Esempio di documentazione

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

1.3 Scopo 3

Analisi 4

1.4 Analisi del dominio 4

1.5 Analisi e specifica dei requisiti 4

1.6 Use case 6

1.7 Pianificazione 6

1.8 Analisi dei mezzi 6

1.8.1 Software 6

1.8.2 Hardware 6

2 Progettazione 7

2.1 Design dell’architettura del sistema 7

2.2 Design dei dati e database 7

2.3 Design delle interfacce 7

2.4 Design procedurale 7

3 Implementazione 8

4 Test 8

4.1 Protocollo di test 8

4.2 Risultati test 9

4.3 Mancanze/limitazioni conosciute 9

5 Consuntivo 9

6 Conclusioni 9

6.1 Sviluppi futuri 9

6.2 Considerazioni personali 9

7 Bibliografia 9

7.1 Bibliografia per articoli di riviste: 9

7.2 Bibliografia per libri 9

7.3 Sitografia 9

8 Allegati 10

# Introduzione

## Informazioni sul progetto

In questo capitolo raccogliere le informazioni relative al progetto, ad esempio:

* Allievo/i e docente/i coinvolti nel progetto e rispettivi ruoli,
* scuola ,sezione, materia/e,
* date di inizio e termine di consegna,
* …

## Abstract

E’ una breve e accurata rappresentazione dei contenuti di un documento, senza notazioni critiche o valutazioni. Lo scopo di un abstract efficace dovrebbe essere quello di far conoscere all’utente il contenuto di base di un documento e metterlo nella condizione di decidere se risponde ai suoi interessi e se è opportuno il ricorso al documento originale.

Può contenere alcuni o tutti gli elementi seguenti:

* **Background/Situazione iniziale**
* **Descrizione del problema e motivazione**: Che problema ho cercato di risolvere? Questa sezione dovrebbe includere l'importanza del vostro lavoro, la difficoltà dell'area e l'effetto che potrebbe avere se portato a termine con successo.
* **Approccio/Metodi**: Come ho ottenuto dei progressi? Come ho risolto il problema (tecniche…)? Quale è stata l’entità del mio lavoro? Che fattori importanti controllo, ignoro o misuro?
* **Risultati**: Quale è la risposta? Quali sono i risultati? Quanto è più veloce, più sicuro, più economico o in qualche altro aspetto migliore di altri prodotti/soluzioni?

Esempio di abstract:

*As the size and complexity of today’s most modern computer chips increase, new techniques must be developed to effectively design and create Very Large Scale Integration chips quickly. For this project, a new type of hardware compiler is created. This hardware compiler will read a C++ program, and physically design a suitable microprocessor intended for running that specific program. With this new and powerful compiler, it is possible to design anything from a small adder, to a microprocessor with millions of transistors. Designing new computer chips, such as the Pentium 4, can require dozens of engineers and months of time. With the help of this compiler, a single person could design such a large-scale microprocessor in just weeks.*

## Scopo

Lo scopo del progetto (scopi didattici/scopi operativi). Dovrebbe descrivere il mandato, ma non vanno ricopiate le informazioni del quaderno dei compiti (che va invece allegato).

## Analisi

## Analisi del dominio

Il prodotto verrà utilizzato sul web, quindi esso sarà accessibile da tutti i dispositivi (sia mobile sia fissi) che dispongono di una connessione internet. La soluzione che offre questo prodotto è già disponibile in rete, esso infatti si inspira a molti di essi e rispetta le convenzioni definite da essi.

## Analisi e specifica dei requisiti

Il progetto sarà strutturato su più pagine (come richiesto nella specifica): la pagina iniziale deve contenere un pulsante che porterà alla pagina di registrazione, all’interno di essa vi sarà un form che servirà a gestire i dati per la registrazione.

