Fish Tank Monitoring System - Work In Progress

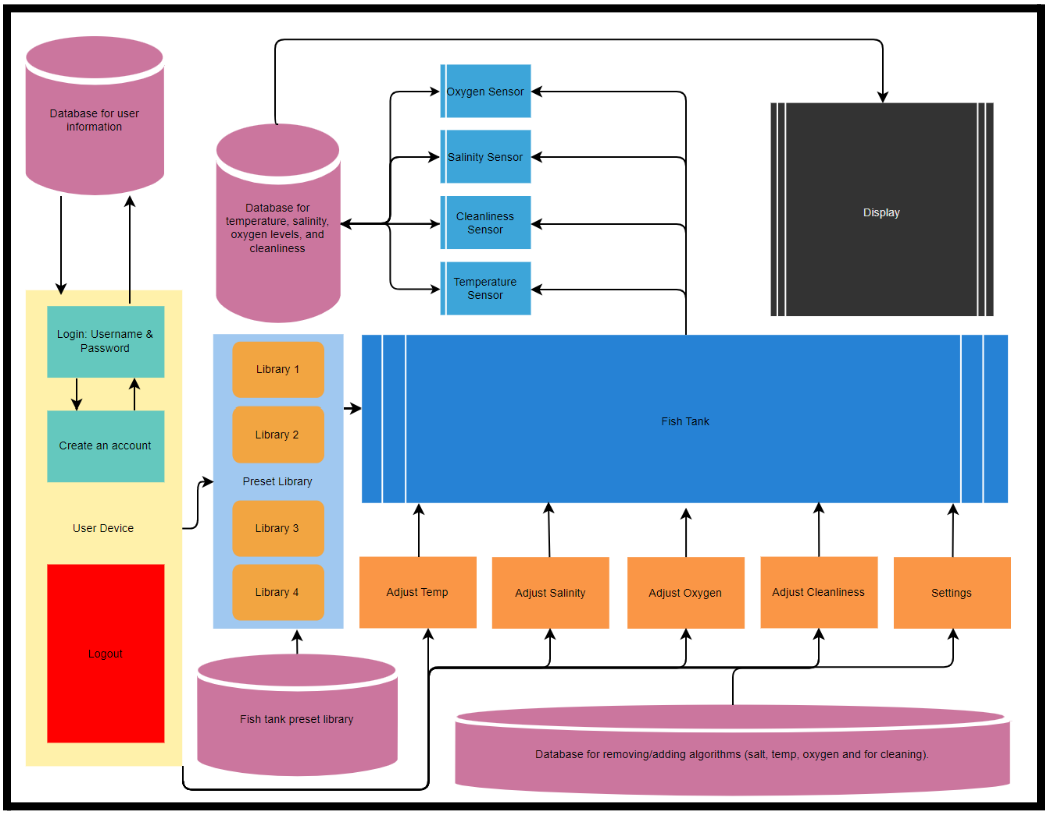
Software Design Specification Document

Development Team:	Role:	Contact:
Jose Garcia	Hardware Engineer	jgarciagomez7895@sdsu.edu
Alex Vo	Software Engineer	avo3681@sdsu.edu
Darren Lee	Security Engineer	dlee0083@sdsu.edu

System Overview

- The idea of the fishtank we are creating is to provide the user with an environment where they are able to access the necessary information to create a thriving aquarium. They will have access to information about a variety of fish as well as what fish can live in similar environments. The user will also be able to adjust certain living conditions for the fish such as temperature, oxygen levels, salinity, and cleanliness. The tank itself will have an automatic chemical balancing feature and will cater to the needs of the users and fish. The user will be able to interact with the fish tank system through the use of a mobile app that interfaces with the tank management system and hardware sensors.

