Litterbug – A Software Approach to Littering

DAVID MEDNIKOV, Oregon State University
HITESH VARMA, Oregon State University
KEVIN ALLEN, Oregon State University
ALEXANDER YFRAIMOV, Oregon State University

1. USER STORIES

1.1 Register for the app

- **Description** – In order to use the app, we will require registration through email or Facebook. If registered through email, the user will have the option of using as alias as their username.
- **Due Date** – This story will be due this week.
- **Tasks**
    1. Set up database to store user profile, this includes the users names, locations and ranking. **1.5 units.**
    2. Set up front page of the app. **2.5 units.**
- **Order of development** – The database will be created first because it will be faster to create the required columns and rows for our project, than determining the design of the front page. Also the code to getting the API to log in for different mobile devices will probably take longer to get working.
- **Time Estimate** – It will probably take a team of two programmers between a **day or two** to complete this task.

1.2 Donate to a charity

- **Description** – A user will be able to click on a button at various stages of using the app and be able to donate to charity.
- **Due Date** – This is a story that we will not be implementing.
- **Tasks**
    1. Set up a database of existing charities. **1.5 units.**
    2. Create a app screen with a list of charities. **1.5 units.**
    3. Set up a method to help users add charities that currently are not available. **1.5 units.**
    4. Setup a payment API through code which accepts various forms of payment **2.0 units.**
- **Order of development** – The database and page design portion would be the fastest to create. Coding a method to add new charities would be faster than setting up and testing the APIs for payment processing.
- **Time Estimate** – This could take approximately 3 days to complete.

1.3 Mark an area as cleaned

- **Description** – Once a user has completed cleaning an area they mark the area they have cleaned as green on the map, indicating they have cleaned it to other users.
- **Due Date** – This is a story that we will not be implementing.

This work is supported by Michael Johnson of Oregon State University.

- **Tasks**
    1. Create app page for current area to be viewed. **1.5 units.**
    2. API for Google Maps will be integrated with page through code. **2.5 units.**
    3. Overlay Color Coding of the App onto the API via code. **2.0 units.**
- **Order of development** – Creating the page for the task will be the fastest, since the style of the app would have been determined from the login page. The Google API will then be the second longest thing to implement, thereafter integrating it with our app should take the longest time.
- **Time Estimate** – This will take 3 to 4 days to complete.

1.4 Submit photo when marking an area as cleaned

- **Description** – After a user has marked an area as cleaned, they will post a picture of the cleaned area.
- **Due Date** – This is a story which the team has determined will be implemented next week.
- **Tasks**
    1. Designing the screen to upload the picture. **1 unit.**
    2. Upload the picture to the database. **1 unit.**
- **Order of development** – The fastest task to do is to complete the app page, then add the code to upload and store the picture to the database.
- **Time Estimate** – This would take approximately 2 days to complete.

1.5 Verify that a user cleaned an area

- **Description** – In order to prevent abuse of the system and users receiving unwarranted credit for work they have not done, other users are able to verify if an area has been cleaned.
- **Due Date** – This is a task the team has determined will be implemented next week.
- **Tasks**
    1. User gets a notification that a red area or uncolored area has been turned on as green. **0.5 units**.
        a. This will be the fastest of the tasks to write.
    2. Write code for secondary user to verify that an area is clean and save that information to the database. **1 unit**.
        a. This would take longer to write as it involves other elements of the program
- **Time Estimate** – This would take approximately two days to complete.

1.6 Report a user

- **Description** – This will allow other users to choose from a popup list which includes items like “Submitted Task but Didn’t Clean”, “Hateful/Abusive Speech”, “Self Harm” “Spam” etc.
- **Due Date** – This will be due this week.
- **Tasks**
    1. Design User Profile Page with Button to report. **1 unit.**
        a. The design in CSS/HTML with Javascript coding should be the shortest to code.
    2. Code popup with options to report. **1.5 units.**
        a. This should take slightly longer than the above.
    3. Store report in users profile in database. **0.5 units.**
        a. This would be done last after the coding portion is tested successfully.
- **Time Estimate** – This will take 2 to 3 days to complete.

1.7 Earn badges/medals

- **Description** – When a user's cleaning area has been verified, or for various other actions (such as donating, confirming a cleaned area, challenging a friend, etc.) they will receive badges.
- **Due Date –** This will be due next week.
- **Tasks**
    1. Record in database user's cleaning has been validated**. 0.5 units.**
    2. Write code so that once validation is received, merits such as badges and medals are implemented**. 1 unit.**
- **Time Estimate –** This should take approximately 2 days to complete.

