Classe CUtente (simile a CUser)

1) IsLogged()->funzione booleana che ritorna lo stato del login (true=loggato)

usa la sessione per capire se il paziente è loggato

2) isBanned()->usa la sessione per controllare su foundation se il paziente/medico è bannato e poi opera su view e sessione distruggendola

3) login()-> usa la sessione per controllare se siamo loggati, altrimenti ci spedisce sul form di login

4) registrazionepaziente()->usiamo i dati dell’array form per creare il paziente con il controllo sulla mail, lo salviamo e poi passiamo alla view

5)registrazionemedico()->usiamo i dati dell’array form per creare il medico con il controllo sulla mail, lo salviamo e poi passiamo alla view

5) checklogin()->controllo delle credenziali per il login sul db

6) logout()->distrugge la sessione e fa il logout

7) home()-> QUESTO è DA VEDERE MEGLIO COME REALIZZARLO

8) personalProfile()-> Per caricare il profilo del paziente loggato

9) profile($username) -> QUESTO DEVE ESSERE RITOCCATO per far comparire la schermata del medico con gli orari al paziente

10) settings()-> accesso alla schermata per la modifica di alcuni dati del paziente (es. numero di telefono)

11) setUserInfo() -> utilizziamo i dati del form per aggiornare il profilo del paziente

12) setUsername()-> pensiamolo ad un caso di update della mail (controlli vari)

13) setPassword()-> stessa cosa per la password (controlli vari)

14) setProPic()-> DA USARE PER IL MEDICO

15) category($category)->usarlo per caricare tutti i dottori che svolgono un certo tipo di esame

16) explore()->PROBABILMENTE INUTILE

17) followers($idUser) -> MODIFICABILE PER GLI APPUNTAMENTI SVOLTI

Classe CPaziente

[paziente]caso d'uso 1 "prenotazione esame" (passaggio array associativi)

1.1) avviaprenotazione()->passa a view $arraymedici che contiene tutte le informazioni per la visualizzazione degli “esami” prenotabili e $tipologie per inserirle nella tendina

1.2) ricercaesame($IdTipologia)->prende in input l’id della tipologia selezionata nella tendina, e fa la stessa cosa del metodo precedente raffinando i risultati usando la tipologia come parametro aggiuntivo per i medici

1.3) dettagli\_prenotazione($IdMedico,$weekdisplacement)->metodo che passa i dati per la visualizzazione degli orari prenotabili dal paziente in $orari\_disponibilità (array bidimensionale dove il primo indice rappresenta il giorno ed il secondo la fascia oraria),mentre per la visualizzazione dei dati relativi al medico si passa a view l’array associativo $arraymedico

1.4) conferma\_appuntamento($IdMedico)->metodo che prende in input i dati dal form relativi all’Id del medico (campo nascosto) alla data e al numero nello slot orario selezionato dal paziente, a questo punto effettua i controlli necessari e crea l’appuntamento salvandolo nel db

[paziente]caso d'uso 2 "visualizzare un referto"

2.1) visualizza\_appuntamenti\_effettuati()->metodo che prende l’id del paziente dalla sessione e lo usa per ricavarne gli appuntamenti effettuati (quindi con data passata rispetto a quella odierna) i dati alla view vengono passati attraverso $arrayappuntamenti, un array associativo per la visualizzazione di tutti i dati e $tipologie per la scelta nel menù a tendina

2.2) ricerca\_appuntamenti\_effettuati()->metodo che effettua le stesse operazioni del precedente semplicemente raffinando i risultati mediante data e tipologia inserite dal paziente per filtrare i risultati, i dati vengono passati a view nel medesimo modo

2.3) visualizza\_referto($IdAppuntamento)->metodo che prendendo in input l’id dell’appuntamento ne restituisce alla view i dati mediante l’array associativo $arrayreferto per essere utilizzati nella creazione del pdf della visualizzazione

[paziente]caso d'uso 3 "recensire un medico"

3.1) visualizza\_appuntamenti\_effettuati()->presente nel caso d’uso precedente

3.2) accedi\_schermata\_recensioni($IdAppuntamento)->metodo che prendendo in input l’id di un appuntamento, porta il paziente alla schermata per l’inserimento di una recensione al medico relativo, mostrando i campi del form, si passa quindi $arraymedico per la visualizzazione dei dati nella schermata

3.3) conferma\_recensione()->prende i dati con post dalla schermata precedente per creare la recensione e non passa niente alla view

[paziente]caso d'uso 9 "modifica appuntamento"

9.1) visualizza\_appuntamenti\_prenotati()->usando l’id del paziente dalla sessione passa alla view tutti gli appuntamenti non ancora svolti passando l’array associativo $arrayappuntamenti

9.2) dettagli\_appuntamento\_modifica()-> con l’id dell’appuntamento mostriamo le informazioni attuali dell’appuntamento, passando a view $arraymedico e $orari\_disponinibilità

9.3) modifica\_appuntamento() -> utlizzando l’id dell’appuntamento e i dati inseriti nel form si effettuano gli adeguati e controlli e si procede con la modifica effettiva dell’appuntamento

Classe CMedico

[medico]caso d'uso 4 "caricare un referto"

4.1) visualizza\_storico\_appuntamenti()-> questo metodo,usando l’id del medico preso dalla sessione, mostra tutti i suoi appuntamenti conclusi, i dati vengono passati alla view tramite l’array associativo $arrayappuntamenti

4.2) ricerca\_storico\_appuntamenti\_medico()-> questo metodo cambia dal precedente soltanto per la raffinazione dei risultati utilizzando la data

