

M4 - Login/Logout and User Stories

Team 64

Link to Github repository

<https://github.com/spxShaw/Clean-Water-Android-Application.git>

User Stories

- 1) As a User, I want to submit water availability reports so that I can tell people if water is available near me. --**Christopher**
- 2) As a User, I want to see existing water purity and availability reports so that I can know where to go for clean water.
- 3) As a User, I want water source reports to include Location, Type of Water, and Condition so that accurate data can be acquired.
- 4) As a Worker, I want to be able to submit water purity charts so that other people can know if a water source is safe to drink. --**Lucille**
- 5) As a Worker, I want Purity reports to include Location, Water Condition, Virus PPM, and Contaminant PPM so that other people can know if a water source is safe to drink.
- 6) As a Manager, I want Purity reports to include Date, Time, Report Number, and the name of the Worker so that I can track employee work.
- 7) As a Manager, I want to view historical reports and trends in water purity so that I can redirect workers to areas with declining water quality. --**Robert**
- 8) As a Manager, I want Availability reports to include Date, Time, Report Number, and the name of the reporter so that I can have historical reports.
- 9) As a Manager, I want to delete inaccurate reports so that users can get accurate information and historical composite reports are not negatively influenced. --**Matt**
- 10) As an Administrator, I want to delete accounts so that unused accounts can be cleaned out.
- 11) As an Administrator, I want to ban specific users from submitting reports so that bad actors don't create bad report spam.

12) As an Administrator, I want to unblock accounts that have been locked for incorrect login attempts so that users can access their account. --**Spencer**

13) As an Administrator, I want to view an application security log so that I can review application history and potential unauthorized access to user accounts.

Individual User Stories

Spencer Shaw

Title: As an Administrator, I want to unblock accounts that have been locked for incorrect login attempts so that users can access their account.

Tasks:

- create a SecurityLog
- make changes to Login to store all login data
- make changes to Login so 'locked' accounts are blocked
- Change accounts so that they have a locked/unlocked setting

Acceptance Scenarios:

- GIVEN an administrator is viewing the security log, WHEN the administrator enters a valid system username, THEN the security log will be filtered to show that username's activity.
- GIVEN an administrator is viewing the security log, WHEN the administrator enters an invalid system username, THEN the security log will notify the user that the username isn't valid.
- GIVEN an administrator is looking up a valid username within the security log, WHEN the username is selected, THEN then the username status will appear (account locked/unlocked).
- GIVEN an administrator is viewing a username status, WHEN the administrator sets the username status to locked/unlocked, THEN the username will be locked/unlocked.
- GIVEN an administrator is changing a username status to locked/unlocked, WHEN the administrator submits the change, THEN the event will be shown in the security log.

Done Done Criteria:

- Unit tests are written or updated to include the security log.
- Code is evaluated by another team member

Matt Busch

Title: As a Manager, I want to delete inaccurate reports so that users can get accurate information and historical composite reports are not negatively influenced.

Tasks:

- create method to remove report from database
- add a delete report button to view
- add an AlertDialog to activity
- implement changes in controller/activity

Acceptance Scenarios:

- GIVEN a manager or administrator is viewing a report, WHEN the manager presses the delete report button, THEN a confirmation AlertDialog box should be displayed with “yes” or “no”
- GIVEN a manager or administrator is deleting a report, WHEN the manager presses “yes” on the AlertDialog box THEN the report should be removed from the database, and no longer visible on the app
- GIVEN a manager or administrator is deleting a report, WHEN the manager presses “no” on the AlertDialog box THEN the report should not be deleted
- GIVEN a worker or user is viewing a report page, WHEN the worker presses the delete report button, THEN the system displays a warning message and does not delete the report

Done Done Criteria:

- Unit tests are written or updated to include deleting reports
- Code is reviewed by another development team member

Robert Cooper

Title: As a Manager, I want to view historical reports and trends in water purity so that I can redirect workers to areas with declining water quality.

Tasks:

- Add a button that allows a manager to generate a historical report.
- Add a method that fetches data given a location and year.
- Add a method to generate a historical report given data.
- Add a view that displays the XY graph.

- Add a button that returns the manager to the home screen from the historical report.
- Implement changes in controller/activity.

Acceptance Scenarios:

- GIVEN a manager is generating a historical report, WHEN the manager chooses the button generate historical report, THEN the manager is prompted to give a location, year, and either virus or contaminant.
- GIVEN a manager wants to view a historical report, WHEN the manager clicks submit after filling out the required information, THEN the manager is shown the correct XY graph with the historical data.
- GIVEN a manager does not want to view a historical report, WHEN the manager chooses cancel after filling out the required information, THEN the manager is brought back to the previous screen.
- GIVEN a manager is finished viewing a historical report, WHEN the manager chooses the finished viewing button, THEN the manager is brought back to the home screen, and the report is deleted.

Done Done Criteria:

- Unit tests are written or updated to include historical reports
- Code is reviewed by another development team member

Lucille Wang

Title: As a Worker, I want to be able to submit water purity charts so that other people can know if a water source is safe to drink.

Tasks:

- Add a method that allow for submission of water purity charts
- Modify model to allow for updates on new information from the water purity charts
- Add a view that allows for submission of water purity charts
- Add a confirmation AlertDialog
- implement changes in controller/activity

Acceptance Scenarios:

- GIVEN a worker wants to submit a water purity chart, WHEN the worker clicks "submit a water purity chart" THEN the screen switches to another screen that allows the worker to fill out all the pertinent information and upload a water purity chart.

- GIVEN a worker starts filling out a submission for a water purity chart, WHEN the worker clicks the cancel button, THEN the information the worker filled out is not saved, nothing is updated, and no report is submitted.
- GIVEN a worker is submitting a water purity chart, WHEN the worker presses submit, THEN a confirmation AlertDialog box should be displayed to confirm the inputted information and with the buttons “confirm” or “continue editing”.
- GIVEN a worker has reviewed all the information and does not want to make any other changes, WHEN the worker clicks “confirm” on the confirmation AlertDialog, THEN the worker is notified by a new confirmation AlertDialog box telling the user he/she has successfully submitted a water purity chart.
- GIVEN a worker has reviewed all the information and wants to make any other changes , WHEN the worker clicks “continue editing” on the confirmation AlertDialog, THEN the AlertDialog box disappears and takes the user back to the page where he/she were filling out the information.

Done Done Criteria:

- Unit tests are written or updated to include submitting water purity charts
- Code is reviewed by another development team member

Christopher Pham

Title: As a User, I want to submit water availability reports so that I can tell people if water is available near me.

Tasks:

- create water availability report class
- make changes to water availability report class
- make changes to Model class
- add “add report” button to view
- add view for submitting new data
- implement changes in controller/activity

Acceptance scenarios:

- GIVEN a user is viewing reports page, WHEN the user clicks “add water availability report” button, THEN the “create report” page appears and allows the user to create a new report.

- GIVEN a user is editing a report, WHEN the user submits a report with valid data, THEN the model is updated with these changes to the reports database, and the view is updated to reflect changes in the database.

- GIVEN a user is editing a report, WHEN the user submits a report with invalid data (no data, no such location, etc.), THEN no changes are made in model, and a dialog appears alerting the user that he/she has submitted invalid data.

- GIVEN a user is editing a report, WHEN the user clicks on the "back" button, no changes are made to the model, and the display returns to the reports page.

Done Done Criteria:

- Unit tests are written or updated to include submitting water availability reports
- Code is reviewed by another team member

