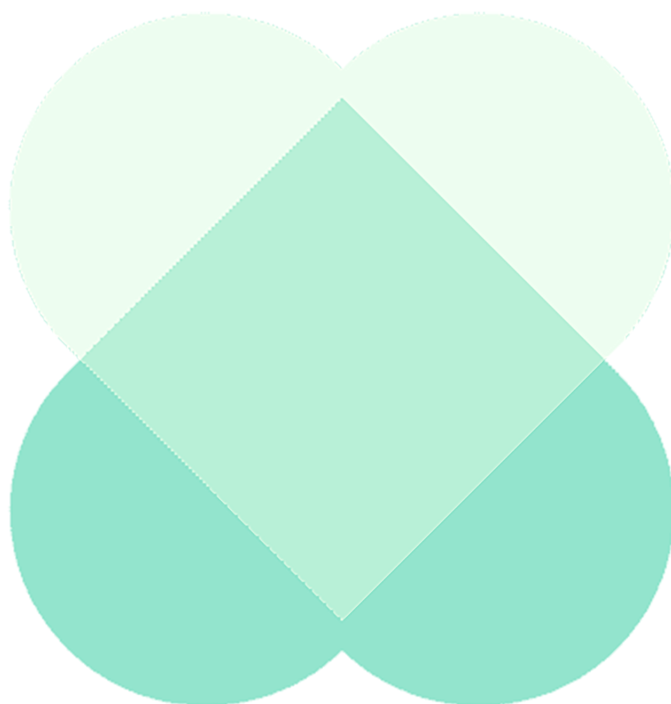


I.T.S. "L. DA VINCI"
Borgomanero



Care4U

Elaborato: "CEROTTO"

Sacco Giacomo

5^B Informatica

A.s. 2020/21

Indice

Analisi del problema	2
Analisi generale	2
Utenti	2
Ipotesi aggiuntive	3
Passaggi implementativi di particolare interesse	4
Schema di funzionamento dell'applicazione	4
MVC (Model View Controller)	4
Routing	5
Autenticazione	5
Supporto fisico (NFC)	6
Responsive CSS	6
Database	7
Modello concettuale	7
Associazioni particolari	7
Modello logico	8

Credenziali d'accesso

	email	password
Admin	admin@gmail.com	admin
Doctor	doctor@gmail.com	doctor
Patient	patient@gmail.com	patient

link: <https://care4u.altervista.org/>

Analisi del problema

Analisi generale

L'obiettivo dell'applicativo è quello di fornire un servizio utilizzabile da medici e pazienti, per ottimizzare lo scambio dei risultati delle misurazioni, eseguite tramite l'utilizzo di un particolare cerotto. Questo cerotto contiene dei reagenti che al contatto con il sudore della pelle assumono diverse colorazioni in base ai livelli di ph, cloruri, acido lattico e glucosio della persona.

Insieme al cerotto il paziente possiede un pass NFC che se avvicinato ad uno smartphone gli consente di accedere automaticamente alla piattaforma.

Utenti

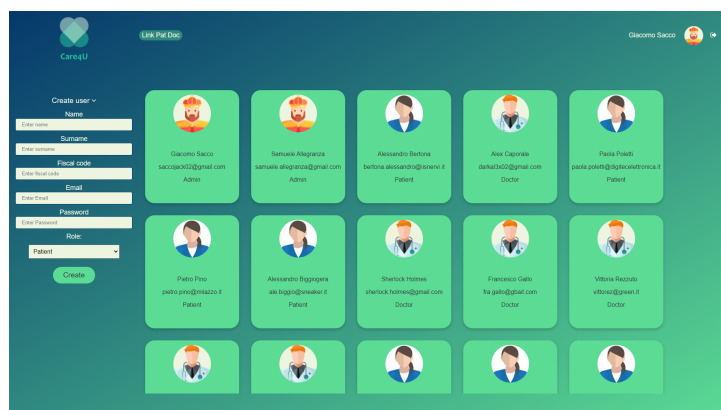
L'applicativo è suddiviso in tre diverse interfacce, una per gli amministratori, una per i medici e una per i pazienti.

Ad amministratori e medici, per accedere alla piattaforma, verranno fornite le credenziali da un amministratore che già possiede l'accesso ad essa, mentre i pazienti potranno registrarsi nell'area apposita inserendo i dati richiesti.

Una volta che l'utente riesce ad accedere all'interfaccia può svolgere diverse attività in base al ruolo corrispondente all'account con cui ha eseguito l'accesso.

