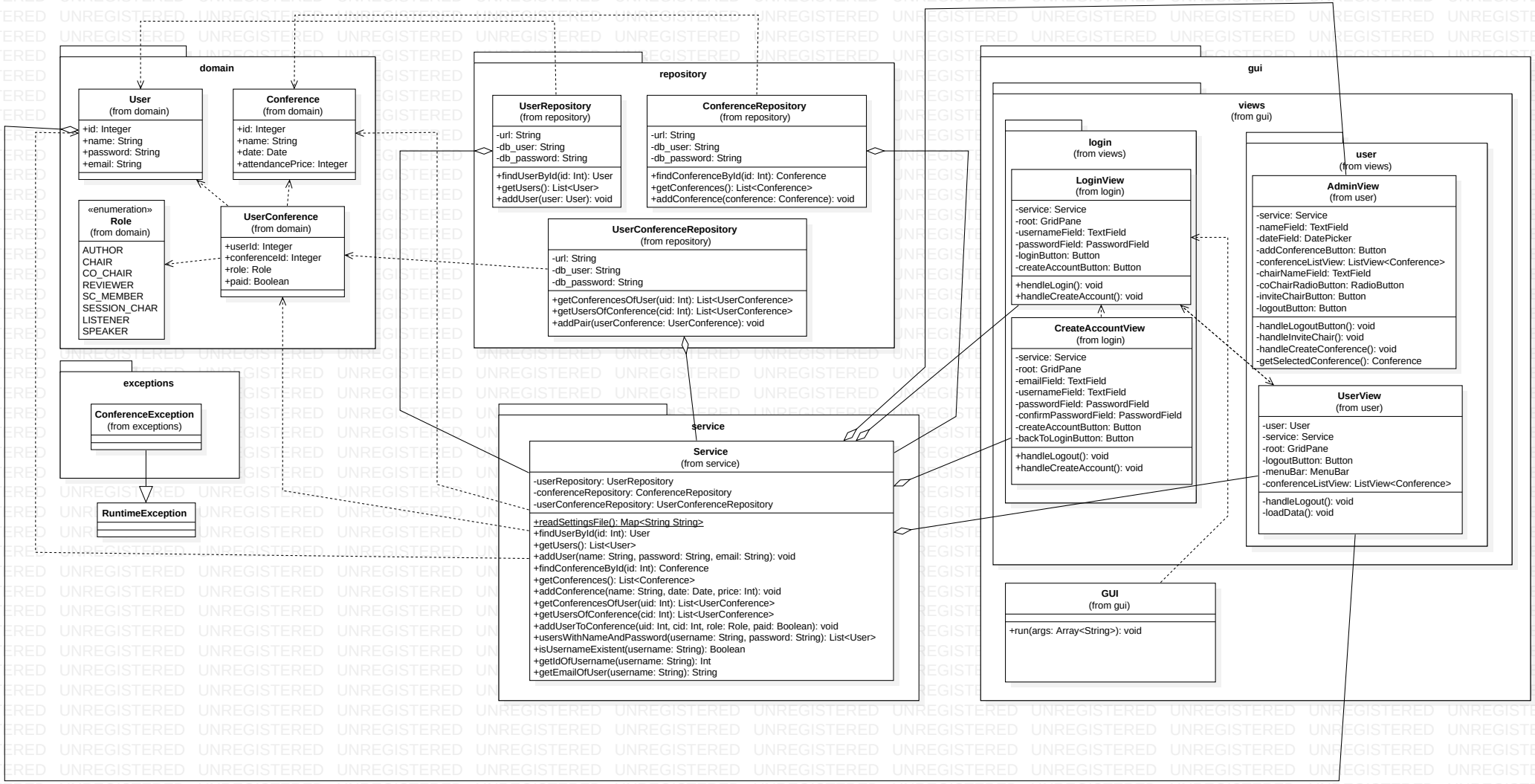
Model.main
domain
User
(from domain)
+id: Integer
+name: String
+password: String
+email: String
Conference
(from domain)
+id: Integer
+name: String
+date: Date
+attendancePrice: Integer
«enumeration»
Role
(from domain)
AUTHOR
CHAIR
CO_CHAIR
REVIEWER
SC_MEMBER
SESSION_CHAR
LISTENER
SPEAKER
UserConference
(from domain)
+userId: Integer
+conferenceId: Integer
+role: Role
+paid: Boolean
repository
UserRepository
(from repository)
-url: String
-db_user: String
-db_password: String
+findUserById(id: Int): User
+getUsers(): List<User>
+addUser(user: User): void
ConferenceRepository
(from repository)
-url: String
-db_user: String
-db_password: String
+findConferenceById(id: Int): Conference
+getConferences(): List<Conference>
+addConference(conference: Conference): void
UserConferenceRepository
(from repository)
-url: String
-db_user: String
-db_password: String
+getConferencesOfUser(uid: Int): List<UserConference>
+getUsersOfConference(cid: Int): List<UserConference>
+addPair(userConference: UserConference): void
exceptions
ConferenceException
(from exceptions)
RuntimeException
service
Service
(from service)
-userRepository: UserRepository
-conferenceRepository: ConferenceRepository
-userConferenceRepository: UserConferenceRepository
+readSettingsFile(): Map<String String>
+findUserById(id: Int): User
+getUsers(): List<User>
+addUser(name: String, password: String, email: String): void
+findConferenceById(id: Int): Conference
+getConferences(): List<Conference>
+addConference(name: String, date: Date, price: Int): void
+getConferencesOfUser(uid: Int): List<UserConference>
+getUsersOfConference(cid: Int): List<UserConference>
+addUserToConference(uid: Int, cid: Int, role: Role, paid: Boolean): void
+usersWithNameAndPassword(username: String, password: String): List<User>
+isUsernameExistent(username: String): Boolean
+getIdOfUsername(username: String): Int
+getEmailOfUser(username: String): String
gui
views
(from gui)
login
(from views)
LoginView
(from login)
-service: Service
-root: GridPane
-usernameField: TextField
-passwordField: PasswordField
-loginButton: Button
-createAccountButton: Button
+hendleLogin(): void
+handleCreateAccount(): void
CreateAccountView
(from login)
-service: Service
-root: GridPane
-emailField: TextField
-usernameField: TextField
-passwordField: PasswordField
-confirmPasswordField: PasswordField
-createAccountButton: Button
-backToLoginButton: Button
+handleLogout(): void
+handleCreateAccount(): void
user
(from views)
AdminView
(from user)
-service: Service
-nameField: TextField
-dateField: DatePicker
-addConferenceButton: Button
-conferenceListView: ListView<Conference>
-chairNameField: TextField
-coChairRadioButton: RadioButton
-inviteChairButton: Button
-logoutButton: Button
-handleLogoutButton(): void
-handleInviteChair(): void
-handleCreateConference(): void
-getSelectedConference(): Conference
UserView
(from user)
-user: User
-service: Service
-root: GridPane
-logoutButton: Button
-menuBar: MenuBar
-conferenceListView: ListView<Conference>
-handleLogout(): void
-loadData(): void
GUI
(from gui)
+run(args: Array<String>): void