# **On-Site Inspection Web App Development Report**

Table of Contents

[**On-Site Inspection Web App Development Report** 1](#_Toc110587602)

[1. Introduction 2](#_Toc110587603)

[2. Literature review 3](#_Toc110587604)

[3. Software Design 5](#_Toc110587605)

[3.1. Wireframe 6](#_Toc110587606)

[3.2. Database design 11](#_Toc110587607)

[3.3. Software design 12](#_Toc110587608)

[4. Software Development Documentation 13](#_Toc110587609)

[4.1. User story 1 13](#_Toc110587610)

[4.2. User story 2 16](#_Toc110587611)

[5. Testing 17](#_Toc110587612)

[6. Conclusions 18](#_Toc110587613)

[References 18](#_Toc110587614)

# Introduction

The site inspection web application is a helpful verification technique and confidence-boosting measure, and the procedure for the inspection needs to be comprehensive and efficient while also protecting the sensitive information of the construction site being Inspected. During an inspection, inspection teams will typically make use of paper checklists, forms, hand transcription of records, and note taking in order to collect information about the subject of the examination. This procedure could take a significant amount of time and is prone to errors caused by humans.

Recent advancements in mobile computing capabilities, such as cellular/wireless communications, audio-video recording capabilities, and streamlined user interfaces, offer the possibility of providing inspection team members and state parties that are being inspected with simple access to information and efficient data recording tools.

There is not a complete description of the benefits and risks associated with exploiting mobile computing capabilities for OSI in terms of tradeoffs. This report serves as a foundation for the description that was given. This research outlines different use cases for mobile technologies in order to provide support for on-site inspection. The potential advantages and drawbacks of each use case are analyzed.

Despite the fact that the capabilities of modern mobile devices like smartphones and tablet computers may differ substantially from one another, these gadgets often share a fundamental collection of qualities. For the purposes of this discussion, we will presume that inspection support tools made using modern devices equipped with touchscreens boasting a high resolution, cutting-edge mobile operating systems, and the ability to communicate via wireless networks.

Other features that are beneficial include a global positioning system (GPS), a compass or magnetometer, a thermometer, cameras with high resolution still and video capabilities, personal productivity applications, e-mail, web browsers, and hardware connectivity and expansion capabilities (via USB or other proprietary interfaces). New capabilities are constantly introduced by producers of electronic devices. Some of today’s mobile devices, for example, come pre-fitted with onboard radiation monitors, carbon monoxide detectors, infrared thermometers, or gas leak detectors. Other modern mobile devices have similar capabilities.

Among the capabilities that are presently approved for on-site inspection are transmission equipment, sampling tools, and containment devices (tags and seals). Some of these items are required to adhere to stringent information security standards including authentication and encryption. Because of its versatility, mobile devices might be thought of as an unanticipated hybrid of a communication device, personal assistant, global positioning system (GPS), camera, and video camera.

They are not so much novel as evolutionary because they are practical combinations of on-site inspection equipment that is widely used (or in certain cases, restricted). Mobile devices are formidable and useful thanks to the fact that they can be substantially changed and programmed; yet, it is difficult to safeguard, regulate, and functionally restrict the functionality of mobile devices.

Use cases were produced after an analysis was conducted to determine the ways in which on-site inspection tasks could be facilitated by mobile technologies. The on-site inspection activities included not only the inspection process itself, but also the training of escorts and inspectors, as well as post-inspection analysis. The inspection process was further segmented into the following stages: notification of the inspection, pre-inspection planning and preparation, travel arrangements, pre-inspection briefing, actual inspection, post-inspection briefing, and the drafting and submission of the inspection report (DTIRP, 2008).

During an inspection, some examples of actions that may be performed include seeing specific structures and facilities, documenting observations by taking pictures or videos, collecting samples, and evaluating documents and records. After conducting an analysis of the ways in which mobile computing could facilitate the various on-site inspection operations, we narrowed our focus to use cases that had the potential to make a significant contribution.

These were information support for inspection teams, managed access, training, and sampling. Use case descriptions that follow may go off into territory that is not actually allowed, but their purpose is to stimulate thought.

The backend is written in PHP language. It has used laravel framework to make it easier to use many library. And the user interface of the application is developed using HTML, CSS and Bootstrap.

The database used for storing inspections, users data in MySQL database. The web server is used XAMPP and the code is written by using Visual Studio Code.

The List of User Stories are mentioned below:

* As a user (Site Inspector) I want to register
* As a user (Site Inspector or manager) I want to login
* As a user (Site Inspector) I want to view the count of all the active inspections in my dashboard page
* As a user (Site Inspector) I want to create a new site inspection
* As a user (Site inspector or manager) I want to view the site inspection list
* As a user (Site Inspector or manager) I want to edit or delete the created inspections
* As a user (manager) I want to register new site inspector
* As a user (manager) I want to view all site inspector as list

# 2. Literature review

Inspection and documenting of construction defects for buildings that are either being built or are currently in use involve a lot of time and effort in the construction business. Project parties do site inspections when a building project is nearly finished to produce punch lists and address concerns before the building is turned over to its owner.

When serious problems are detected in an occupied structure, the owner, designer, contractor, and other parties may visit the place. Therefore, all parties involved including owners, designers, builders, and insurance companies are highly concerned about building flaws. According to published studies, the cost of ratifying faults ranges from 2 to 12.4 percent of the construction cost (Lundkvist et al., 2014).

Given the $975 billion in building value in 2014 (Census Bureau, 2015), the cost to American taxpayers of flaws would have been between $20 and $120 billion in 2014. Defects are a substantial factor to expensive and drawn-out construction disputes and litigation in the United States, in addition to the expenditure of repairs. Both the claimant and the defendant inspect the property and present their findings and inspection results in a construction defect dispute or litigation. The existing inspection, defect classification, and documentation approach is, regrettably, time- and money-consuming.

