1. 1차발표 지적사항

기능이 추가되면(산소량 조절) 좋은 작품이 되리라 보임  
하드웨어 설계를 잘할것, 다른 방식으로 주제를 살리는 것이 좋을 것 같음  
어항관리의 목표를 명확히 할 것

답변: 기능추가부분은 계획한 기능이 많아서 차후에 생각할 예정. 또한 산소량 조절 같은 경우는 이미 시중에 제품이 많기도 하고 구현이 많이 어렵기 때문에 어려울 것으로 보임. 뿐만 아니라 산소량 조절, 여과 등의 기능은 어항의 생태계 상태에 따라 각기 맞는 상황의 제품 혹은 기기를 사용해야 하기 때문에 구현의 양과 난이도가 상당히 높아질 것으로 판단됨.

(ex. 여과기의 예시(측면 여과기))



하드웨어 설계는 어항크기부터 어항구성까지 대략적으로 진행해둔 상황

다른 방식으로 주제를 살리는방향도 고려중. 우선 앱을 통한 어항원격관리라는 큰 틀은 벗어나지 않을 예정

2. 구현현황

1) 라즈베리파이 서버와 아두이노간 통신 완료(node.js 이용)

2) 서버에서 어플로 온도, pH, 스트리밍 데이터 출력 구현(socket.io 이용)

3) 어플 구성 대략 60% 완료 (세부적인 디자인은 추후 계획중). Navigation Drawer를 이용해서 어플의 공간활용을 높였고, 설정부분과 스트리밍부분을 2개의 액티비티로 분할하여 사용자가 어플에 처음 접근했을 때 이용하기 쉽도록 하였음. 또한, 설정액티비티를 프래그먼트(먹이급여, 온도/pH설정)를 이용하여 구성해 액티비티 이동을 최소화 하였음

4) 웹서버에서 버튼을 누를 시 아두이노가 동작하는 것 구현

3. 문제점

1) 아두이노에서 서버로 온도, 시간값, pH값들이 계속해서 들어오는 동시에 워터펌프나 서보모터가 동작해야 하는데, 동시에 할 경우 센서값들이 늦게 들어오는 문제가 있어서 해결하려고 멀티쓰레드방식을 이용하려 함 -> 그러나, 아두이노가 멀티쓰레드를 지원하지 않기 때문에 유저들이 만든 쓰레드 라이브러리를 이용해야 함 -> 이 방법을 적용하였으나 원활하게 동작하지 않아서 방법을 아예 바꾸게 됨

-> 센서(온도,pH)값은 라즈베리파이로 옮기고, 워터펌프나 서보모터같이 동작이 필요한 모듈은 아두이노에 연결하여 해결함. 라즈베리파이에 연결한 온도와 pH센서중에는 온도 센서는 구현하였고, pH센서는 아직 구현중.

2) 웹서버에서 버튼 누를시 아두이노가 동작하지만, 어플에서 버튼 누르면 동작하지 않는 문제가 발견됨 -> 예를들어 웹서버의 웹 페이지에서 아두이노의 서보모터를 동작시키는 버튼을 누르면 클릭한 버튼의 id값을 GET방식으로 넘겨주어서 이 값을 받은 페이지에서 그 id값을 아두이노로 보내면 아두이노가 신호를 받아서 동작한다.

그러나 어플에서 버튼을 클릭하면 어플에서 socket.io로 클릭한 버튼에 대한 값을 웹서버로 넘겨주면 웹서버에서 어플이 보낸 데이터값을 받는다. 이때 받은 데이터를 아두이노로 보내는데, 웹서버의 버튼클릭과 같은 방식으로 전송하지만 동작하지 않는 점에 대해서 아직 해결중.

4. 2월 계획

서버와 관련된 모든 기능 구현을 최우선으로 하고, 시간이 남으면 어플 디자인을 할 예정