

CSC4350 Final Project

Name: Team Cuaderno – Team 7

Project Title: FVSRA IMS

Document Version: Final

Draft Version: Final

1. Problem Definition

1.1 Purpose (Josh V.) –

The purpose of this system is to create an implement a function incident management system for the customer, Fox Valley Special Recreation Association. Their current system is utilized with paper forms needs to be replaced with a web application for all employees and designated admins to use. The project sponsor, David Lash, has set a project length of two full sprints in an agile developing environment. The team members, Chandni, Josh Valentino, Germell, and Andy split the work into three categories: Front end, back end, and database. Germell and Josh worked on the front end, Andy worked on the back end, and Chandni worked on the database.

1.2 Audience (Josh V.) –

The people that will use this system fall into two main categories. The first being all FVSRA employees. The employees are able to login to the system easily and create a form for the designated incident that occurred. The second category is the admins. The admins are able to view any created forms as well as create forms.

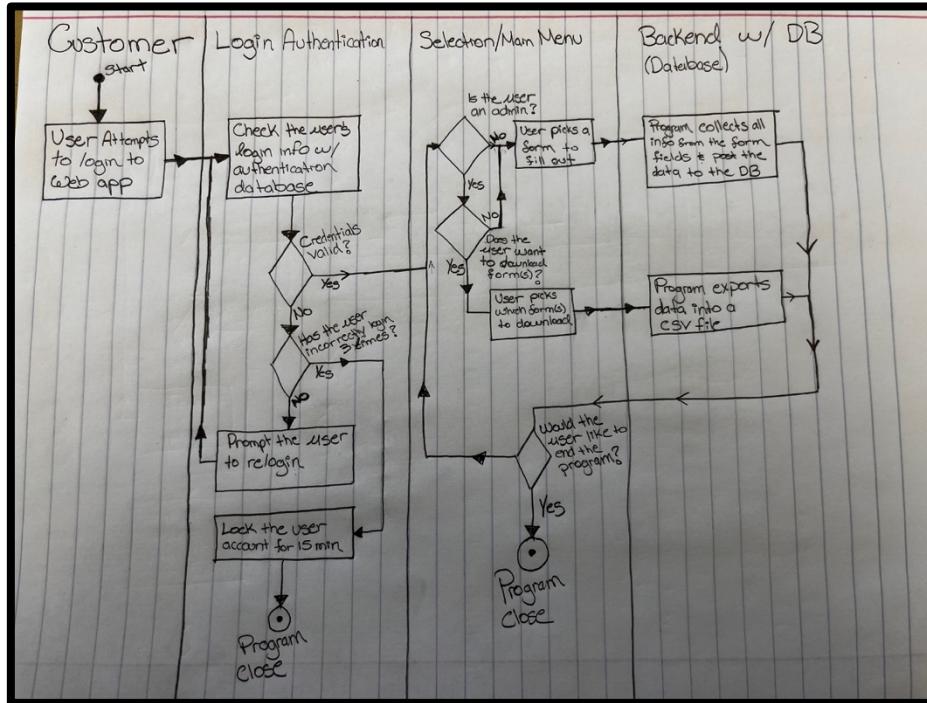
1.3 Scope (Josh V.) –

The scope of this projects includes the incident management system and storing the data for the reports. The system must be functional in terms of form creation and the viewing of the forms. The data stored must be viewable by the admins in the form of an excel file.

Out of the scope of this project would be the vehicle tracking system. This secondary goal was deemed impossible given the task and materials available, thus why it has been completely excluded.

2. Development Process. A definition of your development process that includes

a. Process Diagram -



The first step is the user goes to access the web app, and attempt to login with their credentials that was assigned to them by the manager. The program will check if the credentials that entered is valid. If it's not valid, it will check if this is the third incorrect attempt, it will lock the user's account for 15 minutes and end the program. If this isn't the 3rd unsuccessful attempt, it will prompt the user to try to login again. Once logged in, if this is an admin account there is an option to download form data, if they choose to do so, it will download it into a csv file. If they don't want to do that, they can also file a report with the form feature. If they're not an admin account, they can file a report with the form feature. If either a regular user or an admin chooses to do file a report, then the program will collect the info from the form fields and submit it into the database where the data will be stored, then the user can choose if they want to file another report or end/exit the program. For the admin account, once the admin chooses the form(s) to download, it will ask if they want to do something else. If not, it will end/exit the program. That concludes the activity diagram.

b. Product Backlog (Josh V.) –

AGILE PROJECT BACKLOG GROO								
Project Start Date:				Courtesy of: RS Ricksoft				
Priority	Task ID	Task Type	Task Name	Description	Sprint Ready	Status	Estimates Percent of Sprint	Assigned to Sprint
High	E1	Epic	Project Start up	Build out Infrastructure needed to start projec	Yes		0.8	No
High	T100	InfraStructure	Build DB connection	Write node.js methods that connect to test DB and verify credentials	Yes	Yes	0.2	Sprint1
High	T101	Inastructure	Build Github	Create github account, post document and ensure everyone can	Yes	Yes	0.2	Sprint1
TOTAL							0.4	
High	T102	Infrastructure	Build Database	Form_2 Accident_incident Form Creation and testing	Yes	Yes		Sprint1
High	T103	Infrastructure	Setup Web App Architecture	Setting up the architecture for the app using REST	YES	Yes		Sprint1
High	T110	InfraStructure	Get Element IDs and store into object	Javascript get element ID values need to be stored into an object	Yes	Yes		Sprint2
Low	T104	Infrastructure	Front End: update design	HTML, CSS, JS for fixing the cartoonish design	Yes	Yes		Sprint3
High	T105	Infrastructure	Build Database	Form 3 and 4 Creation and testing	Yes	Yes		Sprint3
High	T106	Infrastructure	Create entities through Node.js; Form 1	Javascript and Node.js testing with postman	Yes	Yes		Sprint1
High	T107	Infrastructure	Read entities through Node.js; Form 1	Javascript and Node.js testing with postman	Yes	Yes		Sprint1
High	T108	Infrastructure	Delete entities through Node.js; Form	Javascript and Node.js testing with postman	Yes	Yes		Sprint1
Low	T109	Infrastructure	Update entities through Node.js; Form	Javascript and Node.js testing with postman	No	Yes		Sprint3
High	T111	Infrastructure	Front End: Create form 1	HTML, CSS for Form 1	Yes	Yes		Sprint1
High	T112	Infrastructure	Front End: Create form 2	HTML, CSS for Form 2	Yes	Yes		Sprint1
High	T113	Infrastructure	Front End: Create form 3	HTML, CSS for Form 3	Yes	Yes		Sprint1
High	T114	Infrastructure	Front End: Create form 4	HTML, CSS for Form 4	Yes	Yes		Sprint1
High	T115	Infrastructure	Front End: Create form 5	HTML, CSS for Form 5	Yes	Yes		Sprint1
...	T116	InfraStructure	Data in form excell or pdf	Test ability of putting data into readable	Yes	Yes		Sprint4
High	T116	InfraStructure	Data in form excell or pdf	Test ability of putting data into readable excell or pdf for final	Yes	Yes		Sprint4
Low	T117	InfraStructure	View data for a form	HTML, CSS, JS for viewing data of a from that is updateable	No	Yes		Sprint4