Several applications for site inspection are accessible across several platforms. Defects app was arguably the most pertinent app. iOS 6.0 or later was required for the Defects app (Contractors Apps, 2015). Although this application can make PDF reports, it didn’t seem to be able to create reports in a more flexible and configurable format, like csv (comma-separated values) files. Additionally, it appeared that the Defects app did not permit associating responsible parties to defects. In Site Inspector offers capabilities that are not present in the current apps as well as extra functions that help with effective site inspection.

The Main Menu, Inspection, Location, Concerned Party, Defect, and Report classes are among the six options contained in on-site inspection activities. The built-in GPS package is utilized in conjunction with these Android activities to locate the position of the site and any difficulties. When a defective image is recorded by the camera, they engage with the Image Handling package. For picture subsampling, the Image Handling option is used. That is, InSite Inspector will automatically assess how much memory will be utilized to load a problem and, if necessary, reduce the image file and/or alter the resolution.

Finally, for restoring and reporting inspection data and information, Android Activities interface with Data, Database, and File Management packages. The inspection data was transformed into a SQLite database. This information is maintained in “.db” files that are hidden on the system.

The inspection data (such as defect photographs and locations) are written to or retrieved from internal or external device storage using the File Management package. The following two portions, which each involve a case study, go into considerable detail on the two primary modules, Site Inspection and Documentation and Reporting.

The case study took place in January 2015 and involved an inspection of an academic building’s façade on the authors’ campus.

# 3. Software Design

User interaction (UI) interfaces or views are constructed making use of pre-installed templates on ZoHo creator as it serves as the basis for a plethora of app development services. The invocation of built-in APIs and the communication between views and app services are both implemented by Bootstrap.

In other words, Bootstrap is used to send events that are triggered by user input to built-in APIs and App services, and Bootstrap is also used by App services and built-in APIs to supply data to views. In this approach, ZoHo creator makes a number of application programming interfaces (APIs) available for access to a variety of devices, including cameras, GPS sensors, audio, local storage, and the network.

Additionally, because of the simplicity of the platform, HTML5, and CSS was easy to use as the platform these languages for website app development services, it is quite straightforward for web developers to create micro-Apps.

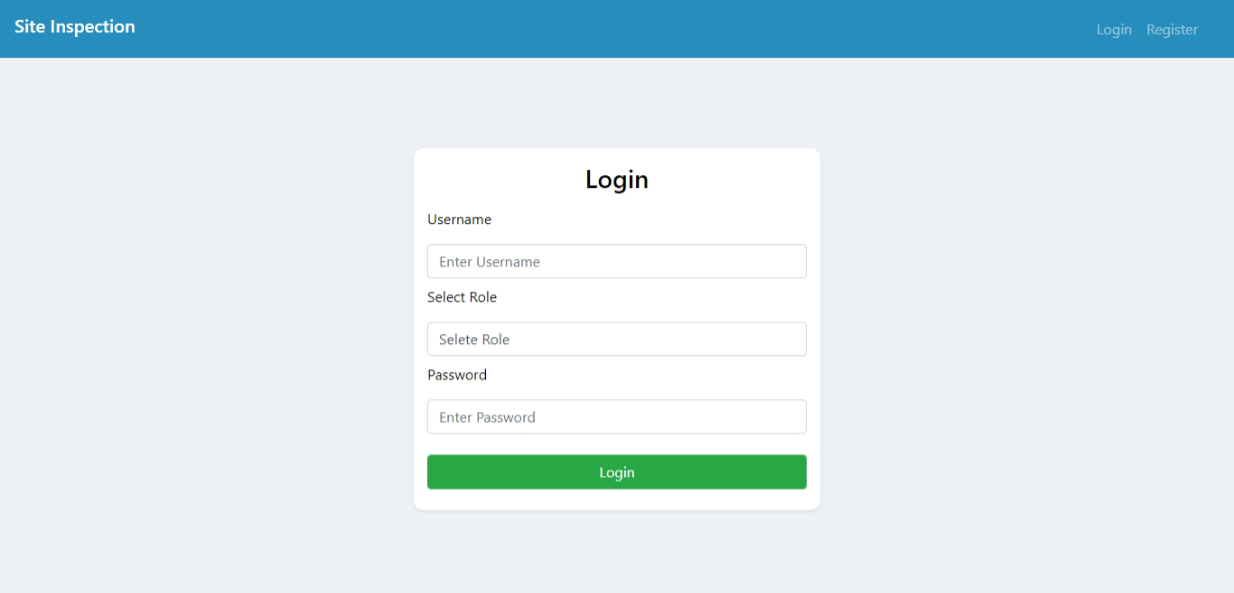
The ZoHo creator platform possesses a number of capabilities, including as GPS sensing, audio and video recording, reading QR codes, and uniform application programming interfaces (APIs) for developers, that reduce the amount of effort required for the development process. The performance of HTML5-based online applications is typically worse than that of native Apps, whereas the performance of micro-Apps is typically on par with that of native Apps.

This is because ZoHo, which has worked hard to improve the performance, has integrated all UI elements and their styles. When user interfaces and designs similar to those seen in ZoHo are utilized, it is easier for users to comprehend and use the application.

