



Università degli Studi dell'Aquila

Documentazione progetto **AuleWeb**

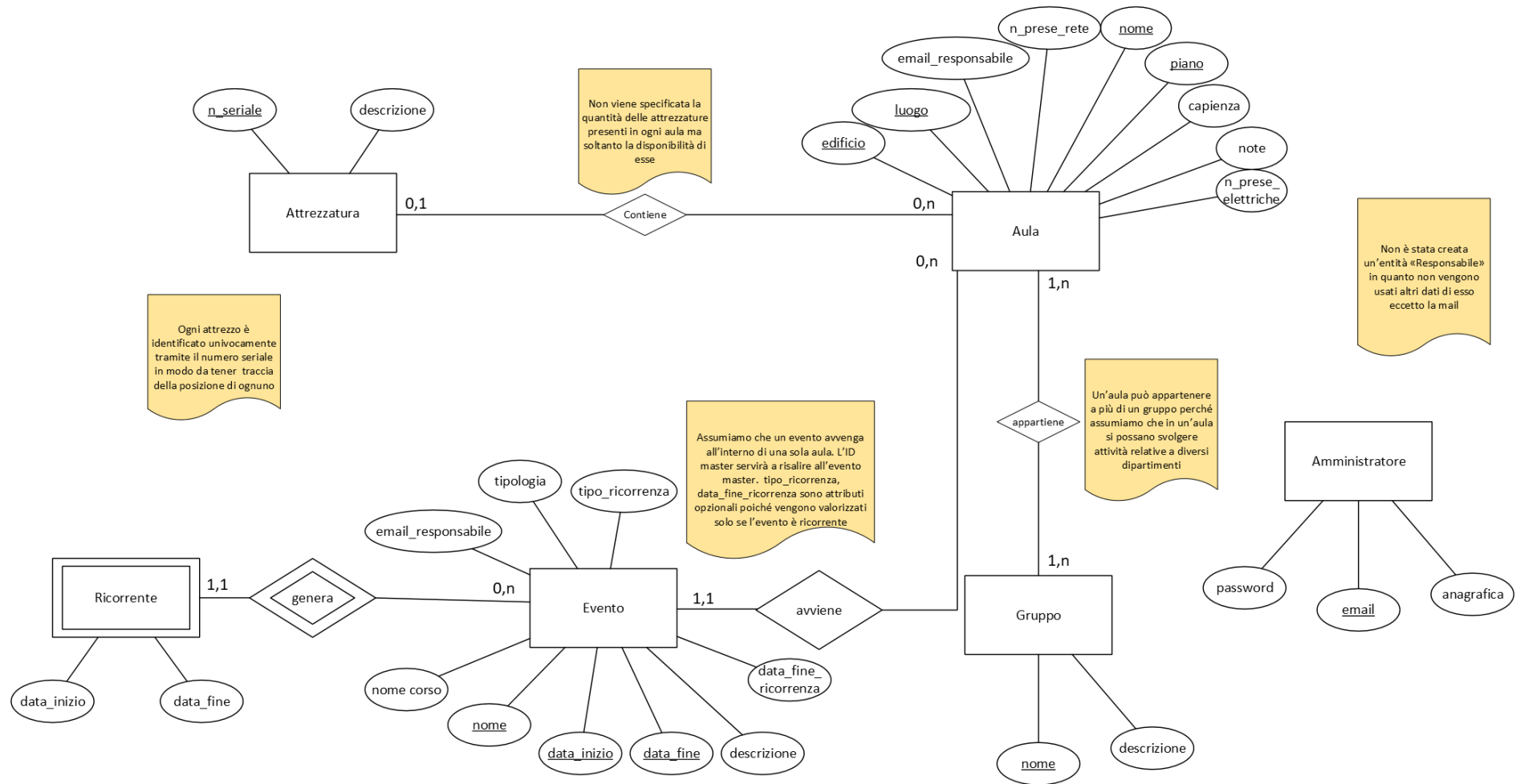
Gruppo *Two Lazy Guys*

Anghel Alex – Croce Marco

Esame di Web Engineering

A.A. 2022-23

Modello E-R



Il database deve permettere la corretta gestione degli eventi e dell'allocazione delle relative aule, difatti sono presenti le entità **Evento** e **Aula**, che rappresentano il “fulcro” dell'intero schema E-R e, quindi, dell'intero database.

