Environmental Collection Application

Final D-Lab Report

Christie Frush and Skylar Bala

June 12, 2019

Design Brief:

Our goal for the quarter was to create an Android application to help log environmental incidents in the Morogoro region of Tanzania. Our client, Venance S. Segere, a Humphrey Fellow who works for as a Technical Environmental Adviser for the Morogoro Government, wanted an Android application to allow locals to send environmental reports directly to their local government. To be successful, this application needed to include an easy to use interface, a database to hold all of the reports, and security to ensure no individual could be traced.

Design Process and Methodology:

Our team was fortunate enough to access the prior art created by the D-Lab students of Winter Quarter 2018. The students created a prospective front-end through an alternative design framework based on XML. Due to inadequate resources detailing the framework and its lack of presence in the tech industry we decided to not use their prototype and switched to more recognized platforms. This switch was based on both our previous knowledge and to allow for easier future development and upkeep. Even though we did not use the frameworks, the design layouts provided by the previous D-Lab students served as an important template for our application.

The structure of the backend was predominantly based on the essential categories of information needed by the government to resolve the incident. We understood that certain criteria and fields were required and others may not be available in all cases but helpful for government officials trying to fix the problem.

Our main concerns came down to the following five traits: mobile performance, readable/maintainable code, usable on the users' front, secure/private data, and evolvable coding practices. One of the largest issues for this project was identifying how to change our metrics from a qualitative measure to a quantitative one. Initially, we planned to measure each of these categories though variations of qualitative scales, but later reassessed and switched to a more quantitative scaling system shown below.

Objective Function	Qualitative/ Quantitative	Testing Procedure	Target Value	Objective Unit/Source
Mobile Performance	Quantitative	Use Performance Monitor to check FPS	60 FPS	Frames Per Second Fujitsu UI Team
Readability/ Maintainability	Qualitative	Writing tests on backend to ensure properly written functions and performing strict code review to evaluate how closely new code follows best practices/ design patterns	10	Scale of 1-10 (10 = code follows Source: Good Software Coding Practices
Understandability/Ease of Usability by Public	Qualitative	Have other people evaluate the ability to easily understand the app	10	Scale of 1-10 (10 = easy to understand) Source: Good Software Coding Practices
Evolvability of Software	Qualitative	Documentation and Strict Code Review/ Have each developer perform strict code review to ensure separation of concerns	10	Scale of 1-10 (10 = interoperable software) Source: Good Software

		and prevent "god-objects"		Coding Practices
Security/ Privacy	Qualitative	Attempt to access sensitive data from a hacker point of view	10	A Scale of 1-10 (10 = app not secure/ easy to hack) Source: an unaffiliated hacker

For mobile performance, our team agreed to abide with the industry standard of 60 frames per second (FPS). According to the Fujitsu UI research team, this is the amount of time where the human brain will register the various still images as a "moving" process without depending on cinematic fillers (2013). This requirement can easily be checked by Android testing suites such as Dumpsys, which will be tested later in the project timeline.

One of the most important categories is usability, which measures how well a potential user could interact with the product without any further instruction. Initially our team planned to use a qualitative measure, but later switched to a quantitative process. During testing, our team planned to measure the number of users who could successfully write an incident report without asking for help using the application. Our final goal was to have a 90% success rate with users creating and viewing reports. Unfortunately, our testing plan fell through due to unforeseen technical difficulties.