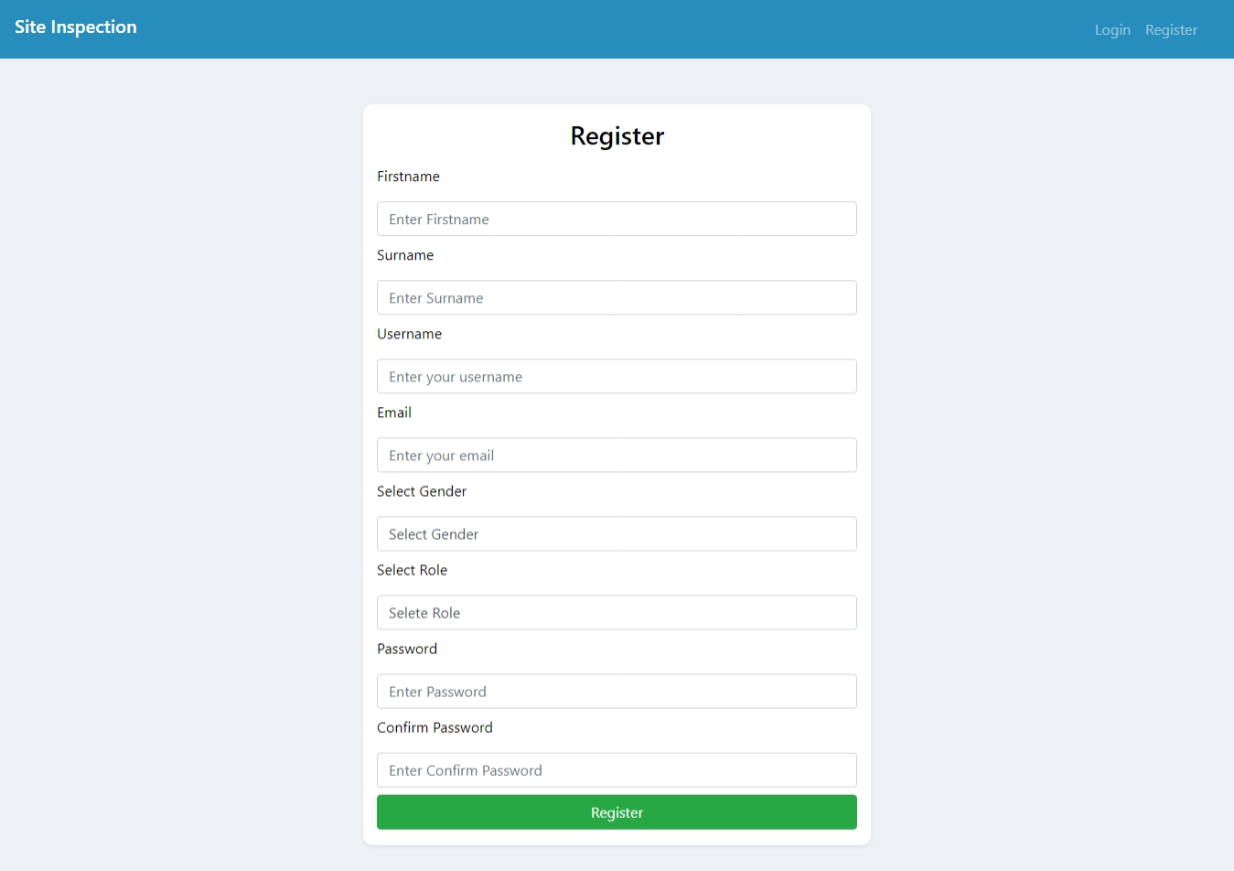
To prevent SQL injection, it is replacing (‘) to (\’). So, it is not getting any SQL injection attack at all. Also, for log of Testing done, PHP echo gave a lot help during building the website. Whenever I was not getting expected values then I was using PHP echo. Such as while registering users POST, inspection form saving, editing inspections, editing registered users, also sometimes used it to check SQL query for SQL injection and so on. When website is finished, I removed those log codes.

## 3.1. Wireframe

**Login**:

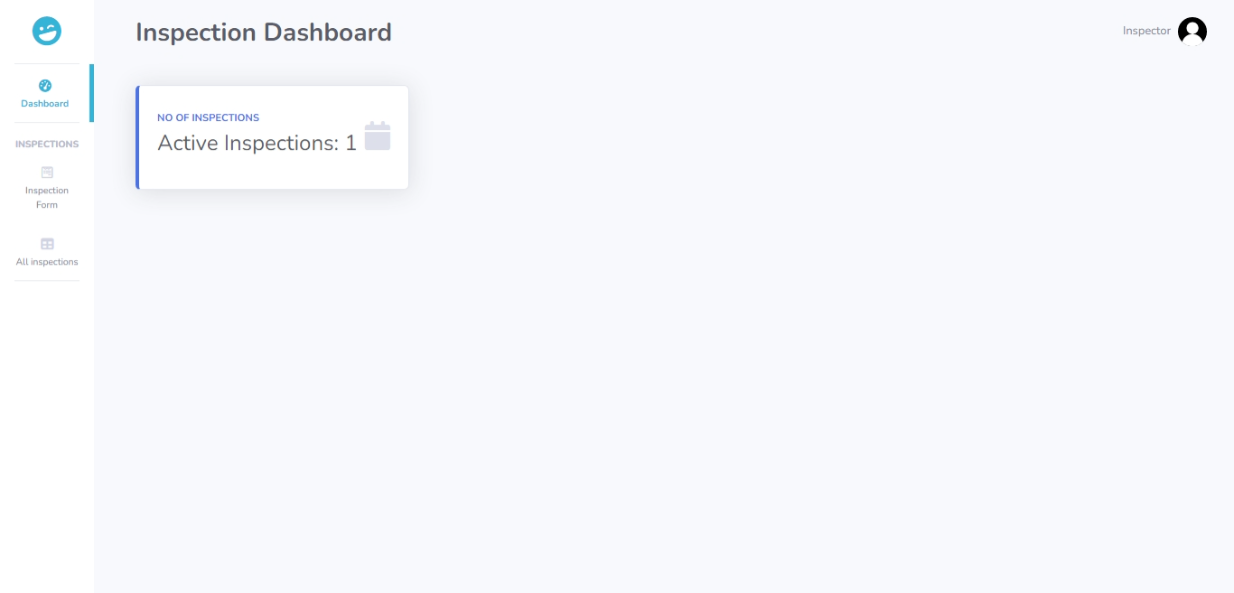


Here a user can login with Site inspector role and manager role. And he has to give username and password. At the top there has big size Login text. Under the big Login text there will show the status of login (Eg. error and success). And below of it there has username, select role and Password fields. And at the end there has Login button.

