

Relazione progetto

SMART LAUNDRY - SDD

CORSO DI LAUREA IN INFORMATICA

Ingegneria del software – A.A. 2022/2023

Componenti del gruppo:

Simone Palladino - 0124002316

Luca Tartaglia - 0124002294

Mattia Di Palma - 0124002448

Sommario

Introduzione.....	3
Scopo del sistema	4
Obiettivi di progettazione.....	5
Definizioni, acronimi e abbreviazioni	5
Riferimenti	6
Sistema corrente	7
Sistema proposto	8
Decomposizione del sistema	9
Hardware/Software mapping	10
Gestione dei dati persistenti.....	11
Controllo accessi e sicurezza	12
Decisioni sul flusso di controllo globale.....	12
Condizioni limite	13
Servizi del sottosistema	14
Glossario.....	15

Introduzione

Il software da noi implementato riguarda un unico elettrodomestico che comprende quattro diversi dispositivi:

- Lavaggio bucato
- Asciugatura bucato
- Deumidificazione area ambiente (esterna all'elettrodomestico)
- Riscaldamento aria ambiente (esterna all'elettrodomestico)

tutte le funzioni devono essere gestite da un unico software in grado di:

- Permettere la gestione singola di ogni funzione
- Permettere la gestione automatica che andrà a controllare più funzioni attivate (due, tre o tutte e quattro)
- La gestione automatica deve gestire il funzionamento contemporaneo di più funzioni armonizzandole al fine di evitare il superamento di una certa soglia di assorbimento di energia elettrica impostata dall'utente, ad esempio 2300 watt/ora
- Permettere di impostare le date di fine lavaggio e fine asciugatura di due diversi carichi di bucato, messi l'uno a lavare e l'altro ad asciugare contemporaneamente.
- Gestire le attività contemporanee di lavaggio e asciugatura rispettando le date richieste dall'utente, con il minor consumo di energia.
- Valutare se le date richieste dall'utente sono realizzabili, in caso contrario, non accettare tali date ma proporre la data o le date più vicine a quelle richieste.
- Indicare per ogni soluzione scelta il consumo totale di energia previsto

Per compiere i calcoli necessari il software dovrà poter accedere a un certo numero di dati, alcuni inseriti dall'utente in fase di installazione dell'elettrodomestico:

Dati geografici e altimetrici:

- Regione (es. Lombardia)
- Altitudine (es. montagna, collina, mare)

Dati ambientali:

- Misure del locale ove è collocato l'elettrodomestico (cm. 300 x 200 x 270), utili per calcolare la quantità di aria disponibile.
- Presenza di aperture nel locale : numero porte, numero finestre.
- Indicazione di dove accedono queste aperture (interno casa oppure esterno casa).

- Se e quali di queste aperture vanno considerate chiuse e quali aperte (questo dato può essere modificato secondo la stagione)

Dati temporali:

- Ora del giorno/giorno del mese/mese dell'anno/ anno

Dati acquisiti da sensori:

- Temperatura aria ambiente
- Umidità relativa aria ambiente
- Tipologia e peso del bucato , sia di quello ad asciugare e sia di quello a lavare

Dopo la premessa riguardante le problematiche che il software deve saper affrontare, iniziamo con l'analisi introduttiva delle figure attoriali da noi riconosciute con le relative responsabilità.

User: colui che si occupa della configurazione iniziale in fase di installazione del software, inserendo i **dati geografici/altimetrici, dati ambientali e dati temporali**.

Si occupa inoltre dell'inserimento delle funzioni specifiche da eseguire con le relative date di terminazione e le soglie di assorbimento di energia elettrica.

SmartController: esso si occupa di valutare il corretto inserimento dei dati da parte dell'utente e nel caso di un eventuale errore, proporre una soluzione quanto più vicina a quella richiesta(ES. Date inserite non coerenti e/o soglie di assorbimento non supportate). Al fine di migliorare l'esperienza lato utente il software tramite il CONTROLLER si occupa anche dell'ottimizzazione dei consumi indicando per ogni soluzione/funzione l'utilizzo totale di energia previsto.

