

# University of Arizona Sports Hub

Team Members:

Mark Roche  
Shabeeb Shah  
Jovan Vance

ECE 373  
Dr. Marefat

## **Project Summary**

The design of this project will implement a sports network application centered on University of Arizona sports teams.

The application will have a Sports section where a user can view the different sports that the university offers. The sports included will be Basketball, Football, Baseball, etc. Each sport will have a game schedule, a corresponding team roster, a coach, a corresponding team to each sport, and a venue where each sport is played.

A Team section will exist where you can find information specific to a Team such as team schedules, coaching staff, and rosters. Team schedules will be a list of games, coaching staff will be a list of coaches, and rosters will be lists of players on the team.

The application will have four types of People, People who use the website that can read and comment on articles in the application, Program Administrators who can update information in the application and can also read and comment on articles, Athletes who play a sport, have personal profiles, and who can also use the application, and finally Coaches who also have personal profiles and can interact with the application.

Venues will also be a part of the application where each Venue can be paired with a different sport and information about a specific venue can be found.

The application will also contain Games where information regarding a specific game can be found. Games will either be games that have been played or games that are in the future. The class will have information on the final score, the venue, the home team, the away team, the date, and information on where to listen/watch the game.

The app will feature different media for the users to read. Media can be blogs or articles. Blogs and articles about the various teams can be seen by anyone (not exclusive to users), however only users can write and comment on blogs. Articles must be imported by the admins and cannot be written by anyone on the app (they are from other sources, since the app is meant to be a compilation of all things UA Sports, including any articles related to UA sports written from other sources). Users will be able to keep a list of their favorite articles and blogs.

# People

The classes for people define the people who will be displayed on the website and who can use the app. Typically, players and coaches are displayed on the app, while users and admins are the ones who actually use the website. Players and coaches, however, create an account to use the website if they would like

Athlete
<div>- major: String</div> <div>- highSchool: String</div> <div>- awardsWon: ArrayList&lt;String&gt;</div> <div>- weight: byte</div> <div>- heightFeet: byte</div> <div>- heightInches: byte</div> <div>- number: byte</div> <div>- sport: Sport</div> <div>- position: String</div> <div>- year: String</div>
<div>+ getMajor(): String</div> <div>+ getHighSchool( highSchool: String): String</div> <div>+ getHeight(): byte</div> <div>+ getWeight(): byte</div> <div>+ getSport(): Sport</div> <div>+ getPostition(): String</div> <div>+ getYear(): String</div> <div>+ setMajor( aMajor: String ): void</div> <div>+ setHighSchool( highSchool: String): void</div> <div>+ setHeight(feet: byte, inches: byte): void</div> <div>+ setWeight(pounds: byte): void</div> <div>+ setSport(aSport: Sport): void</div> <div>+ setPostition(newPos: String): void</div> <div>+ displayAwards(): String</div> <div>+ addAward(award: String): void</div> <div>+ setYear(newYear: String): void</div>

Admin
<div>- duties: ArrayList&lt;String&gt;</div>
<div>+ deleteComment(com: Comment): void</div> <div>+ banUser(usr: User): void</div> <div>+ postArticle(article: Article): void</div> <div>+ postBlog(blog: Blog): void</div>

People
<div>- firstName: String</div> <div>- lastName: String</div> <div>- DOB: String</div> <div>- student: boolean</div> <div>- hometown: String</div> <div>- homeState: String</div>
<div>+ getName(): String</div> <div>+ setName(newName: String):void</div> <div>+ getBirthDate():String</div> <div>+ setBirthDate(newBday: String):void</div> <div>+ isStudent(): boolean</div> <div>+ setStudentStatus(status: boolean) : void</div> <div>+ getHometown(): String</div> <div>+ setHometown(city:String, state:String): void</div>

Coach
<div>- title: String</div> <div>- yearsCoached: byte</div> <div>- pastJobs: ArrayList&lt;String&gt;</div> <div>- almaMater: String</div> <div>- honors: ArrayList&lt;String&gt;</div>
<div>+ getTitle(): String</div> <div>+ getYearsCoached(): byte</div> <div>+ getJobs(): ArrayList&lt;String&gt;</div> <div>+ getAlmaMater(): String</div> <div>+ getHonors(): ArrayList&lt;String&gt;</div> <div>+ setTitle(position: String): void</div> <div>+ setYearsCoached(years: byte): void</div> <div>+ addJob(job: String): void</div> <div>+ setAlmaMater( school: String): void</div> <div>+ addHonor(honor: String): void</div>