Amministratore:

- creare nuovi utenti di tutti e tre i tipi
- visualizzare gli utenti
- gestire le relazioni tra medico e paziente creandone di nuove o rimuovendo quelle già presenti





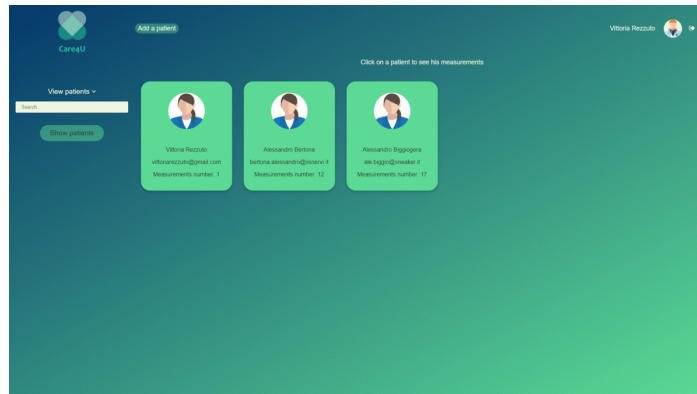
Sacco Giacomo

Care4U

5^B Info

Medico:

- visualizzare le misurazioni dei suoi pazienti
- aggiungere nuove persone alla lista dei suoi pazienti



Paziente:

- inserire nuove misurazioni
- visualizzare lo storico delle misurazioni passate



Ipotesi aggiuntive

Durante la realizzazione dell'elaborato ho analizzato anche le seguenti ipotesi ma per questione di tempo richiesto oppure in quanto si allontanavano dall'effettivo fine del progetto non ho realizzato:

- Controllo SQL injection
- Chat tra medico e paziente
- Algoritmo per il riconoscimento dei colori dei reagenti tramite una foto
- Personalizzazione account
- Email di conferma
- Visualizzazione dei medici di un paziente



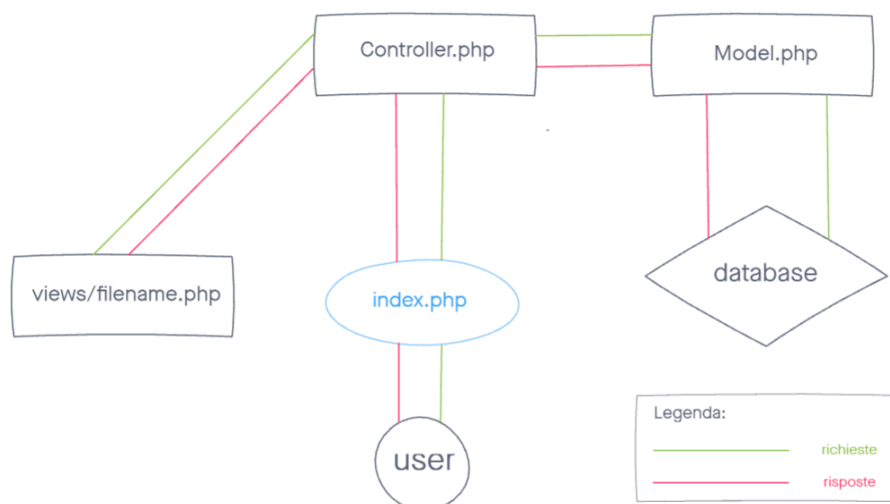
Passaggi implementativi di particolare interesse

Per la realizzazione della piattaforma ho utilizzato il servizio di hosting gratuito di Altervista che mette a disposizione un dominio, un database mySQL ed un server php. L'applicativo è raggiungibile tramite il seguente link: <https://care4u.altervista.org/>

I linguaggi di programmazione o di markup che ho utilizzato sono:

- php: per gestire le dinamiche backend e generare le richieste al database;
- SQL: per effettuare le query al database mySQL;
- JavaScript, HTML e CSS: per gestire le dinamiche frontend, quindi l'effettiva presentazione grafica;

Schema di funzionamento dell'applicazione



MVC (Model View Controller)

Nel processo di sviluppo del codice php ho deciso di seguire lo standard MVC per ottenere una suddivisione quanto più efficace e ordinata del codice.

