lines

Rigid Fluid Lines Materials

Fabricating Rigid Fluid Lines

Search & Filter Pro

SEARCH

This **Search & Filter Pro** WordPress plugin provides a way to search ACF field group values through the **Relevanssi** plugin.

With this plugin we can define search criteria to limit which CPT objects show in the results.

A template page is included and can be styled to each unique search form.

The screenshot shows the WordPress admin dashboard with a sidebar menu on the left containing items like Dashboard, Avada, Fusion Builder, Fusion Slider, Posts, Media, Pages, Comments, Courses, Modules, Lessons, Organizations, Simulations, Portfolio, FAQs, Search & Filter (highlighted), Products, WPForms, Appearance, and Plugins. The main content area is titled 'Edit Search Form' and includes a sub-header 'Course Search' and a button 'Add New Search Form'. Below this is a 'Settings & Defaults' section with tabs for 'General', 'Display Results', 'Posts', 'Tags, Categories & Taxonomies', 'Post Meta', and 'Advanced'. The 'General' tab is active, showing 'Search in the following post types:' with a list of checkboxes. 'Courses' is checked. Other options include Posts, Pages, Media, Custom CSS, Changesets, oEmbed Responses, User Requests, Blocks, WPForms, WPForms Logs, Field Groups, Fields, Products, Variations, Orders, Refunds, Coupons, Modules, Lessons, Organizations, Simulations, Portfolio, FAQs, Elastic Sliders, wpforms-lite Announcements, Avada Page Options, Fusion Slides, Fusion Templates, and Fusion Elements. Below this are settings for 'Results per page:' (set to 10), 'Auto submit form?' (checked), 'Maintain Search Form State' (unchecked), and 'Field relationships:' (set to AND). At the bottom, there is an 'Enable Auto Count:' section with an 'Enable' checkbox and explanatory text.




Search & Filter Pro

SEARCH

EducateWorkforce Marketing Overview
LEARNING ENHANCED



ALL COURSES



Fluid Lines and Fittings

In Collaboration With
Clemson University Center for
Workforce Development


Pacing Type
Self-Paced

Intermediate

— DETAILS

Designed to partially fulfill the curriculum requirements for Aviation Maintenance Technician Schools established by the Federal Aviation Regulations.

LEARN MORE



Workforce Fundamentals

In Collaboration With
Clemson University Center for
Workforce Development

Pacing Type
Self-Paced

Skills Covered
Analytical and Research Skills,
Basic Computer Skills,
Communication Skills, Networking

Beginner

Search & Filter Pro `course` template page

```

if ( $query->have_posts() )
{
    ?>

    Found <?php echo $query->found_posts; ?> Results<br />
    Page <?php echo $query->query['paged']; ?> of <?php echo $query->max_num_pages; ?><br />

    <div class="pagination">

        <div class="nav-previous"><?php next_posts_link( label: 'Older Courses', $query->max_num_pages );>
        <div class="nav-next"><?php previous_posts_link( label: 'Newer Courses' ); ?></div>
    <?php
    /* example code for using the wp_pagenavi plugin */
    if ( function_exists( function_name: 'wp_pagenavi' ) )
    {
        echo "<br />";
        wp_pagenavi( array( 'query' => $query ) );
    }
    ?>
</div>

<?php
$course_count = 1;

while ( $query->have_posts() )
{
    $query->the_post();

    $post = get_post();

    /* Banner fields */
    $page_title = get_the_title( $post->ID );

    /* Use Hero graphics if they exists else define default values */
    $hero = get_field( selector: 'hero', $post->ID );

    /* Override this $hero variable to temporarily display the default image */
    if( empty($hero) ) {
        $banner_graphic_media = get_page_by_title( page_title: 'banner-course-default@1200x700', output: 'raw' );
        $hero['banner_graphic']['url'] = $banner_graphic_media->guid;

        $hero['banner_description'] = 'Discover This Course';
    }

    $open_edx_meta = get_field( selector: 'open_edx_meta' );
}
    
```


CHALLENGES

- Should additional course meta, not passed from the edX Course API, always be entered manually using the ``course-discovery`` service Django admin forms?
 - Objectives
 - Estimated / Actual Effort
- How often should we update the marketing frontend?
 - Triggered after a Studio course content update event.
 - Scheduled cron job task using the `refresh_course_metadata` management command.

