

Documento dei requisiti

Progettazione di software gestionale per

.....

1. Premesse del progetto	3
1.1. Obiettivi e scopo del prodotto	3
1.2. Contesto di business	3
1.3. Stakeholders	3
2. Servizi del sistema	4
2.1. Requisiti funzionali	4
2.2. Requisiti informativi	6
3. Vincoli di sistema	8
3.1. Requisiti di interfaccia	8
3.2. Requisiti di prestazione	8
3.3. Requisiti di sicurezza	8
3.4. Requisiti operativi	8
3.5. Requisiti politici e legali	8
3.6. Altri vincoli	9
4. Aspetti progettuali	9
4.1. Problemi aperti	9
4.2. Programma preliminare	10
4.3. Previsione costi	10
Appendici	11

1. Premesse del progetto

1.1. Obiettivi e scopo del prodotto

Qual è lo scopo/la ragione per cui si vuol sviluppare il prodotto SW.

ESEMPIO

Il prodotto è appositamente studiato per rispondere alle esigenze delle persone anziane che, al tempo del coronavirus, vivono da sole e non possono uscire di casa per fare la spesa (perché malate o perché non vogliono ammalarsi).

1.2. Contesto di business

Qual è il contesto, la situazione in cui si vuol introdurre il prodotto: come vengono svolte le operazioni che si vogliono informatizzare, in che ambiente si vuol introdurre/vendere il prodotto

Si è riscontrata difficoltà nel procurarsi i viveri da parte delle persone anziane in questa situazione di emergenza. Si è deciso pertanto di sviluppare un'applicazione che permette alle persone che si rendono disponibili di aiutare gli anziani.

1.3. Stakeholders

Quali sono le figure, i committenti, gli utilizzatori, ... che sono a vario titolo interessati a modificare/realizzare/utilizzare le funzionalità che il prodotto deve offrire.

Le figure che influenzano lo sviluppo del sistema sono:

- Developers (Analisti, progettisti).
- Persone anziane.
- Lavoratori disponibili

2. Servizi del sistema

Si tratta della parte di documento nella quale, sotto forma di requisiti, si illustrano tutte le funzionalità offerte dal sistema, ovvero i compiti che il sistema deve realizzare. I requisiti funzionali definiscono le attività che devono essere svolte dal sistema, mentre i requisiti informativi determinano quali sono le informazioni che devono essere modellate.

2.1. Requisiti funzionali

- Il lavoratore fornisce la disponibilità per la spesa al sistema
- Il lavoratore può rimuovere la disponibilità dal sistema
- L'anziano richiede una prestazione al sistema
- L'anziano può modificare o annullare una richiesta dal sistema
- Gli anziani devono essere avvisati tramite sms o e-mail dal sistema per eventuali prestazioni non più erogabili
- I lavoratori devono essere informati tramite sms o e-mail dal sistema sull'eventuale modifica o annullamento della richiesta
- I lavoratori possono visualizzare il calendario della propria disponibilità dal sistema

2.2. Requisiti informativi

Descrizione di tutte le informazioni che dovranno essere elaborate/archivate nel prodotto per poter raggiungere lo scopo prefissato

1. Un utente generico è caratterizzato da:

- Cognome
- Nome
- Indirizzo di residenza
- Indirizzo e-mail
- Numero di telefono

2. Una pubblicazione può essere:

- Una richiesta del servizio
- Una data di disponibilità

3. Vincoli di sistema

Si tratta della parte del documento nella quale si definiscono le limitazioni da imporre al sistema nell'eseguire le proprie attività.

3.1. Requisiti di interfaccia

Definiscono in che modo il prodotto si interfaccia con l'utente; riassumono l'aspetto generale dell'interfaccia stessa in relazione alle attività che il sistema deve garantire alle varie tipologie di utenti.

ESEMPIO

L'interfaccia proposta dal programma prevede: una schermata di accesso/registrazione; per il lavoratore un calendario per la selezione delle date in cui fornire la disponibilità; per l'anziano un calendario con le date in cui è presente la disponibilità dei lavoratori.

3.2. Requisiti di prestazione

Non si registrano particolari esigenze in merito alle prestazioni.