1.8 View badges and progress towards badges

- **Description** – A user can view their current status in terms of badges and medals.
- **Due Date –** This was chosen not to be implemented.
- **Tasks**
    1. Create page that shows the status of the user in terms of badges and medals. Show current ranking and how far the user is in terms of maximum ranking. **0.5 units.**
    2. Code access to database to upload status of user ranking. **1 unit.**
- **Time Estimate –** This would take 2-3 days to complete.

1.9 Comment on a user's activity

- **Description** – After a user's activity has been verified, people can make comments on their achievements in that users profile.
- **Due Date –** This was be implemented next week.
- **Tasks**
    1. Write code for database to upload list of achievements. **0.5 units**.
        a. This should be done first since it is easier to do than designing the profile page.
    2. Design profile page with list of achievements and space to comment. **1 unit.**
    3. Write code to save users comments in database. **1 units.**
        a. This needs to be done last since it the most difficult part and we want to complete the easier tasks first where logically possible.
- **Time Estimate –** This would take 2 days to complete.

1.10 Exchange points for currency to donate to a charity

- **Description** – The user can use their current points balance to convert to currency to donate to charity.
- **Due Date –** This will be due this week to implement.
- **Tasks**
    1. Write code to read data from database regarding points balance. **0.5 units**.
        a. This would be the shortest step and therefore should be done first
    2. Write code to check exchange rate and convert points into currency amount. **0.5 units**.
        a. This would take slightly longer than the above.
    3. Create app page to donate to charity with API for payment. **1.5 units**.
        a. This would take the longest because of the API testing.
- **Time Estimate –** This will take 2-3 days to complete.

1.11 Challenge another user

- **Description** – A user can challenge another user to clean up an area.
- **Due Date** – This has been chosen not to be implemented.
- **Tasks**
    1. Create a user profile would be the fastest first step to take. **0.5 units**.
    2. Code button to challenge a user would be the fastest step to complete after that. **0.5 units**.
    3. Code notification to user being challenge. This would be the next logical task to complete. **1 unit**.
    4. Code acceptance of challenge for user. This would be the next logical progression. **0.5 units**.
    5. Code acceptance of challenge to challenger by challenge. This step relies on the other steps being completed so will be coded last. **0.5 units**.
- **Time Estimate** – This will take 3-4 days to implement.