High	T118	Infrastructure	Create entities through Node.js: Form 2	Javascript and Node.js testing with postman	Yes	Yes	Sprint1
High	T119	Infrastructure	Read entities through Node.js: Form 2	Javascript and Node.js testing with postman	Yes	Yes	Sprint1
High	T120	Infrastructure	Delete entities through Node.js: Form	Javascript and Node.js testing with postman	Yes	Yes	Sprint1
High	T121	Infrastructure	Create entities through Node.js: Form 3	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T122	Infrastructure	Read entities through Node.js: Form 3	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T123	Infrastructure	Delete entities through Node.js: Form	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T124	Infrastructure	Create entities through Node.js: Form 4	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T125	Infrastructure	Read entities through Node.js: Form 4	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T126	Infrastructure	Delete entities through Node.js: Form	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T127	Infrastructure	Create entities through Node.js: Form 5	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T128	Infrastructure	Read entities through Node.js: Form 5	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T129	Infrastructure	Delete entities through Node.js: Form	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T130	Infrastructure	Create entities through Node.js: Form 6	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T131	Infrastructure	Read entities through Node.js: Form 6	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
High	T132	Infrastructure	Delete entities through Node.js: Form	Javascript and Node.js testing with postman	Yes	Yes	Sprint2
Low	T133	Infrastructure	Update entities through Node.js: Form 2	Javascript and Node.js testing with postman	No	Yes	Sprint3
Low	T134	Infrastructure	Update entities through Node.js: Form 3	Javascript and Node.js testing with postman	No	Yes	Sprint3
Low	T135	Infrastructure	Update entities through Node.js: Form 4	Javascript and Node.js testing with postman	No	Yes	Sprint3
Low	T136	Infrastructure	Update entities through Node.js: Form 5	Javascript and Node.js testing with postman	No	Yes	Sprint3
Low	T137	Infrastructure	Update entities through Node.js: Form 6	Javascript and Node.js testing with postman	No	Yes	Sprint3
High	T-138	Database	Design Form.. 4	Design database, form4 and form5	No	In Progress	Sprint3
High	T-139	Database	Design Form.. 5	Design database, form4 and form5	No	In Progress	Sprint3
High	T-140	Database	Insert_records Form.4	Inserting records and testing data fields	No	Not Started	Sprint4
High	T-142	Database	Build Database	Design Form..1	Yes	Complete	Sprint1
High	T-141	Database	Insert_recrods Form.5	Inserting records and te	No	Not Started	Sprint4
Medium	T-142	Infrastructure	Front End: Admin View	Create the html, css, an	No	Not Started	Sprint3
Medium	T-143	Infrastructure	Front End: Admin Security	Create the html, css, an	No	Not Started	Sprint4
Medium	T-144	Infrastructure	Front End: Submission Page	Create the html, css, an	No	Not Started	Sprint3
Medium	T-145	Infrastructure	Front End: Admin Setting	Create the html, css, an	No	Not Started	Sprint4
Low	T-146	Database	Check Database	Go through db and shorten variable names and check variables types	No	Not Started	Sprint3
Low	T-147	Database	Add login credentials table	Add table for username and passwords	No	Not Started	Sprint4
Medium	T148	Infrastructure	Link the front-end forms w/ the Database: Accident Incident Report	Record the data from the forms input fields, save them into JSON and send it to the DB to be saved using AJAX	No	Not Started	Sprint5
Medium	T149	Infrastructure	Link the front-end forms w/ the Database: Employee Injury Report	Record the data from the Employee Injury Report form input fields, save them into JSON and send it to the DB to be saved using AJAX	No	Not Started	Sprint5
Medium	T150	Infrastructure	Link the front-end forms w/ the Database: Minor Injury Log	Record the data from the Minor Injury Log form input fields, save them into JSON and send it to the DB to be saved using AJAX	No	Not Started	Sprint5
Medium	T151	Infrastructure	Link the front-end forms w/ the Database: Notification Of Injury To Employer Report	Record the data from the Notification Of Injury To Employer Report form input fields, save them into JSON and send it to the DB to be saved using AJAX	No	Not Started	Sprint5
Medium	T152	Infrastructure	Link the front-end forms w/ the Database: Property Loss Report	Record the data from the Property Loss Report form input fields, save them into JSON and send it to the DB to be saved using AJAX	No	Complete	Sprint4
Medium	T153	Infrastructure	Link the front-end forms w/ the Database: Vehicle Accident Report	Record the data from the Vehicle Accident Report form input fields, save them into JSON and send it to the DB to be saved using AJAX	No	Not Started	Sprint5

C. Sprint Backlog (Josh V.) –

AGILE SPRINT BACKLOG											Courtesy of:	
			Project Start Date:		Sat, 1/01/1900						RS Ricksoft	
Priority	Task ID	Task Type	Backlog Task		Description	Reporter	Assignee	Status	Original Estimate	Start	Finish	
			Task Name									
High	T102,142	Infrastructure	Start Design Database		Form_2: Accident/Incident Form Creation, Form_1 Minor injury Log. Testing through looking phpmysadmin.	Professor Lash	Chandni	Completed	6			
	T103, T105-T107, T118-120	Infrastructure	Create, Read, and Delete entities through Node.js For Forms 1-2		Setup REST architecture w/ controller, model, & routes Create, Read, and Delete entities through Node.js For Forms 1-2	Professor Lash	Andy	Completed	8			
	T111-T115	Infrastructure	Front End: Begin Design of Forms		HTML, CSS, JS for the forms and fixing the cartoonish design. Testing through google chrome dev	Professor Lash	Josh + Germell	Completed	6			
	T104	Infrastructure	Front End: Update design		HTML, CSS, JS for the forms and fixing the cartoonish design. Testing through google chrome dev	Professor Lash	Josh	In Progress	4			
Sprint 2											33	
High	T111-T115, T104	Infrastructure	Front End: Start Create forms, Update Design		HTML, CSS, JS for the forms and fixing the cartoonish design, form 1, form 3, form 4E, Form 4. Testing through google chrome dev	Professor Lash	Josh	In Progress	25			
High	T121-126	Infrastructure	CRUD Functions: Create, Read, and Delete For Forms 3-4		Javascript and API testing with postman	Professor Lash	Andy	Completed	8			
High	T111-T115	Infrastructure	Front End: Start Create forms		HTML, CSS, JS for the forms. Form 2. Minor Injury Log. Testing through google chrome dev	Professor Lash	Germell	In Progress	12			
High	T-138, 139	Database	Finish Design Database		Form_3_property_loss_report and Form_4_employee_injury_report. Testing through php my admin	Professor Lash	Chandni	In Progress	4			
Sprint 3											30	
Medium	T-142	Infrastructure	Front End: Admin View		Make the html, css, and js for the admin view screen functional. Update the styling to match the user log in screen. Testing through google chrome dev	Professor Lash	Josh	In Progress	10			
High	T127-132, T152	Infrastructure	CRUD Functions: Create, Read, and Delete For Forms 5-6, & Link the front-end forms w/ the Database: Property Loss Report		Javascript and API testing with postman, and setup a demo ready form that can collect the info and save it into an object so that it can stored in the DB	Professor Lash	Andy	In Progress	8			
Low	T144	Infrastructure	Front End: Submission Page		Make the html, css, and js for page that displays after a user submits a form. Make functions so it displays whether or not the form was submitted properly. Testing through google chrome dev	Professor Lash	Germell	In Progress	10			
Low	T146	Infrastructure	Check Database		Go through db and shorten variable names and check variable types. Testing through php my admin.	Professor Lash	Chandni	In Progress	2			
Sprint 4											24	
Medium	T-143	Infrastructure	Front End: Admin Security		Create the html, css, and js for the admin security. Testing through google chrome dev	Professor Lash	Josh	In Progress	6			
High	T148-151, T153	Infrastructure	Link the front end forms w/ the Database: All the other 5 forms		Record the data from the other 5 forms input fields, save them into JSON and send it to the DB to be saved using AJAX	Professor Lash	Andy	Not Started	8			
Low	T144	Infrastructure	Front End: Admin Settings and Notifications		Make the html, cs, and js for the page that displays the settings and notifications for the admins. Testing through google chrome dev	Professor Lash	Germell	In Progress	6			
Low	T146	Infrastructure	Add login credentials table		Add table for username and passwords. Testing through php my admin	Professor Lash	Chandni	In Progress	4			
Sprint 5											6	
Medium	T116-117	Infrastructure	View data through the app and download form data via CSV & PDF		Display and Download Data	Professor Lash	Andy	Not Started	6			
Total											117	

Note: We already had completed our sprint backlog by the time the updated version was provided to us on dec 1st. We did not have time to update it to the newer style

d. Sprint Review (Josh V.) –

In our sprint review meeting with professor Lash we gained insight on the timeline of our project. We were initially confused and

to be transparent, by the end still somewhat were. However, in the review we learned that there would be only two sprints.