**Registration**:

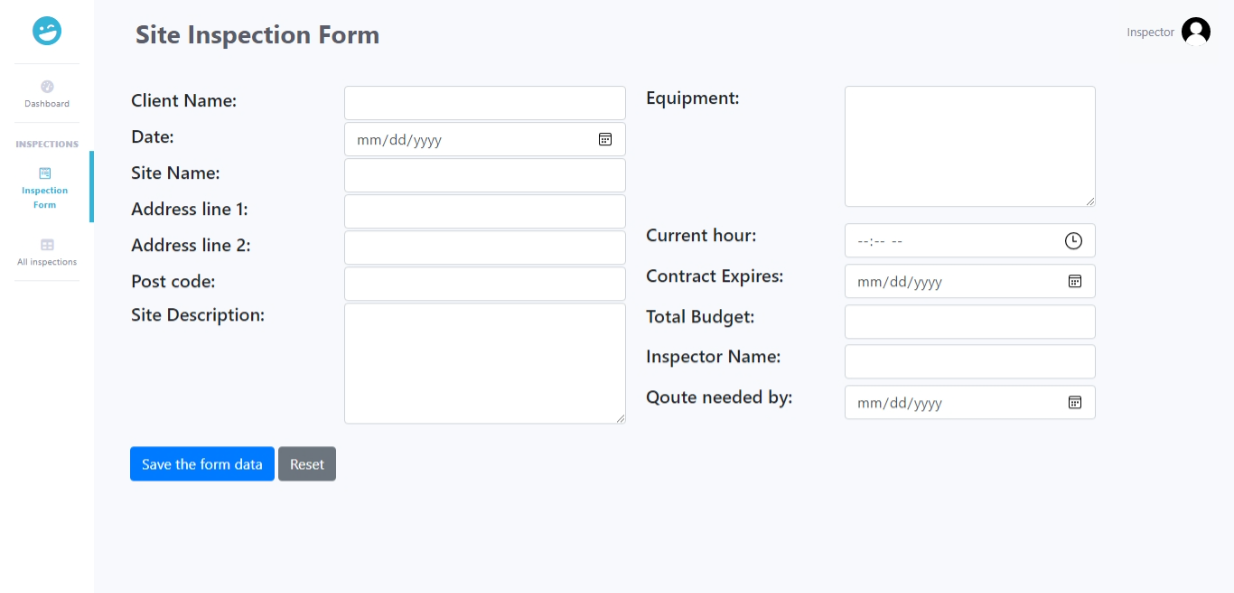
Here user can create Site Inspector account. And all the error will show at the top or below the fields. Same as Login page at the top of it has big Register text. Under it will show Register status (Eg. error and success). Under the Register text there has Firstname, Surname, Username, Email, Select Gender, Select Role (only Site\_Inspector), Password, Confirm Password input fields. Here Password and confirm password are password type input fields. And others are text type input fields.

**Site** **Inspector** **Dashboard**:



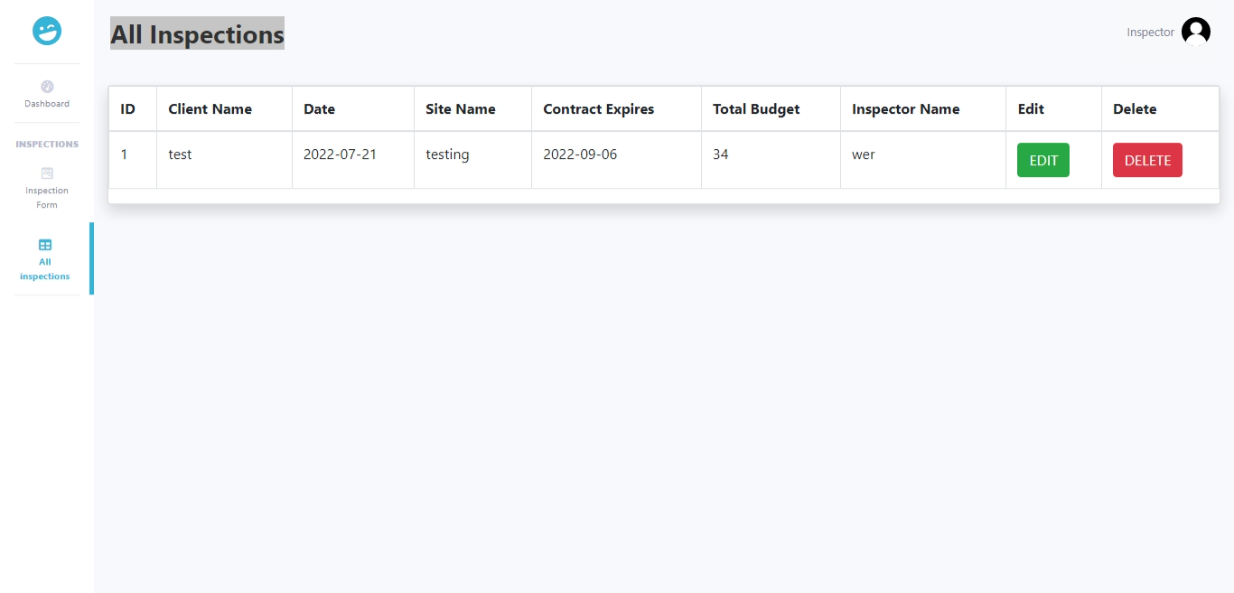
After login with Site inspector role user will be redirected to this page. Here he can see the total number of Active inspection.