User
<div>- userName: String</div> <div>- password: String</div> <div>- securityQ: String</div> <div>- teamsFollowed: ArrayList&lt;Team&gt;</div> <div>- blogsWritten: ArrayList&lt;Blog&gt;</div> <div>- email: String</div> <div>- favoriteArticles: ArrayList&lt;Article&gt;</div>
<div>+ getUserName(): String</div> <div>+ getPassword(): String</div> <div>+ getSecurityQ(): String</div> <div>+ getTeamsFollowed(): ArrayList&lt;Team&gt;</div> <div>+ getBlogsWritten(): ArrayList&lt;Blog&gt;</div> <div>+ getEmailAddress(): String</div> <div>+ getFavArticles(): ArrayList&lt;Article&gt;</div> <div>+ setUserName(newName: String): void</div> <div>+ setPassword(setPass: String): void</div> <div>+ setSecurityQ(question: String): void</div> <div>+ addTeam(newTeam: Team): void</div> <div>+ addBlogWritten(newBlog: Blog): void</div> <div>+ setEmailAddress(newAddress: String): void</div> <div>+ addFavArticle(newArt: Article): void</div>

## Team

- sport : Sport
- division : String
- name : String
- roster : ArrayList<Athlete>
- coaches : ArrayList<Coach>
- schedule : ArrayList<Game>

- + getSport() : Sport
- + setSport(Sport) : void
- + getDivision() : String
- + setDivision(String) : void
- + getRoster() : ArrayList<Athlete>
- + setRoster(ArrayList<Athlete>) : void
- + getCoaches() : ArrayList<Coach>
- + setCoaches(ArrayList<Coach>) : void
- + setSchedule(ArrayList<Game>) : void
- + getSchedule() : ArrayList<Game>

## Blog

- author : User

- + addBody(String) : void
- + getAuthor() : User

## Article

- source : String
- author : String

- + getSource() : String
- + setSource(String) : void
- + getAuthor() : String
- + setAuthor(String) : void
- + addBody(String) : void
- + getBody() : String

## Media

- comments : ArrayList<Comment>
- title : String
- body : String

- + setAuthor(String) : void
- + getComments() : ArrayList<Comment>
- + getBody() : String
- + *addBody(String) : void*
- + *getAuthor() : String*

Sport
- schedule: ArrayList<Game> - team: Team - coach: Coach - roster: ArrayList<Athlete> - venue: Venue
+ Schedule() + setSchedule(aSchedule: ArrayList<Game>) : void + detectCollision(aGame: Game): boolean + addGame(aGame: Game): void + getSchedule(): ArrayList<Game> + setTeam(aTeam: Team): void + getTeam(): Team + setCoach(aCoach: Coach): void + getCoach(): Coach + setRoster(players: ArrayList<Athlete>): void + getRoster(): ArrayList<Athlete> + setVenue(aVenue: Venue): void + getVenue(): Venue

Venue
- name: String - schedule: ArrayList<Game> - teams: ArrayList<Team> - capacity: int
+ Venue() + setName(aName: String): void + getName(): String + setSchedule(aSchedule: ArrayList<Game>): void + detectCollision(aGame: Game): boolean + addGame(aGame: Game): void + getSchedule(): ArrayList<Game> + setTeams(Teams: ArrayList<Team>): void + getTeam(): ArrayList<Team> + setCapacity(numPeople: int): void + getCapacity: int

Game
- date: String - startTime: int - home: Team - away: Team - score: int - completed: boolean - venue: Venue - whereToWatch: String
+ Game() + setDate(Date: String): void + getDate(): String + setStartTime(start: int): void + getStartTime(): int + setHomeTeam(Home: Team): void + getHomeTeam(): Team + setAwayTeam(Away: Team): void + getAwayTeam(): Team + setScore(Score: int): void + getScore(): int + isCompleted(): boolean + setVenue(Venue: Venue): void + getVenue(): Venue + setWhereToWatch(Listings: String): void + getWhereToWatch(): String

Window Title

Sign In

Username

Password

Forgot?

Enter

Sign Up

Above: A simple sign in screen for users to sign into the app.  
Below: A simple account creation screen for users that would like to have an account.

Window Title

Username

Password

Confirm Password

Mother's maiden name? (Security Question)

Email

Import Picture

Continue

Below: Account customization; choosing specific teams to follow for a user account.

Window Title

Which teams would you like to follow?

