*Chang Liu, Kelong Wu*

*University of Illinois at Urbana-Champaign*

*Beckman Institute for Advanced Science and Technology*

Grain Portal Project

Interim Report 1

**Table of Contents**

[1. Project Overview 2](#_Toc472764310)

[2. System Overview 4](#_Toc472764311)

[3. API Development Progress & References (Draft) 5](#_Toc472764312)

[4. Frontend Development Progress & References (Draft) 6](#_Toc472764313)

[5. Function Description for Web Frontend (Draft) 7](#_Toc472764314)

[6. Further Development Plan 15](#_Toc472764315)

# 1. Project Overview

The project basically attempts to implement most of the functionalities proposed by Asavari Tayal in her initial report. In addition, several other features are added per the requirements of researchers and farmers. On the other hand, some features are removed by design due to the technical challenges. While the purposes and functionalities of the portal are not altered, the entire website has been re-implemented with different technologies. In this section, we will briefly discuss the functional changes, as well as the technical changes.

**Functionality Changes:**

1. Considering the lack of Internet access in India, we decided to implement a fully-functional message-based interface. It allows farmers to interact with almost every part of the system without Internet, while in the original design, farmers still need to have Internet access for certain features. The list of features accessible with text messages are listed as below.

* Registration: Farmers would be able to create a new account through text message.
* Manage Sensors: Farmers would be able to bind a sensor to their account, and query the most up-to-date readings from the sensor.
* Alerts: farmers would be able to receive the automated notification from the sensor, if there is an abnormal reading. Messages directly pushed by researchers will also be delivered.
* Question Board: The question board is a Q&A platform allows farmers to post questions related to farming and get answers directly from field investigator or researchers.
* Reporting: Farmers can post reports of their products through a simple formatted message to the portal. This can help researcher to know more about the situation in the field.
* Selling: Farmers can post selling information of their products directly through a simple formatted message. Traders will be able to see those posts immediately online and contact farmers for more information.

1. As required by the researchers, we decided to implement a researcher’s portal besides the original one designed only for farmers and traders. This function will enable researchers to access all the data collected by the portal. For example, researchers may be able to look at each farmers’ profile and contact farmers directly. Researchers may also study historical information including the temperature and humidity created by sensors or the trading history in the market.

**Technical Changes:**

We decide to separate the frontend and backend completely. The advantages of this decision will be discussed in detail in the next section.

1. The entire server backend is re-implemented, due to the following reasons.

* Using Parse Server makes backend developments much easier.
* The original version was built upon SQLite database, which is not efficient and scalable.
* We have discovered that interacting with Twilio is hard with the original Django backend.

1. The web frontend is re-implemented with jQuery and Bootstrap. It has an all-new design, and also is responsive and faster.

* - jQuery technology enables fast and robust communication between frontend and backend. It requires small amount of code, and has very clean structure, improving maintainability as well as extensibility.
* - Bootstrap provides various APIs and templates for quick frontend user interface design reducing the workload for UI design and providing reliability as well as cleanness.

# 2. System Overview

**User Devices**

**APIs**

**Frontend**



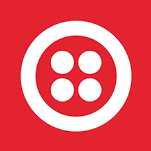
Messages



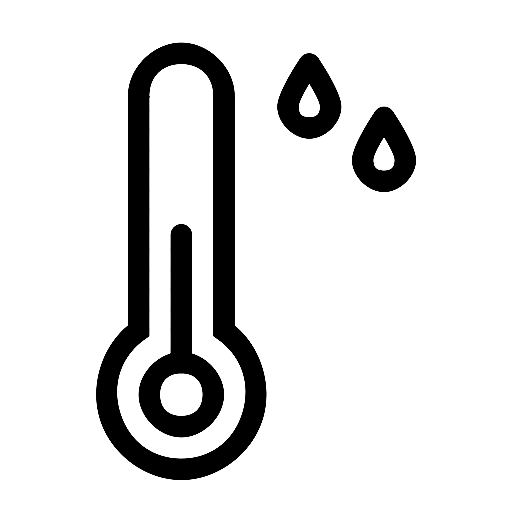
Browser



Web Frontend



Twilio



**Sensors**

/parse

Custom APIs

/twilio

**Database**

****

The framework of the system has been shown in the diagram below. There are two different ways for users to access the portal – either through browsers or text message. The sensors can push automated text messages to the portal for reporting.