Scheduler: si occupa della parallelizzazione delle funzioni attive al fine di armonizzarle per ottimizzare i consumi di energia elettrica, rispettando le richieste inserite in input(ES. date terminazione lavoro) gestite dallo SMARTCONTROLLER.

Sensore: quest'ultimo attore sfrutta i dati inseriti in fase di installazione al fine di ricavare informazioni utili come il peso del bucato, l'umidità e la temperatura dell'aria.

Scopo del sistema

Nel mondo moderno, indipendentemente dall'utilizzo privato o per un fine commerciale, avere sotto controllo la catena dei processi/azioni quotidiane tramite un software è di fondamentale importanza.

Lo scopo del nostro sistema è quello di automatizzare e controllare i processi svolti dall'utente per l'utilizzo di un elettrodomestico con più funzionalità.

Tutto questo viene implementato al fine di velocizzare la produzione/azione dell'utente monitorando inoltre i vari tempi di esecuzione e il consumo della macchina. **Gli attori che si occupano di queste azioni saranno successivamente analizzati nel dettaglio.**

Obiettivi di progettazione

Gli obiettivi che ci siamo posti sono :

- **Response time:** il software dovrebbe fornire soluzioni in tempi che si avvicinano il più possibile alle richieste degli utenti
- **Lower costs:** dovrebbe far risparmiare l'utente tramite il monitoraggio dei consumi
- **Portability:** dovrebbe essere un sistema multiplatforma.
- **Extensibility:** dovrebbe essere tale da permettere l'aggiunta di nuove funzionalità senza dover modificare l'intero sistema.
- **Availability:** dovrebbe garantire, per quanto possibile, il funzionamento del servizio e nel caso di guasti avvertire l'utente con adeguate segnalazioni
- **Accessibility:** dovrebbe garantire un accesso facile e semplice per tutti attraverso ad esempio delle interfacce responsive che rispettano i canoni di progettazione moderna.

Definizioni, acronimi e abbreviazioni

WashCycle: È una classe che inizializza il ciclo di lavaggio.

DryCycle: È una classe che inizializza il ciclo di asciugatura.

Environment: È una classe che rappresenta l'ambiente in cui si trova l'elettrodomestico.

CheckResult: È un' interfaccia che contiene il metodo check per il controllo di situazioni atipiche, come ad esempio il superamento della soglia di energia impostata tramite l'ausilio dei sensori.

LaundryCycle: È una classe astratta che viene estesa da DryCycle e WashCycle che contiene i metodi astratti per startare, stoppare, mettere in pausa e riprendere un ciclo.

InternalSensor: È una classe astratta che contiene i metodi per calcolare il consumo di energia, eseguire tecniche di ottimizzazione attraverso l'ausilio della classe Scheduler e calcolare il peso del bucato ecc.

Riferimenti

[https://studentiuniparthenope.sharepoint.com/:f:/r/sites/IngegneriadelSoftware498/Materiale%20del%20corso/Ingegneria%20del%20software%20\(Barra\)?csf=1&web=1&e=ETpwwd](https://studentiuniparthenope.sharepoint.com/:f:/r/sites/IngegneriadelSoftware498/Materiale%20del%20corso/Ingegneria%20del%20software%20(Barra)?csf=1&web=1&e=ETpwwd)

[https://studentiuniparthenope.sharepoint.com/:f:/r/sites/IngegneriadelSoftware498/Materiale%20del%20corso/Interazione%20uomo-macchina%20\(Staffa\)?csf=1&web=1&e=buxq4c](https://studentiuniparthenope.sharepoint.com/:f:/r/sites/IngegneriadelSoftware498/Materiale%20del%20corso/Interazione%20uomo-macchina%20(Staffa)?csf=1&web=1&e=buxq4c)

Sistema corrente

La scoperta dei requisiti e il conseguente sviluppo del sistema introdotto parte da zero e non si basa su sistemi precedentemente esistenti. Si parla quindi di Greenfield Engineering, ovvero:

- Il progetto viene sviluppato da una base vuota (“blank”).
- Nessun sistema precedente viene utilizzato per lo sviluppo.
- I requisiti vengono estratti dagli utenti.