Software Architecture Overview

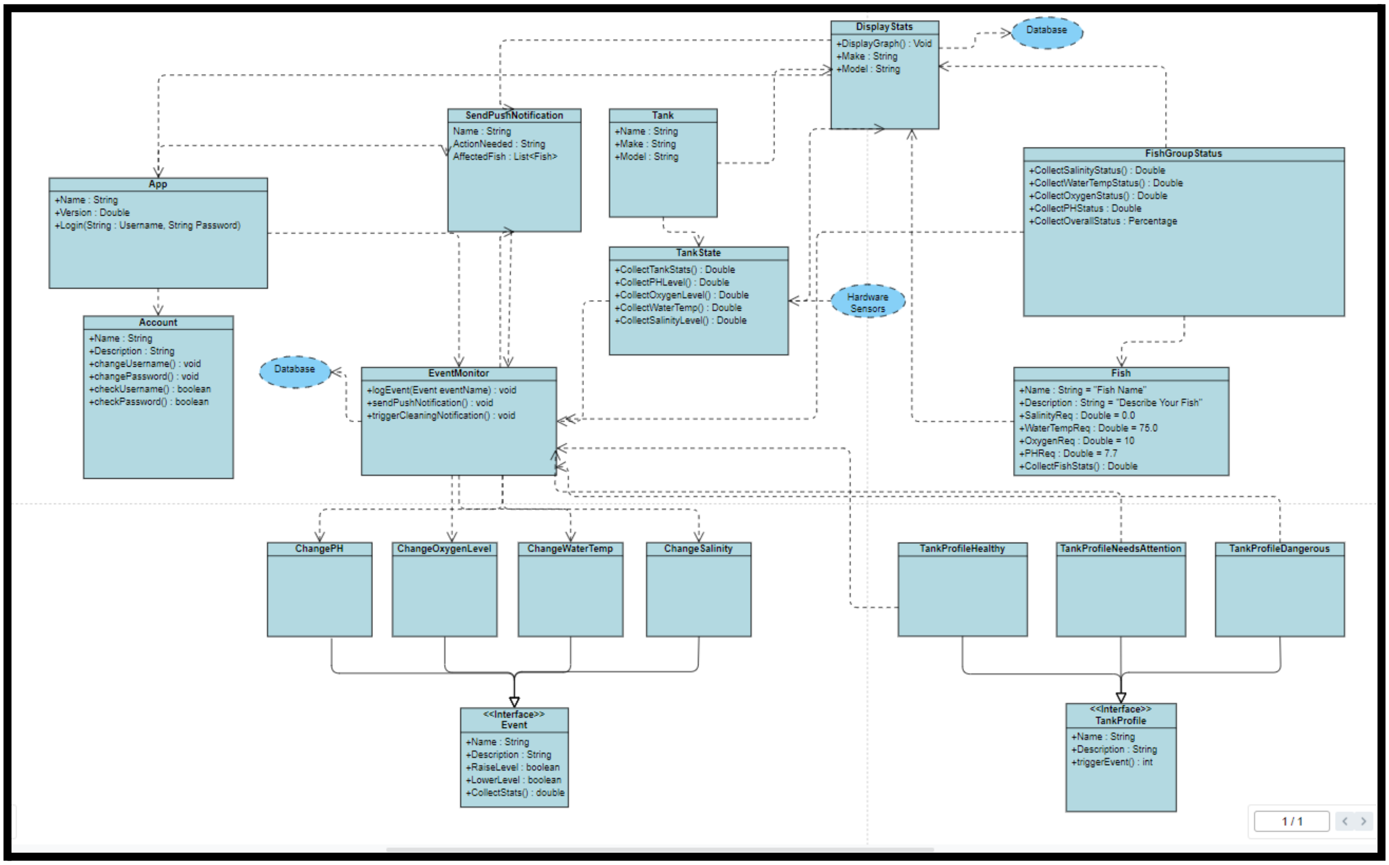


Pictured: Architectural Overview of the Fish Tank Monitoring System

Description of the software architecture diagram

- Our fish tank has sensors that take in information from the fish tank (salinity, temperature, oxygen, and cleanliness). It is then stored in a database so you can access what these conditions were throughout the day. Additionally, the current conditions are put on display on the outside of the fish tank and in the app. The app itself has a login feature where the user information (username, email, phone number, and password) is stored in another database. The fish tank itself will have libraries that the user can access after logging in. The login process consists of a username, phone number, or email address and a password. This essentially creates an account that stores the data and history of a user's fish tank. The database for the library of fish will contain information and the living conditions of different fish so the user can create the optimal living conditions for the fish they want to get as well as what fish go together. From the app, the user can adjust the temperature, salinity, oxygen, and cleanliness manually through the phone. A separate database contains all the methods for those commands. In the settings, the user can adjust the intensity of all the different elements and in what quantities they release.

