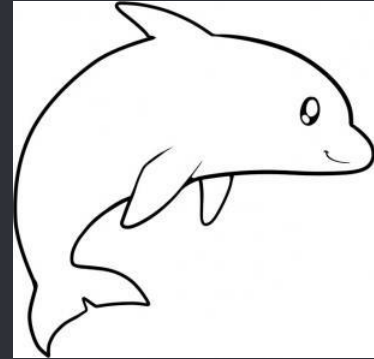


The Dolphin Design

Group 18:
Kandyce Burks
Dieu Do
Luis Hernandez
Brent Yurek



Design Goals

- Sonar Mapping as fast as possible
- Vital information calculations as fast as possible
- Battery lasts longer than average dive
- No bugs!!! (unrealistic reach goal)
- Simultaneously take in input during calculations

Software Architecture

Modifications:

- HUD incorporated
- HUD connected to Tank
- Add rotational sonar on the head that connects to the microprocessor



Modifications:

- Add rotational sonar on the back of tank that connects to microprocessor
- Pressure sensor on the tank's valve that connects to the microprocessor

Modifications:

- Add pressure sensor clipped on the wetsuit that connects to the microprocessor
- Add gyroscope clipped on the wetsuit that connects to the microprocessor

Software Architecture Cont.

Multiple classes all interacting and communicating to provide a seamless workflow!

- HUD
- SonarMapping
- SonarSignaling
- Distance
- Clock
- DepthLevel
- Alerts
- Brightness
- Languages
- Pressure
- Units
- AirTime
- Speed

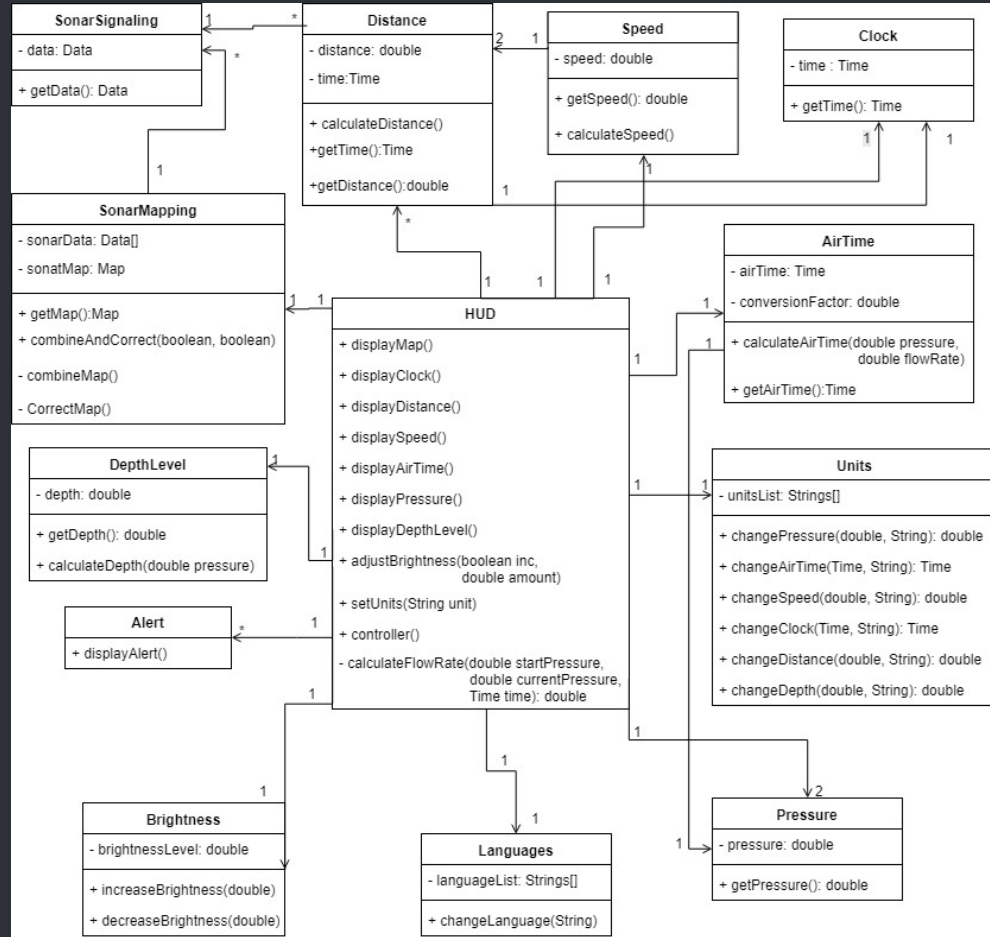



Diagram :