For the normal access through browsers, jQuery is used to communicate with backend to store and retrieve requested information. For both sensors and test messages, Twilio API is used to receive messages as well as sending request to backend for further processing. The costumed APIs is built on node.js technology to provide various functionality. The Parse Server has been used to manage the database.

# 3. API Development Progress & References (Draft)

At mentioned above, we’ve replaced the original Django framework with Node.js and Parse Server for the purpose of better extensibility and faster speed. We built our customized APIs from scratch to handle different types of requests including user registration, notification system, sensor information process, reports, markets, and Q&A platform.

The API receives the requests from the frontend, parse the requests properly using Twillio API when necessary, and retrieve or store data to the database through Parse Server. Each major functionality is written in a separated node.js file to provide a clean and simple backend structure. Listed are node.js file names along with a brief introduction of its purpose.

* farmer\_report.js: including functions for farmer’s report creation and deletion
* send\_text.js: send text message from portal to a given cellphone number
* sersor\_report.js: parsing and storing reports sent from sensor
* sersor.js: find the specific sensor by some given information
* user.js: user creation, or find a specific user based on username or phone number

Overall, the Node.js provides a concrete way of building the backend. The Parse API enables us to manage and query our database easily. Twillio APIs gives us opportunities to provide major functionality through text messages in a clean and robust way.

# 4. Frontend Development Progress & References (Draft)

As motioned, the whole front has been redesigned. The main purpose of doing so is to separate the frontend completely from the backend for extensibility and cleanness. There are few major technologies that we used during the development process.

1. Jekyll. This simple tool enables us to write static webpages in separate modules, which great reduces the amount of code needed for static pages. Here’s a small example of why this can be a useful technology. Alost all the pages of the portal has a sidebar. If we don’t use this technology, then we need to add the code of sidebar in each HTML files, which can be a cumbersome process. With Jekyll, we simply write the code in a separate file, and then “compile” the whole project to get those static pages in seconds.
2. jQuery. This technology enables us to write stable and fast code for backend and frontend communication. For instance, a simple “list.js” file provides all functionality for the market list page including loading all reports from the database, submit a new report, create a table on a webpage to display a result, as well as provide some animation for better user experience.
3. Bootstrap. This technology provides enormous APIs and templates for user interface design. Hence, we can design the UI in a short time by applying carefully designed templates to our websites. For example, our main UI design comes from a template called SB Admin 2. It provides a beautiful layout as well as some great icons and visualization interface. MetisMenus is another useful add-on. It provides a sidebar which is a beautiful, reliable and easy for customization. By using Bootstrap, few lines of code can give us a user friendly interface in minutes.