4.3) inserimento\_referto($IdAppuntamento)-> questo metodo mostra la schermata di inserimento del referto, con le adeguate informazioni relative all’appuntamento, passate a view tramite l’array $arrayappuntamento

4.4) caricamento\_referto()->questo metodo prende i campi dal form, li controlla e dunque effettua il caricamento effettivo sul db del referto

[medico]caso d'uso 5 "inserimento orari di disponibilità"

5.1) mostra\_orari\_disponibilità()-> prendiamo l’id del medico dalla sessione ed effettuiamo la visualizzazione degli orari della settimana corrente, passando a view l’array bidimensionale booleano $orari\_disponibilità

5.2) conferma\_orari\_disponibilità()->metodo che prende i valori dal form (la data e i valori delle checkbox) per effettuare la creazione nel db degli orari di disponibilità del medico per poter essere prenotati dai pazienti, a questo punto basta ricaricare la pagina con i dati aggiornati

[medico]caso d'uso 6 "visualizzare lo storico esami di un paziente

6.1) visualizza\_pazienti()->metodo che fa vedere al medico tutti i pazienti non bannati, quindi questi vengono passati a view attraverso l’array $arraypazienti

6.2) ricerca\_pazienti()-> prendendo nome e cognome dal form effettua la ricerca su tutti i pazienti (anche quelli bannati, possibile fix da fare) e li passa con $arraypazienti

6.3) dettagli\_storico\_paziente($IdPaziente)->grazie all’id paziente vengono mostrati al medico tutti gli appuntamenti effettuati dal paziente con la possibilità di visualizzare il referto relativo (DA IMPLEMENTARE manca il passaggio sulla presenza)

6.4) visualizza\_referto($IdAppuntamento) -> metodo che passa i dati alla view per la visualizzazione del referto tramite pdf

[medico]caso d'uso 7 "visualizzare esami complessivi e guadagni in un periodo di tempo

7.1) visualizza\_statistiche()-> porta semplicemente alla pagina senza passare alcun dato

7.2) calcola\_statistiche()->metodo che prende dal form l’intervallo di date e restituisce le statistiche alla view relative alle singole statistiche attraverso l’array $appuntamenti, mentre le statistiche complessive attraverso le variabili $sommacosti e $numappuntamenti

[medico]caso d'uso 8 "visualizzare e rispondere ad una recensione"

8.1) visualizza\_recensioni()-> metodo che restituisce alla view tutte le recensioni del medico loggato attraverso l’array $recensioni

8.2) dettagli\_recensione($IdRecensione)->metodo che prendendo in input l’id di una recensione la restituisce insieme alla sua risposta alla view per la visualizzazione tramite le variabili $recensione e $risposta

8.3) inserisci\_risposta()->metodo che prendendo i dati dal form crea la risposta alla recensione e la salva nel db

[medico]caso d'uso 9 "modifica appuntamento"

9.1) visualizza\_agenda()->metodo che restituisce gli appuntamenti futuri del medico loggato prendendo il suo id dalla sessione, passa a view l’array $agenda

9.2) dettagli\_appuntamento\_modifica($IdAppuntamento)->prendendo in input l’id dell’appuntamento selezionato passa alla view i dati per la modifica attraverso l’array $orari\_disponibilità e le due variabili $nominativopaziente e $vecchiadataeora

9.3) modifica\_appuntamento()->prendendo i dati dal form, con i controlli adeguati modifica l’appuntamento cambiandone lo slot fascia oraria e salvando i cambiamenti sul db

Classe CAmministratore

[admin]caso d'uso 10 "moderazione account medici e pazienti"

10.1) visualizza\_medici()->prende tutti i medici e li passa alla view con l’array $medici

10.2) ricerca\_medici()->prende i valori nei campi del form e li utilizza per fare la ricerca per restituire l’array $medici filtrato

10.3) moderazione\_medico($IdMedico)-> inverte lo stato di attività del medico selezionato

10.1) visualizza\_pazienti()->prende tutti i pazienti e li passa alla view con l’array $pazienti

10.2) ricerca\_pazienti()->prende i valori nei campi del form e li utilizza per fare la ricerca per restituire l’array $pazienti filtrato

10.3) moderazione\_paziente($IdPaziente)-> inverte lo stato di attività del paziente selezionato

[admin]caso d'uso 11 "gestione appuntamenti"

11.1) gestione\_appuntamenti()->mostra la schermata di tutti gli appuntamenti con l’array $appuntamenti passando anche tutte le tipologie con l’array $tipologie per il menù a tendina

11.2) ricerca\_appuntamenti()->raffina i risultati del metodo precedente utilizzando la data e la tipologia inserite

11.3) dettagli\_appuntamento\_modifica($IdAppuntamento)->metodo che prende in input l’id dell’appuntamento e passa alla view $arraymedico, $arraypaziente e $orari\_disponinibilità per la visualizzazione dei relativi dati

11.4) modifica\_appuntamento()->metodo che prende i dati dal form, effettua i controlli, modifica l’appuntamento e lo salva nel db

[admin]caso d'uso 12 "gestione recensioni"

12.1) gestione\_recensioni()->passa alla view tutte le recensioni esistenti attraverso l’array $recensioni

12.2) ricerca\_recensioni()->raffina i risultati delle recensioni attraverso il nome e cognome del medico presi dal form e ripassa le recensioni filtrate alla view

12.3) elimina\_recensione(IdRecensione)->procede ad eliminare la recensione selezionata ed effettuare il salvataggio sul db