I campi saranno 12

* Nome (Testo) - Obbligatorio
* Cognome (Testo) - Obbligatorio
* Data Nascita (Data) – Obbligatorio
* Via (Testo) - Obbligatorio
* No. Civico (Numerico, 3 cifre) - Obbligatorio
* Città (Testo) - Obbligatorio
* Nap (Numerico, 5 cifre) - Obbligatorio
* No. Telefono (solo cifre, spazi e/o trattini - Obbligatorio
* E-Mail (Testo + Controllo formato e-mail [testo@testo.testo](mailto:testo@testo.testo)) - Obbligatorio
* Sesso (Checkbox F/M) - Obbligatorio
* Hobby (Testo)
* Professione (Testo)

Sul fondo del form ci saranno due bottoni, uno intitolato “Cancella” che si occuperà di “pulire” il form rimuovendo tutti i dati immessi ed il tasto “Avanti”.

Tutti i campi avranno un validatore che controllerà se il dato è valido o meno. Se tutti i dati sono corretti l’utente può cliccare su il tasto “Avanti” che lo porterà ad una pagina riassuntiva per il controllo dati mentre se i validatore hanno trovato uno o più dati invalidi l’utente sarà obbligato a modificarli per registrarsi.

Sul fondo della pagina di controllo ci saranno due tasti, uno intitolato “Correggi” che permetterà di modificare i dati immessi (senza doverli reinserire tutti) ed il pulsante “Registra” che permetterà all’utente di portare a termine la registrazione. Dopo la registrazione l’utente verrà portato in una pagina home dove verranno mostrati tutti i dati della registrazione nello stesso ordine in cui sono stati inseriti.

I dati delle registrazioni verranno salvati in due file CSV, uno contenente le registrazioni odierne e l’altro che conterrà le registrazioni globali (tutte le registrazioni effettuate) con l’aggiunta di un campo che conterrà la data e l’ora della registrazione (lato server). Durante la registrazione i dati verranno scritti in parallelo sia nel file CSV giornaliero sia in quello locale. Se si effettua la prima registrazione del giorno verrà creato un nuovo file CSV giornaliero dove verranno scritti i parallelo i nuovi dati giornalieri nel nuovo file e nel file globale.

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Pagina web iniziale |
| **Priorità** | 3 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Necessita di un tasto “registrati” che porta al link di registrazione |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Pagina di registrazione tramite form |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Pagina nella quale l’utente può fare la registrazione |
| **Sotto requisiti** | |
| **001** | Form da compilare con questra struttura e con queste restrizioni:   * Nome (campo testo) - Obbligatorio * Cognome (campo testo – Obbligatorio * Data di nascita (campo data) - Obbligatorio * Numero Civico (campo numerico, massimo 3 cifre) – Obbligatorio * Città (campo testo) – Obbligatorio * Nap (campo numerico, massimo 5 cifre) – Obbligatorio * Numero di telefono (campo testo, ammessi solo cifre, spazi e/o trattini) - Obbligatorio * E-Mail (campo testo, controllo formato “testo@testo.testo”) – Obbligatorio * Genere (campo testo F/M) – Obbligatorio * Hobby (campo testo) * Professione (campo testo) |
| **002** | Pulsante cancella che permette di eliminare tutti i dati contenuti nel form |
| **003** | Pulsante avanti che permette di continuare nel percorso di registrazione |
| **004** | Vi saranno dei validatori che controlleranno i dati inseriti nei campi |
| **005** | I validatori devono assicurarsi che il dato sia consono al campo in cui è inserito |
| **006** | I validatori devono assicurarsi se il dato inserito abbia un senso logico |
| **007** | La validazione dei dati viene svolta in locale |

|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Pagina di conferma dati |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Questa pagina si occupa di mostrare all’utente tutti i dati che ha inserito all’interno del form. |
| **Sotto requisiti** | |
| **001** | Tabella che mostra all’utente tutti i dati che ha inserito all’interno del form |
| **002** | Pulsante correggi che permette alll’utente di tornare indietro al form di registrazione, esso sarà già compilato con i dati precedentemente inseriti al fine di velocizzare la loro modifica |
| **003** | Pulsante registra con il quale l’utente potrà registrarsi definitivamente |
| **004** | Al momento della registrazione i dati dell’utente verranno scritti su entrambi i file csv in parallelo |

