**­KEY-KNOWLEDGE**



**INDICE**

1)DOMINIO DEL PROBLEMA

INTRO 1

2)SCENARI

**1.L’utente utilizza il sistema per la prima volta 3**

**2.L’utente fa il login 4**

**3.Tutorial 5**

**4.L’utente gioca in modalità classica 8**

**5.L’utente gioca in modalità MISC 11**

3)Requisiti funzionali

[FR01]: 11

[FR02]: 12

[FR03]: 12

[FR05]: 12

[FR06]: 12

[FR07]: 12

[FR08]: 12

[FR09]: 13

[FR10]: 13

4)Requisiti non funzionali

[NFR01]: 14

[NFR02]: 15

[NFR03]: 15

[NFR04]: 15

[NFR05]: 16

[NFR06]: 16

[NFR08]: 16

[NFR09]: 16

**5)Ambiente di destinazione:**

**Destinazione 16**

**6.Scadenze:**

Scadenze 17

**7.Criteri di accettazione dei test:**

**Test 17**

|  |  |  |  |
| --- | --- | --- | --- |
| ****DATA**** | ****VERSIONE**** | ****DESCRIZIONE**** | ****AUTORE**** |
| **17/10/2020** | **1.0** | **Prima versione del problemStatemant** | **Crescenzo Manzone**  **Franco Nicola Fernando**  **Giovanni Battista Mercurio** |
| **23/10/2020** | **1.1** | **Modifica scenari** | **Crescenzo Manzone** |
| **24/10/2020** | **1.2** | **Modifica Requisiti Funzionali** | **Franco Nicola Fernando** |
| **25/10/2020** | **1.3** | **Modifica Requisiti non Funzionali** | **Giovanni Battista Mercurio** |
| **27/10/2020** | **1.4** | **Unione delle modifiche effettuate** | **Crescenzo Manzone**  **Franco Nicola Fernando**  **Giovanni Battista Mercurio** |

**Problem Statement**

1. **Dominio del problema**

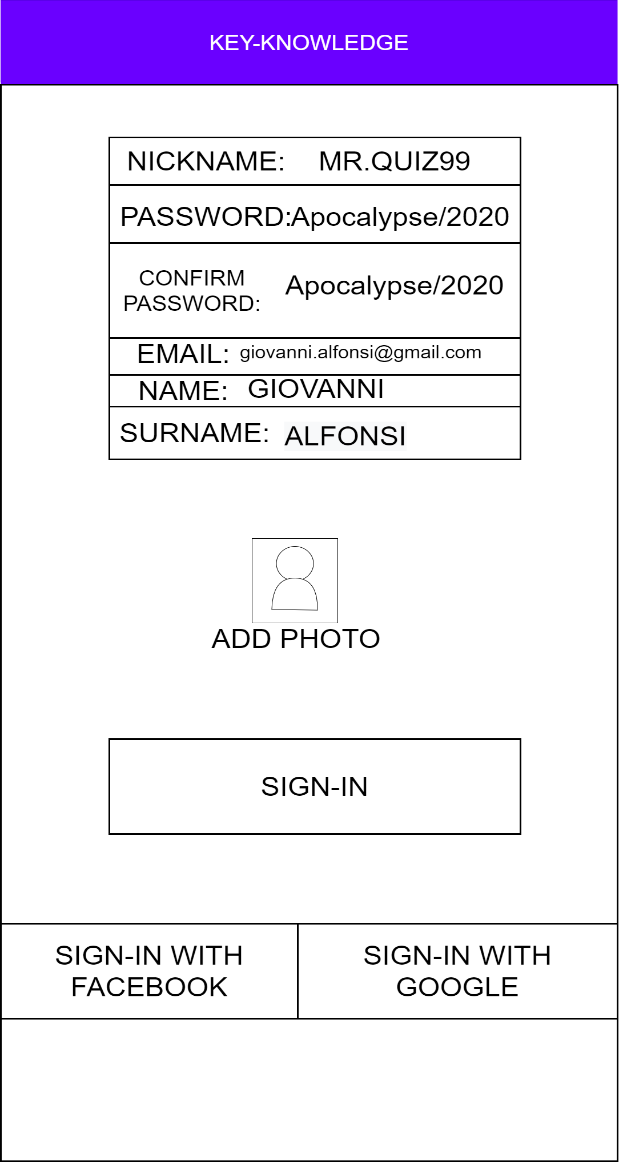
**Il software dovrà fornire la possibilità di interagire con altri utenti e sfidarsi in un quiz. Il vincitore sarà il giocatore che risponde al maggior numero di domande, generate da un agente intelligente che ha la particolarità di evolversi e adattarsi all’utente. Le domande sono suddivise in categorie (STORIA, GEOGRAFIA, SCIENZE, ARTE, CULTURA GENERALE). Una singola partita è sempre 1v1.**