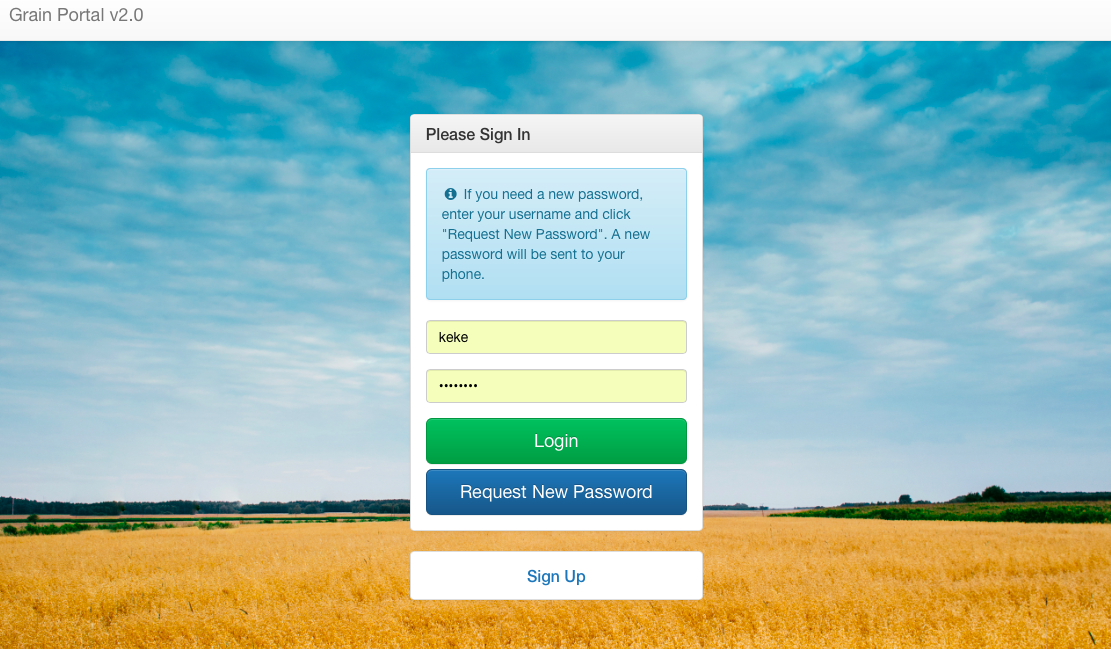
# 5. Function Description for Web Frontend (Draft)

Access

* Enter the address of the portal in a browser to access the portal
* <https://grainportalv2.herokuapp.com/>

Login / Sign Up

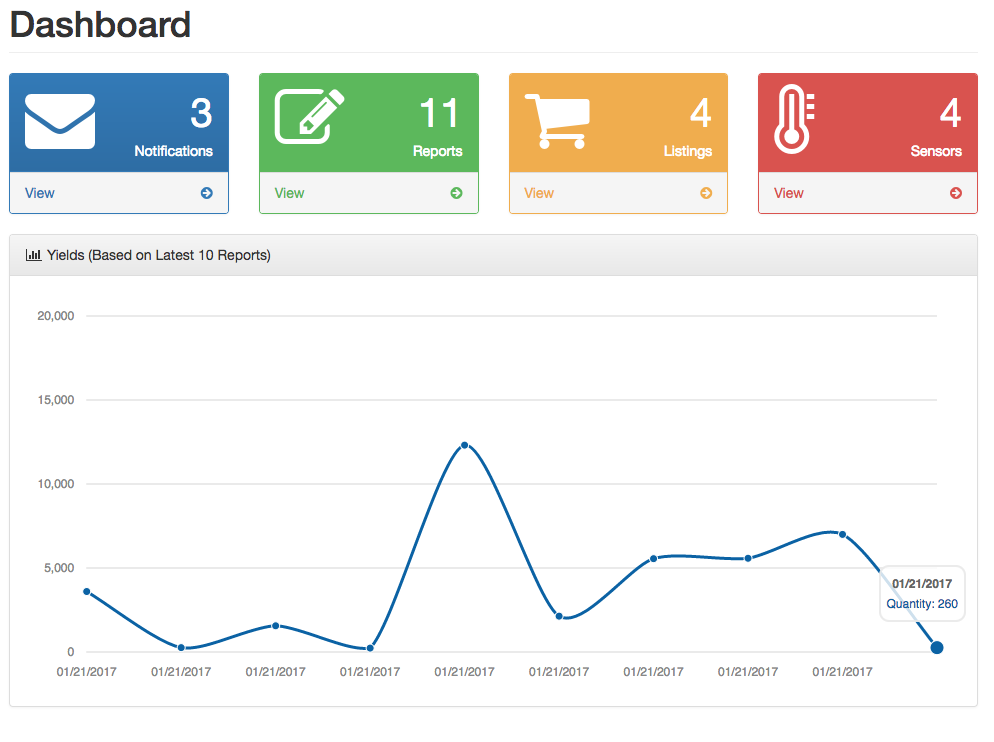
* User will be directed to the login page once the address is entered for login.
* Click the “Sign Up” button to create a new user



Login Page

Dashboard

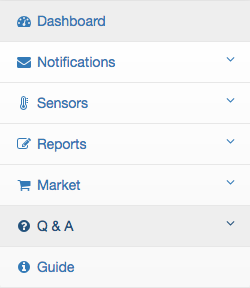
* User will be directed to Dashboard page once logged in.
* It is the “homepage” of the portal which provides an overview of user’s information, including numbers of notifications, reports, or sensor information.
* It also provides visualization of historical information from sensors.
* Click on any of the colored blocks to access corresponding functionalities.