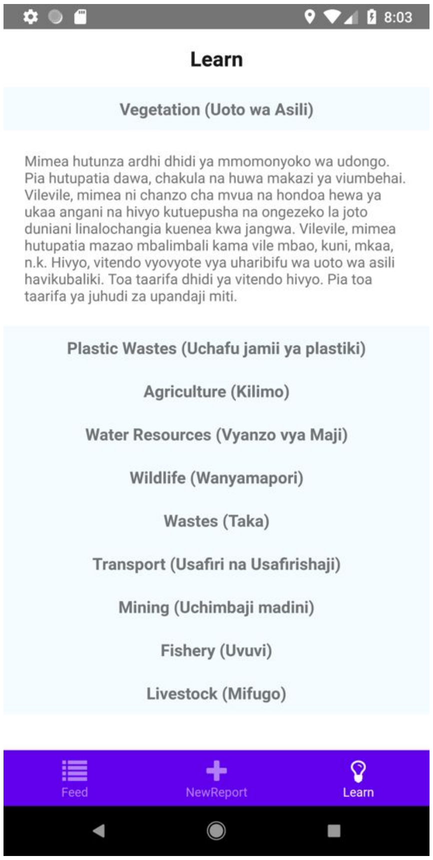
Arguably, security is the most important feature of our application. With the recent ban of plastic in Tanzania, individuals using this application, possibly reporting illegal activities, may receive backlash if identified. Unfortunately, security is hard to measure, as leaks are only discovered after security breaches. Our team concluded that the most feasible method to test our security would be to have a hacker attempt to hack the application and measure its ease on a qualitative scale. Our hypothetical goal value would have been a 10, suggesting that the app was extremely difficult or impossible for hackers to view the data. Currently, this test has not been conducted, but security currently remains far below industry standards.

Relating to security, privacy is also important to users, as they should feel anonymous and protected while using the application. This criteria was another qualitative criterial which we planned to discuss with beta users during testing. Our main design choice included to give our users an illusion of privacy by not using accounts. The users' input will not be available until further testing.

For both readability, writability, and evolvability, our team could not mimic the regular code reviews that are present in industry. Instead our team decided that qualitative measures would suffice. Ideally to test both of these categories, other unaffiliated Android developers would review the code and rate its clarity, Due to time and resource constraints our team will not be able to implement this process until a further date.

Results and Discussion:

During our twelve weeks in D-Lab, we successfully created an application prototype that can be used to bridge the gap between local citizens and government officials when logging environmental data. This application can be used at a potential prototype for expansion of this system as it is anticipated to be tested within a small area of the Morogoro region in Tanzania. Below are the screenshots of the prototype built in D-Lab during spring quarter.



Learn

Vegetation (Uoto wa Asili)

Mimea hutunza ardhi dhidi ya mmomonyoko wa udongo. Pia hutupatia dawa, chakula na huwa makazi ya viumbehai. Vilevile, mimea ni chanzo cha mvua na hondoa hewa ya ukaa angani na hivyo kutuepusha na ongezeko la joto duniani linalochangia kuenea kwa jangwa. Vilevile, mimea hutupatia mazao mbalimbali kama vile mbao, kuni, mkaa, n.k. Hivyo, vitendo vyovyote vya uharibifu wa uoto wa asili havikubaliki. Toa taarifa dhidi ya vitendo hivyo. Pia toa taarifa ya juhudi za upandaji miti.

Plastic Wastes (Uchafu jamii ya plastiki)

Agriculture (Kilimo)

Water Resources (Vyanzo vya Maji)

Wildlife (Wanyamapori)

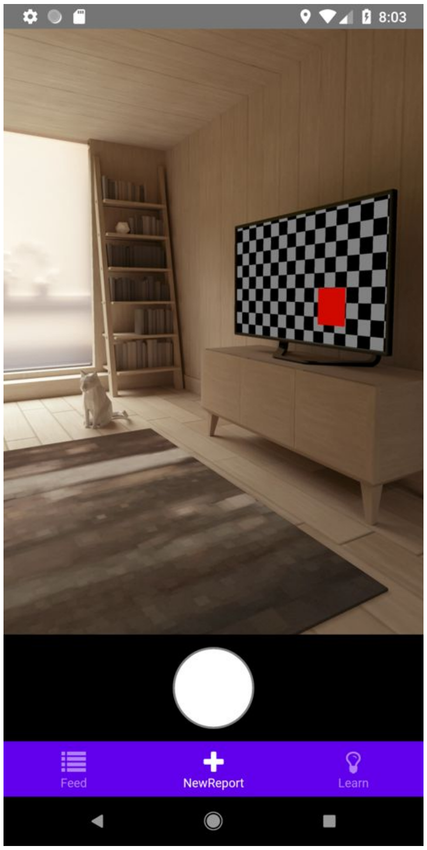
Wastes (Taka)