L'entità **Evento** possiede gli attributi *data_inizio*, *data_fine*, *nome*, *descrizione*, *email_responsabile*, *ID_aula*, *tipologia*, *nome_corso*, *tipo_ricorrenza* e *data_fine_ricorrenza*, così da avere una rappresentazione “completa” di un evento. La chiave di **Evento** è composta da quattro attributi, ovvero *data_inizio*, *data_fine* e *nome*, al fine di identificare univocamente ogni evento senza “correre rischi”. Nel caso in cui l'attributo *tipo_ricorrenza* non sia NULL allora l'evento avrà associati una serie di eventi ricorrenti, difatti è stata definita un'entità **Evento_Ricorrente** che possiede solo gli attributi *data_inizio* e *data_fine* in quanto per avere una rappresentazione “completa” dell'evento ricorrente è sufficiente risalire all'*evento master* associato e prelevarne i dati. L'entità **Evento_Ricorrente** è debole, difatti la chiave è formata da *data_inizio* e *data_fine*, ma è stata definita una relazione *identificante* tra le due entità così da poter identificare univocamente tutti gli eventi ricorrenti. L'entità **Aula** possiede gli attributi *nome*, *luogo*, *edificio*, *piano*, *capienza*, *email_responsabile*, *numero_prese_rete*, *numero_prese_elettriche* e *note*. La chiave di **Aula** è composta da *nome*, *luogo*, *edificio* e *piano*, così da identificare univocamente anche aule con lo stesso nome ma situate in edifici differenti, situazione che può accadere nei grandi atenei. L'entità **Attrezzatura** possiede gli attributi *numero_seriale* e *descrizione*, la chiave è composta, ovviamente, dal solo attributo *numero_seriale*. L'entità **Gruppo** possiede gli attributi *nome* e *descrizione*, ma la chiave è composta soltanto dall'attributo *nome*. Infine, l'entità **Amministratore** possiede gli attributi *nome*, *cognome*, *email*, *password* e *telefono*, la chiave è composta unicamente dall'attributo *email*.

La relazione **contiene** tra **Aula** e **Attrezzatura** permette di identificare l'attrezzatura presente in una determinata aula e, di conseguenza, l'attrezzatura a cui non è stata ancora assegnata un'aula ed è, quindi, “libera”. La relazione **appartiene** tra **Aula** e **Gruppo** consente di definire l'appartenenza delle aule a determinati gruppi, difatti ogni aula può appartenere a più di un gruppo, mentre ad ogni gruppo appartengono più aule. La relazione **avviene** tra **Aula** ed **Evento** è la più importante, difatti gestisce l'allocazione degli eventi nelle varie aule, ciò per cui è stata progettata la base di dati; quindi, un evento avviene in un'aula specifica, però in una determinata aula possono avvenire più eventi, seppur in intervalli temporali “differenti”. La relazione **genera** tra **Evento** ed **Evento_Ricorrente** permette di risalire agli eventi ricorrenti che sono stati generati da un determinato evento, nel caso in cui l'evento possieda una tipologia di ricorrenza, difatti un evento può generare degli eventi ricorrenti ma un evento ricorrente è generato da un solo evento.

Schema relazionale della base di dati

- **Amministratore** (ID, nome, cognome, email, password, telefono)
- **Evento** (ID, data_inizio, data_fine, nome, descrizione, email_responsabile, ID_aula, tipologia, nome_corso, tipo_ricorrenza, data_fine_ricorrenza)
- **Aula** (ID, nome, luogo, edificio, piano, capienza, email_responsabile, numero_prese_rete, numero_prese_elettriche, note)
- **Gruppo** (ID, nome, descrizione)
- **Gruppo_Aula** (ID_aula, ID_gruppo)
- **Attrezzatura** (ID, numero_seriale, descrizione, ID_aula)
- **Evento_Ricorrente** (ID, data_inizio, data_fine, ID_master)

Legenda:

Relazione (Chiave Primaria, attributi, Chiave Esterna)

Dipendenze Software

- **Client**

Javascript:

bootstrap.bundle.min.js

jquery-3.6.0.min.js

CSS:

bootstrap-icons.css

bootstrap.min.css

- **Server**

Apache Commons Proper CSV - *1.10.0*

FreeMarker Java Template Engine - *2.3.32*

Connettore MySQL-Java - *8.0.18*

Java EE Web API – *7.0*

Funzionalità

Le funzionalità richieste nella specifica sono state realizzate tutte, per alcune di esse sono state effettuate delle piccole “modifiche” in quanto considerate funzionali sia dal punto di vista degli utenti finali sia dal punto di vista della struttura dell’applicazione web.

