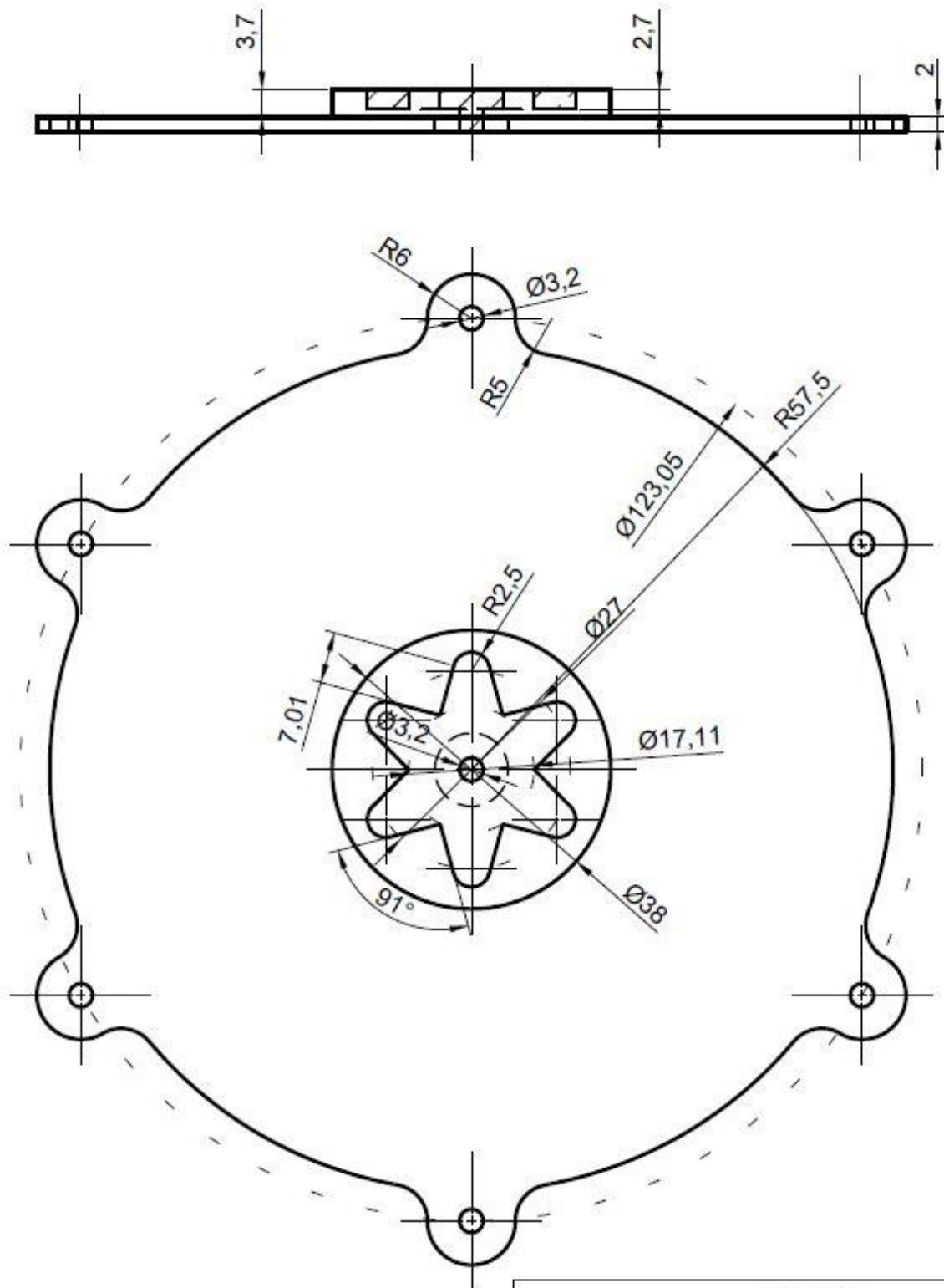
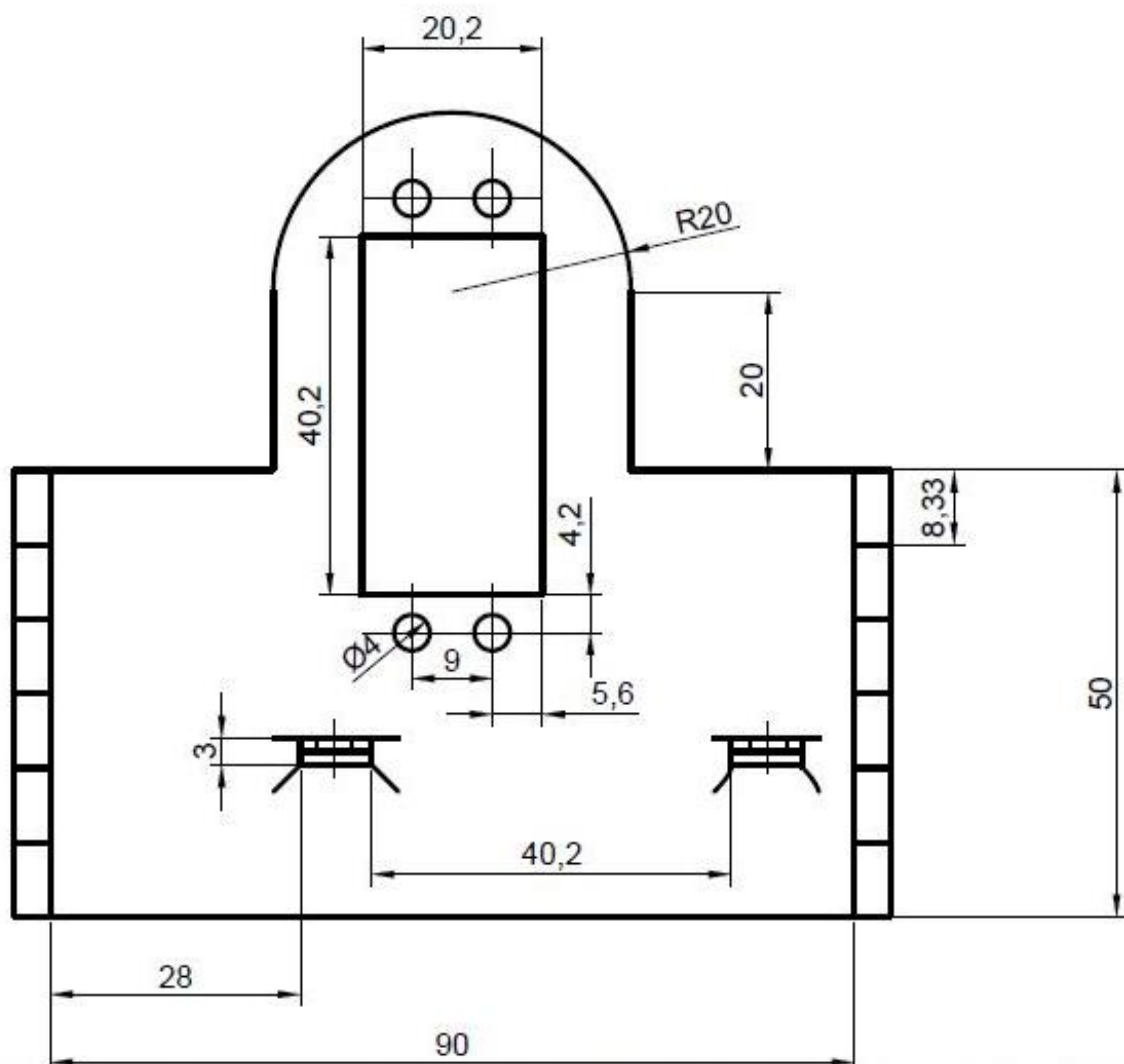
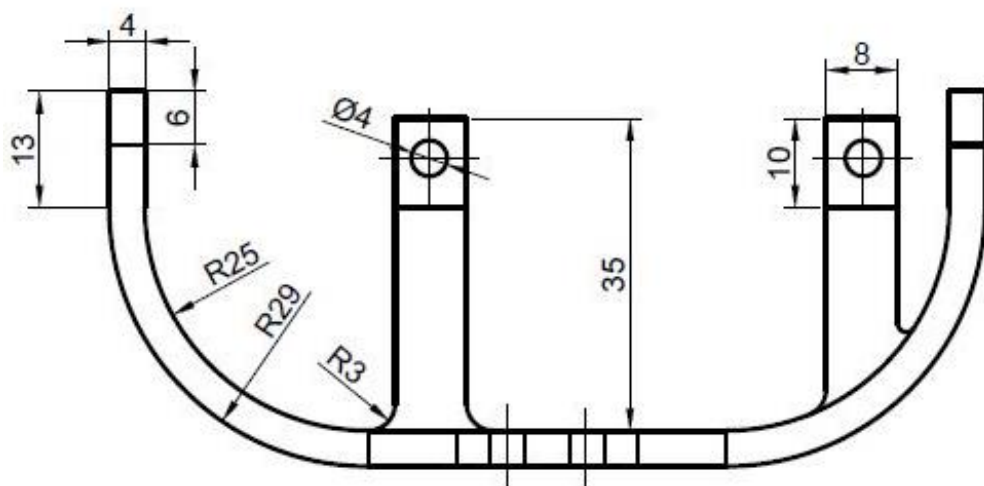


