**Encrypted Email**

**Detailed Design**

**COP4331, Fall 2015**

Team Name: Encrypted Email

Team Members:

* Ryan Patrick | ryanxanpat@knights.ucf.edu
* Alex Chatham | achatham@knights.ucf.edu
* Thomas Tavarez | peepq@knights.ucf.edu
* David Clapp | davidclapp@knights.ucf.edu

Modification History:

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Who | Comment |
| 1.1 | 10/29/15 | Ryan Patrick | Added modification history, updated documentation |
|  |  |  |  |

**Contents of this Document**

Design Issues

Detailed Design Information

Trace of Requirements to Design

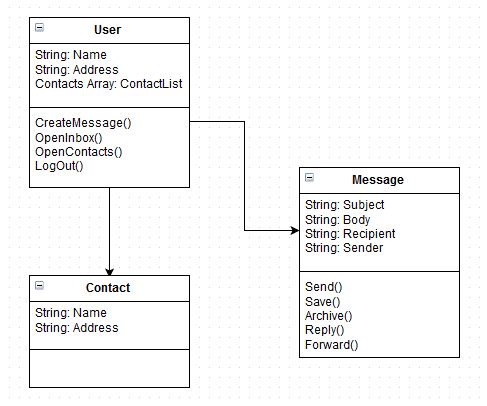
**Detailed Design Issues**

* Reusability
  + There is not a plan to reuse this application after the project for the class is over.
* Maintainability
  + Maintenance should prove to be one of our easier design issues.
    - The maintenance work will include keeping the server running and making sure the server doesn’t become slowed down.
    - The codebase will be easily maintained and tested through Github and then deployed to the server when ready.
* Performance
  + Using HTML5 on the frontend should be light enough for any computer connecting to our application. Performance issues for users are not expected.
  + On the backend, our server will perform message delivery tasks. We do not expect that performance will be an issue for the server.

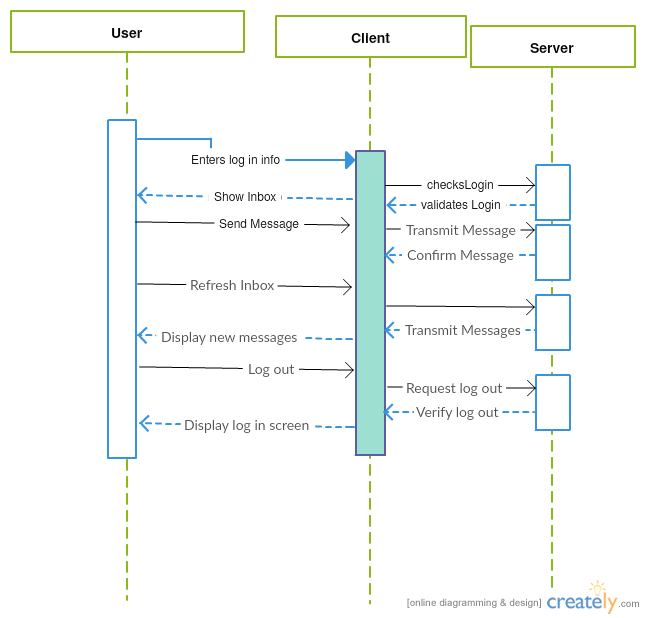
* Testability
  + We plan on testing each module at its completion in the development process.
    - If it requires another module completed to be tested, testing will be postponed until that module is ready.
  + All developers will participate in testing and bug fixing.
    - Bug tracking will occur through Github
    - Bug fixing will be continuous until the product is ready to be delivered.
* Portability
  + Our application is intended to work on the internet as a stand-alone client.
    - We expect that our application will work from both Android and IOS based web browsers allowing mobile support.
    - If time provides, we will try to deploy an Android application as well.
* Prototypes
  + When new functionality is added or a class is implemented, the developer will prototype their work.
  + Prototyping will occur for everything.
* Technical Difficulties
  + Network connection between the client and the server.
  + Learning SSL
  + Learning Ruby on Rails
* Trade-offs in Design Architecture
  + We want to deploy an Android app in addition to our web based app, but will only do so if time permits.
* Rationale for the Architecture
  + By working with a web based interface, our application will be able to be used by any device with a web connection and a modern browser.
  + Our developers are all using different operating systems and browsers. This gives us good testing on a variety of platforms.