|  |  |
| --- | --- |
| **ID: REQ-005** | |
| **Nome** | File csv per conserva dati |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | I dati contenuti in questi file vengono scritti dopo la conferma nella pagina di conferma dati |
| **Sotto requisiti** | |
| **001** | File csv chiamato “Registrazioni\_tutte.csv” che tiene in memoria tutti i dati di tutti gli utenti |
| **002** | File csv chiamato “Registrazione\_aaaa-mm-gg.csv” che contiene i dati degli utenti creati in quel determinato giorno |
| **003** | I file verranno generati automaticamente dalla pagina web senza bisogno di nessun intervento da parte del utente, quindi durante la registrazione del primo utente della giornata verrà pure creato un nuovo file csv. Gli altri utenti della giornata andranno ad aggiungere i loro dati ai file senza sovrascriverli |
| **004** | Questi file si troveranno nella cartella “Registrazioni” che è posta sotto la cartella principale del sito |
| **005** | Ci possono essere utenti doppi, non c’è nessun controllo |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| **Nome** | Pagina Home |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Mostra tutti i dati di tutti gli utenti registrati nella giornata |

## Pianificazione

|  |
| --- |
| Figura 2: Diagramma di Gantt preventivo del progetto |

## Analisi dei mezzi

### Software

SDK, librerie, tools utilizzati per la realizzazione del progetto e eventuali dipendenze.

Per la realizzazione di questo progetto verranno utilizzati i sequenti software:

* GanttProject 2.8.5
* PhpStorm 2018.1
* Webserver tramite Apache 2.4.27 e Php 7.2.10 VC15 x64 Thread Safe

### Hardware

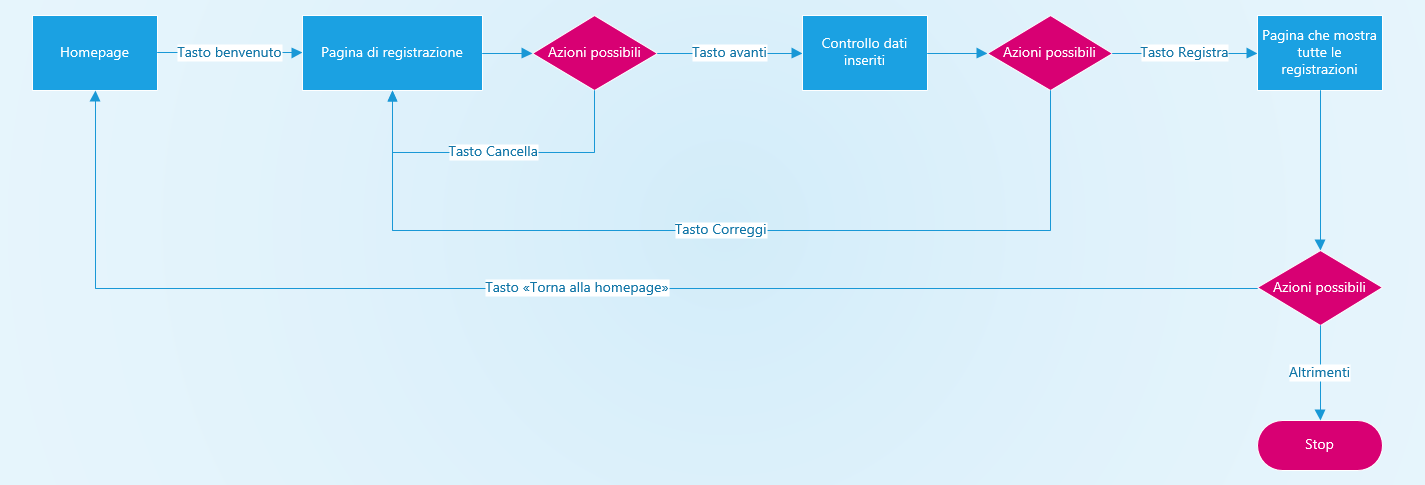
Il prodotto potrà essere utilizzabile su qualsiasi dispositivo abile nella connessione ad una pagina web.

