Design Thinking

# Problem und Problemfeld verstehen

* 1. Wie können wir dem Customer die Individualisierung seiner Produkte möglichst leicht machen?
  2. Wie können wir dem Customer die Bestellung seiner Produkte möglichst leicht machen?
  3. Wie können wir dem Customer seine Bestellungen zustellen?
  4. Wie können wir dem Customer eine hohe Qualität bieten?
  5. Wie können wir dem Customer eine hohe Transparent bieten?
  6. Wie können wir dem Factory Owner helfen seine Produkte anzubieten?
  7. Wie können wir dem Factory Owner helfen seine Fabrik effektiv zu nutzen?
  8. Wie können wir dem Factory Owner helfen den Überblick über seine Fabrik zu behalten?
  9. Wie können wir dem Factory Owner helfen seine Fabrik leicht an neue Situationen anzupassen?
  10. Wie können wir dem Agent Developer die Entwicklung von Agents erleichtern?
  11. Wie können wir dem Agent Developer die Integration von neuen Agents erleichtern?
  12. Wie können wir dem Agent Developer helfen die Performance seiner Agents zu verfolgen?
  13. Wie können wir dem System Designer helfen ein skalierbares System zu entwickeln?

Ein typischer **Customer** legt viel Wert darauf viele verschiedene Joghurtsorten ausprobieren zu können. Da es ihn nervt aktuell dafür bei vielen verschiedenen Shops kaufen zu müssen wünscht er sich einen Shop der ihm alle möglichen Joghurtsorten an einer zentralen Stelle anbietet. Natürlich soll die Bestellung seiner Joghurts ohne großen Aufwand geschehen können und es wird eine hohe Qualität des Bestellprozesses und der Produkte erwartet.

Ein typischer **Factory Owner** legt Wert darauf dass seine Fabrik weitgehend Autonom funktionieren kann um Personalkosten zu sparen.. D.h. die Produktion und der Vertrieb von Joghurts benötigt kein Menschliches Eingreifen, trotzdem soll der Prozess so flexibel sein dass leicht neue Produkte angeboten werden können. Zur Qualitätskontrolle hätte der Factory Owner gerne zu jeder Zeit die Möglichkeit einen Überblick über seine Fabrik zu erhalten.

Ein typischer **Agent Developer** ist vor allem daran interessiert möglichst leicht mit neuen Agenten die Fähigkeiten einer Fabrik zu erweitern. Bisher dauerte die Entwicklung und Integration von Agenten extrem lange und war äußerst Fehleranfällig da immer wieder Menschliches Eingreifen nötig war. Deswegen wünscht sich der Agent Developer mehr autonomie und Fabrik-weite Standards die es ihm ermöglichen seine Agenten schneller und zuverlässiger zu entwickeln und zu integrieren.

Ein typischer **System Designer** legt Wert auf eine hohe Qualität der übergeordneten Software Architektur. Da aktuell viele Einzellösungen existieren die eine Erweiterung des Systems schwierig machen, wünscht sich der System Designer für das neue System die Verwendung von allgemeinen Standards.

