Names: Anushka Krishnakumar, Lisa Walker, Gavin Bowman, Katrina Gutierrez, Kevin McGrath

Mid-Semester Collaboration Report

Sustainable Media

Stage I: Team Identification

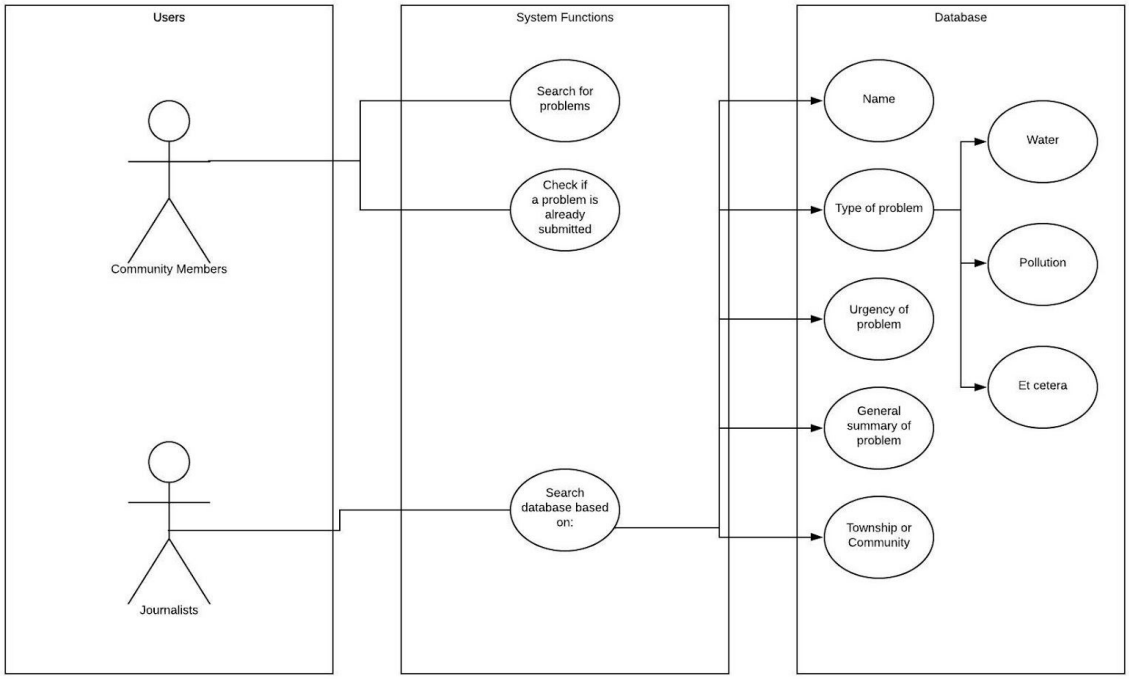
Name	Github ID	Email	Phone Number
Anushka Krishnakumar	Anushkakk	krishna6@tcnj.edu	339-206-8324
Lisa Walker	lwalk8	walkel10@tcnj.edu	609-781-4199
Gavin Bowman	bowmang1	Bowmang1@tcnj.edu	908-763-8773
Katrina Gutierrez	katCloi	gutierk2@tcnj.edu	732-927-0341
Samantha Russo	N/A	russos8@tcnj.edu	347-979-6101
Madison Pena	N/A	penam9@tcnj.edu	917-929-4090
Kevin McGrath	mcgratk3	Mcgratk3@tcnj.edu	732-425-4458

Stage II: Elaboration - Project Proposal Specifications

1. **Problem statement**
    a. In New Jersey, there are several towns and municipalities that have several problems regarding sustainability. These problems could possibly include pollution, water scarcity, energy, food production, and other environmental problems. Although most people in a particular town may be aware of their sustainable issues, others in Jersey would not be. If other people in New Jersey are not aware of the problems affecting their communities, then it's impossible for those problems to be solved. If more people knew about their sustainable issues, then it's possible that a solution could be provided.
2. **Objective of the module.**
    a. The objective of the module is to bring awareness to sustainable issues in New Jersey communities. Our web application should be able to help users have a means of voicing out local sustainability issues while also giving journalists a platform to write about these issues. Members of New Jersey communities and journalists will both benefit from our website because it will give them both a voice and possibly help provide a solution for their local problems.

