CSR Project

cloud computing

Benjamin Cao, Jose Flores, David Jelley Jr.
Computer Science
UMASS Lowell
Lowell, MA United States

The cloud computing component of the CSR Project, a secure user based application that will manage data storage and presentation.

Administration Portal, Ajax, API, CSS, Developer Portal, Developer tools, Framework, HTML, JavaScript, jQuery, MySQL, PHP, User Portal

I. INTRODUCTION

A. Research Question

What is the best way to properly store, share, and present data in an access restricted multiple user application?

B. Goals

- Create an API to handle all requests.
- Create an expandable application framework.
- Create a secure database to store all data.
- Create a standardized user interface.
- Create management portals for developers, users, and Administrators.
- Create tools to help develop and test applications.
- Create a permission and user based system to allow private and shared access to data.'
- Render and display the information provided by external devices as well as application input.

II. MOTIVATIONS

Data is sensitive and can have different uses, some are informative but in certain hands data can be damaging. The complexity of the information contained within the data and their relationships with other seemingly unrelated data sets can be hard to detect. Finally data can be misinterpreted and displayed in ways that are deceiving. It is therefore important to properly handle, store, and display data.

With the amount of data that will be gathered by the project input devices, it is important to the project and the end users to be able to present the data and maintain trust. Therefore the cloud computing application strives to allow the projects data to be stored and rendered in a secure and visually appealing manner.

III. RELATED WORK

A. Similar Usage

Many existing businesses and research teams have shared the same focus. Medical, Financial and Social Media groups are great examples of user based websites, with high levels of security, that supply large amounts of data to their users in an organized way.

B. Examples for Comparison

- Stanford School of Medicine - Handling large user data and abiding by the HIPAA Security Rule[1]
- CareCloud – Provides health record, practice management, and billing/software services through the cloud. Data is neatly presented in a UI that is similar to the goals that we are trying to attain.[2]

C. What is still unknown

When it comes to information stored over the internet, privacy is of utmost concern. Today governments, businesses, and citizens fight over what is private and what is not on a daily basis. Security standards are evolving as current methods are overcome and security become obsolete. This group is therefore investigating how data should be stored, transmitted, and shared while maintaining privacy and trust between the user and the data holders.

IV. PROPOSED APPROACH

A. Plan

We will design a framework based on a work flow diagram of the application. The current iteration of the applications proposed work flow with future possible modification is shown in Figure 1. The framework will be designed with a custom compartmentalizing framework in mind so that public and private sections of the application are separate from each other.

Their access will be restricted to the client and server sides as depicted in the proposed framework structure in Figure 2. We will then Create an API interface to prevent unsolicited interaction with the application. Construct a database to store any data. And finally create an interface to manage user accounts.

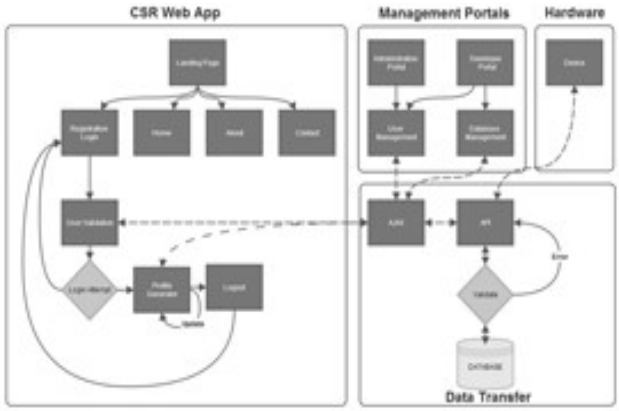

Fig. 1. Work flow

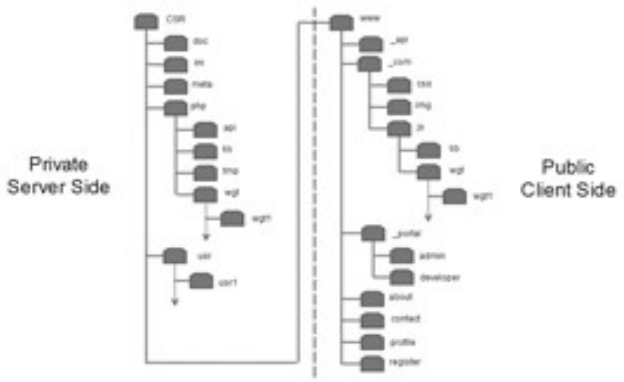

Fig. 2. Framework structure

B. *Features*

- An API
- A framework structure that allows for widgets.
- Multiple access portals.
- Database and data management tools.
- Sand boxed Developer testing tools.
- Permission based user accounts

C. *Implementation*

The application will be run off an AWS WAMP Server, client side code will run off JavaScript and jQuery libraries. PHP generated HTML and CSS will be the main components to the pages themselves.

We plan to design our website to take advantage of a permission based system that uses three primary levels of permissions. For ease of explanation the three levels are labeled tentatively as: doctor, patient, and observer. A user can have any or all levels of permissions. The idea is to create a dynamic permissions structure that will maintain privacy and security while still allowing the free flow of information where it is needed.

Doctor

- Able to view the records of all patients.
- Manage patients (Add, drop, update, transfer)

Patient

- Able to view their own records.
- Able to restrict view of data.
- Manage doctors (Drop, request)
- Manage observers (Add, drop)

Observer

- Manage who they observe (Drop, request)

V. EVALUATION

We will create unit test to evaluate all parts of the application separately and jointly, we want to validate and prevent application conflicts.

We will evaluate the user interface and data presentation through developer and peer focus groups to insure a wider spectrum of preferences, and data accuracy.

Most importantly we must test the privacy and security of our solution. We will validate account access and attempt known malicious intrusions to probe for weaknesses.

VI. TIME LINE

TABLE I. TIME LINE

Week	Goal
1	Proposal, presentation, and framework developed
2	AWS WAMP environment set up. Website template constructed. Developer tools constructed. Database finalized
3 – 4	Create user validation scripts Meet with device groups to discuss API needs. Create API Testing and Bug fixes. Meet with device groups to discuss API capability
4 – 7	Research PHP/ jQuery UI and graphing widgets Create PHP web pages. Create application access portals. Write JavaScript and jQuery client side scripts. Testing and Bug fixes.
8	UI testing focus group. Testing and Bug fixes.
9	Finish Styling and any animations
10	Testing and Bug fixes

Week	Goal
1	Proposal, presentation, and framework developed
11	Final testing and submission.

REFERENCES

[1] "Cloud Computing." *Stanford School of Medicine*. N.p., n.d. Web. 20 Feb. 2014.

[2] *CareCloud*. N.p., n.d. Web. 20 Feb. 2014.