Dashboard Page

Sidebar

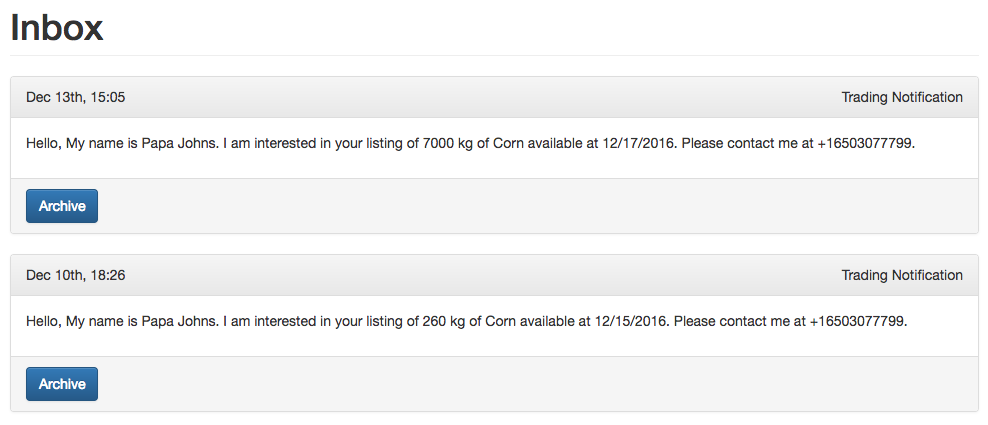
* It will appear at the same position on most pages
* User can easily access all the functions of the portal by clicking the items on the sidebar
* It is expandable and beautiful



Sidebar

Notification

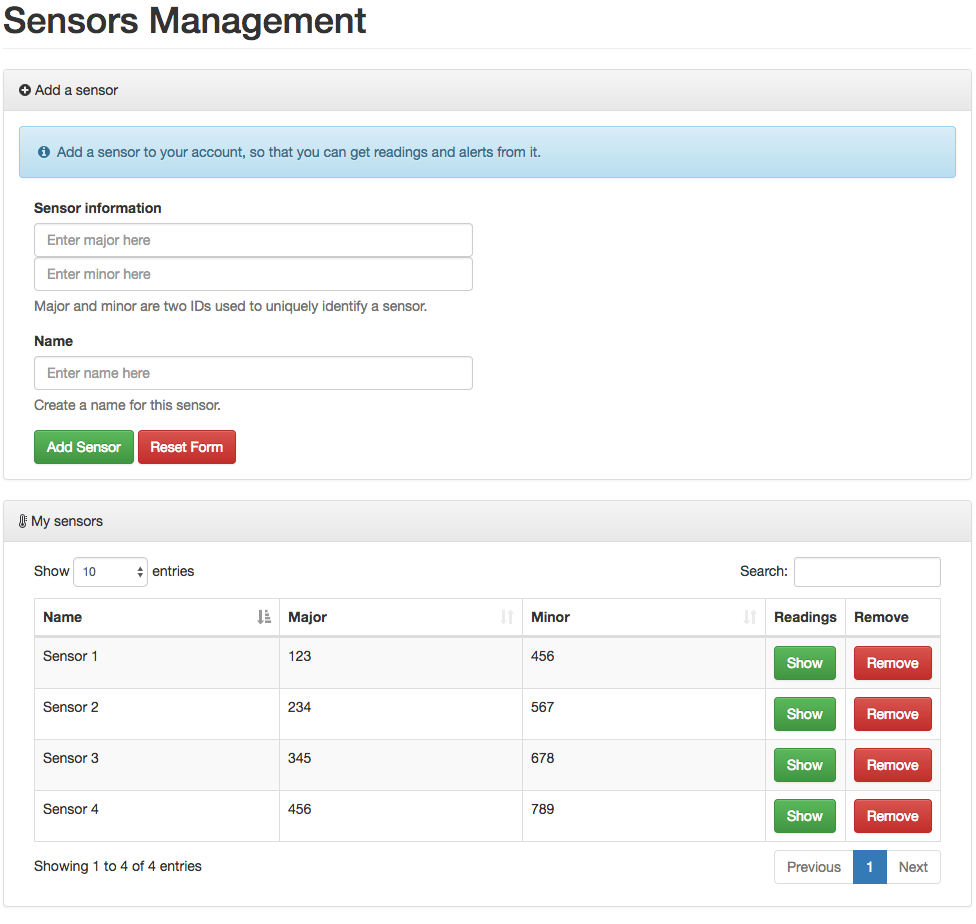
* Click the Notification Icon on the sidebar to access this function
* It has two parts – Inbox and Archived
* Inbox are notifications for farmers to read.
* Users can archive a notification once after reading it.