Subsystem

Application

- Core functionality of the system

User Interface

- Interface that users sees and interact with via HUD display

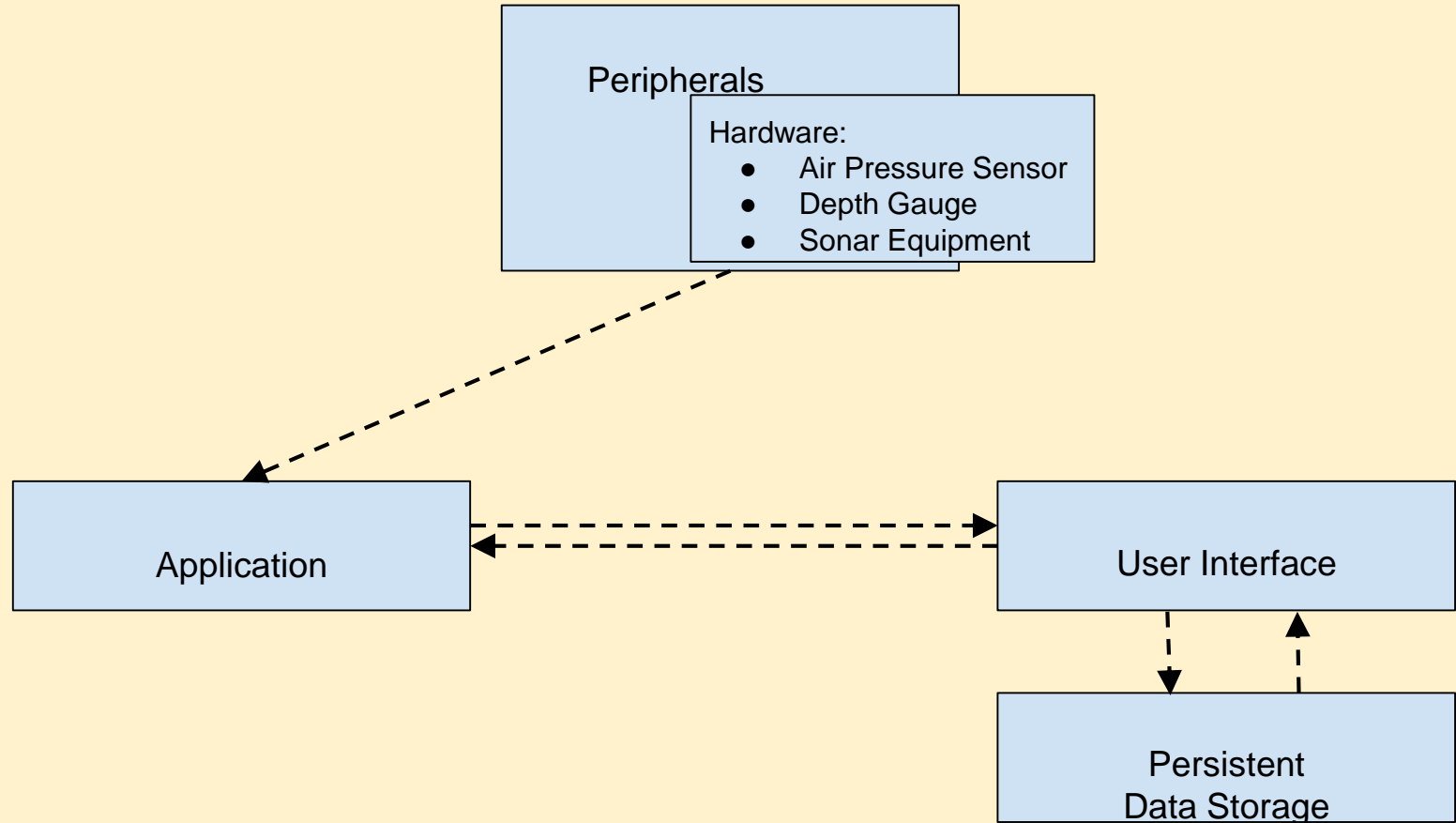
Persistent Data Storage

- Provides data storage

Peripherals

- Takes in data to provide to Application

Hardware and Software Mapping



Subsystem Services: Application

- Input from peripherals

- `getPressure()`
 - `getDepth()`
 - `getData()`
 - `getAirTime()`
-

- Computations

- `calculateAirTime()`
 - `getMap()`
 - `combineMap()`
 - `correctMap()`
 - `calculateDistance()`
 - `calculateSpeed()`
 - `calculateAirTime()`
-

- Sends data to UI for display

Subsystem Services: Peripherals

- All the hardware in the system
- Takes in data and commands to send to Application for computation

