



**UNIVERSITÀ
DI TRENTO**

**Dipartimento di
Ingegneria e Scienza dell'Informazione**

Progetto: TRENTUS

Descrizione di progetto

Doc. Name: *D1-TRENTUS-DescrizioneProgetto*

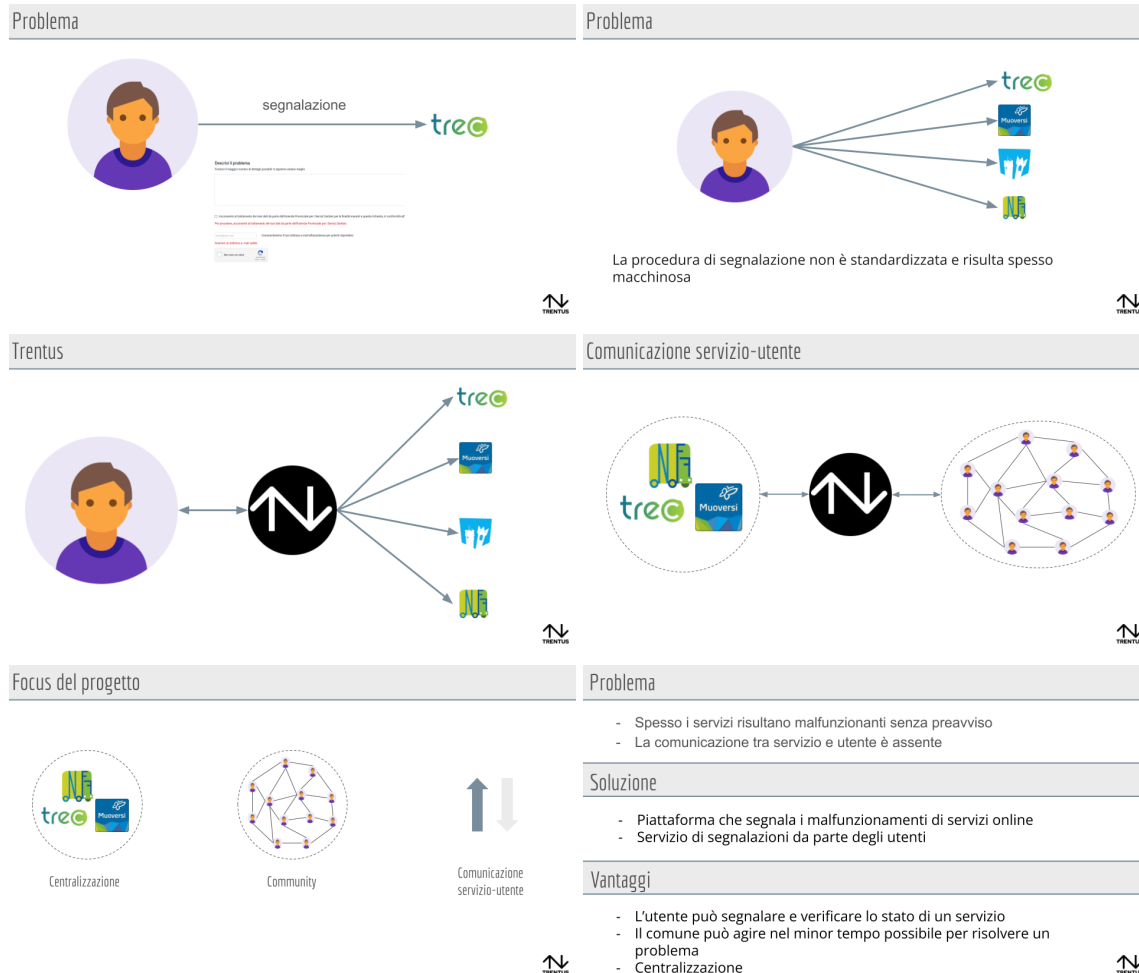
Doc. Number: **D1-V0.1**

INDICE

1. Il progetto TRENTUS	1
2. Requisiti Funzionali	2
3. Requisiti Non Funzionali	3
4. Design Front-end	4

1. Il progetto TRENTUS

Slides Pitch:



Il problema che stiamo risolvendo riguarda la segnalazione di malfunzionamenti di servizi online. Capita spesso che questi abbiano dei problemi, per via di manutenzioni o imprevisti. Qui abbiamo due punti di vista: quello del Gestore del Servizio e quello dell'Utente

1. Il GdS, nel caso di manutenzioni, potrebbe voler segnalare all'Utente la chiusura momentanea del servizio, ma non ha un metodo efficace e centralizzato per farlo, dove l'Utente possa venirne a conoscenza prima di necessitare il servizio.
2. Il Gds, nel caso di problematiche, potrebbe non esserne nemmeno a conoscenza. L'Utente del servizio è sicuramente il migliore per trovare errori e malfunzionamenti, ma se volesse segnalarli non ha un luogo per farlo. Nel caso un metodo ci sia, è in genere macchinoso e poco ovvio, per questo spesso non viene utilizzato.
3. L'Utente, nei casi sopraelencati, vorrebbe essere a conoscenza dell'interruzione del servizio prima di doverne avere necessità, per non essere colto impreparato, così da prendere decisioni alternative o pianificare diversamente le proprie attività.