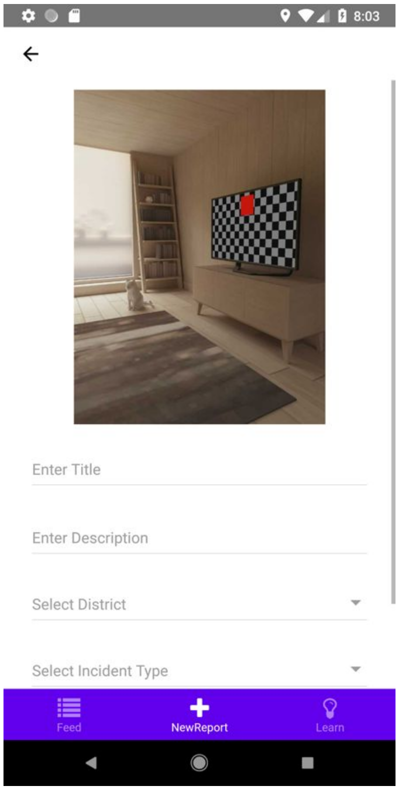
Transport (Usafiri na Usafirishaji)

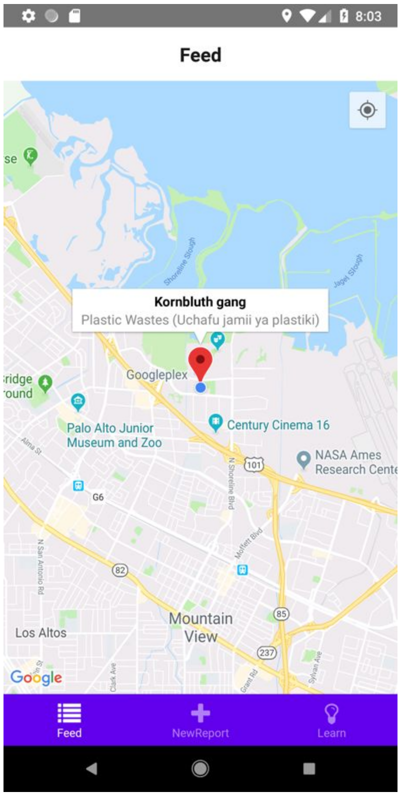
Mining (Uchimbaji madini)

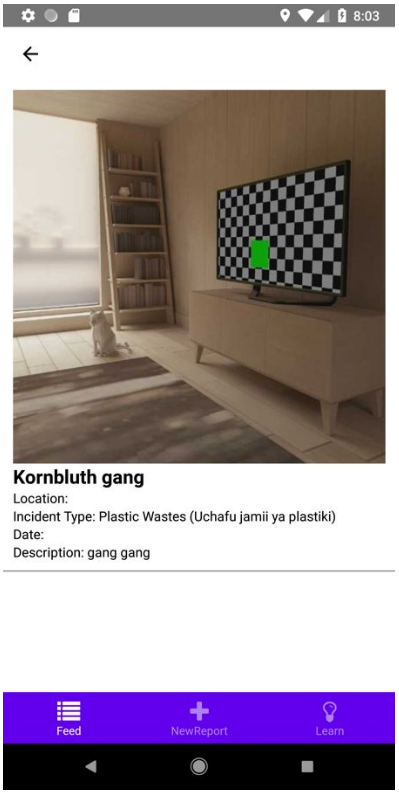
Fishery (Uvuvi)

Livestock (Mifugo)









Kornbluth gang
Location:
Incident Type: Plastic Wastes (Uchafu jamii ya plastiki)
Date:
Description: gang gang

Conclusion:

Looking back on our work, there are some adjustments that could have better been developed or foreseen as a potential problems. One of the largest and most prevalent problem is the potential leaks in the security of the system while other less critical improvements would include a better mapping algorithm and a portal to see the user data.