3. **Description of the desired end product, and the part you will develop for this class.**
    a. For this project, we would like our end product to be a functioning website for both journalists and members of the community. Residents will be able to log on and submit problems going on in their communities, while journalists will be able to access these submitted problems and write articles about them to post on the website. Soon after the articles are written, residents of the community will be able to read them and understand how the problem will be addressed. We hope to implement all these features this semester so that way it could possibly be integrated into the SR Hub website.
4. **Description of the importance and need for the module, and how it addresses the problem.**
    a. This module will help solve problems in communities at a much quicker pace. The quicker these small problems are fixed, the more time a community can spend on improving itself in many different areas, such as technology and sustainability. Not only will it allow journalists to write about issues, it will also let any resident read about a problem so they are aware of it and could even try to help solve it. By adding this feature to the SR Hub website, communities could come together more than ever before to address these issues.
5. **Plan for how you will research the problem domain and obtain the data needed.**
    a. JPW students will do outreach and reporting to find real world problems that CS students can add into the database, as well as brainstorming ways to improve the site and its accessibility.
    b. Each piece reported from the data collected by the JPW students will be stored manually at first. The data represented by the resident "users" will be stored on the website via a form. Then accredited journalists will have access to write articles based on what is written on these forms by the residents.
6. **Other similar systems / approaches that exist, and how your module is different or will add to the existing system.**
    a. One similar system is potentially Facebook. With pages/profiles dedicated to this kind of work, people are able to collaborate. However, our site would be all on one page, which could potentially be different than this system. It would allow for a more concise collaboration, and can distinguish articles written by certified journalists from users who are raising the issues initially.
    b. We will be adding to the websites that already exist
        i. Civicstory.org
        ii. SRhub.org
7. **Possible other applications of the system (how it could be modified and reused.)**
    a. It can be used for schools
        i. Problems in a school environment can be made aware of

8. **Performance – specify how and to what extent you will address this.**
    a. Fast, people both members of communities and journalists should be able to easily access the forms or database.
    b. It will also be easy to navigate and user-friendly. (ADA compliant)
9. **Security – specify how and to what extent you will provide security features.**
    a. Have employees of the website verify a form once it is submitted by a user
    b. Firewalls to make sure the website is not hacked
10. **Backup and recovery – specify how and to what extent you will implement this.**
    a. We would backup the website using a tool
        i. Github?
    b. For database, we can back it up through drive & github
11. **Technologies and database concepts the team will need to learn, and a plan for learning these.**
    a. Postgres
    b. SQL
    c. Python
    d. JavaScript
    e. HTML
    f. CSS
12. **A diagrammatic representation of the system boundary that specifies what data you will model and which queries you will implement.**
    a. Members of Community:
        i. Search for problems
        ii. Check if a problem is already submitted
    b. Journalists will be able to search the database based on:
        i. Name
        ii. Type of problem
            1. Water, pollution, etc.
        iii. Urgency of problem
        iv. General Summary of your problem
        v. Township/Community