## **Annex A. Plànols de les parts del braç robòtic**

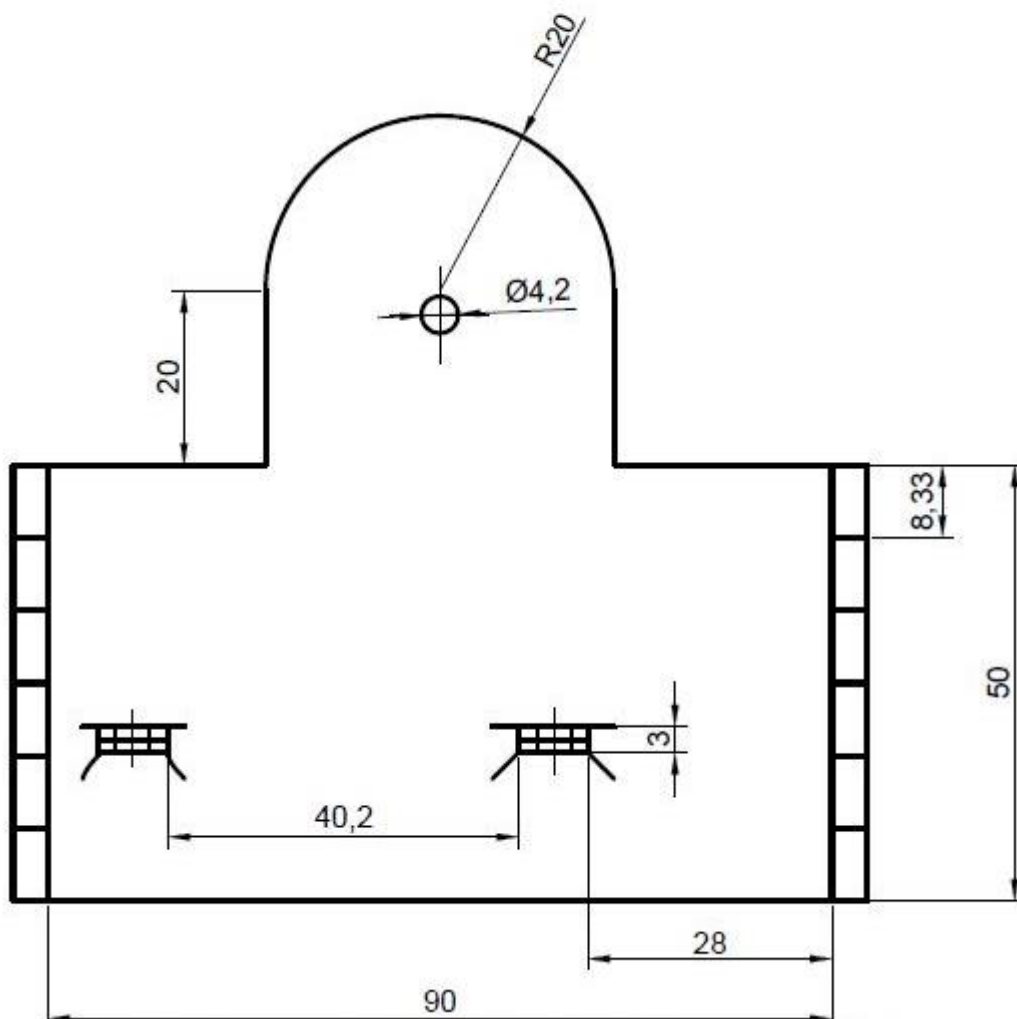
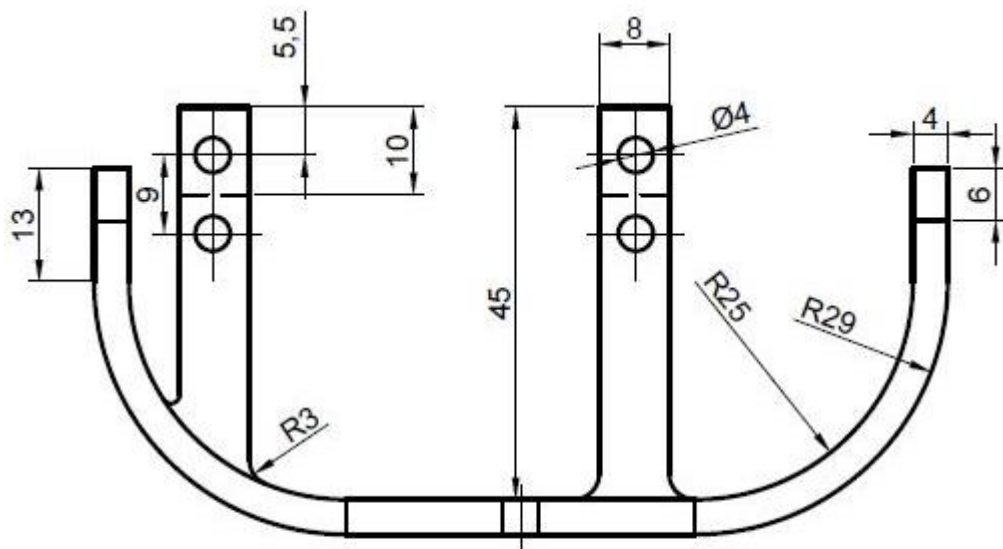
A continuació, s'adjunten els plànols dels components que formen el braç robòtic i que van ser fabricats mitjançant impressió 3D:



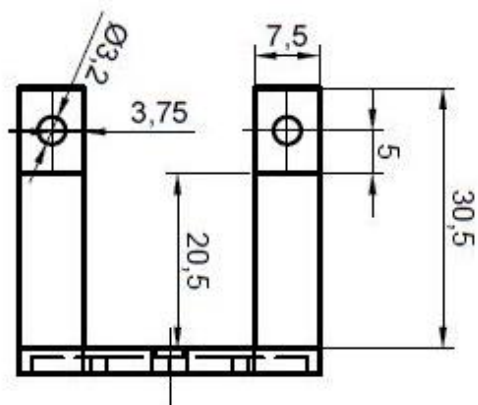
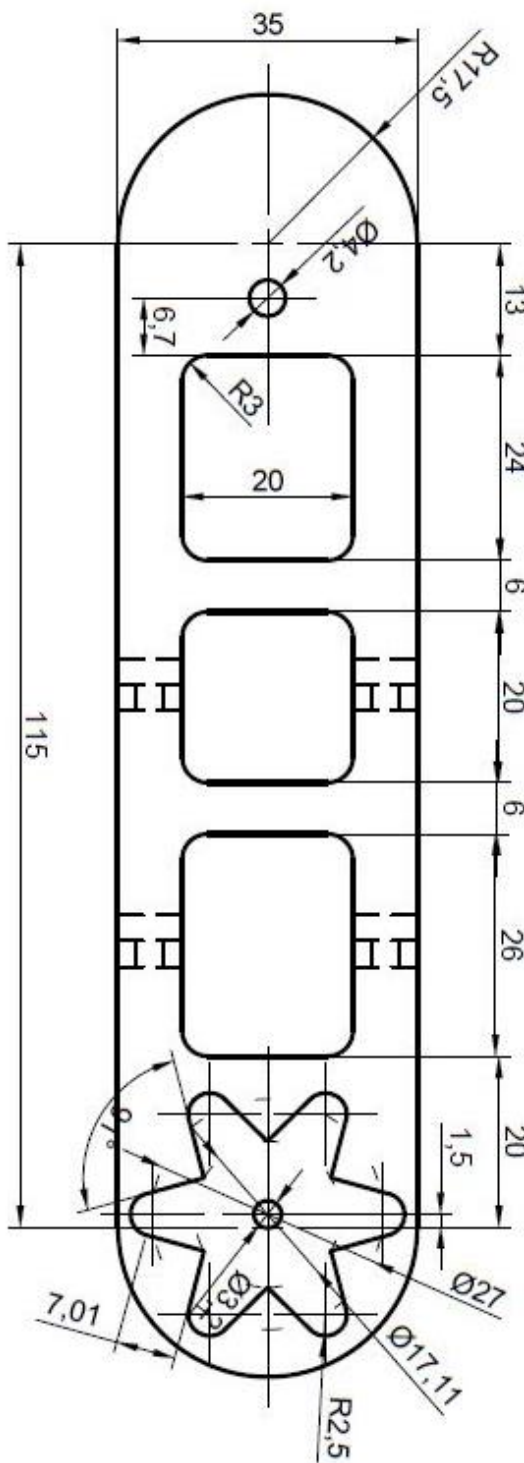
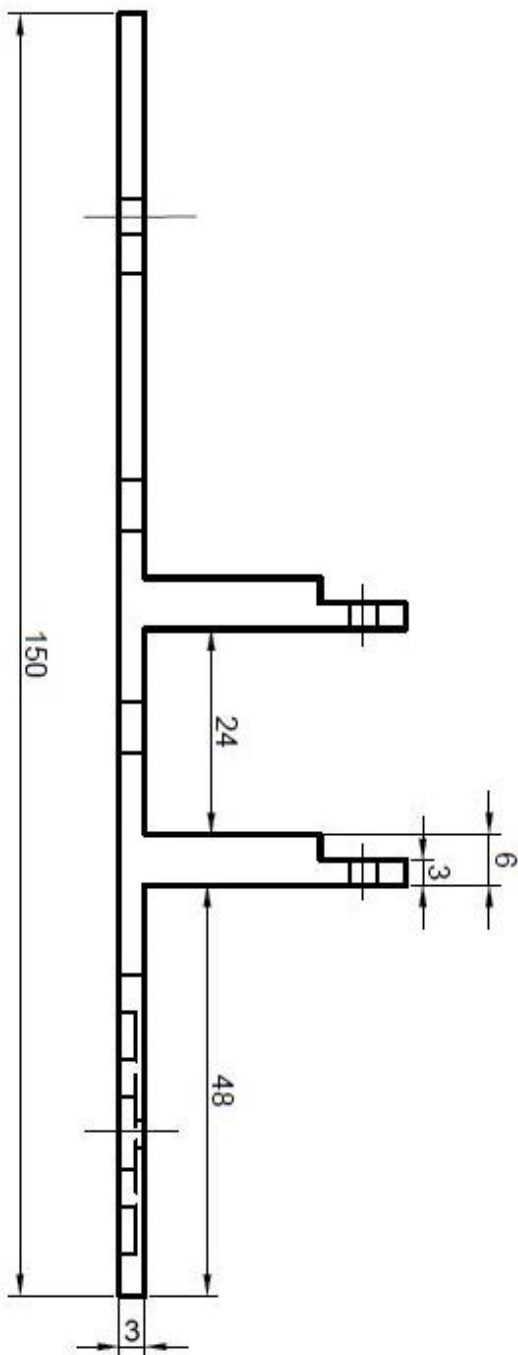
<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Base		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	1