Currently our application documents is where the incident was located via a map. The user creates a pin indicating to where the user believes the incident occurred. This method has many flaws due to user error and lack of precision. This may cause inconvenience on the government side, as one of the benign errors may include inaccurate reported locations.

As a prototype, our application currently lacks basic security. A higher standard of security should be implemented before undergoing any deployment or rigorous user testing. The new security will modify the original code thus exposing new potential bugs.

While building our application we realized that the customer had no way of accessing the user's data. Users could upload reports, but there was no portal where the client could see the incoming report and manage them accordingly. To solve this, we agreed to build a basic website to display the GPS coordinates of the incidents on Google Maps using a Google API. Due to time constraints, this portal will not be finished until after the quarter has ended.

Recommendations:

While our app servers as a wonderful prototype, to implement this on a large-scale population some major changes would be required. These changes include a more robust backend, a data metrics system, improved security and official branding. These improvements would drastically change the utility of the information to better serve the relevant communities.

While the app successfully collects and holds data from the users within a database, the data is virtually useless in the database form. To help solve this problem, we built a basic website to display the data on a rudimentary google map. Our solution to this problem should best be considered as a Band-Aid, as it is not versatile enough to be useful for multiple users or anything apart from obtaining information on a map. A better solution would either be to purchase datamining or data metrics software or to create another project which would use data analytics to better understand the user's data.

One of the most prevalent issues with our application is the lack of privacy and security features to keep users safe. Being that our application creates and uses sensitive data, we need to ensure that users cannot be traced from their reports. If users can be trace, the users may be in danger or blackmailed. If a security breach occurs on the server side, bureaucrats' emails and work materials may become exposed or corrupted. Minimal regular safety features of any industry level application would include encrypting or encoding the data during transportation and while it is contained in the database. Safety could be improved with the use of JSON web tokens (JWT), which encode the data during transfer between the phone and database. Encoding the entire database would be another imperative security improvement. Encoding the entire database ensures that if there were a security breach on the server, that the data would be incoherent and useless to the hackers. Other minor improvements may include blurring any individuals in photos to ensure privacy.

Another major security issue relates to the website created for the client to visualize the data. The current portal for viewing database data poses a dangerous leak and shows the need for a finalize product. Currently, the website includes no security protocols, but plans to include Google authorization, which may be rendered useless if the emails used by the Tanzania government are not associated with Google. This website needs to include strict access protocols possibly managed from a in house security or information technology group to authorize and maintain users access and credentials.

Depending on the number of users, the current type of database used is considered useful for small projects and hobbyists. If the amount of traffic became substantial it would be wise to switch to a database more suited for high traffic, such as Postgres. We initially did not use this type of database to create our prototype as they are widely considered overkill for small projects. Future modifications may be necessary if the app acquires a significant flow of traffic.

Just like thousands of other applications, our app is unknown to the world. When thinking of environmental applications, our app remains nameless, as it floats through the cloud without a logo or name. Thankfully our client has already planned for this scenario, knowing eventually our application will need a distinct image to entice users. He has chosen the image below to serve as the application's logo and has chosen the name Tazama for the application which means to see in Swahili. These changes will be implemented before the application is released on Google Play.

Bibliography:

Hellström, Johan. "The Innovative Use of Mobile Applications in East Africa." Sida, Sweden: Edita, 2010. All. Print. Hellström's book helps show case the statistics of Tanzanian phone usage both including usage patterns surrounding SMS and internet connected phones. This source also showcases the various mobile applications used in East African countries. Hellström discusses scaling, sustainability and marketing for these applications, which can help us better

understand our large audience. The only setback of this article is that it is considerably old for mobile technology, so it may not be as accurate for the current model.

MPOGOLE, Hosea. "Mobile Phones and Poverty Alleviation: A Survey Study in Rural Tanzania." Ed. USANGA, Hidaya. Arusha, Tanzania: Karlstad University, 2008. Ed. TEDRE, Matti. Print.

