Input = {Customer(x).Has\_Height, Customer(x).Has\_Weight, Customer(x).Has\_Age, Customer(x).Gender, Illness[n], Target, HasActiveLevel(Customer(x), PhysicActiveLevel(act\_level\_input)}

Known = {Height, Weight, Gender, Age, PhysicalActiveLevel, }

(Fact, Rule)

Fact:

{

}

Rule:

{

{[1, Height], [1,Weight] | [BMIvalue, funct:BMIvalueCalc(height,weight)]},

{[1, BMIvalue] | [BMIrange, funct:BMIrangeCalc(bmi\_value)]},

{[1, BMIrange] | [BMIlevel, funct:BMIlevelCalc(bmi\_range)]},

{[2, Gender, gender, Value(“Nam”)], [1, Height], [1,Weight], [1, Age] | [BMRvalue, funct:BMRvalueOfMaleCalc(height, weight, age)]},

{[2, Gender, gender, Value(“Nữ”)], [1, Height], [1,Weight], [1, Age] | [BMRvalue, funct:BMRvalueOfFemaleCalc(height, weight, age)]}

{[1, PhysicalActiveLevel] | [PhysicalActiveValue, funct:PAvalueCalc(pa\_level)]},

{[1, BMRvalue], [1,PhysicalActiveValue], [1, Target] | [CaloPerDay, funct:CaloCalc(bmr\_value, pa\_value,target)]},

{[2, Target, target, Value(“Giữ cân”)],[1,CaloPerDay] | [CaloForEachMeal,funct:CaloToKeepWeightCalc(calo\_per\_day)]}

{[2, Target, target, Value(“Tăng cân”)],[1,CaloPerDay] | [CaloForEachMeal,funct:CaloToGainWeightCalc(calo\_per\_day)]}

{[2, Target, target, Value(“Giảm cân”)],[1,CaloPerDay] | [CaloForEachMeal,funct:CaloToLoseWeightCalc(calo\_per\_day)]}

{[1, ListIllness] | [ListAvoidFood, funct:FindAvoidFood(list\_illness)]},//Tách nhóm groupavoid

{[1, ListIllness] | [ListNeedFood, funct:FindNeedFood(list\_illness)]}//Tách nhóm groupneed

{[1, ListIllness] | [ListLimitFood, funct:FindLimitFood(list\_illness)]}//Tách nhóm grouplimit

{[1, ListAvoidFood], [1, ListAllFood] | [ListFood, funct:FilterFood(list\_illness, list\_food)]},

}

Lập thực đơn:

- Tính BMI

- Tính BMR

- Tính Chỉ số hoạt động

- Tính lượng Calo cần trong 1 ngày

- Tính lượng calo tùy đối tượng

- Tìm danh sách các thực phẩm, nhóm thực phẩm phụ thuộc từng tình trạng, nhu cầu, dị ứng (nếu có)

- Lọc ra các thực phẩm không dùng chung

- Đánh giá mức độ quan trọng của từng thực phẩm đối với nhu cầu cũng như tình trạng

- Tùy vào độ ưu tiên, chọn ra các danh sách thực đơn

- Tính định lượng từng thực phẩm trong mỗi thực đơn dựa vào tổng calo cần thiết.

- Xếp hạng cho các thực đơn