**Site Inspection Form:**



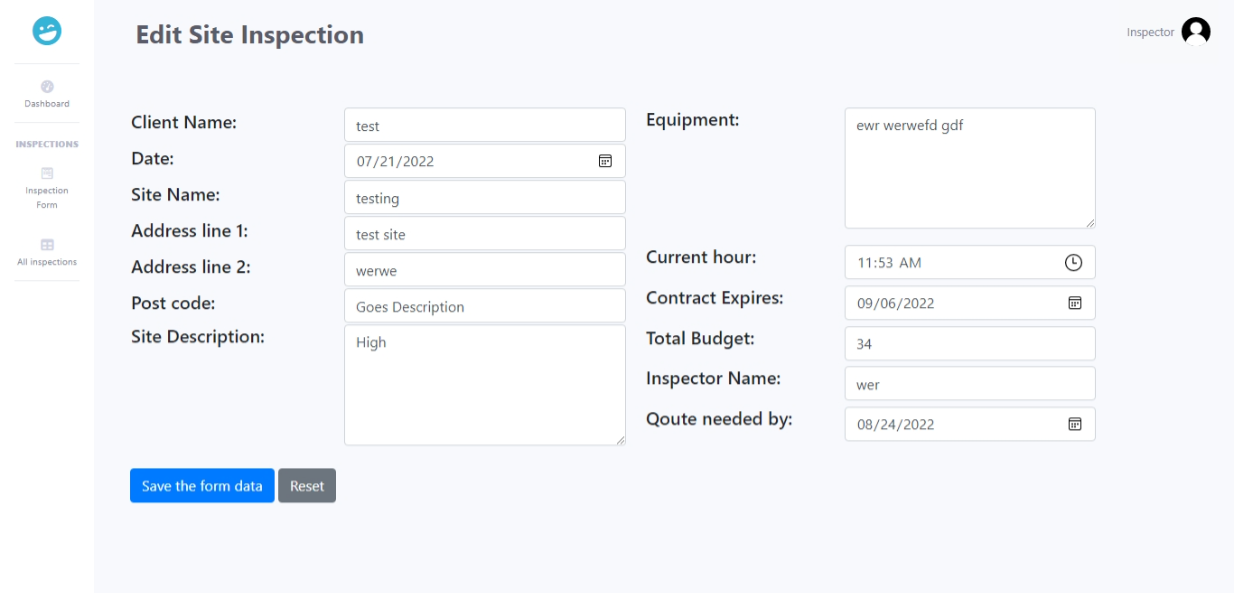
This page is Site Inspection Form. Here Site inspector can create inspections. All these information are saving into the inspection table database. Here Client name, Site Name, Address line 1 & 2, Post code, Total Budget and Inspector Name are text type input fields (One line Text) and Date, Contract Expires, Qoute needed by are date type input fields. And the Current hour is different type of field (Time).

**All Inspections:**



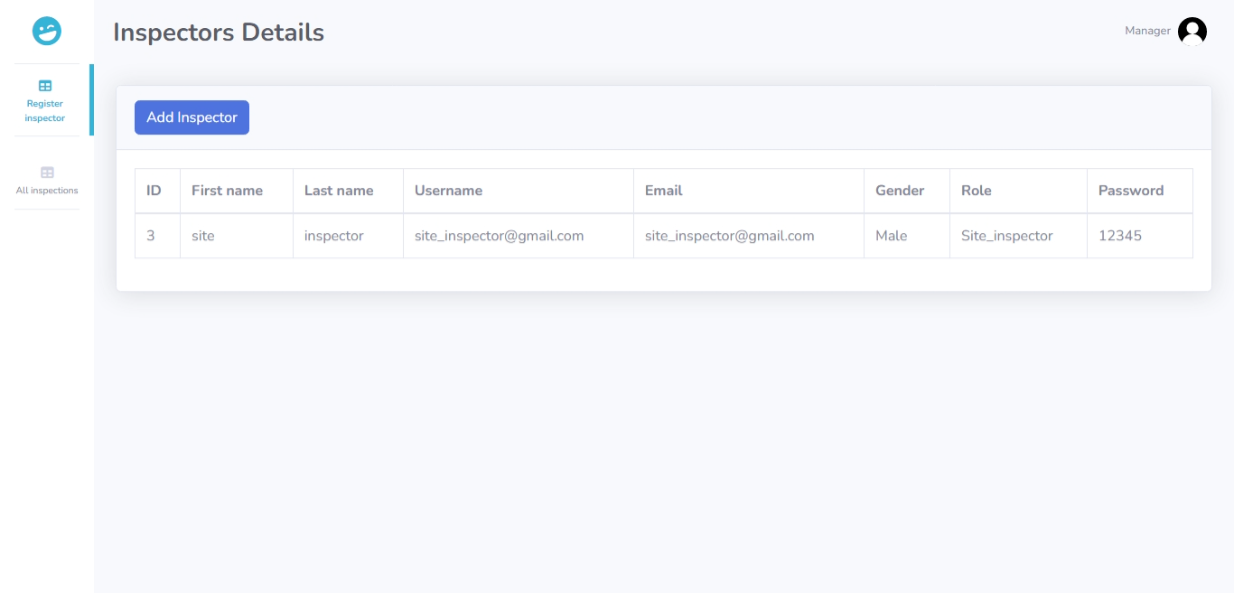
All the inspections are showing on this page. Site inspector and manager both can see this page. Here site inspector and manager can edit and delete inspections. All the inspections are showing in a table. In the Edit and Delete column all the rows will contain Edit and Delete button. And others column contains text.