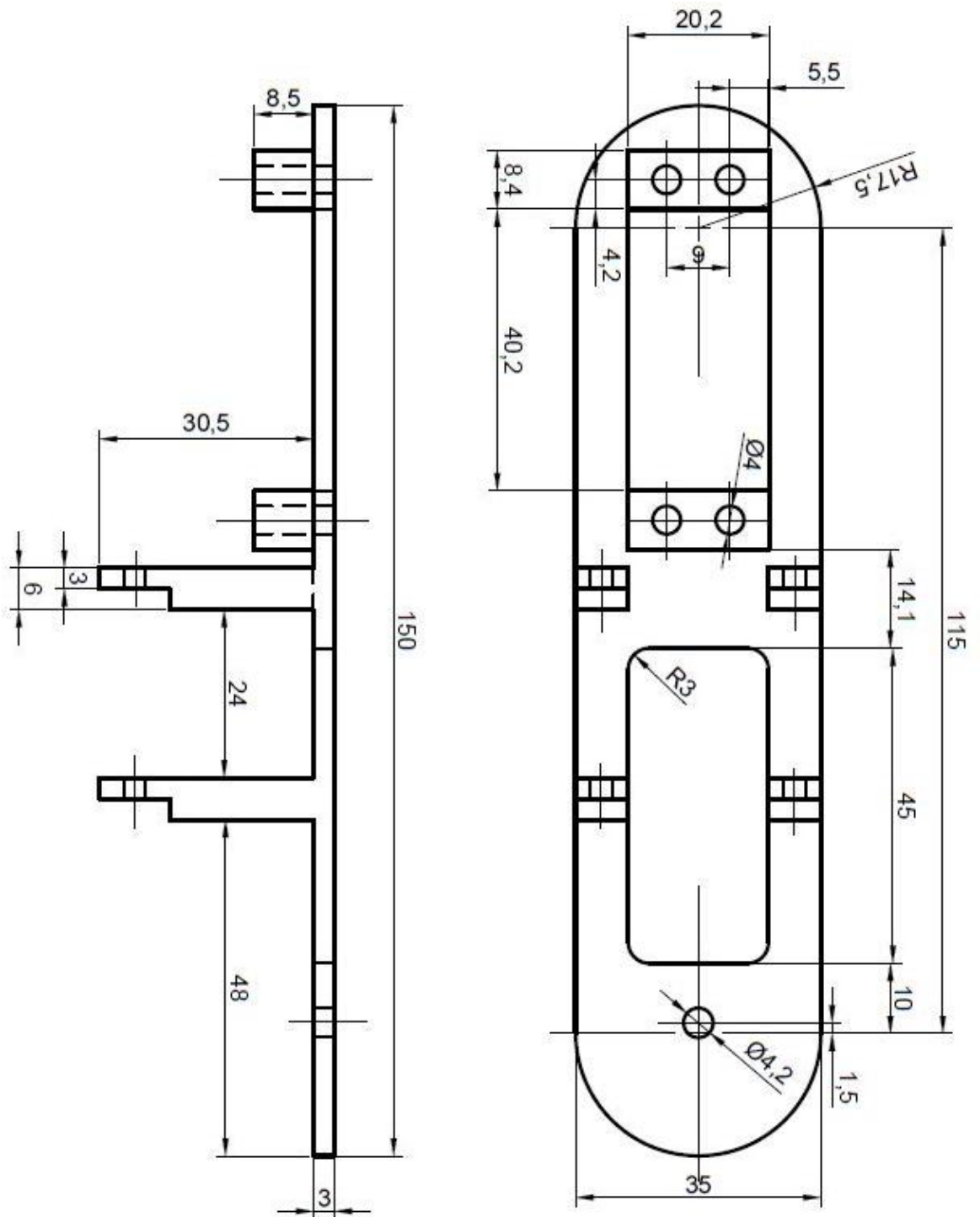
<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Primera meitat del tronc		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	2



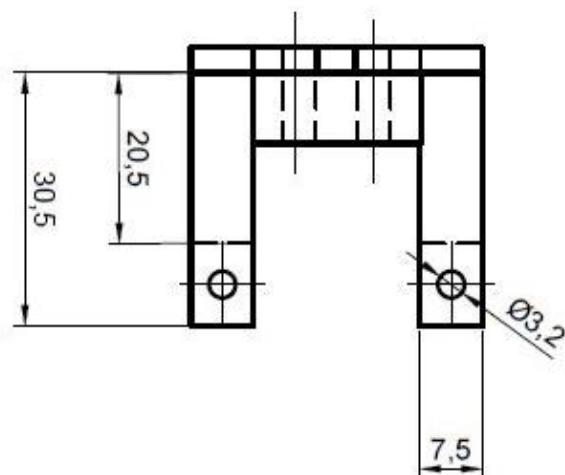
<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Segona meitat del tronc		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	3

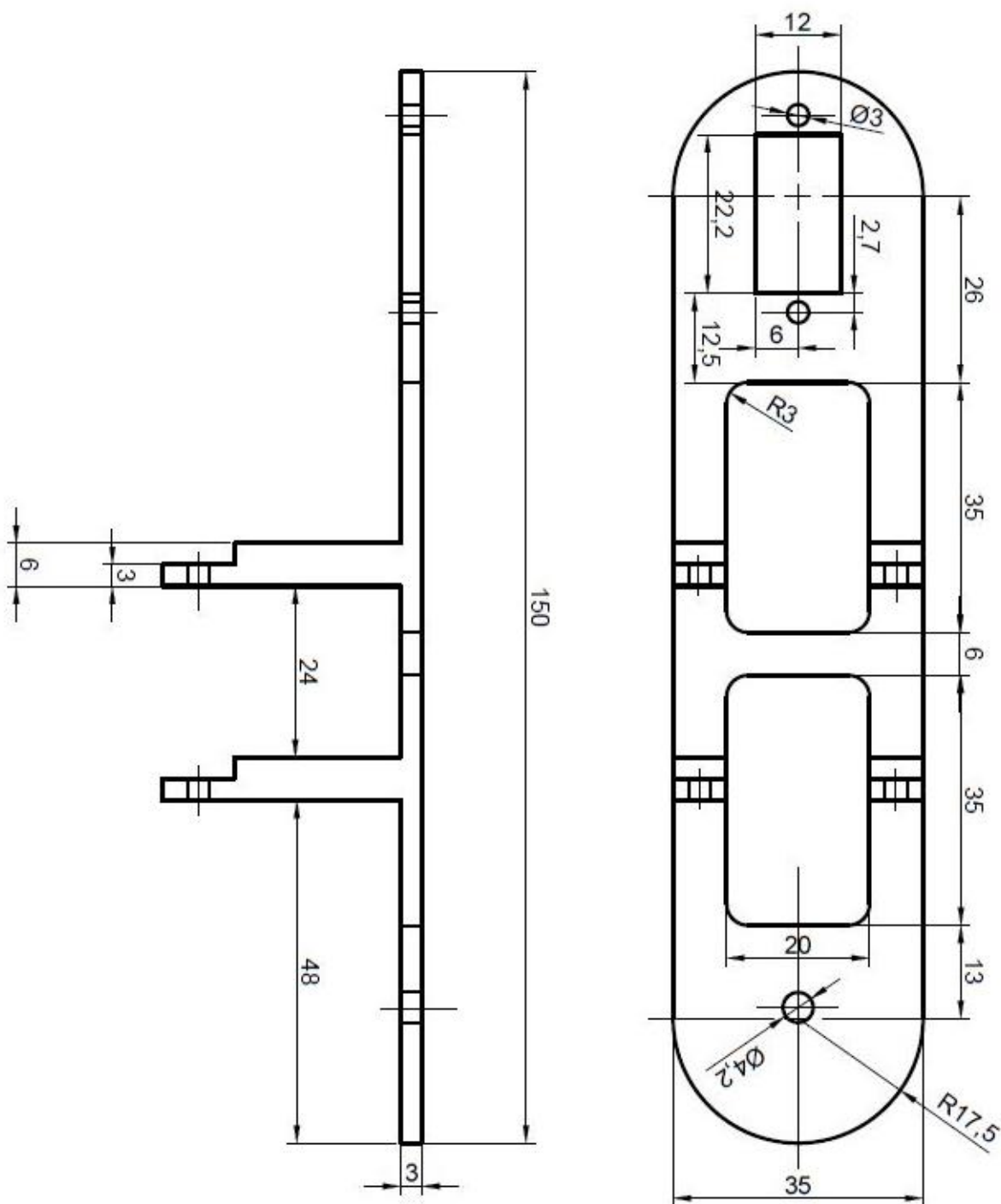


<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Baula inferior esquerra		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	4