2. USER STORIES DUE THIS WEEK

2.1 Report a user

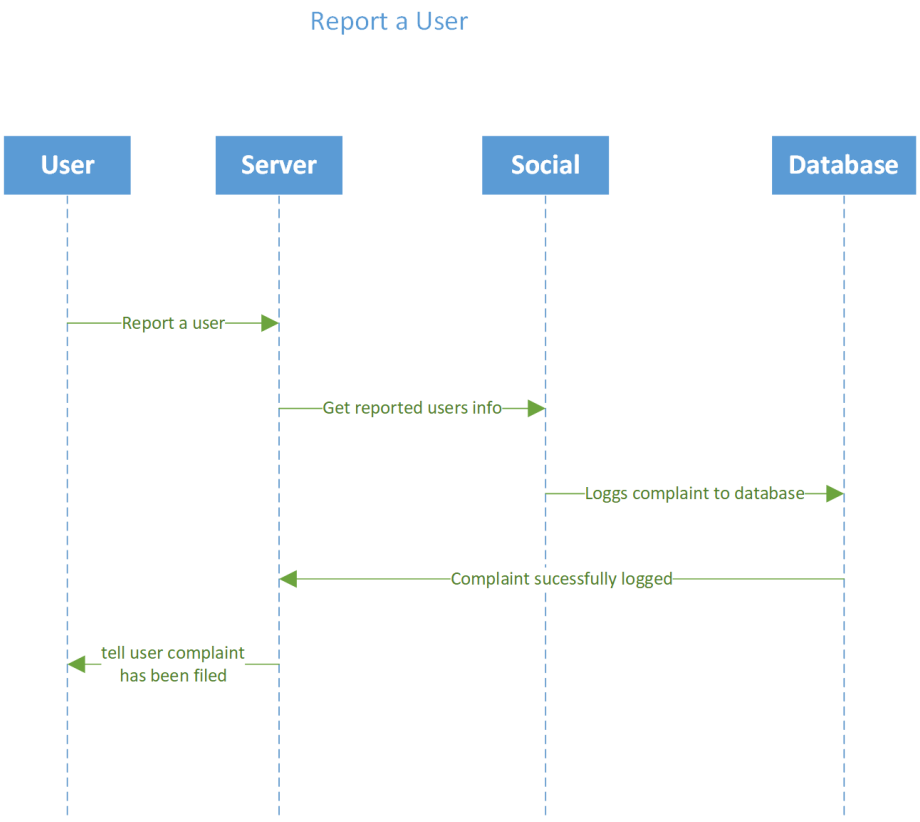

Fig. 1. "Report a User" user story UML diagram

2.2 Exchange Points for Currency to Donate to Charity

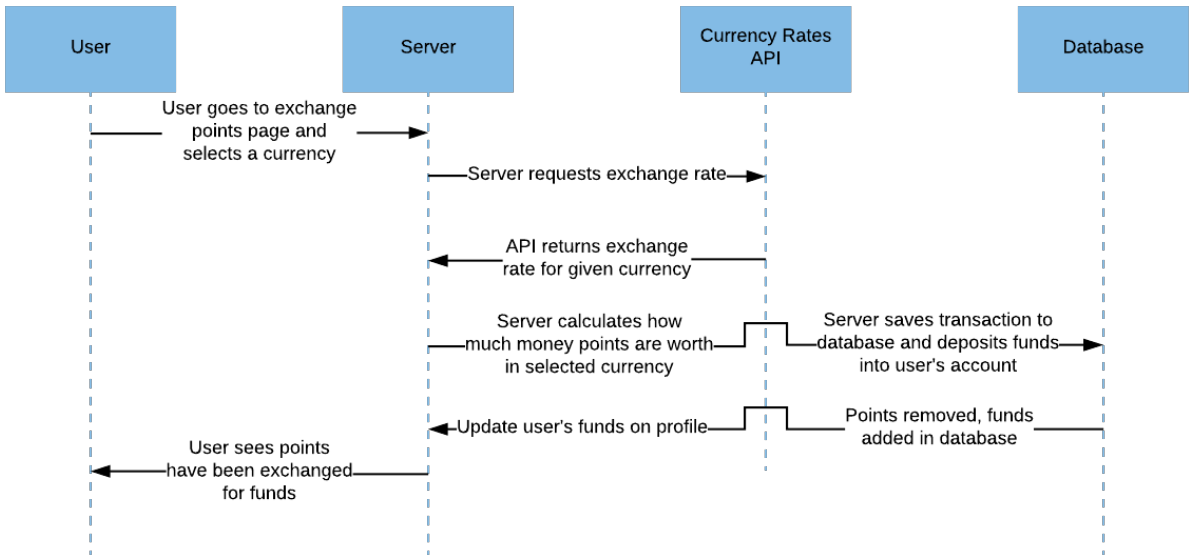

Fig. 2. "Exchange Points for Currency to Donate to Charity" user story UML diagram