La macchina sulla quale sarà sviluppato il prodotto avrà queste specifiche:

* GPU: Nvidia GeForce GTX 1060 6GB
* RAM: 16 GB
* CPU: Intel Core i7 7700HQ 2.8GHz

# Progettazione

## Design dell’architettura del sistema



## Design dei dati e database

I dati verrano salvati all’interno dei file csv globale con questo schema:

id,data\_registrazione,nome,cognome,data di nascita,numero civico, città,NAP, numero di telefono, email, genere, hobby, professione

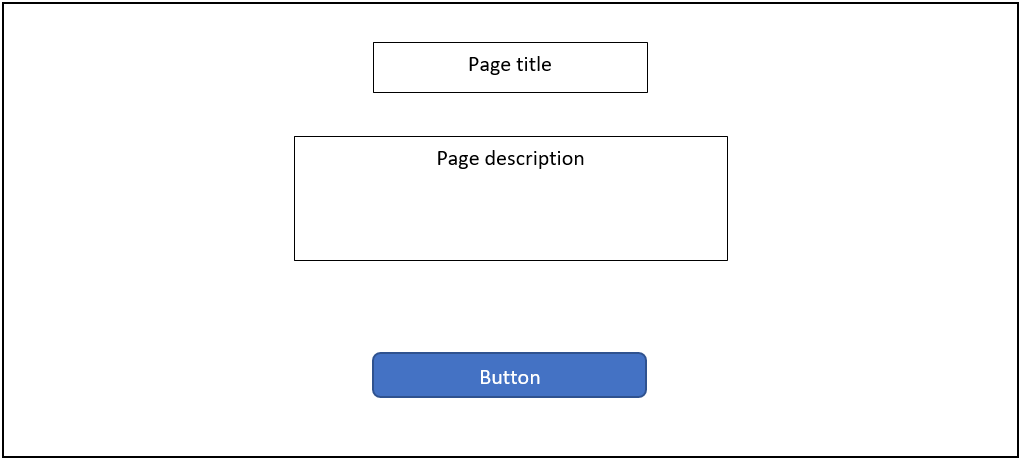
Invece i dati del file csv giornaliero verranno salvato con quest’altro schema molto simile:

id nome,cognome,data di nascita,numero civico, città,NAP, numero di telefono, email, genere, hobby, professione

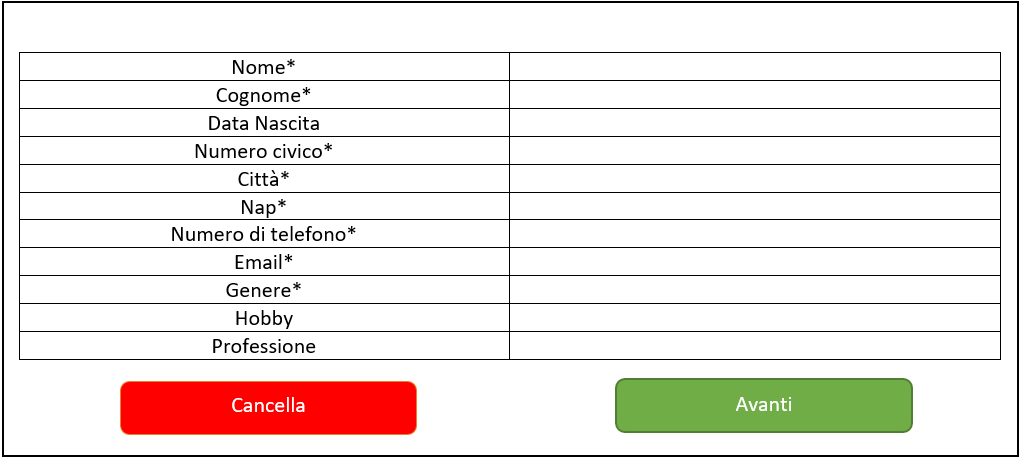
I dati all’interno del file csv saranno separati dal carattere ‘,’. Inoltre gli id dei dati salvati all’interno del file csv globale avranno il classico id incrementale e quindi non avranno nessuna relazione con gli id del file csv locale.