UML Class Diagram



Description of classes

EventMonitor	The event monitor responds to incoming events and creates events depending on the data from the: - TankState - TankProfile - FishGroupStatus
<<interface>> Event	Event is an interface which is implemented as new variables are added to the tank.
ChangeSalinity	ChangeSalinity implements the Event interface and changes the salt levels within the tank.
ChangeWaterTemp	ChangeWaterTemp implements the Event interface and changes the water temperature levels within the tank.
ChangeOxygenLevel	ChangeOxygenLevels implements the Event interface and changes the oxygen level inside the tank.

ChangePH	ChangePHLevels implements the Event interface and changes the acidity level inside the tank.
CleanTank	Sends notification to the user to clean the tank, cannot be done chemically.
TankState	Tank State represents the current state of the tank based off of the data from the hardware sensors.
FishGroupStatus	FishGroupStatus takes the total status of all the fish based on the biological needs of the fish.
Fish	This class represents a fish and represents the biological needs and description of a particular fish.
<<interface>> Tank Profile	Tank profile is an interface that is implemented by different types of tank profiles.
Tank Profile - Healthy	This class represents the conditions for a healthy tank.
Tank Profile - Needs Attention	This class represents the conditions for an unhealthy tank that requires intervention or assistance.
Tank Profile - Dangerous	This tank represents the conditions for an unhealthy tank in which the inhabitants inside are at extreme risk.
Tank	The tank class represents the physical attributes along with the make and model of the Tank.
Account	The account class represents a user and includes a username and password. Outside classes can use the checkUsername() and the checkPassword() functions for authentication. These functions return a boolean to validate if the user authenticated successfully.

Attributes and Operations

EventMonitor	+NotifyUser - Creates a user push notification +LogEvent - Logs an event attempt in the database
<<interface>> Event	+Name +Description +RaiseLevel +LowerLevel
ChangeSalinity	Implements Event Interface

ChangeWaterTemp	Implements Event Interface
ChangeOxygenLevel	Implements Event Interface
ChangePH	Implements Event Interface
TankState	+CollectTankStats
FishGroupStatus	+CollectSalinityStatus - Check to see whether the salinity of the tank matches overall fish needs. +CollectWaterTempStatus - Check to see whether the water temperature of the tank matches overall fish needs. +CollectOxygenStatus - Check to see whether the oxygen levels of the tank matches overall fish needs. +CollectPHStatus - Check to see whether the PH level of the tank matches overall fish needs. +CollectOverallStatus - Collects all statuses and creates an average for the statuses of all the fish.
Fish	+Name +Description +SalinityReq +WaterTempReq +OxygenReq +PHReq +CollectFishStats()
<<interface>> Tank Profile	+Name +Description +TriggerEvent() - Triggers an event specified in the implementation. Returns a 1 for success, and 0 for failure.
Tank Profile - Healthy	+Implements TankProfile Interface
Tank Profile - Needs Attention	+Implements TankProfile Interface
Tank Profile - Dangerous	+Implements TankProfile Interface
Tank	+Name +Description

Account	+Name
	+Description
	+Username()
	+Password()

	+ChangeUsername()
	+ChangePassword()

Development Plan & Timeline

Name	Role
Jose Garcia	Responsible for the hardware integration with the software system. Will develop APIs to connect with the other backend software used by Alex. Will be responsible for developing unit and integration tests in order to ensure each hardware module operates as intended.
Alex Vo	Responsible for software integration of the system and ensuring proper control logic for the tank monitoring system. Alex will be responsible for creating the overall end to end testing suite along with ensuring the proper alerts and notifications are sent between each class.
Darren Lee	Darren will be responsible for setting up the security and authentication of the system. His responsibilities will include verifying the software integrity of the system such as ensuring that hardware does not fail. In addition, he will ensure that all users are authenticated successfully. Because this system can be accessed remotely through an app we would like to prevent our clients' fish tank system from being accessed.