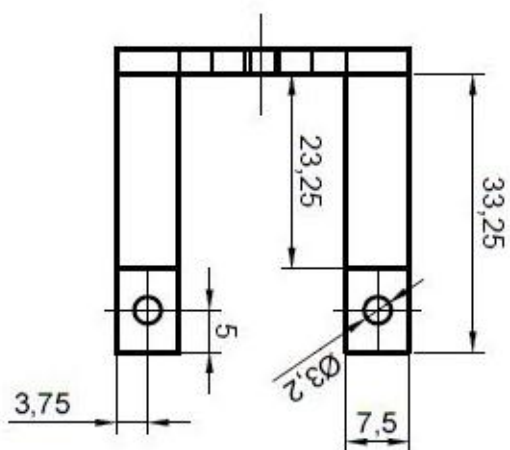


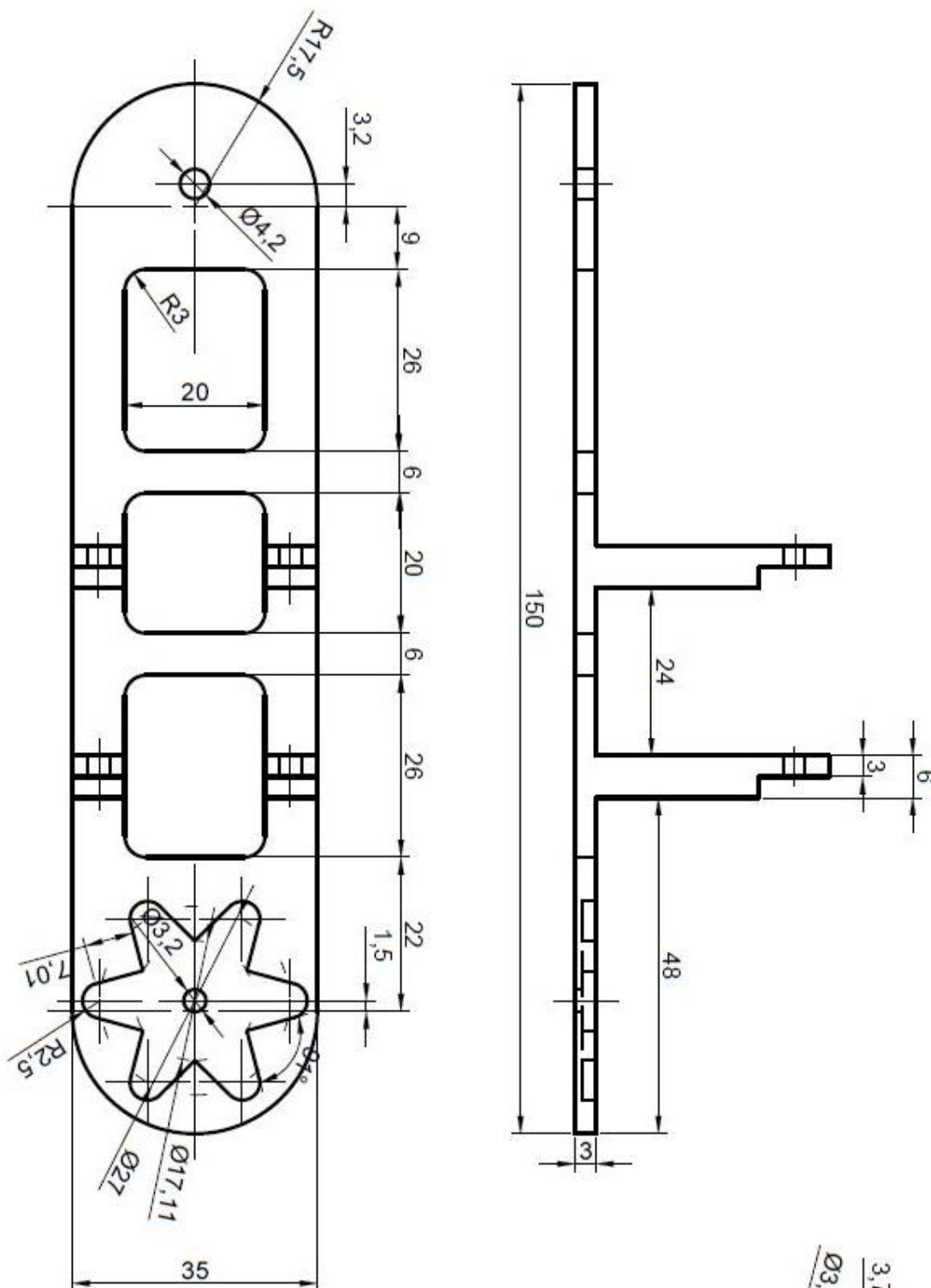
<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Baula inferior dreta		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	5



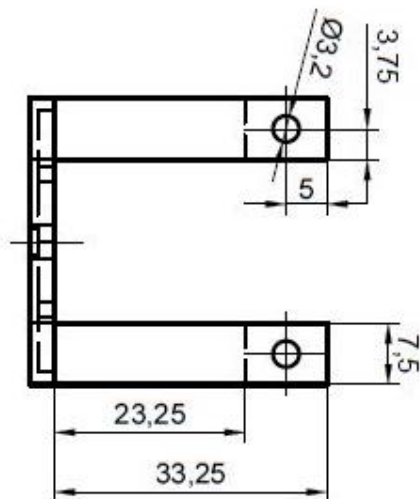


<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Baula superior esquerra		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	6

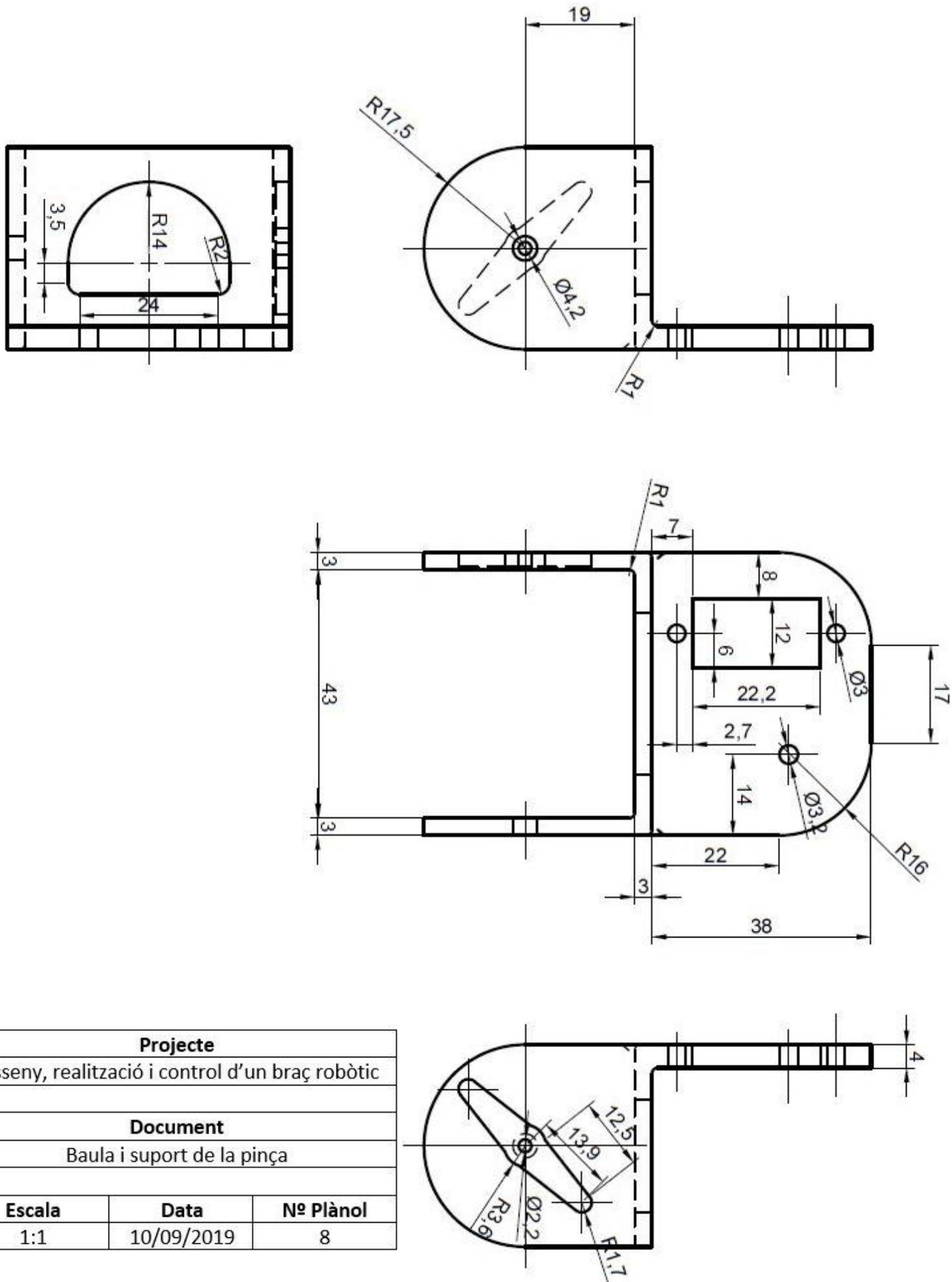




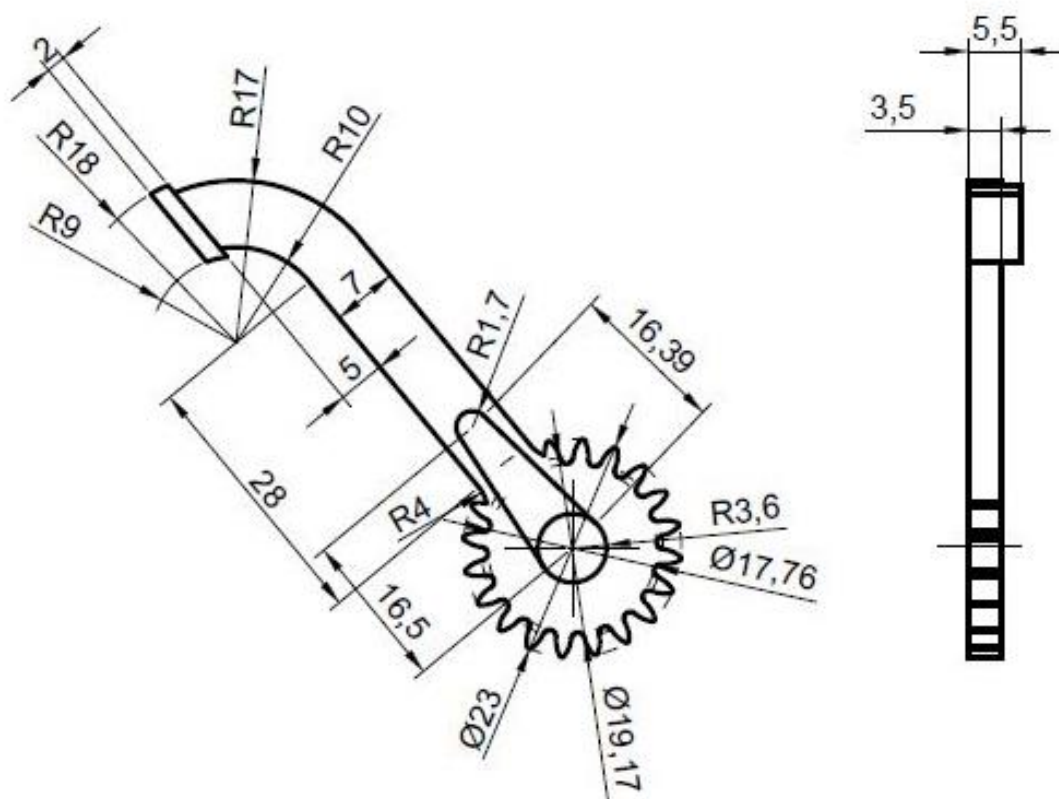
<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Baula superior dreta		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	7