In particolare, nel “*mostrare gli eventi associati a ciascuna aula in un determinato giorno*” è stato deciso di non mostrare *tutte* le aule (del gruppo) ma soltanto quelle a cui è associato almeno un evento in quel determinato giorno; mentre nel “*mostrare tutti gli eventi attuali e quelli delle prossime tre ore*” un evento viene definito attuale se è iniziato ma non è ancora finito, inoltre vengono mostrati gli eventi la cui ora di inizio è prevista durante le prossime tre ore.

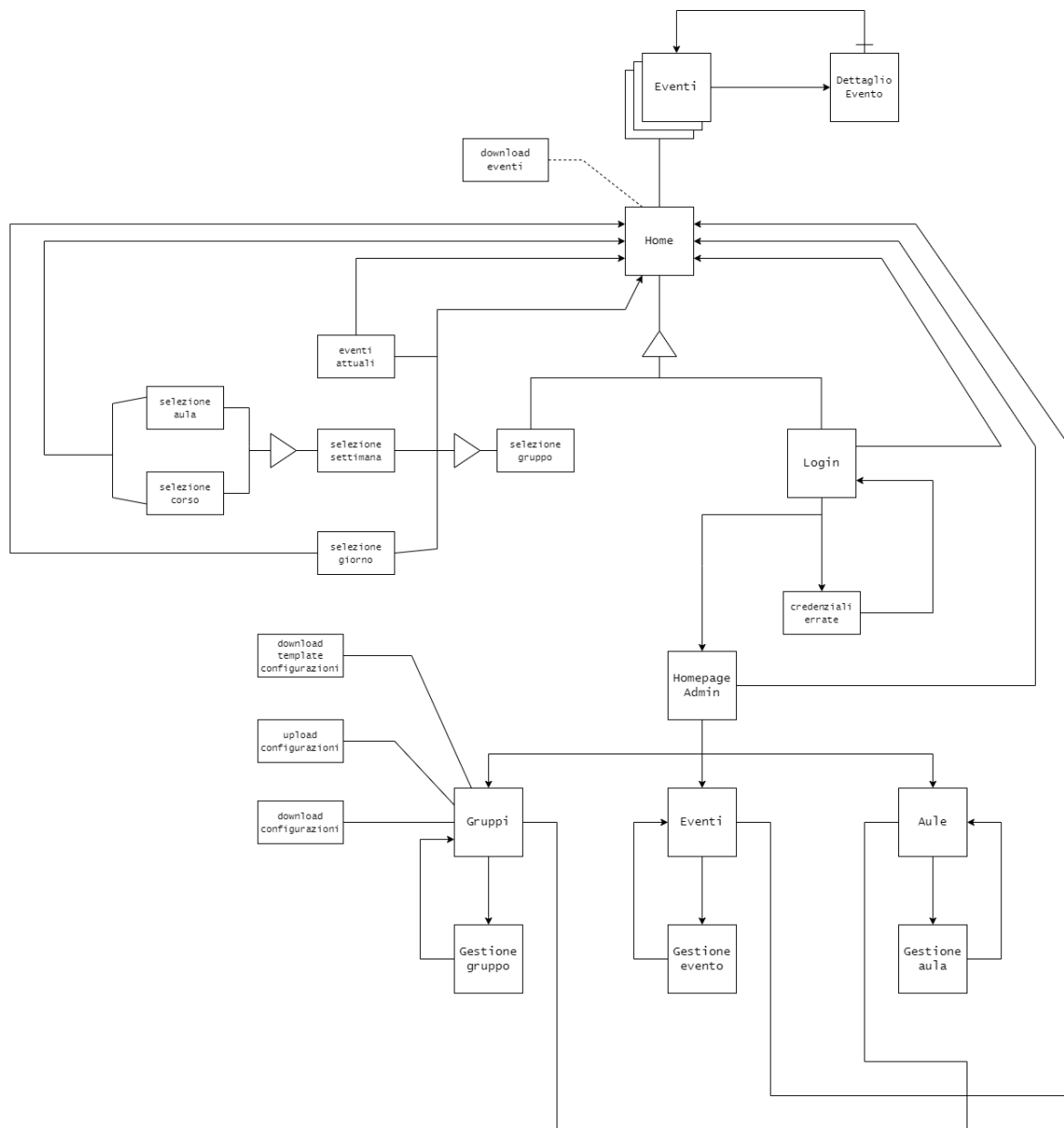
Gli eventi vengono mostrati all’utente tramite una semplice tabella in cui *a colpo d’occhio* è possibile visualizzare le informazioni principali, mentre per visualizzare dettagliatamente tutte le informazioni relative ad un determinato evento è sufficiente cliccare sulla riga della tabella corrispondente e una finestra *pop-up* permetterà di ottenere un dettaglio dell’evento.