1. **Scenari:**

**1.L’utente utilizza il sistema per la prima volta:**

**Giovanni Alfonsi scarica l’applicazione e la apre per la prima volta. Gli appare davanti la schermata di registrazione che riempie con i suoi dati. Dopo aver inserito correttamente i suoi dati Giovanni clicca sul pulsante SIGN-IN.**

**A sinistra una REGISTRAZIONE con dati corretti, a destra una REGISTRAZIONE con dati non corretti.**



**Gli appare un pop-up.**

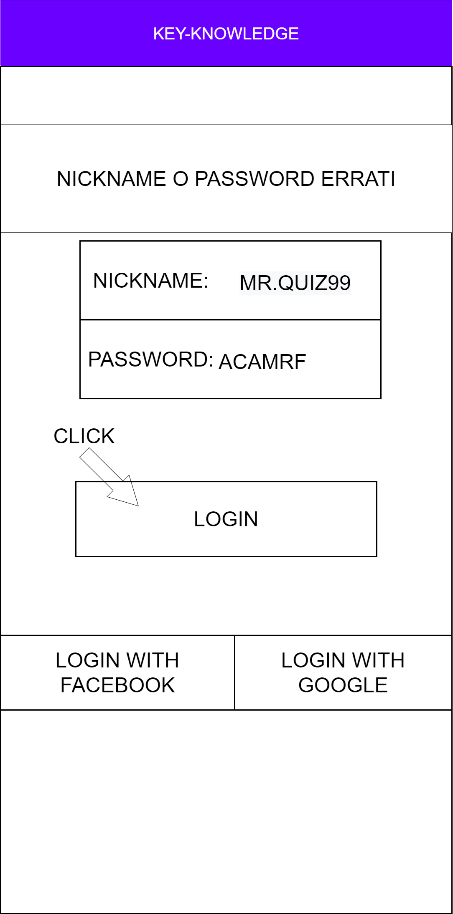
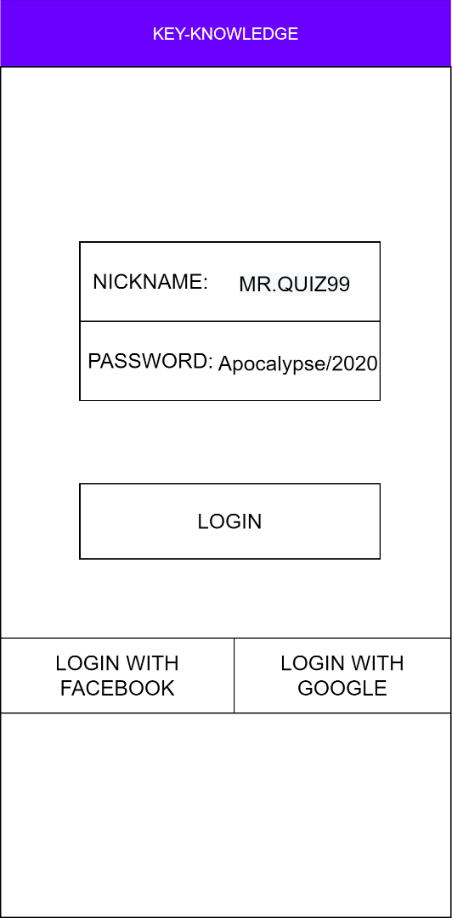


**Giovanni va nella sua casella per confermare l’account, torna sull’applicazione per effettuare il login.**

**2.L’utente fa il login:**

**Giovanni inserisce correttamente i suoi dati e clicca sul pulsante LOG-IN. Essendo il primo utilizzo dell’app da parte di Giovanni, dopo il click su LOG-IN, partirà il tutorial.**

**A sinistra un LOGIN con dati corretti, a destra un LOGIN con dati non corretti.**



**3.Tutorial:**

**Giovanni si trova davanti una schermata dove gli vengono esposte le regole del gioco.**

**L’unica azione che Giovanni può fare per andare avanti è cliccare su NEXT, quindi clicca su NEXT, che lo fa proseguire con il tutorial.**

