Making the Grade:

How to Optimize Your Higher Education WebOps
Strategy for Student Recruitment and Retention

I/O amazee.io

What is WebOps in Higher Education?

Web operations (WebOps) is a set of principles governing the way your institution deploys, operates, maintains, and scales its websites and web applications. A strong WebOps strategy will facilitate the creation of workflows that are fast, safe, and repeatable and boost productivity.

Development, Information Technology, and Security teams face many challenges in higher education. Whether it's reduced budgets, shifting priorities, limited personnel or something else, these teams need to be empowered to use agile methodologies that allow them to adapt to changing circumstances while avoiding unnecessary friction. Without the proper foundation in place, you cannot move the needle in a meaningful way.

Your website is often the first impression that prospective students will have of your institution. It is a communication hub, marketing asset, and recruitment tool. If your WebOps and website are not aligned with your mission, you risk losing credibility with your audience.

Websites and web applications can be powerful assets for higher education institutions, particularly when it comes to student recruitment and success. A failure to keep up with routine maintenance and new content could mean missing organizational goals around recruitment and retention.

Could your current WebOps strategy be hindering the functionality of your sites and web apps?
Let's look at some steps you can take to build a WebOps strategy that meets organizational goals as well as improves student recruitment and retention through user experience best practices.

First, ask yourself this:

"What do I want to accomplish?"

You probably want to build and run websites and web applications that are attractive to prospective students while also meeting the needs of your current students, faculty, and staff. This is often easier said than done.

Many higher education institutions struggle to implement the continuous integration and continuous delivery workflows that would allow them to work efficiently and effectively. Your developers may even be stuck in a cycle of working on repetitive and redundant tasks. When productivity lags in this way, your team isn't serving your students and staff or the educational mission of your institution; they are serving a website that likely isn't meeting goals.

Take some time to think about the innovations you want to implement. New features, faster deployments, worry-free operations—the answer will depend on your mission. Are you focused on student recruitment? You may want to build a web application that your administration can use to analyze existing student data. Is your goal student retention? Your team may want to build a new student portal that engages students on what's happening on campus. The possibilities are endless. Understanding your mission is the first step in building, running, and securing high-performing websites and web applications.

Build the foundation

Your website won't do just one thing. Should it excite and inform prospective students? Yes. Should it effectively communicate to existing students and strengthen your alumni network? Also yes. Should it empower your faculty and staff to share research and perform their jobs? Again, yes. (And it will probably do many more things, such as supporting fundraising goals and other initiatives.) However, creating the best website user experience in the world isn't going to impress anyone if they can't even access your website.

What is keeping you online? It's critical for you to understand the foundation of your website or web application because every moment these things are offline translates to a missed opportunity, whether that's a student that wants to apply to your institution, an alumnus that wants to make a donation, or a faculty member that wants to submit research.

Before you begin thinking about the bells and whistles of features and innovative digital experiences, create a website roadmap and think about the way your site is built. What will your website need to do? **Think about the infrastructure and architecture. Think about stability and reliability. Think about the technology stack. These are the things that will support your institution's mission online.**

amazee.io's WebOps platform has been purposely built for Kubernetes-based modern cloud environments and supports any web technology or hosting infrastructure. It is also designed to handle unplanned spikes in traffic and usage. Your server's resources will scale automatically when needed.

Consider a CI/CD model

You need to deliver your websites and web applications in the way that makes the most sense for your application, your team, and your institution and its mission. A CI/CD model can help you create repeatable workflows that let your developers eliminate redundant tasks from their daily work. It also makes them more productive!

Continuous integration (CI) is an automated process that merges the code changes that your developers make into a shared repository. This means multiple developers can contribute code without causing integration problems in your project. Automated testing is used to ensure the correctness of code before it's integrated into the repository. CI puts all your developers on the same page and gives them back a valuable resource: time.

Continuous delivery (CD) is an extension of CI, with the goal of building, testing, and releasing software faster and more often. After a build, code changes are deployed automatically to testing and/or production environments. With CD, your developers work in short cycles and can release on whatever schedule makes the most sense for your institution (daily, weekly, etc.).

CI/CD at its heart is about continuous improvement. You can deliver websites and web applications faster and more reliably with automation. Higher education institutions that follow a CI/CD model make better digital experiences, faster.

Think about the worst-case scenario

When you're building websites and web applications, it's easy to focus on all the things that will go well upon launch, but we suggest you also spend time thinking about what could go wrong. It's not that we want to focus on the negative; it's just that being prepared is critical to successful outcomes. For example, what happens if your website goes down? What will the end user see? In many cases, the user will be served a generic error page, but you can offer a better experience. With a CDN you can load balance origins and serve a static version of the website. Planning ahead for the worst-case scenario will help you avoid needing to scramble to solve problems.

> **Benjamin Franklin once said,**
>
> By failing to prepare, **you are preparing to fail.**

In practical terms, we recommend thinking about, at a minimum, your plans and procedures relating to test and production environments, version control, access control, data backups, security patches, software and technology updates, and high-traffic events.

Define your performance metrics

What are your KPIs? Do you have accessibility targets? Is page load speed a critical metric for you? What about time to deployment? Think about the performance metrics that matter to your institution and supporting its mission. How will you measure these KPIs? (And how will you use them to guide improvements?)

When it comes to performance, for example, you should load test your system to prepare for high-traffic events and create a more resilient system. A load test can be used to confirm your current system limits or test new ones, but it can also be used to identify problems early on. You may need to run these tests multiple times, but once you're finished you'll have a good metric for how much traffic your system can handle.

Defining and understanding your metrics isn't just useful from a technical perspective. You can also use the data you collect to demonstrate the business value of your activities to your administration and senior leadership. This information will be particularly useful around budget time. Robust metrics can help you make a case for additional funding and resources.

What's next?

As a higher education institution your web presence is critical. After all, prospective students are likely to visit your website before they visit your campus. It's also a critical portal for communication, fundraising, information sharing among your current students, alumni, faculty, and staff. The failure of your websites and web applications will create a domino effect across your organization. Your enrollment, retention, finances, and institutional reputation may suffer.

Does your higher education institution need a trusted WebOps partner to help you build, run, and scale high-performing and secure sites and web applications?

amazee.io is more than just a hosting provider—we're your end-to-end WebOps partner. Our highly skilled team of technical experts is available 24/7 to provide enterprise-grade WebOps services so your team can harness the full potential of Kubernetes in production. **Get in touch** with us today!

I/O amazee.io