****

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T**est **C**ase  **S**ummary **R**eport  Easy Pass   |  |  | | --- | --- | | Riferimento |  | | Versione | 1.0 | | Data | 4/12/2021 | | 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 Descrizione del Documento 5

1. Relazioni con altri Documenti 5
2. Panoramica del Sistema 5
3. Funzionalità da testare e no 5
4. Criteri di successo/fallimento 5
5. Approccio 5

6.1 Testing di unità 5

6.2 Testing Funzionale 5

1. [Strumenti per il testing (Hardware/Software) 5](#_Toc530825398)
2. [Test Case 5](#_Toc530825398)
3. [Pianificazione del testing e assegnazione dei ruoli 5](#_Toc530825398)

RevisionHistory

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autori** |
| 4/12/2021 | 0.1 | Prima stesura: aggiunta test case registrazione, test case inserisci studenti e test case ricerca report | Alberto Montefusco  Martina Mulino |
| 6/12/2021 | 0.1 | Aggiunta test case invio Green Pass | Viviana Rinaldi |

Responsabilità

|  |  |
| --- | --- |
| **Artefatto** | **Autori** |
| Test case registrazione,    Test case inserisci studenti | Martina Mulino |
| Test case ricerca report | Alberto Montefusco |
| Test case formato report | Gennaro Spina |
| Test case invio Green Pass | Viviana Rinaldi |

1. Introduzione
   1. Descrizione del Documento

Il Test Case Document del progetto Easy Pass presenta la pianificazione e la specifica di quelli che sono i Test Case, ovvero l’insieme di input e di risultati attesi che servono a testare una componente del sistema. In particolare, vengono descritte le funzionalità di Easy Pass di cui si ritiene opportuno testare il corretto funzionamento. Nel caso in cui delle attività di testing evidenziassero degli errori che possano causare comportamenti diversi da quelli attesi e che possano compromettere il buon utilizzo del sistema da parte degli utenti, quest’ultimo può essere sottoposto ad un processo di correzione degli errori individuati, fino a garantire di fornire agli utenti finali un prodotto software che rispecchi tutte le specifiche finora stabilite nelle precedenti fasi di sviluppo.

Nello specifico in questo documento tratteremo il testing Black Box, cioè andremo a testare se una determinata componente si comporta come richiesto dal documento di specifica e di analisi dei requisiti.

1. Relazioni con altri Documenti

Il seguente Documento presenta diversi punti di correlazione con i documenti stilati durante le fasi precedenti dello sviluppo di Easy Pass. In particolare, il presente documento fa riferimento ai casi d’uso descritti nel Requirement Analysis Document.

1. Panoramica del Sistema

Il sistema Easy Pass fornisce tutte le sue funzionalità attraverso una Web Application. Per assicurarsi che ciascuna funzione si comporti come previsto, bisogna quindi assicurarsi che vengano testate tutte le funzionalità offerte dal Sistema. Nel caso in cui queste funzionalità si comportino nella maniera prevista, si può ritenere che il Sistema Easy Pass soddisfi gli obiettivi che si erano prefissati.

1. Funzionalità da testare e no

Le attività di testing previste per il Sistema Easy Pass prevedono di testare il corretto funzionamento della maggior parte delle funzionalità del sistema. Queste sono:

1. Registrazione Docente;
2. Aggiunta di un gruppo di studenti da validare;
3. Ricerca dei report attraverso dei filtri;
4. Impostare il formato di un report;
5. Invio di un Green Pass al Sistema.
6. Criteri di successo/fallimento

Un caso di test ha esito positivo se l’output osservato è differente dal risultato previsto dall’oracolo; al contrario, un caso di test ha esito negativo se l’output osservato coincide con il risultato previsto dall’oracolo. Pertanto, le attività di test hanno successo nei casi in cui riescono ad individuare dei comportamenti anomali nell’esecuzione delle funzionalità del sistema. Nel caso in cui uno o più casi di testi riscuotono successo, è possibile attuare un’opportuna procedura di correzione del difetto riscontrato e, successivamente, ricorrere ad un test di regressione per testare nuovamente la funzionalità modificata ed accertarsi che il problema sia stato risolto.

1. Approccio

Testing di Unità

Testing Funzionale

Testing di Sistema (RAD)?

Testing di Integrazione

Abbiamo utilizzato la tecnica del Black-Box Testing per lo Unit Testing

1. Strumenti per il testing (Hardware/Software)
2. Test Plan

Per sviluppare i Test Cases sarà utilizzato il metodo del Category Partition. Questo metodo consiste nell’identificare per ogni funzionalità da testare dei parametri; per ogni parametro verranno

individuate delle categorie, le quali poi saranno suddivise in scelte. Alle scelte verrà assegnato un

valore. I Test Cases verranno definiti nel documento di Test Cases Specification (TCS).