MVC è suddiviso in tre componenti:

- **View**: comprende tutte le pagine con cui l'utente si interfaccia (principalmente codice HTML).
- **Controller**: è una classe php che riceve le richieste dall'utente tramite una view ed in base alla richiesta richiede a sua volta dei dati al *model* e infine si occupa di visualizzare la view aggiornata con i nuovi dati.
- **Model**: è una classe php che riceve delle richieste dal *controller* e a sua volta effettua delle richieste al database e infine restituisce i dati richiesti al *controller*.

Routing

Per l'accesso alle pagine viene sempre eseguita la pagina *index.php* che si occupa, tramite uno switch, di reindirizzare l'utente alla pagina desiderata. In base al valore della variabile `$_GET['page']`, situata alla fine dell'indirizzo della pagina, lo switch si occupa di richiamare una differente funzione del **Controller**.

Autenticazione

Per gestire il processo di autenticazione ho creato il file *login.php* che offre all'utente la possibilità di accedere utilizzando un account di qualsiasi tipo oppure registrarti come un nuovo paziente.

In entrambi i casi i dati inseriti dall'utente vengono inviati al *controller*, che in base al valore del campo nascosto `$_POST['id']` avvia la procedura di validazione del login, oppure inserisce il nuovo utente nel database.

Login:

```
$user = $this->model->validateLogin($_POST['email'], $_POST['password']);
```

Nuovo paziente:

```
$user = $this->model->addUser(self::PATIENT, $_POST["name"],  
$_POST["surname"], $_POST["fiscalCode"], $_POST["email"], $_POST["password"]);
```

Entrambe le funzioni, in caso di successo, restituiscono un oggetto contenente le informazioni dell'utente; questo oggetto viene successivamente inserito nella variabile di sessione `$_SESSION['user']`.

```
$_SESSION['user'] = $user;
```

Questo oggetto risulta fondamentale in quanto viene utilizzato ogni volta che l'utente richiede l'accesso a una nuova pagina, per verificare se esso è autorizzato a visualizzare quella determinata pagina, in caso contrario viene reindirizzato alla pagina di login.



Supporto fisico (NFC)

Il supporto fisico per il paziente utilizza la tecnologia NFC (Near-field communication). Il compito del pass NFC dato al paziente è quello di reindirizzare lo smartphone direttamente alla pagina di inserimento delle misurazioni.

Per fare ciò ho inserito nei tag NFC, tramite l'applicazione android "TagWriter", il link contenente l'email e la password del paziente come parametri GET.

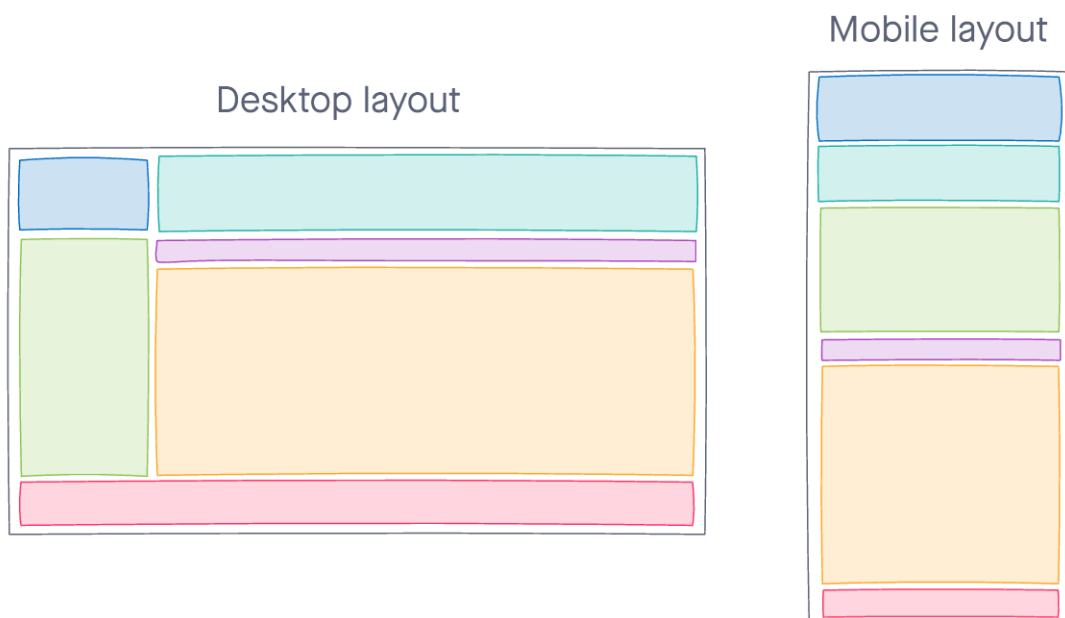
<https://care4u.altervista.org/?page=nfc&email=<email>&password=<password>>

Così facendo i parametri GET *email* e *password* vengono convalidati da una procedura nel *controller* e nel caso in cui le credenziali risultano corrette il paziente viene indirizzato alla pagina di inserimento.