Inbox Page of Notifications Function

Sensors

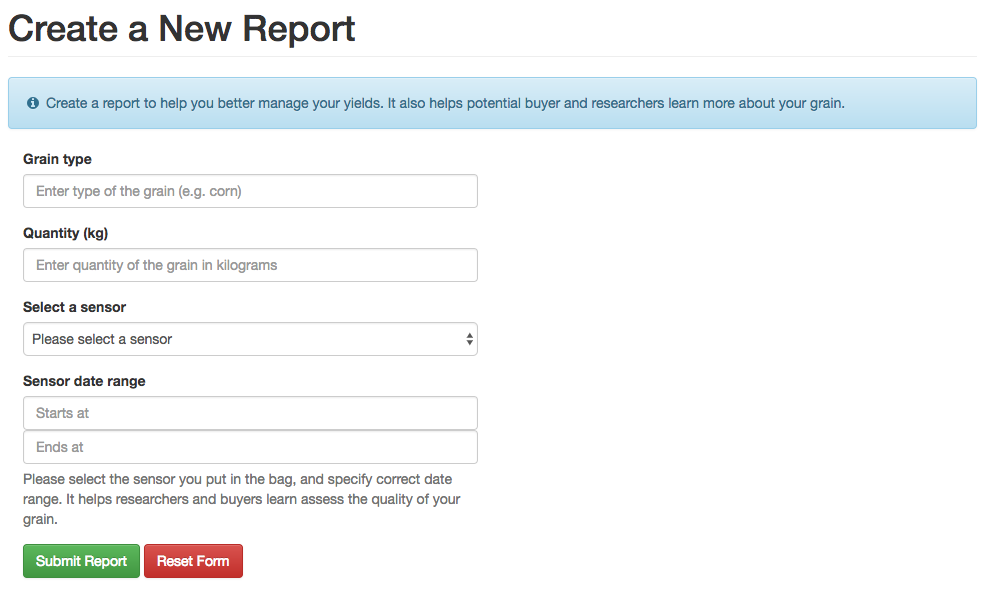
* Click the Sensors Icon on the sidebar to access this function
* It has two parts – Management and Readings
* Management provides a way of adding and deleting sensors
* Readings displays historical records of a specific sensor



