# Security Use Cases

## Create User

API URL: /apcffl-api/account/accountCreation

NOTE: email1 field must be unique as a primary email across all users

Inputs:

email1 : mandatory (primary email can only be used once as a primary email)

email2: optional

email3: optional

firstName: mandatory

lastName: mandatory

userName: mandatory (must be unique)

password: mandatory

Outputs:

Status message

Processing steps

1. Create user record as user group GUEST
2. Create owner record (with no league affiliation)
3. Send welcome email to the email1 (primary) to the newly registered user
4. Send a notification to the apcffl email notifying a user was registered

## UserName Recovery

API URL: /apcffl-api/security/userNameRecovery/email/{email1}

Where: {email1} is the primary email

Outputs:

Status message

Processing stops:

1. Retrieve the userName for the provided primary email (email1)
2. Send the email to the primary email with the username

## Password Reset

## Step 1: (request a password reset token)

API URL: /apcffl-api/security/passwordResetToken/userName/{userName}

Where: {username} is the user name

Processing Steps:

1. Retrieve the primary email address (email1) for the username
2. Generate a password reset token and set a lifespan on it
3. Email the password reset token to the primary email

## Step 2: (reset the password)

API URL: /apcffl-api/security/passwordReset

Inputs:

userName: mandatory (user name)

passwordResetToken: mandatory (from the email)

password: mandatory (the new password)

Outputs:

Status message

# Account Use Cases

## Login

API URL: /apcffl-api/security/login

Inputs:

userName: mandatory

password: mandatory

Outputs:

userName

userGroupName

securityToken

leagueName

teamName

## Assign League to Account

Prerequisites:

User is logged in

NOTE: Only an ADMINISTRATOR may perform this

This flow assigns a new user to a league, and changes their user group from GUEST to TEAM\_OWNER, and sets the activeFlag to true.

The account is assigned a team with the given team name.

## Step 1: Retrieve all available leagues

API URL: /apcffl-api/league/allLeagues

Inputs:

securityToken: mandatory

username: mandatory

userGroupName: mandatory (ADMINISTRATOR required)

Outputs:

List of leagues:

leagueName

numberOfDivisoions

numberOfTeams

## Step 2: Retrieve all available users

API URL: /apcffl-api/account/accountRetrieveAll

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (ADMINISTRATOR required)

Outputs:

List of accounts:

leagueName

teamName

firstName

lastName

email1

email2

email3

activeFlag

## Step 3: Assign user account to a league

API URL: /apcffl-api/account/ownerLeagueAssignment

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (ADMINISTRATOR required)

ownerUserName: mandatory (the owner userName to update)

ownerLeagueName: mandatory (the league to assign the owner to)

ownerTeamName: mandatory

Outputs:

Status message if an error occurred. Otherwise no message

## Update Account

Prerequisites:

User is logged in

NOTE: Only the account owner and a registered TEAM\_OWNER may perform this

NOTE: email1 field must be unique as a primary email across all users

API URL: /apcffl-api/account/accountUpdate

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (TEAM\_OWNER or ADMINISTRATOR required)

firstName: mandatory

lastName: mandatory

email1: mandatory (primary email)

email2: optional

email3: optional

Outputs:

Error message if an error occurs, no error message if successful

# Communications Use Cases

## Create Message Board Entry

Prerequisites:

User is logged in

NOTE: A message can only be created on the board of the team owner’s assigned league

NOTE: Team owner must be of user group: TEAM\_OWNER or ADMINISTRATOR

API URL: /apcffl-api/communications/newMessageBoard

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (TEAM\_OWNER or ADMINISTRATOR required)

leagueName: mandatory

message: mandatory (must be between 1 – 250 characters)

Outputs: (all messages on the board for this league within the past 7 days)

userName

userGroupName

leagueName

teamName

message

createDate

## Retrieve Message Board

Prerequisites:

User is logged in

NOTE: Messages can only be retrieved from team owner’s assigned league

NOTE: Team owner must be of user group: TEAM\_OWNER or ADMINISTRATOR

NOTE: if either a startDate or an endDate are not provided then messages for the previous 7 days will be retrieved

API URL: /apcffl-api/communications/allMessageBoard

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (TEAM\_OWNER or ADMINISTRATOR required)

leagueName: mandatory

startDate: optional

endDate: optional

Outputs: (all messages on the board for this league within the past n days)

userName

userGroupName

leagueName

teamName

message

createDate

# League Use Cases

## Assign a team to a division in a league

Prerequisites:

