Executive Summary
Dumetella is a web analytics platform. We want to give web analysts the power to view their users’ visits as though they were in the room next to them, peering over their shoulder. To do this, we innovate in two ways. First, for a given website, we focus analytics data entirely on the users of the site. Second, we collect a lot of data.

Richer Data
The most advanced web analytics platforms track every click that users make in real-time. We want to go 10 steps further. We want our users to be able to replay user visits as though they were watching someone from over their shoulder. To do this we will track every mouse movement, click, and keyboard press. We will take a screenshot every 50ms if we need to. To my knowledge, no analytics platforms currently offer these “User Visit Replays.”

User Focus
A site’s data will focus wholly on the user. We will store our users as complex documents in a document store database. We will begin by using MongoDB, changing data stores if possible.

Identifiers
For each visit, we will collect the following identifying data for users (each unique combination of user-agent and OS represents a “device”).

- User
  - Name (optional)
  - Devices
    - Browser
    - OS/platform

Analyst identifiers, [Websites]
Website [Users]
User name, identifiers, [Devices]
Device identifiers, [Visits]
Visit [PageViews]
PageView time, [Actions]
Action type, time, location
At each level, custom analyst-specified parameters (e.g. gender, “first-time”, age group, etc.) may be included as well.

**Aggregate Statistics**
We will collect the following aggregate statistics. Each view can be filtered by identifier, or by a combination of identifiers.

**Distributions**
We will aggregate the following distributions.
- Visit duration distribution
- Visit number distribution
- Visit time-of-day distribution
- Visit number of clicks distribution

**Heatmaps**
We will aggregate the following heatmaps.
- Click heatmap
- Scroll heatmap
- Mousemove heatmap

**User Visit Replays**
A UVR will be a direct re-creation of a visit. It should have the following characteristics.

**Mouse movements**
This will likely use a simulation of a cursor (e.g. a transparent black shadowy circle).

**Clicks**
These clicks may load new pages. Even in that case, the same visit must be playing after the new page load.

**Scrolls**
These scrolls must be smooth and performant. They must not interrupt mouse movements, or cause the cursor to disappear.

**Time tracking**
There must be a way for the analyst to see the length of the visit, the time spent on each page, and how much time has passed in the replay. (e.g. a YouTube style scrubber with stopping points for page switches).

**Ability to pause and skip around**
The analyst must be able to pause the replay and skip around to previous parts in the replay.

**Notification when the user is not in the current tab**
There must be a way of notifying the analyst when the user backgrounds the tab, or otherwise shifts focus.

**Quick loading**
We want a UVR to be ready to view in a reasonable time, after the analyst requests it. For example, no more than 5 seconds.

**Quick availability**
We want a UVR to be available for viewing soon after the user visits the site. For example, no more than 10 minutes afterward.

**User agent emulation**
The UVR should depict the experience on the user’s browser.

**Location emulation**
The UVR should be localized to the same place as the original view.

**Device size emulation**
We want the screen size of the user’s device to be accurately represented.

**Window size synchronicity**
We want the browser window size of the user’s device to be accurately represented.

**Logged in experience**
You must be replaying the visit as the user saw it. We may have to examine and save session cookies, to “friendly hijack” a session later on. If there is an easier way to do this, that would be good, but we may have to duplicate a user’s cookies.

**Content synchronicity**
We want to show content in the same form during a UVR that it was in when the user visited. For example, if edits have been made to a blog post, those should not be applied during the UVR.

**Third party synchronicity**
We want to show third party content in the same form that it appeared when the user visited. For example, ads should be the same, third party plugins (e.g. Disqus comments) should reflect the content as it was on the date of the visit.

**Realistic load times**
We want a UVR to accurately depict the bandwidth and connection speed that a user was experiencing.

**Deliverables**

**Main Analytics Platform**
The main analytics platform will have as many of the features outlined above as possible. A successful project will include 70% - 100% of the features above.

**User Visit Replays**
Because User Visit Replays are somewhat uncharted waters, it is hard to say whether a smooth experience is technologically achievable in current browsers. A successful project will include or prove unachievable 70% - 100% of the features above.

**Report**
At the end of the semester, I will hand in a final report of my project. The report, which will be between 3 and 6 pages, will summarize the project and what I have accomplished. It will compare our project to alternatives, as well as discuss difficulties we faced in building main analytics platform and the UVR engine. If I think there are next steps worth mentioning, I will mention them.