In terms of the database we learned that we need more representation, including an ERD or some other form of diagram.

In terms of the forms, we already were aware, but it reinforced that our styling needed work.

In terms of the backend we felt that we had sufficiently met the mark at that given time.

e. **Sprint Retrospective (Josh V.) -**

In terms of what was good our group prevailed with a number of positive outcomes. The first being, the system functions. This is good because it shows that we understand the material well and are able to create a functioning prototype. The downside to this is that it is not a fully complete prototype. There are a lot of aspects that still need to be done in order to make it ready for deployment.

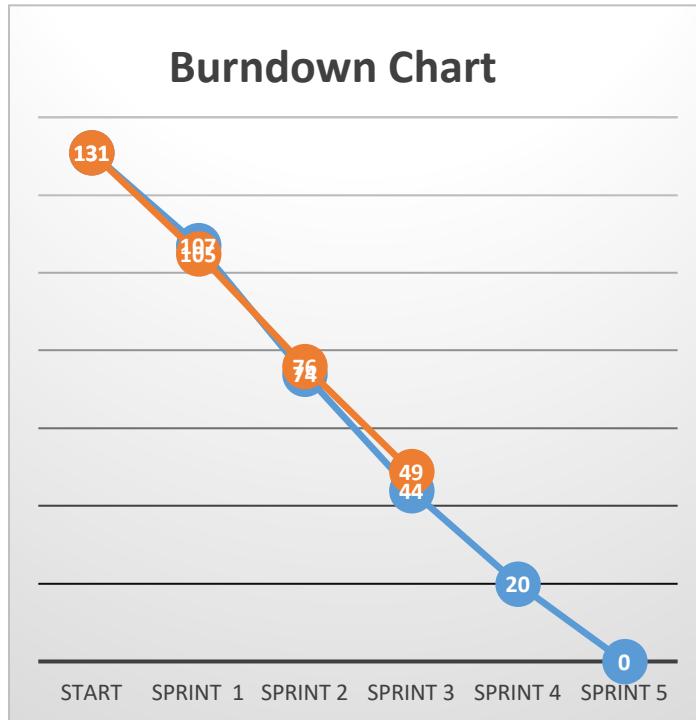
Another positive aspect of our sprint was our work ethic and communication in the last week. We were very focused and communicative due to the impending due date. Conversely, in previous weeks we were not always able to meet due to scheduling and we did not use discord that much.

Another positive that came from the project was the learning experience of making a functioning project for a real business. It was a great change of pace from the standard small homework assignments in other csc courses. The downside to this was things were unclear and confusing at times. We felt some things could be explained better and the timeline could have been more structured.

A great idea that comes to mind is splitting up the work via segment instead of end. In our project we split up the work by front-end, back-end, and the database. Instead, what we heard another group did, was they split up the work into forms. Each member did the front-end, back-end, and database table for their respective form. By taking action in the next group project, we could decide to spit up the work this way. One immense positive would be the ability for all members to learn all aspects of the system. Since we split it up

by end, some of our members did not get to learn the back-end which was taught in class.

f. Sprint Burn



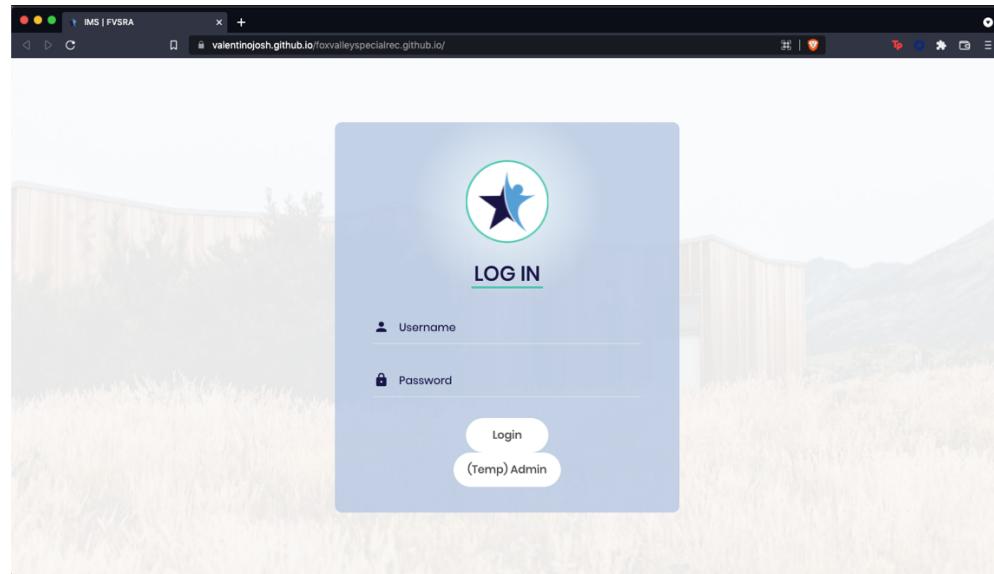
Total Backlog Estimated	Hours Planned for Current Sprint (Velocity)	Hours Planned Left	Hours Actual Worked	Hours Actual Left
Start	n/a	131	n/a	131
Sprint 1	24	107	26	105
Sprint 2	33	74	29	76
Sprint 3	30	44	27	49
Sprint 4	24	20		
Sprint 5	20	0		

We are about to finish on Sprint 3, which has a velocity of 33 work hours. We are a bit behind schedule since we underestimated how long some of the forms can be, such as the vehicle accident report with over 100 fields, which wasn't accounted for. In the first 2 sprints we were on schedule, with the first sprint estimating 24 hours and we performed 26 actual hours, but on sprint 2, we

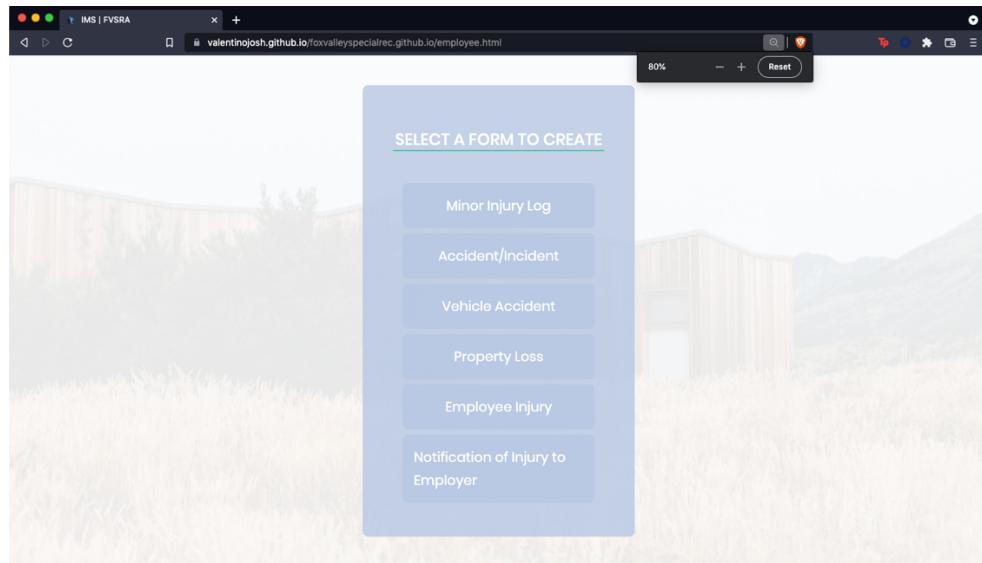
underestimated how much work some of the forms required and had a few hiccups along the way. We were short by 4 more work hours (33 planned hours), but we just carried that work to sprint 3. For sprints after 3, we project that it'll take another 49 hours for the project to be fully functional and be able to be deployed.

