

FlowerPower

Front-end websida – Systemdokumentation

Abstract

Det här dokumentet beskriver den behövda funktionerna och gränssnitten av front-end websidan.

Version History

Date	Version	Author	Description
28/05/2018	<Version1>	<Kim Säther>	Skapat teknisk dokumentation för front-end websida



Innehåll

1	Introduktion.....	3
1.1	Dokumentets Syfte.....	3
1.2	Dokumentomfattning	3
1.3	Dokument Översikt.....	3
2	Obligatoriska Funktioner	Error! Bookmark not defined.
3	Implementeringsbegränsningar	4
4	Tillhandahållna Gränssnitt	5
5	Obligatoriska Granssnitt	5
6	Intern Struktur	5
	Referenser	6

1 Introduktion

1.1 Dokumentets Syfte

Syftet med detta dokument är att beskriva beteendet och gränssnittet för komponenten <Front-end websida> för att:

- Fastställa beroenden som finns med andra komponenter
- Stödja den oberoende utvecklingen av denna komponent
- Ge en grund för att testa att komponenten fungerar efter behov.

1.2 Dokumentomfattning

Omfattningen av detta dokument är begränsat till:

- Specifikationen av de nödvändiga funktionerna för denna komponent
- Specifikationen av gränssnitten som denna komponent måste genomföra
- Listar de gränssnitt som krävs för denna komponent
- Ange komponentens interna struktur och design.

Dokumentet omfattar inte följande:

- Enhetstester som krävs för att verifiera att denna komponent överensstämmer och utför det som är specificerat
- Specifikationen eller genomförandet av andra komponenter som är beroende av denna komponent eller denna komponent beror på.

1.3 Dokument Översikt

Det här dokumentet innehåller följande sektioner:

- **Obligatoriska Funktioner** - funktioner som krävs av komponenten när det gäller de ansvarsområden som den måste uppfylla och relaterade krav.
- **Implementeringsbegränsningar** - begränsningar som implementeringsmiljö, genomförandespråk, relaterade mönster, riktlinjer och standarder.
- **Tillhandahållna gränssnitt** - de gränssnitt som komponenten måste tillhandahålla specificerad med avseende på deras signaturer och begränsningar.
- **Obligatoriska gränssnitt** - listar de gränssnitt som komponenten beror på.
- **Intern struktur** - specificerar komponentens interna designstruktur med avseende på dess beståndsdelar och deras relationer.
- **Referenser** - ger fullständiga referensuppgifter för alla dokument, böcker och länkar som refereras av detta dokument.

2 Obligatoriska Funktioner

I det här avsnittet anges de funktioner som krävs för komponenten i termer av:

- De beteende eller tjänster som den måste tillhandahålla för sina kunder
- De icke-funktionella kraven som är relaterade till var och en av dessa
- Övriga krav på underhåll, dokumentation, återanvändning, kontroll eller loggning som gäller komponenten.

Dessa funktionella funktioner tillhandahåller websidan:

- Visa blommans fuktvärde
- Ange fuktkänslighetsvärde för notis
- Vattna blomman

Dessa icke-funktionella funktioner tillhandahåller websidan:

- Logga in
- Logga ut

Koden behöver inte underhållas annat än vid genomförandet av ändringar. Det finns inget krav på dokumentation av koden som skrivs i språket HTML eller CSS, är det däremot något som är otydligt så är det ändå önskvärt med en kommentar. Det får inte finnas någon duplicerad kod i systemet. Systemet ska också vara säkert med kontroller på sådant som kan skada användaren eller systemet, exempelvis inmatningen av användaruppgifter ska kontrolleras och krypteras. Systemet har inget som behöver loggas i nuläget, men det kan komma att byggas till senare.

3 Implementeringsbegränsningar

Detta avsnitt anger de begränsningar som påverkar komponentens genomförande, såsom:

- Utvecklingsmiljöerna inom vilka den måste fungera
- Vilket språk det ska skrivas i
- De ramverk den ska fungera inom
- Designmönster den ska följa
- Riktlinjer som ska följas, inklusive modelleringsriktlinjer, kodningsriktlinjer och standarder
- Miljömässiga begränsningar som minnesfotavtryck och andra resursanvändningsbegränsningar.