Development Timeline, 9 Months~

March

- Investigate market to determine best commercial off the shelf hardware options
- Experiment with different hardware to identify the best reliability.
-

May

- Develop software notification system, start development on android and iOS platforms.

July

- Meet with the client to determine to demonstrate working model of tank

August

- Refine the product, start creating unit tests and identify any bugs

October

- Deliver final product to client.

December

- Follow up with the client, and resolve any issues/bugs

OTHER RANDOM INFORMATION BELOW:

System Description

- Brief overview of system
    - Fish tank monitoring system to help owners better manage the wellbeing of their fish.
    - Capabilities
        - Fish Tank Monitor
        - - Push notifications to end users phone
        - - Settings to configure monitoring system (salinity, cleanliness, water temperature, oxygen levels)
        - - Hardware sensors to take in data
        - - Display/graph for the data
        - - Fish tracking, catalog
        - - Fish tank chemical auto balancing

ATM Fish Diagram

Software Architecture Overview

- Architectural diagram of all major components - Jose
- Description of the software architecture diagram - Jose

- UML Class Diagram - Darren
- Description of classes - Darren
- Description of attributes - Darren
- Description of operations - Darren

* descriptions should be detailed and specify datatypes, function interfaces, parameters, etc..

Development plan and timeline

- Partitioning of tasks
- Team member responsibilities

Extra Information:

NOTES:

- These are things we need to do to polish project
- Ensure everything has enough information and is documented
- Ensure UML diagram matches the operations and attributes
- Fill in timeline make it look nicer

FISH TANK MONITOR

System Description: A monitoring system connects between user's mobile devices and the fish tank settings.

- Purpose:
  - Tracking fish's movement
  - Constantly monitors, takes in data, and regulates the tank's temperature, salinity, oxygen levels, cleanliness to pre-set settings
- Consists of:
  - Database for storing users' personal information for security and synchronizing across devices shared same account
  - Database for information: using sensors to collect all data from the fish tank (temperature, salinity, oxygen levels, cleanliness)
  - Hardware that store algorithms, adjust the settings, outputs graph/data information to display and report to users' devices.

Development plan and timeline:

- Partitioning of tasks:
  - **Jose**: Designing the software system's architecture and its components, such as the data acquisition module, data processing module, user interface, and alerting module. Also UML class diagram, and connectivity between the app on users' devices and the hardware on the fish tank. Design the user-friendly interface for the app, and interface for the software system that displays the data in an intuitive manner and allows the user to interact with the software.

- **Alex**: Developing a backend system algorithm for processing the data collected from the sensors such as water temperature, pH levels, and water quality to provide meaningful information that the user can interpret. This can include converting data into charts, graphs, and alerts. Develop a frontend system to display data collected from the backend system in the most intuitive format for users to regulate the system from their reference. Maintain, fix, and update the operating system to the most up-to-date.
- **Darren**: Gather and document requirements for the app, such as what features it should have, what devices it should support, and what data it should display. Testing the software system thoroughly to ensure that it functions as intended and performs reliably. Performing routine maintenance to keep the software system updated and functioning correctly. Deploy the app across platforms and other distribution channels such as websites or custom hardware devices. Deploy marketing strategies to promote the device and the software behind it.

- Timeline:

Task	Timeline
Define the architecture needs and features	1 week
Prepare all needed tools and appropriate technology stack	1 week
Design the software architecture	1 week
Implement the software architecture	4 week
Test the software	2 week

Deploy the software	1 week
Maintenance and updates	On going

- Team member responsibilities:
  - Jose and Darrent: Title page and Software Architecture Overview
  - Alex: System Description and Development Plan and Timeline