Software Requirement Specification for Monitoring Solution Technology

Goals of Software:

- Create a complete monitoring solution on uptime and usage of Camara computers on location
- While the monitoring software is primarily intended to gather information for Camara, school principals should have access to relevant data
- Client software should track application usage, in a non-obtrusive manner
- Client software should attempt to determine, on startup, whether the last shut down of the computer was legitimate or due to crashing
- Uptime of client computers should be tracked
- Server software should aggregate client application usage information, track client computer uptime and provide a manner of accessing this information
- Accessible information should vary depending on user accessing the data - school principals should only be able to access information on their own school; Camara admins should be able to view all data at a school, region, country or overall level
- Editing of data should be highly restricted - students, principals and Camara users should have no reason to edit the data, provided the system is working correctly

Functional Description:

Client Application:

- Both Windows and Linux versions will be available.
- The application will run constantly in the background, starting once a user logs in.
- As the application will be constantly running in the background, the application will be made to be as lightweight as possible.
- On installation, the application must establish the Afritrack ID of the computer it is installed on. It will assume that the name of the computer is the Afritrack ID, but ask for confirmation and allow the ID to be amended.
- Every ***X*** seconds, the application will check what program is responsible for the foreground window (default value of ***X*** will be 10, will be possible to adjust as desired).
- On startup, the client application will attempt to determine whether the last shutdown was intentional or the result of a crash.
- If a local classroom server exists,at regular intervals (e.g. every half hour), the client application will report the uptime and usage information since the last report to the server.
- If no classroom server exists, but an internet connection exists, once every 24 hours the client application will update the central database with uptime and usage information gathered since the last report.
- If no server or internet connection is available, the application will store the data locally, available to be retrieved manually if/when an on-site inspection occurs.
- Daily information will be consolidated into weekly information on a weekly basis. Weekly information will be consolidated into annual information on an annual basis, to reduce storage requirements.
- While running, the application will sit in the taskbar tray as a small icon.

Server Application:

- The server application will need to work both as a local server in a classroom, possibly offline, and as a cloud server application, for consolidating the data from all Camara computers and supplying that data to those who wish to retrieve it.
- The server application will also serve a web application, the interface for viewing the data.
- The primary function of the server will be to receive data from each client application, and attempt to establish the state of Camara computers that are not reporting in.
- To establish the state of Camara computers that are not reporting their usage statistics, the server will ping the client computers, allowing the server to determine whether the client is entirely offline, or online but otherwise malfunctioning.
- The Server Application will similarly compress the data it collects over time, consolidating into weekly and yearly information.
- Care must be taken to design the database carefully, as a large amount of data will be collected and an inefficiently or poorly designed database will soon create many

problems. There must be access control tools set up to handle different permission levels for schools and Camara staff.

Web Application:

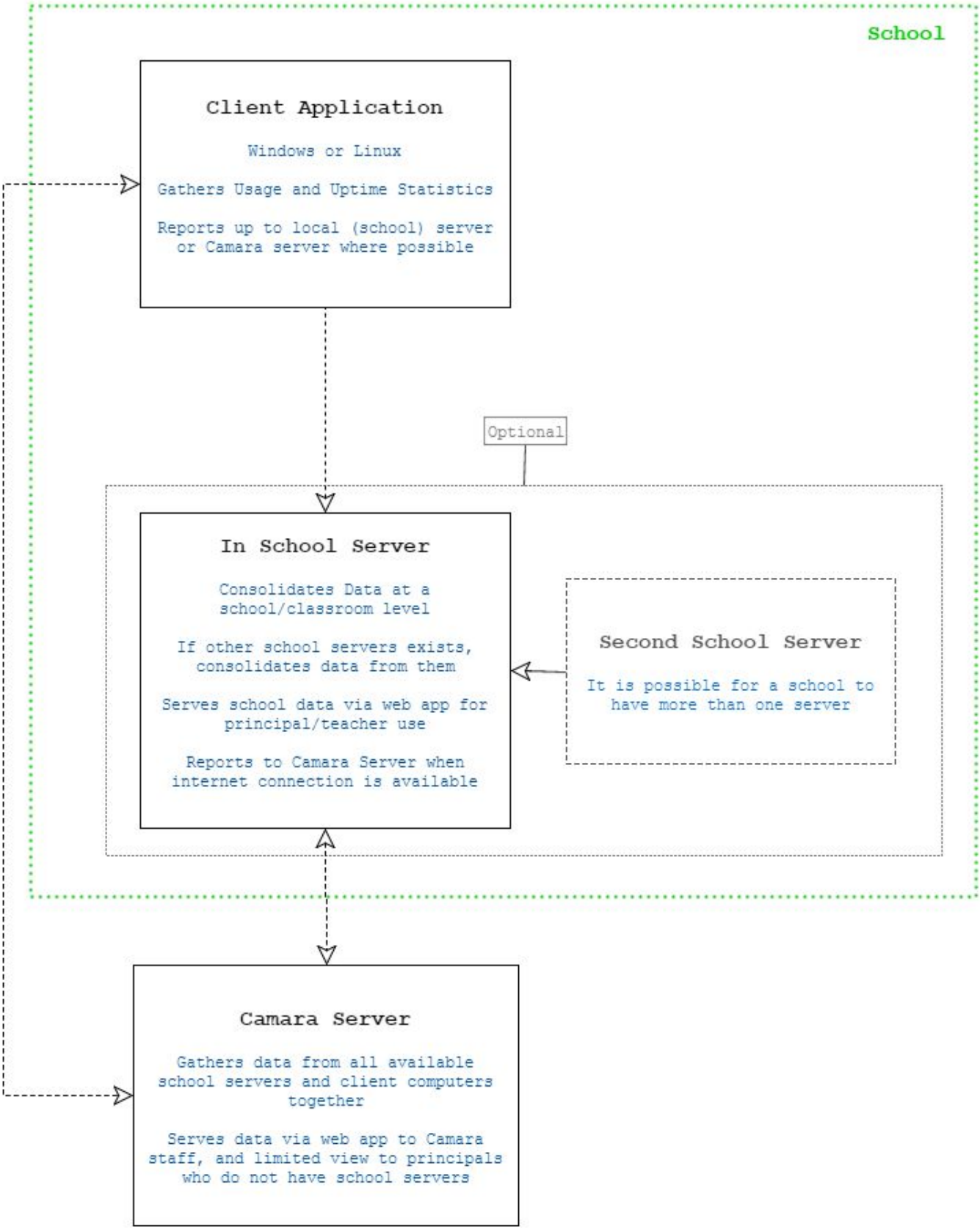
- The web application will simply be a manner of displaying the aggregated data.
- It will be necessary for the web app to have a login, as different amounts of information will be available to different people - for example, school principals should have access only to the information about their own schools.
- The web app should display alerts for schools or computers with extended periods of downtime, frequent periods of downtime, or lack of use.
- The web application must have a straightforward interface that is reasonably visually appealing, or it will not be used.

Non-Goals:

The following features will **not** be part of this software:

- **Hardware**
  This is a software only solution, hardware solutions such as the Intel Education Access Point are not being considered.
- **Major operating system modifications**
  E.g. reskinning Ubuntu to look like Windows.
- **Individual student tracking**
  While this may be added in the future, tracking the usage of each individual student is beyond the scope of this project.
- **Custom versions of open source educational software**
  This software will stand alone from the educational software. Khan Academy and similar programs will not be modified to allow for granular tracking of student progress. This software will provide a more flexible solution, allowing new educational software to be adopted without having to write new code to adapt the program for tracking.

Data Flow Chart:



Hierarchical Overview of Data: