Final Architecture Plan

WeCrypt Messenger

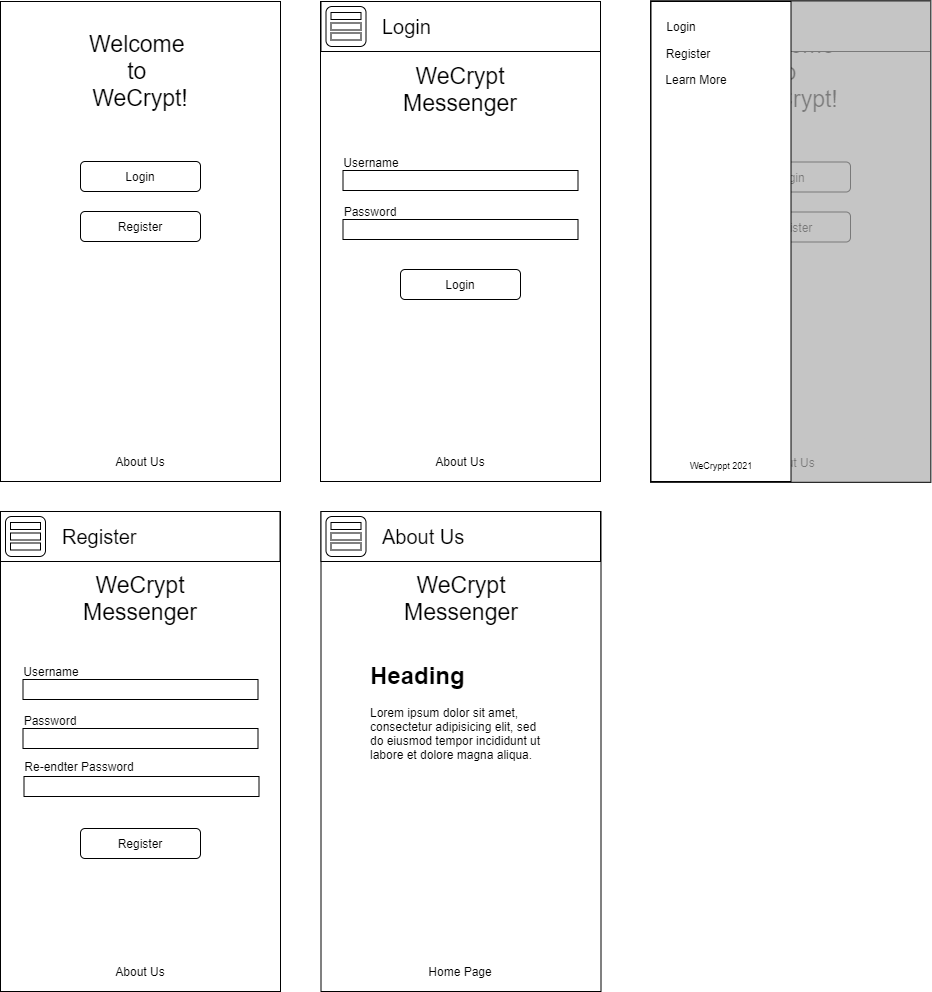
Prepared By Jonathan Nungaray

|  |
| --- |
| Design Planning Summary |

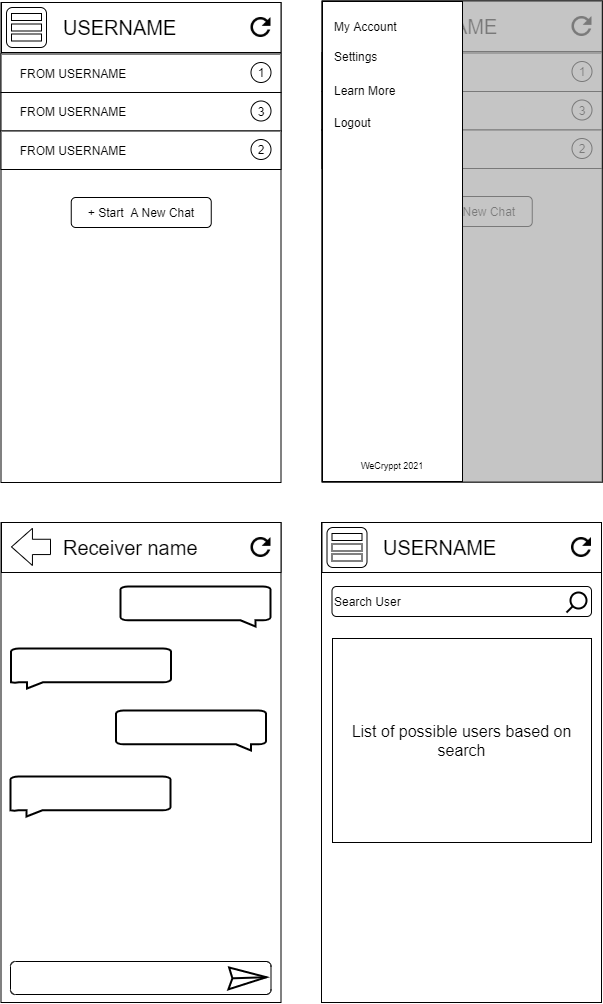
This project is being developed and created for the specific purpose of creating a truly secure messaging system. Currently there are encrypted messaging systems out there but none of them can be considered fully secure. There is always a log of the messages and someone who is able to access those messages and it is possible for them to be retrieved. With this application your conversations will be completely secure and unobtainable. This will be done by having no database that holds message logs, all messages will be encrypted from end to end, and no one will have access to these messages once the application has been closed out. By leaving no message history and fully encrypting the messages from end to end, there leaves little to no opportunity for the message data to be stolen or intercepted. This is ultimately the goal of this project. The reason behind this is that some group or organization could have access to data whether it be the government or application owners if the message history was stored somewhere, but if none of it is kept then no one can gain access.