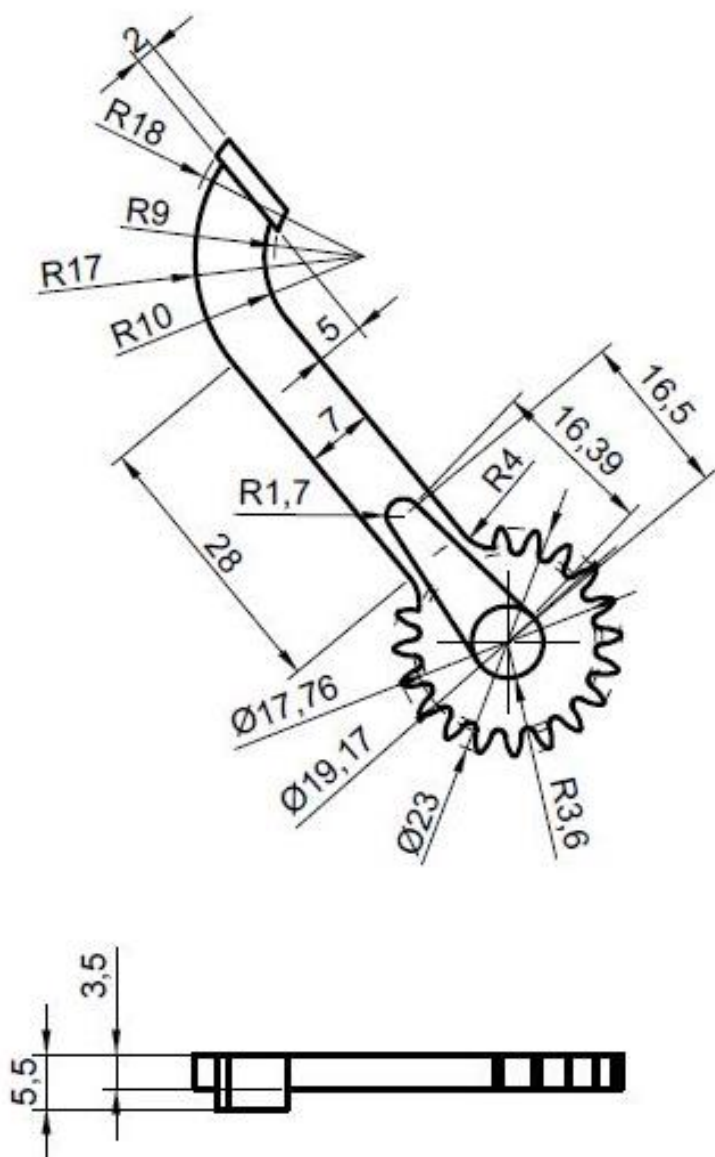




<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Baula i suport de la pinça		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	8



<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Meitat esquerra de la pinça		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	9



<b>Projecte</b>		
Disseny, realització i control d'un braç robòtic		
<b>Document</b>		
Meitat dreta de la pinça		
<b>Escala</b>	<b>Data</b>	<b>Nº Plànol</b>
1:1	10/09/2019	10

## Annex B. Programa de control a través d'una taula d'angles

```
#include <WiFiNINA.h>
#include <Servo.h>

Servo servoTronc;    // Crea l'objecte servoTronc per controlar el servo del tronc.
Servo servoEspatlla; // Crea l'objecte servoEspatlla per controlar el servo de l'espatlla.
Servo servoColze;    // Crea l'objecte servoColze per controlar el servo del colze.
Servo servoCanell;   // Crea l'objecte servoCanell per controlar el servo del canell.
Servo servoPinza;    // Crea l'objecte servoPinza per controlar el servo de la pinça.

// Crea les variables on guardar la posició dels servos:

int angleTronc = 90;
int angleEspatlla = 90;
int angleColze = 90;
int angleCanell = 90;
int anglePinza = 90;

// Crea les constants on guardar els números dels pins digital:

const int pinTronc = 3;
const int pinEspatlla = 5;
const int pinColze = 6;
const int pinCanell = 9;
const int pinPinza = 10;

//Dades de la xarxa Wi-Fi:

char ssid[] = "XXX";    // Nom de la xarxa SSID
char pass[] = "YYY";    // Contrasenya de xarxa

int status = WL_IDLE_STATUS;
WiFiServer server(80);

void setup() {

    // Configura els pins digitals com a outputs:

    pinMode(pinTronc, OUTPUT);
    pinMode(pinEspatlla, OUTPUT);
    pinMode(pinColze, OUTPUT);
    pinMode(pinCanell, OUTPUT);
```

```
pinMode(pinPinza, OUTPUT);

// Assigna els objectes servos als pins digitals corresponents:

servoTronc.attach(pinTronc);
servoEspatlla.attach(pinEspatlla);
servoColze.attach(pinColze);
servoCanell.attach(pinCanell);
servoPinza.attach(pinPinza);

// Posiciona els servos en la seva posició inicial:

servoTronc.write(angleTronc);
servoEspatlla.write(angleEspatlla);
servoColze.write(angleColze);
servoCanell.write(angleCanell);
servoPinza.write(anglePinza);

Serial.begin(9600); // Inicialitza la comunicació amb el Monitor Serie.

// Inicia el mòdul Wi-Fi:

// Si la comunicació falla, no es continua:

if (WiFi.status() == WL_NO_MODULE) {
  Serial.println("La comunicació amb el mòdul Wi-Fi ha fallat!");
  while (true);
}

// Intenta connectar amb la xarxa Wi-Fi:

while (status != WL_CONNECTED) {
  Serial.print("Intentant connectar amb la xarxa Wi-Fi anomenada: ");
  Serial.println(ssid); // Mostra el nom de la xarxa Wi-Fi.

  // Connectat a la xarxa WPA/WPA2 (canviar aquesta línia si s'utilitza
  // una xarxa oberta o WEP):
  status = WiFi.begin(ssid, pass);
  delay(10000); // Espera 10 segons a la connexió.
}
server.begin(); // Inicialitza el servidor web en el port 80.

// Connexió realitzada, mostra l'estat de la xarxa:

// Mostra la SSID de la xarxa a la que s'està connectat:
```

```

Serial.print("SSID: ");
Serial.println(WiFi.SSID());

// Mostar l'adreça IP de la placa Arduino:
IPAddress ip = WiFi.localIP();
Serial.print("Adreça IP: ");
Serial.println(ip);

// Mostra la força de la senyal rebuda:
long rssi = WiFi.RSSI();
Serial.print("Força de la senyal (RSSI):");
Serial.print(rssi);
Serial.println(" dBm");
// Mostra la direcció a utilitzar en el navegador d'internet:
Serial.print("Per controlar el braç robot, obrir en el navegador web la direcció següent:
http://");
Serial.println(ip);
}

void loop() {
  WiFiClient client = server.available(); // Prepara el servidor per possibles entrades
  (clients)

  if (client) { // Quan detecti un client a través d'una petició HTTP,
    Serial.println("nou client"); // mostra un missatge per el Monitor Serie.
    String currentLine = ""; // Crea un string per guardar les dades entrants
    while (client.connected()) { // loop mentre el client estigui connectat.
      if (client.available()) {
        char c = client.read(); // Llegeix la petició HTTP i
        Serial.write(c); // mostra-la pel Monitor Serie.
        if (c == '\n') { // Quan rebí una línia en blanc vol dir que la petició HTTP
          ha acabat.

          // Aquest és el final de la sol·licitud HTTP del client i el servidor web està llest per
          enviar una resposta:
          if (currentLine.length() == 0) {
            // S'envia al client una resposta HTTP
            client.println("HTTP/1.1 200 OK");
            client.println("Content-type:text/html");
            client.println("Connexió: tancada");
            client.println();

            // Crea la pàgina web en format HTML:

```

```
client.println("<h1>Control Robot</h1>"); // Títol de la pàgina
client.println("<table border=1 style='text-align:center'>"); // Crea una taula des
de la que controlar el robot.
client.println("<tr><th>Servo</th><th>Control</th></tr>"); // Comença a omplir la
taula.
```

```
// Segueix omplint la taula:
```

```
// Crea els títols per cada servo i els angles de posicionament de cada servo:
```

```
// Servo Tronc
```

```
client.print("<tr><td>Servo Tronc</td>");
client.println("<td><a href='/sTronc/10'>10</a> / <a href='/sTronc/20'>20</a> / <a
href='/sTronc/30'>30</a> / <a href='/sTronc/40'>40</a> / <a href='/sTronc/50'>50</a> / <a
href='/sTronc/60'>60</a> / <a href='/sTronc/70'>70</a> / <a href='/sTronc/80'>80</a> / <a
href='/sTronc/90'>90</a> / <a href='/sTronc/100'>100</a> / <a
href='/sTronc/110'>110</a> / <a href='/sTronc/120'>120</a> / <a
href='/sTronc/130'>130</a> / <a href='/sTronc/140'>140</a> / <a
href='/sTronc/150'>150</a> / <a href='/sTronc/160'>160</a> / <a
href='/sTronc/170'>170</a></td></tr>");
```

```
// Servo Espatlla
```

```
client.print("<tr><td>Servo Espatlla</td>");
client.println("<td><a href='/sEspatlla/10'>10</a> / <a href='/sEspatlla/20'>20</a> /
<a href='/sEspatlla/30'>30</a> / <a href='/sEspatlla/40'>40</a> / <a
href='/sEspatlla/50'>50</a> / <a href='/sEspatlla/60'>60</a> / <a
href='/sEspatlla/70'>70</a> / <a href='/sEspatlla/80'>80</a> / <a
href='/sEspatlla/90'>90</a> / <a href='/sEspatlla/100'>100</a> / <a
href='/sEspatlla/110'>110</a> / <a href='/sEspatlla/120'>120</a> / <a
href='/sEspatlla/130'>130</a> / <a href='/sEspatlla/140'>140</a> / <a
href='/sEspatlla/150'>150</a> / <a href='/sEspatlla/160'>160</a> / <a
href='/sEspatlla/170'>170</a></td></tr>");
```

```
// Servo Colze
```

```
client.print("<tr><td>Servo Colze</td>");
client.println("<td><a href='/sColze/10'>10</a> / <a href='/sColze/20'>20</a> / <a
href='/sColze/30'>30</a> / <a href='/sColze/40'>40</a> / <a href='/sColze/50'>50</a> / <a
href='/sColze/60'>60</a> / <a href='/sColze/70'>70</a> / <a href='/sColze/80'>80</a> / <a
href='/sColze/90'>90</a> / <a href='/sColze/100'>100</a> / <a
href='/sColze/110'>110</a> / <a href='/sColze/120'>120</a> / <a
href='/sColze/130'>130</a> / <a href='/sColze/140'>140</a> / <a
href='/sColze/150'>150</a> / <a href='/sColze/160'>160</a> / <a
href='/sColze/170'>170</a></td></tr>");
```

```
// Servo Canell
```

```
client.print("<tr><td>Servo Canell</td>");
```

```

        client.println("<td><a href='/sCanell/10'>10</a> / <a href='/sCanell/20'>20</a> / <a
href='/sCanell/30'>30</a> / <a href='/sCanell/40'>40</a> / <a href='/sCanell/50'>50</a> /
<a href='/sCanell/60'>60</a> / <a href='/sCanell/70'>70</a> / <a
href='/sCanell/80'>80</a> / <a href='/sCanell/90'>90</a> / <a href='/sCanell/100'>100</a>
/ <a href='/sCanell/110'>110</a> / <a href='/sCanell/120'>120</a> / <a
href='/sCanell/130'>130</a> / <a href='/sCanell/140'>140</a> / <a
href='/sCanell/150'>150</a> / <a href='/sCanell/160'>160</a> / <a
href='/sCanell/170'>170</a></td></tr>");

```

```

// Servo Pinça

```

```

client.print("<tr><td>Servo Pinça</td>");

```

```

client.println("<td><a href='/sPinza/10'>10</a> / <a href='/sPinza/20'>20</a> / <a
href='/sPinza/30'>30</a> / <a href='/sPinza/40'>40</a> / <a href='/sPinza/50'>50</a> / <a
href='/sPinza/60'>60</a> / <a href='/sPinza/70'>70</a> / <a href='/sPinza/80'>80</a> / <a
href='/sPinza/90'>90</a> / <a href='/sPinza/100'>100</a> / <a href='/sPinza/110'>110</a>
/ <a href='/sPinza/120'>120</a> / <a href='/sPinza/130'>130</a> / <a
href='/sPinza/140'>140</a> / <a href='/sPinza/150'>150</a> / <a
href='/sPinza/160'>160</a> / <a href='/sPinza/170'>170</a> / <a
href='/sPinza/180'>180</a></td></tr>");

```

```

client.println("</table>");

```

```

// La resposta HTTP acaba amb una altra línia en blanc:

```

```

client.println();

```

```

// Surt del while loop:

```

```

break;

```

```

} else { // Si hi ha una nova línia, neteja el string.

```

```

    currentLine = "";

```

```

}

```

```

} else if (c != '\r') { // Si hi ha alguna cosa més,

```

```

    currentLine += c; // afegeix-lo a la variable currentLine.

```

```

}

```

```

// Si es selecciona alguna de les opcions de control de la web,

```

```

// mou el servo corresponent a la posició seleccionada i actualitza el valor de la seva
posició.

```

```

if (currentLine.endsWith("GET /sTronc/10")) {

```

```

    servoTronc.write(10);

```

```

    angleTronc = 10;

```

```

} else if (currentLine.endsWith("GET /sTronc/20")) {

```

```

    servoTronc.write(20);

```

```

    angleTronc = 20;