Inizialmente l'elettrodomestico è composto da quattro dispositivi distinti, ognuno dei quali ha la propria interfaccia utente e funziona in modo indipendente dagli altri. Non esiste ancora un software centrale in grado di gestire le diverse funzioni in contemporanea o di ottimizzare il consumo energetico. L'utente deve gestire manualmente ciascuna funzione e verificare costantemente il consumo di energia per evitare di superare la soglia impostata dall'elettrodomestico.

Sistema proposto

Ora si ha un nuovo elettrodomestico che è composto da quattro dispositivi integrati in un unico sistema. Un software centrale chiamato LaundrySmart è anche in grado di gestire più funzioni contemporaneamente. Inoltre, il software è in grado di monitorare il consumo energetico una volta che l'utente ha impostato i parametri iniziali. Inoltre, il software è in grado di verificare se i parametri inseriti dall'utente sono esatti e in caso contrario proporre dei parametri coerenti. Il software ha accesso a diversi dati ambientali e temporali tramite sensori e può adattarsi alle situazioni atipiche per garantire sempre prestazioni migliori.

The UML class diagram illustrates the architecture of a smart home system, organized into several functional packages:

- Smart Home Interface:** Contains the `UserInterface` class, which serves as the primary interaction point for users, handling commands and status updates.
- Smart Home Core:** The central logic layer, including:
 - `LaundrySmart`: Manages laundry cycles, interacting with the `UserInterface` and `SmartController`.
 - `SmartController`: The main control unit, coordinating various smart home components like the `Scheduler`, `EnergyMonitor`, and `WeatherMonitor`.
 - `Scheduler`: Manages and schedules smart home activities.
 - `EnergyMonitor`: Tracks energy consumption across different parts of the house.
 - `WeatherMonitor`: Interfaces with the `WeatherSensor` to provide weather-related data.
- Smart Home Actuators:** Components that execute commands, such as:
 - `Heater`, `WashingProgram`, `DryingProgram`, and `Humidifier`: All inherit from the `LaundryAppliance` base class.
 - `Door`, `Opening`, and `Window`: Represent physical smart home features.
 - `DryCommand`, `WashCommand`, and `SchedulerCommand`: Specific command objects for scheduling and controlling appliances.
- Smart Home Sensors:** Components that collect data from the environment:
 - `TemperatureObserver` and `Observer`: Part of an observation pattern for monitoring temperature.
 - `WeatherSensor`: Provides real-time weather data to the `WeatherMonitor`.
 - `Sensor`: A base class for various smart home sensors.
- Smart Home Strategies:** A collection of algorithms for energy and weight calculations:
 - `SyntheticEnergyCalculationStrategy`, `CottonEnergyCalculationStrategy`, `DelicateEnergyCalculationStrategy`, and `CottonWeightCalculationStrategy` are all part of the `EnergyCalculationStrategy` hierarchy.
 - `DelicateWeightCalculationStrategy` and `SyntheticWeightCalculationStrategy` are part of the `WeightCalculationStrategy` hierarchy.

The diagram uses standard UML notation, including solid lines for associations, dashed lines for dependencies, and hollow triangle arrows for generalization (inheritance). Packages are represented by rectangles with dashed borders.

Hardware/Software mapping

Il sistema utilizza un architettura client/server. Il collegamento tra i dispositivi e la comunicazione tra di essi richiede quindi una connessione ad internet. Il linguaggio di programmazione utilizzato per lo sviluppo del programma è principalmente Java. Può essere inoltre implementato anche come applicazione multi piattaforma.

