**Minimap**

* Kurzerklärung Elemente auf Minimap
* Map wird also nach beschriebenem Algorithmus analyisert
* Was ist aktuell auf der Minimap zu sehen?
  + Gesehene Fläche blau
  + Unbekannt weiss
  + Sicht bis zu den Hindernissen
  + Geplanter Pfad (Erklärung folgt)
* Senser liefert regelmässig Daten
* Infos werden der intern verwalteten Karte hinzugefügt
* Minimap ist also genaue Abbildung des Wissensstandes des Roboters
* Rote Punkte erwähnt. Was bedeuten die genau?

**Pfadfindung 1**

* Ursprüngliche Idee in die grösste unerforschte Zone leiten -> Wieso nicht?
  + Theoretisch: Was wenn Karte unendlich gross ist? Kein Zentrum vorhanden.
  + Wahrscheinlichkeit einer Kollision bis ins Zentrum enorm gross.
  + -> Verworfen
* Weitere Idee: Wo gibt’s bestimmt etwas zu entdecken?
  + Dort wo Schatten geworfen wird. Sprich auf Kante, also genau die Linie zwischen erkundetem und unerforschtem Gebiet.
  + Zentrum der längsten dieser Kanten wird angesteuert, Wahrscheinlichkeit mehr von der Karte zu entdecken gross.
* Nächstes Problem: Wie Roboter dorthin bewegen? -> Dijkstra Implementierung.
* Basierend auf was?
* Karte rastern
* Bild

**Pfadfindung 2**

* Gute Voraussetzung um Weg zum nächsten Zwischenziel zu finden.
* Theoretisch Dijkstra auch mit jedem Pixel der Karte möglich. Wieso nicht?
  + Performance
  + Breite des Roboters muss beachtet werden
* Simulation laufen lassen mit Comments
* Kollision
  + Kommt vor, sogar bewusst / gewollt.
  + Roboter interpretiert unerforschtes Gelände als befahrbar.
    - Kante weit in unerforschtem Gebiet -> Steuert gerade darauf zu
  + Wäre verhinderbar mit häufigerem Berechnen des Weges mit Dijkstra
  + Oder: unerforscht = unbefahrbar
  + Nach Kollision Neukalibrierung des Roboters.
    - Rückkehrern zu einem letzten, sicheren Dijkstra Waypoint
  + Kollision-Funktionalität bereits durch Karteneditor gegeben

**Herausforderungen des Roboters**

* …

**Projektablauf**

* Grundsätzlich sehr gut eingehalten
* Keine Zähne ausbeissenden Komplikationen die eine grosse Verschiebung bedeutet hätten.
* Kleine Änderung während des Testens. Tests gingen recht fix, darum schneller Konzentration auf Path-Finding.
* Dafür Tests nach kürzeren Intervallen und iterativ.

**Fazit**

* Sehr herausfordernd
  + Wenn wir das aktuelle Resultat vor ½ Jahr gesehen hätten -> kaum realistisch.
  + Deshalb umso mehr zufrieden
* Zusammenarbeit grossartig
* Kurze, prägnante Diskussionen wie / was / wann
* Code zusammenführen grösstenteils reibungslos
* Sackgassen ohne grösseren Impact auf Zeitplanung
* Grosse Schwierigkeit: Jonglieren mit riesigen Datenmengen.
  + Pixel der Karte analysieren, handhaben
  + Bei Fehler Ursache sehr schwer zu finden