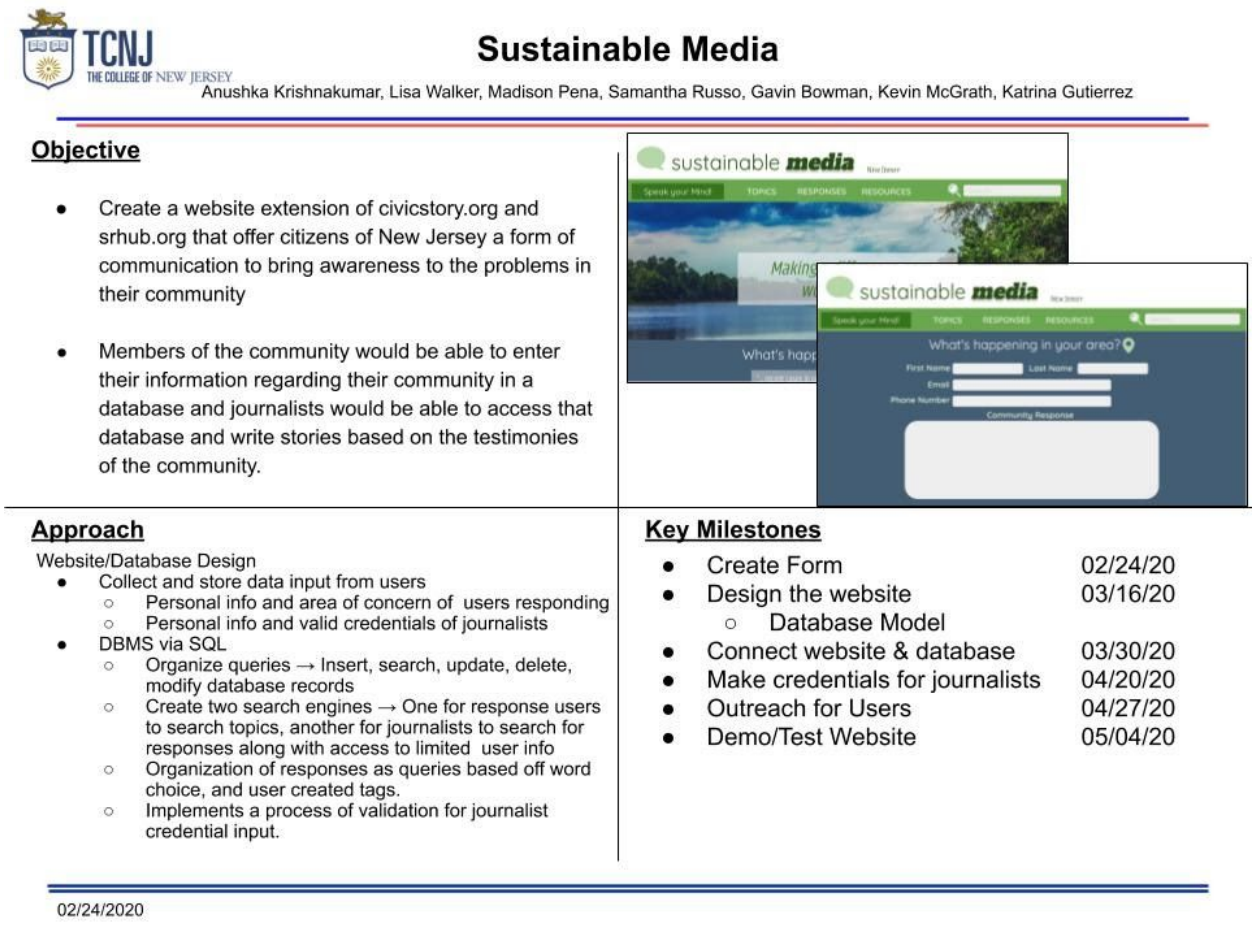


13. Name of the Website:

a. Sustainable Media

14. Roles of Members:

a. Anushka Krishnakumar: Website Design/Database/Project Leader
b. Lisa Walker: Website Design/Database
c. Madison Pena: User Outreach/Form Verifiers
d. Samantha Russo: User Outreach/Form Verifiers
e. Gavin Bowman: Database/Security
f. Kevin McGrath: Database/Security
g. Katrina Gutierrez: Website Design/ Database