# Erforschen/Recherchieren

* 1. Unser Customer will seine Bestellungen komplett individualisieren können *(Warum legt er so einen Wert auf die Individualisierung?)*
  2. Unser Customer will über eine Web App bestellen? *(Warum nicht eine Native App o.ä.?)*
  3. Unser Customer will dass seine Bestellungen immer zu ihm nach Hause geliefert werden? *(Warum nicht woanders hin?)*
  4. Unser Customer will die Produktion seiner Bestellung in der Fabrik verfolgen können *(Warum ist die Verfolgung der Zustellung nicht so wichtig?)*
  5. Unser Factory Owner will dass Produkte über eine Web App angeboten werden *(Warum nicht eine Native App o.ä.?)*
  6. Unser Factory Owner will dass alle Produkte angeboten werden die seine Fabrik herstellen kann *(Warum konzentriert er sich nicht auf einige bestimmte?)*
  7. Unser Factory Owner will das Produkte “just-in-time” produziert werden *(Warum nicht im voraus produzieren um Effizienz zu erhöhen?)*
  8. Unser Factory Owner will unabhängig von den Verwendeten Maschinen sein
  9. Unser Factory Owner will ein Dashboard zu seiner Fabrik als Web App *(Warum nicht eine Native App o.ä.?)*
  10. Unser Factory Owner will detailierte Informationen zu seinen Maschinen sehen können *(Warum reichen nicht allgmeine Informationen über den Produktionsprozess?)*
  11. Unser Factory Owner will das Platform Provider automatisch Agenten bereitstellen können (Warum sollen externe Partner Agenten bereitstellen können?)
  12. Unser Factory Owner will dass Agenten den Produktionsprozess selbstständig optimieren
  13. Unser Factory Owner will manuell in die Produktion eingreifen können *(Warum keine Vollautomatisierung?)*
  14. Unser Agent Developer will die Software seiner Agenten auf einer Abstrakten Ebene schreiben können
  15. Unser Agent Developer will Protokolle, Frameworks und Interfaces benutzen mit denen er sich auskennt *(Warum will er nichts neues lernen?)*
  16. Unser Agent Developer will dass alle Agents gleich aufgebaut sind *(Warum keine spezifischen Lösungen für spezielle Anwendungsfälle?)*
  17. Unser Agent Developer will sehen können wie seine Agenten im vergleich zu anderen performen *(Warum ist unserem Agent Developer das wichtig?)*
  18. Unser System Designer will dass Agenten unabhängig voneinander existieren können und nur kommunizieren *(Warum dürfen keine Abhängigkeiten bestehen?)*
  19. Unser System Designer will dass allgemeine Standards verwendet werden *(Warum nicht spezielle Tools für spezielle Aufgaben?)*
  20. Unser System Designer will dass die Produktionslogik zentral bereitgestellt wird *(Warum nicht verteilt über alle Agenten?)*

# Synthese

# Lösungsfindung

* 1. Customers werden Schritt für Schritt durch die Erstellung von Joghurts geführt, um es so leicht wie möglich zu machen
  2. Customers bekommen die Zutaten nach relevanz sortiert angezeigt, um auf interessante Zutaten aufmerksam zu machen
  3. Customer können einen Account anlegen der Eingaben speichert, um Bestellungen unkomplizierter zu machen
  4. Customer bestellen über eine Web App, da diese von überall erreichbar ist und keine Installation erfordert
  5. Registrierte Nutzer haben einen Bereich der alle Bestellungen inklusive Status anzeigt
  6. Fabriken teilen dem Shop mit aus welchen Zutaten Joghurts hergestellt werden könne, um eine möglichst hohe Vielfalt zu gewährleisten
  7. Fabriken teilen dem Shop den Produktionsstatus von Bestellungen mit, damit der Customer den Fortschritt verfolgen kann
  8. Dashboard wird dem Factory Owner als Web App bereit gestellt, da diese von überall erreichbar ist und keine Installation erfordert
  9. Dashboard bietet die Option die Produktion zu stoppen, um die Möglichkeit zu geben in die Produktion einzugreifen
  10. Maschinen können sich dynamisch bei dem Dashboard registrieren, um unabhängig von einer bestimmten Fabrik Konfiguration zu sein
  11. Registrierte Maschinen können dem Dashboard ihren Status mitteilen, um detailierte Informationen zu liefern
  12. Das Dashboard zeigt alle registrierten Maschinen mit detailierten Informationen an
  13. Agenten können sich bei dem Dashboard registrieren (Wie bei Maschinen)
  14. Aus “Datenfeed” von Agenten wird berechnet wie effizient diese Arbeiten
  15. Eine API stellt allen Agenten Informationen zu Bestellungen und Produktionsplänen bereit

# Ideen durch Prototyping erlebbar machen

Prototypen können den PSPs der Sprints “15.11.2017 - 22.11.2017” und “23.11.2017 - 29.11.2017” entnommen werden.

# Testen