CS Assessment

STEP 1.

Continuous Integration & Monitoring

Possible solutions/tools for continuous integration(CI) and real time error monitoring(RTEM) we can use in the future.

Continuous Integration

Among the many options to choose from I think GitLab CI is the best choice for us. Why Use GitLab? The main benefit of using GitLab is that it allows all the team members to collaborate in every phase of the project. GitLab offers tracking from planning to creation to help developers automate the entire DevOps lifecycle and achieve the best possible results

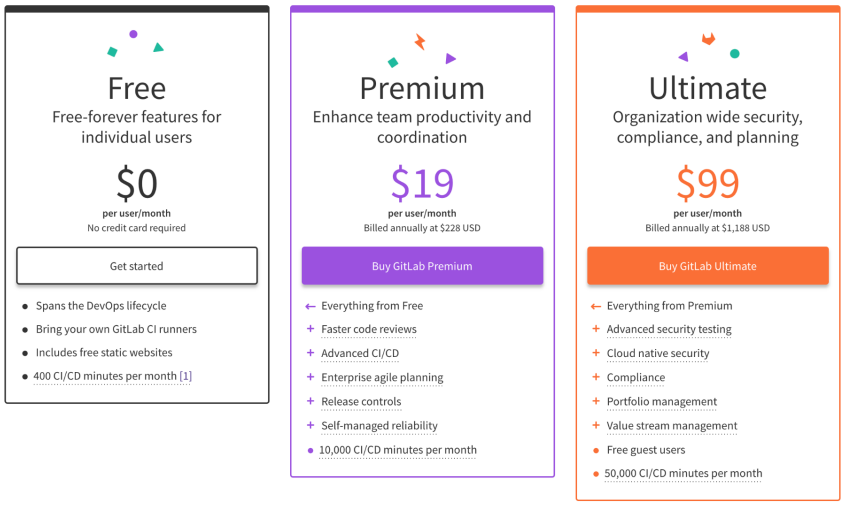
GitLab CI

Getting started: https://docs.gitlab.com/ee/ci/quick_start/ . Follow this guide and set up a test project to see how it works.

The documentation for GitLab CI is comprehensive and also has specific documentation for transitioning from CirlceCI here https://docs.gitlab.com/ee/ci/migration/circleci.html that will be very helpful.

GitLab was started in 2014 with commits as early as Sept 2010 so it is one of the more modern CI tools but it has been around long enough to be widely adopted and tested. GitLab is very popular and more than just a CI tool with features that we may be able to implement that other CI tools don't include. For more information about GitLab and everything it can do check out this page https://about.gitlab.com/ . According to Wikipedia, GitLab has over 100,000 users and is used by large, well-known organizations such as **IBM**, **Sony**, **Goldman Sachs**, and **NASA**

Pricing



Real Time Error Monitoring

Among the many options to choose from I think Raygun is the best choice for us.

Raygun

Raygun Error Monitoring is a powerful, user-centric tool that gives you actionable insights into errors and crashes impacting your users. Raygun monitors your full tech stack in real-time, from client-side and server-side, to desktop and mobile. Have the peace of mind knowing that Raygun will detect and surface every error, alongside who was affected, where they encountered the issue, as well as other real user insights. You'll have the power to dig down to the line of code to diagnose what causes the issue, alongside rich contextual information to determine commonalities between users running into the same problem.

Key features

- See the exact line of code that caused an error
- Monitor deployments to determine what caused a spike or decrease in error count

- Full stack trace information and diagnostic details for every error occurrence or crash
- Filter through your errors by date, time, version, tag, host, OS, browser, custom tags, and more
- Reduce noise with configurable filters for machine name, version, IP address, hostname, and more
- Groups errors by affected users (this also helps reduce noise)
- 180-day data retention
- Support for all major languages and frameworks
- Easy setup using lightweight SDKs
- Works seamlessly alongside Real User Monitoring and APM for full visibility into your users digital experience.

Language/ framework support

Integrations

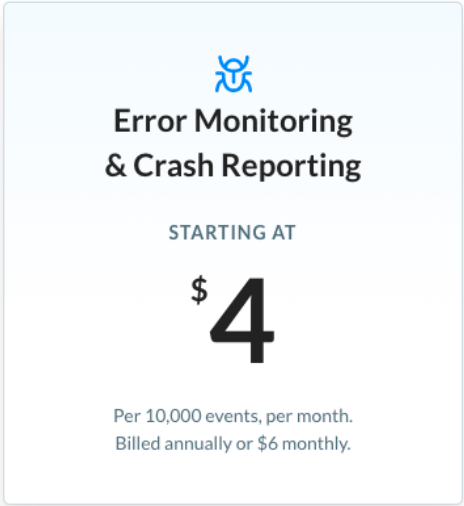
- GitHub
- Jira Software
- Slack
- Amazon SQS
- Asana
- Azure DevOps Services

How to install Raygun

Raygun is quick and easy to set up, only taking a few minutes. Simply follow the installation instructions to get started with just a couple lines of code.

Pricing

Raygun Error Monitoring and Crash reporting is priced on-demand. If charged annually, Raygun will cost $4 per 10,000 events, or $6 per 10,00 events if charged monthly.



STEP 2

Results for the extraLargeArray
insert .unshift 1.142000189 s
append .push 3.798473 ms

	XL	L	M	S	T
Insert	1.014636832 s	6.428166 ms	1.20625 ms	8.272 µs	1.756 µs
Append	0.004365936 s	0.643146 ms	0.255402 ms	9.342 µs	5.103 µs

It looks like append scales way better than insert. When the arrays are small insert is actually faster but by the time you reach medium sized data append is way faster.

.unshift is O(n)
.push is O(1)