**­KEY-KNOWLEDGE**



**Object Design Document**

Version 1.0

**TABELLA REVISIONI**

|  |  |  |  |
| --- | --- | --- | --- |
| **DATA** | **VERSIONE** | **DESCRIZIONE** | **AUTORE** |
| **04/11/2020** | **1.0** | **Prima versione del documento Requisiti e casi d’uso** | **Crescenzo Manzone**  **Franco Nicola Fernando**  **Giovanni Battista Mercurio** |

***Package***

***VIEW***

|  |  |
| --- | --- |
| Nome | Descrizione |
| Activity\_main.xml | Home del sistema |
| Home.xml | Home dell’Applicazione |
| Knowledge.xml | Schermata knowledge |
| Login.xml | Area di accesso all’APP |
| Match.xml | Schermata della partita |
| Search\_player.xml | Schermata di abbinamento giocatori |

***CONTROL***

|  |  |
| --- | --- |
| Nome | Descrizione |
| MainControl | Gestione della home |
| KnowledgeControl | Gestione dei dati riguardanti I giocatori |
| LoginControl | Gestione del Log-in |
| MatchControl | Gestione delle partite |
| PairingControl | Gestione degli abbinamenti |

***Interfaccia delle classi***

***USER***

|  |  |
| --- | --- |
| Nome | User |
| Descrizione | Rappresenta un utente registrato nel sistema |
| Attributi | * String nickname * String email * String password * String stato * String ruolo * Int numPartiteVinte * int numPartiteGiocate |
| Signature dei metodi | * getid() * setId() * getMode() * setMode() * getNumQuesiti() * setNumQuesiti() * getPunteggioG1 () * setPunteggioG1 () * getPunteggioG2 () * setPunteggioG2 () * User() |
| Pre-condizioni | **context** User::setPassword(p) **pre**:  p.matches(‘^(?=.\*[0-9])(?=.\*[a-z,A-Z])([a-zA-Z0-9]+){8,64}$’) |
| **context** User::nickName(u) **pre**:  u.matches(‘^[A-Za-z0-9]{​​6,25}​​$’) |
| **context** User:setEmail(e) **pre**:  e.matches(‘\S+@\S+\.\S+’) |
| **context** User:setStato(t) **pre**:  t.matches(‘^[a-z,A-Z]$’) |
|  |
| Post-condizioni |  |
| Invarianti |  |

***QUIZ***

|  |  |
| --- | --- |
| Nome | Quiz |
| Descrizione | Rappresentano i quiz durante le partite |
| Attributi | * int id * String mode * int numQuesiti * user user1 * user user2 * Int punteggioG1 * int punteggioG2 * String status |
| Signature dei metodi | * Quiz() * getNickname() * setNickname() * getEmail() * setEmail() * getPassword() * setPassword() * getStato() * setStato() * getRuolo() * setRuoto() * getNumPartiteVinte() * setNumPartiteVinte() * getNumPartiteGiocate() * setNumPartiteGiocate() * getStatus() * setStatus() |
| Pre-condizioni |  |
|  |
|  |
|  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***QUESTION***

|  |  |
| --- | --- |
| Nome | Question |
| Descrizione | Rappresentano tutte le domande |
| Attributi | * String id * String testo * String risposta1 * String risposta2 * String risposta3 * String risposta4 * String categoria * Int risposta\_esatta |
| Signature dei metodi | * Question() * getId() * setId() * getTesto() * setTesto() * getRisposta1() * setRisposta1() * getRisposta2() * setRisposta2() * getRisposta3() * setRisposta3() * getRisposta4() * setRisposta4() * getCtegoria() * setCategoria() * getRisposta\_corretta() * setRisposta\_corretta() |
| Pre-condizioni |  |
|  |
|  |
|  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***LOGIN***

|  |  |
| --- | --- |
| Nome | Login |
| Descrizione | Rappresenta l’area di accesso all’APP |
| Attributi | * SharedPreferences pref * EditText us * EditText pass * LoginControl control |
| Signature dei metodi | * onCreate() * message() * access() * saveUser() |
| Pre-condizioni |  |
|  |
|  |
|  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***PAIRING***

|  |  |
| --- | --- |
| Nome | Pairing |
| Descrizione | Rappresenta l’abbinamento tra 2 giocatori |
| Attributi | * User user * String mode * Quiz quiz |
| Signature dei metodi | * onCreate() * message() * setQuiz() * onBackPressed() |
| Pre-condizioni |  |
|  |
|  |
|  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***KNOWLEDGE***

|  |  |
| --- | --- |
| Nome | Knowledge |
| Descrizione | Rappresenta la schermata Knowledge |
| Attributi | * User user |
| Signature dei metodi | * onCreate() |
| Pre-condizioni |  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***MATCH***

|  |  |
| --- | --- |
| Nome | Match |
| Descrizione | Rappresenta la partita |
| Attributi | * TextView text * int currentQuestion * Quiz quiz * Question question * MatchControl control |
| Signature dei metodi | * onCreate() * setQuestions() |
| Pre-condizioni |  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***LOGIN MANAGER***

|  |  |
| --- | --- |
| Nome | Login\_manager |
| Descrizione | Modella le interazioni riguardante l’area di log-in |
| Attributi | * SharedPreferences pref * EditText us * EditText pass * LoginControl control |
| Signature dei metodi | * onCreate() * message() * access() * saveUser() |
| Pre-condizioni |  |
|  |
|  |
|  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***MATCH MANAGER***

|  |  |
| --- | --- |
| Nome | Match\_manager |
| Descrizione | Modella le interazioni riguardante le partite |
| Attributi | * String table * int CATEGORIES * int LEVELS * int QUESTIONS * Sring[] categories * Sring[] levels * Sring[] questions * Quiz quiz * DataBaseReference mDatabase * MatchControl control |
| Signature dei metodi | * MatchManager() * getQuestion() |
| Pre-condizioni |  |
|  |
|  |
|  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***PAIRING MANAGER***

|  |  |
| --- | --- |
| Nome | Pairing\_manager |
| Descrizione | Modella le interazioni tra i 2 giocatori |
| Attributi | * int MAX\_ROOMS * int LAST\_ROOM * String TABLE * DatabaseReference mBatabase * PairingControl control1 * MatchControl control2 |
| Signature dei metodi | * PairingManager() * createMatch() * resetMatch() |
| Pre-condizioni |  |
|  |
|  |
|  |
|  |
| Post-condizioni |  |
| Invarianti |  |

***BRIDGE PATTERN***

Il Bridge Pattern permette di separare l’interfaccia di una classe dalla sua implementazione.