It is known that our target audience may include more impoverished individuals. This article helps clarify the misconceptions with the lower-class Tanzanians and their mobile phone usage. This article is particularly helpful with explaining the relationships between family in Tanzania and how it plays a role in their behavior. This is article is somewhat dated so the actual statistics may be incorrect.

Nyamtiga, Baraka W. "Enhanced Security Model for Mobile Banking Systems in Tanzania." Ed. Sam, Anael. Arusha, Tanzania: INTERNATIONAL JOURNAL OF TECHNOLOGY ENHANCEMENTS AND EMERGING ENGINEERING RESEARCH, 2013. 4 - 19. Vol. 1. Ed. Laizer, Loserian S. Print. While this article is outdated, it contains many important points about mobile security in Tanzania. Our application must handle some sort of security, and this article illustrates many notable points about mobile security in Tanzania. The article also summarizes many of the classical problems with mobile security and their attempted solution.

Wesselink, Anna, and Rob Lemmens. "Not Just a Tool. Taking Context into Account in the Development of a Mobile App for Rural Water Supply in Tanzania." Ed. Hoppe, Robert. Water Alternatives: University of Twente, 2015. Vol. 8. Print. Culturally Tanzania and the US are drastically different, one of the largest ways can be seen in Tanzania's eGovernance policies, discussed in the paper. This hype revolves around using technology to make the government more efficient and more transparent. This paper showcases a particular mobile application made to help regulate rural

water supply. This article helps branch from the US system of development and countries standards. Being that Tanzanian individuals has less available 3G/4G network than the US, it showcases possible solutions to designing an application that is more disconnected.

Z, Mtema. "Mobile Phones as Surveillance Tools: Implementing and Evaluating a Large-Scale Intersectoral Surveillance System for Rabies in Tanzania." Ed. J, Changalucha: Mtema et al., 2016. Ed. S, Cleaveland. Print. This article illustrates an example of how a government application can help create a database and utilize it for greater change. The application used in southern Tanzania helped collect data and rabies to further a plan to prevent future outbreaks. Similar to our application this rabies prevention application illustrate how an application can be developed for the general population to improve quality of life.

Ueno, T, and M Tanimura. “Smartphone User Interface.” Fujitsu Scientific &amp; Technical Journal, Fujitsu, Apr. 2013, www.fujitsu.com/global/documents/about/resources/publications/fstj/archives/vol49-2/paper14.pdf. This paper explains the basic rules of phone user interface and why industry uses the certain metrics, such as why 60 frames per second (fps) remains the industry standard. The paper also delves into the basic history and why other standards are used in other industries such as cinema and filmography.

Appendix

Deliverable 1:

CHRISTIE FRUSH

Hello, I am Christie, I also go by any pun or play on my last name “Frush”, as people think it is fun to say. If I were a designer or decorator my company would be called “A Frush New Look”, but sadly, I chose computer science instead. While the only foreign language I speak is some French, I know a lot of coding languages such as C++/C, Java, Node JS, etc. and have coded multiple projects in various languages. My other experience includes some work in IT for a large retail company, an iOS application to help wildlife/conservation, and a large helping of customer service work in the restaurant industry. The most recent skill that I am working on is learning how to start and finish tasks in which I have little direction and no idea where to begin.

Project	Impact	weight	Design vs. Hands On	weight	Time	weight	Relevant Skills	weight	Resume	weight	Total
Crop Wash Station	2	4	1	3	3	2	1	5	1	4	26
Environmental Data Collection App	5	4	3	3	2	2	5	5	5	4	78
Recycled Materials Barcode	3	4	2	3	3	2	4	5	5	4	64
Rice Post Harvest Packaging	3	4	3	3	3	2	2	5	2	4	45
Smallholder Farm Coffee Roaster	3	4	5	3	4	2	2	5	2	4	53

3 Top Project Choices:

1. Data Collection App
2. Recycled Materials Barcode
3. Smallholder Farm Coffee Roaster

Deliverable 2:

1. Who is the client and what is their business?
   The client is Mr. Venance Soza who works with the Morogoro Regional Secretariat, which is the local government of the Morogoro region.