Stage III: Elaboration - Database Model

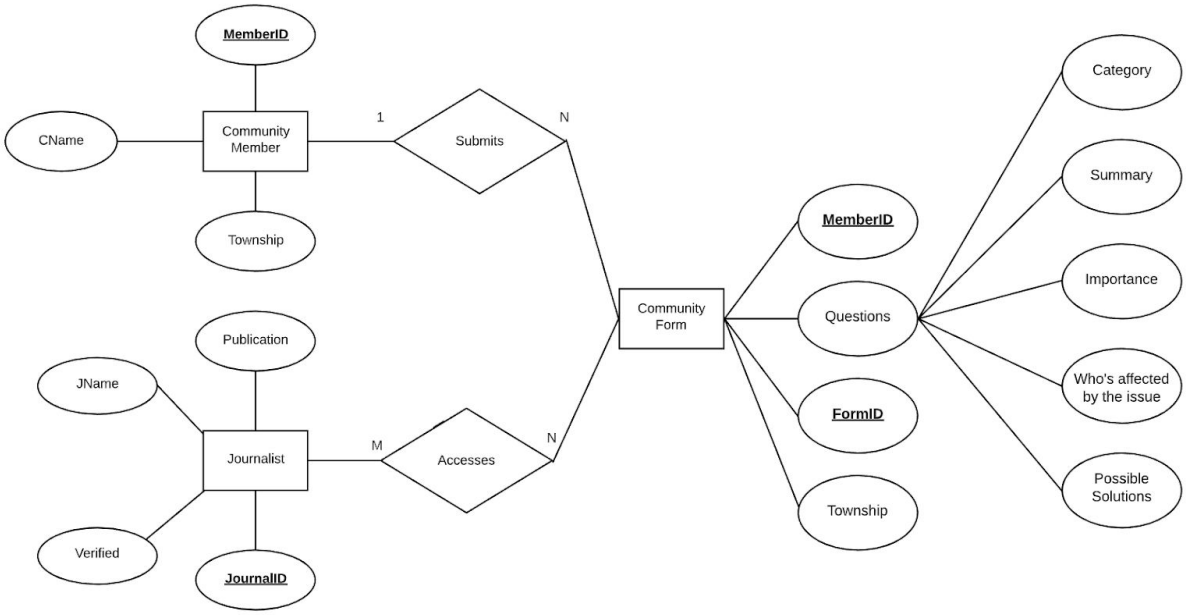
ENTITIES: Attributes

- **Journalist: JournalID, JName, Verification, Publication**
- **Community Member: MemberID, CName, Township,**
- **Community Form: FormID, MemberID, Township, Questions(Category, Summary, Importance, Who's Affected by Issue, Possible Solutions)**

Map the ER diagram to a relational schema

- **Community Member Submits Form (1:N)**
- **Journalist Accesses Form (M:N)**

https://www.lucidchart.com/invitations/accept/06969d02-1f46-4739-84ea-88f181d1cf50



Additionally, based on your research and understanding of the social justice issue, estimate the following:

- We anticipate our database to initially be limited in size based on our resources. We hope to be able to populate it with at least one hundred entries initially. Ideally having one hundred journals available for members to consult will give them the necessary resources to continue to contribute and add to the database.
- Types and average number of searches
    - Searching through the database content is predicted to be more useful for journalists than community members. Types of searches we estimate that they would use are keyword searches either by searching the words or a typed out string. String searches would use a stop word search function that selects only keywords that are followed by a stop word (such as *the, an, at, for, from, then*). These queries then search through all user submissions and select where it is most frequently used in the form. Categorical searches may also be implemented for better searching through the forms. If a categorical search is implemented for journalist access only, for example, to find problems in a certain community, then queries will be modified to allow searches that collect attributes that match the search in that column of the database.
    - A categorical and keyword search may be also used for community members who want to browse through problems in their communities if we do to implement it.
    - The average number of searches we estimate to occur is limited. The number of searches would be from the community members and journalists, which we believe will be around a few searches per member. We do not expect our search queries to back up too much.

Stage IV – Elaboration: Design

Each team will review the Database Model document with stakeholders, and update the model as needed.

- JOURNALIST: (JournalID) → JName, Verified, Publication
- COMMUNITY_MEMBER: (MemberID) → CName, Township
- COMMUNITY_FORM: (FormID) → MemberID, Township, Questions (Category, Summary, Importance, Who's Affected by Issue, Possible Solutions)

*UPDATE***: Separate databases are now a single database*

Demonstrate that all relations in the relational schema are normalized to BCNF.