3.3. Requisiti di sicurezza

Definiscono quelle che sono le restrizioni che gli utenti possono avere sull'accesso o l'esecuzione di alcune informazioni.

ESEMPIO

Il prodotto consente la gestione dei vari livelli di accesso alle sue funzionalità mediante l'uso di username e passwords differenziate da inserire durante la fase di login. Di seguito si riportano le diverse tipologie di accesso con relativi requisiti funzionali consentiti:

1. accesso completo: 2.1.1; 2.1.2; 2.1.3; 2.1.4; 2.1.5; 2.1.6; 2.1.7

2. accesso limitato (impossibilità di accedere ad informazioni sull'operato di colleghi): 2.1.2.3, 2.1.2.4, 2.1.2.5, 2.1.2.6, 2.1.2.7, 2.1.2.8; 2.1.3.1, 2.1.3.8, 2.1.3.9, 2.1.3.10, 2.1.3.11, 2.1.3.12; 2.1.4; 2.1.5; 2.1.6; 2.1.7.1, 2.1.7.3, 2.1.7.4

3. accesso semplice (sola consultazione): 2.1.4.9, 2.1.4.10, 2.1.4.11

....

3.4. Requisiti operativi

Definiscono l'ambiente hardware/software sul quale il sistema opera.

ESEMPIO

Visual studio -> C#

3.5. Requisiti politici e legali

3.5.1. Protezione dei dati personali e dei dati sensibili degli utenti

ESEMPIO

Il sistema software Omero© dovrà occuparsi della gestione in maniera sicura delle informazioni riguardanti il nome utente e la password di ogni singolo impiegato e del direttore, conformemente a quanto previsto dal decreto legislativo sulla privacy Dlgs 12/96.

Per tali motivazioni l'applicazione dovrà osservare i seguenti criteri di sicurezza:

- Irrecuperabilità dei dati sensibili di un utente durante l'utilizzo del software da parte di terzi
- Mascheramento a video dei dati sensibili durante la loro digitazione

Per quanto riguarda i dati personali, il sistema dovrà garantire quanto segue:

- in nessun modo un utente autenticato per un dato livello di accesso possa accedere a dati personali di altri utenti eccezion fatta per il direttore
- dovrà risultare impossibile la trasmissione in qualsiasi modo delle informazioni personali riguardanti i lettori associati alla biblioteca verso l'esterno del sistema informatico, quando questi però siano già stati introdotti in esso.

3.5.2. Licensing and Patent Policy

ESEMPIO

Il sistema dovrà corrispondere alle esigenze di licenza definite come "software semi-libero" per cui il software è corredato di codice sorgente, distribuito con permesso di uso, copia, modifica e distribuzione

per scopi limitati. Inoltre dovrà essere comprendente di documentazione di progetto che verrà fornita necessariamente sotto le prescrizioni imposte da una politica di Shared License.

3.5.3. Copyright

ESEMPIO

Il sistema non dovrà violare diritti sul copyright di nessun tipo di nessun prodotto già esistente.

Il sistema non dovrà contenere riferimenti diretti o indiretti a software già esistenti se non esplicitamente espresso dalla casa di produzione del software in oggetto. Quanto detto va esteso a tutti i marchi registrati in genere in quanto requisito legale necessario è quello di rispettare la normativa sull'uso di titoli già esistenti ed adeguatamente registrati e la normativa regolante le forme di pubblicità indiretta.

Il sistema in ogni sua singola componente non dovrà ricalcare il design grafico di nessun altro prodotto già in commercio e che goda di Author Rights.

Il Sistema può riutilizzare solo parti di architetture non proprietarie o sottostanti a License Authoring di tipo libero o semi-libero oppure Open Source.

Il sistema non dovrà riutilizzare formati già esistenti e regolarmente brevettati.

3.5.4. Distribuzione

ESEMPIO

Il sistema dovrà avere la componente applicativa che possa essere riconducibile all'interno della categoria del software con licenza proprietaria e nello specifico dovrà essere garantita l'appartenenza alla sottocategoria definita "Nessuna licenza" (No License).

4. Aspetti progettuali

4.1. Problemi aperti

4.2. Programma preliminare

4.3. Previsione costi