

# Delivering University Sites Quickly with Drupal: FFLCH case study

**Thiago Gomes Veríssimo**

University of São Paulo (USP) - Faculty of Philosophy, Languages and Literature,  
and Human Sciences (FFLCH)



24 March 2025

DrupalCon Atlanta 2025



# Introduction

- Universities have multiple organizational levels, including:
  - Departments, research groups, student organizations.
  - Administrative units, postgraduate programs, laboratories.
- Each level requires a dedicated online presence.
- Traditional solutions impose a high maintenance burden on users and IT teams.



# Problems with Traditional Solutions



- Shared servers and virtual machines require continuous maintenance.
- Users fail to update CMS platforms, leading to security vulnerabilities.
- Inconsistent configurations create security risks and technical debt.
- High support demand from faculty, students, and staff.

# The Case of FFLCH USP



- Large academic unit:
  - 10,000 undergraduate students.
  - 3,000 graduate students.
  - 300 administrative staff.
  - 600 faculty members.
- Previous web hosting solutions:
  - Shared hosting and standalone virtual machines.
  - No centralized management, security issues, lack of standardization.

# Why We Chose Drupal

- Open-source CMS with a strong community and regular security updates.
- Highly customizable with modules and themes.
- Scalable multi-site capabilities.
- Secure authentication and role-based access control.



# The Solution - Drupal

- Development of a custom Drupal distribution (Drupal profile).
- Centralized management of all university sites.
- Standardized security policies and updates.
- Scalable infrastructure supporting over 300+ sites.



# Features of Our Drupal Implementation

- Pre-configured modules for academic needs.
- Multi-site management with shared resources.
- Automated updates, backups, and security patches.
- Integration with university local systems.



# Benefits of the Solution

- Standardization simplifies user support.
- Reduced security risks with centralized updates
- To ensure security patches are applied.
- Custom integrations with internal university systems (e.g., payments, authentication, certification).
- Self-service site creation with automated deployment (Automated site provisioning via Aegir and BOA).
- Centralized access control via university-wide authentication (OAuth).
- Custom security policies enforced across all sites (example: captcha in webform).
- We have a dedicated support and development team specialized in Drupal.



# Custom Module Development

- Payment processing and billing using local payment system.
- Certification and document generation for webform (generated contributions to the Drupal community).
- To control the content over of all sites, enabling general newsletter creation, and report generation (Helping the communications team retrieve content from all sites and better promote important information.).
- Audit module
- Report module
- Custom policies module

Contributions to the Webform module: Issue 3165998, Issue 3072170



# User Training and Support

- Regular training sessions for faculty, staff, and students.
- Comprehensive online documentation (videos and tutorials).
- Dedicated support channels for troubleshooting.

# Challenges and Lessons Learned

- Initial resistance to change from users.
- Understanding Drupal API
- Balancing customization with maintainability.
- Ensuring security without limiting usability.
- Managing user expectations and feature requests.

# Future Plans

- Expand automation for self-service administration.
- Enhance user experience and accessibility.
- Increase integration with additional university services.
- Continue open-source development and contributions.

# Conclusion

- Drupal provides a scalable, secure, and flexible solution for universities.
- Meets the needs for high-volume site creation and management.
- Enables integrations, automation, and better support for content managers.