**Edit Site Inspection:**



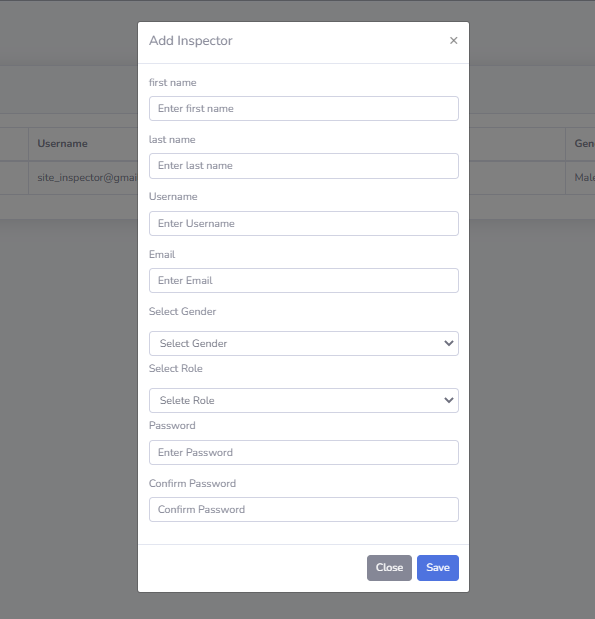
When site inspector and manager click on Edit button on ‘All inspections’ page, they will be redirected to this page. And here after he changed he will have to save in order to update.

**Inspector’s details:**



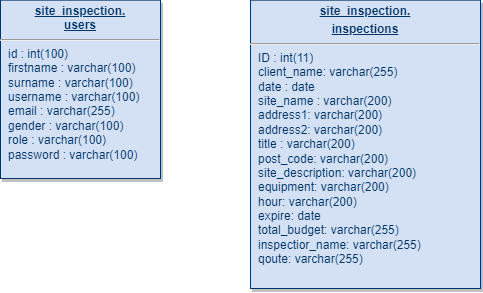
Only manager can see this page. Here manager can add inspector and view the site inspector list. All the site inspectors profile information will show here as list in a table. But manager can’t Edit or Delete. Only he can add a new site inspector.

**Add Inspector:**



After manger clicked on Add inspector this modal will show up.

## 3.2. Database design



MySQL database is used for database site inspection. Two tables are present. They are inspections and users. Id, first and last names, usernames, emails, gender, roles, and passwords are all listed in the users table. Additionally, the inspections table includes the following information: ID, client name, date, site name, Address line 1 and 2, title, post code, site description, equipment, hour, expire, total budget, inspector’s name and qoute. On localhost, the MySQL database is now active. Additionally, include/connection.php in the code contains the database connection.

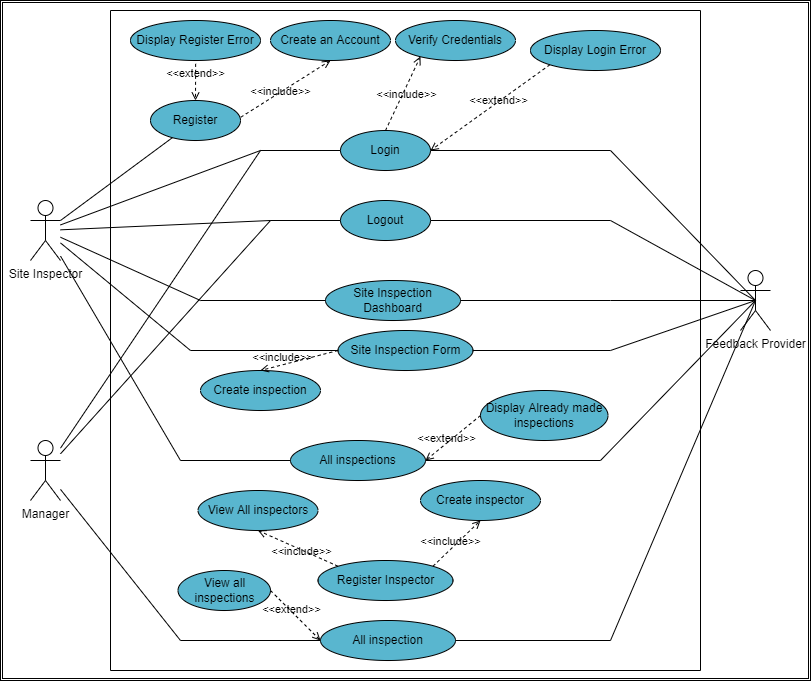
Each role's ID, Name, and perhaps additional characteristics should be listed in the table of roles so that we can grant each role access. The table role rel user, where each row's UserID and RoleID columns are filled with the matching IDs for each user and role, is the last place where the relationship between users and their related roles is recorded.

In order to distinguish between records for safety inspection and quality inspection, TypeID is a flag utilized. In the interim, the priority level, such as common or most important, and the status, such as pending or resolved, are stored in the columns LevelID and StatusID. The CreatedBy, OperatorIDs, and CC columns contain information on the creators, possible users who might be responsible for the problem, and users who should be informed of it.

These tables make it possible to properly gather and store data for users and on-site quality and safety assessment. The compilation of issue reports and further data analysis will help the construction management decision-making process.

## 3.3. Software design

**Use case diagram:**



User interaction (UI) interfaces or views are constructed making use of pre-installed templates on ZoHo creator as it serves as the basis for a plethora of app development services. The invocation of built-in APIs and the communication between views and app services are both implemented by Bootstrap.