User is logged in

NOTE: Only an ADMINISTRATOR may perform this

## Step 1: Retrieve all available leagues (and associated divisions)

API URL: /apcffl-api/league/allLeagues

Inputs:

securityToken: mandatory

username: mandatory

userGroupName: mandatory (ADMINISTRATOR required)

Outputs:

List of leagues:

leagueName

numberOfDivisoions

numberOfTeams

List of available Divisions

## Step 2: Retrieve all available owners for a selected league

NOTE: if available, return assigned Team and division of owner

API URL: /apcffl-api/league/leagueOwners

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (ADMINISTRATOR required)

ownerLeagueName: mandatory

Outputs:

List of league owners:

firstName: mandatory

lastName: mandatory

email: mandatory (primary email)

userName: mandatory

activeFlag: mandatory (only active ‘true’ are wanted)

teamName: optional (if present)

divisionName: optional (if present)

## Step 3: Re-assign Division for Team for Owner

NOTE: A team cannot be created for an owner that has an existing one

NOTE: A team cannot be assigned to a division of no team name exists

NOTE: No more than 4 teams may be assigned per division

NOTE: The UI will allow multiple teams to be assigned at one time

API URL: /apcffl-api/league/teamsDivisionAssignment

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (ADMINISTRATOR required)

List of teams (at least one required):

userName: mandatory (the owner userName to update)

activeFlag: mandatory

leagueName: mandatory (the league of the owner)

teamName: mandatory

division: optional (if null, remove division association)

Outputs:

List of league owners:

firstName: mandatory

lastName: mandatory

email: mandatory (primary email)

userName: mandatory

activeFlag: mandatory (only active ‘true’ are wanted)

teamName: optional (if present)

divisionName: optional (if present)

## Team Owner retrieves roster

Prerequisites:

User is logged in

NOTE: Only a team owner may perform this on their own team

API URL: /apcffl-api/roster/teamRoster

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (TEAM\_OWNER required)

teamName: mandatory

leagueName: mandatory

Outputs:

Team:

userName

teamName

leagueName

roster: (list of teamRoster)

rosterEntry:

schoolName

conferenceName

conferenceAbbr

scholarshipPointValue

## Roster Additions and Subtractions

Prerequisites:

User is logged in

NOTE: Only an ADMINISTRATOR may perform this

## Step 1: Retrieve all active owners’ rosters for a selected league

NOTE: only active owners will be retrieved

NOTE: if an owner does not have an existing roster a blank one will be returned

API URL: /apcffl-api/roster/activeLeagueRosters

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory (ADMINISTRATOR required)

ownerLeagueName: mandatory

Outputs:

List of active league owners rosters:

userName:

teamName:

roster: (list of teamRoster)

rosterEntry:

schoolName

conferenceName

conferenceAbbr

scholarshipPointValue

## Step 2: Retrieve all available conferences (and associated schools)

NOTE: only FBS conferences and schools will be returned at this time

API URL: /apcffl-api/roster/allConferences

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory

Outputs:

List of conferences:

conferenceName

conferenceAbbr

conferenceType

List of schools:

schoolName

conferenceName

conferenceAbbr

## Step 3: Perform roster additions and subtractions

NOTE: The total of all scholarship point values for a single roster <= 200

NOTE: No more than 15 schools may be on a single roster

NOTE: No more than 3 schools from one conference may be on a single roster

NOTE: A school may NOT be on more than one roster at a time

API URL: /apcffl-api/roster/updates

Inputs:

securityToken: mandatory

userName: mandatory

userGroupName: mandatory

List of league owners rosters:

teamOwnerUserName: mandatory

teamOwnerTeamName: mandatory

roster: (list of teamRoster) (mandatory)

rosterEntry:

schoolName (mandatory)

conferenceName (mandatory)

conferenceAbbr (mandatory)

scholarshipPointValue (mandatory)

schoolAddFlag (mandatory)

true: add school to roster

false: remove school from roster

Outputs:

List of active league owners rosters:

userName:

teamName:

teamRoster: (list of rosterEntry)

rosterEntry:

schoolName

conferenceName

conferenceAbbr

scholarshipPointValue