****

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **S**ystem **D**esign  **D**ocument  Easy Pass   |  |  | | --- | --- | | Riferimento |  | | Versione | 1.0 | | Data | 29/11/2021 | | Destinatario | Prof. C. Gravino | | Presentato da | Montefusco Alberto  Mulino Martina  Rinaldi Viviana  Spina Gennaro | | Approvato da |  | |

Sommario

[Sommario 2](#_Toc530825397)

[RevisionHistory 3](#_Toc530825396)

[Responsabilità 4](#_Toc530825396)

1. [Introduzione 5](#_Toc530825398)

1.1 Scopo del Sistema 5

1.2 Design Goals 5

1.2.1 Design Trade-off 7

1.3 Definizioni, acronimi e abbreviazioni .............................................................................................. 7

1.4 Riferimenti ............................................................................................................................................ 7

1.5 Panoramica ............................................................................................................................................ 7

1. Architettura del Sistema Corrente ................................................................................................................ 7
2. Architettura del Sistema Proposto ................................................................................................................ 8

3.1 Panoramica ............................................................................................................................................ 8

3.2 Decomposizione in Sottosistemi ....................................................................................................... 9

3.3 Mapping Hardware/Software .......................................................................................................... 10

3.4 Gestione dei dati persistenti ............................................................................................................... 6

3.5 Controllo degli accessi e sicurezza ..................................................................................................... 6

3.6 Controllo del flusso globale del Software ......................................................................................... 6

3.7 Condizione limite ................................................................................................................................. 6

1. Servizi dei Sottosistemi.................................................................................................................................. 6
2. Glossario .......................................................................................................................................................... 6

RevisionHistory

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autori** |
| 19/11/2021 | 0.1 | Prima stesura: design goals, trade-off, architettura del sistema proposto. | Alberto Montefusco  Gennaro Spina  Viviana Rinaldi  Martina Mulino |
| 22/11/2021 | 0.2 | Aggiunta “Controllo degli accessi e sicurezza”.  Aggiunta “Gestione dei dati persistenti” con relativo modello concettuale, regole di vincolo e creazione del database in MySQL. | Viviana Rinaldi  Alberto Montefusco |
| 23/11/2021 | 0.3 | Aggiunta “Condizione limite” e “Gestione del flusso globale del Sistema”.  Aggiunta del modello MVC, deployment diagram, component diagram, divisione del sistema in sottosistemi e mapping Hardware/Software. | Alberto Montefusco  Viviana Rinaldi  Martina Mulino |
| 24/11/2021 | 0.4 | Aggiunta dei servizi offerti dal sottosistema. | Viviana Rinaldi |
| 29/11/2021 | 1.0 | Revisione SDD. | Alberto Montefusco  Gennaro Spina  Viviana Rinaldi  Martina Mulino |

Responsabilità

|  |  |
| --- | --- |
| **Artefatto** | **Autori** |
| Design Goals,  trade-off | Alberto Montefusco, Gennaro Spina, Viviana Rinaldi, Martina Mulino |
| Architettura del Sistema corrente e del Sistema proposto | Martina Mulino |
| Decomposizione del Sistema in sottosistemi con relativi diagrammi UML | Martina Mulino |
| Mapping Hardware/Software | Martina Mulino |
| Gestione dei dati persistenti | Alberto Montefusco |
| Controllo degli accessi e sicurezza | Viviana Rinaldi |
| Controllo del flusso globale del Software e condizione limite | Alberto Montefusco, Viviana Rinaldi |
| Servizi offerti dai sottosistemi | Viviana Rinaldi |

1. Introduzione
   1. Scopo del Sistema

--------------------------

* 1. Design Goals

Di seguito, vengono descritti i design goals che devono essere rispettati nella realizzazione del sistema. Tali obiettivi sono organizzati in cinque categorie:

1. **Performance**

* **Tempo di risposta**
* La validazione del Green Pass deve essere effettuata entro 2 secondi.
* **Throughput**
* Deve essere supportata la validazione contemporanea di almeno 20 Green Pass per sessione.
* **Memoria**
* La quantità di memoria occupata dal sistema dipende da quella necessaria al mantenimento del database.

1. **Dependability**

* **Robustezza**
* Gli input non validi inseriti dall’utente devono essere segnalati con messaggi d’errore.
* **Affidabilità**
* Il Sistema deve garantire che i Green Pass ricevuti per la validazione non siano presenti in copie.
* Il report generato deve essere consistente con le informazioni contenute nei singoli esiti.
* **Disponibilità**
* Il Sistema deve essere disponibile durante orario universitario per il Docente e H24 per il Direttore di Dipartimento.
* **Tolleranza agli errori**
* In caso di errore durante la validazione di uno o più Green Pass la sessione di validazione deve essere invalidata.
* **Sicurezza**
* Tutte le password devono essere crittografate prima di essere memorizzate.
* Il Sistema deve garantire il rispetto delle leggi sulla privacy.

