# 1, Chi tiết kế hoạch Sprint 2

* Tuấn:
  + Tiếp tục tìm hiểu RTOS, sẵn sàng phương án ghép nối code
* Quân:
  + Bắt đầu code phần mềm xử lý giọng nói
* Trung:
  + Bắt đầu xây dựng thuật giải, code cho động cơ
  + Module hóa code
* Ánh + Tùng:
  + Làm mạch nguồn đầu ra 5V – 3,3V
  + Bắt đầu code cho module wifi
  + Xây dựng thuật giải gửi nhận gói tin trên Atmega16

# 2, BÁO CÁO TỔNG KẾT SPRINT 2

* Tuấn:
  + Đã nắm được cơ chế hoạt động của RTOS
  + Chưa nắm được cách điều khiển các tiến trình rõ ràng
* Quân:
  + Đã xong phần mềm xử lý Speech to Text trên Java – Android
* Trung:
  + Đã điều khiển được L298
  + Đã cho ra được thuật giải
  + Chưa tổng hợp thành module trong hàm
* Ánh:
  + Đã hoàn thành bộ nguồn
* Tùng:
  + Đã hoàn thành cơ chế xử lý, cách thức gửi nhận qua module Wifi
  + Chưa tổng hợp thành module trong hàm

# 3, CHI TIẾT BÁO CÁO

* Phần mềm điều khiển trên di động:
  + Gặp phải khó khăn trong việc kết hợp API Google vào Qt
  + Gặp phải khó khăn trong việc lập trình Chat trên Java
  + Đã có giải pháp
    - Join 1 thành viên tạm thời vào: Trần Thanh Tĩnh – sv CNTT 1.02 k57 BKHN
    - Đã lập trình Chat trên Java ok
    - Đã ghép được API Google vào
    - Đã lập trình được giao diện cụ thể, có demo
* Phần điều khiển động cơ:
  + Mặc dù đã có thuật giải, nhưng vẫn chưa điều khiển được động cơ theo ý
  + Tùng hỗ trợ Trung, đã giải quyết được điều khiển động cơ
* Gói tin Packet có thiết lập lại cho đơn giản: gồm 3 bytes
  + Byte 1: loại lệnh (mode)
    - 0: điều khiển động cơ
    - 1: nhận thông tin xe gửi về (cảm biến)
  + Byte 2: dữ liệu kèm – Data
    - Nếu là điều khiển động cơ thì: (3 bit vẫn có chức năng như cũ)
      * Data < 4 (cho là 011 = 3): dừng động cơ
      * 100 = 4: lùi
      * 101 = 5: quay trái
      * 110 = 6: quay phải
      * 111 = 7: tiến lên
    - Nếu là nhận thông tin gửi về thì Data là dữ liệu số của nhiệt độ, …
  + Byte 3: Reserved – Dự trữ