☐

Men's Basketball

☐

Women's Basketball

☒

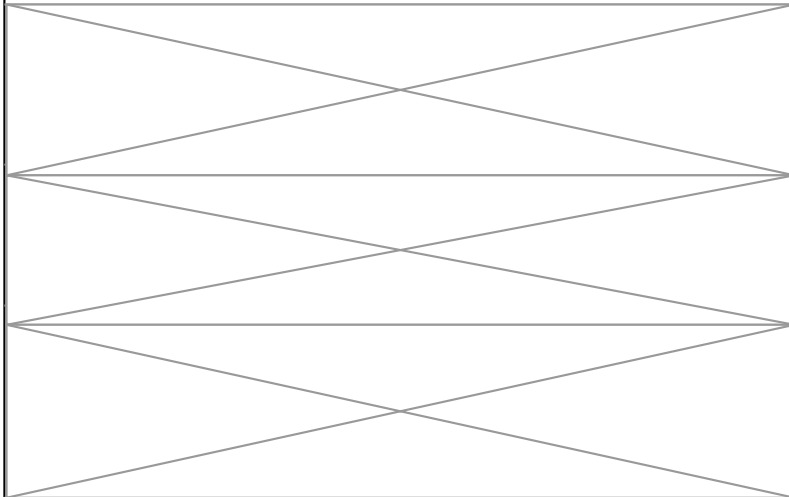
Baseball

☐

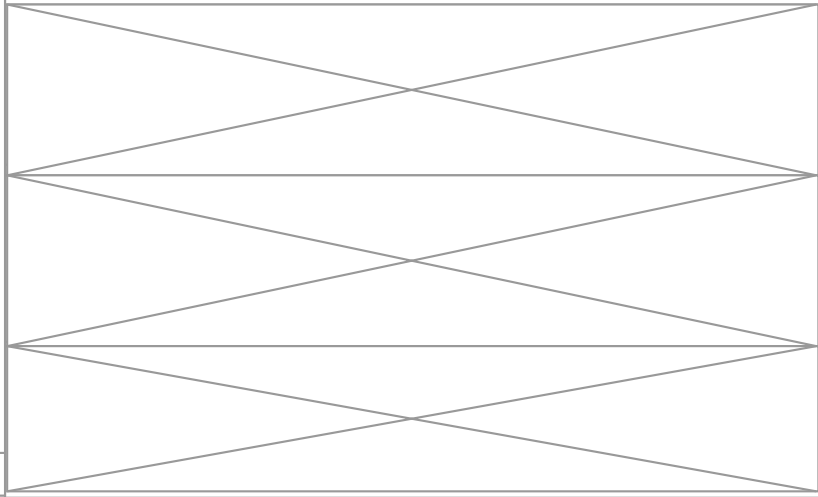
Football



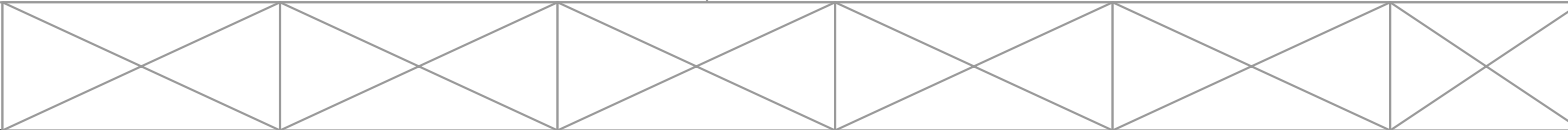
# Top News



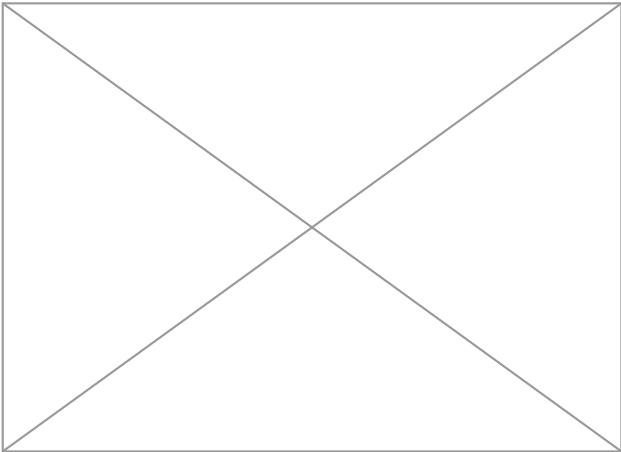
# Featured Blogs



## Top Scores



# TOP RECRUIT GORDON COMMITS TO ARIZONA



Aaron Gordon, fourth ranked recruit in the nation, has announced that he will be attending the University of Arizona next year. Gordon picked Arizona over Kentucky, Washington, and Oregon.

