***Object Constraint Language***

1. **Player**

**C1:** **context Player inv: email.matches('^[a-zA-Z0-9.\_%+-]**[**+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}**](mailto:+@[a-zA-Z0-9.-]+\.%5ba-zA-Z%5d%7b2,%7d)**)**

**C2 : context Organizer inv : username.matches ( ^[a-zA-Z0-9]+$)**

**C3: context Player inv: (username <> null) and (email <> null )**

**and (password <> null)**

**C4: context Player:: login (username: String, password: Int)**

**pre: username <> null and password <> null**

**post: result = (username = username //in registers**

**and password = password //in registers**

**C5: context Player inv: (name <> null) and (email <> null) and( password <> null) and (phoneNum<> null)**

**C6: context Player:: updateInformation (name: String, email: String,**

**password: int,phoneNum:int)**

**post:**

**self.name = name and**

**self.email = email and**

**self.password = password and**

**self.phoneNume = phoneNum**

**C7: context Player::JoinForTournament(tournament: Tournament)**

**pre: tournament.status = true // "Available"**

**post: player joined in tournament.**

1. **Organizers**

**C8: context Organizer inv: email.matches('^[a-zA-Z0-9.\_%+-]**[**+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}**](mailto:+@[a-zA-Z0-9.-]+\.%5ba-zA-Z%5d%7b2,%7d)**)**

**C9 : context Organizer inv : username.matches ( ^[a-zA-Z0-9]+$] )**

**C10: context Organizer inv: (username <> null) and (email <> null )**

**and (password <> null)**

**C11: context Organizer:: login (username: String, password: Int)**

**pre: username <> null and password <> null**

**post: result = (username = username //in registers**

**and password = password //in registers**

**C12: context Organizer inv: (name <> null) and (email <> null) and( password <> null) and (phoneNum<> null)**

**C13: context Organizer:: updateInformation (name: String, email: String,**

**password: int,phoneNum:int)**

**post:**

**self.name = name and**

**self.email = email and**

**self.password = password and**

**self.phoneNume = phoneNum**

**C14: context organizer ::createTournment (name, description,startDate,endDate,state)**

**pre: Tournament = null //not created**

**post : Tournament created success**

**C15: context Organizer**

**def: canEditTournament(organizer: Organizer, tournament: Tournament): Boolean =**

**organizer.tournaments->includes(tournament)**

**def: canDeleteTournament(organizer: Organizer, tournament: Tournament): Boolean = canEditTournament(organizer, tournament)**

**def: canCreateTournament(organizer: Organizer):**

**Boolean = true**

1. **Admin**

**C16: context Admin:: login (username: String, password: Int)**

**pre: username <> null and password <> null**

**post: result = (username = username //in registers**

**and password = password //in registers**

**C19: context Admin:: removePlayer() : Boolean**

**post: self.removePlayer(player:Player)**

**C20: context Admin:: removeOrganizer () : Boolean**

**post: self.removeOrganizer (organizer: Organizer)**