1. **Cost**

* **Costi di sviluppo**
* Il costo complessivo del progetto ammonta ad un massimo di 200 ore (max 50 ore per ogni membro del team).

1. **Maintenance**

* **Estensibilità**
* Il Sistema deve essere progettato in modo tale che sia possibile aggiungere moduli su richiesta del cliente.
* **Adattabilità**
* Il Sistema deve essere progettato su una struttura generica in modo da poter essere utilizzato in altri ambiti professionali (es. aziende, convegni, ecc.).
* **Portabilità**
* Il Sistema deve essere fruibile su tutti i dispositivi mobile e desktop in maniera indipendente dal sistema operativo o dall’hardware utilizzato.
* **Tracciabilità dei requisiti**
* La tracciabilità dei requisiti deve essere garantita da una matrice di tracciabilità che permette di ricondurre ogni artefatto al proprio requisito.

1. **End User Criteria**

* **Usabilità**
* Il Sistema deve avere un’interfaccia semplice e immediata in modo da consentire un’interazione rapida e efficiente, così da ridurre i tempi di controllo.
  1. Design Trade-off

**Tempo di rilascio Vs Funzionalità**

Per rispettare le scadenze del Progetto potrebbe essere necessaria l’implementazione parziale di alcune delle funzionalità richieste.

**Usabilità Vs memoria**

Poiché uno degli obiettivi del Sistema è l’alta usabilità per gli utenti finali, è necessario memorizzare una grande porzione di dati e ciò porta ad un elevato uso della memoria del database.

**Velocità vs memoria**

Per favorire un rapido accesso ai dati risulta essere più vantaggioso utilizzare una quantità di memoria più elevata.

* 1. Definizioni, acronimi e abbreviazioni

In questa sezione descriveremo i termini che sono stati utilizzati all’interno del documento stesso divisi in tre sezioni principali: definizioni, acronimi ed abbreviazioni.

1. **Definizioni**:

* **COVID-19**: nome di una malattia infettiva espiratoria causata dal virus denominato

SARS-CoV-2 appartenente alla famiglia dei coronavirus.

* **Green Pass**: nome con il quale indichiamo un certificato interoperabile all'interno dell'Unione europea, contenente le informazioni che attestano che il titolare è stato vaccinato contro la COVID-19, o ha da poco effettuato un test diagnostico per SARS-CoV-2 con risultato negativo, oppure è guarito dalla COVID-19.
* **Esito**: nome con cui indichiamo il risultato della validazione (positivo/negativo) del Green Pass.
* **Report**: nome con il quale ci si riferisce all’insieme degli esiti raccolti durante una lezione tenuta da un docente.
* **Sessione di validazione**: nome con il quale indichiamo la durata temporale in cui viene effettuata una validazione.
* **Server di validazione**: server creato appositamente per la validazione dei Green Pass tramite API fornite dal Ministero dello Salute.

1. **Acronimi**:

* **RV[n]**: Regole di Vincolo[numero]
* **VIR[n]**: Vincolo di Integrità Referenziale[numero]
* **RDBMS**: Relational Database Management System (gestore di database basati sul modello relazionale)
* **DB**: DataBase, ovvero “Base di Dati” utilizzata per memorizzare i dati persistenti
* **MySQL**: è un RDBMS basato sul linguaggio SQL, composto da un client a riga di comando e un server.
* **MVC**: Model View Controller, ovvero un pattern architetturale utilizzato per costruire una Web Application.

1. **Abbreviazioni**:

* **GP**: Green Pass
* **Direttore**: si intende il Direttore di Dipartimento
  1. Riferimenti

Per stilare la presente documentazione, si è preso come riferimento le slide fornite dal docente del corso di Ingegneria del Software, Carmine Gravino, inserite nella sezione “M3” della piattaforma di e-learning della facoltà di Informatica. Inoltre, è stato consultato il libro di testo “Object-Oriented Software Engineering Using UML, Patterns and Java: Third Edition, di Bernd Bruegge ed Allen H. Dutoit”.

* 1. Panoramica

Dopo questa prima sezione di introduzione del presente documento, il punto 2 descriverà brevemente l’architettura del Sistema corrente, mentre al punto 3 verrà fornita una dettagliata descrizione dell’architettura del Sistema proposto. In particolare, questa sezione descriverà la decomposizione in sottosistemi, la corrispondenza tra hardware e software, la gestione dei dati persistenti, il controllo degli accessi, la sicurezza, il flusso di controllo e la gestione delle condizioni limite del sistema proposto. Infine, il punto 4 descriverà i servizi offerti dai sottosistemi individuati.

1. Architettura del Sistema corrente

---------------------

1. Architettura del Sistema proposto

Il Sistema proposto è una Web Application che vuole facilitare, velocizzare e rendere più sicuro il controllo dei Green Pass degli studenti, effettuato dai docenti in aula. Poiché l’usabilità è uno degli obiettivi principali del Sistema, è d’obbligo fornire una facile interazione con il Sistema. Pertanto, la scelta dell’architettura è ricaduta sul modello MVC, solitamente selezionato per la realizzazione di sistemi interattivi.