1. What is the Project Goal Statement? This speaks to the need your design address. It should be concise (1 or 2 sentences).

a. Our goal is to create a mobile app and supporting backend that allows Tanzanian locals to document/log environmental incidents. This application will notify the correct governmental entities, so these incidents can be resolved more effectively.

1. What are the specific project goals? Why?

Our goal is to create a polished environmental data collection mobile app and supporting backend. In creating our app we hope to work with the client to further frame and reframe the project scope and define the design space. Being that our time is considerably limited for building an app, we plan on using the relevant prior art to help build the app as quickly as possible. Through rapid prototyping, we hope to work with the client to refine design constraints and conduct a formidable set of testing suites to evaluate and ensure the effectiveness of the app.

1. Who is the target market/ customer?

a. Our target market is the people and the governmental constituents of Tanzania.

1. Any known benchmarks?

a. Apart from the final deliveries, we aren't aware of any benchmarks at this time.

1. What is the approximate budget?

a. A $5-10/monthly subscription for a hosted cloud server. This may increase as more data is held on the server or more users are added. An additional flat fee of $50 is will also be required to publish on Google Play Store.

1. What is the appropriate timeline?

a. Week 2- Identify resources needed and create a design document

b. Week 3- Mockup backend REST API using Node.js/ Start Front End using React Native/Check with client

c. Week 4- Hook up server/ Finish Front End

d. Week 5- Finish 3 Endpoints/Check with Client

e. Week 6- Finish Rest of Endpoints

f. Week 7- Begin Testing with Postman

g. Week 8- Enable Endpoints in Front End/Check in with Client

h. Week 9- Testing

i. Week 10- Testing

1. What are the final deliverables?

A fully functional app that can be used by the government agencies and the locals to document and manage environmental incidents.

Deliverable 3:

1. Who is the client? Who are you designing the technology for?

Our client is Venance S. Segere who is a Humphrey fellow and Technical Environmental Adviser who wants an Android application for a citizens and locals living in the Morogoro region and possibly expanding to all citizens of Tanzania.

1. Who is the target customer? Who will actually be using the device? Be specific.

The target customer is locals ages 18 - 65 who own a photo and regularly use applications.

1. What are the specifications if any given for the technology?

Apart from the application running on Android, there are no current specifications for the desired product.

1. What are the technical, social, environmental, and financial considerations?

Being that we were given a front-end design, there are few concerns pertaining to the design of our app. The backend may need to be upgraded depending on the number of total users, or increased in size to hold the data from citizens using the application.

1. What are other existing designs?

Currently the other D-Lab group has created a layout for the user interface of the application. Our client has also mentioned that he would like a design similar to Citizen Science Tahoe or Snap Send and Solve, both of which are applications for reporting environmental incidents.

1. What will a successful design do?

A successful design will be easy and intuitive for the user to use. The user should be able to create a post and send it to the government server without fail even when mobile data or wifi is not accessible.

1. What is the timeline?

The proposed timelines is to build the application in five weeks and then spend the rest of the time testing or modifying the application according to clients requests.

1. What is the project budget? What is the cost and quantity of products needed?

The product budget is under $5 a month for the server and possibly extra to incentivize testers to test the application.

1. What is the end deliverable?

The end deliverable will be a functional application that a user can post a photo successfully to the server and the client can see all the posts on the server.

1. What skills and information will you need to design a successful product?

To successfully build this project we will need the mobile limitations from the client as well as an understanding of the citizens using this application. For skills we will need to know React Native for the front-end and Node-js/SQLite to create the backend.

Considerations:

Testing, Product Quality, testing quality, Safety of Languages/Technologies used, appeal to customers, scope, customer versus client, use cases, mobile usage in Tanzania, security risks/leaks, performance, upkeep, upgrades/scalability.

Deliverable 4:

1. Has your project goal statement changed since learning more information?
    a. No our project goal remains the same: our goal is to build and deploy an environmental data collection app to help the Tanzanian government assess

which locations require the most clean-up and raise the environmental consciousness of those in Tanzania.

