Link Repository : https://github.com/JosuePugliese/Aeroporto

SCELTE PROGETTUALI TRACCIA NUMERO 3: AEROPORTO

Classi del progetto:

**Utente:**

Rappresenta un generico utente del sistema, per riuscire a separare il ruolo dell’utente generico da quello dell’amministratore.

Attributi: username, password

Metodi: getUsername(), setUsername(), getPassword(), setPassword(),Utente(), loginUtente()

**UtenteGenerico:**

E’ una sottoclasse di Utente che può prenotare voli oppure vedere gli orari di partenze e arrivi senza poter eseguire operazioni dello staff specializzato.

Metodi: aggiornaPrenotazione(), cancellaPrenotazione()

**Amministratore**:

Persona che gestisce i voli inserendone di nuovi e aggiornandoli

Metodi: modificaAssegnazione()

**Volo**:

Rappresenta un singolo volo all’interno dell’aeroporto. E’ associato ad un Gate e a una serie di prenotazioni da parte di vari utenti.

Attributi: idVolo, compagniaAerea, aeroportoDiOrigine, aeroportoDiDestinazione, data, orario, inRitardo, arrivo\_Partenza

**Prenotazione**:

Rappresenta la prenotazione effettuata da un singolo utente di un volo. E’ per questo associata ad uno e un solo volo.

Attributi: nome, cognome, codFiscale, numBiglietto, postoAssegnato

Metodi: cercaEModifica()

**Gate**:

La porta attraverso cui passare per imbarcarsi sull’aereo

Attributo: idGate

**StatoVolo(enumeration):**

Enumerazione che indica se il volo è stato programmato, decollato, cancellato, in\_ritardo , atterrato. Associato alla classe Volo

**StatoPrenotazione(enumeration):**

Enumerazione che indica se la prenotazione è stata confermata, in\_attesa o cancellata. Associato alla classe Prenotazione.

**Controller**:

Gestisce i dati del model e funge da collegamento con la GUI tramite alcuni metodi:

-loginUtente(): controlla se i dati dell’utente sono uguali a quelli inseriti in input

-prenotaVolo(): prenota un volo per l’utente finale.

-modificaVolo():

-aggiungiVolo():

**loginForm (GUI):**

E’ la principale classe per il login utente. Mostra a schermo un form per il login.

Attributi:

-Panel 1 : JPanel

-loginButton: JButton

-textField1: JTextField

-passwordField1: JPasswordField

Metodi:

loginUtente(Controller controller)

**AreaClienti:**

**Area alla quale accedono gli utenti le cui credenziali di login corrispondono a un cliente nel DB.**

**Attributi:**

-panel1: JPanel

-button1: JButton

-button2: JButton

-button3: JButton

-button4: JButton

-button5: JButton

**AreaAmministratore**:

**Area alla quale accedono gli utenti le cui credenziali di login corrispondono a un amministratore nel DB.**

-panel1: JPanel

-MODIFICAButton: JButton

-PRENOTAButton: JButton

-VOLO2TextField: JTextField

-MODIFICAButton1: JButton

-PRENOTAButton1: JButton

Immagine che contiene testo, schermata, Rettangolo, design

Il contenuto generato dall'IA potrebbe non essere corretto.