## Design delle interfacce

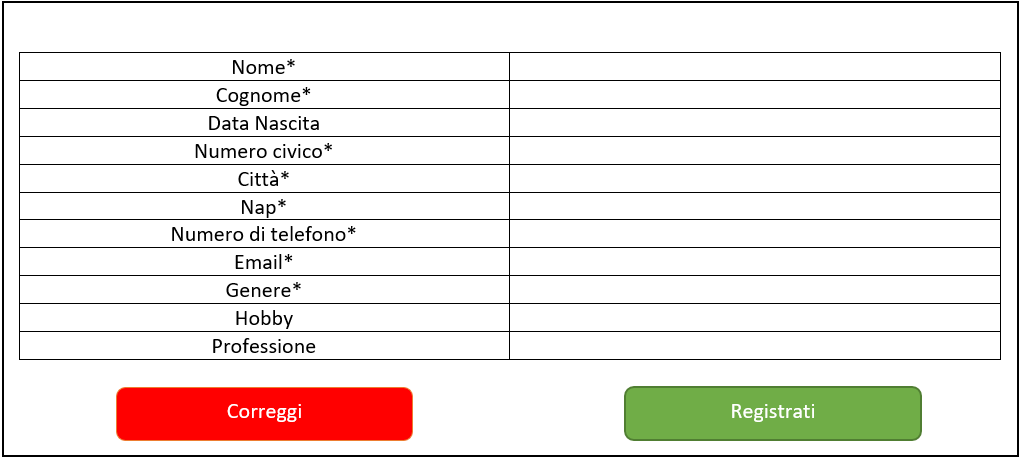
Home page:



Pagina di registrazione:



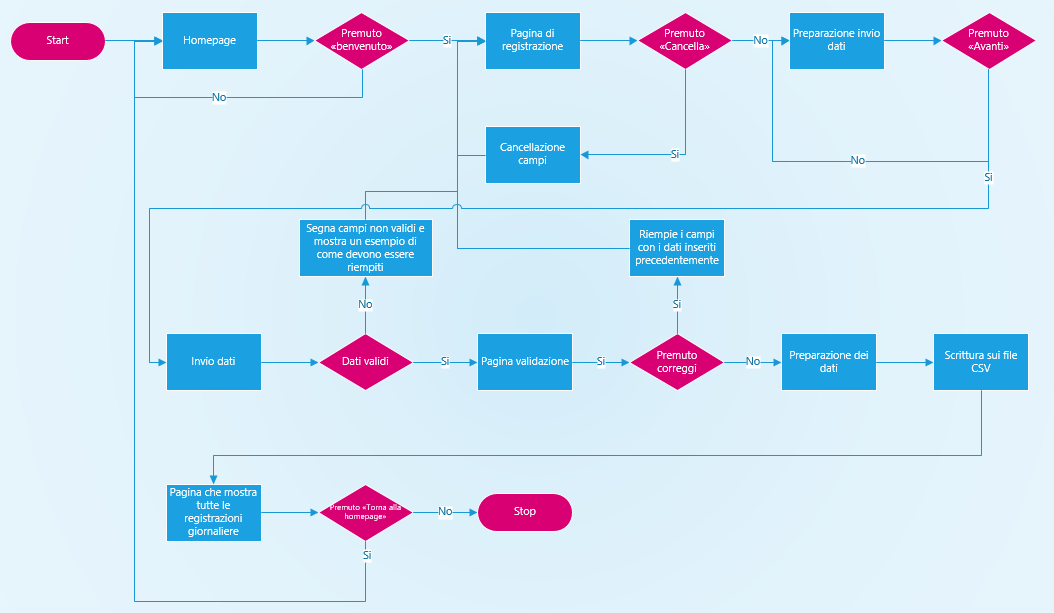
Pagina di controllo per i dati inseriti:



Pagina che mostra tutti i dati delle registrazioni giornaliere:



## Design procedurale



# Implementazione

In questo capitolo dovrà essere mostrato come è stato realizzato il lavoro. Questa parte può differenziarsi dalla progettazione in quanto il risultato ottenuto non per forza può essere come era stato progettato.

Sulla base di queste informazioni il lavoro svolto dovrà essere riproducibile.

In questa parte è richiesto l’inserimento di codice sorgente/print screen di maschere solamente per quei passaggi particolarmente significativi e/o critici.

Inoltre dovranno essere descritte eventuali varianti di soluzione o scelte di prodotti con motivazione delle scelte.

Non deve apparire nessuna forma di guida d’uso di librerie o di componenti utilizzati. Eventualmente questa va allegata.

Per eventuali dettagli si possono inserire riferimenti ai diari.