- For each table, specify whether it is BCNF or not, and explain why.
    - All of our tables are in BCNF. Given that each table has only one primary key, they cannot be derived any further. More specifically, in the JOURNALIST table, JournalID determines JName and publication; in COMMUNITY_MEMBER determines CName and township; in COMMUNITY_FORM, FormID determines CName, MemberID, Township, and Questions. Since each table determines these attributes independently from the others, we know that they are all in Boyce-Codd Normal Form.
- For each table that is not in BCNF, show the complete process that normalizes it to BCNF
    - All of our tables are BCNF.

Define the different views required. For each view list the data and transaction requirements.

- In a database, a view is the result set of a stored query on the data, where the database users can query just as they would in a persistent database collection object.
    - The journalist users will be able to create read-only views with their queries.
        - They can view community forms, whether or not a journalist has been verified, list the names of people who have submitted community forms.
    - Sustainable Media Database Administrators will be able to create updatable views with their queries.
        - They can INSERT, UPDATE, and DELETE community forms, verified journalists, names of people who have submitted forms and other actions/queries.
    - To make our database secure, we will be using query authorization statements such as GRANT and REVOKE, in order to hide certain tuples from the journalist users.
    - Journalists will only be able to view the COMMUNITY_MEMBER database.

- Give a few examples of queries, in English, to illustrate.
    - List community forms from the township "Ewing"
        - $\pi_{\text{Township = "Ewing"}}$ COMMUNITY_FORMS
    - List community forms that categorize under the "Pollution" category

        - $\pi_{\text{Category = "Pollution"}}$COMMUNITY_FORMS
    - List community members that have submitted a form
        - $\pi_{\text{Fname, LName}}$(COMMUNITY_MEMBER $\bowtie_{\text{(MemberID)}}$ COMMUNITY_FORMS)
    - List Journalists that have been verified (Assuming a JournalID is a valid number once they are verified)
        - VERIFIED $\rightarrow \pi_{\text{JournalID}}$ JOURNALIST
        - UNVERIFIED $\rightarrow \pi_{\text{JournalID= NULL}}$ JOURNALIST
        - ALL_VER $\rightarrow$ VERIFIED - UNVERIFIED
        - RESULT $\rightarrow \pi_{\text{FName, LName}}$ (JOURNALIST * ALL_VER)

Design a complete set of queries to satisfy the transaction requirements identified in the previous stages.

- SELECT * FROM JOURNALIST;
    - This is a query in SQL with everything stored in the JOURNALIST table
    - It will have information for all the following attributes:JournalID, JName(FName, LName), Publication
- SELECT * FROM COMMUNITY_FORM;
    - This is a query in SQL with everything stored in the COMMUNITY_FORM table
    - It will have information for all the following attributes: FormID, CName(FName, LName), MemberID, Township, Questions(Category, Summary, Importance, Who's Affected by Issue, Possible Solutions)
- SELECT * FROM COMMUNITY_MEMBER;
    - This is a query in SQL with everything stored in the COMMUNITY_MEMBER table
    - It will have information for all the following attributes: MemberID, CName(FName, LName), Township

Stage V(a) — Construction: Tables in PostgreSQL & Queries in SQL

a. Write and execute SQL data definition queries to create the tables and views in PostgreSQL. Ensure that all constraints are specified in the queries. Write scripts / programs that may be required to obtain and format the data. Populate the tables with valid data with all constraints being enforced.
b. Write SQL data manipulation queries that were designed in the previous stages. The queries must be elegant and make effective use of complex query constructs such as subqueries. Execute and test these queries to ensure that they work correctly. Examine the outputs carefully to verify that the queries do not return tuples.