Responsive CSS

L'interfaccia grafica è stata sviluppata per poter garantire l'utilizzo della web app anche tramite dispositivi mobili, come gli smartphone.

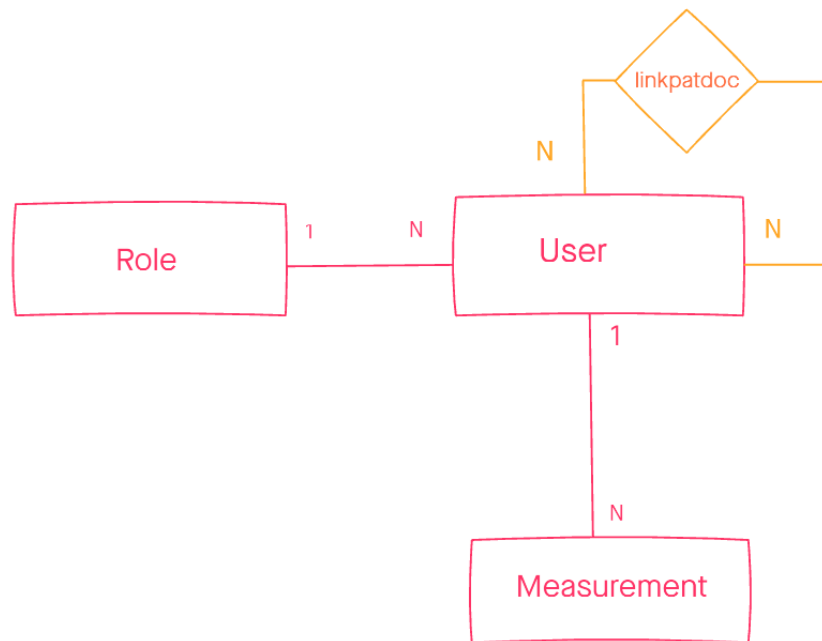
Per ottenere questo risultato ho utilizzato ho sviluppato due diversi tipi di layout utilizzando la funzionalità *grid layout* di CSS.





Database

Modello concettuale



Associazioni particolari

Associazione ISA (1:N): questo tipo di associazione l'ho utilizzata tra le tabelle *Role* e *User* per differenziare gli utenti presenti nella tabella assegnandogli un ruolo. Il punto di favore di questo tipo di associazione è il fatto di poter utilizzare un'unica tabella per tutti i tipi di utente.

Associazione ricorsiva (N:N): questo tipo di associazione è caratteristica perché serve per collegare entità appartenenti ad una stessa tabella. Nel mio caso l'ho utilizzata per creare la tabella ausiliaria *linkpatdoc* che serve per collegare i pazienti ai medici.



Modello logico

Role (**idrol**, role, icon):

idrol (int): chiave primaria

role (char): nome del ruolo assunto da un utente

icon (char): percorso per l'immagine del ruolo

User (**iduse**, codrol, email, password, name, surname, fiscalcode)

iduse (int): chiave primaria

codrol (int): chiave esterna dalla tabella *Role*

email (char): email univoca per ogni utente

password(char[32]): password criptata con MD5

name (char): nome dell'utente

surname (char): cognome dell'utente

fiscalcode (char[16]): codice fiscale dell'utente

Measurement(**idmea**, codpat, ph, chlorides, lactic_acid, glucose, date)

iduse (int): chiave primaria

codpat (int): chiave esterna che si riferisce ad un paziente nella tabella *User*

ph (char[7]): esadecimale del colore della misurazione

chlorides (char[7]): esadecimale del colore della misurazione

lactic_acid (char[7]): esadecimale del colore della misurazione

glucose (char[7]): esadecimale del colore della misurazione

date (timestamp): data e ora in cui è stata effettuata la misurazione

Linkpatdoc(**idlin**, codpat, coddoc)

idlin (int): chiave primaria

codpat (int): chiave esterna che si riferisce ad un paziente nella tabella *User*

coddoc (int): chiave esterna che si riferisce ad un dottore nella tabella *User*