```

```

} else if (currentLine.endsWith("GET /sTronc/30")) {

```



```
servoTronc.write(30);
angleTronc = 30;
} else if (currentLine.endsWith("GET /sTronc/40")) {
servoTronc.write(40);
angleTronc = 40;
} else if (currentLine.endsWith("GET /sTronc/50")) {
servoTronc.write(50);
angleTronc = 50;
} else if (currentLine.endsWith("GET /sTronc/60")) {
servoTronc.write(60);
angleTronc = 60;
} else if (currentLine.endsWith("GET /sTronc/70")) {
servoTronc.write(70);
angleTronc = 70;
} else if (currentLine.endsWith("GET /sTronc/80")) {
servoTronc.write(80);
angleTronc = 80;
} else if (currentLine.endsWith("GET /sTronc/90")) {
servoTronc.write(90);
angleTronc = 90;
} else if (currentLine.endsWith("GET /sTronc/100")) {
servoTronc.write(100);
angleTronc = 100;
} else if (currentLine.endsWith("GET /sTronc/110")) {
servoTronc.write(110);
angleTronc = 110;
} else if (currentLine.endsWith("GET /sTronc/120")) {
servoTronc.write(120);
angleTronc = 120;
} else if (currentLine.endsWith("GET /sTronc/130")) {
servoTronc.write(130);
angleTronc = 130;
} else if (currentLine.endsWith("GET /sTronc/140")) {
servoTronc.write(140);
angleTronc = 140;
} else if (currentLine.endsWith("GET /sTronc/150")) {
servoTronc.write(150);
angleTronc = 150;
} else if (currentLine.endsWith("GET /sTronc/160")) {
servoTronc.write(160);
angleTronc = 160;
} else if (currentLine.endsWith("GET /sTronc/170")) {
servoTronc.write(170);
angleTronc = 170;
}
```

```
    else if (currentLine.endsWith("GET /sEspatlla/10")) {
        servoEspatlla.write(10);
        angleEspatlla = 10;
    } else if (currentLine.endsWith("GET /sEspatlla/20")) {
        servoEspatlla.write(20);
        angleEspatlla = 20;
    } else if (currentLine.endsWith("GET /sEspatlla/30")) {
        servoEspatlla.write(30);
        angleEspatlla = 30;
    } else if (currentLine.endsWith("GET /sEspatlla/40")) {
        servoEspatlla.write(40);
        angleEspatlla = 40;
    } else if (currentLine.endsWith("GET /sEspatlla/50")) {
        servoEspatlla.write(50);
        angleEspatlla = 50;
    } else if (currentLine.endsWith("GET /sEspatlla/60")) {
        servoEspatlla.write(60);
        angleEspatlla = 60;
    } else if (currentLine.endsWith("GET /sEspatlla/70")) {
        servoEspatlla.write(70);
        angleEspatlla = 70;
    } else if (currentLine.endsWith("GET /sEspatlla/80")) {
        servoEspatlla.write(80);
        angleEspatlla = 80;
    } else if (currentLine.endsWith("GET /sEspatlla/90")) {
        servoEspatlla.write(90);
        angleEspatlla = 90;
    } else if (currentLine.endsWith("GET /sEspatlla/100")) {
        servoEspatlla.write(100);
        angleEspatlla = 100;
    } else if (currentLine.endsWith("GET /sEspatlla/110")) {
        servoEspatlla.write(110);
        angleEspatlla = 110;
    } else if (currentLine.endsWith("GET /sEspatlla/120")) {
        servoEspatlla.write(120);
        angleEspatlla = 120;
    } else if (currentLine.endsWith("GET /sEspatlla/130")) {
        servoEspatlla.write(130);
        angleEspatlla = 130;
    } else if (currentLine.endsWith("GET /sEspatlla/140")) {
        servoEspatlla.write(140);
        angleEspatlla = 140;
    } else if (currentLine.endsWith("GET /sEspatlla/150")) {
        servoEspatlla.write(150);
        angleEspatlla = 150;
    }
```

```
} else if (currentLine.endsWith("GET /sEspatlla/160")) {
  servoEspatlla.write(160);
  angleEspatlla = 160;
} else if (currentLine.endsWith("GET /sEspatlla/170")) {
  servoEspatlla.write(170);
  angleEspatlla = 170;
}

else if (currentLine.endsWith("GET /sColze/10")) {
  servoColze.write(10);
  angleColze = 10;
} else if (currentLine.endsWith("GET /sColze/20")) {
  servoColze.write(20);
  angleColze = 20;
} else if (currentLine.endsWith("GET /sColze/30")) {
  servoColze.write(30);
  angleColze = 30;
} else if (currentLine.endsWith("GET /sColze/40")) {
  servoColze.write(40);
  angleColze = 40;
} else if (currentLine.endsWith("GET /sColze/50")) {
  servoColze.write(50);
  angleColze = 50;
} else if (currentLine.endsWith("GET /sColze/60")) {
  servoColze.write(60);
  angleColze = 60;
} else if (currentLine.endsWith("GET /sColze/70")) {
  servoColze.write(70);
  angleColze = 70;
} else if (currentLine.endsWith("GET /sColze/80")) {
  servoColze.write(80);
  angleColze = 80;
} else if (currentLine.endsWith("GET /sColze/90")) {
  servoColze.write(90);
  angleColze = 90;
} else if (currentLine.endsWith("GET /sColze/100")) {
  servoColze.write(100);
  angleColze = 100;
} else if (currentLine.endsWith("GET /sColze/110")) {
  servoColze.write(110);
  angleColze = 110;
} else if (currentLine.endsWith("GET /sColze/120")) {
  servoColze.write(120);
  angleColze = 120;
} else if (currentLine.endsWith("GET /sColze/130")) {
```

```
servoColze.write(130);
angleColze = 130;
} else if (currentLine.endsWith("GET /sColze/140")) {
servoColze.write(140);
angleColze = 140;
} else if (currentLine.endsWith("GET /sColze/150")) {
servoColze.write(150);
angleColze = 150;
} else if (currentLine.endsWith("GET /sColze/160")) {
servoColze.write(160);
angleColze = 160;
} else if (currentLine.endsWith("GET /sColze/170")) {
servoColze.write(170);
angleColze = 170;
}

else if (currentLine.endsWith("GET /sCanell/10")) {
servoCanell.write(10);
angleCanell = 10;
} else if (currentLine.endsWith("GET /sCanell/20")) {
servoCanell.write(20);
angleCanell = 20;
} else if (currentLine.endsWith("GET /sCanell/30")) {
servoCanell.write(30);
angleCanell = 30;
} else if (currentLine.endsWith("GET /sCanell/40")) {
servoCanell.write(40);
angleCanell = 40;
} else if (currentLine.endsWith("GET /sCanell/50")) {
servoCanell.write(50);
angleCanell = 50;
} else if (currentLine.endsWith("GET /sCanell/60")) {
servoCanell.write(60);
angleCanell = 60;
} else if (currentLine.endsWith("GET /sCanell/70")) {
servoCanell.write(70);
angleCanell = 70;
} else if (currentLine.endsWith("GET /sCanell/80")) {
servoCanell.write(80);
angleCanell = 80;
} else if (currentLine.endsWith("GET /sCanell/90")) {
servoCanell.write(90);
angleCanell = 90;
} else if (currentLine.endsWith("GET /sCanell/100")) {
servoCanell.write(100);
```

```
    angleCanell = 100;
  } else if (currentLine.endsWith("GET /sCanell/110")) {
    servoCanell.write(110);
    angleCanell = 110;
  } else if (currentLine.endsWith("GET /sCanell/120")) {
    servoCanell.write(120);
    angleCanell = 120;
  } else if (currentLine.endsWith("GET /sCanell/130")) {
    servoCanell.write(130);
    angleCanell = 130;
  } else if (currentLine.endsWith("GET /sCanell/140")) {
    servoCanell.write(140);
    angleCanell = 140;
  } else if (currentLine.endsWith("GET /sCanell/150")) {
    servoCanell.write(150);
    angleCanell = 150;
  } else if (currentLine.endsWith("GET /sCanell/160")) {
    servoCanell.write(160);
    angleCanell = 160;
  } else if (currentLine.endsWith("GET /sCanell/170")) {
    servoCanell.write(170);
    angleCanell = 170;
  }
}

else if (currentLine.endsWith("GET /sPinza/10")) {
  servoPinza.write(10);
  anglePinza = 10;
} else if (currentLine.endsWith("GET /sPinza/20")) {
  servoPinza.write(20);
  anglePinza = 20;
} else if (currentLine.endsWith("GET /sPinza/30")) {
  servoPinza.write(30);
  anglePinza = 30;
} else if (currentLine.endsWith("GET /sPinza/40")) {
  servoPinza.write(40);
  anglePinza = 40;
} else if (currentLine.endsWith("GET /sPinza/50")) {
  servoPinza.write(50);
  anglePinza = 50;
} else if (currentLine.endsWith("GET /sPinza/60")) {
  servoPinza.write(60);
  anglePinza = 60;
} else if (currentLine.endsWith("GET /sPinza/70")) {
  servoPinza.write(70);
  anglePinza = 70;
}
```

```
} else if (currentLine.endsWith("GET /sPinza/80")) {
    servoPinza.write(80);
    anglePinza = 80;
} else if (currentLine.endsWith("GET /sPinza/90")) {
    servoPinza.write(90);
    anglePinza = 90;
} else if (currentLine.endsWith("GET /sPinza/100")) {
    servoPinza.write(100);
    anglePinza = 100;
} else if (currentLine.endsWith("GET /sPinza/110")) {
    servoPinza.write(110);
    anglePinza = 110;
} else if (currentLine.endsWith("GET /sPinza/120")) {
    servoPinza.write(120);
    anglePinza = 120;
} else if (currentLine.endsWith("GET /sPinza/130")) {
    servoPinza.write(130);
    anglePinza = 130;
} else if (currentLine.endsWith("GET /sPinza/140")) {
    servoPinza.write(140);
    anglePinza = 140;
} else if (currentLine.endsWith("GET /sPinza/150")) {
    servoPinza.write(150);
    anglePinza = 150;
} else if (currentLine.endsWith("GET /sPinza/160")) {
    servoPinza.write(160);
    anglePinza = 160;
} else if (currentLine.endsWith("GET /sPinza/170")) {
    servoPinza.write(170);
    anglePinza = 170;
} else if (currentLine.endsWith("GET /sPinza/180")) {
    servoPinza.write(180);
    anglePinza = 180;
}
}
}
}
// Tanca la connexió::
client.stop();
Serial.println("client desconnectat");
}
}
```

## Annex C. Programa de control mitjançant barres lliscants

```
#include <WiFiNINA.h>
```

```
#include <Servo.h>
```

```
Servo servoTronc; // Crea l'objecte servoTronc per controlar el servo del tronc.
```

```
Servo servoEstatlla; // Crea l'objecte servoEstatlla per controlar el servo de l'estatlla.
```

```
Servo servoColze; // Crea l'objecte servoColze per controlar el servo del colze.
```

```
Servo servoCanell; // Crea l'objecte servoCanell per controlar el servo del canell.
```

```
Servo servoPinza; // Crea l'objecte servoPinza per controlar el servo de la pinça.
```

```
// Crea les variables on guardar la posició dels servos:
```

```
int angleTronc = 90;
```

```
int angleEstatlla = 90;
```

```
int angleColze = 90;
```

```
int angleCanell = 90;
```

```
int anglePinza = 90;
```

```
// Crea les constants on guardar els números dels pins digital:
```

```
const int pinTronc = 3;
```

```
const int pinEstatlla = 5;
```

```
const int pinColze = 6;
```

```
const int pinCanell = 9;
```

```
const int pinPinza = 10;
```

```
// Crea les variables auxiliars que s'utilitzaran al llarg del programa:
```

```
int position1 = 0;
```

```
int position2 = 0;
```

```
int position3 = 0;
```

```
int position4 = 0;
```

```
int position5 = 0;
```

```
int position6 = 0;
```

```
String valueStringTronc = String(angleTronc);
```

```
String valueStringEstatlla = String(angleEstatlla);
```

```
String valueStringColze = String(angleColze);
```

```
String valueStringCanell = String(angleCanell);
```

```
String valueStringPinza = String(anglePinza);
```

```
//Dades de la xarxa Wi-Fi:

char ssid[] = "XXX";    // Nom de la xarxa SSID
char pass[] = "YYY";    // Contrasenya de xarxa

int status = WL_IDLE_STATUS;
WiFiServer server(80);

void setup() {

    // Assigna els objectes servos als pins digitals corresponents:

    servoTronc.attach(pinTronc);
    servoEstatlla.attach(pinEstatlla);
    servoColze.attach(pinColze);
    servoCanell.attach(pinCanell);
    servoPinza.attach(pinPinza);

    // Posiciona els servos en la seva posició inicial:

    servoTronc.write(angleTronc);
    servoEstatlla.write(angleEstatlla);
    servoColze.write(angleColze);
    servoCanell.write(angleCanell);
    servoPinza.write(anglePinza);

    Serial.begin(9600);    // Inicialitza la comunicació amb el Monitor Serie.

    // Inicia el mòdul Wi-Fi:

    // Si la comunicació falla, no es continua:

    if (WiFi.status() == WL_NO_MODULE) {
        Serial.println("La comunicació amb el mòdul Wi-Fi ha fallat!");
        while (true);
    }

    // Intenta connectar amb la xarxa Wi-Fi:

    while (status != WL_CONNECTED) {
        Serial.print("Intentant connectar amb la xarxa Wi-Fi anomenada: ");
        Serial.println(ssid);    // Mostra el nom de la xarxa Wi-Fi.

        // Connectat a la xarxa WPA/WPA2 (canviar aquesta línia si s'utilitza una xarxa oberta
        // o WEP):
```



```
    status = WiFi.begin(ssid, pass);
    delay(10000); // Espera 10 segons a la connexió.
}
server.begin(); // Inicialitza el servidor web en el port 80.

// Connexió realitzada, mostra l'estat de la xarxa:

// Mostra la SSID de la xarxa a la que s'està connectat:
Serial.print("SSID: ");
Serial.println(WiFi.SSID());

// Mostar l'adreça IP de la placa Arduino:
IPAddress ip = WiFi.localIP();
Serial.print("Adreça IP: ");
Serial.println(ip);

// Mostra la força de la senyal rebuda:
long rssi = WiFi.RSSI();
Serial.print("Força de la senyal (RSSI):");
Serial.print(rssi);
Serial.println(" dBm");
// Mostra la direcció a utilitzar en el navegador d'internet:
Serial.print("Per controlar el braç robot, obrir en el navegador web la direcció següent:
http://");
Serial.println(ip);
}

void loop() {

    WiFiClient client = server.available(); // Prepara el servidor per possibles entrades
    (clients).

    if (client) { // Quan detecti un client a través d'una petició HTTP,
        Serial.println("nou client"); // mostra un missatge per el Monitor Serie.
        String currentLine = ""; // Crea un string per guardar les dades entrants.
        while (client.connected()) { // loop mentre el client estigui connectat.
            if (client.available()) {
                char c = client.read(); // Llegeix la petició HTTP i
                Serial.write(c); // mostra-la pel Monitor Serie.
                if (c == '\n') { // Quan rebí una línia en blanc vol dir que la petició HTTP
                    ha acabat.

                        // Aquest és el final de la sol·licitud HTTP del client i el servidor web està llest per
                        enviar una resposta:
```

```

if (currentLine.length() == 0) {
    // S'envia al client una resposta HTTP:
    client.println("HTTP/1.1 200 OK");
    client.println("Content-type:text/html");
    client.println("Connection: close");
    client.println();

    // Crea la pàgina web en format HTML:

    client.println("<!DOCTYPE html><html>");
    client.println("<head><meta name=\"viewport\" content=\"width=device-width,
initial-scale=1\">");
    client.println("<link rel=\"icon\" href=\"data:;\">");
    client.println("<style>body { text-align: center; font-family: \"Trebuchet MS\", Arial;
margin-left:auto; margin-right:auto;}");
    client.println(".slider { width: 300px; }</style>");
    client.println("<script
src=\"https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js\"></script>");

    client.println("</head><body><h1>Control Servos</h1>"); // Títol de la pàgina

    // Crea les barres lliscants:

    client.println("<h2>Servo Tronc</h2>"); // Crea el títol de la barra lliscant.
    client.println("<p>Posicio: <span id=\"servoPosTronc\"></span></p>"); // Crea la
variable on es mostrarà la posició actual del servo .
    client.println("<input type=\"range\" min=\"10\" max=\"170\" class=\"slider\"
id=\"servoTroncRange\" onmouseup=\"moverServos(servoTroncRange.value,
servoEspatllaRange.value, servoColzeRange.value, servoCanellRange.value,
servoPinzaRange.value)\" onchange=\"servoTronc(this.value)\"
value=\"\"+valueStringTronc+\"\"/>"); // Crea la barra lliscant.

    client.println("<h2>Servo Espatlla</h2>");
    client.println("<p>Posicio: <span id=\"servoPosEspatlla\"></span></p>");
    client.println("<input type=\"range\" min=\"10\" max=\"170\" class=\"slider\"
id=\"servoEspatllaRange\" onmouseup=\"moverServos(servoTroncRange.value,
servoEspatllaRange.value, servoColzeRange.value, servoCanellRange.value,
servoPinzaRange.value)\" onchange=\"servoEspatlla(this.value)\"
value=\"\"+valueStringEspatlla+\"\"/>");

    client.println("<h2>Servo Colze</h2>");
    client.println("<p>Posicio: <span id=\"servoPosColze\"></span></p>");
    client.println("<input type=\"range\" min=\"10\" max=\"170\" class=\"slider\"
id=\"servoColzeRange\" onmouseup=\"moverServos(servoTroncRange.value,