❖ Air pressure sensor

❖ Depth Gauge

❖ Sonar Emitter and Receiver

❖ Buttons

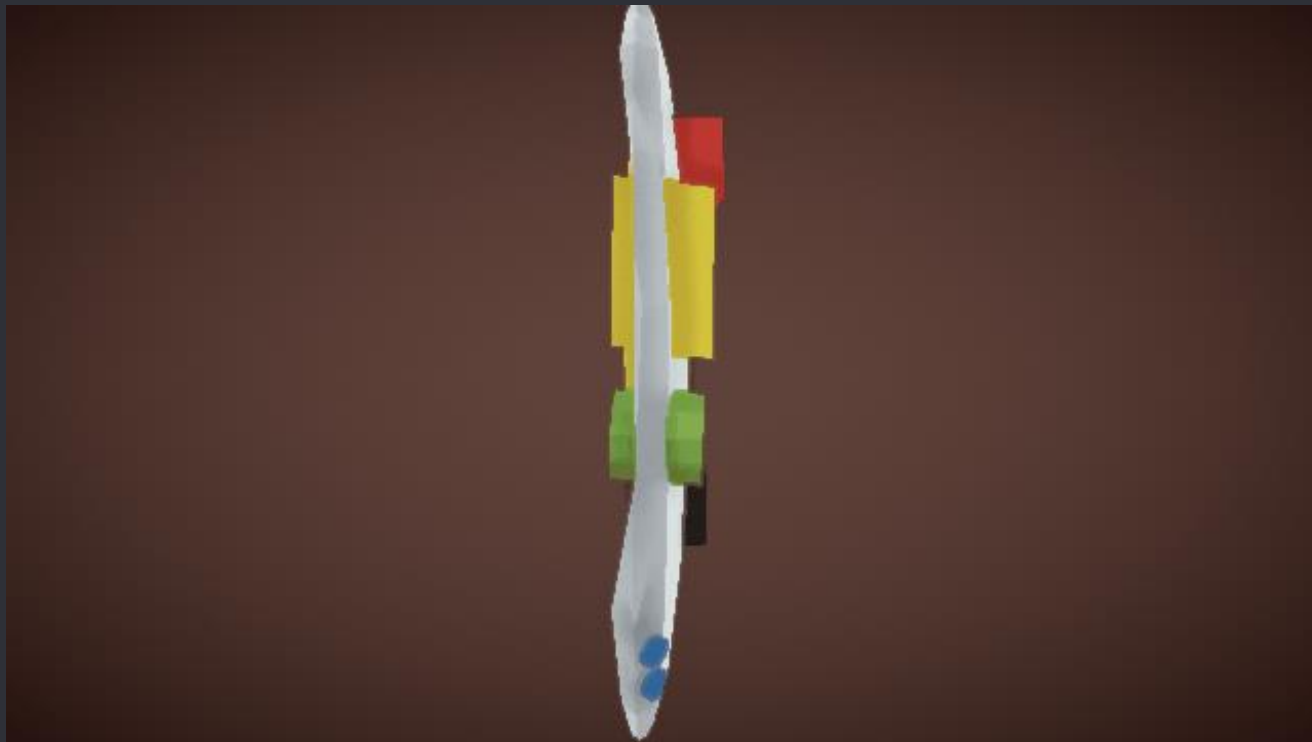
● Subsystem Services: Persistent Data Management

- Flat files
- No database
- Used for persisted settings
 - Brightness
 - Units
 - Language



User Interface Design

- All information will be display on the sides
- Leaves center open for user to see while diving



- ❖ Face
- ❖ Alerts
- ❖ Dive information
- ❖ Sonar Mapping
- ❖ Buttons

Subsystem Services: User Interface

Displays information to user

- displayMap()
 - displayClock()
 - displayDistance()
 - displaySpeed()
 - displayAirTime()
 - displayPressure()
 - displayDepthLevel()
 - displayAlerts()
-


Customizable

- Units
 - Brightness
 - Language
-

Users command to Application

- changeLanguage()
- increaseBrightness()
- decreaseBrightness()
- setUnits()

Access control

Object Actors	Parts	Functionality
Diver	Buttons	adjustBrightness() setUnits() changeLanguage()
Air tank sensor	Air Tank Pressure sensor	calculateAirTime() getAirTime() unitsList[]
Depth Gauge	Depth Gauge	getDepth() unitsList[]
Sonar	Frequency emitters	getData(): Data sonarData: Data[] getMap() combineMap() correctMap() 

Boundary Conditions

- Initialization
 - Defaults: English, Metric units, 50% brightness
 - Adjustable once turned on
 - Usable out of the box!

- Normal Start-up
 - Retains persisted setting from last shut off

- Shutdown
 - Saves current settings for next usage



○ Thanks for listening, here's a dolphin!