In other words, Bootstrap is used to send events that are triggered by user input to built-in APIs and App services, and Bootstrap is also used by App services and built-in APIs to supply data to views. In this approach, ZoHo creator makes a number of application programming interfaces (APIs) available for access to a variety of devices, including cameras, GPS sensors, audio, local storage, and the network.

Additionally, because of the simplicity of the platform, HTML5, and CSS was easy to use as the platform these languages for website app development services, it is quite straightforward for web developers to create micro-Apps.

The ZoHo creator platform possesses a number of capabilities, including as GPS sensing, audio and video recording, reading QR codes, and uniform application programming interfaces (APIs) for developers, that reduce the amount of effort required for the development process. The performance of HTML5-based online applications is typically worse than that of native Apps, whereas the performance of micro-Apps is typically on par with that of native Apps.

This is because ZoHo, which has worked hard to improve the performance, has integrated all UI elements and their styles. When user interfaces and designs similar to those seen in ZoHo are utilized, it is easier for users to comprehend and use the application.

To prevent SQL injection, it is replacing (‘) to (\’). So, it is not getting any SQL injection attack at all. Also, for log of Testing done, PHP echo gave a lot help during building the website. Whenever I was not getting expected values then I was using PHP echo. Such as while registering users POST, inspection form saving, editing inspections, editing registered users, also sometimes used it to check SQL query for SQL injection and so on. When website is finished, I removed those log codes.

# 4. Software Development Documentation

## 4.1. User story 1

***Providing Informational Support to an Inspection Team***

On mobile devices, users have access to a convenient bundle that contains a wide variety of useful tools related to OSI. The use of mobile devices such as smartphones, tablets, and even wearable computers could be beneficial for a variety of the tasks involved in on-site inspections. The inspection team will have an immediate interest in a wide variety of communication channels, photography, videography, GIS applications for team tracking and orienting the inspectors on the ground, real-time information access, situationally-aware information feeds, and sensor monitoring, among other things.

Any facility or location that is considered to be sensitive will most likely have restrictions on the types of technologies that are allowed. Photography and videography give rise to unique worries regarding one’s personal safety because of the ability to visually record private or confidential events. In addition, there are some sites that will have restrictions placed on transmitting devices (Bluetooth, Wi-Fi, and cellular connections).

Some of the safety precautions have to do with the potential for contamination, while others are solely concerned with the use of electrical equipment in close proximity to combustible chemicals. In certain circumstances, it may be possible to negotiate the use of these devices if adequate preparation and administrative and engineering controls are put into place. Under some on-site inspection apps, there is a possibility of concerns with the movement of digital storage media away from the location.

Even in the case where the digital media must remain within the confines of a facility, it is possible that on-site storage of the digital media under seal is required in order to ensure that earlier inspection photographs are not altered.

Another option is to delegate management of the mobile device to the employees of the hosting or escorting organization. This would ensure that the tool is not being misused and that confidential information is not being inadvertently shared by accident.

By adding a gatekeeper into the communication process, this strategy prevents or, at the very least, slows the instantaneous and unfiltered flow of information to the inspectors. However, depending on the circumstances, it can be an acceptable alternative in some cases. Reviewing inspection locations, facilities, and regulations on a case-by-case basis will be necessary in order to determine whether or not mobile computers may be incorporated into a certain scenario at all.

The most recent generation of smartphones is able to provide users with an extensive feature set that is also highly portable, which will assist the inspection team in their work. Smartphones appear to be better suited as devices for transmitting data and recording data rather than for entering data directly into a computer system. Although reading a lengthy paper can be exhausting, keying in text that involves more than a few sentences can be done quickly.

By utilizing templates and forms that have been meticulously prepared with useful pull-down options and information that has been pre-populated, it is possible to improve both the user experience as well as the accuracy of data entry. Smartphones are ideally suited for a variety of data collecting tasks, including voice recording, still or video photography, and videography. The view screens that are currently available frequently reflect light, making them difficult to use when the sun is directly overhead.

Smartphone browsers make it simple to produce results from online database searches and questions asked of those databases. It is essential to make effective use of responsive design in order to make reading easier on smaller monitors. Apps that are specifically designed for a user’s device may provide additional benefits, such as the ability to retrieve previously cached data in the event that the device’s connection to the network or the phone is lost.

Tablets, in contrast to smartphones, feature displays that are significantly larger, greater memory capacities, and battery lives that are far longer. Despite these differences, tablets nonetheless provide many of the same features. The natural consequence of this is an increase in both size and weight.

A larger screen makes it easier to enter data because the virtual keyboard is larger and easier to read on it. This simplifies the process. Typing speed can be substantially increased with an external keyboard; however, this type of keyboard does require a table or another stable horizontal surface. The view screens on the most recent generation of mobile devices, such as smartphones, are susceptible to glare and can be difficult to operate in bright sunlight on occasion. If specialized voice-recognition vocabulary is developed, then form-filling and report-writing software that converts speech to type could be used.

Web browsers that are running on tablets are able to deliver results from internet searches as well as database queries with relative ease. Apps that are specifically designed for a user’s device may provide an improved level of user experience, including the potential to make use of previously cached data sets in the event that the device’s associated network is disrupted.

In spite of the fact that when utilizing extremely portable wearable gadgets, both hands are free to do other things, the devices’ restricted capability is due to their straightforward user interface. They can be of great use for taking still photographs, producing short video clips, and capturing audio of spoken notes, interviews, or other sounds.

Text-to-voice technology enables users to hear the device read text aloud to them through headphones or bone conduction transducers, making it possible for them to read massive amounts of information that would be hard to read on very small screens.

Programs such as Skype, Google Hangouts, or Apple’s Facetime make it possible to share your desktop with a connected device and participate in real-time video teleconferences that allow for two-way communication.

Finding a path and keeping track of it can be accomplished with the help of maps or satellite imagery that shows the location and direction in real time is part of what the on-site inspection app does.