3. Project Results (Josh V.) –

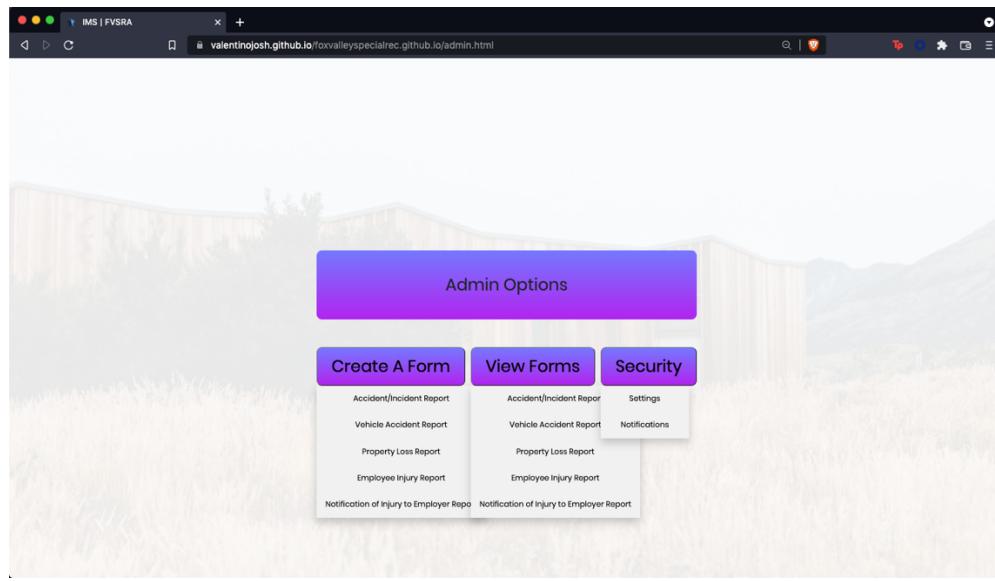
a. A solid implementation of each major project UI view (Josh V.) –



The figure above, figure one, displays the user login page. The FVSRA logo is featured above the log in text. The username and password fields are dignified with symbols and placeholders. Underneath the text fields are two login buttons. The login button is for employees and the temp admin button is for admins.



The figure above, figure 2, displays the employee form selection page. Each of the forms are able to be selected and viewed for creation.



The figure above, figure 3, displays the current front-end for the admin selection page. The create form, view forms, and security button are able to be clicked resulting in a drop down of options.

The screenshot shows a web-based form titled "Minor Injury Log | FVSA". At the top, there's a header with the PDRMA logo and the text "Park District Risk Management Agency". Below the header is a table with eight columns: Date, Time, Name of Injury, Injury Location, Treatment, How injury occurred, Facility/Location where injury occurred, and Full Name of staff who noticed guardian. The "Date" column contains fields for month, day, and year. The "Time" column contains fields for hour, minute, and second. The "Name of Injury" column has a dropdown menu showing "i.e Incision". The "Injury Location" column has a dropdown menu showing "i.e Right Leg". The "Treatment" column has a dropdown menu showing "i.e Bandage". The "How injury occurred" column has a dropdown menu showing "i.e Slipped on wet floor". The "Facility/Location where injury occurred" column has a dropdown menu showing "i.e Food Court". The "Full Name of staff who noticed guardian" column has a text input field containing "John Doe". A large "SUBMIT" button is located below the table.

The figure above, figure 4, displays the minor injury log. The submit button is below the form, when clicked it will process the form and redirect the user to the submission page.

The screenshot shows a web-based form titled "Accidental Incident Report | FVSA". At the top, there's a header with the PDRMA logo and the text "Park District Risk Management Agency". Below the header is a large form titled "PDRMA Accident/Incident Report Form 01". The form includes sections for "Incident Information", "Injured Person Information", "Witness Information", and "Medical Information". It contains numerous text input fields, dropdown menus, and checkboxes. For example, in the "Incident Information" section, there are fields for "Date of incident", "Time of incident", "Place of incident", and "Type of incident". In the "Injured Person Information" section, there are fields for "Name", "Address", "Phone number", and "Email". A "SUBMIT" button is located at the bottom of the form.

This screenshot shows the first page of a two-page accident report form. The title is "PDRMA Accident/Incident Report Form 01". It includes sections for "Who did you administer?", "First aid administered by agency personnel", "Details of the agency affected body and type of injury (e.g., collision, smoke, vibration, sprain, break, etc.)", and "Did first aid involve CPR?". Below these are fields for "Name and address of person who administered first aid", "Phone number", and "Email address". A note states "If yes, choose medical or PDRMA medical AED form." There is also a section for "Officer and vehicle involved in agency response" and "Details of request and callout".

The figures above, figure 5 and 6 respectively, display the accident/incident report. There is no submit button because this form is not functional.

This screenshot shows the second page of a two-page vehicle accident report form. The title is "PDRMA Vehicle Accident Report Form 02". It includes sections for "Agency name", "Date of incident (month/year)", "Time of incident (hour/min)", "Location of incident (city/town)", "Incident details", "How did the incident occur? (Provide a brief factual description, do not speculate or fault find.)", "Name of the location (park, post, community center etc. Smith Post Johnson Community Center or nearest intersection)", "Is there an address for incident location? (If no, provide a detailed description of the location)", "Address", "City", "State", "Zip code", "Phone number", and "Email address". A note states "If yes, choose medical or PDRMA medical AED form." There is also a section for "Officer and vehicle involved in agency response" and "Details of request and callout".

Vehicle Accident Report | FVS | +

valentinojosh.github.io/foxvalleyspecialrec.github.io/forms/VehicleAccidentReport.html

PDRMA Vehicle Accident Report Form 02 (pg. 2)

1 Who first aid administered? Yes _____ No _____ Unknown _____

2 Name and position of person who administered first aid _____

3 What first aid was given? _____

4 Did first aid involve AED and/or CPR? Yes _____ No _____ Unknown _____

5 Yes, please submit a PDRMA post-AED form.

6 Who provided services (if other)? _____ Offered and called _____ Offered, refused, called by agency anyone _____

7 Offered and refused _____

8 Who police called _____

9 Were police called? Yes _____ If yes, provide the following information _____

10 Name of police department _____

11 Name of officer _____

12 Did you expect this person to submit a claim? Yes _____ No _____ Unknown _____

13 Did you expect this person to call a hospital or medical facility? Yes _____ No _____ Unknown _____

14 Did you, or was the person who called you, involved in the incident? _____

15 Name of property adjacent to park district _____

16 Vehicle Owner _____ Other _____

17 Last name (or business name) _____ First name (if necessary / business name) _____

18 Address _____

19 City _____ State _____ Zip code _____ Phone number _____

20 Describe the property damage _____

21 Was there a witness(es) to the accident/incident? Please provide the following information:

22 Last name _____ First name _____

23 Address _____

24 City _____ State _____ Zip code _____ Phone number _____

25 Did witness take any statements? Yes _____ No _____ Unknown _____

26 Did you, or did the witness say? _____

27 Where was witness when the accident/incident occurred? _____

Vehicle Accident Report | FVS | +

valentinojosh.github.io/foxvalleyspecialrec.github.io/forms/VehicleAccidentReport.html

PDRMA Vehicle Accident Report Form 02 (pg. 3)

1 Agency name _____

2 Date of incident (month/year) _____

3 Time of incident (month/day/year) _____

4 Name of person completing the report _____

5 Contact phone _____

6 Contact email _____

7 How did the incident occur? (Provide a brief, factual description; do not speculate on fault, etc.) _____

8 Name of park unit, community center, Ex: Smith Park, Johnson Community Center or nearest intersection _____

9 Was an address for incident location? Yes _____ No _____ Unknown _____

10 Name of park unit, community center, Ex: Smith Park, Johnson Community Center or nearest intersection _____

11 Name _____

12 Address _____

13 City _____ State _____ Zip code _____

14 Location (Specify the exact type of location/facility where injury occurred, Ex: maintenance garage, sports field, aquatic center, golf course, etc.) _____