```

```
servoEspatllaRange.value, servoColzeRange.value, servoCanellRange.value,
servoPinzaRange.value)" onchange="servoColze(this.value)"
value="" +valueStringColze+"/>");
```

```
client.println("<h2>Servo Canell</h2>");
client.println("<p>Posicio: <span id=\"servoPosCanell\"></span></p>");
client.println("<input type=\"range\" min=\"10\" max=\"170\" class=\"slider\"
id=\"servoCanellRange\" onmouseup=\"moverServos(servoTroncRange.value,
servoEspatllaRange.value, servoColzeRange.value, servoCanellRange.value,
servoPinzaRange.value)\" onchange=\"servoCanell(this.value)\"
value=\"\" +valueStringCanell+\"/>");
```

```
client.println("<h2>Servo Pinza</h2>");
client.println("<p>Posicio: <span id=\"servoPosPinza\"></span></p>");
client.println("<input type=\"range\" min=\"10\" max=\"170\" class=\"slider\"
id=\"servoPinzaRange\" onmouseup=\"moverServos(servoTroncRange.value,
servoEspatllaRange.value, servoColzeRange.value, servoCanellRange.value,
servoPinzaRange.value)\" onchange=\"servoPinza(this.value)\"
value=\"\" +valueStringPinza+\"/>");
```

// Crea els *scripts*<sup>8</sup> per actualitzar les variables que ens mostren la posició de cada servo en la pàgina web:

```
client.println("<script>var sliderTronc =
document.getElementById(\"servoTroncRange\");");
client.println("var sliderEspatlla =
document.getElementById(\"servoEspatllaRange\");");
client.println("var sliderColze =
document.getElementById(\"servoColzeRange\");");
client.println("var sliderCanell =
document.getElementById(\"servoCanellRange\");");
client.println("var sliderPinza = document.getElementById(\"servoPinzaRange\");");
client.println("var servoPosTronc = document.getElementById(\"servoPosTronc\");
servoPosTronc.innerHTML = sliderTronc.value;");
client.println("var servoPosEspatlla =
document.getElementById(\"servoPosEspatlla\"); servoPosEspatlla.innerHTML =
sliderEspatlla.value;");
client.println("var servoPosColze = document.getElementById(\"servoPosColze\");
servoPosColze.innerHTML = sliderColze.value;");
client.println("var servoPosCanell =
document.getElementById(\"servoPosCanell\"); servoPosCanell.innerHTML =
sliderCanell.value;");
```

<sup>8</sup> Un *script* és un programa inserit dins d'un document HTML i que es interpreta i executa pel navegador de l'usuari.

```

        client.println("var servoPosPinza = document.getElementById(\"servoPosPinza\");
servoPosPinza.innerHTML = sliderPinza.value;");
        client.println("sliderTronc.oninput = function() { sliderTronc.value = this.value;
servoPosTronc.innerHTML = this.value; }");
        client.println("sliderEspatlla.oninput = function() { sliderEspatlla.value = this.value;
servoPosEspatlla.innerHTML = this.value; }");
        client.println("sliderColze.oninput = function() { sliderColze.value = this.value;
servoPosColze.innerHTML = this.value; }");
        client.println("sliderCanell.oninput = function() { sliderCanell.value = this.value;
servoPosCanell.innerHTML = this.value; }");
        client.println("sliderPinza.oninput = function() { sliderPinza.value = this.value;
servoPosPinza.innerHTML = this.value; }");

```

// Crea el *script* que fa que la pàgina web ens retorni una seqüència amb l'angle de cada servo i uns símbols per separar-los.

// Els *strings* que guarden els angles de la seqüència sempre tindran tres xifres. Si l'angle es troba entre zero i nou, ambdós inclosos, caldrà afegir dos zeros davant l'angle. Si l'angle es troba entre déu i noranta-nou, ambdós inclosos, caldrà afegir un zero davant l'angle.

```

        client.println("function moverServos(angleTronc, angleEspatlla, angleColze,
angleCanell, anglePinza) { if (angleTronc < 100) {; angleTronc = \"0\" + angleTronc;} if
(angleTronc < 10) {; angleTronc = \"00\" + angleTronc;} if (angleEspatlla < 100) {;
angleEspatlla = \"0\" + angleEspatlla;} if (angleEspatlla < 10) {; angleEspatlla = \"0\" +
angleEspatlla;} if (angleColze < 100) {; angleColze = \"0\" + angleColze;} if (angleColze <
10) {; angleColze = \"00\" + angleColze;} if (angleCanell < 100) {; angleCanell = \"0\" +
angleCanell;} if (angleCanell < 10) {; angleCanell = \"00\" + angleCanell;} if (anglePinza <
100) {; anglePinza = \"0\" + anglePinza;} if (anglePinza < 10) {; anglePinza = \"00\" +
anglePinza;}");
        client.println("$.get(\"srs\" + \"$\" + angleTronc + \"=\" + angleEspatlla + \"*\" +
angleColze + \"%\" + angleCanell + \"!\" + anglePinza + \"&\"); {Connection:
close};</script>");
        client.println("</body></html>");

```

// La resposta HTTP acaba amb una altra línia en blanc.

```
client.println();
```

// Surt del *while loop*:

```
break;
```

```
}
```

else { // Si hi ha una nova línia, neteja el *string*.

```
currentLine = "";
```

```
}
```

```
}
```

else if (c != '\r') { // Si hi ha alguna cosa més,

```
    currentLine += c;    // afegiu-lo a la variable currentLine.  
}
```

// Si es mou alguna de les barres lliscants, llegeix tots els angles de la seqüència que retorna la pàgina web i mou els servos que correspongui a les noves posicions:

```
if (currentLine.endsWith("&")) {  
  
    position1 = currentLine.indexOf("$");  
    position2 = currentLine.indexOf("=");  
    position3 = currentLine.indexOf("*");  
    position4 = currentLine.indexOf("%");  
    position5 = currentLine.indexOf("!");  
    position6 = currentLine.indexOf("&");  
  
    valueStringTronc = currentLine.substring(position1+1,position2);  
    valueStringEpatlla = currentLine.substring(position2+1,position3);  
    valueStringColze = currentLine.substring(position3+1,position4);  
    valueStringCanell = currentLine.substring(position4+1,position5);  
    valueStringPinza = currentLine.substring(position5+1,position6);  
  
    // Rota els servos:  
  
    angleTronc = valueStringTronc.toInt();  
    servoTronc.write(angleTronc);  
    Serial.println(angleTronc);  
  
    angleEpatlla = valueStringEpatlla.toInt();  
    servoEpatlla.write(angleEpatlla);  
    Serial.println(angleEpatlla);  
  
    angleColze = valueStringColze.toInt();  
    servoColze.write(angleColze);  
    Serial.println(angleColze);  
  
    angleCanell = valueStringCanell.toInt();  
    servoCanell.write(angleCanell);  
    Serial.println(angleCanell);  
  
    anglePinza = valueStringPinza.toInt();  
    servoPinza.write(anglePinza);  
    Serial.println(anglePinza);  
}  
}  
}
```

```
// Tanca la connexió:  
client.stop();  
Serial.println("client desconnectat");  
}  
}
```