Gli utenti hanno la possibilità di esportare gli eventi all’interno di un file .csv, in particolare possono esportare tutti gli eventi relativi ad un intervallo di tempo dalla schermata principale e tutti gli eventi associati ad un determinato corso dalla schermata che viene mostrata, però, soltanto nel momento in cui l’utente richiede la visualizzazione degli eventi associati ad un determinato corso in una specifica settimana.

Gli amministratori, una volta effettuata la login, possono inserire, modificare ed eliminare gli eventi, è stato deciso di non rendere possibile la *gestione* degli eventi ricorrenti ma unicamente del relativo evento *master*, difatti agli amministratori non vengono mostrati tutti gli eventi ma unicamente gli eventi *non ricorrenti* e gli eventi *master*. La gestione degli eventi è stata realizzata seguendo questa logica perché si compensa con i trigger realizzati all’interno del database, che effettuano la gestione degli eventi ricorrenti in maniera “automatizzata”. Inoltre, nel momento in cui un amministratore *modifica* oppure *gestisce* un evento ha massima *libertà*, nel senso che può, ad esempio, selezionare aule che in realtà sono già *occupate*, oppure può specificare, in fase di modifica, un intervallo di tempo per un evento in una determinata aula per cui ci sono eventi già *pianificati*. Ciò viene permesso perché i controlli vengono effettuati lato server tramite i trigger definiti nel database, difatti sono stati definiti numerosi trigger nel database che effettuano un sacco di “controlli” utili al fine di garantire che i dati riguardanti gli eventi e l’applicazione, in generale, siano totalmente corretti.

Gli amministratori, inoltre, hanno la possibilità di inserire e modificare la configurazione delle aule e dei relativi gruppi, ciò può essere realizzato dalla pagina di *gestione* dei gruppi, dove è possibile inserire, modificare ed eliminare un gruppo, ma anche aggiungere oppure rimuovere un’aula; ma anche dalla pagina di *gestione* delle aule, dove è possibile inserire e modificare un’aula, ma anche aggiungere oppure rimuovere attrezzatura e gruppi. In alternativa, dalla pagina di *gestione* dei gruppi è possibile effettuare l’upload (oppure il download) della configurazione in un file .csv, inoltre è stata prevista una funzionalità mediante la quale è possibile effettuare il download di un file .csv che viene fornito con un’intestazione *adeguata* alla configurazione, esso può essere facilmente valorizzato con un’editor per poi usarlo per uploadare la configurazione automaticamente mediante la funzionalità già descritta in precedenza.

Navigation Model



Layout del sito web

Nella parte superiore del sito web è stata definita una navbar, la quale è stata realizzata mediante bootstrap. Sono state realizzate due navbar, la *prima* viene visualizzata dagli utenti, mentre la *seconda* viene visualizzata dagli amministratori nel momento in cui effettuano la login ed entrano nella pagina di amministrazione dell'applicazione web. L'unica pagina nella quale la navbar viene completamente "rimossa" è la pagina di login, difatti viene mostrata unicamente la form con cui gli amministratori possono loggarsi.

Nella navbar degli utenti è presente una *select* tramite la quale è possibile selezionare un gruppo, una volta selezionato sarà possibile per gli utenti "perfezionare" il filtraggio degli eventi, a meno che l'utente non voglia visualizzare tutti gli eventi associati a quel determinato gruppo. Ogni volta che l'utente *perfeziona* il filtraggio degli eventi da visualizzare il gruppo scelto rimane settato per futuri filtri, ciò per rendere l'applicazione maggiormente usabile. Quindi, l'utente può visualizzare gli eventi *attuali* cliccando sull'apposito pulsante; visualizzare gli eventi programmati in un determinato *giorno* cliccando sull'apposito pulsante, ma soltanto dopo aver valorizzato un campo di input di tipo *date*; visualizzare gli eventi programmati in una determinata *settimana* e associati a uno specifico *corso* oppure a una specifica *aula*, ovviamente l'utente potrà selezionare il corso oppure l'aula soltanto in seguito alla valorizzazione di un campo di input di tipo *week*, con cui si andrà a definire la settimana in cui effettuare la ricerca. Le aule e i corsi che vengono mostrati agli utenti appartengono, ovviamente, al gruppo che è stato selezionato in precedenza. Dalla navbar è consentito, inoltre, "aprire" la schermata di login dove è possibile inserire email e password per *loggarsi* all'interno del sistema come amministratore, di conseguenza verrà "aperta" l'area di amministrazione e verrà visualizzato un messaggio di benvenuto per l'amministratore.