Den använda utvecklingsmiljön är Visual Studio Code men det finns ingen anledning att begränsa vilken ide som ska användas, så vill man fortsätta utveckla systemet så kan man välja ide efter egen smak och tycke. Språket som används för hemsidan är HTML och det grafiska är med CSS. Sedan används även en fil som är gränssnittet mot back-end websidan och den är skriven i python. Den filen ingår inte i front-end websidan, men den nämns här för att det kan vara bra att ha koll på den om man

jobbar med front-end websidan. Front-end websidan ska fungera i ramverket flask och ska tillsammans med back-end websidan följa designmönstret MVC. Riktlinjer som bör följas är att koden bör vara inkapslad, ha hög sammanhållning och låg koppling [1]. Front-end websidan är inte utvecklad på ett miljövänligt sätt då systemet måste ladda all information på nytt varje gång och inte bara det som är nytt på sidan. Detta är något att jobba med framöver om man vill fortsätta utveckla produkten.

4 Tillhandahållna Gränssnitt

I det här avsnittet anges de gränssnitt som komponenten måste tillhandahålla när det gäller deras signaturer och begränsningar.

Grafiskt gränssnitt och gränssnitt mot back-end websidan tillhandahålls av front-end websidan. Det grafiska är det som användaren ser och det är designat med avseende på dem 10 användbarhets heuristikerna för användargränssnitt [2]. Det som gränsar till back-end är skapat av back-end utvecklaren som kommunicerade med front-end utvecklaren för att få dem olika delarna av systemet att fungera tillsammans.

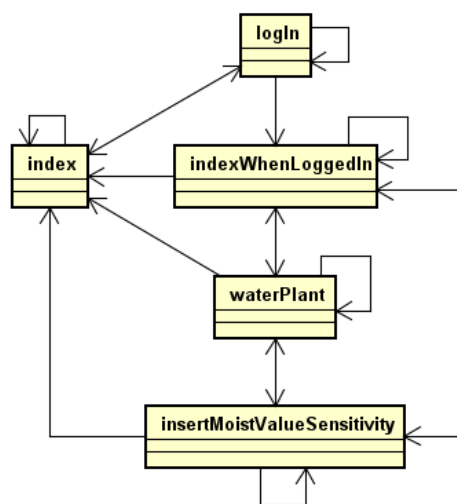
5 Obligatoriska Gränssnitt

I det här avsnittet anges de gränssnitt som komponenten beror på för att uppfylla sitt angivna beteende.

För att alla funktioner ska fungera krävs det att hela systemet är på plats. Det innebär att alla fem komponenter, front-end websida, backend, externals controller, external integration och arduino kommunicerar och att en användare använder det grafiska gränssnittet för att styra systemet.

6 Intern Struktur

Det här avsnittet anges komponentens interna designstruktur i termer av dess ingående komponenter och hur dessa samverkar för att leverera dem nödvändiga komponentbeteenden.



När front-end websidan öppnas så visas först index-sidan, sedan kan man klicka sig vidare till login-sidan. Från login-sidan kan man antingen navigera tillbaka till index-sidan eller till indexWhenLoggedIn-sidan. Därifrån kan man navigera sig till sidorna waterPlant, insertMoistValueSensitivity eller tillbaka till index. När man sedan står på waterPlant eller insertMoistValueSensitivity så kan man navigera fritt mellan dem eller till indexWhenLoggedIn eller index, se figur 1.

Med denna design får man ett användarvänligt och lättanvänt system, det är lätt att veta var man ska navigera utifrån vad man vill göra.

Figur 1. UML-diagram över komponent

Referenser

Använd det här avsnittet för att ge fullständiga referensuppgifter för alla dokument, böcker och länkar som refereras av detta dokument.

- [1] Leif Lindbäck 4 May 2018. Kapitel 5.2 Design Concepts.
A First Course in Object Oriented Development A Hands-On Approach
KTH, Kungliga Tekniska Högskolan
- [2] Jakob Nielsen 1 Januari 1995. 10 Usability Heuristics for User Interface Design
<https://www.nngroup.com/articles/ten-usability-heuristics/>
NN/g, Nielsen Norman Group