Hexenkessel Auf.4

USER-CASE-DIAGRAMM

CLIENT

- start
- create name
- create description
- check recipe
- recive confirmation
- send recipe

USER

SERVER

- change ingridients
- change price
- send confirmation
- recive recipe

PROF.

`<input> type="text"`

Name: [ ]

`<select>`

Effect: [▼]

Colour: [▼]

`<input> type="raye"`

Dauer: |———●———|
30m   4h

Temp. |——●——|
0°   60°

Zutaten:

`<select>`

`<input> type="radio"`

☒ ～～～ [1] □   □ ～～
□ ～～～   ☒ ～～～ [1] □ ～～
□ ～～～   □ ～～

Description: [...I]

`<input> type="text"`

`<h1>`

`<div> id=recipe`

Rezept:
～～～～
～～   . . .

12 $

`<div> id="price"`

Activity Diagramm:

● → install load listener

load → handle load 门 → ◉

click button Anweisung → display Anweisung 门 → ◉

click button BasicInfo → displayInfo → ◉

click button send → send recipe → ◉

handleload

generate Content 门 → find div BasicInfo, install click-listener → find div Anweisung install click-listener → ◉

displayInfo [_event: click]

delete previous data → data = get data from BasicInfo → print paragraph → ◉

ingridients
name: string
price: number

Data
[paragraph: string] []Ingredient

send recipe

recipe = get data from form → query = create query from recipe → send query to server → ▬ → ◉

wait for response → inform user about notification → ◉

generateContent [_data: Data]

append Child to div → ◉

paragraph: string = next key
ingredients: Ingredients [] nextIngredientList

group = create Select () from Ingredients 门 → div = get div with id = Anweisung

displayAnweisung

_event:click

data = get data from Anweisung

print paragraph

get price of ingredients

calculate total

print total

createSelect

_ingredients: ingredients [ ] → group=newDiv

group:div

ingredient = next ingredient
dropdown = new option

set option price
value = ingredient.name

append option
to group

handleLoad

request ingredients

wait for response → response → extract ingr. from response

ingredients ← wait for ingr. ←

data = passed Zutaten
(JSON)