2. What are the important design considerations? Narrow down to 5 from the previous deliverable and refine them into criteria
3. How you will evaluate and test the design? Will these be qualitative or quantitative?
4. What are the metrics?
5. What are the target Values for each metric and in what are the units?
6. Make a Testing with this information

Objective Function	Qualitative/ Quantitative	Testing Procedure	Target Value	Objective Unit/Source
Mobile Performance	Quantitative	Use Performance Monitor to check FPS	60 FPS	Frames Per Second Fujitsu UI Team
Readability/ Maintainability	Qualitative	Writing tests on backend to ensure properly written functions and performing strict code review to evaluate how closely new code follows best practices/ design patterns	10	Scale of 1-10 (10 = code follows Source: Good Software Coding Practices
Understandability/Ease of Usability by Public	Qualitative	Have other people evaluate the ability to easily understand the app	10	Scale of 1-10 (10 = easy to understand) Source: Good Software Coding Practices
Evolvability of Software	Qualitative	Documentation and Strict Code Review/ Have each developer perform strict code review to ensure separation of concerns and prevent “god-objects”	10	Scale of 1-10 (10 = interoperable software) Source: Good Software Coding Practices

Security/ Privacy	Qualitative	Attempt to access sensitive data from a hacker point of view	10	A Scale of 1-10 (10 = app not secure/ easy to hack) Source: an unaffiliated hacker

Deliverable 5:

Project Update: No new information, our point of contact will be in Davis today to speak about the project and discuss deliverable 3 and any changes to the project.

Possible additions/renovations to application

- Social Element
    - Share to Facebook
    - Share once the clean-up is finished
    - Viral karma -> That trash competition thing
    - Notifications to cause change in your area
    - Notifications on the status updates of previous claims
- Credits/Incentives of some sort
    - Monetary -> coupons
    - Monetary -> rewards program
    - Improvement on the community/ greater communities
        - -> Like forest app, which plants trees
        - -> Donations of rice to places in need
    - Categorizations of virtual Goodness
        - -> a saved visualization of “goodness” done
        - ->In app awards
        - ->Updated photos of reported incidents
- Anonymity
    - No users
    - Encrypted users
    - Users have profile that is only locally saved on phone(never pushed to cloud)
    - No attached user id to report
    - Encrypted/ Untraceable IP address?
- User Interface
    - Scrollable new feed of other anonymous projects
- Backend
    - A website/database for the government to easily access/use data
    - Export to a cwd file
    - Export/Pipeline an alternative way into their database (need more info)

~Sketches include in D-Lab physical notebook~

Deliverable 6:

Choose A Design Direction

- Investigate your top 5 design concepts using your evaluative matrix

Criteria	Qualitative/ Quantitative	Testing Procedure	Target Value	Metric
Mobile Performance	Quantitative	Use Performance Monitor to check FPS	60 FPS	Frames Per Second
Readability/ Maintainability	Qualitative	Writing tests on backend to ensure properly written functions and performing strict code review to evaluate how closely new code follows best practices/ design patterns	10	Scale of 1-10 (10 = code follows
Understandability/Ease of Usability by Public	Qualitative	Have other people evaluate the ability to easily understand the app	10	Scale of 1-10 (10 = easy to understand)
Evolvability of Software	Qualitative	Documentation and Strict Code Review/ Have each developer perform strict code review to ensure separation of concerns and prevent "god-objects"	10	Scale of 1-10 (10 = interoperable software)
Security/ Privacy	Qualitative	Attempt to access sensitive data from a hacker point of view	10	A Scale of 1-10 (10 = app not secure/ easy to hack)

-In terms of UX/UI, most of the design has been determined last quarter
-This leaves us responsible for design with regards to the technological composition of the app.
-Based on this evaluative matrix we've decided that we choose technologies that allow for maintainability and security. These criteria are especially important as we will probably not be the sole software maintainers of the app. Thus, it's important that we choose a framework that would allow for interoperability as well as popularity to have our app survive for years to come as well as having the code base be easily maintainable.
-Security is also another huge component of the app both on the software front and user experience front, as this app collects very sensitive data that could potentially be used to enable malicious intent. Anonymity is an important component of the security our app will provide. Being able to convey that this is a safe and secure app is just as important as having built the app securely.

