```cpp
1  #ifndef CONSTANTS_CLASS
2  #define CONSTANTS_CLASS
3
4  #include <string>
5  #include <IPAddress.h>
6
7
8   //@brief Contains all Constants for the firmware. Change them in       ⮑
       constants.cpp
9  class constants
10 {
11     public:
12     static const char version[];
13
14     //@brief All custom Pins
15     class pins
16     {
17         public:
18
19         class led
20         {
21             public:
22             static uint16_t Interval;
23             static uint16_t BottomLEDsIntervalOff;
24             static uint16_t BottomLEDsIntervalOn;
25             static uint16_t BottomLEDsIntervalBetween;
26             static uint8_t Top;
27             static uint8_t Led1;
28             static uint8_t Led2;
29         };
30
31         class motor
32         {
33             public:
34             static uint8_t FrontLeft_Dir;
35             static uint8_t FrontLeft_Speed;
36             static uint8_t FrontRight_Dir;
37             static uint8_t FrontRight_Speed;
38             static uint8_t BackLeft_Dir;
39             static uint8_t BackLeft_Speed;
40             static uint8_t BackRight_Dir;
41             static uint8_t BackRight_Speed;
42
43             static uint8_t ServoLowerPin;
44             static uint8_t ServoUpperPin;
45             static uint8_t ServoExtenderPin;
46
47             static uint8_t ServoExtenderLimitSwitchUpper_Pin;
48             static uint8_t ServoExtenderLimitSwitchLower_Pin;
49         };
50
51         class buzzer
52         {
```

```cpp
53                    public:
54                    static uint8_t buzzer_pin;
55                    static uint8_t pwm_channel;
56              };
57          class uss
58          {
59                  public:
60                  static uint8_t Trigger;
61                  static uint8_t Front_Sens;
62                  static uint8_t Back_Sens;
63                  static uint8_t Left_Sens;
64                  static uint8_t Right_Sens;
65              };
66      };

67

68      class uss
69      {
70          public:
71          static uint8_t FrontBack_Min_Distance;
72          static uint8_t Side_Min_Distance;
73      };

74

75      class sdcardIO
76      {
77          public:
78          static const char ConfigFilePath[];
79      };

80

81      class controller_vars
82      {
83          public:
84          static std::string BT_ADDR;
85      };

86

87      static bool setValue(std::string, String);
88  };

89

90  #endif
```