2.3 Register for the app

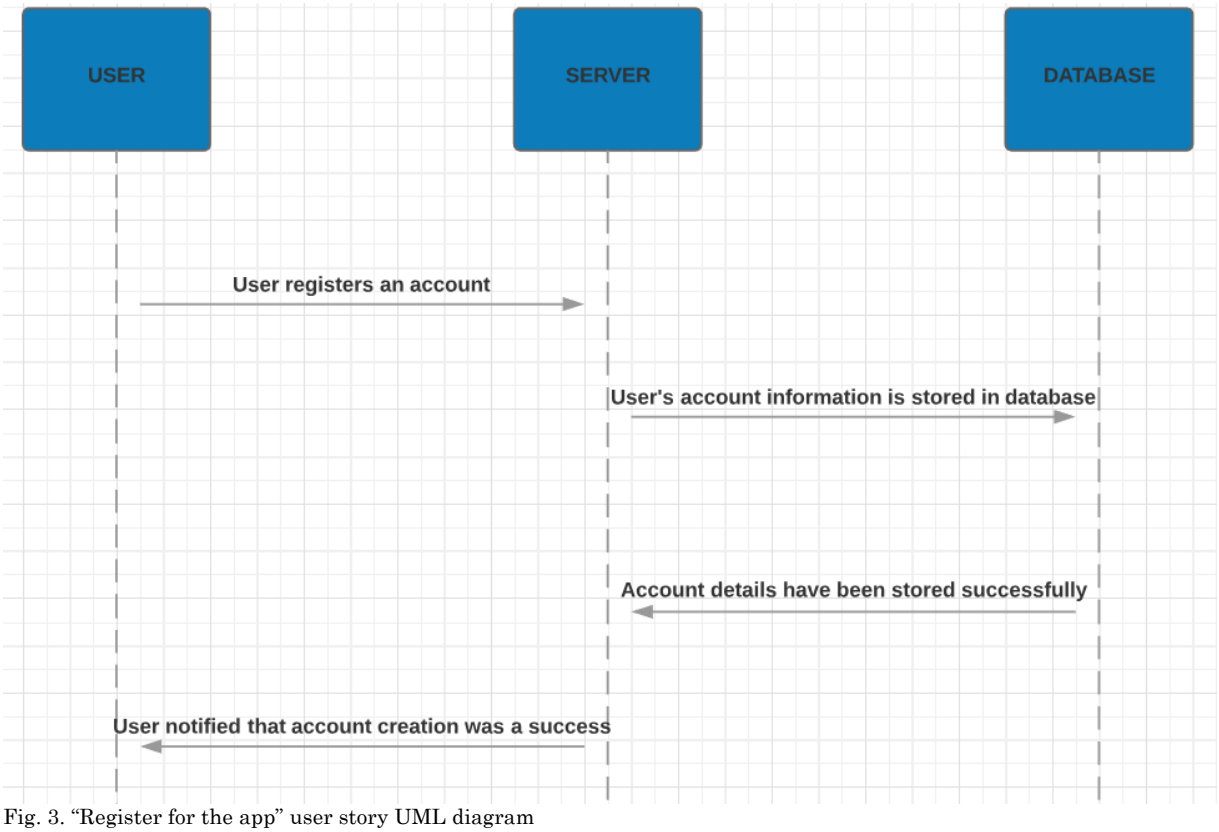

Fig. 3. "Register for the app" user story UML diagram

3. PLANS FOR USER STORIES DUE NEXT WEEK

3.1 Verify that a user cleaned an area

David and another teammate will be implementing the ability to verify that a user did in fact clean an area next week. The fastest task to implement will be the notifications for users asking them to verify that a green area that they are located in or near that was recently marked as cleaned, was actually cleaned. Then we would write the code where the second user is able to leave some sort of feedback about the task and submit it to our server. Last, it should trigger the system to give or revoke badges, medals, or other incentives to the user that initially submitted the task.

3.2 Earn badges/medals

Alex and another teammate will be implementing the earning badge/medals functionality next week. This particular user story has a dependency on another one of the user stories in the sense that it relies heavily on the output of that other user story. That other user story, once implemented, will introduce the function of verifying that a user cleaned an area.

In order for the badge/medal functionality to execute properly it must receive confirmation from the verification function to ensure we don't issue rewards to users who may be making false reports about cleaning an area in an effort to acquire badges/medals. This means that we will need to work closely with the other pair who will be implementing that function so we know what we should be expecting as output from that verification function.

3.3 Comment on a user's activity

Hitesh and another teammate will be implementing the ability to comment on a user's activity next week. The first step will be to write code for the database where the user's list of tasks can be stored. This should be done first since it is easier to do than designing the profile page. Then the profile page with list of tasks should be designed, with space left to comment. The last step would be to write code to actually save user comments in the database. This needs to be done last because it is the most difficult part and we want to complete easier tasks first when possible.

3.4 Submit a photo when marking an area as cleaned

Kevin and another teammate will be implementing the ability to attach a photo when submitting a task that an area was cleaned. This will be done next week. Since the easiest part of this will be the UI, we will first develop the app page where users can attach a photo when submitting a task for an area they cleaned. After that, we will write the code to store the photo in the database and associate it with the task entry itself.

4. MEETING WITH THE CUSTOMER

The user was willing and able to meet with us on the evening of Wednesday, August 1st. The customer was very reasonable with the work he expected from us by Tuesday.

5. SUMMARY OF GROUP MEMBER CONTRIBUTIONS

- David Mednikov
    - Created a UML diagram for the "Exchange points to currency" user story and wrote the plans for implementing "Verify that a user cleaned an area" next week
    - Compile the Google doc into the ACM format

- Hitesh Varma
    - Wrote the tasks, descriptions, and time estimates for our 11 user stories
    - Added plans for the user stories that we will be implementing next week
    - Wrote the plans for implementing “Comment on a user’s activity” next week
- Alexander Yfraimov
    - Created a UML diagram for the “Register for the app” user story and wrote the plans for implementing “Earn badges/medals” next week
- Kevin Allen
    - Created a UML diagram for the “Report a user” user story and wrote the plans for implementing “Submit photo when marking an area as cleaned” next week