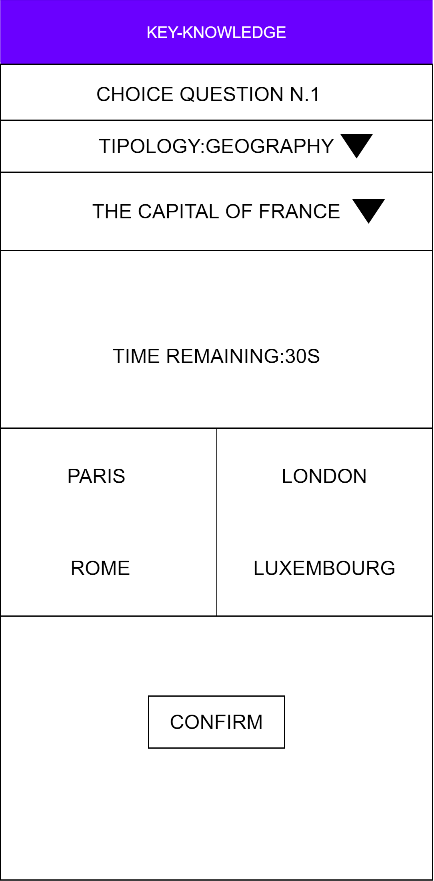


**Giovanni si trova davanti una domanda a cui deve rispondere con il tempo che scorre, gli viene indicato di cliccare su una risposta e successivamente su CONFIRM. Quindi Giovanni clicca su una risposta e poi su CONFIRM. Che lo fa proseguire con il tutorial**



**Giovanni si trova davanti una domanda da scegliere, prima di fare qualsiasi azione, gli appare un pop-up dove gli spiega che questa è una domanda che lui sceglie e che viene usata per il suo pseudo-avversario, visto che questo è un tutorial non ha un avversario. Clicca su NEXT per scegliere la domanda.**

**Giovanni sceglie la tipologia, dopodiché sceglie una delle 4 domande relative alla tipologia scelta e infine clicca su CONFIRM. Giovanni va avanti con il tutorial.**

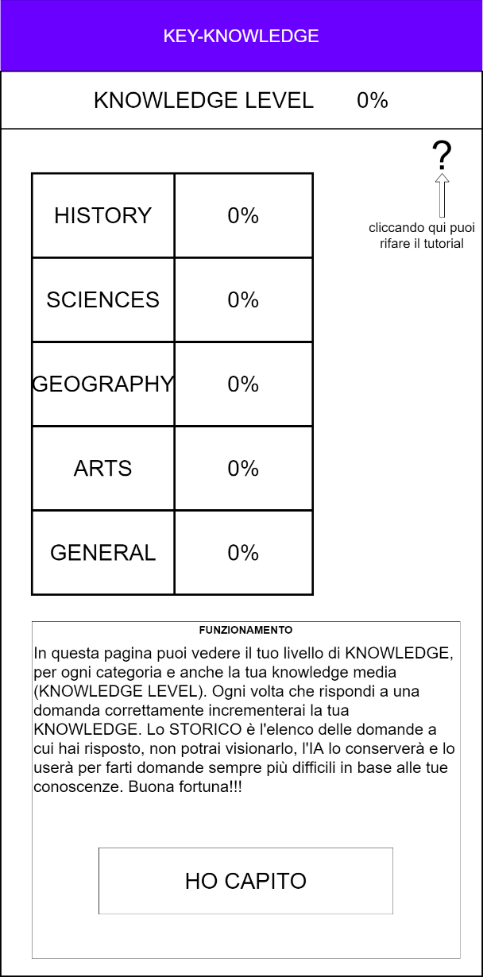
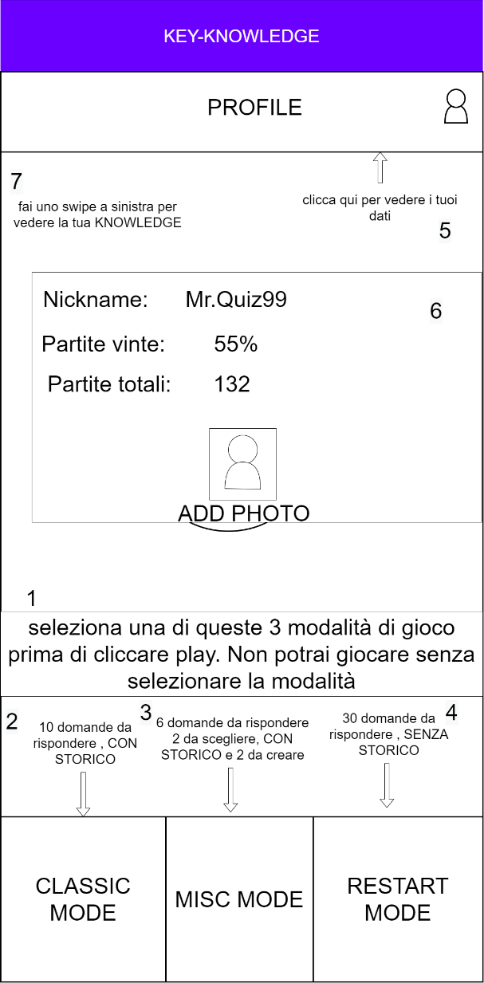