Deliverables:

Submit to Canvas

- Test files with the data definition and population queries (submitted with .txt extension).
- Data files
- Scripts used to obtain and format data and populate tables
- Text files with the data manipulation queries (submitted with .txt extension)
- SQL script files that demonstrate successful execution of the queries
- Documents for previous stages, revised if applicable

Submit on the VM

- The schema created in PostgreSQL, and the tables populated
- Text files with the data definition and population queries (submitted with .txt extension)
- Data files
- Scripts used to obtain and format data and populate queries (submitted with .txt extension)
- SQL script files that demonstrate successful execution of the queries

Submit on GitHub

- Text files with the data definition and population queries (submitted with .txt extension)
- Scripts used to obtain and format data and populate tables
- Text files with the data manipulation queries (submitted with .txt extension).
- Well-documented source code files
- Documents for previous stages updated if applicable
- Project milestones and issues create and/or updated

```
osc@osc-VirtualBox:~/Downloads/Exercises/Stage_V$ psql -f createTables.sql test_SM
CREATE TABLE
CREATE TABLE
CREATE TABLE
osc@osc-VirtualBox:~/Downloads/Exercises/Stage_V$ psql -f Tables_Views.sql test_SM
INSERT 0 5
INSERT 0 6
INSERT 0 5
osc@osc-VirtualBox:~/Downloads/Exercises/Stage_V$ psql -f Queries.sql test_SM
      jname      | journalid | verified |   publication
-----------------+-----------+----------+-----------------
 Jenna Foxworthy |    314159 | Verified | New York Times
 Bill Haggerdy   |    987654 | Pending  | New York Times
 Harry Truman    |    456789 | Pending  | People Magazine
 Jeff Marbles    |    123456 | Verified | Vogue
 Helin Hopper    |    385768 | Verified | The Atlantic
(5 rows)

      cname       | memberid |   township
------------------+----------+--------------
 Lori Long        |        1 | Ewing
 Aiesha Gilliam   |        2 | Trenton
 Tim Simons       |        3 | Trenton
 Keyleigh Galindo |   293847 | Warren
 Debbie McGrath   |   439827 | Dunellen
 Zac Kavanagh     |    98765 | Robbinsville
(6 rows)

 formid | memberid | township |    category     |         summary         | importance |   affected   |       solutions
--------+----------+----------+-----------------+-------------------------+------------+--------------+-----------------------
      1 |        1 | EWING    | Water           | Dirty Water             | High       | Neighborhood | Replace Tank?
      2 |        2 | Trenton  | Pollution       | Air Pollution           | High       | Township     | Contact Business Owner
      3 |        3 | Trenton  | Water           | Dirty Water             | High       | Township     | Change Water Tank
      4 |   293847 | Warren   | Pollution       | Air Pollution           | High       | County       | Contact Factory Owner
      5 |   439827 | Dunellen | Hazardous Waste | Neighbor has lead paint | Medium     | Neighborhood | Re-paint house
(5 rows)

 memberid | importance |    summary
----------+------------+--------------
        1 | High       | Dirty Water
        2 | High       | Air Pollution
        3 | High       | Dirty Water
   293847 | High       | Air Pollution
(4 rows)
```

```
 memberid | category | summary | importance | township
----------+----------+---------+------------+----------
(0 rows)

 memberid | category  |    summary    | importance | township
----------+-----------+---------------+------------+----------
       2 | Pollution | Air Pollution | High       | Trenton
  293847 | Pollution | Air Pollution | High       | Warren
(2 rows)

      cname
------------------
 Lori Long
 Aiesha Gilliam
 Tim Simons
 Keyleigh Galindo
 Debbie McGrath
(5 rows)

      jname
-----------------
 Jenna Foxworthy
 Jeff Marbles
 Helin Hopper
(3 rows)

osc@osc-VirtualBox:~/Downloads/Exercises/Stage_V$ psql -f dropTables.sql test_SM
DROP TABLE
psql:dropTables.sql:2: NOTICE:  drop cascades to constraint community_form_memberid_fkey on table community_form
DROP TABLE
DROP TABLE
osc@osc-VirtualBox:~/Downloads/Exercises/Stage_V$
```

Stage V(a) — Construction: Tables in PostgreSQL & Queries in SQL

- UI was demonstrated in class.
- Well documented source code can be found in the GitHub repository.