15 Injury location (Specify injury location, Ex: hip joint, left ear, shoulder, etc.) _____

16 Name of person injured (Ex: patient, citizen, visitor, employee, guest, honoree, etc.) _____

17 Did a person sustain an injury? Yes _____ No _____ Unknown _____

18 Give the following information:

19 Last name _____ First name _____

20 Address _____

21 City _____ State _____ Zip code _____

22 Name _____

23 Address _____

24 City _____ State _____ Zip code _____

25 Name _____

26 Address _____

27 City _____ State _____ Zip code _____

28 Did the injury affect body part and time of injury? Ex: common, knee, shoulder, neck, hand, etc. _____

29 Did injured person make any statements? Yes _____ No _____ Unknown _____

30 Did injured person say? _____

Vehicle Accident Report | FVS | +

valentinojosh.github.io/foxvalleyspecialrec.github.io/forms/VehicleAccidentReport.html

PDRMA Vehicle Accident Report Form 02 (pg. 4)

1 Who first aid administered? Yes _____ No _____ Unknown _____

2 Name and position of person who administered first aid _____

3 What first aid was given? _____

4 Did first aid involve AED and/or CPR? Yes _____ No _____ Unknown _____

Vehicle Accident Report | FVSRC

valentinojosh.github.io/foxvalleyspecialrec.github.io/forms/VehicleAccidentReport.html

PDRMA **Vehicle Accident Report** Form 02
(Accident involving agency vehicle. May involve bodily injury or property damage.)

Attorney/Court Privileged Document

Who had the accident?
 Me My spouse Unknown

Name and position of person who witnessed the accident
 Me My spouse Unknown

Time and place of accident?
 Unknown

Time and place of accident CPTM
 Me No Unknown

Was a medical service offered?
 Yes No Unknown

Officer or other law enforcement personnel present?
 Officer Other Unknown

Officer not called?
 Officer Other Unknown

Officer not called or called?
 Officer Other Unknown

Officer not called or called, called by agency attorney?
 Officer Other Unknown

If yes, provide the following information:
 Name of police department
 Name of officer

Do you suspect this person to be drunk?
 Yes No Unknown

Was property damaged as a result of this accident/incident?
 Yes No Unknown

If yes, how was the person involved in the accident/incident?
 Driver of property adjacent to park desired Person

Driver of property adjacent to park desired
 Person Other

and name (or business name) Post office (or delivery/Shipping name)
 Address

CITY STATE ZIP CODE Phone number
 CITY STATE ZIP CODE Phone number

Describe the property damage
 None

Please give a witness to the accident/incident, please provide the following information:
 and name Post office
 ADDRESS

CITY STATE ZIP CODE Phone number
 CITY STATE ZIP CODE Phone number

Did witness make any statements?
 Yes No Unknown

If yes, what did witness say?
 None

Can witness return after the accident/incident occurred?
 Yes No Unknown

The figures above, figure 7, 8, 9, and 10 respectively, display the vehicle accident report. Some fields do not match, more time is required for the style. There is no submit button because this form is not functional.

Form 3

valentinojosh.github.io/foxvalleyspecialrec.github.io/forms/PropertyLossReport.html

PDRMA
PDRMA
For the use of agency property
Attorney/Client Privileged Document

67%

Agency name Today's date
Date of incident (mm/dd/yyyy) Time of incident (hh:mm, a.m./p.m.)
mm / dd / yyyy
Name of person completing the report Title of person completing report
Business phone Business email
200-000-0000
How did the incident occur and what property was damaged? (Provide a brief factual summary)

None of the location (park, pool, community center, Ex. Smith Pool, Johnson Community Center) or nearest intersection where the incident occurred.
Is there an address for incident location? If yes, please provide the following:
Street address
City State Zip code
Location (Specify the exact type of location/facility damaged, listing multiple locations/facilities if necessary. Ex. maintenance garage, sports field)
Primary location (Identify the exact area of damage. Ex. tool storage, building cage)
Estimate of loss
Contact person at facility
Contact person's email
Contact person's phone number
Was damage caused by third party (non-agency) individual? Yes No Unknown
If yes, who is responsible for damage been identified? If yes, provide the following contact information for the person or persons identified:
Name Street address
City State Zip code
Has a police agency conducted an investigation? Yes No Unknown
What police agency investigated the incident? What is the police report number?
Were criminal charges brought against the responsible party? If yes, what were the charges?

The figure above, figure 11, displays the property loss report. The submit button is below the form, when clicked it will process the form and redirect the user to the submission page.

PDRMA Employee Injury Report Form 04
 Park District Risk Management Agency

Complete an Employee Injury Report for each employee injured.

Agency name: Today's date: mm / dd / yyyy a.m. / p.m.
 10a.m. / 12p.m. (noon),
 12p.m. / 2p.m., etc.

Name of person completing the report: Name of person completing report:

Business phone: Business email:
 200-000-0000

How did the incident occur? (Provide a one-line factual description.)

Name of the location (park, pool, community center; Ex. Smith Pool, Johnson Community Center) or nearest intersection where the incident occurred.

Is there an address for incident location? If yes, please provide the following:
 Street address:
 City: State: Zip code: 00000

Location (Specify the exact type of location/facility damaged, listing multiple locations/facilities if necessary. Ex. maintenance garage, sports field):

Primary location (Specify exact location. Ex. lap pool, cart storage, classroom, pavilion):

Employee's FEN:

If the employee miss more than three (3) scheduled workdays? Yes: No: Unknown:
 What was the employee doing when the accident occurred?
 How did the incident occur? (Provide a detailed factual description.)

Employee last name: First name:
 Address:
 City: State: Zip code: 00000
 Home phone #: Work phone #: Cell phone #: 000-000-0000
 Email:
 Best number to contact employee:
 Social security number: Date of birth (mm/dd/yyyy): Gender: Male: Female:
 mm / dd / yyyy
 Marital status: Number of dependents: Does employee speak English?: Yes: No:
 (divorced/married/single/unknown)
 Average weekly wage: Job title/occupation:
 000-000-0000

PDRMA Employee Injury Report Form 04
 Park District Risk Management Agency

What is the employee's employment status?
 Permanent full-time: Permanent part-time: Seasonal: Intern: Other:
 Data hired (mm/dd/yyyy): What is the employee's tenure? Length of employment:
 mm / dd / yyyy Less than 1 year: 1-3 yrs: 4-9 yrs: 10-19 yrs: More than 20 yrs:
 (mm / dd / yyyy) began work on day of incident (Format x.x in y.m.)
 (mm / dd / yyyy) last day employee worked prior to date of incident (mm/dd/yyyy):
 (mm / dd / yyyy)

If the employee died as a result of the accident, give the date of death. (mm/dd/yyyy):
 (mm / dd / yyyy)

Did the accident occur on an agency property? Yes: No: Unknown:
 Injury: Illness:
 Describe the injury or illness (affected body part and type of injury. Ex. concussion, bruise, laceration, sprain, break, etc.):

What object or substance, if any, directly harmed the employee?

Did the injured employee seek medical treatment? Yes: No: Unknown:
 Was the treatment given away from the worksite? Yes: No: Unknown:
 Was the employee treated in an emergency room? Yes: No: Unknown:
 Was the employee hospitalized overnight? Yes: No: Unknown:
 as an inpatient?