- As a group, select your top 3 ideas. Use you Design Criteria (deliverable #3) to understand strengths and weaknesses

-Based on the previous criteria and concepts mentioned. We're leaning towards a React Native and Node.js stack for the mobile app and backend, respectively. React Native and Node.js are very popular frameworks that are unlikely to see deprecation anytime soon, and they also allow us to work more effectively and quickly given our time constraints. Maintainability and security will be handled through test-driven development.

- Write down in words pros and cons of each possible direction. How do they compare to the benchmarks you have found?
    - -There are other frameworks we can use such as Django for the backend. Django provides built-in security; however, it has been dropping in popularity and for this reason, we want to stay away (because of un-popularity and smaller community, it is more likely for this technology to be unsupported in the future). We also have the option of writing in native technologies such as Java and Swift for mobile. These technologies will probably never go away as most mobile software is written using these languages. The problem, however, is the maintenance of two codebases for both Android and iOS

Deliverable 7:

See Bibliography

Deliverable 8:

Elevator Pitch:

- Introduction:
    - Skylar and Christie
    - UC Davis Students
- Creating an Android Application for the Venance Soza, who works as a technical Environmental advisor for the Tanzania government, App will

- Allows users to anonymously track and report environmental incidents to the Government
- Allows government a database of all incidents with their status and general information

Prior Art:

- UI predominantly Based on previously developed frameworks from DLAB last quarter

~Show images of old mock ups~

- Backend based on needs of Government/ speaking with our client
- Unclear on exportable database service -> large CSV files, but plenty of other options

Considerations:

Criteria	Qualitative/ Quantitative	Testing Procedure	Target Value	Metric
Mobile Performance	Quantitative	Use Performance Monitor to check FPS	60 FPS	Frames Per Second
Understandability/Ease of Usability by Public	Qualitative	Have regular phone users evaluate the ability to easily understand the app	90%	Percentage of Users who can successfully create a post
Security/ Privacy	Qualitative	Attempt to access sensitive data from a hacker point of view	10/10	Had to hack, users are untraceable

Budget and Timeline:

- Frontend

Apart from bug fixes and updates, the front end should not cost any significant value to maintain

- Server

The backend will have an initial cost of $5 a month. Depending on the amount of users, the backend may require more space on a server, which would increase costs.

- Database

The current database we are using is considered good for smaller projects, but not recommended for large scale projects. Depending on the number of users our database may require a change which would take up more room on the server and could increase costs

Current Design Concepts:

See Design mockups in D-Lab folder

Deliverable 10

Our prototype does is currently functional, as we have tested creating requests with it as well as testing the UI. While we have tested our product in house, we still need users to test the product and attempt to overload and/or break it. Our product includes basic functionality but is still missing certain features. Missing desired features include adding a remote backend server and an access point for the client to access the data on the server. With all of our work, our product satisfies the Project Goal Statement and even goes a bit further to help the client access and use the data. When speaking with our client, the main discussion revolved around how he would like to use and view the data. We concluded that if possible, instead of a full website as few simple scripts would suffice until later improvements could be added. This would allow the client functionally use the data without requiring another large time consuming project. From our design table the two major improvements that need to be addressed are the security needs and usability, neither of which have been tested.

Deliverable 11:

The client has discussed changed the frontend to include a map instead of a scrollable feed and requested a website to help utilize the data from the users. Unfortunately, our testing plan fell through as Skylar lost his computer and did not upload his changes to Github. This means that there has been no progress on the testing mentioned in Deliverable 10. Being that the website is slightly out of the project scope, we agreed to create a temporary website and offered other recommendations on how to fully finish the interactive website using UC Davis resources.