This should be great news for Arizona, who look to go further in the big dance than their Sweet Sixteen run from the previous year. Gordon will be joining the likes of former top recruits Brandon Ashley and Caleb Tarczewski, Nick Johnson, and junior transfer TJ McConnell.

Gordon thinks the Wildcats were the best fit and is eager to play with his future teammates under Head Coach Sean Miller. "I hope to be the X-factor," Gordon said. "I can pretty much do anything they need me to do."

By Mark Roche on 11/17/2013  
Source: ESPN

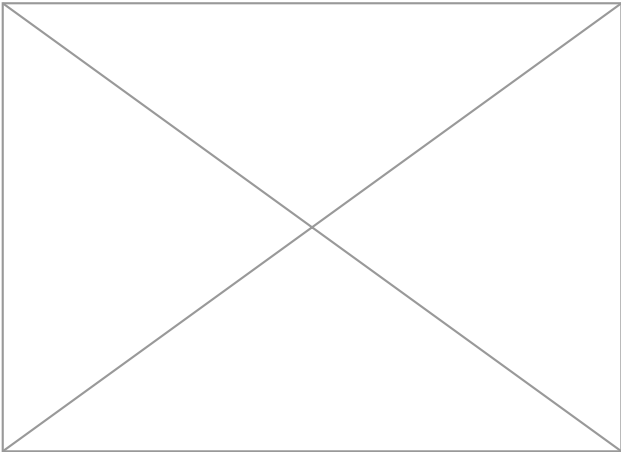
## COMMENTS

Write comment here

- BearDownAZ**  
Yeah baby! U of A going all the way this year!!
- ILoveWilbur**  
I can't wait for this season!!!!

Above picture shows how users see articles and how they can comment on the article. The picture below shows how users will be able to see blog posts and how they can comment.

# HOW AARON GORDON COMPLETES THE TEAM



By Mark Roche on 11/17/2013

As we have recently learned, one of the nation's top recruits is taking his talents to Tucson, Arizona. This news is big enough to make even grown men squeal. During my time here in the Dirty T, the 'Cats have had decent seasons, but have not faired well in the Pac-12 Tournament. Same can be said for the Big Dance.

Aaron Gordon will be the team's saving grace. After Grant Jarrett left the team (dumb, but that is a whole other issue), the team was left not much depth at big men. Not only is Gordon a massive upgrade from Jarrett, but he is versatile on the floor. He can play on the wing or down low, giving Coach Miller some options depending on the situation. He will provide a spark and desire to win to the Cats who can be a bit complacent sometime.

Thank you to the Basketball Gods for blessing us with Aaron Gordon. Now please take us to the promised land which is the National Championship.

## COMMENTS

Write comment here

- BearDownAZ**  
Nice writing man! Keep it up!
- ILoveWilbur**  
I totally agree. Jarrett sucks



## Work Distribution

For the most part, the classes are divided into three categories: People, Sports Related Classes, and App Related Classes (Software). These categories basically translate into three different packages: People, SportRelated, and Software. Each team member will be responsible for writing the classes outlined in the preliminary UML class descriptions. Classes may be added or thought of, and depending on the category of the class, team members will have to work together to get the class written and functioning.

The duties of each team member are as follows:

- Shabeeb Shah
  - Responsible for the SportRelated package
  - Will create classes initially described in UML class descriptions including, but not limited to: Venue, Game, Sport, and Team.
- Mark Roche
  - Responsible for the People package
  - Will create classes initially described in UML class descriptions including, but not limited to: People, Coach, Athlete, User, and Admin
- Jovan Vance
  - Responsible for the Software package
  - Will create classes initially described in UML class descriptions including, but not limited to: Media, Article, Blog, and driver related classes.

All team members will contribute to the construction of a user-friendly GUI. Compared to the construction of the functionality of the app, the gui will be a more collaborative effort so that the assembly of the GUI is easier.

Classes do not have to be exclusively done by one person. The classes will most likely be made in during group meetings, so making all of the classes will be a collaborative effort in the end. The number one goal is to have a hustling and bustling, prime time, all-in-one University of Arizona Sport Hub, so team members will do *whatever* it takes to **get the job done.**