



UNIVERSITÀ
DI TRENTO

DIPARTIMENTO DI INGEGNERIA E SCIENZA DELL'INFORMAZIONE

Sleep Code

PROGETTO PER IL CORSO DI INGEGNERIA DEL SOFTWARE
ANNO ACCADEMICO 2023-2024

Analisi dei Requisiti

Descrizione: documento di analisi dei requisiti funzionali, non funzionali, front-end e back-end.

Numero documento: D1

Membri del gruppo:

Raffaele CASTAGNA

Alberto ROVESTI

Zeno SALETTI

Numero gruppo: G17

Ultima revisione: 4 ottobre 2023

Indice

1	Introduzione e scopi	2
1.1	Scopo del documento	2
1.2	Obiettivo del progetto	2
1.3	Pubblico di riferimento ed esigenze	2
2	Requisiti funzionali	3
2.1	Accesso	3
2.2	Consultazione dei problemi	3
2.3	Esercitazione	4
2.4	Gestione profilo e account	5
3	Requisiti non funzionali	6
3.1	Caratteristiche delle risorse	6
3.2	Caratteristiche di sistema	6
3.3	Privacy e sicurezza	7
4	Design front-end	8
5	Design back-end	8

1 Introduzione e scopi

1.1 Scopo del documento

Le informazioni contenute in questo documento concorrono ad esporre l'analisi dei requisiti relativa al progetto *SleepCode*. In particolare, dopo aver specificato gli obiettivi e gli attori coinvolti, verranno definiti i requisiti funzionali e non funzionali; verrà presentata una proposta di design di back-end; infine saranno riportati i sistemi esterni di back-end coi quali il servizio dovrà interfacciarsi.

1.2 Obiettivo del progetto

Il progetto proposto si prefigge, come scopo fondante, di fornire alla comunità di giovani informatici un servizio online di *esercitazione* mirata alla programmazione e alla progettazione di piccoli algoritmi risolutivi, mediante la scrittura di codice.

1.3 Pubblico di riferimento ed esigenze

Il servizio vuole rendersi utile soprattutto a coloro che sono coinvolti in percorsi di studio attinenti all'ambito informatico, ma specialmente anche a chiunque desideri cimentarsi nella risoluzione di piccoli problemi di programmazione; pertanto ci si aspetta che chiunque desideri usufruire del servizio possieda almeno le conoscenze basilari della programmazione. Esempi di queste nozioni pregresse, che tuttavia non devono necessariamente essere ampie e approfondite per utilizzare il servizio¹, sono: cosa si intende per algoritmo e linguaggio di programmazione, familiarità nell'uso di qualche linguaggio di programmazione, tipi e strutture di dati più comuni.

Di fatto, il progetto che verrà sviluppato ha come scopo principale di creare una piattaforma accessibile online a singoli utilizzatori che desiderano esercitarsi, valutare e approfondire le personali conoscenze e abilità di *problem solving* legate alla programmazione. D'ora in avanti, in questo e nei successivi documenti, questo pubblico di individui appena descritti verrà indicato con il termine *utenti*.

¹Gli utenti più esperti possono indubbiamente trarre vantaggio dal loro bagaglio culturale per approcciarsi con maggior facilità al servizio.

2 Requisiti funzionali

Vengono di seguito elencati i principali requisiti funzionali (RF) del progetto. Ogni sottosezione di questa parte del documento risponde a diversi scopi precedentemente accennati, suddividendo eventuali macro-funzioni in requisiti minori nel caso di obiettivi di più ampia portata.

2.1 Accesso

RF 1. Registrazione Il sistema deve mettere a disposizione dei nuovi utenti una pagina nella quale sia possibile effettuare la registrazione alla piattaforma online. Devono essere richiesti i seguenti campi:

- Indirizzo email.
- Nome utente.
- Password: la password deve essere oscurata durante l'inserimento. Al fine di limitare gli errori di digitazione e l'inserimento di una password che non corrisponde a quella scelta dall'utente, deve essere presente un secondo campo di conferma, dove la password deve essere riscritta. La password deve in ogni caso essere conforme a quanto specificato nel RNF 11.