**Detailed Design Information**

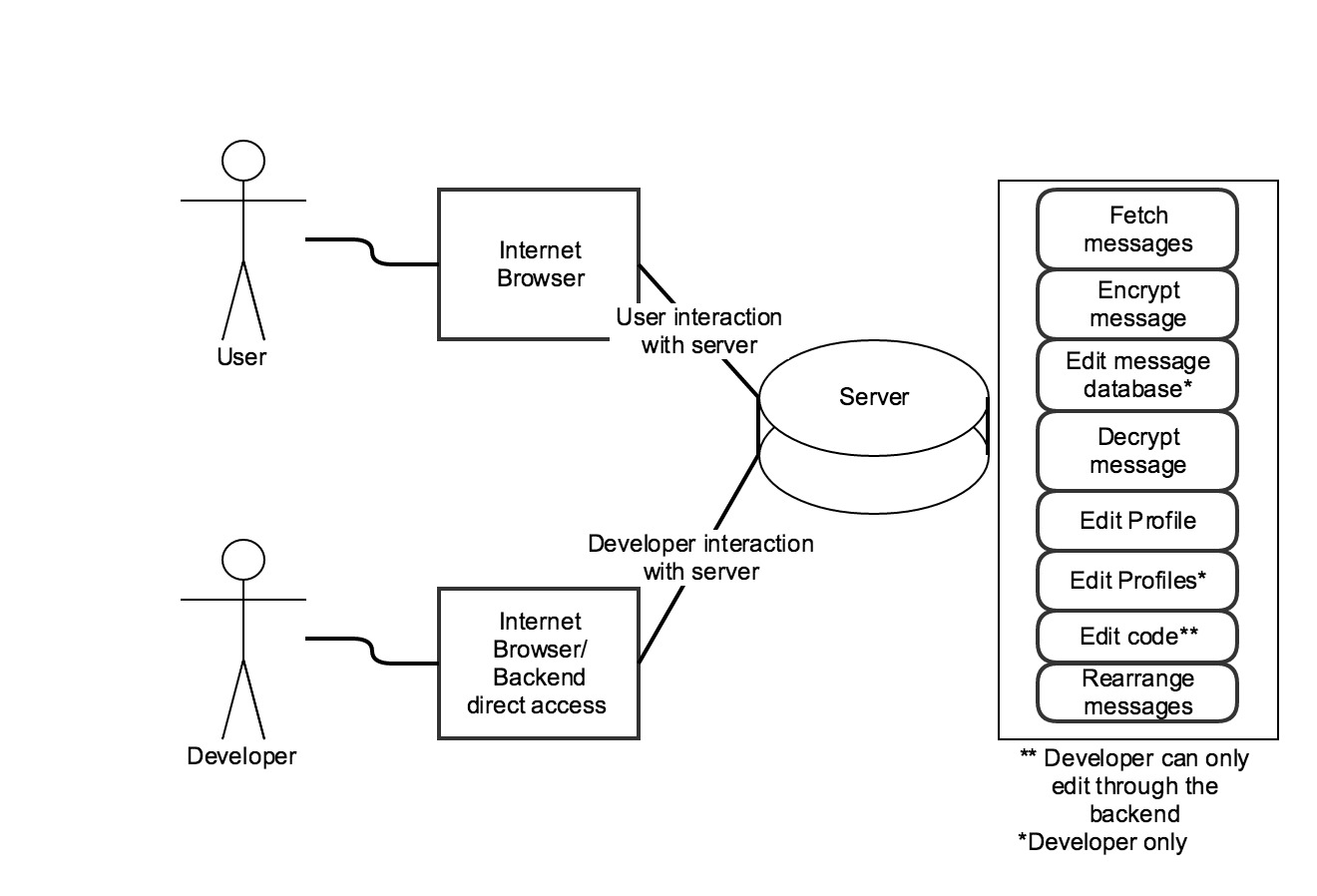
Class Diagram



Sequence Diagram

****

Activity Diagram



**Trace of Requirements to Design**

|  |  |
| --- | --- |
| Requirement | Trace to design |
| 3.1.1 Messages | The Web Application shall be able to send and receive electronic messages between system users |
| 3.1.3 New User | Users shall be able to create account which they will be able to log in and out of the system |
| 3.8.1 Accounts | Messages shall only be accessible by accounts that the message was sent |