Nella navbar degli admin sono presenti quattro *pulsanti* tramite i quali è possibile recarsi nell'area di gestione delle aule, nell'area di gestione degli eventi, nell'area di gestione dei gruppi, rispettivamente, infine è possibile effettuare il logout e tornare alla home.

Nell'area di gestione delle aule è presente una tabella in cui ogni riga rappresenta un'aula definita all'interno del sistema e un pulsante tramite il quale è possibile inserire una nuova aula. Inoltre, in ogni riga della tabella è presente un pulsante tramite il quale è possibile *gestire* l'aula, difatti viene mostrata una nuova pagina con una form per modificarne i dati.

Nell'area di gestione degli eventi è presente una tabella in cui ogni riga rappresenta un evento definito all'interno del sistema e un pulsante tramite il quale è possibile pianificare un nuovo evento. Inoltre, in ogni riga della tabella è presente un pulsante tramite il quale è possibile *gestire* l'evento, difatti viene mostrata una nuova pagina con una form per modificarne i dati.

Nell'ara di gestione dei gruppi, invece, è presente una tabella in cui ogni riga rappresenta un gruppo definito all'interno del sistema e un pulsante tramite il quale è possibile definire un nuovo gruppo. Inoltre, in ogni riga della tabella è presente un pulsante tramite il quale è possibile *gestire* il gruppo, difatti viene mostrata una nuova pagina con una form per modificarne i dati.

Nel “body” del sito web viene visualizzata una tabella al cui interno sono presenti le informazioni di base degli eventi che sono stati *filtrati* in base alle richieste degli utenti. Cliccando su di una delle righe della tabella viene mostrata una finestra *modale* tramite la quale è possibile visualizzare tutte le informazioni relative a quello specifico evento. Gli eventi vengono mostrati *ordinatamente* in base alla data e ora di inizio. Nel caso in cui venga richiesta la visualizzazione di tutti gli eventi programmati in una determinata settimana ed associati a uno specifico corso verrà mostrato anche un pulsante tramite il quale è possibile effettuare il download di un file .csv contenente tutti gli eventi associato al corso selezionato in precedenza.

Nella parte inferiore del sito web è stato definito in maniera “fissa” un *pulsante* tramite il quale è possibile effettuare il download di un file .csv contenente tutti gli eventi programmati in un determinato intervallo di tempo, difatti il file è “scaricabile” soltanto in seguito alla valorizzazione di due campi di input di tipo *date*. Il pulsante viene mostrato soltanto nel caso in cui l’applicazione web si trova nella *home* oppure se sono stati visualizzati gli eventi “filtrati” per gruppo.

Tecnologie utilizzate

Per la realizzazione dell'interfaccia web del progetto è stato utilizzato il noto *framework* **Bootstrap**, così da garantire agli utenti la fruizione di un'interfaccia "comune" rispetto alle altre applicazioni web e che sia user-friendly anche se visualizzata mediante un dispositivo mobile, difatti **Bootstrap** ha, tra i suoi numerosi vantaggi, quello di essere un framework che supporta largamente il *responsive* web design, al fine di garantire che la visualizzazione dell'interfaccia sia ottimale su ogni tipologia di dispositivo. Quindi, **Bootstrap** fornisce *librerie* .css (cascading style sheets) e .js (javascript) specifiche per la realizzazione di determinate "viste" e "controlli", esse sono largamente utilizzate dagli sviluppatori al giorno d'oggi.

Per effettuare delle "azioni" lato client è stata utilizzata una libreria di javascript, ovvero **jQuery**.

Per gestire il caricamento e il rendering delle viste del progetto è stato utilizzato un *template engine* totalmente gratuito basato su Java, *Apache* **FreeMarker**.