L'utente dotato di un account sarà considerato dal servizio come *utente registrato*.

RF 2. Login Il sistema deve permettere all'utente già registrato al servizio di accedere alla piattaforma mediante l'inserimento dell'indirizzo email e la password, impostate in fase di registrazione. L'utente deve essere notificato qualora non sia possibile concludere correttamente l'operazione di login, quindi se l'email inserita non sia registrata o la password non corrisponda; è comunque possibile ritentare il login un numero illimitato di volte.

RF 3. Recupero password L'utente registrato deve poter recuperare la password del proprio account qualora tale dato dovesse essere dimenticato in fase di login. La procedura di recupero deve prevedere:

- L'invio di un messaggio all'indirizzo email attualmente associato all'account dell'utente; l'email contiene un link che guida l'utente nella creazione di una nuova password.
- L'inserimento (in una pagina dedicata) di una nuova password che, come in fase di registrazione, deve essere scritta, confermata ed essere conforme al RNF 11.

RF 4. Logout L'utente registrato e autenticato deve poter interrompere la sessione di accesso al servizio. Questa procedura di *logout* realizza il passaggio dell'utente dallo stato *autenticato* a quello *anonimo*.

2.2 Consultazione dei problemi

RF 5. Consultazione del catalogo dei problemi Il servizio deve mettere a disposizione un insieme di problemi sui quali l'utente possa esercitarsi. L'utente deve poter consultare un catalogo, atto a raccogliere i quesiti, e navigare al suo interno. Deve quindi essere possibile:

1. Visualizzare tale catalogo in una vista dedicata. Il catalogo mostra i campi di ogni problema, descritti nel RNF 2.
2. Cercare uno o più problemi specifici mediante ricerca filtrata per campi (specificati dai metadati elencati al RNF 2).

3. Selezionare dal catalogo un problema specifico, cosicché la descrizione dell'intero problema possa essere visionata per mezzo delle funzionalità di cui al RF 6.

RF 6. Consultazione di un problema Deve essere fornito un visualizzatore che mostri le informazioni del problema precedentemente selezionato dal catalogo. La visualizzazione deve rendere disponibile alla vista dell'utente i dati relativi al problema e specificati nel RNF 1.

2.3 Esercitazione

RF 7. Avviare una sessione di esercitazione L'utente deve poter scegliere dal catalogo il problema desiderato e cominciare a risolverlo. La sessione di esercitazione avviene in una pagina apposita, dove l'utente deve poter disporre di tutti gli strumenti necessari alla realizzazione delle seguenti funzionalità:

1. Visualizzare il contenuto dell'esercizio (RNF 1).
2. Visualizzare i test cases (RF 9.2).
3. Avviare e fermare a proprio piacimento un timer per cronometrare il tempo speso per risolvere il quesito scelto.
4. Essere al corrente di quale linguaggio di programmazione sia attualmente attivo per la scrittura di codice. Deve altresì essere possibile selezionare uno dei linguaggi messi a disposizione dalla piattaforma².
5. Scrivere, sotto forma di codice nel linguaggio di programmazione scelto, l'algoritmo risolutivo del problema attualmente attivo.

RF 8. Correttezza sintattica del codice L'utente deve poter verificare che il codice scritto sia corretto e in grado di risolvere il problema. Quindi devono essere messe a disposizione le seguenti funzionalità:

1. Compilazione del codice.
2. Visualizzazione di avvisi relativi a eventuali errori di compilazione *oppure* di compilazione andata a buon fine. In caso di errori di scrittura, l'utente deve poter correggere tali errori riscrivendo nell'area destinata al codice.

