



**REMEDY**

CS-C2130 Project Proposal

Page 1 of 2

## Project Proposal - Engine and Tools Monitoring Dashboard

### 1. Introduction

The growth of Remedy since the release of Control means that we are currently working on 5 different games built on top of 2 different toolsets across multiple branches, and all at different stages of development. There are a lot of things that can and do go wrong every day, and though we have a wealth of monitoring, logging, telemetry, continuous integration, and automated crash reporting solutions, we are still often in the dark as to the current health of our technology. It is currently very difficult to access these different streams of information or for any individual to monitor all of them in any meaningful way.

It would be incredibly helpful for each development team to be able to monitor the health of their own project. But for the centralized Northlight technology team, it is even more difficult to know how the engine is doing across all the projects that are using it. Are there any new crashes that have started occurring recently? Has the performance of the editor gone down? Is anyone even using the new editor feature that was just released? Is anyone working on this crash? These are the types of questions that we currently cannot quickly answer, even though all the information to answer them exists.

### 2. Project Goals

The main goal of this project is to create a dashboard that will pull information from all our various services (JIRA, Graylog, Sentry, Grafana, and potentially others), and visualize that information in various interesting ways in a centralized, configurable dashboard. Certain views will be required for this dashboard to be useful, such as the daily top crashes and current crash frequency across all projects, but there is still a lot of leeway to develop novel visualizations.

A good solution will be configurable enough to allow project teams to display only information relevant to their project, while allowing Northlight teams to have a wider overview of all projects. Individual users should also be able to configure their dashboards to their own preferences and have those preferences persist.

A great solution would allow you to drill down to view more information about a single crash, or to visualize a timeline of a single run of the editor all the way up until it crashed – every keystroke, command and exception as a data point.

All of this should also be linked to the actual sources of the data, as well as any relevant information. For example: a JIRA badge that links to the JIRA issue tracking the crash, the log of the session before the crash, the crash dump of the crash, a screenshot from the game from the moment it crashed. The more information we can surface without it just looking like noise the better.

### 3. Technologies

- The dashboard must be accessible as an internal website.
- Grafana is a strong candidate for creating the dashboard, but other options will be considered. Research into these alternatives is included in the scope of the project. Grafana plugins are written in TypeScript for the frontend, Go for the backend.
- Integration with Graylog, Sentry, Grafana and JIRA APIs is required.
- As Remedy heavily utilizes C# in our tools, any required standalone backends must be implemented in C# using .NET Core.



**REMEDY**

CS-C2130 Project Proposal

Page 2 of 2

#### 4. Requirements for the Students

- Remedy will provide computers and facilities for the development. Remote desktop can be used to work on these computers, or you can choose to work from our office in Espoo.
- Remedy will provide development infrastructure for the team.
- Basic knowledge of the Git version control system is a must, knowledge of the Perforce version control system is a plus.
- A minimum viable product can potentially be achieved with Grafana plugins.
- The scope of the project very flexible and can be adjusted based on the skills and enthusiasm of the team.
- All code and documentation should be written in English.
- C# experience is a plus.

#### 5. Legal Issues

**Intellectual Property Rights (IPR):** The client gets all IPRs to the results.

**Confidentiality:** The client will share some confidential information with the students. Signing an NDA provided by Remedy is required.

#### 6. Client

We have allocated time from our other personnel to assist the group regarding programming, UX and mapping out the requirements. Meeting rooms at the Remedy office can be utilized by the team for collaborative work if the team so wishes.

Remedy Entertainment Plc is a pioneering, globally renowned video game company founded in 1995 and headquartered in Finland with an office in Stockholm, Sweden. Known for its story-driven and visually stunning action games, Remedy has created multiple successful, critically acclaimed franchises such as Control, Alan Wake and Max Payne. Remedy also develops its own Northlight® game engine and tools technology that powers many of its games. The company employs over 370 game industry professionals from 34 different countries. Remedy's shares are listed on Nasdaq Helsinki's official list.

Client Representatives

**Senior Backend Developer**

Jere Nevalainen

[jere.nevalainen@remedygames.com](mailto:jere.nevalainen@remedygames.com)

+358 50 411 9885

**Principal Tools Programmer**

Sebastian Nordgren

[sebastian.nordgren@remedygames.com](mailto:sebastian.nordgren@remedygames.com)

+358 50 358 2719