GeoPost SRS/SDS Changelog

Michael Davis davism78@uw.edu
Megan Drasnin mdrasnin@uw.edu
Ethan Goldman-Kirst egk35@uw.edu
Matthew Hertogs mhertogs@uw.edu
Neil Hinnant nrhinnan@uw.edu
Katherine Madonna madonk@uw.edu
Andrew Repp ajrepp@uw.edu
Duncan Smith duncan26@uw.edu

THIS CHANGELOG IS CUMULATIVE OVER ALL RELEASES

V1.0 Changelog

Schedule:

-Updated Schedule during postmortem

Extensibility Exercise:

-Added to SDS document

Feature-Complete Release Changelog

UML Diagram:

-Updated the UML diagram for logging in through Facebook.

Beta Release Changelog

SRS

Product Description:

-Added a link to the user-facing website

Process Description

-Updated database documentation to show we’ve switched from AWS to Parse

-See tradeoff discussion in the SDS changelog section under “Data Storage”

-Neil is now assisting Duncan on the test team

SDS

Main Modules:

-Updated naming

Data Storage:

-Noted that we are using Facebook’s Parse as our database system. We were initially going to use AWS SimpleDB, but we became aware of Parse, and the decision to switch was easy. Parse provides a free tier of operation with no possibility of overages, it provides a level of abstraction that enhances security (database security is entirely handled on Parse’s end), and it provides built-in integration with Facebook. We could identify no downsides to switching to Parse over AWS. Both databases are easily scalable, and both have strong documentation and company support.

Assumptions:

-Clarified security concerns. Parse handles many security concerns for us.

-XSS/injections

-Parse’s special permissions system allows us to handle permissions on objects easily

Diagrams:

-The Class diagram was updated to match our new architecture

-We added the ParsePin class

Risk Assessment:

-Addendum to the Database system risk. We did have to change DB platforms.

Schedule:

-Modified the completion dates of Beta features to 5/16, as that’s when they are likely to be complete. We had initially hoped to complete different features in a piece by piece fashion but things have ended up being developed in parallel. We found that the interrelation between the Database and Core App teams required they work in parallel to maintain interface consistency.

Team Structure:

-Updated assignments to be designated by feature team rather than individuals

Bug Tracking:

-Added the GeoPost standard bug tracking format.

Zero-Feature Release Changelog (old)

Use Cases:

Primarily, we added user steps to each of our use cases. This shows the workflow for the user in performing each of GeoPost’s functions. Additionally, we had revised GeoPost’s spec to disallow viewing posts at locked locations, so we removed the “View Post at Other Location” use case, as it no longer applies.

Process Description

- We added a section describing our plans for user testing.

- In the Software Toolset section, we initially said that storing user data would be a stretch feature. We revised the documentation to say it's a core
feature.

- We changed the Risk Summary section to match the Architecture Document.

- We replaced the old schedule with our updated one from the Architecture Document.