**Giovanni si trova davanti una schermata dove deve scrivere una domanda. Prima di fare qualsiasi azione gli appare un pop-up che gli spiega che la domanda che lui scriverà sarà fatta rispondere dal suo pseudo-avversario. Giovanni scrive la domanda e le risposte e sceglie anche la domanda giusta, dopodiché può confermare andando cosi avanti con il tutorial.**



**Giovanni si trova nella schermata principale dove gli vengono spiegate le azioni che può fare. Il tutorial gli farà apparire i messaggi in 7 passaggi arrivando alla schermata di KNOWLEDGE, qui sarà spiegato l’utilità di questa pagina e il funzionamento dell’IA.**

**A sinistra la schermata principale, a destra la schermata di KNOWLEDGE.**



**Giovanni legge il funzionamento e clicca su HO CAPITO, il tutorial termina e ritorna alla schermata principale.**

**4.L’utente gioca in modalità classica:**

**Giovanni ha terminato il tutorial e crede di essere in grado di affrontare una partita reale con un altro giocatore. Trovandosi nella schermata principale sceglie la modalità di gioco classica e clicca su play.**

**A sinistra le azioni di Giovanni, al centro la schermata di caricamento che combina Giovanni con un altro utente. A destra il caricamento è terminato Giovanni è stato combinato con l’utente che ha come nickname SOLVERMASTER. In basso la schermata di caricamento è fallita perché non è stato trovato nessun utente. Fortunatamente Giovanni viene abbinato con l’utente SOLVERMASTER e cominciano a giocare.**





**Giovanni si trova davanti una domanda da rispondere e contemporaneamente anche il suo avversario risponde a una domanda. A sinistra la domanda di Giovanni, a destra quella del suo avversario.**

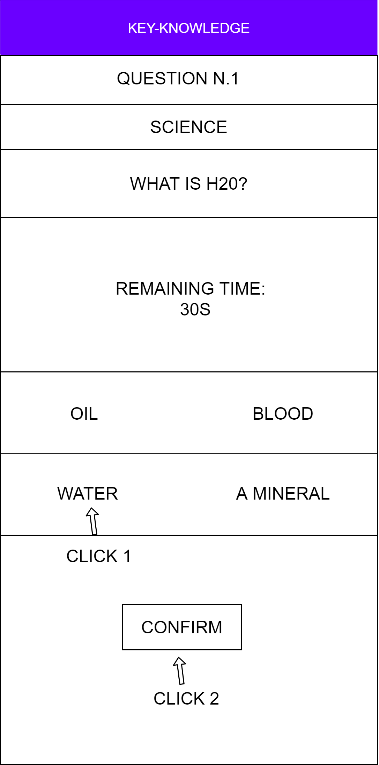
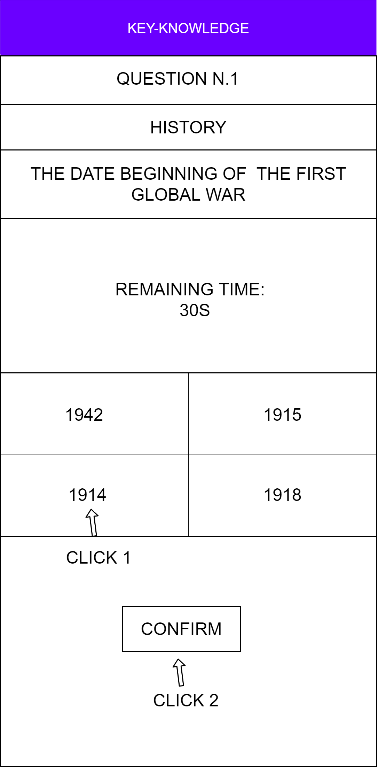