Gestione dei dati persistenti

La gestione dei dati persistenti è una parte fondamentale del sistema poiché i dati geografici e altimetrici devono essere memorizzati in modo che possano essere utilizzati per le varie misurazioni. Inoltre, le informazioni sulle aperture del locale, come il numero di porte e finestre, devono essere memorizzate in modo che il software possa sapere quali spazi considerare aperti o chiusi in base alla stagione. Inoltre, i dati sui carichi di bucato devono essere conservati, in modo che il software possa tenere traccia di quali di questi sono stati già lavati o asciugati e quali sono in coda per essere gestiti.

I dati ambientali devono essere conservati in modo che il software possa di conseguenza gestirli per un corretto funzionamento dell'elettrodomestico. In questo caso i dati vengono memorizzati nelle classi, ad esempio `environment` o `smartcontroller`, in attributi specifici.

Controllo accessi e sicurezza

Il software deve essere sia robusto che sicuro, in modo da proteggere l'elettrodomestico da possibili attacchi informatici. È necessario quindi includere l'utilizzo di pratiche di sviluppo sicure, il software inoltre deve essere necessariamente sempre aggiornato.

Inoltre, il software deve essere in grado di controllare il superamento della soglia di assorbimento di energia elettrica.

Decisioni sul flusso di controllo globale

Le decisioni sul flusso di controllo globale del progetto riguardano la gestione del software nel suo insieme e come esso gestisce il flusso di esecuzione del programma. Le principali decisioni riguardano:

1. il software deve essere in grado di gestire le quattro funzioni dell'elettrodomestico in modo singolo o automatico.
2. il software deve essere in grado di gestire le attività di lavaggio e asciugatura contemporaneamente
3. il software deve valutare se le date richieste dall'utente sono realizzabili. In caso contrario, il software non accetta le date richieste ma propone la data o le date più vicine a quelle richieste.
4. il software deve essere in grado di accedere a un certo numero di dati, come i dati geografici e altimetrici, i dati ambientali, i dati temporali e i dati del bucato, tramite sensori o inseriti dall'utente in fase di installazione dell'elettrodomestico.

Queste decisioni sono importanti perché influenzano il flusso del software generale.

Condizioni limite

Sovraccarico elettrico: se l'utente imposta un valore di soglia per l'assorbimento di energia elettrica troppo basso, potrebbe verificarsi il superamento di tale soglia, con conseguente disattivazione dell'energia. Il software deve essere in grado di gestire in modo adeguato l'energia elettrica.

Malfunzionamento dell'elettrodomestico: il software deve essere in grado di rilevare eventuali malfunzionamenti dell'elettrodomestico e segnalare all'utente eventuali problemi.

Condizioni ambientali estreme: il software dovrebbe essere in grado di gestire situazioni ambientali estreme, come temperature o umidità eccessive o venti forti, che potrebbero influenzare il funzionamento del prodotto.

Dati di input non coerenti: il software dovrebbe essere in grado di verificare che i dati di input inseriti dall'utente siano coerenti con le funzioni attivate. In caso contrario, il software deve essere in grado di suggerire all'utente dati coerenti.

Servizi del sottosistema

Il sistema in questo specifico caso è composto da 3 sottosistemi, ognuno dei quali svolge una funzione ben precisa.

Il sottosistema Utente permette di gestire tutta la parte relativa all'inserimento dei parametri che serviranno poi successivamente per avviare un determinato ciclo.

Il sottosistema SmartController è la parte più importante del sistema poiché consente di gestire i parametri impostati precedentemente nella fase iniziale dall'utente e inoltre permette la gestione di più funzioni in contemporanea attraverso lo scheduler.

Il sottosistema Sensori permette di gestire tutta la parte relativa ai sensori, una parte importantissima poiché consente di calcolare il peso del bucato, determinare la strategia anche in base alla soglia massima di energia elettrica e inoltre permette di gestire i dati ambientali.

Glossario

User: utente che usufruisce del sistema.

SmartController: gestore di elettrodomestici

Scheduler: gestore di funzioni simultanee

SmartLaundry: è il software che gestisce le funzioni