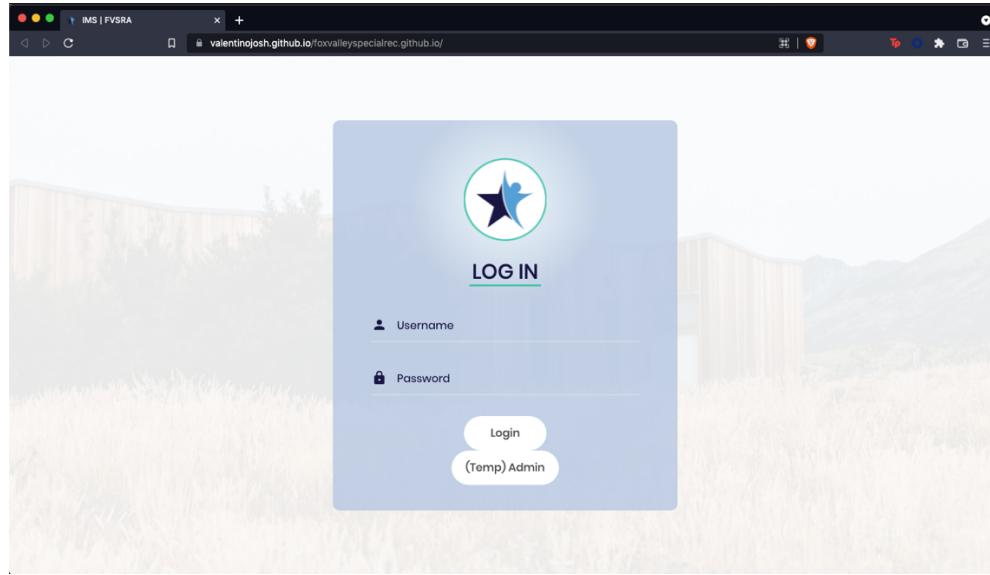
Name of treating physician, health care provider, or emergency room:
 Address:
 City: State: Zip code: 00000
 Phone number: 000-000-0000

The figures above, figure 12 and 13 respectively, display the employee injury report. There is no submit button because this form is not functional.

The figure above, figure 14, displays the notification of injury to employer report. There is a submit button below the final page, however, it is not functional.

The figure above, figure 14, displays the notification of injury to employer report. There is a submit button below the final page, however, it is not functional.

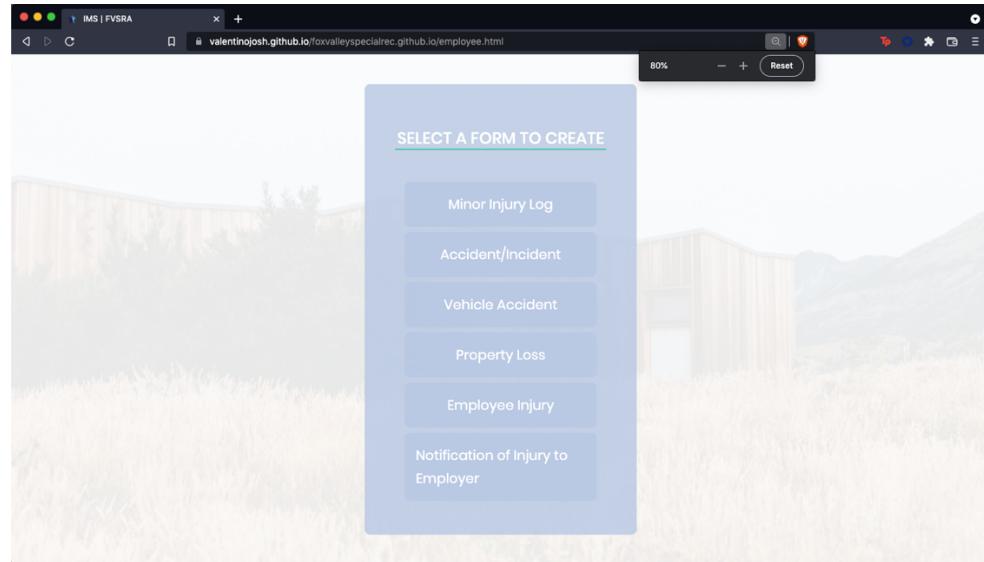
b. User Completed Features (Josh V.) –



User Log In: Status: Ready-for-use

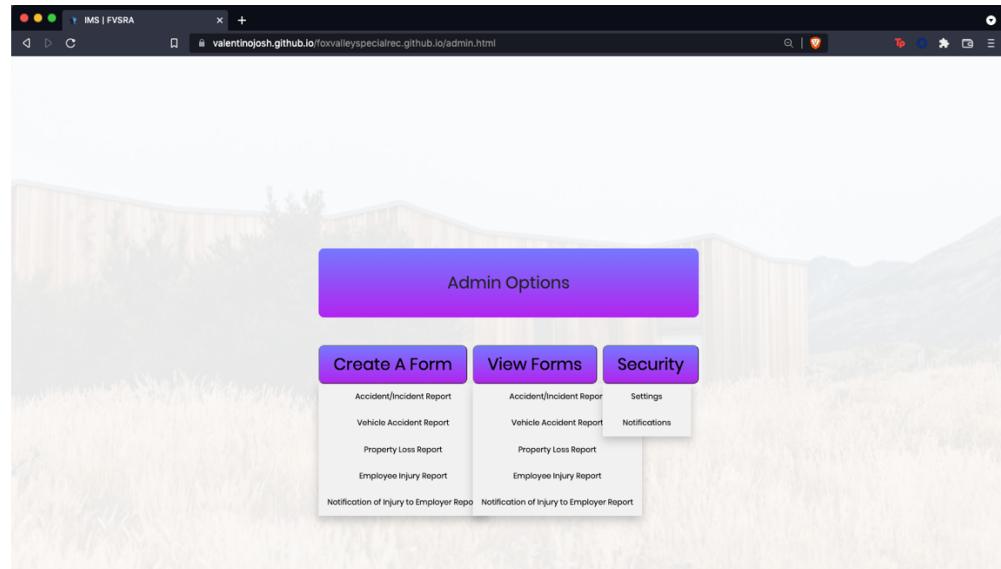
The figure above, figure 15, displays the current front-end for the user log in page. While it is ready to be utilized, this does not include the username and password verification. The feature is able to

provide the user to login as an employee via the login button or login as an admin by clicking the admin button.



Employee Form Selection: Status: Ready-for-use

The figure above, figure 16, displays the current front-end for the employee form selection page. Each of the forms are able to be selected and viewed for creation.



Admin Selection Page: Status: In development

The figure above, figure 17, displays the current front-end for the admin selection page. The create form, view forms, and security button are able to be clicked resulting in a drop down of options, however, the drop downs are not linked to anywhere. The create form feature is able to be accessed through the employee form creation selection page for now. The view forms feature has not been created on the back end so it is not able to be set up. The security feature has also not been created on the backend.

PDRMA **Property Loss Report** **F 67%**
 (For damage to agency property)
 Attorney/Client Privileged Document
 03

1. Agency name _____
 2. Date of incident (mm/dd/yyyy) Time of incident (hh/mm)
 3. Name of person completing the report _____
 4. Business phone _____
 5. Business fax _____
 6. How did the incident occur and what property was damaged? (Provide a brief factual summary.)
 7. Name of the location (park, pool, community center; Ex. Smith Pool, Johnson Community Center) or nearest intersection where the incident occurred.
 8. Is there an address for incident location? If yes, please provide the following:
 Street address _____
 City _____ State _____ Zip code _____
 9. Location (Specify the exact type of location/facility damaged, listing multiple locations/facilities if necessary. Ex. maintenance garage, sports field)
 10. Primary location (Identify the exact area of damage. Ex. Tool storage, bottling cage)
 11. Estimate of loss _____
 12. Contact person's facility _____
 13. Contact person's email _____
 14. Contact person's phone number _____
 15. Was damage caused by third-party (non-governmental entity)? _____ Yes _____ No _____ Unknown _____
 16. Has the party responsible for damage been identified? If yes, provide the following contact information for the person or persons identified:
 Name _____ Street address _____
 City _____ State _____ Zip code _____
 17. Has a police agency conducted an investigation? _____ Yes _____ No _____ Unknown _____
 18. What police agency investigated the incident? _____ What is the police report number? _____
 19. Were criminal charges brought against the responsible party? If yes, what were the charges? _____
 [Submit]

Form Submission: Status: Ready-for-use (2/5), in development (3/5)

The figure above, figure 18, displays the current front-end for the form submission page. Note, the page is zoomed out to fit the entire form in the figure. The property loss report and the minor injury log are able to be submitted properly and updated to the database, while the other three are not. The remaining three are styled and able to be viewed for creation, but submission does not function.

localhost:63342 says
Report has been reported.