**Giovanni risponde velocemente a tutte e 10 le domande, ma il suo avversario sta ancora rispondendo. A sinistra Giovanni che ha finito ma attende, a destra la partita terminata con il messaggio di vittoria o sconfitta. Dopo di che Giovanni viene mandato sulla schermata principale.**



**5.L’utente gioca in modalità MISC:**

**Giovanni si trova sulla schermata principale e vuole fare un’altra partita, ma stavolta nella modalità MISC. Quindi clicca prima su MISC MODE e successivamente su PLAY. Raggiunge la schermata di caricamento e viene abbinato con un utente che ha nickname Alex1996 e iniziano a giocare. Alla prima domanda risponde Giovanni per primo. A sinistra le domande, a destra ciò che appare a Giovanni.**



**Questo meccanismo di attesa permane per ogni domanda in questa modalità.**

**Alla fine della partita Giovanni torna alla schermata principale e chiude il gioco.**

**3.Requisiti funzionali**

**[FR01]:**

GESTIONE UTENTE (sign-in)

-Il sistema deve fornire all’utente la possibilità di registrarsi con queste credenziali:

. E-MAIL

. PASSWORD

. NOME

. COGNOME

. NICKNAME

. FOTO (opzionale)

Oppure:

. Sign with facebook-account

. Sign with Google-account

**[FR02]:**

GESTIONE UTENTE (confirm)

-Il sistema deve fornire la possibilità di confermare l’account, tramite conferma via e-mail.

**[FR03]:**

GESTIONE UTENTE (log-in)

-Il sistema deve fornire all’utente la possibilità di accedere alle funzionalità dell’app e quindi poter sfidare altri utenti registrati inserendo:

-E-MAIL/NICKNAME

-PASSWORD

**[FR03]:**

GESTIONE UTENTE (profile management)

-Il sistema deve fornire all’utente la possibilità di visionarie le proprie statistiche (domande sbagliate/vittorie/sconfitte…)

**[FR05]:**

GESTIONE UTENTE (update user)

-Il sistema deve fornire all’utente la possibilità di modificare i propri dati personali

**[FR06]:**

GESTORE UTENTI (show users)

-Il sistema fornisce all’utente admin, la possibilità di visionare l’username, la foto profilo e le statistiche di ogni utente registrato.

**[FR07]:**

GESTORE UTENTI (remove users)

-Il sistema fornisce all’utente admin, la possibilità di rimuovere utenti inattivi per molto tempo.

**[FR08]:**

GESTORE CATEGORIE (update categories)

-Il sistema deve fornire all’utente admin, la possibilità di modificare o aggiungere nuove modalità di gioco specificando le caratteristiche necessarie della nuova modalità.

**[FR09]:**

UTENTE-GIOCATORE (start match)

-Il sistema deve fornire all’utente la possibilità di avviare una sfida contro un altro giocatore, scegliendo opportunamente la modalità di gioco preferita prima di cominciare.

**[FR10]:**

UTENTE-GIOCATORE (end match)

-Il sistema deve fornire all’utente la possibilità di interrompere un match con la conseguente rimozione dei dati di quella partita.

**[FR11]:** GIOCATORE (playing)

-Il sistema deve fornire all’utente, a seconda della modalità di gioco scelta, la possibilità di: scegliere la risposta corretta, scegliere la domanda da porre all’avversario e di scrivere una domanda per l’avversario.

