EE/CprE/SE 492 WEEKLY REPORT 4

3-1 - 3/15

Group number: 11

Project title: Smart Water Leak Shutoff Valve

Client &/Advisor: Cheng Huang

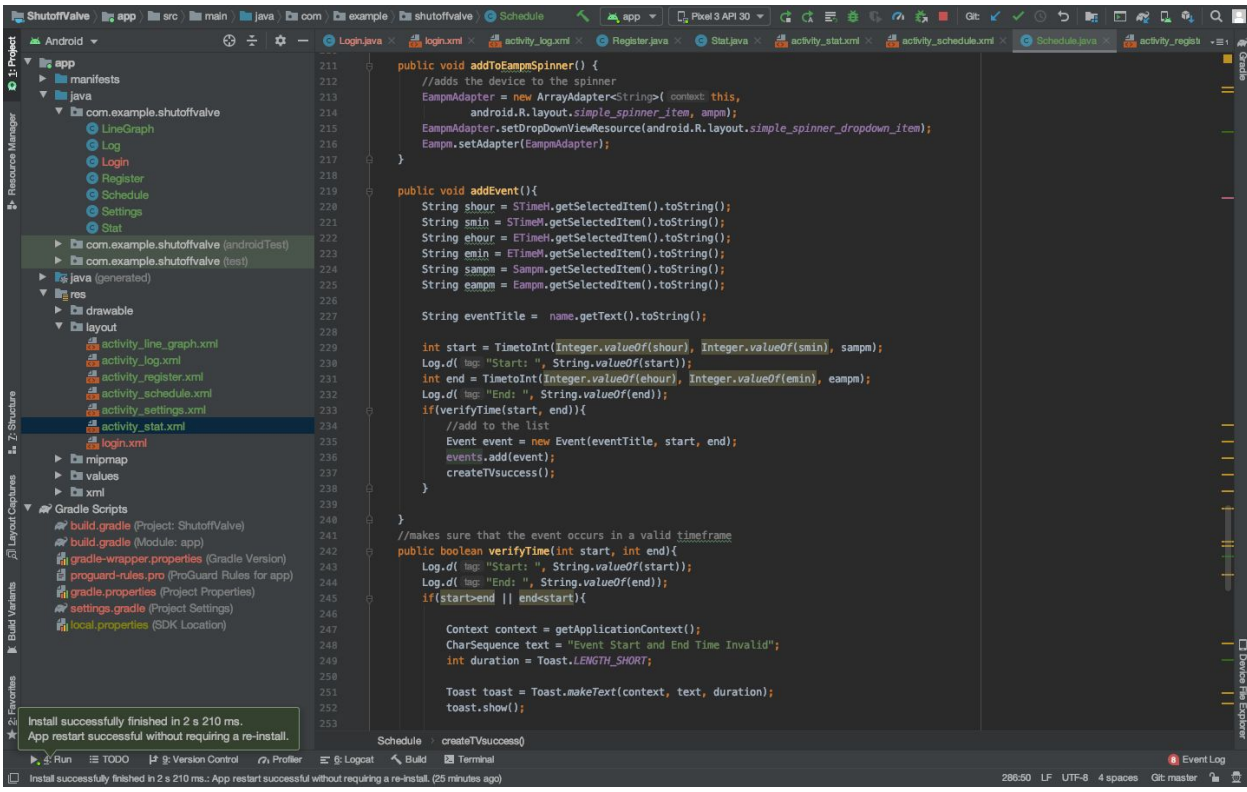
Team Members/Role: Matthew Brandt, Curt Kissel, Cody Juracek, Wolfgang Morton, Grace Wilkins