# Test

## Protocollo di test

Definire in modo accurato tutti i test che devono essere realizzati per garantire l’adempimento delle richieste formulate nei requisiti. I test fungono da garanzia di qualità del prodotto. Ogni test deve essere ripetibile alle stesse condizioni.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-012 | **Nome:** | Import a card with KIC, KID and KIK keys, but not shown with the GUI |
| **Descrizione:** | Import a card with KIC, KID and KIK keys with no obfuscation, but not shown with the GUI | | |
| **Prerequisiti:** | Store on local PC: Profile\_1.2.001.xml (appendix n\_n) and Cards\_1.2.001.txt (appendix n\_n).  PIN (OTA\_VIEW\_PIN\_PUK\_KEY) and ADM (OTA\_VIEW\_ADM\_KEY) user right not set. | | |
| **Procedura:** | 1. Go to “Cards manager” menu,  in main page click “Import Profiles” link, Select the “1.2.001.xml” file, Import the Profile 2. Go to “Cards manager” menu,  in main page click “Import Cards” link, Select the “1.2.001.txt” file, Delete the cards,  Select the “1.2.001.txt” file, Import the cards 3. Research the “41795924770” Card, Click the imsi card link Check the card details 4. Execute the SQL: SELECT imsi, dir, keyset, cntr, rawtohex(kickey), rawtohex(kidkey), rawtohex(kikkey), rawtohex(chv), rawtohex(dap)FROM otacardkey a where imsi='340041795924770' ORDER BY keyset; | | |
| **Risultati attesi:** | Keys visible in the DB (OtaCardKey) but not visible in the GUI (Card details) | | |

## Risultati test

Tabella riassuntiva in cui si inseriscono i test riusciti e non del prodotto finale. Se un test non riesce e viene corretto l’errore, questo dovrà risultare nel documento finale come riuscito (la procedura della correzione apparirà nel diario), altrimenti dovrà essere descritto l’errore con eventuali ipotesi di correzione.

## Mancanze/limitazioni conosciute

Descrizione con motivazione di eventuali elementi mancanti o non completamente implementati, al di fuori dei test case. Non devono essere riportati gli errori e i problemi riscontrati e poi risolti durante il progetto.

# Consuntivo

Consuntivo del tempo di lavoro effettivo e considerazioni riguardo le differenze rispetto alla pianificazione (cap 1.7) (ad esempio Gannt consuntivo).

# Conclusioni

Quali sono le implicazioni della mia soluzione? Che impatto avrà? Cambierà il mondo? È un successo importante? È solo un’aggiunta marginale o è semplicemente servita per scoprire che questo percorso è stato una perdita di tempo? I risultati ottenuti sono generali, facilmente generalizzabili o sono specifici di un caso particolare? ecc

## Sviluppi futuri

Migliorie o estensioni che possono essere sviluppate sul prodotto.

## Considerazioni personali

Cosa ho imparato in questo progetto? ecc

# Bibliografia

## Bibliografia per articoli di riviste:

1. Cognome e nome (o iniziali) dell’autore o degli autori, o nome dell’organizzazione,
2. Titolo dell’articolo (tra virgolette),
3. Titolo della rivista (in italico),
4. Anno e numero
5. Pagina iniziale dell’articolo,

## Bibliografia per libri

1. Cognome e nome (o iniziali) dell’autore o degli autori, o nome dell’organizzazione,
2. Titolo del libro (in italico),
3. ev. Numero di edizione,
4. Nome dell’editore,
5. Anno di pubblicazione,
6. ISBN.

## Sitografia

1. URL del sito (se troppo lungo solo dominio, evt completo nel diario),
2. Eventuale titolo della pagina (in italico),
3. Data di consultazione (GG-MM-AAAA).

**Esempio:**

* http://standards.ieee.org/guides/style/section7.html, *IEEE Standards Style Manual*, 07-06-2008.

# Allegati

Elenco degli allegati, esempio:

* Diari di lavoro
* Codici sorgente/documentazione macchine virtuali
* Istruzioni di installazione del prodotto (con credenziali di accesso) e/o di eventuali prodotti terzi
* Documentazione di prodotti di terzi
* Eventuali guide utente / Manuali di utilizzo
* Mandato e/o Qdc
* Prodotto
* …