4. L'Utente potrebbe voler segnalare il problema riscontrato, soprattutto per esortare il GdS a risolverlo. Questo non è sempre possibile oppure non ha una procedura chiara e veloce per farlo. Se esiste, molto spesso è abbastanza nascosta. In questo caso l'Utente potrebbe semplicemente arrendersi e non lasciare il feedback.
5. L'Utente, quando ha un problema, spesso pensa che sia solo suo, anche quando non lo è. Questo lo spinge a chiamare il servizio clienti e presto il GdS avrà i telefoni intasati per un motivo che si poteva evitare.

Il nostro progetto ha come obiettivo la realizzazione di una WebApp per centralizzare le segnalazioni dei disservizi. Sarà progettata per creare una comunicazione efficace tra l'Utente ed il GdS. Trentus offrirà all'Utente un metodo centralizzato, rapido e comodo per visualizzare lo stato dei servizi online comunali e inviare segnalazioni. Inoltre permetterà al GdS di avere uno strumento di monitoraggio del proprio servizio.

La nostra WebApp mira a risolvere tutti i problemi sopraelencati:

1. Il GdS, nel caso di manutenzioni, potrà segnalare ai suoi Utenti la chiusura momentanea del servizio in modo semplice, dove gli utenti interessati potranno essere notificati in un istante.
2. Il Gds, nel caso di problematiche, potrà essere notificato dagli utilizzatori del servizio in modo efficace, così da potersi subito mettere a lavoro per risolvere il problema.
3. L'Utente, in casi del genere, potrà essere notificato ancor prima di dover avere necessità del servizio, per non essere colto impreparato, così da prendere decisioni alternative o pianificare diversamente le proprie attività.
4. L'Utente potrà segnalare il problema riscontrato grazie ad un sondaggio esplicitamente ideato per il servizio, così da poter inviare la segnalazione in modo efficace e conciso.
5. L'Utente, invece che intasare il servizio clienti del GdS, saprà che l'unica cosa che può fare è aspettare.

Trentus sarà completamente fruibile tramite browser, senza la necessità di installare applicazioni aggiuntive, e sarà progettato per garantire un'esperienza intuitiva e sicura. Sarà inoltre conforme alle normative vigenti in materia di protezione dei dati personali, garantendo che le informazioni sensibili degli utenti siano adeguatamente protette. Offrirà anche funzionalità avanzate come notifiche automatiche sullo stato dei servizi che interessano all'Utente, e sulle segnalazioni riguardanti il servizio al GdS.

Vantaggi per il Gestore del Servizio:

- **Segnalazioni centralizzate:** Il GdS sarà in grado di segnalare agli Utenti della WebApp qualsiasi malfunzionamento del servizio, oppure riguardo un periodo di sciopero/manutenzione.
- **Sondaggi Utente:** L'Utente sarà in grado di segnalare qualsiasi malfunzionamento del servizio, così che il GdS possa rendersene conto, nel caso non lo sapesse, e risolverlo il prima possibile.

- **Facile estensibilità:** Il GdS principale per questo progetto è il comune, e sicuramente avrà la necessità di delegare alcuni servizi a privati. La nostra WebApp offre la possibilità di aggiungere velocemente qualsiasi servizio. Anche l'Utente potrebbe essere interessato ad altri servizi utili, un esempio i trasporti pubblici
- **Controlli tramite ping (da discutere).** Trentus, oltre a fare affidamento sull'Utente, può essere integrata con un metodo di ping periodico per verificare la stabilità di un sito in modo sì meno preciso, ma automatico e sempre all'erta.

Vantaggi per l'Utente:

- **Stati centralizzati:** L'Utente sarà in grado di vedere la disponibilità di molteplici servizi tutti in un unico programma, ed in modo veloce.
- **Segnalazioni centralizzate:** L'Utente potrà segnalare al GdS qualsiasi malfunzionamento che possa riscontrare in un unico luogo, indipendentemente dal servizio a cui sta usufruendo.

Limiti dell'applicazione:

- **Notifiche:** Essendo una WebApp e non un'applicazione da telefono, le notifiche potrebbero essere più complesse da gestire
- **Permessi.** Il GdS deve acconsentire all'utilizzo della WebApp.
- **Personale.** Nonostante il nostro obiettivo sia di rendere Trentus più automatica possibile, dalla parte del GdS deve esserci qualcuno che controlli le notifiche Utente e legga i commenti, e che prenda il tempo di segnalare disservizi.