Management Page of Sensors Function

Reports

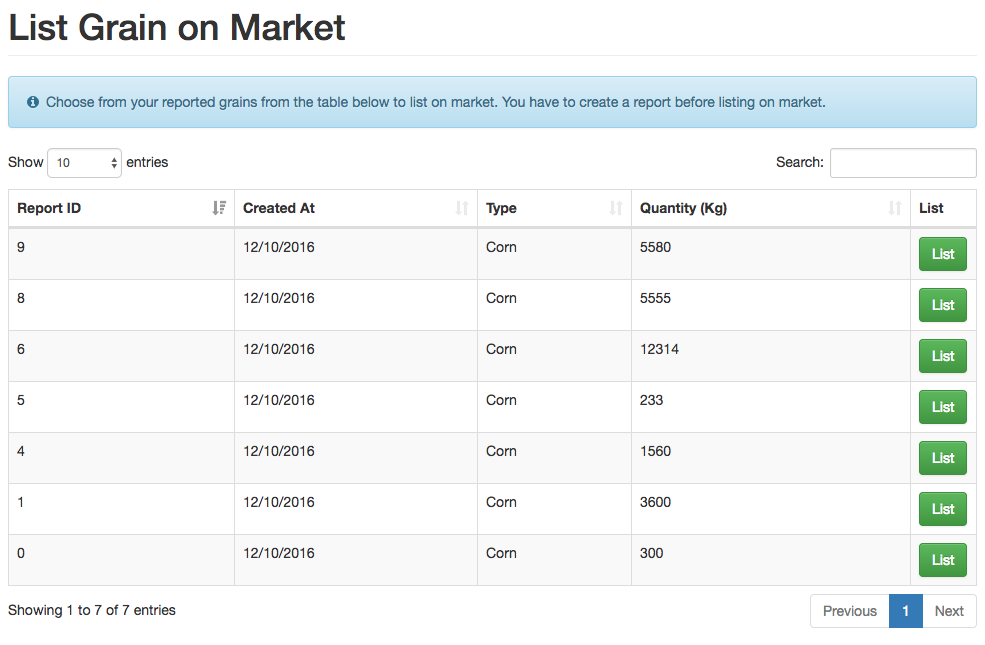
* Click the Reports Icon on the sidebar to access this function
* It has two parts – Create and View
* Create page is used to create new report
* View page is used to view created reports



Create Page of Reports Function

Market

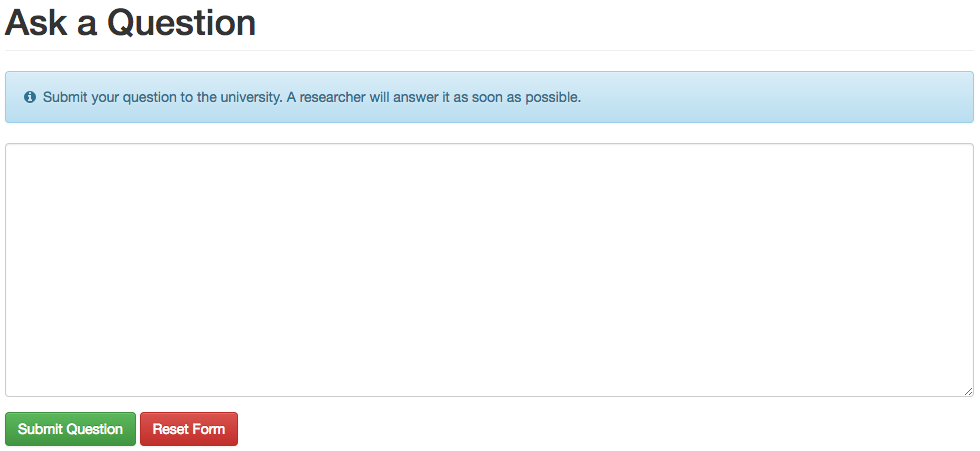
* Click the Notification Icon on the sidebar to access this function
* It has two parts – List and Selling
* List shows all available reports so that user can post them to the market
* Selling shows all posted reports on the market so that user can remove it from the market.



List Page of Market

Q&A

* Click the Q&A Icon on the sidebar to access this function
* It has three parts – Ask, Answered, Unanswered
* Ask is used to post a new question
* Answered displays all answered questions
* Unanswered displays all unanswered questions



Ask Page of Q&A

# 6. Further Development Plan

For the future development, we would like to provide various tools for researchers to analysis all the data. We’d like to create a blog-like page with WordPress for researchers or field investigators to put useful educational recourses on the portal so that farmers can learn more about farming through the portal.

Moreover, we might deploy the whole portal on our own host for better performance and large data storage. We will also closely with the team of sensor development to make sure those sensors work well with our portal. There will also be a integrated guide book on the portal.