Per gestire la lettura e la scrittura di dati all'interno di file .csv (comma-separated values) è stata utilizzata una *libreria* sviluppata da *Apache* all'interno del progetto *Apache Commons*, sezione *Proper*, ovvero **Commons CSV**. *Apache* ha sviluppato la *libreria* con lo scopo di fornire agli utilizzatori del codice che sia indipendente da terze librerie e da eventuali modifiche future, aumentandone, di conseguenza, la stabilità. Inoltre, il codice è totalmente open source.

Per instaurare la connessione tra l'applicazione e il database, al fine di effettuare query e trasferire il risultato dal database al client, è stato utilizzato il *driver* **JDBC**, che funge da "interfaccia" tra l'applicazione *Java* ed il database *MySQL*.

Compatibilità nei browser

Il sito risulta fruibile sui principali browser che vengono utilizzati, usualmente, dagli utenti “target” dell’applicazione web sviluppata. Sono stati comunque riscontrati piccoli bug e malfunzionamenti per alcuni elementi su determinati browser:

- 1) **<input type=“week”>** non è supportato da Firefox, Safari e Safari on iOS, inoltre abbiamo riscontrato un “problema” testando l’applicazione web tramite Chrome, *alcune volte* viene impostata la settimana precedente a quella richiesta dall’utente.
- 2) **autocomplete=“off”** è supportato da tutti i browser ma ha dei bug su Chrome, Chrome Android, Samsung Internet e WebView Android
- 3) **text-overflow** è supportato da tutti i browser ma ha dei bug su Firefox e Firefox for Android
- 4) **margin: auto** non è supportato da Edge in modalità *quirks*
- 5) **<input type=“email”>** è supportato da tutti i browser ma ha un piccolo bug su Safari on iOS

Screenshot delle pagine più importanti del sito

AuleWeb

DISIM

Settimana --, ----

Aula

Corso

Eventi attuali

Visualizza giorno

gg/mm/aaaa

Login

EVENTI DEL DISIM PER IL CORSO DI RICERCA OPERATIVA

Data	Inizio	Fine	Aula	Nome	Descrizione
04-07-2023	09:30	13:00	C1.9	esame parziale di Ricerca Operativa	esame scritto
06-07-2023	14:30	17:30	Digital Class	esame di Ricerca Operativa	esame scritto

Scarica tutti gli eventi relativi al corso

GESTIONE EVENTI

Nome	Data	Inizio	Fine	Aula	Modifica / Elimina
esame di Analisi Matematica	03-07-2023	10:00	13:00	A0.4 - Alan Turing	<button>Gestisci</button>
Lauree di Psicologia	03-07-2023	10:00	13:00	D2.31 - Paride Stefanini	<button>Gestisci</button>
esame di Inglese B2	03-07-2023	11:30	13:30	Digital Class - Alan Turing	<button>Gestisci</button>
esame di Inglese B2	03-07-2023	14:30	16:30	Digital Class - Alan Turing	<button>Gestisci</button>
Inglese B1 - lezione 14	03-07-2023	16:30	18:30	A1.7 - Alan Turing	<button>Gestisci</button>
esame di Basi di Dati	04-07-2023	08:30	12:30	Digital Class - Alan Turing	<button>Gestisci</button>
TOLC-B	04-07-2023	09:00	11:30	D2.29 - Paride Stefanini	<button>Gestisci</button>
esame parziale di Architettura Tecnica	04-07-2023	09:30	11:30	B-0.6 - Blocco B	<button>Gestisci</button>
esame parziale di Fisica	04-07-2023	09:30	11:30	1.1 - Renato Ricamo	<button>Gestisci</button>

DICEAA DIPARTIMENTO DI INGEGNERIA CIVILE, EDILE - ARCHITETTURA E AMBIENTALE

Nome

DICEAA

Descrizione

Dipartimento di Ingegneria Civile, Edile - Architettura e Ambientale

CONFERMA

ELIMINA GRUPPO

Aggiungi un'aula al gruppo

1.1 Renato Ricamo

AGGIUNGI

Rimuovi un'aula dal gruppo

B-0.5 Blocco B

RIMUOVI

[Torna alla gestione dei gruppi](#)

Accedi

Email

Password

[Torna alla Home](#)

NOTA:

Il progetto è stato sviluppato da un gruppo di due persone, in linea di massima il lavoro è stato svolto "insieme", anche se *Marco Croce* si è dedicato maggiormente allo sviluppo del codice in Java, mentre *Alex Anghel* si è dedicato maggiormente alla realizzazione dell'interfaccia grafica.