RF 9. Verifica della correttezza dell'algoritmo³ L'utente deve poter verificare la correttezza del codice scritto eseguendolo e testandolo:

1. Il codice deve essere eseguito sottoponendolo ad un certo numero di test cases (al minimo 3), cioè fornendo opportune istanze di input (Per esempio, se un problema richiede di sommare due numeri interi, il codice risolutivo proposto dall'utente verrà eseguito fornendo ad esso coppie di interi e registrando i risultati in output.)
2. L'utente deve essere in grado di essere notificato se il codice supera tutti i test case oppure se almeno un test case non è stato superato. In entrambi i casi l'utente verrà avvisato. Per ogni test case, sia l'input che l'output desiderato sono visibili. Inoltre, l'utente deve poter riscrivere e perfezionare l'algoritmo e sottoporre ripetutamente il codice ai test cases.

²Per approfondimenti sulla disponibilità dei linguaggi, si veda RNF 3.1.

³La *correttezza* di cui si parla in questo caso riguarda solo l'efficacia risolutiva dell'algoritmo.

3. Nel caso in cui l'utente autenticato sia in grado di superare tutti i test case disponibili, la sessione di esercitazione termina: deve essere visualizzato il tempo impiegato e, qualora l'utente sia autenticato, il problema deve essere contrassegnato come *risolto* nel campo *stato* del problema e i progressi devono essere aggiornati.

RF 10. Soluzione Per ogni problema viene messo a disposizione un video che mostra una possibile soluzione. Il video è accessibile sia dal catalogo che dalla finestra dedicata allo svolgimento del problema.

2.4 Gestione profilo e account

RF 11. Progressi L'utente autenticato deve poter monitorare i propri progressi, accedendo ai dati seguenti:

- Il numero totale di problemi risolti.
- Il numero di problemi risolti suddivisi per difficoltà.

RF 12. Preferiti L'utente deve poter contrassegnare un qualsiasi problema come *preferito*. Tale informazione deve essere visibile nel catalogo, accanto ad ogni problema contrassegnato in questo modo.

RF 13. Aggiornamento account L'utente registrato e autenticato deve poter modificare i dati chiave del proprio account in conformità con i requisiti non funzionali relativi alla sicurezza [...]:

1. L'utente deve poter migrare ad un indirizzo email differente. A tale indirizzo deve essere inviato un messaggio di conferma, previo inserimento della password nella pagina dedicata a questa particolare operazione. L'email deve cambiare effettivamente solo nel momento in cui l'utente conferma la ricezione del messaggio presso il nuovo indirizzo.
2. L'utente deve poter modificare la password del proprio account. La password può essere cambiata previo inserimento di quella attualmente associata all'account. La nuova password deve essere digitata due volte per conferma e deve rispettare quanto specificato dal RNF 11. L'utente deve essere notificato di tale modifica anche tramite l'indirizzo email attualmente associato all'account.

3 Requisiti non funzionali

Vengono ora elencati i requisiti non funzionali (RNF) del servizio.

3.1 Caratteristiche delle risorse

Alcuni requisiti funzionali sono dedicati alla descrizione delle principali risorse fornite dalla piattaforma, di cui le principali sono i problemi.

RNF 1. Struttura di un problema Tutti i problemi forniti possiedono:

- Un titolo.
- Un testo, scritto prevalentemente in linguaggio naturale, che descrive uno scenario che richiede di essere risolto per mezzo di un algoritmo. Eventuali immagini e proposizioni matematiche formali possono accompagnare i testi.
- Almeno 3 esempi di input insieme al relativo output che mostra il risultato atteso.

RNF 2. Metadati dei problemi Ogni problema è essenzialmente un'entità caratterizzata dai seguenti dati descrittivi:

- Stato: i problemi già risolti dall'utente sono contrassegnati in un campo *stato* apposito.
- Nome identificativo: si tratta di una stringa alfanumerica, che riprende una o più parole chiave del titolo del problema.
- Difficoltà: ogni problema possiede un'etichetta (*tag*) associata che ne valuta indicativamente la difficoltà: bassa, intermedia e alta.
- Categoria: la categoria indica la principale area di interesse del problema, come ad esempio le strutture di dati di cui tratta (array, grafi, code, ecc.).
- Soluzione: un video che mostra come il problema potrebbe essere risolto.
- Preferito: la marcatura *preferito* è visibile sul problema quando esso compare nel catalogo e quando viene consultato.