## 4.2. User story 2

It is possible for cellphones equipped with GPS systems to alert one or both parties when managed access areas are getting close by, as well as track the whereabouts of the inspection party, with the help of an app that has been carefully built. There is a possibility that escorts will be made aware of the particular location as well as the nature of sensitive information, materials, procedures, and regions.

If an area cannot be visited by inspectors, they may be able to substitute a live video broadcast from a mobile device whose location and validity can be validated in its place. A video of this kind may be used as a substitute for restricted viewing of an area, but it would first need to be confirmed.

Sampling is yet another way of regulated access that can stand in for direct physical examination. Using a live two-way video feed, inspectors may be able to observe the collection of a sample from associated devices without having to physically enter the area being inspected. This would go a long way toward increasing public confidence in the method and make it possible to draw more trustworthy conclusions on compliance.

The platform allows for a telepresence inspection to be carried out by providing a link that is encrypted, authenticated, and capable of sharing voice, video, and data between a facility that is being inspected and virtual site visitors in a remote location. It is hoped that doing so will boost confidence. The objective of the project is to provide states that have had minimal past experience with international on-site inspections with an opportunity to learn about OSI mechanisms, managed access measures, and treaty verification. The system is designed to work best on either the iPad Air or iPad Mini as the primary computing platform.

If telepresence inspections conducted by users can be used in place of some proportion of physical inspections by foreign nationals, then mobile devices have the potential to actually lower the likelihood of the release of sensitive information while still revealing information that is relevant to the treaty. In the future, it will be helpful to do research to evaluate the benefits and drawbacks of telepresence versus physical inspections.

When the Inspected State Party is providing inspectors with a video feed, augmented reality technologies could dynamically hide or virtually redact certain features from the video in real time.

Validating the hardware and software systems would be something that both parties would have to do in order to feel comfortable with the procedure. While the viewer and the object were being moved around, prototype apps were able to monitor and dynamically obscure small 3D objects. This was done while the object was being viewed.

Alternative information may be easily supplied wirelessly from the inspection operations center and transmitted in a timely manner to teams operating in the field. This would eliminate the need to deal with cumbersome paper records.

# 5. Testing

The web application for on-site inspection was put through its paces by being tested with a number of different user logins, as well as by running a variety of on-site simulations, all of which resulted in good and immediate responses.

I have tested in Login, Register, Inspection form, displaying inspections, displaying all inspectors, editing inspection.

The complete test process are given in below table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Feature | Testing Scenario | Expected  Results | Actual  Results | Result |
| 1 | Login | Correct Username, Password & Role | Login Successful | Successfully  Logged in | pass |
| 2 |  | Correct Username, Password & Incorrect Role | Error | Invalid login credentials | pass |
| 3 |  | Correct Username, Incorrect Password & Role | Error | Invalid login credentials | pass |
| 4 |  | Incorrect Username & Correct Password, Role | Error | Invalid login credentials | pass |
| 5 |  | Empty Username & Correct Password, Role | Ask for input | username required | pass |
| 6 |  | Correct Username, Role & Empty Password | Ask for input | Password required | pass |
| 7 |  | Correct Username, Password & Empty Role | Ask for input | Role required | pass |
| 8 |  | SQL injection Attack | Prevent Attack | Invalid login credentials | pass |
| 9 | Register | Already Registered User | Error | Failed to register | pass |
| 10 |  | Empty all fields | Error | Firstname is Empty | pass |
| 11 |  | All filled & Empty username | Error | username is empty | pass |
| 12 |  | All filled & Empty email | Error | email is empty | pass |
| 13 |  | All filled & Empty password, confirm password | Error | Enter Password | pass |
| 14 |  | All filled & incorrect confirm password | Error | Both password do not match | pass |
| 15 |  | All are filled & all are new information | Success Registration | Registration  Successfully  Completed |  |
| 16 |  | SQL injection Attack and all valid | Prevent Attack and Register successfully | Registration  Successfully  Completed | pass |
| 17 | Site Inspection Form | All are Empty | Error | Date is Empty | pass |
| 18 |  | All are filled & Empty Site name | Error | Site Name is empty | pass |
| 19 |  | All are filled & Empty Client name | Error | Client Name is empty | pass |
| 20 |  | All valid information | Successfully Saved | Successfully Saved! | pass |
| 21 |  | SQL injection Attack and all valid | Prevent Attack and Register successfully | Registration  Successfully  Completed | pass |
| 22 | All Inspections | Show new added inspections after refreshing page | Showing including new inspections | Showing all inspections | pass |
|  |  | Edit specific inspection | Showing selected inspection | Showing selected inspection | pass |
|  |  | Delete specific inspection | Delete one inspection | Deleted one inspection | pass |

# 6. Conclusions

Through this project, I learned about PHP, HTML, CSS, JS language. Also it increased my Xampp, Composer, and better designing skills. It also let me understand about many advantages of Framework. Furthermore, I got a lot of user-friendly ideas of this application. Also I can fix many errors, bugs from now on.

# References

DTIRP. (2008). "The Arms Control Inspector." Product No. 406P. . *DTIRP Outreach*.

Gough, R., & Alan, Z. (1993). *Arms Control Compliance: Information Value of Verification Measures. SAND93-1347C.*

Lundkvist, R., Meiling, J. H., & Sandberg, M. (2014). A Proactive Plan-Do-Check-Act Approach to Defect Management Based on a Swedish Construction Project. *Construction Management and Economics, 32(11)*, 1051-1065.

Park, C., Lee, D., Kwon, O., & Wang, X. (2013). A Framework for Proactive Construction Defect Management Using BIM, Augmented Reality and Ontology-based Data Collection Template. *Automation in Construction, 33*, 61-71.