player login api service user controller auth.ts user.ts player database Ein Auth Bearer Token wird im Header gesetzt: Authorization: Bearer <token> /user/login body: {"username": name} getUserByName(name) select name, uuid from users where name is name user_names[] Der Benutzer wird aus einem Array von Attributen in ein Array von User-Objekten umgewandelt user_names[User(user_id, name)] [user_names.length >= 1] alt Der Benutzer existiert bereits und muss nicht erneut in der Datenbank registriert werden. Ein JWT-Token wird generiert, mit dem sich der Benutzer authentifizieren kann. generateAccessToken(user) token registerAuthtoken(user.user_id, token) **UPDATE** "user" SET token = \$1 WHERE user_id = \$2; success success alt [success == true] token 200 OK ThrowHTTPException(500) Der Benutzer existiert nicht und muss in der Datenbank registriert werden. Ein JWT-Token wird generiert, mit dem sich der Benutzer authentifizieren kann. registerUser(user_name) insert (name) values (user_name) into users user_name, user_uuid user_name, user_uuid generateAccessToken(user) token registerAuthtoken(user.user_id, token) UPDATE "user" SET token = \$1 WHERE user_id = \$2; success success token 200 OK database player auth.ts user.ts user controller