Ai fini della consultazione del catalogo, i filtri applicabili riguardano i campi *stato*, *difficoltà*, *categoria*, *preferito* e *nome* (quest'ultima mediante barra di ricerca).

3.2 Caratteristiche di sistema

RNF 3. Scalabilità L'infrastruttura del servizio deve essere scalabile e aperta alle esigenze derivanti dall'aumento di nuovi utenti. Questo requisito è motivato dalla disponibilità online del servizio che verrà sviluppato. In particolare:

1. L'infrastruttura del servizio deve essere adattabile a eventuali crescite nel numero di utenti, in modo da prevenire possibili cali di prestazioni eccessivi.
2. Data l'eterogeneità di linguaggi di programmazione esistenti al momento della stesura di questo documento, è importante che il servizio sia in grado di accogliere con l'avanzare del tempo codici scritti in linguaggi differenti.

RNF 4. Compatibilità La piattaforma del servizio deve essere accessibile mediante i principali browser attualmente disponibili in commercio (in particolare Chrome, Edge, Firefox).

RNF 5. Usabilità La piattaforma del servizio deve permettere all'utente di sfruttare le funzionalità disponibili al proprio livello di accesso senza l'ausilio di istruzioni scritte e verbose. L'intuitività dell'interfaccia deve essere sufficiente a guidare l'utente nella realizzazione dei suoi scopi.

RNF 6. Aspetto L'interfaccia deve presentarsi gradevole alla vista dell'utente, preferendo gradazioni cromatiche scure e un contrasto sufficientemente equilibrato, al fine di garantire la leggibilità e contribuire alla riduzione dell'affaticamento della vista.

RNF 7. Lingua di sistema Il servizio sarà erogato in lingua italiana. Altrettanto sarà fatto per i testi dei problemi. I video relativi alla soluzione dei problemi possono essere in italiano oppure in inglese.

RNF 8. Affidabilità

RNF 9. Prestazioni [decommenta]

3.3 Privacy e sicurezza

RNF 10. Privacy e trattamento dei dati Il servizio deve essere progettato e realizzato in ottemperanza delle vigenti disposizioni di legge in materia di tutela della privacy e trattamento dei dati:

1. L'applicazione fornita dal servizio deve essere conforme al regolamento [UE n.2016/679](#) (GDPR) per la protezione dei dati.

RNF 11. Password In tutti gli scenari nei quali è richiesto l'inserimento di una password, devono essere rispettate le seguenti caratteristiche:

- Lunghezza compresa tra 8 e 64 caratteri.
- Contenere almeno una lettera maiuscola.
- Contenere almeno una lettera minuscola.
- Contenere almeno un numero.
- Contenere almeno un carattere speciale scelto tra i seguenti:

! ? # \$ % & @ * + . , ; : / - = _ \ () [] { }

RNF 12. Integrità e operazioni su dati sensibili L'integrità di dati sensibili dell'account, quali email ma soprattutto password, è garantita da opportune procedure di conferma e notifica all'utente. Data la natura di queste operazioni di sicurezza, i requisiti funzionali relativi alla gestione dei dati sensibili (RF 1, RF 3, RF 13) provvedono a descrivere il comportamento del servizio nel caso di errori e incongruenze.

4 Design front-end

5 Design back-end

BE 1. Video - YouTube Per reperire e riprodurre i video contenenti le soluzioni dei problemi, il servizio si affiderà alla piattaforma YouTube.

BE 2. Database - Firebase

BE 3. Sistema di invio email L'invio dei messaggi di posta elettronica, specificati nei requisiti, sarà gestito da servizi di terze parti.