Form 03

1 Agency name Fox Valley Special Recreation Association Today's date 11/30/2021

2 Date of incident (mm/dd/yyyy) 11/28/2021 Time of incident (hh/mm, a.m./p.m.) 08:34 AM

3 Name of person completing the report Rudy Johnson Title of person completing report Supervisor

4 Business phone 730-567-8912 Business email rudy@fvsla.org

5 How did the incident occur and what property was damaged? (Provide a brief factual summary.)
Guy was playing basketball and when dunking, he broke the rim

6 Name of the location (park, pool, community center; Ex. Smith Pool, Johnson Community Center) or nearest intersection where the incident occurred. Community Center

7 Is there an address for incident location? If yes, please provide the following:
Street address 123 Sesame St
City Aurora State IL Zip code 60504

8 Location (Specify the exact type of location/facility damaged, listing multiple locations/facilities if necessary. Ex. maintenance garage, sports field)
Basketball Court

9 Primary location (Identify the exact area of damage. Ex. tool storage, batting cage)
Basketball Rim

10 Estimate of loss \$69.00

11 Contact person at facility Arthur Jones

12 Contact person's email AJones@thepark.com

13 Contact person's phone number 630-111-2222

14 Was damage caused by third-party (non-agency) individual? Yes No Unknown

15 Has the party responsible for damage been identified? If yes, provide the following contact information for the person or persons identified:
Name George Miles Street address 456 Highland Ave
City Aurora State IL Zip code 60506

16 Has a police agency conducted an investigation? Yes No Unknown

17 What police agency investigated the incident? Aurora Police Dept What is the police report number? D-6789

18 Were criminal charges brought against the responsible party? If yes, what were the charges?
Need to pay for a new rim

Submit

Showing rows 0 - 2 (3 total. Query took 0.0003 seconds.)

```
SELECT * FROM `Property_Loss_Report`
```

property_loss_id	Name_of_the_Agency	Todays_Date	Date_of_Incident	Time_of_Incident	Name_of_the_person_Completing_the_Report	Title_of_Person_Completing_the_Report	Business_Phone
1	Fox Valley Special Recreation Association	2021-11-30	2021-11-29	15:20:29	Brad	Worker	2147483647
2	Fox Valley Special Recreation Association	2021-11-30	0000-00-00	20:25:00	Jimmy	Worker	630-555-6666
3	Fox Valley Special Recreation Association	2021-11-30	0000-00-00	08:34:00	Rudy Johnson	Supervisor	730-567-8912

Query results operations: Print, Copy to clipboard, Export, Display chart, Create view

The figure to the left, figure 19, shows the filled form 3 after the submit button is clicked.

The figure above, figure 20, shows part of the query from the database in selecting all data from the property loss report. Property loss id 3 is the form that was pictured in figure 19. The name of the person completing the form as well as title and phone number also match from figure 19. The complete query was not shown due to the horizontal length of the resulting capture.

Fox Valley Project / Property Loss Report / Get An Individual Property Loss Report

Save ... Edit Send Copy

GET {{url}}/fvsra/propertyLossReport/3

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Body	Cookies	Headers (8)	Test Results	200 OK 48 ms 1.52 KB Save Response
Pretty	Raw	Preview	Visualize	JSON Copy Search

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
{
  "property_loss_id": 3,
  "Name_of_the_Agency": "Fox Valley Special Recreation Association",
  "Todays_Date": "2021-11-30T06:00:00.000Z",
  "Date_of_Incident": "0000-00-00",
  "Time_of_Incident": "08:34:00",
  "Name_of_the_person_Completing_the_report": "Rudy Johnson",
  "Title_Of_Person_Completing_the_report": "Supervisor",
  "Business_Phone": "730-567-8912",
  "How_did_the_incident_occur_and_what_property_was_damaged": "Guy was playing basketball and when dunking, he broke the rim",
  "Name_of_location": "Community Center",
  "Business_Email": "rudyj@fvsra.org",
  "Street_Address_Incident_Location": "123 Sesame St",
  "City_Incident_Location": "Aurora",
  "Zip_Code": 60504,
  "Location_Specify": "Basketball Court",
  "Primary_Location": "Basketball Rim",
  "Estimate_of_Loss": 69,
  "Contact_Person_Facility": "Arthur Jones",
  "Contact_Person_Email": "AJones@thepark.com",
  "Contact_Person_Phone_Number": "630-111-2222",
  "Damage_Third_Party": "Yes",
  "Person_Responsible_Name": "George Miles",
  "Person_Responsible_Street_Address": "456 Highland Ave",
  "Person_Responsible_State": "IL",
  "Zip_Code_Person_Responsible": 50506,
  "Police_Investigation_Yes_No_Unknown": "Yes",
  "What_Police_Agency": "Aurora Police Dept",
  "Police_Report_Number": "D-6789",
  "Were_Criminal_Charges": "Need to pay for a new rim",
  "Person_Responsible_City": "Aurora",
  "State_Incident_Location": "IL"
}

```

The figure above, figure 21, shows the full query of information through a get request in postman.

Date	Time	Name of Injury	Injury Location	Treatment	How injury occurred	Facility/Location where injury occurred	Full Name of staff who noticed guardian
12/01/2021	01:00 PM	Incision	Right Leg	Bandage	Slipped on Wet Floor	Food Court	John Doe

```

injury_date: '2021-12-01',
injury_time: '13:00',
name_of_injured: 'Incision',
injury_location: 'Right Leg',
treatment: 'Bandage',
how_injury_occurred: 'Slipped on Wet Floor',
facility_where_injury_occurred: 'Food Court',
full_name_of_staff: 'John Doe'
}
11
    
```

The figure above, figure 22, displays the minor injury log form filled out after the submit button was clicked.

	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	minor_injury_id	injury_date	injury_time	name_of_injured	injury_location	treatment	how_injury_occurred	facility_where_injury_occurred	full_name_of_staff
<input type="checkbox"/>	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	2021-09-01	11:20 AM	David Smith	Finger	Therapy	Tripped over	Perks park	Ryes Green
<input type="checkbox"/>	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	2021-09-03	12:05 PM	Holly Cruz	knee	Ice-Pack	Lost Control	Junction City	Brown Gill
<input type="checkbox"/>	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	2021-09-04	10:30 AM	John Doe	Aurora University	Band-Aid	Fell down some stairs	Library	Steven Garcia
<input type="checkbox"/>	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	2021-09-04	02:45 AM	Ben Kyle	Aurora University	Ointment	Tripped over a tree branch	On Campus	Kurt Johnson
<input type="checkbox"/>	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	2021-11-14	2:30pm	Hailey Buzz	Forehead	First-Aid	Fell from Chair	Pickle park	Mr Brown
<input type="checkbox"/>	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	2021-09-04	12:45 PM	Joe Swanson	Aurora University	Ice pack	Got hit by a ball on the head	Dorms	Maria Sanchez
<input type="checkbox"/>	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	11	2021-12-01	13:00	Incision	Right Leg	Bandage	Slipped on Wet Floor	Food Court	John Doe

The figure above, figure 23, shows that data that was input into the form, updated to the database.

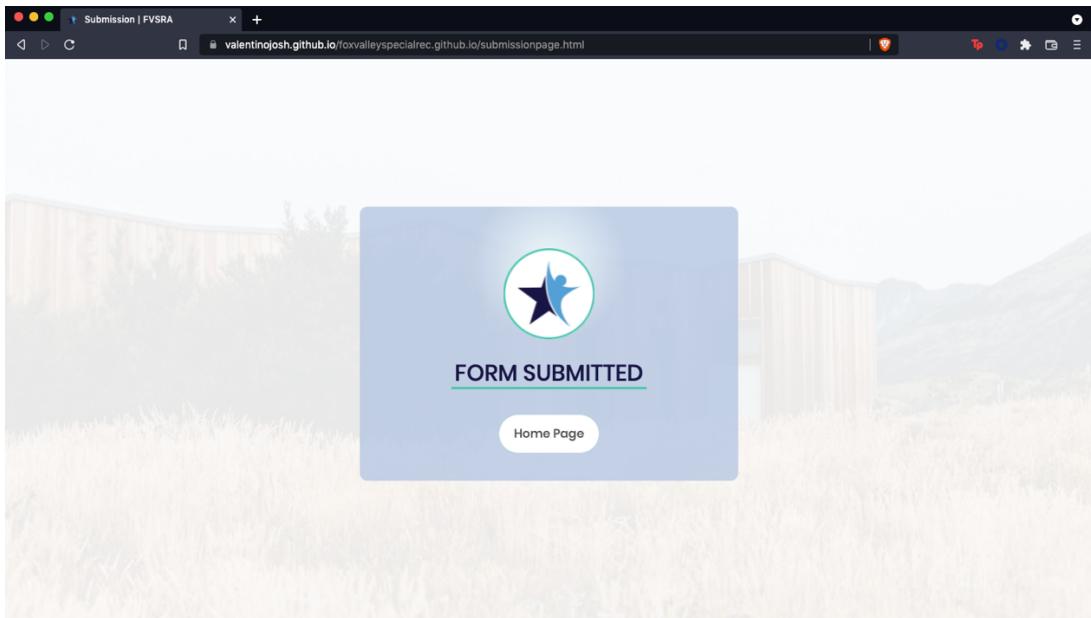
The figure shows the Postman application interface. On the left, the 'My Workspace' sidebar lists various collections, APIs, environments, and other resources. In the center, a specific collection named 'Fox Valley Project' is selected, and its sub-collection 'Minor Injury Log' is expanded. A 'GET' request for 'Get An Individual Injury Event' is highlighted. The request URL is `({{url}})/fvsra/minorInjuryLog/11`. The 'Params' tab shows a single parameter 'Key'. The 'Body' tab displays a JSON response with the following data:

```

1  {
2   "minor_injury_id": 11,
3   "injury_date": "2021-12-01T06:00:00.000Z",
4   "injury_time": "13:00",
5   "name_of_injured": "Incision",
6   "injury_location": "Right Leg",
7   "treatment": "Bandage",
8   "how_injury_occurred": "Slipped on Wet Floor",
9   "facility_where_injury_occurred": "Food Court",
10  "full_name_of_staff": "John Doe"
11
12
13

```

The figure above, figure 24, shows the full query of information through a get request in postman.



Submission Page: Status: Ready-for-use

The figure above, figure 25, displays the form submission page which the user is redirected to after submitting a form. It gives them the option to click the home page button routing them back to the index on click.

c. **Next Steps Assessment (Josh V.) –**

The current quality of the product is near deployment. All the forms have been created and display properly. Only 2 of the 5 forms are able to update properly to the database: The Property Loss Report and the Minor Injury Log. The user login screen functions, however, there is no verification for usernames and passwords. Any user can select to use the employee side or the admin side. The admin side is currently nonfunctional. The employee side is where all of the forms can be created, in which the two previously mentioned are able to be filled out and submitted with the database saving all of the information. The remaining three forms are able to be viewed for creation, but not able to submit. After submitting the forms, currently it links to a form submission page. This page says the form was submitted whether there were errors or not

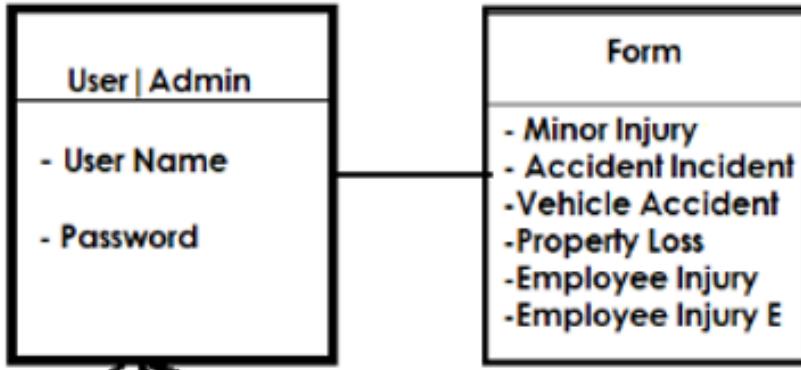
As far as what needs to be done in order to make the product ready for deployment, there is about an estimated forty hours of needed work. The HTML and CSS for the forms needs to be tweaked to display more aesthetically as well as organize the code itself. The database needs to have a final run through, and the variables need to be renamed to be shorted and double check all of the variable types. The database also needs an additional table to store usernames and passwords. This means the backend would also need to implement a way for the admin to create usernames and passwords for their employees. The front-end user login screen needs to implement this part of the database so the back-end can securely verify the username and login information to whichever user they are. The back-end also needs to create a way to output the data from the database into an excel file for admins to read when they need to view forms. As well as create the security features for the admins, which were deemed the creation and changing of usernames and

passwords, as well as notification settings. The submission page also needs to be updated so upon submitting a form it checks for errors before continuing.

d. A complete database diagram -

Vehicle Accident Report	Property Loss Report	Notification of Injury to Employer Report	Minor Injury Log Form	Employee Injury Report	Accident Incident Report
<pre>CREATE TABLE `Vehicle_Accident_Report` (`vehicle_accident_id` int(11) NOT NULL, `Name_of_the_Agency` varchar(50) DEFAULT NULL, `Todays_Date` date DEFAULT NULL, `Date_of_Incident` date NOT NULL, `Time_of_Incident` datetime NOT NULL, `</pre>	<pre>CREATE TABLE `Property_Loss_Report` (`property_loss_id` int(11) NOT NULL, `Name_of_the_Agency` varchar(50) NOT NULL, `Todays_Date` date NOT NULL, `Date_of_Incident` date NOT NULL, `Time_of_Incident` time NOT NULL, `</pre>	<pre>CREATE TABLE `Notification_of_Injury_to_Employer_Report` (`employee_injury_id` int(11) NOT NULL, `Employee_Name` varchar(30) NOT NULL, `Date_of_Incident` date NOT NULL, `Time_of_Incident` datetime(6) NOT NULL, `Specific_location_of_incident` varchar(50) NOT NULL, `Are_you_reporting_the_injury_for_the_firs...</pre>	<pre>CREATE TABLE `Minor_Injury_Log` (`minor_injury_id` int(11) NOT NULL, `injury_date` date NOT NULL, `injury_time` text NOT NULL, `name_of_injured` varchar(100) NOT NULL, `injury_location` varchar(100) NOT NULL, `treatment` varchar(200) NOT NULL, `how_injury_occurred` varchar(300) NOT NULL, `</pre>	<pre>CREATE TABLE `Employee_Injury_Report` (`employee_injury_id` int(11) NOT NULL, `Agency_name` varchar(30) DEFAULT NULL, `Today's_Date` date DEFAULT NULL, `Time_of_Incident` datetime NOT NULL, `Name_of_person_completing_report` varchar(30) NOT NULL, `Title_of_person_completing_report` varchar(30) NOT NULL, `Business_phone` int(11) NOT NULL, `Title_of_Person_Completing_the_Report`...</pre>	<pre>CREATE TABLE `Accident_Incident_Report` (`accident_incident_id` int(11) NOT NULL, `Agency_Name` varchar(50) DEFAULT NULL, `Todays_Date` date NOT NULL, `Date_of_Incident` date NOT NULL, `Time_of_incident` datetime NOT NULL, `Name_of_the_person_Completing_the_Report` int(30) NOT NULL, `Title_of_Person_Completing_the_Report`...</pre>

e. Design Diagram -



Log-in The system starts with the user and admin. They utilize their username and password in order to access the list of forms. They then select which form to create.

Forms-There are 6 different forms and they are at their individual stand there are no relationship to each other, Each table have their own unique ID.

Vehicle Accident Report- This form records all accidents related to vehicle.

Property Loss Report- This form records any loss to the property anything damaged to agency's property.

Notification of injury to Employer Report- This form records the injury of the employee and should be submitted to the supervisor or HR.

Minor Injury Log Form- This forms Records all minor injured people information.

Employee Injury Report- This form records injury of the employee.

Accident Incident Report- This forms records any type of Accidents Incident happened either to the person or to the property.