2. Requisiti funzionali

Requisiti funzionali comuni:

- **RF1: Login.** Trentus permette all'Utente di registrarsi attraverso delle credenziali (eventualmente usando SPID, in modo da garantire l'unicità dell'accesso al servizio da parte della singola persona).
- **RF2: Stato IRT.** Trentus mostra all'Utente una lista di servizi in tempo reale, specificando se questi sono operativi o meno, oppure possono avere dei problemi. Inoltre, deve essere possibile vedere il numero di segnalazioni sul servizio.

Requisiti funzionali per l'Utente:

- **RF3: Segnalazione problemi.** Trentus permette all'Utente di segnalare il malfunzionamento di un servizio. L'Utente, scrollando la lista dei servizi, o cercandolo attraverso l'apposita sezione, segnala un problema attraverso il bottone dedicato, compilando un sondaggio.
- **RF4: Sondaggi su misura** Trentus permette all'Utente di specificare il tipo di problema riscontrato durante l'uso del servizio, attraverso un menu apposito, realizzato appositamente dal GdS per identificare l'errore efficacemente.
- **RF5: Lista preferiti.** Trentus permette all'Utente di selezionare un servizio come "preferito" per permettere di controllarlo rapidamente.
- **RF6: Notifiche push.** Trentus notifica all'Utente registrato, in caso di servizio "preferito", l'eventuale segnalazione del malfunzionamento di un servizio o se ha ricevuto un numero elevato di segnalazioni, attraverso delle notifiche push.
- **RF7: Commenti.** Trentus permette all'Utente registrato di commentare la propria segnalazione per fornire ulteriori dettagli agli operatori comunali, che così sapranno meglio su cosa concentrarsi per la risoluzione del problema.

Requisiti funzionali per il GdS

- **RF9: Estensibilità.** Trentus permette al GdS di aggiungere un servizio alla lista, specificando nome, link al servizio, aree di possibile problema.
- **RF10: Segnalazioni.** Trentus permette di segnalare l'assenza del servizio o eventuali manutenzioni in anticipo da parte del GdS, specificando quali parti del servizio sono coinvolte e un tempo stimato prima del ritorno in funzione.
- **RF11: Sondaggi su misura.** Trentus richiede che Il GdS metta a disposizione un sondaggio semplice e conciso per far sì che l'Utente possa lasciare una segnalazione precisa riguardo problematiche sul servizio.
- **RF12: Server di ping automatico**
(to fill)

3. Requisiti non funzionali

- **RNF1: Compatibilità:** La nostra WebApp mira ad essere compatibile per ogni browser, ed accessibile sia da telefono che da desktop
- **RNF2: Scalabilità:** La nostra WebApp mira ad essere semplice e facilmente ed economicamente scalabile.
- **RNF3: Modularità:** La nostra WebApp mira ad essere facilmente configurabile, essendo composta da unità indipendenti
- **RNF4: Estensibilità:** La nostra WebApp mira ad essere facilmente estensibile, tale che qualsiasi servizio possa essere aggiunto con facilità
- **RNF5: Affidabilità:** La nostra WebApp mira ad essere affidabile, ponendo la propria fiducia sulla fedeltà degli utenti e sulla costante attenzione del GdS
- **RNF6: Performance:** la nostra webapp mira ad essere lightweight
- **RNF7: Comprensibilità:** la nostra WebApp mira ad avere un'interfaccia semplice ed elegante. L'aggiunta da parte dell'utente dei servizi fattibile con un semplice click. Da parte del GdS una visione delle segnalazioni riguardanti i servizi a colpo d'occhio.
- **RNF8: Facilità di implementazione.** La nostra WebApp mira ad essere implementabile da parte del GdS in modo relativamente semplice: Basterà creare un utente come GdS, dare un nome al servizio, dare il suo DNS, ed altre informazioni. Per l'Utente non c'è nessun setup da fare se non accedere. Nemmeno questo è in realtà richiesto poiché sarà disponibile una lista dei preferiti salvabile attraverso i cookies, solo le notifiche richiederanno un accesso, come recapito.
- **RNF9: Costo minimo.** La nostra WebApp non dovrebbe avere alcun costo se non l'utilizzo di un server da usare come sito e database (la cui potenza ovviamente scalerà con il numero di utilizzatori) ed un DNS. (note sul ping automatizzato?)
- **RNF10: Minimo sforzo di utilizzo.** La nostra WebApp mira ad essere facilmente implementabile. Dal lato utente basta selezionare i preferiti e registrarsi nel caso si vogliano notifiche in tempo reale. Per inviare segnalazioni basterà cliccare sul servizio e compilare un semplice form. Dalla parte del GdS invece basterà tenere aperta la finestra sul browser. Verranno inviate notifiche nel caso in cui venga superata una certa soglia di notifiche in un determinato lasso di tempo. Per segnalare un disservizio basterà premere un bottone e scriverne il motivo. L'unica azione che potrebbe richiedere del tempo è la creazione del sondaggio, ma è necessaria solo una volta, durante il setup del servizio.

4. Design Front-end