|  |
| --- |
| Overview Of Design Concepts |

For this barebones and simple application, we wanted to give a very simple design and flow to the application. The goal with this is to make the user comfortable and not have them worry about unnecessary functionality at first such as needing personal information, the need for a profile picture, or anything of that sort. For that reason our basic design will follow the mockup wireframes that are to come. First we will show off the concepts for a user who is logged out of the application or does not have an account to login to.



We wanted to keep the options limited and simple. One item missing in these mockups is the hamburger menu on the home page. It should be there but the design was later updated and this page design did not get that update. We opted out of any form of personal information that the user would need to sign up initially, only requiring they provide a username and password that they desire to use. This is to further prove our desire for the user’s security. Because of the lack of an email we also have the lack of a “Forgot Password” feature for the time being. We would later include the ability to optionally input an email for this reason. To prevent an overload of accounts we will be deleting any account that remains inactive for a period of 3 months in a row. All of this information will be outlined in the “About Us” and “Learn More” pages. Next we will go over the logged in features and functionality.

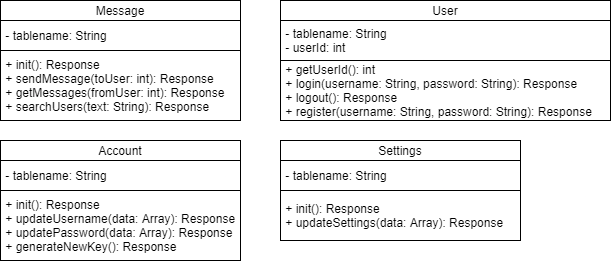


As you can see from the mock ups, the first page the user will be taken to after logging in is the image in the top left. It is a basic screen that will show you all your unread messages as well as how many unread you have in each chat. The menu is very similar but has a few extra items. These items include some account information and settings that you will be able to change such as username, notifications, and timeout settings. All pages also include a refresh button in the top right to manually refresh the page. Next, if you click on a chat that you are already in, you will be taken to a page similar to that of the image in the bottom right. It will show what messages exist in your current session that have not been erased and a box at the bottom to type in and send a new message. Finally, if you are not already in a chat you can create a new one by selecting “Start a new chat” which will take you to a screen where you are able to search for a user by their username and start a chat with them. They will then be on your landing page as long as a message history exists between the two of you.

Our design was very focused on mobile, since that is where we want our users to be for the most part. However, this application will be fully functioning on a desktop web browser as well.

|  |
| --- |
| Detail Solution Architecture |

This design will be composed on the front end of the web application, being made in Vue and utilizing Vuetify, a frontend library that consists of all the necessary components to create the proposed design. Since Vue is a Javascript framework, there are no classes that need to be created for this application as far as the front end work goes. All information that will be passed through it will be done through ajax calls to the backend of the application. For the backend, there are a few classes that will be in PHP. The great thing about PHP is that you can create a whole application without any real need for classes, but for the sake of ease of use when working with a relational database we created a few classes.



Graphical user interface, text, application

Description automatically generated

These PHP classes will cover all the needs for our application since we are keeping it so simple to begin with. For encryption of messages we will be using a randomized hash that will be run through md5 then stored in the database per individual user. This way, we can keep the messages safe and secure, unobtainable for anyone else even if they were able to retrieve the encrypted data.