* Main
  + Módulo principal
* Password (Matrix, Bluetooth)
  + Módulo que evalúa si la contraseña es correcta en función de los datos obtenidos por el teclado matricial, el cual necesita siete pines GPIO, o en el módulo bluetooth, el cual necesita el protocolo de comunicación UART.
* Indicators (LEDs, Buzzer)
  + Módulo de control sobre los LEDs verde y rojo que indican si la contraseña es correcta o incorrecta respectivamente, y del buzzer, el cual reproduce un tono agudo o grave a la vez de los leds en la situación respectiva antes mencionada.
* Control (Motor control, Relay solenoid, Relay lightbulb)
  + Módulo de control de los relevadores del solenoide de la cerradura y de la bombilla de luz, así como del circuito de control del motor.
* GPIO
  + Enums
    - Ports
      * PortA
      * PortB
      * PortC
      * PortD
      * PortE
    - Pins
      * Pin0
      * Pin1
      * Pin2
      * …
      * Pin31
    - IO
      * INPUT
      * OUTPUT
  + Funciones
    - void GPIO\_vfnPortInit(uint8\_t port, uint8\_t pin, uint8\_t IO);
      * port: Recibe una letra para indicar el puerto a habilitar
      * pin: Recibe un número el cual indica el pin a declarar como GPIO
      * IO: Recibe un INPUT (0) para declarar el pin como input, o un OUTPUT (1) para declararlo como output.
    - uint8\_t GPIO\_bfnSetData(uint8\_t port, uint8\_t pin);
      * Descripción
        + Escribir un 1 en el pin seleccionado. Si sí lo pudo escribir, regresa un 1, si no, regresa un 0.
      * Parámetros
        + port: Letra del puerto en el que se escribirá
        + pin: Numero del pin a escribir
    - uint8\_t GPIO\_bfnReadData(uint8\_t port, uint8\_t pin, uint8\_t \*value);
      * Parámetros
        + port: Recibe una letra para indicar el puerto a habilitar
        + pin: Recibe un número del pin a leer el dato
        + value: Apuntador a variable donde se almacenará el dato leído