Questo pattern architetturale si compone di tre layer:

1. **Model**: si occupa di mantenere le informazioni del dominio applicativo;
2. **View**: permette la visualizzazione delle informazioni fornite dal Model;
3. **Controller**: è responsabile delle interazioni con l’utente, recuperando i dati dal Model mandandoli alle Views per la visualizzazione.



* 1. Panoramica

Di seguito, vengono illustrati i risultati della fase di progettazione del Sistema proposto. In particolare, viene descritta l’architettura del sistema, stabilita in base alla sua struttura da sito web e alle funzionalità che deve offrire. Particolare attenzione è stata riposta nella gestione dei dati persistenti, i quali risultano fondamentali allo scopo del sistema proposto.

* 1. Decomposizione in Sottosistemi

Il Sistema si compone in totale di 14 componenti, suddivisi tra i layer View, Model e Controller.

Il **Controller** contiene le componenti:

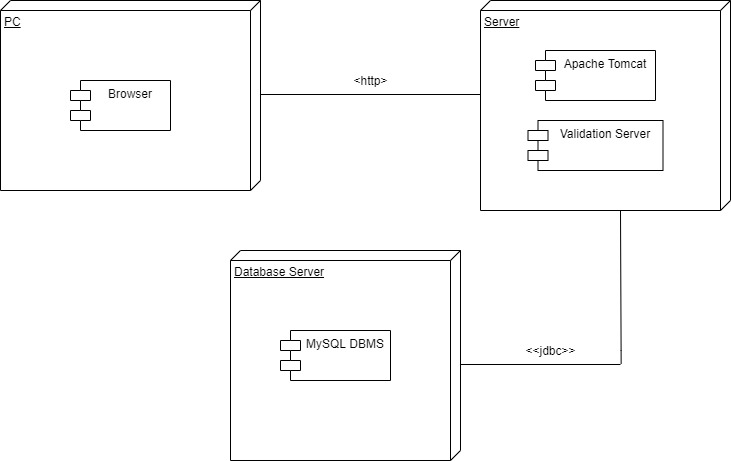
* **AccessController**: si occupa di tutte le operazioni che riguardano il controllo degli accessi degli utenti che utilizzano il Sistema, come ad esempio il login del Direttore e del Docente.
* **SessionController**: si occupa di tutte le operazioni che concernono lo svolgimento di una sessione di validazione: dall’avvio della sessione, alla ricezione e validazione dei Green Pass, fino alla terminazione.
* **ReportController**: si occupa di gestire tutte le operazioni che possono essere effettuate dal Direttore di Dipartimento sui report, quindi ricerca, download, cancellazione e selezione del formato.

Il **Model** contiene le componenti:

* **Direttore di Dipartimento**: modella le caratteristiche di un Direttore di Dipartimento e consente di interfacciarsi con l’entità nel database per compiere le operazioni CRUD.
* **Docente**: Modella le caratteristiche di un Docente e consente di interfacciarsi con l’entità nel database per compiere le operazioni CRUD.
* **Esito**: modella le informazioni relative all’esito della validazione di un Green Pass e si occupa dell’interfacciamento con il database per compiere le operazioni CRUD.
* **Dipartimento**: modella un Dipartimento e le sue caratteristiche principali.
* **Report**: modella un report di una validazione e consente di interfacciarsi con la base di dati per compiere le operazioni CRUD.
* **Sessione di Validazione**: modella una Sessione di Validazione conservandone le proprietà e consente di interfacciarsi con il database per compiere le operazioni CRUD.
* **Formato**: si occupa di mantenere le proprietà di un Formato e si interfaccia con il database per compiere le operazioni CRUD.
* **Green Pass**: modella un Green Pass.
* **Studente**: modella uno Studente mantenendo le principali proprietà.

Il **View** contiene le componenti:

* **StudenteGUI**: interfaccia grafica per lo Studente.
* **DocenteGUI**: interfaccia grafica per il Docente.
* **DirettoreDiDipartimentoGUI**: interfaccia grafica per il Direttore di Dipartimento.



1. Glossario

In questa sezione descriveremo i termini tecnici che sono stati utilizzati all’interno del documento stesso.

* **Login**: operazione di autenticazione di un docente o di un Direttore di Dipartimento mediante l’uso di credenziali personali.
* **Logout**: disconnessione dalla propria area personale.
* **Form**: insieme di campi da compilare e sottomettere. I campi possono essere obbligatori o facoltativi.
* **Web Application**: applicazione accessibile via Web per mezzo di una rete Internet.
* **User**: persona che sfrutta le funzionalità del Sistema.
* **Scansione**: lettura di un QR code tramite appropriato dispositivo.