- Weekly Summary – A new screen has been created within the Android application that allows the user to create a new event. Each event will specify a time in which the user does not expect water usage (for example, when they are asleep or at work). Created text alerts for when the device is automatically turned off. Configured the back end to run with https to encrypt communication for security purposes. Planned a scheduler for the device to use to automatically monitor water usage. For hardware, we asked ourselves a what-if scenario when either the internet or power goes down while the shutoff valve is in operation. We started looking into a secondary power source lasting for at least four hours; afterall, the consumer would still have running water. The goal is to have the device operate and be controlled manually for either scenario.
- Past Week Accomplishments
    - Matthew – Modified the back end application to work with https. Added the ability to send the user text messages when the device shuts off the waterflow. Created entity for scheduling times for device to work automatically.
    - Curt – Began creation of the event scheduler. Allows user to specify a period of time in which there should not be water usage.
    - Cody – Analyzed and mapped the circuit, its electrical properties. Determined what is needed in a secondary power source, whether it is consumable or rechargeable.
    - Wolfgang – Helped in determining the secondary power. Looked into solutions to prevent power surging, and considered different options for fused and breaker protections for the device.
    - Grace – Continued to work with the wifi module. Looked into different options for fuse or breaker as protection.
- Pending issues
    - Still don’t have communication between wifi module and application.
    - Adding a secondary power source will greatly increase the cost of the product by $40-70. Something to consider, originally we had a target cost of $250 - we pushed this cost down to $150 after we initially started building the circuit. The product should not be more than $200.
    - Size of memory on arduino may not be big enough

- Individual Contributions

Name	Contributions	Hours this Week	Cumulative Hours
Matthew	Text alerts for user and entity to schedule events	10	36
Curt	New app screen that allows user to schedule events	11	40
Cody	Investigated the use of a rechargeable battery	16	40
Wolfgang	Considered different options to better protect the device from power issues	13	37
Grace	Researched effect ways to implement the wifi module; debugged potential codes for wifi module and components	10	36

- Plans for upcoming week
    - Matthew – Work on code for arduino to make calls to back end application. See if communication can be established through https. Work on storing schedule data on arduino so that the device can run automatically without wifi connection.
    - Curt – Finish creation of the schedule screen. Attempt to get functionality while app is closed. Attempt to send notifications to the user while the app is closed.
    - Cody – We are going with a different WiFi module, because it will allow for easier communication between the Arduino and router. Have a secondary power source chosen and implemented, all is left is determining the A-hr being consumed by the circuit. The circuit requires 1Amp, it's powered by 12Volts. We want 6 hours of battery life. Primary goal: have the device connected to the backend.
    - Wolfgang – Finalize what component will be used to help mitigate damage due to the device from power issues (i.e. fuse or breaker). Finalize the secondary powersource to be used in emergency power failure. Possibly start permanent circuit assembly for the device.
    - Grace – Connect the solenoid to the wifi module to a web interface.



- Snippet of code from the Scheduler in Android Studio for the application

Code to enable https

```
@EnableWebSecurity
public class SecurityConfig extends WebSecurityConfigurerAdapter {

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http.authorizeRequests()
                .antMatchers("/**")
                .permitAll();
        http.cors().and().csrf().disable();
    }

    @Bean
    CorsConfigurationSource corsConfigurationSource() {
        CorsConfiguration configuration = new CorsConfiguration();
        configuration.setAllowedOrigins(Arrays.asList("*"));
        configuration.setAllowedMethods(Arrays.asList("*"));
        configuration.setAllowedHeaders(Arrays.asList("*"));
        configuration.setAllowCredentials(true);
        UrlBasedCorsConfigurationSource source = new UrlBasedCorsConfigurationSource();
        source.registerCorsConfiguration("/**", configuration);
        return source;
    }
}
```

Code to send text alerts. Currently hard coded to send texts to one phone number

```
    if(Device.getMode() == "off")
    {
        Twilio.init("AC86de0b1b65a269603a15f5be91af2c90", "b5780596040b799cd6e2617c85f83e03");
         Message message = Message.creator(
         new PhoneNumber("+17082966650"),
         new PhoneNumber("+19852564495"),
         "Your Device: '" + Device.getName() + "', has been turned off")
        .create();
    }
}
```