**Tabella delle priorità:**

(U): utente

(GC): gestore categorie

(GU): gestore utenti

(UG): utente giocatore

|  |  |
| --- | --- |
| **Requisito** | **Livello di priorità** |
| (U)sign-in | Alto |
| (U)Log-in | Alto |
| (U)Profile management | Medio |
| (U)Update-user | Medio |
| (GU)Show users | Medio |
| (GU)Remove users | Medio |
| (GC)Update categories | Basso |
| (UG)Start-match | Alto |
| (UG)End-match | Medio |
| (UG)Playing | Alto |
| (U)Confirm | Medio |

**4.Requisiti non funzionali**

1. **Usability**

**[NFR01]:**

Level of expertise

Esempio di interfaccia utente:



**[NFR02]:**

Documentation

**Regole del quiz:**

**In ogni partita l’utente dovrà rispondere a una serie di domande, in alcuni casi potrà sceglierle per l’avversario o anche crearle per l’avversario. In ogni caso ogni domanda da rispondere, scegliere o creare ha un tempo di azione (intorno ai 30 secondi). Se il tempo di azione termina prima che l’utente ha fatto una scelta gli sarà assegnata una penalità. Non influirà sullo storico poiché non ha dato nessuna risposta. Le penalità si accumulano, ognuna conta -0,25. Le risposte corrette segnano 1 punto e le risposte sbagliate 0. Vince l’utente che fa più punti, in caso di pareggio chi ha il tempo minore. Se l’utente guadagna 2 penalità, alla terza viene messo offline e il suo avversario vince. Oltretutto in base alla modalità di gioco scelta cambieranno i tempi di attesa.**

**Nella modalità classica e nella modalità RESTART gli utenti risponderanno in maniera autonoma alle domande e quando avranno finito di rispondere dovranno attendere l’avversario o viceversa per conoscere il risultato della partita. Nella modalità MISC ad ogni domanda da rispondere, scegliere o creare l’utente dovrà aspettare il suo avversario o viceversa.**

**Funzionamento Intelligenza Artificiale:**

**L’obiettivo dell’intelligenza sarà quello di mettere in difficoltà l’utente proponendogli domande sempre più difficili. Ogni volta che l’utente risponde a una domanda, il suo risultato viene usato dall’intelligenza per capire su quali argomenti è più o meno bravo l’utente. Se un utente risponde in maniera corretta a diverse domande di storia sarà meno probabile che l’intelligenza gli proporrà ancora domande di storia.**

**Scelte implementative**

**L’utente non deve avere la possibilità di vedere se ha risposto in maniera corretta o meno ad una specifica domanda. Lo scopo del nostro sistema è di spronare l’utente a informarsi su quelli che sono gli argomenti più difficoltosi per lui.**

1. **Reliability**

**[NFR03]:**

Availability

Il gioco una volta installato è operativo in qualsiasi momento, 24h/24, tranne in ore specifiche di stallo utilizzate per poter controllare, aggiornare il sistema o risolvere errori di vario tipo, ma notificate all’utente tramite schermata di “work in progress”.

**[NFR04]:**

Robustness (errors and exceptions) Se si presentano anomalie che interrompono il match, l’utente visionerà una schermata di errore e sarà trasportato alla schermata di inizio e la partita non verrà contata.

1. **Performance**

**[NFR05]:**

Il sistema ha una rapidità di esecuzione: il giocatore, dopo aver risposto alla domanda, conosce subito il risultato corrente in base alla sua scelta.

Il sistema supporterà un alto numero di utenti simultanei.

1. **Supportability**

**[NFR06]:**

Maintainability

Al lancio del Software nei primi mesi di funzionamento prevediamo delle revisioni del sistema, dato che durante svariati utilizzi possono uscire bug non visionati. in caso di modifiche grafiche o di codice tramite Android studio possono essere apportate velocemente. Per poi mettere online una nuova versione dell’applicazione.

1. **Implementation**

**[NFR08]:**

L’applicazione sarà implementata usando linguaggio **JAVA**

IDE utilizzata: **Android Studio**

1. **Packaging**

**[NFR09]:**

L’applicazione dev’essere installata dal play-store di google oppure, alternativamente da un file di installazione apposito (. apk), che aggiungerà l’app tra le applicazioni già esistenti sul dispositivo.

**5.Ambiente di destinazione:**

**-Sistemi operativi che supportano l’app: Android**

**-Compatibilità: alta, supportata dal 70% dei dispositivi in circolazione.**

**6.Scadenze:**

6 novembre: Requisiti e casi d’uso

20 novembre: Requirements Analysis Document

3 dicembre: System Design Document

5 gennaio: piano di test e specifica interfacce dei moduli del sistema

29 gennaio: consegna progetto

**7.Criteri di accettazione dei test:**

**per completare la fase di testing, il sistema dovrà soddisfare i requisiti attesi e attenersi agli scenari elencati.**